
Subject: [PATCH 4/4] Virtualization/containers: uts name
Posted by [Kirill Korotaev](#) on Mon, 06 Feb 2006 22:20:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

This patch virtualizes uts name.
main changes are done in container.h, uts_name.h,
all other places are just replacement of system_utsname with uts_name.

Signed-Off-By: Kirill Korotaev <dev@openvz.org>

Kirill

```
--- ./arch/alpha/kernel/osf_sys.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/alpha/kernel/osf_sys.c 2006-02-07 01:18:50.000000000 +0300
@@ -402,15 +402,15 @@ osf_utsname(char __user *name)
```

```
    down_read(&uts_sem);
    error = -EFAULT;
- if (copy_to_user(name + 0, system_utsname.sysname, 32))
+ if (copy_to_user(name + 0, uts_name.sysname, 32))
    goto out;
- if (copy_to_user(name + 32, system_utsname.nodename, 32))
+ if (copy_to_user(name + 32, uts_name.nodename, 32))
    goto out;
- if (copy_to_user(name + 64, system_utsname.release, 32))
+ if (copy_to_user(name + 64, uts_name.release, 32))
    goto out;
- if (copy_to_user(name + 96, system_utsname.version, 32))
+ if (copy_to_user(name + 96, uts_name.version, 32))
    goto out;
- if (copy_to_user(name + 128, system_utsname.machine, 32))
+ if (copy_to_user(name + 128, uts_name.machine, 32))
    goto out;

    error = 0;
@@ -449,8 +449,8 @@ osf_getdomainname(char __user *name, int

    down_read(&uts_sem);
    for (i = 0; i < len; ++i) {
- __put_user(system_utsname.domainname[i], name + i);
- if (system_utsname.domainname[i] == '\0')
+ __put_user(uts_name.domainname[i], name + i);
+ if (uts_name.domainname[i] == '\0')
        break;
    }
    up_read(&uts_sem);
@@ -608,11 +608,11 @@ asmlinkage long
osf_sysinfo(int command, char __user *buf, long count)
```

```

{
    static char * sysinfo_table[] = {
-   system_utsname.sysname,
-   system_utsname.nodename,
-   system_utsname.release,
-   system_utsname.version,
-   system_utsname.machine,
+   uts_name.sysname,
+   uts_name.nodename,
+   uts_name.release,
+   uts_name.version,
+   uts_name.machine,
        "alpha", /* instruction set architecture */
        "dummy", /* hardware serial number */
        "dummy", /* hardware manufacturer */
--- ./arch/i386/kernel/sys_i386.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/i386/kernel/sys_i386.c 2006-02-07 01:18:50.000000000 +0300
@@ -217,7 +217,7 @@ asmlinkage int sys_utsname(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, &uts_name, sizeof (*name));
    up_read(&uts_sem);
    return err?-EFAULT:0;
}
@@ -233,15 +233,15 @@ 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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.machine,__OLD_UTS_LEN);
    error |= __put_user(0,name->machine+__OLD_UTS_LEN);

    up_read(&uts_sem);
--- ./arch/m32r/kernel/sys_m32r.c.utsnamex 2006-02-07 01:18:42.000000000 +0300

```

```

+++ ./arch/m32r/kernel/sys_m32r.c 2006-02-07 01:18:50.000000000 +0300
@@ -199,7 +199,7 @@ 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, &uts_name, sizeof (*name));
    up_read(&uts_sem);
    return err?-EFAULT:0;
}
--- ./arch/mips/kernel/linux32.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/mips/kernel/linux32.c 2006-02-07 01:18:50.000000000 +0300
@@ -1150,7 +1150,7 @@ 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,&uts_name,sizeof *name))
    ret = -EFAULT;
    up_read(&uts_sem);

--- ./arch/mips/kernel/syscall.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/mips/kernel/syscall.c 2006-02-07 01:18:50.000000000 +0300
@@ -229,7 +229,7 @@ out:
    */
    asmlinkage int sys_uname(struct old_utsname * name)
    {
- if (name && !copy_to_user(name, &system_utsname, sizeof (*name)))
+ if (name && !copy_to_user(name, &uts_name, sizeof (*name)))
        return 0;
        return -EFAULT;
    }
@@ -246,15 +246,15 @@ 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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.machine,__OLD_UTS_LEN);
  error = __put_user(0,name->machine+__OLD_UTS_LEN);
  error = error ? -EFAULT : 0;

```

```

@@ -290,10 +290,10 @@ asmlinkage int _sys_sysmips(int cmd, lon
  return -EFAULT;

