
Subject: Re: [PATCH 4/9] namespaces: utsname: switch to using uts namespaces
Posted by [rdunlap](#) on Fri, 19 May 2006 00:02:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Thu, 18 May 2006 10:49:36 -0500 Serge E. Hallyn wrote:

```
> Replace references to system_utsname to the per-process uts namespace
> where appropriate. This includes things like uname.
>
> Changes: Per Eric Biederman's comments, use the per-process uts namespace
> for ELF_PLATFORM, sunrpc, and parts of net/ipv4/ipconfig.c
>
> Signed-off-by: Serge E. Hallyn <serue@us.ibm.com>
>
> ---
>
> 9ee063adf4d2287583dbb0a71d1d5f80d7ae011f
> diff --git a/arch/i386/kernel/sys_i386.c b/arch/i386/kernel/sys_i386.c
> index 8fdb1fb..4af731d 100644
> --- a/arch/i386/kernel/sys_i386.c
> +++ b/arch/i386/kernel/sys_i386.c
> @@ -210,7 +210,7 @@ asmlinkage int sys_uname(struct old_utsn
> if (!name)
> return -EFAULT;
> down_read(&uts_sem);
> - err=copy_to_user(name, &system_utsname, sizeof (*name));
> + err=copy_to_user(name, utsname(), sizeof (*name));
```

It would be really nice if you would fix spacing while you are here,
like a space a each side of '='.

and a space after ',' in the function calls below.

```
> up_read(&uts_sem);
> return err?-EFAULT:0;
> }
> @@ -226,15 +226,15 @@ asmlinkage int sys_olduname(struct oldol
>
> down_read(&uts_sem);
>
> - error = __copy_to_user(&name->sysname,&system_utsname.sysname,__OLD_UTS_LEN);
> + error = __copy_to_user(&name->sysname,&utsname()->sysname,__OLD_UTS_LEN);
> error |= __put_user(0,name->sysname+__OLD_UTS_LEN);
> - error |=
__copy_to_user(&name->nodename,&system_utsname.nodename,__OLD_UTS_LEN);
> + error |= __copy_to_user(&name->nodename,&utsname()->nodename,__OLD_UTS_LEN);
> error |= __put_user(0,name->nodename+__OLD_UTS_LEN);
> - error |= __copy_to_user(&name->release,&system_utsname.release,__OLD_UTS_LEN);
```

```

> + error |= __copy_to_user(&name->release,&utsname()->release,__OLD_UTS_LEN);
> error |= __put_user(0,name->release+__OLD_UTS_LEN);
> - error |= __copy_to_user(&name->version,&system_utsname.version,__OLD_UTS_LEN);
> + error |= __copy_to_user(&name->version,&utsname()->version,__OLD_UTS_LEN);
> error |= __put_user(0,name->version+__OLD_UTS_LEN);
> - error |= __copy_to_user(&name->machine,&system_utsname.machine,__OLD_UTS_LEN);
> + error |= __copy_to_user(&name->machine,&utsname()->machine,__OLD_UTS_LEN);
> error |= __put_user(0,name->machine+__OLD_UTS_LEN);
>
> up_read(&uts_sem);
> diff --git a/arch/m32r/kernel/sys_m32r.c b/arch/m32r/kernel/sys_m32r.c
> index 670cb49..11412c0 100644
> --- a/arch/m32r/kernel/sys_m32r.c
> +++ b/arch/m32r/kernel/sys_m32r.c
> @@ -206,7 +206,7 @@ asmlinkage int sys_uname(struct old_utsn
> if (!name)
> return -EFAULT;
> down_read(&uts_sem);
> - err=copy_to_user(name, &system_utsname, sizeof (*name));
> + err=copy_to_user(name, utsname(), sizeof (*name));

```

spacing

```

> up_read(&uts_sem);
> return err?-EFAULT:0;
> }
> diff --git a/arch/mips/kernel/linux32.c b/arch/mips/kernel/linux32.c
> index a7d2bb3..66f999b 100644
> --- a/arch/mips/kernel/linux32.c
> +++ b/arch/mips/kernel/linux32.c
> @@ -1040,7 +1040,7 @@ asmlinkage long sys32_newuname(struct ne
> int ret = 0;
>
> down_read(&uts_sem);
> - if (copy_to_user(name,&system_utsname,sizeof *name))
> + if (copy_to_user(name,utsname(),sizeof *name))

```

spacing

```

> ret = -EFAULT;
> up_read(&uts_sem);
>
> diff --git a/arch/mips/kernel/syscall.c b/arch/mips/kernel/syscall.c
> index 2aeaa2f..8b13d57 100644
> --- a/arch/mips/kernel/syscall.c
> +++ b/arch/mips/kernel/syscall.c
> @@ -232,7 +232,7 @@ out:
> */

```

```

> asmlinkage int sys_uname(struct old_utsname __user * name)
> {
> - if (name && !copy_to_user(name, &system_utsname, sizeof (*name)))
> + if (name && !copy_to_user(name, utsname(), sizeof (*name)))

```

OK, here's my big comment/question. I want to see <nodename> increased to 256 bytes (per current POSIX), so each field of struct <variant>_utsname needs be copied individually (I think) instead of doing a single struct copy.

