Subject: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Cedric Le Goater on Thu, 30 Aug 2007 10:05:07 GMT
View Forum Message <> Reply to Message

Hello All,

Some of us will meet next week for the first mini-summit on containers.
Many thanks to Alasdair Kergon and LCE for the help they provided in
making this mini-summit happen !

It will be help on Monday the 3rd of September from 9:00 to 12:45 at LCE
in room D. We also might get a phone line for external participants and,
if not, we should be able to set up a skype phone.

Here's a first try for the Agenda.

Global items

[let's try to defer discussion after presentation]

* Pavel Emelianov status update
* Serge E. Hallyn Container Roadmap including
 . task containers (Paul Menage)
 . resource management (Srivatsa Vaddagiri)

Special items

[brainstorm sessions which we would like to focus on]

* builing the global container object ('a la' openvz or vserver)
* container user space tools
* container checkpoint/restart

Thanks,

C.

====================== Section 1 ======================
=Introduction
====================== Section 1 ======================

We are trying to create a roadmap for the next year of
'container' development, to be reported to the upcoming kernel
summit. Containers here is a bit of an ambiguous term, so we are
taking it to mean all of:

1. namespaces
   kernel resource namespaces to support resource isolation
   and virtualization for virtual servers and application
   checkpoint/restart.
2. task containers framework
   task containers provide a framework for subsystems which associate
   state with arbitrary groups of processes, for purposes such as
   resource control/monitoring.
3. checkpoint/restart

===================== Section 2 ======================

=Detailed development plans

===================== Section 2 ======================

A (still under construction) list of features we expect to be worked on
next year looks like this:

1. completion of ongoing namespaces
    - pid namespace
        - push merged patchset upstream
        - kthread cleanup
            - especially nfs
            - autofs
        - af_unix credentials (stores pid_t?)
    - net namespace
    - ro bind mounts
2. continuation with new namespaces
    - devpts, console, and ttydrivers
    - user
    - time
    - namespace management tools
    - namespace entering (using one of:)
        - bind_ns()
        - ns container subsystem
        - (vs refuse this functionality)
    - multiple /sys mounts
        - break /sys into smaller chunks?
        - shadow dirs vs namespaces
    - multiple proc mounts
        - likely need to extend on the work done for pid namespaces
        - i.e. other /proc files will need some care
    - virtualization of statistics for 'top', etc
3. any additional work needed for virtual servers?
    - i.e. in-kernel keyring usage for cross-usernamespace permissions, etc
        - nfs and rpc updates needed?
        - general security fixes
            - per-container capabilities?

device access controls
e.g. root in container should not have access to /dev/sda by default)
filesystems access controls
'container object'?
implementation (perhaps largely userspace abstraction)
container enter
container list
container shutdown notification

4. task containers functionality
base features
hierarchical/virtualized containers
support vserver mgmnt of sub-containers
locking cleanup
control file API simplification
userpace RBCE to provide controls for
users
groups
pgrp
executable
specific containers targeted:
split cpusets into
cpuset
memset
network
connect/bind/accept controller using iptables
memory controller (see detail below)
cpu controller d (see detailbelow)
io controller (see detail below)
network flow id control
per-container OOM handler (userspace)
per-container swap
per-container disk I/O scheduling
per container memory reclaim
per container dirty page (write throttling) limit.
network rate limiting (outbound) based on container
misc
User level APIS to identify the resource limits that is allowed to a
job, for example, how much physical memory a
process can use. This should seamlessly
integrated with non-container environment as
well (may be with ulimit).
Per container stats, like pages on active list, cpus usage, etc
memory controller
users and requirements:
1. The containers solution would need resource
management (including memory control and per container swap files).
Paul Menage, YAMOMOTO Takshi, Peter Zijlstra, Pavel Emelianov have all shown

