Subject: Re: [RFC][PATCH 2/7] RSS controller core
Posted by xemul on Tue, 13 Mar 2007 15:32:08 GMT
View Forum Message <> Reply to Message

Herbert Poetzl wrote:
> On Tue, Mar 13, 2007 at 10:17:54AM +0300, Pavel Emelianov wrote:
>> Herbert Poetzl wrote:
>>> On Mon, Mar 12, 2007 at 12:02:01PM +0300, Pavel Emelianov wrote:
>>>>>>> Maybe you have some ideas how we can decide on this?
>>>>>> We need to work out what the requirements are before we can
>>>>>> settle on an implementation.
>>>>> Linux-VServer (and probably OpenVZ):
>>>>>
>>>>> - shared mappings of 'shared' files (binaries
>>>>>   and libraries) to allow for reduced memory
>>>>>   footprint when N identical guests are running
>>>> This is done in current patches.
>>> nice, but the question was about _requirements_
>>> (so your requirements are?)
>>>
>>>>> - virtual 'physical' limit should not cause
>>>>>   swap out when there are still pages left on
>>>>>   the host system (but pages of over limit guests
>>>>>   can be preferred for swapping)
>>>> So what to do when virtual physical limit is hit?
>>>> OOM-kill current task?
>>> when the RSS limit is hit, but there _are_ enough
>>> pages left on the physical system, there is no
>>> good reason to swap out the page at all
>>>
>>> - there is no benefit in doing so (performance
>>>   wise, that is)
>>>
>>> - it actually hurts performance, and could
>>>   become a separate source for DoS
>>>
>>> what should happen instead (in an ideal world :)
>>> is that the page is considered swapped out for
>>> the guest (add guest penality for swapout), and
>> Is the page stays mapped for the container or not?
>> If yes then what's the use of limits? Container mapped
>> pages more than the limit is but all the pages are
>> still in memory. Sounds weird.
>
> sounds weird, but makes sense if you look at the full picture
>
> just because the guest is over its page limit doesn't
> mean that you actually want the system to swap stuff

> out, what you really want to happen is the following:
>
> - somehow mark those pages as 'gone' for the guest
> - penalize the guest (and only the guest) for the
>   'virtual' swap/page operation
> - penalize the guest again for paging in the page
> - drop/swap/page out those pages when the host system
>   decides to reclaim pages (from the host PoV)

Yeah! And slow down the container which caused global
limit hit (w/o hitting it's own limit!) by swapping
some others' pages out. This breaks the idea of isolation.

>>> when the page would be swapped in again, the guest
>>> takes a penalty (for the 'virtual' page in) and
>>> the page is returned to the guest, possibly kicking
>>> out (again virtually) a different page
>>>
>>>>> - accounting and limits have to be consistent
>>>>>   and should roughly represent the actual used
>>>>>   memory/swap (modulo optimizations, I can go
>>>>>   into detail here, if necessary)
>>>> This is true for current implementation for
>>>> booth - this patchset ang OpenVZ beancounters.
>>>>
>>>> If you sum up the physpages values for all containers
>>>> you'll get the exact number of RAM pages used.
>>> hmm, including or excluding the host pages?
>> Depends on whether you account host pages or not.
>
> you tell me? or is that an option in OpenVZ?

In OpenVZ we account resources in host system as well.
However we have an opportunity to turn this off.

> best,
> Herbert
>
>>>>> - OOM handling on a per guest basis, i.e. some
>>>>>   out of memory condition in guest A must not
>>>>>   affect guest B
>>>> This is done in current patches.
>>>> Herbert, did you look at the patches before
>>>> sending this mail or do you just want to
>>>> 'take part' in conversation w/o understanding
>>>> of hat is going on?
>>> again, the question was about requirements, not
>>> your patches, and yes, I had a look at them _and_

>>> the OpenVZ implementations ...
>>>
>>> best,
>>> Herbert
>>>
>>> PS: hat is going on? :)
>>>
>>>>> HTC,
>>>>> Herbert
>>>>>
>>>>>> Sigh. Who is running this show?   Anyone?
>>>>>>
>>>>>> You can actually do a form of overcommittment by allowing multiple
>>>>>> containers to share one or more of the zones. Whether that is
>>>>>> sufficient or suitable I don't know. That depends on the requirements,
>>>>>> and we haven't even discussed those, let alone agreed to them.
>>>>>>
>>>>>> ____
>>>>>> Containers mailing list
>>>>>> Containers@lists.osdl.org
>>>>>> https://lists.osdl.org/mailman/listinfo/containers
>

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