
Subject: Re: [ckrm-tech] [PATCH 4/7] UBC: syscalls (user interface)

Posted by [Magnus Damm](#) on Mon, 21 Aug 2006 09:18:08 GMT

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

On Mon, 2006-08-21 at 11:03 +0200, Andi Kleen wrote:

> On Monday 21 August 2006 10:42, Magnus Damm wrote:

>

> > No problem. The second URL pointed to a x86_64 version where I tried to

> > break out code to make some kind of generic NUMA emulation layer. At

> > that time no one seemed interested in that strategy as a simple resource

> > control solution so I gave that up.

> >

> > For x86_64 I think it's only worth mucking around with the code if

> > people believe that it is the right way to go for in-kernel resource

> > control.

>

> Does it by chance fix the existing code? Andrew has been complaining

> (and I could reproduce) that numa=fake=16 makes it triple fault at boot.

> The theory was that it didn't like empty nodes which can happen this way.

> I unfortunately didn't have time to look into it closely so far.

The code does rearrange how the boundaries are calculated, and it may

happen to fix that specific problem. I'll try to find some time later

this week to have a look at it.

> > The x86_64 patches above include code to divide each real NUMA node into

> > several smaller emulated nodes, but that is kind of pointless if people

> > only use it for non-resource control purposes, ie just to play with

> > CPUSETS and NUMA on non-NUMA hardware. For simple purposes like that I

> > think the existing NUMA emulation code for x86_64 works perfectly well.

> >

> > I still think that i386 users would benefit from NUMA emulation though.

> > If you want me to up-port the i386-specific code just let me know.

>

> I personally have my doubts about 32bit NUMA -- it will always have

> ZONE_NORMAL only on a single node, which limits it very much.

> But ok I guess it might be useful to somebody.

Very true. I was mainly thinking about the i386 code as a simple way for people to play with NUMA and CPUSETS.

/ magnus