```

```

  down_write(&uts_sem);
- strncpy(system_utsname.nodename, nodename, len);
+ strncpy(uts_name.nodename, nodename, len);
  nodename[__NEW_UTS_LEN] = '\0';
- strncpy(system_utsname.nodename, nodename,
-       sizeof(system_utsname.nodename));
+ strncpy(uts_name.nodename, nodename,
+       sizeof(uts_name.nodename));
  up_write(&uts_sem);
  return 0;
}

```

```

--- ./arch/mips/kernel/sysirix.c.utsnamex 2006-02-07 01:18:42.000000000 +0300

```

```

+++ ./arch/mips/kernel/sysirix.c 2006-02-07 01:18:50.000000000 +0300

```

```

@@ -904,7 +904,7 @@ asmlinkage int irix_getdomainname(char _
  down_read(&uts_sem);
  if (len > __NEW_UTS_LEN)
    len = __NEW_UTS_LEN;
- err = copy_to_user(name, system_utsname.domainname, len) ? -EFAULT : 0;
+ err = copy_to_user(name, uts_name.domainname, len) ? -EFAULT : 0;
  up_read(&uts_sem);

```

```

  return err;
@@ -1147,11 +1147,11 @@ struct iuname {
  asmlinkage int irix_uname(struct iuname __user *buf)
  {
    down_read(&uts_sem);
- if (copy_from_user(system_utsname.sysname, buf->sysname, 65)
-     || copy_from_user(system_utsname.nodename, buf->nodename, 65)
-     || copy_from_user(system_utsname.release, buf->release, 65)
-     || copy_from_user(system_utsname.version, buf->version, 65)
-     || copy_from_user(system_utsname.machine, buf->machine, 65)) {
+ if (copy_from_user(uts_name.sysname, buf->sysname, 65)
+     || copy_from_user(uts_name.nodename, buf->nodename, 65)
+     || copy_from_user(uts_name.release, buf->release, 65)
+     || copy_from_user(uts_name.version, buf->version, 65)
+     || copy_from_user(uts_name.machine, buf->machine, 65)) {
    return -EFAULT;
  }
  up_read(&uts_sem);

```

```

--- ./arch/parisc/hpux/sys_hpux.c.utsnamex 2006-02-07 01:18:42.000000000 +0300

```

```

+++ ./arch/parisc/hpux/sys_hpux.c 2006-02-07 01:18:50.000000000 +0300
@@ -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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.machine,HPUX_UTSLEN-1);
    error |= __put_user(0,name->machine+HPUX_UTSLEN-1);

    up_read(&uts_sem);
@@ -373,8 +373,8 @@ int hpux_utssys(char *ubuf, int n, int t
    /* TODO: print a warning about using this? */
    down_write(&uts_sem);
    error = -EFAULT;
- if (!copy_from_user(system_utsname.sysname, ubuf, len)) {
-     system_utsname.sysname[len] = 0;
+ if (!copy_from_user(uts_name.sysname, ubuf, len)) {
+     uts_name.sysname[len] = 0;
    error = 0;
}
    up_write(&uts_sem);
@@ -400,8 +400,8 @@ int hpux_utssys(char *ubuf, int n, int t
    /* TODO: print a warning about this? */
    down_write(&uts_sem);
    error = -EFAULT;
- if (!copy_from_user(system_utsname.release, ubuf, len)) {
-     system_utsname.release[len] = 0;
+ if (!copy_from_user(uts_name.release, ubuf, len)) {
+     uts_name.release[len] = 0;
    error = 0;
}
    up_write(&uts_sem);
@@ -422,13 +422,13 @@ int hpux_getdomainname(char *name, int l

    down_read(&uts_sem);

