Subject: Re: L3 network isolation
Posted by Daniel Lezcano on Thu, 07 Dec 2006 22:33:30 GMT
View Forum Message <> Reply to Message

Vlad Yasevich wrote:
> Hi Daniel
>
>> Hi all,
>>
>> Dmitry and I, we thought about a possible implementation allowing the
>> l2/l3 to coexists.
>>
>> The idea is assuming the l3 network namespaces are the leaf in the l2
>> namespace hierarchy tree. By default, init process is l2 namespace. From
>> a layer 3, it is impossible to do a new network namespace unshare.
>>
>> All the configuration is done into the l2 namespace. When a l3 is
>> created a new IP address should be created into the l2 namespace and
>> "pushed" into the l3. When the l3 dies, the IP is pulled to its parent,
>> aka the l2. In order to ensure security into the l3, the NET_ADMIN
>> capability is lost when doing unsharing for l3.
>> There is no extra code for socket virtualization. It is a common part.
>>
>> How to setup a l3 namespace ?
>> ----------
>>
>> 1 - setup a new IP address in l2 namespace
>> 2 - create a l3 namespace
>> 3 - specific socket ioctl to "push" the IP address from the l2
>> namespace to the newly created l3 namespace
>
> This means that there is some kind of identifier for the l3 namespace, right?

Not exactly. The bind_ns allows to assign an identifier to a namespace.
The namespace is an aggregation of the different namespace ressources
(ipc, pid, net, utsname, ...). But the result is the same, we use the
namespace identifier instead of a l3 namespace identifier.

>
>> The l2 lose visibility on the IP address and l3 gains visibility on the
>> IP address. A ifconfig or a ip command shows only the IP address
>> assigned to the namespace. Loopback address is always visible.
>
> Hmm.... I've been thinking about this, and I think this OK from the sockets point
> of view, i.e. binds() in l2 lose visibility to the new l3 address. There is
> a concern for a potential race here though.

Do you mean, someone in the l2 namespace can use the IP address before

pushing it the l3 namespace ? That is right, perhaps the call should be
done in one shoot (set address + pushing it to l3)

> However, it would be really nice to be able to see l3 namespace addresses in
> the parent l2 tagged in some way.

>
>> How to handle outgoing traffic ?
>> --------------------------------
>>
>> The bind must be checked with the IP addresses belonging to the l3
>> namespace and with all the derivative addresses (multicast, broadcast,
>> zero net, loopback, ...).
>>
>> The IP addresses will rely on aliased IP address. The source address
>> must be filled with the IP address belonging the l3 namespace when not
>> set. This is a trivial operation, because we know which IP addresses are
>> assigned to the l3 namespace.
>
> Can you provide a little more info?

I think I already answered this question in the previous email. I am
afraid this paragraph is not very clear ... ;)

>
>> When the route are resolved, the l3 namespace switch the its parent,
>> that is to say the l2 namespace, and the virtualization follows its
>> normal path.
>>
>> How to handle incoming traffic ?
>> --------------------------------
>>
>> Because we can have several sockets listening on the same
>> INADDR_ANY:port, we must find the network namespace associated with the
>> destination IP address.
>> For unicast, this is a trivial operation, because that can be checked
>> with the assigned IP address again. For broadcast and multicast, some
>> extra work should be done in order to store the namespaces which are
>> listening on a broadcast address. As soon as the namespace is found, we
>> switch to it. This can be done with netfilters.
>
> The problem is with multicasts. Multicast groups are joined on the interface
> bases. Every socket that bound *:multicast_port will receive multicast
> traffic once a single app joined the group. Since l3 namespaces don't have
> share the conceptual interface, theoretically, all l3 namespaces should receive
> multicast traffic.

Right. You sunk my battleship :)
Need to be thought...

>
>> Routes and co.
>> --------------
>>
>> - Routes: they are not isolated, each l3 namespace can see all the
>> routes from the other namespaces. That allows the routing engine to see
>> all the routes and choose the loopback when two network namespaces in
>> the same host try to communicate.
>>
>> - Cache: the routing cache must be isolated, otherwise the socket
>> isolation will not work. The l3 namespace code does not impact the l2
>> namespace code and route cache isolation is a common part if the l3
>> namespace switching is done in the right place.
>>
>>
>> Dmitry has posted the l2 namespace relying on the net namespace empty
>> framework, I will post the l3 namespace relying on the l2 namespace
>> today or tomorrow.
>>
>
> Looking forward to it.

Fixing a kref problem...

Thanks for all your comments.

  -- Daniel

_____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers