

batman-adv - Bug #248

BATMAN V: interfaces can get stuck in "enabling" state

03/18/2016 01:23 PM - Matthias Schiffer

Status:	Closed	Start date:	03/18/2016
Priority:	Normal	Due date:	
Assignee:	Marek Lindner	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	2016.1		
Description			
The following sequence of commands will cause dummy0 to get stuck in "enabling" state:			
<pre>echo BATMAN_V > /sys/module/batman_adv/parameters/routing_algo modprobe dummy ip link add bat0 type batadv ip link set dummy0 master bat0 ip link set dummy0 up</pre>			
In this situation, ip link set dummy0 down fixes the issue by returning it to "inactive" state. If one tries to remove bat0 while dummy0 is stuck "enabling", or tries to unload batman-adv, a refcounting issue will occur ("unregister_netdevice: waiting for bat0 to become free. Usage count = 3") and a forced reboot is necessary.			

History

#1 - 03/18/2016 09:50 PM - Sven Eckelmann

- Assignee set to Antonio Quartulli

It can be reproduced sometimes(tm) with [Emulation Environment](#) and running as script following commands

```
insmod /host/batman-adv/net/batman-adv/batman-adv.ko
echo BATMAN_V > /sys/module/batman_adv/parameters/routing_algo
sleep 1
ip link add bat0 type batadv
sleep 1
ip link set dummy0 master bat0
sleep 1
ip link set dummy0 up
sleep 1
ip link del bat0
```

Doesn't seem to happen always and sometimes a reboot + running it again is required to reproduce it. I wasn't able to reproduce it with BATMAN_IV

#2 - 03/19/2016 11:15 AM - Sven Eckelmann

Just in case somebody is interested: The first commit which has this problem during a bisect is b05bbab5e1fc ("batman-adv: B.A.T.M.A.N. V - implement neighbor comparison API calls"). But this commit is also the one which makes it possible to select BATMAN_V the first time (so the problem is most likely not hidden in this single commit).

There are still following commits:

```
b05bbab5e1fc ("batman-adv: B.A.T.M.A.N. V - implement neighbor comparison API calls")
bedcadfaa92b ("batman-adv: ELP - send unicast ELP packets for throughput sampling")
5c3245172c01 ("batman-adv: ELP - compute the metric based on the estimated throughput")
cdc4545350d1 ("batman-adv: keep track of when unicast packets are sent")
c513176e4b7a ("batman-adv: add throughput override attribute to hard_ifaces")
667996ebeab4 ("batman-adv: OGMv2 - implement originators logic")
632835348e65 ("batman-adv: OGMv2 - add basic infrastructure")
82da7d6cae2e ("batman-adv: ELP - adding sysfs parameter for elp interval")
2c2fe0ba9ea5 ("batman-adv: ELP - creating neighbor structures")
a4b88af77e28 ("batman-adv: ELP - adding basic infrastructure")
ec46535b8275 ("batman-adv: Add hard_iface specific sysfs wrapper macros for UINT")
```

The interesting part can be seen by modifying the elp code a little bit:

```
diff --git a/net/batman-adv/bat_v_elp.c b/net/batman-adv/bat_v_elp.c
index f257897..aeda5ba 100644
--- a/net/batman-adv/bat_v_elp.c
+++ b/net/batman-adv/bat_v_elp.c
@@ -254,8 +254,10 @@ static void batadv_v_elp_periodic_work(struct work_struct *work)
     goto out;

    /* the interface was enabled but may not be ready yet */
-   if (hard_iface->if_status != BATADV_IF_ACTIVE)
+   if (hard_iface->if_status != BATADV_IF_ACTIVE) {
+       printk("Restart timer %s:%u %u\n", __func__, __LINE__, hard_iface->if_status);
+       goto restart_timer;
+   }

    skb = skb_copy(hard_iface->bat_v.elp_skb, GFP_ATOMIC);
    if (!skb)
```