interest in the memory controller patches.
2. The memory controller can account for page
cache as well, all people interested in limiting page cahce control, can
theoratically put move all page cache hungry applications under the same
container.
Planned enhancements to the memory controller
1. Improved shared page accounting
2. Improved statistics
3. Soft-limit memory usage
generic infrastructure work:
1. Enhancing containerstats
a. Working on per controller statistics
b. Integrating taskstats with containerstats
2. CPU accounting framework
a. Migrate the accounting to be more precis
cpu controller
users and requirements:
1. Virtualization solutions like containers and
KVM need CPU control. KVM for example would
like to have both limits and guarantees
supported by a CPU controller, to control CPU
allocation to a particular instance.
2. Workload management products would like to exploit this for providing
guaranteed cpu bandwidth and also (hard/soft) limiting cpu usage.
work items
1. Fine-grained proportional-share fair-group scheduling.
2. More accurate SMP fairness
3. Hard limit
4. SCHED_FIFO type policy for groups
5. Improved statistics and debug facility for group scheduler
io controller
users and requirements:
1. At a talk presented to the Linux Foundation
(OSDL), the attendees showed interest in an IO
controller to control IO bandwidth of various
filesystem operations (backup, journalling,
etc)
work items:
1. Proof of concept IO controller and community discussion/feedback
2. Development and Integration of the IO controller with containers
open issues
1. Automatic tagging/resource classification engine

5. checkpoint/restart
memory c/r
(there are a few designs and prototypes)
(though this may be ironed out by then)

per-container swapfile?

overall checkpoint strategy (one of:)

in-kernel

userspace-driven

hybrid

overall restart strategy

use freezer API

use suspend-to-disk?

sysvipc

"set identifier" syscall

pid namespace

clone_with_pid()

live migration

====================== Section 3 ======================

=Use cases

====================== Section 3 ======================

1, Namespaces:

The most commonly listed uses for namespaces are virtual servers and checkpoint restart. Other uses are debugging (running tests in not-quite-virtual-servers) and resource isolation, such as the use of mounts namespaces to simulate multi-level directories for LSPP.

2. Task Containers:

(Vatsa to fill in)

3. Checkpoint/restart

load balancing:
applications can be migrated from high-load systems to ones with a lower load. Long-running applications can be checkpointed (or migrated) to start a short-running high-load job, then restarted.

kernel upgrades:
A long-running application - or whole virtual server - can be migrated or checkpointed so that the system can be rebooted, and the application can continue to run

====================== Section 4 ======================

=Involved parties

====================== Section 4 ======================

In the list of stakeholders, I try to guess based on past comments and
contributions what *general* area they are most likely to contribute in.
I may try to narrow those down later, but am just trying to get something
out the door right now before my next computer breaks.

Stakeholders:
Eric Biederman
everything
google
task containers
ibm (serge, dave, cedric, daniel)
namespaces
checkpoint/restart
bull (benjamin, pierre)
namespaces
checkpoint/restart
ibm (balbir, vatsa)
task containers
kerlabs
checkpoint/restart
openvz
everything
NEC Japan (Masahiko Takahashi)
checkpoint/restart
Linux-VServer
namespaces+containers
zap project
checkpoint/restart
planetlab
everything
hp
network namespaces, virtual servers?
XtreemOS
checkpoint/restart
Fujitsu/VA Linux Japan
resource control
BLCR (Paul H. Hargrove)
checkpoint/restart

Is anyone else still missing from the list?

thanks,
-serge

____
Containers mailing list
Containers@lists.linux-foundation.org

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Oren Laadan on Fri, 31 Aug 2007 03:26:22 GMT
View Forum Message <> Reply to Message

Cedric Le Goater wrote:
> Hello All,
>
> Some of us will meet next week for the first mini-summit on containers.
> Many thanks to Alasdair Kergon and LCE for the help they provided in
> making this mini-summit happen !
>
> It will be help on Monday the 3rd of September from 9:00 to 12:45 at LCE
> in room D. We also might get a phone line for external participants and,
> if not, we should be able to set up a skype phone.
>
> Here's a first try for the Agenda.
>
> Global items
>
>   [let's try to defer discussion after presentation]
>
> * Pavel Emelianov status update
> * Serge E. Hallyn Container Roadmap including
> . task containers (Paul Menage)
> . resource management (Srivatsa Vaddagiri)
>
> Special items
>
> [brainstorm sessions which we would like to focus on]
>
> * builing the global container object ('a la' openvz or vserver)
> * container user space tools
> * container checkpoint/restart

```
	5. checkpoint/restart
		memory c/r
			(there are a few designs and prototypes)
			(though this may be ironed out by then)
			per-container swapfile?
		overall checkpoint strategy  (one of:)
			in-kernel
			userspace-driven
			hybrid
		overall restart strategy
		use freezer API
```

use suspend-to-disk?

sysvipc
"set identifier" syscall
pid namespace
clone_with_pid()
There are other identifiers - pseudo terminals, message queues (mq)
(if you insist on supporting these ...). In general, we need a way
to specify the virtual id of a resource that is created. I suggest
that this should be part of an interface between c/r and containers
(see below)

live migration
aka pre-copy (which can be used for live migration but also to reduce
the downtime due to a checkpoint).

how about adding incremental checkpoint to the list ?

