Subject: Re: Network virtualization/isolation
Posted by Daniel Lezcano on Mon, 04 Dec 2006 10:18:09 GMT
View Forum Message <> Reply to Message

Hi Jamal,

thanks for taking the time read the document.

The objective of the document was not to convince one approach is better than other. I wanted to show the pros and the cons of each approach and to point that the 2 approaches are complementary.

Currently, there are some resources moved to a namespace relative access, the IPC and the utsname and this is into the 2.6.19 kernel. The work on the pid namespace is still in progress.

The idea is to use a "clone" approach relying on the "unshare_ns" syscall. The syscall is called with a set of flags for pids, ipcs, utsname, network ... You can then "unshare" only the network and have an application into its own network environment.

For a l3 approach, like a l2, you can run an apache server into a unshared network environment. Better, you can run several apaches server into several network namespaces without modifying the server's network configuration.

Some of us, consider l2 as perfectly adapted for some kind of containers like system containers and some kind of application containers running big servers, but find the l2 too much (seems to be a hammer to crush a beetle) for simple network requirements like for network migration, jails or containers which does not take care of such virtualization. For example, you want to create thousands of containers for a cluster of HPC jobs and just to have migration for these jobs. Does it make sense to have l2 approach ?

Dmitry Mishin and I, we thought about a l2/l3 solution and we thing we found a solution to have the 2 at runtime. Roughly, it is a l3 based on bind filtering and socket isolation, very similar to what vserver provides. I did a prototype, and it works well for IPV4/unicast.

So, considering, we have a l2 isolation/virtualization, and having a l3 relying on the l2 network isolation resources subset. Is it an acceptable solution ?

-- Daniel