

FreeNAS - Feature #74721

Add ng_* modules to the kernel

02/11/2019 12:11 PM - Brandon Schneider

Status: Done	Estimated time: 0.00 hour
Priority: No priority	
Assignee: Ryan Moeller	
Category: OS	
Target version: 11.3-ALPHA1	
Severity: New	Needs Merging: No
Reason for Closing:	Needs Automation: No
Reason for Blocked:	Support Suite Ticket: n/a
Needs QA: No	Hardware Configuration:
Needs Doc: No	

Description

Currently we lack some like ng_bridge and ng_iface, iocage is growing netgraph support, and we may possibly be using ng_nat for NAT in iocage. So we'll need these available.

Assigning to Ryan per discussion with Mav

History

#1 - 02/11/2019 02:32 PM - Ryan Moeller

- Status changed from Unscreened to In Progress

Brandon: I think netgraph is pretty neat, so this is very exciting :)
To confirm, you are asking for *all* the netgraph modules (ng_*)?

#2 - 02/11/2019 04:31 PM - Ryan Moeller

Taking a closer look, most of the netgraph modules probably are not too useful for jails.

The list I have narrowed it down to is:

- bridge
- eiface
- ether (already a module)
- iface
- nat
- netgraph (already a module)
- socket (already a module)

Are these better as modules or built in to the kernel?

There are several others that could be interesting to have (bpf, ipfw, device, one2many, vlan, source, pipe, ...), but they are not generally applicable to the jails use case.

#3 - 02/11/2019 04:39 PM - Alexander Motin

While I liked NetGraph very much 10 years ago, these days it has performance problems due to very fine-grained locking. That is why I am not sure I'd like to see it as replacement for `if_bridge` or `ipfw nat`, at east without thinking twice.

What's about static linking into the kernel, I'd ask to avoid that unless there is very good reason for that. Adding few more modules cost us only some megabytes on boot device, which we have, while linking to kernel consumes DTrace type information slots, which we have only 32K and most are already used.

#4 - 02/11/2019 06:32 PM - Ryan Moeller

Ok, modules it is!

You have mentioned the locking problem before, I remember that. And I've previously done some measurements with mixed results. Some scenarios can saturate 10 Gbits, others max out around 3-4 with heavy CPU usage in the receive queue, and my measurements of `ng_bridge` to `ng_ether` on a tap for `bhyve` were pathetic. I haven't looked further in to how much of this is due to different hardware or different software versions or tap. I don't have written down how well `ng_bridge` does locally, either.

I sent a question to BSD Now several months ago asking about netgraph and why it fell out of use, but either I missed it or they didn't respond yet. Mostly I had hoped to spark interest in the audience so maybe performance optimizations would get discussed.

#5 - 02/11/2019 06:42 PM - Ryan Moeller

master PR: <https://github.com/freenas/build/pull/173>

#6 - 02/11/2019 07:49 PM - Ryan Moeller

- Status changed from *In Progress* to *Ready for Testing*

- Needs Merging changed from *Yes* to *No*

#7 - 02/12/2019 09:39 AM - Ryan Moeller

- Status changed from *Ready for Testing* to *Passed Testing*

#9 - 03/26/2019 09:39 AM - Dru Lavigne

- Status changed from *Passed Testing* to *Done*

#10 - 03/26/2019 10:08 AM - Dru Lavigne

- Needs QA changed from *Yes* to *No*

- Needs Doc changed from *Yes* to *No*

#11 - 04/02/2019 08:23 AM - Dru Lavigne

- Target version changed from *11.3-BETA1* to *11.3-ALPHA1*