
Subject: Re: [RFC][PATCH 0/5] uts namespaces: Introduction
Posted by [ebiederm](#) on Fri, 07 Apr 2006 19:39:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Serge E. Hallyn" <serue@us.ibm.com> writes:

> Quoting Eric W. Biederman (ebiederm@xmission.com):
>> "Serge E. Hallyn" <serue@us.ibm.com> writes:
>>
>> > Introduce utsname namespaces. Instead of a single system_utsname
>> > containing hostname domainname etc, a process can request it's
>> > copy of the uts info to be cloned. The data will be copied from
>> > it's original, but any further changes will not be seen by processes
>> > which are not it's children, and vice versa.
>> >
>> > This is useful, for instance, for vserver/openvz, which can now clone
>> > a new uts namespace for each new virtual server.
>> >
>> > This patchset is based on Kirill Korotaev's Mar 24 submission, taking
>> > comments (in particular from James Morris and Eric Biederman) into
>> > account.
>> >
>> > Some performance results are attached. I was mainly curious whether
>> > it would be worth putting the task_struct->uts_ns pointer inside
>> > a #ifdef CONFIG_UTS_NS. The result show that leaving it in when
>> > CONFIG_UTS_NS=n has negligible performance impact, so that is the
>> > approach this patch takes.
>>
>> Ok. This looks like the best version so far.
>>
>> I like the utsname() function thing to shorten the
>> idiom of current->uts_ns->name.
>>
>> We probably want to introduce utsname() and an init_utsname()
>> before any of the other changes, and then perform the substitutions,
>
> This is the same as what Sam is saying, right? Just making sure I
> understand.

Yes.

>> before we actually change the code so the patchset can make it
>> through a git-bisect. This will also allows for something
>
> Ok, I've finally got the rest of git doing my bidding, I'll go read
> up on git-bisect.

Basically git-bisect is an automated binary search through patches

to help find bugs. If you ever can't compile at an intermediate patch git-bisect and other people walking through the patches looking for bugs won't like it.

It's not mandatory that you never break anything in a patchset, but it is much friendlier when you can avoid breakage.

Eric