```

```

- nlen = strlen(system_utsname.domainname) + 1;
+ nlen = strlen(uts_name.domainname) + 1;

if (nlen < len)
    len = nlen;
if(len > __NEW_UTS_LEN)
    goto done;
- if(copy_to_user(name, system_utsname.domainname, len))
+ if(copy_to_user(name, uts_name.domainname, len))
    goto done;
    err = 0;
done:
--- ./arch/powerpc/kernel/syscalls.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/powerpc/kernel/syscalls.c 2006-02-07 01:18:50.000000000 +0300
@@ -259,7 +259,7 @@ long ppc_newuname(struct new_utsname __u
    int err = 0;

    down_read(&uts_sem);
- if (copy_to_user(name, &system_utsname, sizeof(*name)))
+ if (copy_to_user(name, &uts_name, sizeof(*name)))
    err = -EFAULT;
    up_read(&uts_sem);
    if (!err)
@@@ -272,7 +272,7 @@ int sys_uname(struct old_utsname __user
    int err = 0;

    down_read(&uts_sem);
- if (copy_to_user(name, &system_utsname, sizeof(*name)))
+ if (copy_to_user(name, &uts_name, sizeof(*name)))
    err = -EFAULT;
    up_read(&uts_sem);
    if (!err)
@@@ -288,19 +288,19 @@ int sys_olduname(struct oldold_utsname _
    return -EFAULT;

    down_read(&uts_sem);
- error = __copy_to_user(&name->sysname, &system_utsname.sysname,
+ error = __copy_to_user(&name->sysname, &uts_name.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, &uts_name.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, &uts_name.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, &uts_name.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, &uts_name.machine,
    __OLD_UTS_LEN);
    error |= override_machine(name->machine);
    up_read(&uts_sem);
--- ./arch/sh/kernel/setup.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sh/kernel/setup.c 2006-02-07 01:18:50.000000000 +0300
@@ -485,7 +485,7 @@ static int show_cpuinfo(struct seq_file
    seq_printf(m, "machine\t\t: %s\n", get_system_type());

    seq_printf(m, "processor\t: %d\n", cpu);
- seq_printf(m, "cpu family\t: %s\n", system_utsname.machine);
+ seq_printf(m, "cpu family\t: %s\n", uts_name.machine);
    seq_printf(m, "cpu type\t: %s\n", get_cpu_subtype());

    show_cpuflags(m);
--- ./arch/sh/kernel/sys_sh.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sh/kernel/sys_sh.c 2006-02-07 01:18:50.000000000 +0300
@@ -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, &uts_name, sizeof (*name));
    up_read(&uts_sem);
    return err?-EFAULT:0;
}
--- ./arch/sh64/kernel/sys_sh64.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sh64/kernel/sys_sh64.c 2006-02-07 01:18:50.000000000 +0300
@@ -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, &uts_name, sizeof (*name));
    up_read(&uts_sem);
    return err?-EFAULT:0;
}
--- ./arch/sparc/kernel/sys_sparc.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sparc/kernel/sys_sparc.c 2006-02-07 01:18:50.000000000 +0300
@@ -470,13 +470,13 @@ asmlinkage int sys_getdomainname(char __

    down_read(&uts_sem);

```



```

- nlen = strlen(system_utsname.domainname) + 1;
+ nlen = strlen(uts_name.domainname) + 1;

    if (nlen < len)
        len = nlen;
    if (len > __NEW_UTS_LEN)
        goto done;
- if (copy_to_user(name, system_utsname.domainname, len))
+ if (copy_to_user(name, uts_name.domainname, len))
    goto done;
    err = 0;
done:
--- ./arch/sparc/kernel/sys_sunos.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sparc/kernel/sys_sunos.c 2006-02-07 01:18:50.000000000 +0300
@@ -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], &uts_name.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], &uts_name.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], &uts_name.release[0], sizeof(name->rel) - 1);
+ ret |= __copy_to_user(&name->ver[0], &uts_name.version[0], sizeof(name->ver) - 1);
+ ret |= __copy_to_user(&name->mach[0], &uts_name.machine[0], sizeof(name->mach) - 1);
    }
    up_read(&uts_sem);
    return ret ? -EFAULT : 0;
--- ./arch/sparc64/kernel/sys_sparc.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sparc64/kernel/sys_sparc.c 2006-02-07 01:18:50.000000000 +0300
@@ -476,13 +476,13 @@ asmlinkage long sys_getdomainname(char _

    down_read(&uts_sem);

- nlen = strlen(system_utsname.domainname) + 1;
+ nlen = strlen(uts_name.domainname) + 1;

    if (nlen < len)
        len = nlen;