I think that it is also important to discuss an interface between c/r and
containers, each of which stands on it own. For instance, how to request
a specific virtual id (during restart), define required notifiers (to
set/unset c/r related data on/off a task), control c/r-related setting of
container (e.g. frozen, restarting) that may affect behavior, such as
signal handling, and so forth. Also, such an interface can allow existing
c/r implementations to work with different virtualization implementations
as they become available.

Many of these were discussed in a recent Zap paper present in USENIX:
http://www.ncl.cs.columbia.edu/publications/usenix2007_fordist.pdf
The paper describes important design choices in Zap (but I'm biased ...).
I think it may serve as an appetizer for the discussion :P

Oren.

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Cedric Le Goater on Fri, 31 Aug 2007 14:26:02 GMT
View Forum Message <> Reply to Message

Hello Oren,

Oren Laadan wrote:
> Cedric Le Goater wrote:

>> Hello All,
>>
>> Some of us will meet next week for the first mini-summit on containers.
>> Many thanks to Alasdair Kergon and LCE for the help they provided in
>> making this mini-summit happen !
>>
>> It will be help on Monday the 3rd of September from 9:00 to 12:45 at LCE
>> in room D. We also might get a phone line for external participants and,
>> if not, we should be able to set up a skype phone.
>>
>> Here's a first try for the Agenda.
>>
>> Global items
>>
>>   [let's try to defer discussion after presentation]
>>
>> * Pavel Emelianov status update
>> * Serge E. Hallyn Container Roadmap including
>> . task containers (Paul Menage)
>> . resource management (Srivatsa Vaddagiri)
>>
>> Special items
>>
>> [brainstorm sessions which we would like to focus on]
>>
>> * builing the global container object ('a la' openvz or vserver)
>> * container user space tools
>> * container checkpoint/restart
>
>   5. checkpoint/restart
>   memory c/r
>   (there are a few designs and prototypes)
>   (though this may be ironed out by then)
>   per-container swapfile?
>   overall checkpoint strategy (one of:)
>   in-kernel
>   userspace-driven
>   hybrid
>   overall restart strategy
>   use freezer API
>   use suspend-to-disk?
>
>   sysvipc
>   "set identifier" syscall
>   pid namespace
>   clone_with_pid()
> There are other identifiers - pseudo terminals, message queues (mq)

right, we have plans for developing these if needed (cf 2.)

> (if you insist on supporting these ...). In general, we need a way
> to specify the virtual id of a resource that is created.

right, pierre peiffer has sent such a pachset for the sysvipc namespace.
I'm looking at a clone_with_pid() for pid namespace.

> I suggest
> that this should be part of an interface between c/r and containers
> (see below)
>
> live migration
> aka pre-copy (which can be used for live migration but also to reduce
> the downtime due to a checkpoint).

yes that's usually what the buzz term "live migration" is used for.

> how about adding incremental checkpoint to the list ?

sure. I think it's a bit early to address these topic but we should have
them in mind as some implementations already exist. And we need to gather
all the needs.

> I think that it is also important to discuss an interface between c/r and
> containers, each of which stands on it own. For instance, how to request
> a specific virtual id (during restart), define required notifiers (to
> set/unset c/r related data on/off a task), control c/r-related setting of
> container (e.g. frozen, restarting) that may affect behavior, such as
> signal handling, and so forth.

This is exactly what we want to talk about.

We need to identify these C/R needs, talk and agree about possible APIS
and then convince the linux subsystem maintainers that they are useful
for a large set of C/R solutions based on containers.

> Also, such an interface can allow existing c/r implementations to work with
> different virtualization implementations as they become available.

what you call "virtualization" (private identifier namespaces), is I think
being covered by the namespaces. These namespaces are not complete (like
we're missing a way to reassign ids) but they are going in the right
direction, IMO. However, I don't think there will be different
"virtualization" implementations in mainline.

> Many of these were discussed in a recent Zap paper present in USENIX:
> http://www.ncl.cs.columbia.edu/publications/usenix2007_fordist.pdf

> The paper describes important design choices in Zap (but I'm biased ...).
> I think it may serve as an appetizer for the discussion :P

Thanks, I hope we all have time to read it.

C.

