
Subject: Re: [RFC][PATCH 4/5] utsname namespaces: sysctl hack
Posted by [dev](#) on Wed, 19 Apr 2006 15:10:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

Serge,

can we do nothing with sysctls at this moment, instead of committing hacks?

Thanks,
Kirill

> Sysctl uts patch. This clearly will need to be done another way, but
> since sysctl itself needs to be container aware, 'the right thing' is
> a separate patchset.

>

> Signed-off-by: Serge E. Hallyn <serue@us.ibm.com>

> ---

> kernel/sysctl.c | 38 ++++++-----

> 1 files changed, 28 insertions(+), 10 deletions(-)

>

> 40f7e1320c82efb6e875fc3bf44408cdfd093f21

> diff --git a/kernel/sysctl.c b/kernel/sysctl.c

> index e82726f..c2b18ef 100644

> --- a/kernel/sysctl.c

> +++ b/kernel/sysctl.c

> @@ -233,8 +233,8 @@ static ctl_table kern_table[] = {

> {

> .ctl_name = KERN_OSTYPE,

> .procname = "ostype",

> - .data = system_utsname.sysname,

> - .maxlen = sizeof(system_utsname.sysname),

> + .data = init_uts_ns.name.sysname,

> + .maxlen = sizeof(init_uts_ns.name.sysname),

> .mode = 0444,

> .proc_handler = &proc_doutsstring,

> .strategy = &sysctl_string,

> @@ -242,8 +242,8 @@ static ctl_table kern_table[] = {

> {

> .ctl_name = KERN_OSRELEASE,

> .procname = "osrelease",

> - .data = system_utsname.release,

> - .maxlen = sizeof(system_utsname.release),

> + .data = init_uts_ns.name.release,

> + .maxlen = sizeof(init_uts_ns.name.release),

> .mode = 0444,

> .proc_handler = &proc_doutsstring,

> .strategy = &sysctl_string,

> @@ -251,8 +251,8 @@ static ctl_table kern_table[] = {

```

> {
> .ctl_name = KERN_VERSION,
> .procname = "version",
> - .data = system_utsname.version,
> - .maxlen = sizeof(system_utsname.version),
> + .data = init_uts_ns.name.version,
> + .maxlen = sizeof(init_uts_ns.name.version),
> .mode = 0444,
> .proc_handler = &proc_doutsstring,
> .strategy = &sysctl_string,
> @@ -260,8 +260,8 @@ static ctl_table kern_table[] = {
> {
> .ctl_name = KERN_NODENAME,
> .procname = "hostname",
> - .data = system_utsname.nodename,
> - .maxlen = sizeof(system_utsname.nodename),
> + .data = init_uts_ns.name.nodename,
> + .maxlen = sizeof(init_uts_ns.name.nodename),
> .mode = 0644,
> .proc_handler = &proc_doutsstring,
> .strategy = &sysctl_string,
> @@ -269,8 +269,8 @@ static ctl_table kern_table[] = {
> {
> .ctl_name = KERN_DOMAINNAME,
> .procname = "domainname",
> - .data = system_utsname.domainname,
> - .maxlen = sizeof(system_utsname.domainname),
> + .data = init_uts_ns.name.domainname,
> + .maxlen = sizeof(init_uts_ns.name.domainname),
> .mode = 0644,
> .proc_handler = &proc_doutsstring,
> .strategy = &sysctl_string,
> @@ -1619,6 +1619,24 @@ static int proc_doutsstring(ctl_table *t
> {
> int r;
>
> + switch (table->ctl_name) {
> + case KERN_OSTYPE:
> + table->data = utsname()->sysname;
> + break;
> + case KERN_OSRELEASE:
> + table->data = utsname()->release;
> + break;
> + case KERN_VERSION:
> + table->data = utsname()->version;
> + break;
> + case KERN_NODENAME:
> + table->data = utsname()->nodename;

```

```
> + break;
> + case KERN_DOMAINNAME:
> + table->data = utsname()->domainname;
> + break;
> + }
> +
> if (!write) {
>   down_read(&uts_sem);
>   r=proc_dostring(table,0,filp,buffer,lenp, ppos);
```
