
Subject: Re: [RFC][mm PATCH 1/8] Memory controller resource counters (v3)

Posted by [Paul Menage](#) on Fri, 20 Jul 2007 20:20:32 GMT

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

On 7/20/07, Balbir Singh <balbir@linux.vnet.ibm.com> wrote:

```
> +
> +ssize_t res_counter_read(struct res_counter *counter, int member,
> +      const char __user *userbuf, size_t nbytes, loff_t *pos)
> +{
> +    unsigned long *val;
> +    char buf[64], *s;
> +
> +    s = buf;
> +    val = res_counter_member(counter, member);
> +    s += sprintf(s, "%lu\n", *val);
> +    return simple_read_from_buffer((void __user *)userbuf, nbytes,
> +      pos, buf, s - buf);
> +}
```

I think it should be possible to use the support built-in to task containers to export a uint64 rather than having to create a separate function here.

```
> +
> +ssize_t res_counter_write(struct res_counter *counter, int member,
> +      const char __user *userbuf, size_t nbytes, loff_t *pos)
> +{
> +    int ret;
> +    char *buf, *end;
> +    unsigned long tmp, *val;
> +
> +    buf = kmalloc(nbytes + 1, GFP_KERNEL);
> +    ret = -ENOMEM;
> +    if (buf == NULL)
> +        goto out;
> +
> +    buf[nbytes] = '\0';
> +    ret = -EFAULT;
> +    if (copy_from_user(buf, userbuf, nbytes))
> +        goto out_free;
> +
> +    ret = -EINVAL;
> +    tmp = simple_strtoul(buf, &end, 10);
> +    if (*end != '\0')
> +        goto out_free;
> +
> +    val = res_counter_member(counter, member);
> +    *val = tmp;
```

```
> +    ret = nbytes;
> +out_free:
> +    kfree(buf);
> +out:
> +    return ret;
> +}
```

I should probably add a generic "write uint64" wrapper to task containers as well.

Paul

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