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Cedric Le Goater on Fri, 31 Aug 2007 14:59:16 GMT
View Forum Message <> Reply to Message

>> Many of these were discussed in a recent Zap paper present in USENIX:
>> http://www.ncl.cs.columbia.edu/publications/usenix2007_fordist.pdf
>> The paper describes important design choices in Zap (but I'm biased ...).
>> I think it may serve as an appetizer for the discussion :P
>
> Thanks, I hope we all have time to read it.

The abstract says :

"...
Our results show checkpoint and restart times 3 to 55 times faster than
OpenVZ and 5 to 1100 times faster than Xen."

I'm impressed ! :) When can we play it ?

Thanks for the appetizer !

C.

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by dev on Fri, 31 Aug 2007 15:59:15 GMT
View Forum Message <> Reply to Message

Cedric Le Goater wrote:
>>>Many of these were discussed in a recent Zap paper present in USENIX:
>>>http://www.ncl.cs.columbia.edu/publications/usenix2007_fordist.pdf

>>>The paper describes important design choices in Zap (but I'm biased ...).
>>>I think it may serve as an appetizer for the discussion :P
>>
>>Thanks, I hope we all have time to read it.
>
>
> The abstract says :
>
> "...
> Our results show checkpoint and restart times 3 to 55 times faster than
> OpenVZ and 5 to 1100 times faster than Xen."
>
> I'm impressed ! :) When can we play it ?
>
> Thanks for the appetizer !

It is totally unfair to compare full virtualization solution such as OpenVZ
with sync on VE stop (for quotas consistency) and which doesn't require shared storage for migration
with POC which uses shared storage in the paper.

I'm not sure why author didn't pay attention to these HUGE differences in configuration...
Maybe because 1100x times is so incredible :@)

Thanks,
Kirill

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Oren Laadan on Fri, 31 Aug 2007 18:20:50 GMT

View Forum Message <> Reply to Message

Cedric Le Goater wrote:
> Hello Oren,
>
> Oren Laadan wrote:
>> Cedric Le Goater wrote:
>>> Hello All,
>>>
>>> Some of us will meet next week for the first mini-summit on containers.
>>> Many thanks to Alasdair Kergon and LCE for the help they provided in
>>> making this mini-summit happen !
>>>

```
>>> It will be help on Monday the 3rd of September from 9:00 to 12:45 at LCE
>>> in room D. We also might get a phone line for external participants and,
>>> if not, we should be able to set up a skype phone.
>>>
>>> Here's a first try for the Agenda.
>>>
>>> Global items
>>>
>>>   [ let's try to defer discussion after presentation ]
>>>
>>> * Pavel Emelianov status update
>>> * Serge E. Hallyn Container Roadmap including
>>>  . task containers (Paul Menage)
>>>  . resource management (Srivatsa Vaddagiri)
>>>
>>> Special items
>>>
>>>  [ brainstorm sessions which we would like to focus on ]
>>>
>>> * builing the global container object ('a la' openvz or vserver)
>>> * container user space tools
>>> * container checkpoint/restart
>>      5. checkpoint/restart
>>              memory c/r
>>                      (there are a few designs and prototypes)
>>                      (though this may be ironed out by then)
>>                      per-container swapfile?
>>              overall checkpoint strategy  (one of:)
>>                      in-kernel
>>                      userspace-driven
>>                      hybrid
>>              overall restart strategy
>>              use freezer API
>>              use suspend-to-disk?
>>
>>              sysvipc
>>                      "set identifier" syscall
>>   pid namespace
>>                      clone_with_pid()
>> There are other identifiers - pseudo terminals, message queues (mq)
>
> right, we have plans for developing these if needed (cf 2.)
>
>> (if you insist on supporting these ...). In general, we need a way
>> to specify the virtual id of a resource that is created.
>
> right, pierre peiffer has sent such a pachset for the sysvipc namespace.
> I'm looking at a clone_with_pid() for pid namespace.
```

>
>> I suggest
>> that this should be part of an interface between c/r and containers
>> (see below)
>>
>>   live migration
>> aka pre-copy (which can be used for live migration but also to reduce
>> the downtime due to a checkpoint).
>
> yes that's usually what the buzz term "live migration" is used for.
>
>> how about adding incremental checkpoint to the list ?
>
> sure. I think it's a bit early to address these topic but we should have
> them in mind as some implementations already exist. And we need to gather
> all the needs.

exists in Zap; many lessons learned ;)