I've been working on this for the past few weeks (among other things). Sorry about the timing.

I could send patches for this against mainline in a few days, but I'll be glad to listen to how it would be easiest for all of us to handle.

I'm probably a little over half done with my patches.

They will end up adding a lib/utsname.c that has functions for:

```

put_oldold_uname() // to user
put_old_uname() // to user
put_new_uname() // to user
put_posix_uname() // to user

```

```

> return 0;
> return -EFAULT;
> }
> @@ -249,15 +249,15 @@ asmlinkage int sys_olduname(struct oldol
> if (!access_ok(VERIFY_WRITE,name,sizeof(struct oldold_utsname)))
> return -EFAULT;
>
> - error = __copy_to_user(&name->sysname,&system_utsname.sysname,__OLD_UTS_LEN);
> + error = __copy_to_user(&name->sysname,&utsname()->sysname,__OLD_UTS_LEN);
> error -= __put_user(0,name->sysname+__OLD_UTS_LEN);
> - error -=
__copy_to_user(&name->nodename,&system_utsname.nodename,__OLD_UTS_LEN);
> + error -= __copy_to_user(&name->nodename,&utsname()->nodename,__OLD_UTS_LEN);
> error -= __put_user(0,name->nodename+__OLD_UTS_LEN);
> - error -= __copy_to_user(&name->release,&system_utsname.release,__OLD_UTS_LEN);
> + error -= __copy_to_user(&name->release,&utsname()->release,__OLD_UTS_LEN);
> error -= __put_user(0,name->release+__OLD_UTS_LEN);
> - error -= __copy_to_user(&name->version,&system_utsname.version,__OLD_UTS_LEN);
> + error -= __copy_to_user(&name->version,&utsname()->version,__OLD_UTS_LEN);
> error -= __put_user(0,name->version+__OLD_UTS_LEN);
> - error -= __copy_to_user(&name->machine,&system_utsname.machine,__OLD_UTS_LEN);
> + error -= __copy_to_user(&name->machine,&utsname()->machine,__OLD_UTS_LEN);

```

```
> error = __put_user(0,name->machine+__OLD_UTS_LEN);
> error = error ? -EFAULT : 0;
```

spaces

```
> diff --git a/arch/parisc/hpux/sys_hpux.c b/arch/parisc/hpux/sys_hpux.c
> index 05273cc..9fc2c08 100644
> --- a/arch/parisc/hpux/sys_hpux.c
> +++ b/arch/parisc/hpux/sys_hpux.c
> @@ -266,15 +266,15 @@ static int hpux_uname(struct hpux_utsnam
>
> down_read(&uts_sem);
>
> - error = __copy_to_user(&name->sysname,&system_utsname.sysname,HPUX_UTSLEN-1);
> + error = __copy_to_user(&name->sysname,&utsname()->sysname,HPUX_UTSLEN-1);
> error |= __put_user(0,name->sysname+HPUX_UTSLEN-1);
> - error |=
__copy_to_user(&name->nodename,&system_utsname.nodename,HPUX_UTSLEN-1);
> + error |= __copy_to_user(&name->nodename,&utsname()->nodename,HPUX_UTSLEN-1);
> error |= __put_user(0,name->nodename+HPUX_UTSLEN-1);
> - error |= __copy_to_user(&name->release,&system_utsname.release,HPUX_UTSLEN-1);
> + error |= __copy_to_user(&name->release,&utsname()->release,HPUX_UTSLEN-1);
> error |= __put_user(0,name->release+HPUX_UTSLEN-1);
> - error |= __copy_to_user(&name->version,&system_utsname.version,HPUX_UTSLEN-1);
> + error |= __copy_to_user(&name->version,&utsname()->version,HPUX_UTSLEN-1);
> error |= __put_user(0,name->version+HPUX_UTSLEN-1);
> - error |= __copy_to_user(&name->machine,&system_utsname.machine,HPUX_UTSLEN-1);
> + error |= __copy_to_user(&name->machine,&utsname()->machine,HPUX_UTSLEN-1);
> error |= __put_user(0,name->machine+HPUX_UTSLEN-1);
```