You can see that it will spam periodically in the kernel log (2 BATADV_IF_INACTIVE, 4 BATADV_IF_TO_BE_ACTIVATED):

```
root@node01:~# /host/testme.sh
batman_adv: B.A.T.M.A.N. advanced 2015.2-134-gb05bbab-dirty (compatibility version 15) loaded
batman_adv: bat0: Adding interface: dummy0
batman_adv: bat0: The MTU of interface dummy0 is too small (1500) to handle the transport of batman-adv packet
s. Packets going over this interface will be fragmented on layer2 which could impact the performance. Setting
the MTU to 1532 would solve the problem.
batman_adv: bat0: Not using interface dummy0 (retrying later): interface not active
Restart timer batadv_v_elp_periodic_work:258 2
Restart timer batadv_v_elp_periodic_work:258 2
batman_adv: bat0: Interface activated: dummy0
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
```

```
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
Restart timer batadv_v_elp_periodic_work:258 4
unregister_netdevice: waiting for bat0 to become free. Usage count = 3
```

```
Message from syslogd@sven-edge at Mar 19 10:08:20 ...
kernel:unregister_netdevice: waiting for bat0 to become free. Usage count = 3
```

The first commit introducing this code was a4b88af77e28 ("batman-adv: ELP - adding basic infrastructure"). But I haven't checked the code in detail. Especially not why it isn't marked as active via `batadv_schedule_bat_ogm`

#3 - 03/19/2016 11:31 AM - Sven Eckelmann

I can see that `batadv_hardif_disable_interface` is called for this interface but it is not removed because it is not yet marked as "BATADV_IF_INACTIVE".

```
--- a/net/batman-adv/hard-interface.c
+++ b/net/batman-adv/hard-interface.c
@@ -571,12 +571,15 @@ void batadv_hardif_disable_interface(struct batadv_hard_iface *hard_iface,
 {
     struct batadv_priv *bat_priv = netdev_priv(hard_iface->soft_iface);
     struct batadv_hard_iface *primary_if = NULL;
+    printk("%s:%u\n", __func__, __LINE__);

     if (hard_iface->if_status == BATADV_IF_ACTIVE)
         batadv_hardif_deactivate_interface(hard_iface);
+    printk("%s:%u\n", __func__, __LINE__);

     if (hard_iface->if_status != BATADV_IF_INACTIVE)
         goto out;
+    printk("%s:%u\n", __func__, __LINE__);

     batadv_info(hard_iface->soft_iface, "Removing interface: %s\n",
                 hard_iface->net_dev->name);
```

```
root@node01:~# /host/testme.sh
batman_adv: B.A.T.M.A.N. advanced 2016.0-94-g77ddf97-dirty (compatibility version 15) loaded
batman_adv: bat0: Adding interface: dummy0
batman_adv: bat0: The MTU of interface dummy0 is too small (1500) to handle the transport of batman-adv packet
s. Packets going over this interface will be fragmented on layer2 which could impact the performance. Setting
the MTU to 1532 would solve the problem.
batman_adv: bat0: Not using interface dummy0 (retrying later): interface not active
Restart timer batadv_v_elp_periodic_work:260 2
batman_adv: bat0: Interface activated: dummy0
batadv_hardif_disable_interface:574
batadv_hardif_disable_interface:578
unregister_netdevice: waiting for bat0 to become free. Usage count = 3
```

This works fine for BATMAN_IV

```
root@node01:~# /host/testme.sh
```

```
batman_adv: B.A.T.M.A.N. advanced 2016.0-94-g77ddf97-dirty (compatibility version 15) loaded
batman_adv: bat0: Adding interface: dummy0
batman_adv: bat0: The MTU of interface dummy0 is too small (1500) to handle the transport of batman-adv packet
s. Packets going over this interface will be fragmented on layer2 which could impact the performance. Setting
the MTU to 1532 would solve the problem.
batman_adv: bat0: Not using interface dummy0 (retrying later): interface not active
batman_adv: bat0: Interface activated: dummy0
batadv_hardif_disable_interface:574
batman_adv: bat0: Interface deactivated: dummy0
batadv_hardif_disable_interface:578
batadv_hardif_disable_interface:582
batman_adv: bat0: Removing interface: dummy0
```

