
Subject: Re: [RFC][PATCH 3/5] Container Freezer: Implement freezer cgroup subsystem

Posted by [Matt Helsley](#) on Wed, 30 Apr 2008 21:28:25 GMT

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

On Wed, 2008-04-23 at 23:47 -0700, Matt Helsley wrote:

<snip>

```
> +static ssize_t freezer_write(struct cgroup *cgroup,
> +    struct cftype *cft,
> +    struct file *file,
> +    const char __user *userbuf,
> +    size_t nbytes, loff_t *unused_ppos)
> +{
> +    char *buffer;
> +    int retval = 0;
> +    enum freezer_state goal_state;
> +
> +    if (nbytes >= PATH_MAX)
> +        return -E2BIG;
> +
> +    /* +1 for nul-terminator */
> +    buffer = kmalloc(nbytes + 1, GFP_KERNEL);
> +    if (buffer == NULL)
> +        return -ENOMEM;
> +
> +    if (copy_from_user(buffer, userbuf, nbytes)) {
> +        retval = -EFAULT;
> +        goto free_buffer;
> +    }
> +    buffer[nbytes] = 0; /* nul-terminate */
> +    strstrip(buffer);
> +    if (strcmp(buffer, "RUNNING") == 0)
> +        goal_state = STATE_RUNNING;
> +    else if (strcmp(buffer, "FROZEN") == 0)
> +        goal_state = STATE_FROZEN;
> +    else {
> +        retval = -EIO;
> +        goto free_buffer;
> +    }
> +
> +    cgroup_lock();
> +
> +    if (cgroup_is_removed(cgroup)) {
> +        retval = -ENODEV;
> +        goto unlock;
> +    }
```

I think this was copy/paste'd from `cgroup_common_file_write()` which modifies the cgroup hierarchy. However this function does not modify the cgroup hierarchy and we're not getting the cgroup from the task. So I don't think `cgroup_lock()/unlock()` are needed here. Paul, do you agree?

```
> + retval = freezer_freeze(cgroup, goal_state);  
> + if (retval == 0)  
> +   retval = nbytes;  
> +unlock:  
> + cgroup_unlock();  
> +free_buffer:  
> + kfree(buffer);  
> + return retval;  
> +}
```

Cheers,
-Matt Helsley

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