Subject: Re: [PATCH 4/4] get rid of populate for memcg
Posted by Glauber Costa on Wed, 21 Mar 2012 07:36:19 GMT
View Forum Message <> Reply to Message

On 03/20/2012 10:31 PM, Tejun Heo wrote:

```
> Hello, Glauber.
>
> On Tue, Mar 20, 2012 at 08:50:56PM +0400, Glauber Costa wrote:
>> @@ -4929,7 +4929,9 @@ mem_cgroup_create(struct cgroup *cont)
>>    atomic_set(&memcg->refcnt, 1);
>>    memcg->move_charge_at_immigrate = 0;
>>    mutex_init(&memcg->thresholds_lock);
>> - return&memcg->css;
>> +
>> + if (!register_kmem_files(memcg,&mem_cgroup_subsys))
>> +  return&memcg->css;
>
> After the change, I think register_kmem_files() is a quite misleading
> name.
```

how about init_kmem() ?

Remember the slab bits will are likely to end up here as well in the end.

```
>> @@ -2484,6 +2484,11 @@ int proto_register(struct proto *prot, int alloc_slab)
>>     }
>>    }
>>
>> +#ifdef CONFIG_CGROUP_MEM_RES_CTLR_KMEM
>> + if (prot->init_cgroup)
>> +  prot->init_cgroup(NULL, NULL);
>> +#endif
>
> So, init_cgroup() is overloaded to do two things - one load time init
> and per-cgroup init, depending on the args.
```

Yes. I don't love it, but there is quite a bunch of precedents for this.
Like the shrinkers in vmscan, for instance.

a NULL argument is a probe, a valid argument should have action taken.

```
>> @@ -37,7 +37,6 @@ static struct cftype tcp_files[] = {
>>    },
>>    { } /* terminate */
>>   };
>> -CGROUP_SUBSYS_CFTYPES(mem_cgroup_subsys, tcp_files);
>
> What I don't get is why you can't just keep this.  Is it because the
```

> files might appear before the protocol is registered? Wouldn't it be
> much better to add ipv4_tcp_init_cgroup() or whatever call to
> inet_init() instead of overloading init_cgroup() with mostly unrelated
> stuff?
>

The reason is that this has to be kept generic for protocols that may want to implement this in the future - since the pressure controls themselves are generic, the per-cgroup versions should be as well.

And in general, a protocol can live in a module, or not be registered despite being compiled in.

When the root memcg is created, prot_register() is usually not yet called, at least for tcp.

Now, what we do with the files, are our decision in the end. If you want, we can use CGROUP_SUBSYS_CFTYPES(mem_cgroup_subsys, tcp_files) as you suggested. tcp itself is always available if it is compiled in. Then in the future, if anyone cares about adding support for a protocol that may differ in that aspect, we can put the files nevertheless, and use ENOTSUPP as kame suggested for the swap accounting.

What's your take ?