
Subject: Re: [ckrm-tech] [PATCH] BC: resource beancounters (v4) (added user memory)

Posted by [Balbir Singh](#) on Mon, 18 Sep 2006 11:27:07 GMT

[View Forum Message](#) <> [Reply to Message](#)

Pavel Emelianov wrote:

> Kirill Korotaev wrote:

>

> [snip]

>>> I have a C program that computes limits to obtain desired guarantees

>>> in a single 'for (i = 0; i < n; n++)' loop for any given set of guarantees.

>>> With all error handling, beautifull output, nice formatting etc it weights

>>> only 60 lines.

>

> Look at http://wiki.openvz.org/Containers/Guarantees_for_resources

> I've described there how a guarantee can be get with limiting in details.

>

> [snip]

>

>>> I do not 'do not like guarantee'. I'm just sure that there are two ways

>>> for providing guarantee (for unreclaimable resorces):

>>> 1. reserving resource for group in advance

>>> 2. limit resource for others

>>> Reserving is worse as it is essentially limiting (you cut off 100Mb from

>>> 1Gb RAM thus limiting the other groups by 900Mb RAM), but this limiting

>>> is too strict - you have to reserve less than RAM size. Limiting in

>>> run-time is more flexible (you may create an overcommitted BC if you

>>> want to) and leads to the same result - guarantee.

>> I think this deserves putting on Wiki.

>> It is very good clear point.

>

> This is also on the page I gave link at.

The program (calculate_limits()) listed on the website does not work for the following case

```
N=2;
```

```
R=100;
```

```
g[2] = {30, 30};
```

The output is -10 and -10 for the limits

For

```
N=3;
```

```
R=100;
```

```
g[3] = {30, 30, 10};
```

I get -70, -70 and -110 as the limits

Am I interpreting the parameters correctly? Or the program is broken?

--

Balbir Singh,
Linux Technology Center,
IBM Software Labs