But this is to be expected because the interface is marked as BATADV_IF_ACTIVE. And if this is really the problem then this can be worked around (please check this twice) via:

```
diff --git a/net/batman-adv/hard-interface.c b/net/batman-adv/hard-interface.c
index 3240a67..40e8e5a 100644
--- a/net/batman-adv/hard-interface.c
+++ b/net/batman-adv/hard-interface.c
@@ -572,7 +572,8 @@ void batadv_hardif_disable_interface(struct batadv_hard_iface *hard_iface,
     struct batadv_priv *bat_priv = netdev_priv(hard_iface->soft_iface);
     struct batadv_hard_iface *primary_if = NULL;

-    if (hard_iface->if_status == BATADV_IF_ACTIVE)
+    if (hard_iface->if_status == BATADV_IF_ACTIVE ||
+        hard_iface->if_status == BATADV_IF_TO_BE_ACTIVATED)
         batadv_hardif_deactivate_interface(hard_iface);

     if (hard_iface->if_status != BATADV_IF_INACTIVE)
```

#4 - 03/19/2016 02:15 PM - Sven Eckelmann

Just had a look at the code and it looks like batadv_schedule_bat_ogm is called but the interface is still in mode INACTIVE.

BATMAN_IV does the same but calls it again later with the active interface:

```
batman_adv: bat0: Adding interface: dummy0
batman_adv: bat0: The MTU of interface dummy0 is too small (1500) to handle the transport of batman-adv packet
s. Packets going over this interface will be fragmented on layer2 which could impact the performance. Setting
the MTU to 1532 would solve the problem.
batman_adv: bat0: Not using interface dummy0 (retrying later): interface not active
batadv_hardif_enable_interface:572
batadv_schedule_bat_ogm:434
batadv_schedule_bat_ogm:440 2
batman_adv: bat0: Interface activated: dummy0
batadv_schedule_bat_ogm:440 4
batadv_schedule_bat_ogm:440 3
batman_adv: bat0: Interface deactivated: dummy0
batman_adv: bat0: Removing interface: dummy0
```

"Problem" is here that BATMAN_V doesn't do anything with batadv_v_ogm_schedule. So it doesn't call batadv_schedule_bat_ogm again to active the interface. The interface will therefore forever in the TO_BE_ACTIVATED state and thus "inactive" for batman-adv.

BATMAN_IV instead uses bat_ogm_schedule to queue itself up for additional sends and thus later sets this interface to the state BATADV_IF_ACTIVE. This is done via batadv_send_outstanding_bat_ogm_packet for each batadv_forw_packet (see batadv_iv_ogm_aggregate_new).

The commit adding the problematic empty bat_ogm_schedule was a4b88af77e28 ("batman-adv: ELP - adding basic infrastructure")

#5 - 03/27/2016 09:38 AM - Marek Lindner

The main topic of this issue has been addressed with the patch Sven proposed and I just merged into the next branch. What remains is that once a slave interface is turned off BATMAN V is never able to recover. That is largely due to the fact that BATMAN V does not take advantage of the general queuing mechanism (emit / schedule functions) but implements its own queue. Also missing is the batman_queue_left mechanism.

#6 - 04/13/2016 09:38 PM - Antonio Quartulli

- Assignee changed from Antonio Quartulli to Matthias Schiffer

I just [posted a patch](#) on the mailing list which should fix the problem of the interface getting stuck in "enabling" state. Matthias could you give it a try ?

Thanks!

#7 - 04/18/2016 03:44 PM - Matthias Schiffer

I can confirm that Antonio's patch fixes the issue.

#8 - 04/19/2016 04:49 AM - Antonio Quartulli

- Assignee changed from Matthias Schiffer to Marek Lindner

Marek, I am assigning this ticket to you because the patch is currently pending on the ML. Could you please update this ticket whenever the destiny of the patch is decided? :)

Thanks !

#9 - 04/21/2016 12:18 PM - Marek Lindner

- Status changed from New to Closed

Antonio Quartulli wrote:

Marek, I am assigning this ticket to you because the patch is currently pending on the ML. Could you please update this ticket whenever the destiny of the patch is decided? :)

Merged! thanks for testing Matthias!

#10 - 02/11/2017 08:36 AM - Sven Eckelmann

- Target version set to 2016.1