Subject: Re: [patch 2/6] [Network namespace] Network device sharing by view
Posted by Herbert Poetzl on Wed, 28 Jun 2006 14:15:48 GMT
View Forum Message <> Reply to Message

On Wed, Jun 28, 2006 at 06:31:05PM +1200, Sam Vilain wrote:
> Eric W. Biederman wrote:
> > Have a few more network interfaces for a layer 2 solution
> > is fundamental. Believing without proof and after arguments
> > to the contrary that you have not contradicted that a layer 2
> > solution is inherently slower is non-productive. Arguing
> > that a layer 2 only solution most prove itself on guest to guest
> > communication is also non-productive.
> >
>
> Yes, it does break what some people consider to be a sanity condition
> when you don't have loopback anymore within a guest. I once experimented
> with using 127.* addresses for per-guest loopback devices with vserver
> to fix this, but that couldn't work without fixing glibc to not make
> assumptions deep in the bowels of the resolver. I logged a fault with
> gnu.org and you can guess where it went :-).

this is what the lo* patches address, by providing
the required loopback isolation and providing lo
inside a guest (i.e. it looks and feels like a
normal system, except that you cannot modify the
interfaces from inside)

> I don't think it's just the performance issue, though. Consider also
> that if you only have one set of interfaces to manage, the overall
> configuration of the network stack is simpler. `ip addr list' on the
> host shows all the addresses on the system, you only have one routing
> table to manage, one set of iptables, etc.
>
> That being said, perhaps if each guest got its own interface, and from
> some suitably privileged context you could see them all, perhaps it
> would be nicer and maybe just as fast. Perhaps then *devices* could get
> their own routing namespaces, and routing namespaces could get iptables
> namespaces, or something like that, to give the most options.
>
> > With a guest with 4 IPs
> > 10.0.0.1 192.168.0.1 172.16.0.1 127.0.0.1
> > How do you make INADDR_ANY work with just filtering at bind time?
> >
>
> It used to just bind to the first one. Don't know if it still does.

no, it _alway_ binds to INADDR_ANY and checks
against other sockets (in the same context)

comparing the lists of assigned IPs (the subset)

so all checks happen at bind/connect time and
always against the set of IPs, only exception is
a performance optimization we do for single IP
guests (where INADDR_ANY gets rewritten to the
single IP)

best,
Herbert

> Sam.