>
>> I think that it is also important to discuss an interface between c/r and
>> containers, each of which stands on it own. For instance, how to request
>> a specific virtual id (during restart), define required notifiers (to
>> set/unset c/r related data on/off a task), control c/r-related setting of
>> container (e.g. frozen, restarting) that may affect behavior, such as
>> signal handling, and so forth.
>
> This is exactly what we want to talk about.
>
> We need to identify these C/R needs, talk and agree about possible APIS
> and then convince the linux subsystem maintainers that they are useful
> for a large set of C/R solutions based on containers.
>
>> Also, such an interface can allow existing c/r implementations to work with
>> different virtualization implementations as they become available.
>
> what you call "virtualization" (private identifier namespaces), is I think
> being covered by the namespaces. These namespaces are not complete (like
> we're missing a way to reassign ids) but they are going in the right
> direction, IMO. However, I don't think there will be different
> "virtualization" implementations in mainline.

I do hope so too. I'm thinking that the current ones may take some time
to converge, and even then there may be out-of-mainline (experimental ?
alternative ?) implementation as it so happens with linux at time :)
In that case defining an interface can be useful (apart from the fact
that you tackle issues when you actually define one).
There is also the other side -- multiple c/r implementations (mainline

or not) that may be geared toward different goals depending on desires
performance, functionality etc.

>
>> Many of these were discussed in a recent Zap paper present in USENIX:
>> http://www.ncl.cs.columbia.edu/publications/usenix2007_fordist.pdf
>> The paper describes important design choices in Zap (but I'm biased ...).
>> I think it may serve as an appetizer for the discussion :P
>
> Thanks, I hope we all have time to read it.
>
> C.

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Kirill Kolyshkin on Sun, 02 Sep 2007 22:49:54 GMT

View Forum Message <> Reply to Message

So, this is just to confirm the final details about container
mini-summit which will be held tomorrow.

Time: starting at 9am 3th Sept.
Place: Cambridge's University Arms Hotel, room Churchill D.

Let's meet at the hotel lobby close to 9am and when go to the room.

Eric, Paul,
Can you please clarify whether will you be able to present or not?

PS sorry if you got this message a few times -- some DNS problems on my
mail server.

On 30/08/07, Cedric Le Goater <clg@fr.ibm.com> wrote:
>
> Hello All,
>
> Some of us will meet next week for the first mini-summit on containers.
> Many thanks to Alasdair Kergon and LCE for the help they provided in
> making this mini-summit happen !
>
> It will be help on Monday the 3rd of September from 9:00 to 12:45 at LCE
> in room D. We also might get a phone line for external participants and,

> if not, we should be able to set up a skype phone.
>
> Here's a first try for the Agenda.
>
> Global items
>
>   [let's try to defer discussion after presentation]
>
> * Pavel Emelianov status update
> * Serge E. Hallyn Container Roadmap including
>   . task containers (Paul Menage)
>   . resource management (Srivatsa Vaddagiri)
>
> Special items
>
>   [brainstorm sessions which we would like to focus on]
>
> * builing the global container object ('a la' openvz or vserver)
> * container user space tools
> * container checkpoint/restart
>
>
> Thanks,
>
> C.
>
>
>
> ===================== Section 1 =====================
> =Introduction
> ===================== Section 1 =====================
>
> We are trying to create a roadmap for the next year of
> 'container' development, to be reported to the upcoming kernel
> summit. Containers here is a bit of an ambiguous term, so we are
> taking it to mean all of:
>
>   1. namespaces
>   kernel resource namespaces to support resource isolation
>   and virtualization for virtual servers and application
>   checkpoint/restart.
>   2. task containers framework
>   task containers provide a framework for subsystems which
> associate
>   state with arbitrary groups of processes, for purposes
> such as
>   resource control/monitoring.
>   3. checkpoint/restart

```
>
> ======================  Section 2  ======================
> =Detailed development plans
> ======================  Section 2  ======================
>
> A (still under construction) list of features we expect to be worked on
> next year looks like this:
>
>        1. completion of ongoing namespaces
>                pid namespace
>                        push merged patchset upstream
>                        kthread cleanup
>                                especially nfs
>                                autofs
>                        af_unix credentials (stores pid_t?)
>                net namespace
>                ro bind mounts
>        2. continuation with new namespaces
>                devpts, console, and ttydrivers
>                user
>                time
>                namespace management tools
>                namespace entering  (using one of:)
>                        bind_ns()
>                        ns container subsystem
>                        (vs refuse this functionality)
>                multiple /sys mounts
>                        break /sys into smaller chunks?
>                        shadow dirs vs namespaces
>                multiple proc mounts
>                        likely need to extend on the work done for pid
> namespaces
>                        i.e. other /proc files will need some care
>                                virtualization of statistics for 'top',
> etc
>        3. any additional work needed for virtual servers?
>                i.e. in-kernel keyring usage for cross-usernamespace
> permissions, etc
>                        nfs and rpc updates needed?
>                        general security fixes
>                                per-container capabilities?
>                        device access controls
>                                e.g. root in container should not have
> access to /dev/sda by default)
>                        filesystems access controls
>                'container object'?
>                        implementation (perhaps largely userspace
> abstraction)
>                       container enter
>                       container list
>                       container shutdown notification
>
>       4. task containers functionality
>               base features
>                       hierarchical/virtualized containers
>                               support vserver mgmnt of sub-containers
>                       locking cleanup
>                       control file API simplification
>               userpace RBCE to provide controls for
>                       users
>                       groups
>                       pgrp
>                       executable
>               specific containers targeted:
>                       split cpusets into
>                               cpuset
>                               memset
>                       network
>                               connect/bind/accept controller using
> iptables
>                       memory controller (see detail below)
>                       cpu controller d (see detailbelow)
>                       io controller (see detail below)
>                       network flow id control
>                       per-container OOM handler (userspace)
>                       per-container swap
>                       per-container disk I/O scheduling
>                       per container memory reclaim
>                       per container dirty page (write throttling) limit.
>                       network rate limiting (outbound) based on
> container
>               misc
>                       User level APIS to identify the resource limits
> that is allowed to a
>                               job, for example, how much physical memory
> a
>                               process can use.  This should seamlessly
>                               integrated with non-container environment
> as
>                               well (may be with ulimit).
>                       Per container stats, like pages on active list,
> cpus usage, etc
>               memory controller
>                       users and requirements:
>                               1. The containers solution would need
> resource
```

