

batman-adv - Feature #339

Make "batctl log" usable with network namespaces

07/13/2017 05:09 AM - Linus Lüssing

Status:	New	Start date:	07/13/2017
Priority:	Normal	Due date:	
Assignee:	batman-adv developers	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
Currently, this fails as the socket is only available via debugfs right now. And for debugfs we have no namespace support.			

History

#1 - 07/13/2017 05:10 AM - Linus Lüssing

- Tracker changed from Bug to Feature

#2 - 07/13/2017 05:17 AM - Linus Lüssing

I only had a brief look at the network namespace code so far. But it seems to me that we cannot easily create subdirectories in debugfs for per network namespace sockets as the NS name is something specific to ip-netns, userland, mapped in /var/run. In kernelspace it seems PIDs are used internally instead?

I guess, netlink is not suitable to stream log data? Nor is plain printk'ing an option, I suppose. What facilities could be suitable?

#3 - 07/13/2017 09:40 AM - Sven Eckelmann

There was already a patchset from Andrew to create per netns folders in debugfs. So it is at least theoretically possible. I personally would not invest too much time into it because it was already planned to deactivate/remove the batadv debugfs in the future.

- <https://patchwork.open-mesh.org/project/b.a.t.m.a.n./patch/1456867148-31883-5-git-send-email-andrew@lunn.ch/>
- <https://patchwork.open-mesh.org/project/b.a.t.m.a.n./patch/1458157811-15809-1-git-send-email-andrew@lunn.ch/>

Generic netlink itself has a NLA_NUL_STRING attribute type. So it can transport stings. And it has a multicast link mode which you can use to send messages to all interested clients. But something tells me that we don't want to use for our logging. But (pure) netlink was already used for audit.c

None of the solutions seems to be perfect. Just wanted to provide my 2 cents

#4 - 02/27/2018 11:57 PM - Linus Lüssing

Another option I was wondering about: Dumping the logs as UDP packets to an IPv6 node-local multicast address on the according batX interface.

When using node-local multicast then we could probably mostly disregard the Linux IPv6 stack and skip the IPv6 Neighbor Discovery Protocol part (duplicate address detection etc.). And just preallocate an skb to copy with a static IPv6 source and destination address and UDP ports.

Probably quite "unconventional" but since we are in networking... why not use our own, flexible and fast infrastructure?