spacing

```
> up_read(&uts_sem);

> diff --git a/arch/sh/kernel/sys_sh.c b/arch/sh/kernel/sys_sh.c
> index 917b2f3..e4966b2 100644
> --- a/arch/sh/kernel/sys_sh.c
> +++ b/arch/sh/kernel/sys_sh.c
> @@ -267,7 +267,7 @@ asmlinkage int sys_uname(struct old_utsn
> if (!name)
> return -EFAULT;
> down_read(&uts_sem);
> - err=copy_to_user(name, &system_utsname, sizeof (*name));
> + err=copy_to_user(name, utsname(), sizeof (*name));
```

spacing

```

> up_read(&uts_sem);
> return err?-EFAULT:0;
> }
> diff --git a/arch/sh64/kernel/sys_sh64.c b/arch/sh64/kernel/sys_sh64.c
> index 58ff7d5..a8dc88c 100644
> --- a/arch/sh64/kernel/sys_sh64.c
> +++ b/arch/sh64/kernel/sys_sh64.c
> @@ -279,7 +279,7 @@ asmlinkage int sys_uname(struct old_utsn
> if (!name)
> return -EFAULT;
> down_read(&uts_sem);
> - err=copy_to_user(name, &system_utsname, sizeof (*name));
> + err=copy_to_user(name, utsname(), sizeof (*name));

```

spacing

```

> up_read(&uts_sem);
> return err?-EFAULT:0;
> }
> diff --git a/arch/sparc/kernel/sys_sunos.c b/arch/sparc/kernel/sys_sunos.c
> index 288de27..9f9206f 100644
> --- a/arch/sparc/kernel/sys_sunos.c
> +++ b/arch/sparc/kernel/sys_sunos.c
> @@ -483,13 +483,13 @@ asmlinkage int sunos_uname(struct sunos_
> {
> int ret;
> down_read(&uts_sem);
> - ret = copy_to_user(&name->sname[0], &system_utsname.sysname[0], sizeof(name->sname) -
1);
> + ret = copy_to_user(&name->sname[0], &utsname()->sysname[0], sizeof(name->sname) - 1);
> if (!ret) {
> - ret |= __copy_to_user(&name->nname[0], &system_utsname.nodename[0],
sizeof(name->nname) - 1);
> + ret |= __copy_to_user(&name->nname[0], &utsname()->nodename[0], sizeof(name->nname)
- 1);
> ret |= __put_user('\0', &name->nname[8]);
> - ret |= __copy_to_user(&name->rel[0], &system_utsname.release[0], sizeof(name->rel) - 1);
> - ret |= __copy_to_user(&name->ver[0], &system_utsname.version[0], sizeof(name->ver) - 1);
> - ret |= __copy_to_user(&name->mach[0], &system_utsname.machine[0], sizeof(name->mach)
- 1);
> + ret |= __copy_to_user(&name->rel[0], &utsname()->release[0], sizeof(name->rel) - 1);
> + ret |= __copy_to_user(&name->ver[0], &utsname()->version[0], sizeof(name->ver) - 1);
> + ret |= __copy_to_user(&name->mach[0], &utsname()->machine[0], sizeof(name->mach) - 1);

```

Oh, please limit to 80 column width while you are here (+ other places).

```

> }
> up_read(&uts_sem);

```

```

> return ret ? -EFAULT : 0;

> diff --git a/arch/um/kernel/syscall_kern.c b/arch/um/kernel/syscall_kern.c
> index 37d3978..d90e9ed 100644
> --- a/arch/um/kernel/syscall_kern.c
> +++ b/arch/um/kernel/syscall_kern.c
> @@ -110,7 +110,7 @@ long sys_uname(struct old_utsname __user
> if (!name)
> return -EFAULT;
> down_read(&uts_sem);
> - err=copy_to_user(name, &system_utsname, sizeof (*name));
> + err=copy_to_user(name, utsname(), sizeof (*name));