> management (including memory control and
> per container swap files).
> Paul Menage, YAMOMOTO Takshi, Peter
> Zijlstra, Pavel Emelianov have all shown
> interest in the memory controller patches.
> 2. The memory controller can account for
> page
> cache as well, all people interested in
> limiting page cahce control, can
> theoratically put move all page cache
> hungry applications under the same
> container.
> Planned enhancements to the memory controller
> 1. Improved shared page accounting
> 2. Improved statistics
> 3. Soft-limit memory usage
> generic infrastructure work:
> 1. Enhancing containerstats
> a. Working on per controller
> statistics
> b. Integrating taskstats with
> containerstats
> 2. CPU accounting framework
> a. Migrate the accounting to be
> more precis
> cpu controller
> users and requirements:
> 1. Virtualization solutions like
> containers and
> KVM need CPU control. KVM for example
> would
> like to have both limits and guarantees
> supported by a CPU controller, to
> control CPU
> allocation to a particular instance.
> 2. Workload management products would like
> to exploit this for providing
> guaranteed cpu bandwidth and also
> (hard/soft) limiting cpu usage.
> work items
> 1. Fine-grained proportional-share
> fair-group scheduling.
> 2. More accurate SMP fairness
> 3. Hard limit
> 4. SCHED_FIFO type policy for groups
> 5. Improved statistics and debug facility
> for group scheduler
> io controller

```
>			users and requirements:
>				1. At a talk presented to the Linux
> Foundation
>				(OSDL), the attendees showed interest in
> an IO
>				controller to control IO bandwidth of
> various
>				filesystem operations (backup,
> journalling,
>				etc)
>			work items:
>				1. Proof of concept IO controller and
> community discussion/feedback
>				2. Development and Integration of the IO
> controller with containers
>			open issues
>				1. Automatic tagging/resource
> classification engine
>
>
>	5. checkpoint/restart
>		memory c/r
>			(there are a few designs and prototypes)
>			(though this may be ironed out by then)
>			per-container swapfile?
>		overall checkpoint strategy  (one of:)
>			in-kernel
>			userspace-driven
>			hybrid
>		overall restart strategy
>		use freezer API
>		use suspend-to-disk?
>		sysvipc
>			"set identifier" syscall
>		pid namespace
>			clone_with_pid()
>		live migration
>
>
> ======================  Section 3  ======================
> =Use cases
> ======================  Section 3  ======================
>
>	1, Namespaces:
>
>	The most commonly listed uses for namespaces are virtual
>	servers and checkpoint restart.  Other uses are debugging
>	(running tests in not-quite-virtual-servers) and resource
```

