

Re: Screw Linux, I'm going back to Windows!

Source: <http://www.derkeiler.com/Newsgroups/comp.os.linux.security/2002-10/4815.html>

From: Kasper Dupont (kasperd@daimi.au.dk)

Date: 09/14/02

From: Kasper Dupont <kasperd@daimi.au.dk>

Date: Sat, 14 Sep 2002 00:49:06 +0200

Fredderic wrote:

>

> *Another type is the CD that does it the same as most car CD's, using neither
> caddy nor tray. Also not very common, but they do exist.*

Yes, I didn't know that also existed on computers. My brother had a player with this kind of loader, it didn't work too well.

>

> *But in most cases, autoeject would be nice. I've lost count of the number
> of times I've needed to start up the computer just to eject a CD.*

I think most of us has experienced that too many times.

>

> *Urgh. I've always hated that under Windoze... I *DO* hope Gnome lets you
> disable that bad habit.*

I think so, but I don't remember how.

>

> > *With no CD in the drive, I have seen the following happen:*

> > *– Every 10 minutes, cron calls rmmmod -a.*

> > *– The module unloader decides the CD driver is no longer
> > needed, and unloads it.*

> > *– 10 seconds later gnome probes for a CD.*

> > *– The kernel doesn't have a driver for the device and
> > thus executes the module loader.*

> > *– The module loader loads the module.*

> > *– A message from the driver pops up on the screen.*

>

> *Gnome's probing of the drive doesn't count as module usage? I'd have*

> *thought that as long as Gnome's probing interval was less than the modules*

> *unused timeout, it wouldn't happen.*

Have been thinking the same too. It does behave the way I explained, I don't understand why.

comp.os.linux.security: Re: Screw Linux, I'm going back to Windows!

>
> > *But we need some kernel support. A virtual block device,
> > that can be bound to physical devices on command, might be
> > what we need. It should be possible to disassociate the
> > physical device from the block device even when it is
> > mounted, in that case our userspace application should be
> > notified if any process needs to access the drive.*
>
> *How about a symlink? I'd think that simply having a directory containing
> symlinks associating the drive's label and/or serial number to the physical
> drive, would do.*

That wouldn't work. Following the link is done at mount time,
not at every block access which is what we need.

> *If it's made and broken by the mount process, it should be
> pretty reliable.*

We need it to be changed even during the time the filesystem
stays mounted.

> *Even a virtual block device would need to be created and
> deleted as appropriate.*

Yes.

> *Unless you did something similar to the automounter.*

That is actually what I had in mind.

> *And even then, making it appear as a symlink should work just
> fine,*

No.

> *with the added bonus of making it easy to determine the hardware
> device we're referring to.*

Perhaps we can achieve that some other way.

>
> > *We also need automatic remounting between readonly and
> > readwrite. The remounting from readwrite to readonly can
> > be done periodically every second, though help from the
> > kernel would be slightly more efficient. But the other
> > way is needed on demand. Of course this feature must be
> > optional, we still want to be able to make a usual ro
> > mount.*
>
> *Indeed. Though if we're going to jump between readonly and readwrite all
> the time (which would require kernel support), may as well just make a hybrid*

Re: Screw Linux, I'm going back to Windows!

comp.os.linux.security: Re: Screw Linux, I'm going back to Windows!

- > *writeback mode -- someone suggested "semi-sync", or being a C programmer, I*
- > *suggested "volatile".*

You are right. That makes sense. The ability of actually being able to do something in userspace during the transition could come in handy. But having the kernel automatically finish writings is cleaner than trying to enforce it by remounting readonly. Still the changing of the dirty flag in the superblock at start/end of writes is a part of the feature. (Yes AmigaOS does that.)

- > *Regardless of name, the concept is simply to write*
- > *all the appropriate metadata to "close" the filesystem right after all*
- > *available data writes have been completed.*

Yes.

- > *Presumably should new data*
- > *become available for writing before the closure is completed, the new data*
- > *would be written immediately and the process of closing the filesystem*
- > *simply restarted right after.*

Yes, that is more or less how it should work. The needed operations at start and end of writes are cheap, but not free. Doing it for every single block would be too expensive. We need to take advantage of a filesystem staying busy.

- >
- > *It would, though, be advantageous for the system to be able to detect disk*
- > *insertion and removal, otherwise it would be necessary for the driver to*
- > *verify that the correct disk was still in the drive before commencing a*
- > *write. Things could get ugly otherwise.*

Yes.

--
Kasper Dupont -- der bruger for meget tid på usenet.
For sending spam use <mailto:aaarep@daimi.au.dk>
or <mailto:mcxumhvenwblvtl@skrammel.yaboo.dk>

- ***Next message:*** [Luke Vogel: "Re: KOREAN SPAM: HOWTO deal with it???"](#)
- ***Previous message:*** [: "Re: KOREAN SPAM: HOWTO deal with it???"](#)
- ***In reply to:*** [: "Re: Screw Linux, I'm going back to Windows!"](#)
- ***Next in thread:*** [: "Re: Screw Linux, I'm going back to Windows! – STOP X-POSTING THIS!!!!!!!!!!!!!!!"](#)
- ***Reply:*** [: "Re: Screw Linux, I'm going back to Windows! – STOP X-POSTING THIS!!!!!!!!!!!!!!!"](#)
- ***Reply:*** [Fredderic: "Re: Screw Linux, I'm going back to Windows!"](#)
- ***Messages sorted by:*** [\[date \] \[thread \] \[subject \] \[author \] \[attachment \]](#)