Subject: Re: OpenVZ Density
Posted by dowdle on Wed, 30 Jul 2008 12:56:04 GMT
View Forum Message <> Reply to Message

With regards to OpenVZ and memory related system resource parameters, swap is included as if it were part of RAM. How would you include it in your calculations? Well, you'd add however big your host node swap is to your total physical RAM. You can add, right?

Why would you have to view a web page to read the vzsplit man page? How does it help? Well, you can see what values it gives for the various parameters for different numbers of containers. You can also give it an impossible value for the number of containers you want (say 99999999) and it'll tell you what it thinks is the maximum capacity of your machine. Of course the maximum capacity of your machine is just that, the max... and not a recommended number for real world deployment.

You say your containers will be all alike... software and process / configuration-wise that might be true... but that may not always be true load wise... unless you yourself happen to be the only consumer of your containers services.

A container is a group of child processes under a unique init process, and while you can run different Linux distributions in a container, you only need the resources necessary to run the processes... and don't need to follow a particular distribution's system requirements for memory. That is to say that if a distribution says it requires a minimum of 256MB of RAM to run, that does not hold true with a container running that distribution... especially when the OpenVZ flavor of that distribution as provided by a pre-created OS Template has been trimmed down considerably from a standard distribution install. For example, most distributions these days have a GUI installer and install X11, and set the default runlevel to 5 for a graphical login. None of the pre-created OS templates provided by the OpenVZ project include X11, they do not use the distribution's native installer, and there isn't a kernel running inside of the container so all of the resources needed by a kernel and all of the threads are not there. Since OpenVZ (by default anyway) isn't trying to provide you with a desktop OS a lot of the standard services of many distros are not required... like an X font server, hal, sound server, etc, etc, etc. A container really needs to run only those processes you want. Given all of these factors, a container can use substantially less memory / resources than a full distribution running on a physical machine or under machine / hardware virtualization.

Your comment about the smallest pre-created OS Template (Mandriva) has no bearing. The size of the OS Template represents files on the disk and not necessarily running processes that are going to be started up. I haven't installed a wide variety of different distros lately... but from my past experiences many of the "minimal" OS Templates do not include apache... or much beyond the bear minimum for a running system. Like I said, I've seen a few containers created from "minimal" pre-created OS Templates that only have 2 or 3 processes running by default when they start up. Since I've not installed the Mandriva OS Template you mention, I looked and it is 28MB in size. That is pretty darn minimal I must say if you compare that to the stock Mandriva install media size. I don't know if that includes Apache or not... but I would be surprised if it did. Once you get a little more experience playing around with different OS Templates you'll start to realize that I'm not making this up... and some of them do use a tiny amount of RAM. I'm not saying that a container made from a minimal OS Template is that useful for anything but they can

be handy for testing... and answering questions like... what's the maximum number of containers that is doable?

How do you ensure you don't overcommit? You don't configure however many containers you create to have access to more memory (again, physical and swap together) than you have. There are three camps... those who think overcommitment of resources should become the 8th deadly sin, those to don't care, and those who routinely overcommit resources. I fall into the later and I've been running OpenVZ for about 3 years now and haven't run into any significant problems as a result of resource overcommitment. Of course that's with the containers I run and the loads I have. YMMV.

You haven't made any progress solving this? Well then.. STFU and do some testing. You'll learn more by doing than bitching. Am I being harsh? Perhaps... but it seems to me that I've spent quite a bit of time attempting to answer your questions and I've only gotten whining thus far.

I do recommend that you play around with OpenVZ for a little while, do some testing, and gain a little confidence with it before trying to create a production environment. Read the Users Guide (which is admittedly a little outdated), the man pages for the various tools, and any wiki articles you find related to questions that come up while on you journey. I wish you luck on that journey and will be happy to answer any additional questions you might come up with in the future.