> isolation, such as the use of mounts namespaces to simulate
> multi-level directories for LSPP.
>
> 2. Task Containers:
>
> (Vatsa to fill in)
>
> 3. Checkpoint/restart
>
> load balancing:
> applications can be migrated from high-load systems to ones
> with a lower load. Long-running applications can be checkpointed
> (or migrated) to start a short-running high-load job, then
> restarted.
>
> kernel upgrades:
> A long-running application - or whole virtual server - can
> be migrated or checkpointed so that the system can be
> rebooted, and the application can continue to run
>
>
> ===================== Section 4 =====================
> =Involved parties
> ===================== Section 4 =====================
>
> In the list of stakeholders, I try to guess based on past comments and
> contributions what *general* area they are most likely to contribute in.
> I may try to narrow those down later, but am just trying to get something
> out the door right now before my next computer breaks.
>
> Stakeholders:
> Eric Biederman
> everything
> google
> task containers
> ibm (serge, dave, cedric, daniel)
> namespaces
> checkpoint/restart
> bull (benjamin, pierre)
> namespaces
> checkpoint/restart
> ibm (balbir, vatsa)
> task containers
> kerlabs
> checkpoint/restart
> openvz
> everything
> NEC Japan (Masahiko Takahashi)

>   checkpoint/restart
>   Linux-VServer
>   namespaces+containers
>   zap project
>   checkpoint/restart
>   planetlab
>   everything
>   hp
>   network namespaces, virtual servers?
>   XtreemOS
>   checkpoint/restart
>   Fujitsu/VA Linux Japan
>   resource control
>   BLCR (Paul H. Hargrove)
>   checkpoint/restart
>
> Is anyone else still missing from the list?
>
> thanks,
> -serge
>
>

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Alasdair G Kergon on Mon, 03 Sep 2007 00:03:17 GMT

View Forum Message <> Reply to Message

We are still hoping to have a speaker phone set up - you may want to prepare
and distribute a dial-in number.

Alasdair
--
agk@redhat.com

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by ebiederm on Mon, 03 Sep 2007 00:25:36 GMT

View Forum Message <> Reply to Message

"Kirill Kolyshkin" <kolyshkin@gmail.com> writes:

> So, this is just to confirm the final details about container mini-summit which
> will be held tomorrow.
>
> Time: starting at 9am 3th Sept.
> Place: Cambridge's University Arms Hotel, room Churchill D.
>
>
> Let's meet at the hotel lobby close to 9am and when go to the room.
>
> Eric, Paul,
> Can you please clarify whether will you be able to present or not?

Not physically.   I might be able to dial in if that is available,
depends on how much I adjust my sleep schedule today before my trip.
I won't be present physically until sometime on the 4th.

Eric

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Paul Menage on Mon, 03 Sep 2007 03:51:45 GMT

View Forum Message <> Reply to Message

On 9/2/07, Kirill Kolyshkin <kolyshkin@gmail.com> wrote:
>
> Eric, Paul,
> Can you please clarify whether will you be able to present or not?
>

I'll be dialling in or on Skype, depending on what's available.

Paul

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007

Posted by Srivatsa Vaddagiri on Mon, 03 Sep 2007 04:44:52 GMT
View Forum Message <> Reply to Message

On Sun, Sep 02, 2007 at 11:49:54PM +0100, Kirill Kolyshkin wrote:
> So, this is just to confirm the final details about container
> mini-summit which will be held tomorrow.

I am planning to attend this on phone (along with Dhaval and Vaidya from IBM).

--
Regards,
vatsa

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Srivatsa Vaddagiri on Mon, 03 Sep 2007 08:22:03 GMT
View Forum Message <> Reply to Message

On Sun, Sep 02, 2007 at 11:49:54PM +0100, Kirill Kolyshkin wrote:
> So, this is just to confirm the final details about container
> mini-summit which will be held tomorrow.
>
> Time: starting at 9am 3th Sept.
> Place: Cambridge's University Arms Hotel, room Churchill D.

Hi Kirill,
What's the callin details for this conference?

Regards,
vatsa

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Cedric Le Goater on Mon, 03 Sep 2007 08:45:57 GMT
View Forum Message <> Reply to Message

Hello !

