

Re: Accessing disks via their serial numbers.

Re: Accessing disks via their serial numbers.

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2006-06/msg00130.html>

- *From:* "Poul-Henning Kamp" <phk@xxxxxxxxxxxxxxxx>
 - *Date:* Tue, 27 Jun 2006 18:43:19 +0000
-

In message <20060627.120424.-1625880159.imp@xxxxxxxxxx>, "M. Warner Losh" writes:

In message: <62122.1151427182@xxxxxxxxxxxxxxxxxxxx>
"Poul-Henning Kamp" <phk@xxxxxxxxxxxxxxxx> writes:

: There is no problem with fully enumerated devices, as long as
: they don't cause an explosion in the number of devices.

That's my view as well! We agree!

Well, not quite, but lets leave the deep UNIX philosophy questions behind for a moment.

: We have the devfs(8) rules for that.

And no way to audit them. The basic problem that I'd have in this specific case (serial numbers ala some variation of /dev/ad/ABCDEFGF) is that the system administrator cannot set and verify the permissions of the filename.

My take on this is different than yours.

I don't think we should allow names that are not "under control", and by not "under control" I mean device names which the device driver writer doesn't control or at the very least sanitize.

For instance, if you want to create names that match random strings, like the tape labels in your robot, the sensible and security concious device driver writer makes sure the names have a unique prefix:

/dev/tape/\$label

Re: Accessing disks via their serial numbers.

Re: Accessing disks via their serial numbers.

or similar, so that devfs(8) rules can be written in a surefire way.

A simple fix to this would be to have a sysctl that says to filter or allow magic characters in the label name.

I really don't think it should be optional. A vis(3) in some form should always happen.

: The reason why I am advocating using "on-demand" names for
: what Pawel is proposing is that way the names only exist
: if people ask for them, and only the names they ask for exists.

Making them on-demand makes it impossible to audit. Right now, if I'm worried about disk security, I can do:

That is why I'm not terribly keen on any kind of user-controlled /dev filenames.

: In addition to avoiding a wanton doubling of the geom mesh
: size (because he does it at the very bottom) that also
: adds significant flexibility and security to the table.

However, I'm not sure I understand the flexibility and security side of things. Properly written and implemented, I'm not sure how it affects security.

With an on-demand scheme, the scalability issue disappears, so we can add hard labels, soft labels, physical position (bus:id:lun), OEM labels, anything you can think of.

With a fully enumerated scheme, the scalability bites hard.

The only way to collapse these two views would be to allow drivers to register directories in DEVFS, so that they get to enumerate the issue when necessary, but without allocating cdevs for all the unnecessary nodes.

That is heading straight down the Linux procfs path.

If we want to go that way: fine, personally I think it leads to madness.

And please remember: This entire thing only comes up because

Re: Accessing disks via their serial numbers.

Re: Accessing disks via their serial numbers.

Pawel doesn't want to solve the problem correctly for g_label, this is the fall-back "quick&dirty" solution.

The correct solution is to give the users a reliable tool for stealing the necessary labelsector from the end of a filesystem.

—

Poul-Henning Kamp | UNIX since Zilog Zeus 3.20
phk@xxxxxxxxxxx | TCP/IP since RFC 956
FreeBSD committer | BSD since 4.3-tahoe
Never attribute to malice what can adequately be explained by incompetence.

freebsd-arch@xxxxxxxxxxx mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-arch>

To unsubscribe, send any mail to "freebsd-arch-unsubscribe@xxxxxxxxxxx"