

---

Subject: Re: Re: [patch 0/2][NETNS49][IPV4][IGMP] activate multicast per namespace

Posted by [ebiederm](#) on Mon, 15 Oct 2007 18:03:22 GMT

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

---

Daniel Lezcano <dlezcano@fr.ibm.com> writes:

> Denis V. Lunev wrote:

>> Daniel Lezcano wrote:

>>> Eric W. Biederman wrote:

>>>> Daniel Lezcano <dlezcano@fr.ibm.com> writes:

>>>>

>>>>> The following patches activate the multicast sockets for  
>>>>> the namespaces. The results is a traffic going through different  
>>>>> namespaces. So if there are several applications  
>>>>> listenning to the same multicast group/port, running in  
>>>>> different namespaces, they will receive multicast packets.

>>>>

>>>> At a first glance this feels wrong. I don't see any per  
>>>> namespace filtering of multicast traffic. Unless the  
>>>> multicast traffic is routed/bridged between namespaces  
>>>> it should be possible to send multicast traffic in one  
>>>> namespace and listen for that same traffic in another  
>>>> namespace and not get it.

>>>

>>> The described behavior is the case were the namespaces are communicating via  
>>> veth like:

>>>

>>> eth0

>>> |

>>> | ----- nsA

>>> veth0 <--|--> veth1 |

>>> | -----

>>> |

>>> | -----nsB

>>> veth2 <--|--> veth3 |

>>> | -----

>>>

>>>

>>> If an application is listening in nsA and nsB. And if in nsA, an application  
>>> sends multicast traffic, both will receive the packets because they are  
>>> routed by the pair device.

>>> As you said this is the correct behavior, if we have two machines hostA and  
>>> hostB in the same network and both are listening on the multicast address and  
>>> if an application on hostA send multicast packets, both should receive the  
>>> multicast packets.

>>> If the traffic is not routed, multicast will not pass through the namespaces.

>>>

>>> The description I gave in the patchset introduction was to describe such  
>>> behavior which is, IMHO, important for inter-container communication.  
>>> Perhaps, I should have not gave this description which seems to sow confusion  
>>> in mind, sorry for that.

>>>

>>> Anyway, I hope the patchset is ok :)

>>

>> hmm, by the way, will this work with macvlan?

>

> I will check that. At the first glance, IMO it will not work if the

> IP\_MULTICAST\_LOOP option is not set. Need to check ...

>

>> also, I am dumb with multicasts :) who will clone the packet if there are more

>> than one namespace listen and there are some listeners behind ethernet?

>

> For local delivery, the function is:

>

> \_\_udp4\_lib\_mcast\_deliver

>

> For packet emission and loopbacking packets to ourself, it is:

>

> ip\_mc\_output

>

> For behind ethernet, the packet is multicasted to the network, so it is up the

> peers to manage the packet.

Right. So macvlan last I looked should handle the case of receiving an external multicast transmission from ethernet fine as it replicates the packet. However macvlan currently doesn't replicate the packet to other macvlan devices on packet transmission, because with only a single namespace the kernel will ensure local listeners get their multicast packets by going through the loopback device. The same problem exists for broadcast packets as well. So is likely we want to tweak the macvlan code just a bit before we use it to heavily.

Eric

---

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>

---