Cedric Le Goater wrote:
> Hello All,
>
> Some of us will meet next week for the first mini-summit on containers.
> Many thanks to Alasdair Kergon and LCE for the help they provided in
> making this mini-summit happen !
>
> It will be help on Monday the 3rd of September from 9:00 to 12:45 at LCE
> in room D. We also might get a phone line for external participants and,
> if not, we should be able to set up a skype phone.
>
> Here's a first try for the Agenda.
>
> Global items
>
>   [let's try to defer discussion after presentation]
>
> * Pavel Emelianov status update

slides are available here :

 http://download.openvz.org/~xemul/minisummit.odp

thanks,

C.

> * Serge E. Hallyn Container Roadmap including
> . task containers (Paul Menage)
> . resource management (Srivatsa Vaddagiri)
>
> Special items
>
> [brainstorm sessions which we would like to focus on]
>
> * builing the global container object ('a la' openvz or vserver)
> * container user space tools
> * container checkpoint/restart
>
>
> Thanks,
>
> C.
>
>
>
>
> --

>
> ____
> Containers mailing list
> Containers@lists.linux-foundation.org
> https://lists.linux-foundation.org/mailman/listinfo/containers

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Paul Menage on Mon, 03 Sep 2007 09:03:06 GMT

View Forum Message <> Reply to Message

On 9/3/07, Cedric Le Goater <clg@fr.ibm.com> wrote:
>
>   http://download.openvz.org/~xemul/minisummit.odp
>

I notice ?s against "Task scheduler" and "Network scheduler".

Is "Task scheduler" meant to represent "CPU scheduler" or "task count limit". If the former, CFS in the mainline should provide a lot of what we need, and has already been linked with task containers by Srivatsa Vaddagiri.

Network scheduling is already fairly advanced in Linux - all we need is a way to be able to feed container information into existing Linux traffic control concepts. We've played with an approach that lets us tag a container with a particular id, and then use that id as the primary classifier in a standard HTB controller, and it seems to be fairly successful.

Paul

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Pavel Emelianov on Mon, 03 Sep 2007 09:32:46 GMT

View Forum Message <> Reply to Message

Paul Menage wrote:

> On 9/3/07, Cedric Le Goater <clg@fr.ibm.com> wrote:
>>   http://download.openvz.org/~xemul/minisummit.odp
>>
>
> I notice ?s against "Task scheduler" and "Network scheduler".
>
> Is "Task scheduler" meant to represent "CPU scheduler" or "task count
> limit". If the former, CFS in the mainline should provide a lot of
> what we need, and has already been linked with task containers by
> Srivatsa Vaddagiri.

Yes, task scheduler is the CPU scheduler.

> Network scheduling is already fairly advanced in Linux - all we need
> is a way to be able to feed container information into existing Linux
> traffic control concepts. We've played with an approach that lets us
> tag a container with a particular id, and then use that id as the
> primary classifier in a standard HTB controller, and it seems to be
> fairly successful.
>
> Paul
>

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Paul Menage on Mon, 03 Sep 2007 09:48:23 GMT

View Forum Message <> Reply to Message

On 9/3/07, Pavel Emelyanov <xemul@openvz.org> wrote:
>
> Yes, task scheduler is the CPU scheduler.

OK. Am I right in thinking that CFS is expected to provide most of the
CPU scheduler support that we need, when enhanced with Vatsa's group
scheduling patches?

Paul

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Pavel Emelianov on Mon, 03 Sep 2007 09:50:59 GMT
View Forum Message <> Reply to Message

Paul Menage wrote:
> On 9/3/07, Pavel Emelyanov <xemul@openvz.org> wrote:
>> Yes, task scheduler is the CPU scheduler.
>
> OK. Am I right in thinking that CFS is expected to provide most of the
> CPU scheduler support that we need, when enhanced with Vatsa's group
> scheduling patches?

I hope so :)

> Paul
>

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007
Posted by Srivatsa Vaddagiri on Mon, 03 Sep 2007 10:16:20 GMT
View Forum Message <> Reply to Message

On Mon, Sep 03, 2007 at 02:48:23AM -0700, Paul Menage wrote:
> OK. Am I right in thinking that CFS is expected to provide most of the
> CPU scheduler support that we need, when enhanced with Vatsa's group
> scheduling patches?

CFS pretty much provides the core logic to fairly divide the cpu as per
the weight of each group. One complication is with respect to SMP
load balance, to ensure that each group gets its fair share on all the
cpus put together.

We have been experimenting with few ideas on the smp group fairness and expect
to send out the patches to Andrew in a week or two.

--
Regards,
vatsa

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers