

Re: Fw: [PATCH 2.6.16-rc1-git4] accessfs: a permission managing filesystem

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-01/msg11450.html>

- *From:* James Morris <jmorris@xxxxxxxxx>
 - *Date:* Tue, 31 Jan 2006 10:50:12 -0500 (EST)
-

> Can you please include this patch in -mm, to give it wider testing?
>
> Accessfs is a permission managing filesystem. It allows to control
> access to system resources, based on file permissions. It also
> includes two modules. One module allows granting capabilities based
> on user-/groupid. The second module allows to grant access to lower
> numbered IP ports based on user-/groupid.

I don't think this code is suitable for mainline inclusion.

The kernel already a mechanism for implementing extended security models for networking in SELinux, which is far more general and also provides a system-wide approach where all security-relevant objects and subjects and the interactions between them are controlled.

Also, I think capabilities are inherently problematic in that here, they introduce a mechanism for unbounded privilege escalation. Your security model is granting privileges to non-root processes, but not then providing any means to contain these privileges.

There are also a lot of hard-coded uid==0 assumptions in userspace which will break badly once you start handing out privileges to uid!=0 processes. With SELinux we see a lot of these userspace assumptions, although, because SELinux is restrictive (i.e. only further restricts access), they do not lead to privilege escalation.

This scheme does not integrate well with SELinux, which will be able to reject access requests before accessfs sees them; while accessfs will be able to reject access requests that SELinux has already granted. These mechanism are also not aware of each other from a policy point of view.

Rather than proliferating new security models in the kernel such as these, which tend to be fairly narrowly focused, I think it would be better to look at how SELinux (which itself is a general purpose security framework) can be adapted to the same or similar purpose.

In this case, aside from the permissive nature of your security model

Re: Fw: [PATCH 2.6.16-rc1-git4] accessfs: a permission managing filesystem

(i.e. granting permissions instead of just restricting them), it seems that accessfs is primarily a policy interface. It may be possible to achieve something very similar by creating a highly abstracted interface to SELinux policy. Most or even all of which I think could be done in userspace.

– James

--

James Morris

<jmorris@xxxxxxxxxx>

–

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>

-
- Prev by Date: **[Re: Rescan SCSI Bus without /proc/scsi?](#)**
 - Next by Date: **[Re: \[PATCH 2.6.15-git9a\] aoe \[1/1\]: do not stop retransmit timer when device goes down](#)**
 - Previous by thread: **[\[PATCH\] OOM kill: children accounting](#)**
 - Next by thread: **[Dont use num_processors as index to generate logical cpu# in x86_64](#)**
 - Index(es):
 - ◆ **[Date](#)**
 - ◆ **[Thread](#)**