```

spacing

```

> up_read(&uts_sem);
> return err?-EFAULT:0;
> }
> @@ -126,19 +126,19 @@ long sys_olduname(struct oldold_utsname
>
> down_read(&uts_sem);
>
> - error = __copy_to_user(&name->sysname,&system_utsname.sysname,
> + error = __copy_to_user(&name->sysname,&utsname()->sysname,
> __OLD_UTS_LEN);
> error |= __put_user(0,name->sysname+__OLD_UTS_LEN);
> - error |= __copy_to_user(&name->nodename,&system_utsname.nodename,
> + error |= __copy_to_user(&name->nodename,&utsname()->nodename,
> __OLD_UTS_LEN);
> error |= __put_user(0,name->nodename+__OLD_UTS_LEN);
> - error |= __copy_to_user(&name->release,&system_utsname.release,
> + error |= __copy_to_user(&name->release,&utsname()->release,
> __OLD_UTS_LEN);
> error |= __put_user(0,name->release+__OLD_UTS_LEN);
> - error |= __copy_to_user(&name->version,&system_utsname.version,
> + error |= __copy_to_user(&name->version,&utsname()->version,
> __OLD_UTS_LEN);
> error |= __put_user(0,name->version+__OLD_UTS_LEN);
> - error |= __copy_to_user(&name->machine,&system_utsname.machine,
> + error |= __copy_to_user(&name->machine,&utsname()->machine,
> __OLD_UTS_LEN);
> error |= __put_user(0,name->machine+__OLD_UTS_LEN);

```

spacing

```

> diff --git a/arch/x86_64/ia32/sys_ia32.c b/arch/x86_64/ia32/sys_ia32.c
> index f182b20..6e0a19d 100644
> --- a/arch/x86_64/ia32/sys_ia32.c

```

```

> +++ b/arch/x86_64/ia32/sys_ia32.c
> @@ -801,13 +801,13 @@ asmlinkage long sys32_olduname(struct ol
>
>   down_read(&uts_sem);
>
> - error = __copy_to_user(&name->sysname,&system_utsname.sysname,__OLD_UTS_LEN);
> + error = __copy_to_user(&name->sysname,&utsname()->sysname,__OLD_UTS_LEN);
>   __put_user(0,name->sysname+__OLD_UTS_LEN);
> - __copy_to_user(&name->nodename,&system_utsname.nodename,__OLD_UTS_LEN);
> + __copy_to_user(&name->nodename,&utsname()->nodename,__OLD_UTS_LEN);
>   __put_user(0,name->nodename+__OLD_UTS_LEN);
> - __copy_to_user(&name->release,&system_utsname.release,__OLD_UTS_LEN);
> + __copy_to_user(&name->release,&utsname()->release,__OLD_UTS_LEN);
>   __put_user(0,name->release+__OLD_UTS_LEN);
> - __copy_to_user(&name->version,&system_utsname.version,__OLD_UTS_LEN);
> + __copy_to_user(&name->version,&utsname()->version,__OLD_UTS_LEN);
>   __put_user(0,name->version+__OLD_UTS_LEN);

```

spacing

```

> {
>   char *arch = "x86_64";
> @@ -830,7 +830,7 @@ long sys32_uname(struct old_utsname __us
>   if (!name)
>     return -EFAULT;
>   down_read(&uts_sem);
> - err=copy_to_user(name, &system_utsname, sizeof (*name));
> + err=copy_to_user(name, utsname(), sizeof (*name));

```

ditto

```

> up_read(&uts_sem);
> if (personality(current->personality) == PER_LINUX32)
>   err |= copy_to_user(&name->machine, "i686", 5);

> diff --git a/fs/cifs/connect.c b/fs/cifs/connect.c
> index d2ec806..b6c0886 100644
> --- a/fs/cifs/connect.c
> +++ b/fs/cifs/connect.c
> @@ -765,12 +765,12 @@ cifs_parse_mount_options(char *options,
>   separator[1] = 0;
>
>   memset(vol->source_rfc1001_name,0x20,15);
> - for(i=0;i < strlen(system_utsname.nodename,15);i++) {
> + for(i=0;i < strlen(utsname()->nodename,15);i++) {

```

spacing

```

>  /* does not have to be a perfect mapping since the field is
>  informational, only used for servers that do not support
>  port 445 and it can be overridden at mount time */
>  vol->source_rfc1001_name[i] =
> -  toupper(system_utsname.nodename[i]);
> +  toupper(utsname()->nodename[i]);
>  }
>  vol->source_rfc1001_name[15] = 0;
>  /* null target name indicates to use *SMBSEVR default called name

> diff --git a/kernel/sys.c b/kernel/sys.c
> index 0b6ec0e..bcaa48e 100644
> --- a/kernel/sys.c
> +++ b/kernel/sys.c
> @@ -1671,7 +1671,7 @@ asmlinkage long sys_newuname(struct new_
>  int errno = 0;
>
>  down_read(&uts_sem);
> - if (copy_to_user(name,&system_utsname,sizeof *name))
> + if (copy_to_user(name,utsname(),sizeof *name))

```

spacing

```

>  errno = -EFAULT;
>  up_read(&uts_sem);
>  return errno;

```

Thanks,

~Randy