Subject: Re: [PATCH 4/4] The control group itself
Posted by Sukadev Bhattiprolu on Tue, 12 Feb 2008 07:42:17 GMT
View Forum Message <> Reply to Message

This patchset does fix the problem I was having before with null and
zero devices. Overall, it looks like pretty good.

I am still reviewing the patches. Just some nits I came across:

Pavel Emelianov [xemul@openvz.org] wrote:
| Each new group will have its own maps for char and block
| layers. The devices access list is tuned via the
| devices.permissions file. One may read from the file to get
| the configured state.
|
| The top container isn't initialized, so that the char
| and block layers will use the global maps to lookup
| their devices. I did that not to export the static maps
| to the outer world.
|
| Good news is that this patch now contains more comments
| and Documentation file :)
|
| Signed-off-by: Pavel Emelyanov <xemul@openvz.org>
|
| ---
|
| diff --git a/Documentation/controllers/devices.txt b/Documentation/controllers/devices.txt
| new file mode 100644
| index 0000000..dbd0c7a
| --- /dev/null
| +++ b/Documentation/controllers/devices.txt
| @@ -0,0 +1,61 @@
| +
| + Devices visibility controller
| +
| +This controller allows to tune the devices accessibility by tasks,
| +i.e. grant full access for /dev/null, /dev/zero etc, grant read-only
| +access to IDE devices and completely hide SCSI disks.
| +
| +Tasks still can call mknod to create device files, regardless of
| +whether the particular device is visible or accessible, but they
| +may not be able to open it later.
| +
| +This one hides under CONFIG_CGROUP_DEVS option.
| +
| +

| +Configuring
| +
| +The controller provides a single file to configure itself -- the
| +devices.permissions one. To change the accessibility level for some
| +device write the following string into it:
| +
| +[cb] <major>:(<minor>|*) [r-][w-]
| + ^ ^ ^
| + | | |
| + | | +--- access rights (1)
| + | |
| + | +-- device major and minor numbers (2)
| + |
| + +-- device type (character / block)
| +
| +1) The access rights set to '--' remove the device from the group's
| +access list, so that it will not even be shown in this file later.
| +
| +2) Setting the minor to '*' grants access to all the minors for
| +particular major.
| +
| +When reading from it, one may see something like
| +
| + c 1:5 rw
| + b 8:* r-
| +
| +Security issues, concerning who may grant access to what are governed
| +at the cgroup infrastructure level.
| +
| +
| +Examples:
| +
| +1. Grand full access to /dev/null

Grant.

| + # echo c 1:3 rw > /cgroups/<id>/devices.permissions
| +
| +2. Grant the read-only access to /dev/sda and partitions
| + # echo b 8:* r- > ...

This grants access to all scsi disks, sda..sdp and not just 'sda' right ?

| +
| +3. Change the /dev/null access to write-only
| + # echo c 1:3 -w > ...
| +
| +4. Revoke access to /dev/sda

```
| + # echo b 8:* -- > ...
| +
| +
| +  Written by Pavel Emelyanov <xemul@openvz.org>
| +
| diff --git a/fs/Makefile b/fs/Makefile
| index 7996220..5ad03be 100644
| --- a/fs/Makefile
| +++ b/fs/Makefile
| @@ -64,6 +64,8 @@ obj-y    += devpts/
|
|  obj-$(CONFIG_PROFILING)  += dcookies.o
|  obj-$(CONFIG_DLM)  += dlm/
| +
| +obj-$(CONFIG_CGROUP_DEVS) += devscontrol.o
|
|  # Do not add any filesystems before this line
|  obj-$(CONFIG_REISERFS_FS) += reiserfs/
| diff --git a/fs/devscontrol.c b/fs/devscontrol.c
| new file mode 100644
| index 0000000..48c5f69
| --- /dev/null
| +++ b/fs/devscontrol.c
| @@ -0,0 +1,314 @@
| +/*
| + * devscontrol.c - Device Controller
| + *
| + * Copyright 2007 OpenVZ SWsoft Inc
| + * Author: Pavel Emelyanov <xemul at openvz dot org>
| + *
| + * This program is free software; you can redistribute it and/or modify
| + * it under the terms of the GNU General Public License as published by
| + * the Free Software Foundation; either version 2 of the License, or
| + * (at your option) any later version.
| + *
| + * This program is distributed in the hope that it will be useful,
| + * but WITHOUT ANY WARRANTY; without even the implied warranty of
| + * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
| + * GNU General Public License for more details.
| + */
| +
| +#include <linux/cgroup.h>
| +#include <linux/cdev.h>
| +#include <linux/err.h>
| +#include <linux/devscontrol.h>
| +#include <linux/uaccess.h>
| +#include <linux/fs.h>
| +#include <linux/genhd.h>
```

```
| +
| +struct devs_cgroup {
| + /*
| +  * The subsys state to build into cgrous infrastructure
| +  */
```

... into cgroups

```
| + struct cgroup_subsys_state css;
| +
| + /*
| +  * The maps of character and block devices. They provide a
| +  * map from dev_t-s to struct cdev/gendisk. See fs/char_dev.c
| +  * and block/genhd.c to find out how the ->open() callbacks
| +  * work when opening a device.
| +  *
| +  * Each group will have its onw maps, and at the open()
```

own maps

```
| +  * time code will lookup in this map to get the device
| +  * and permissions by its dev_t.
| +  */
| + struct kobj_map *cdev_map;
| + struct kobj_map *bdev_map;
| +};
| +
| +static inline
| +struct devs_cgroup *css_to_devs(struct cgroup_subsys_state *css)
| +{
| + return container_of(css, struct devs_cgroup, css);
| +}
```

'devs' as prefix/suffix does not look very clear.

How about css_to_devs_cg() ? Similarly below for dev_cg_create(),
dev_cg_destroy() ?