```



```

if (len > __NEW_UTS_LEN)
    goto done;
- if (copy_to_user(name, system_utsname.domainname, len))
+ if (copy_to_user(name, uts_name.domainname, len))
    goto done;
err = 0;
done:
--- ./arch/sparc64/kernel/sys_sunos32.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sparc64/kernel/sys_sunos32.c 2006-02-07 01:18:50.000000000 +0300
@@ -439,16 +439,16 @@ asmlinkage int sunos_uname(struct sunos_
    int ret;

    down_read(&uts_sem);
- ret = copy_to_user(&name->sname[0], &system_utsname.sysname[0],
+ ret = copy_to_user(&name->sname[0], &uts_name.sysname[0],
        sizeof(name->sname) - 1);
- ret |= copy_to_user(&name->nname[0], &system_utsname.nodename[0],
+ ret |= copy_to_user(&name->nname[0], &uts_name.nodename[0],
        sizeof(name->nname) - 1);
    ret |= put_user('\0', &name->nname[8]);
- ret |= copy_to_user(&name->rel[0], &system_utsname.release[0],
+ ret |= copy_to_user(&name->rel[0], &uts_name.release[0],
        sizeof(name->rel) - 1);
- ret |= copy_to_user(&name->ver[0], &system_utsname.version[0],
+ ret |= copy_to_user(&name->ver[0], &uts_name.version[0],
        sizeof(name->ver) - 1);
- ret |= copy_to_user(&name->mach[0], &system_utsname.machine[0],
+ ret |= copy_to_user(&name->mach[0], &uts_name.machine[0],
        sizeof(name->mach) - 1);
    up_read(&uts_sem);
    return (ret ? -EFAULT : 0);
--- ./arch/sparc64/solaris/misc.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/sparc64/solaris/misc.c 2006-02-07 01:18:50.000000000 +0300
@@ -239,7 +239,7 @@ asmlinkage int solaris_utssys(u32 buf, u
    /* Let's cheat */
    err = set_utsfield(v->sysname, "SunOS", 1, 0);
    down_read(&uts_sem);
- err |= set_utsfield(v->nodename, system_utsname.nodename,
+ err |= set_utsfield(v->nodename, uts_name.nodename,
        1, 1);
    up_read(&uts_sem);
    err |= set_utsfield(v->release, "2.6", 0, 0);
@@ -263,7 +263,7 @@ asmlinkage int solaris_utsname(u32 buf)
    /* Why should we not lie a bit? */
    down_read(&uts_sem);
    err = set_utsfield(v->sysname, "SunOS", 0, 0);
- err |= set_utsfield(v->nodename, system_utsname.nodename, 1, 1);
+ err |= set_utsfield(v->nodename, uts_name.nodename, 1, 1);

```

```

err |= set_utsfield(v->release, "5.6", 0, 0);
err |= set_utsfield(v->version, "Generic", 0, 0);
err |= set_utsfield(v->machine, machine(), 0, 0);
@@ -295,7 +295,7 @@ asmlinkage int solaris_sysinfo(int cmd,
case SI_HOSTNAME:
    r = buffer + 256;
    down_read(&uts_sem);
-   for (p = system_utsname.nodename, q = buffer;
+   for (p = uts_name.nodename, q = buffer;
        q < r && *p && *p != '.'; *q++ = *p++);
    up_read(&uts_sem);
    *q = 0;
--- ./arch/um/kernel/syscall_kern.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/um/kernel/syscall_kern.c 2006-02-07 01:18:50.000000000 +0300
@@ -110,7 +110,7 @@ long sys_uname(struct old_utsname * name
    if (!name)
        return -EFAULT;
    down_read(&uts_sem);
-   err=copy_to_user(name, &system_utsname, sizeof (*name));
+   err=copy_to_user(name, &uts_name, sizeof (*name));
    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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.machine,
        __OLD_UTS_LEN);
    error |= __put_user(0,name->machine+__OLD_UTS_LEN);

```

```

--- ./arch/um/sys-x86_64/syscalls.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/um/sys-x86_64/syscalls.c 2006-02-07 01:18:50.000000000 +0300
@@ -21,7 +21,7 @@ asmlinkage long sys_uname64(struct new_u
{
    int err;
    down_read(&uts_sem);
- err = copy_to_user(name, &system_utsname, sizeof (*name));
+ err = copy_to_user(name, &uts_name, sizeof (*name));
    up_read(&uts_sem);
    if (personality(current->personality) == PER_LINUX32)
        err |= copy_to_user(&name->machine, "i686", 5);
--- ./arch/x86_64/ia32/sys_ia32.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/x86_64/ia32/sys_ia32.c 2006-02-07 01:18:50.000000000 +0300
@@ -868,13 +868,13 @@ 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,&uts_name.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,&uts_name.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,&uts_name.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,&uts_name.version,__OLD_UTS_LEN);
    __put_user(0,name->version+__OLD_UTS_LEN);
    {
        char *arch = "x86_64";
@@ -897,7 +897,7 @@ 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, &uts_name, sizeof (*name));
    up_read(&uts_sem);
    if (personality(current->personality) == PER_LINUX32)
        err |= copy_to_user(&name->machine, "i686", 5);
--- ./arch/x86_64/kernel/sys_x86_64.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/x86_64/kernel/sys_x86_64.c 2006-02-07 01:18:50.000000000 +0300
@@ -148,7 +148,7 @@ asmlinkage long sys_uname(struct new_uts
{
    int err;
    down_read(&uts_sem);
- err = copy_to_user(name, &system_utsname, sizeof (*name));
+ err = copy_to_user(name, &uts_name, sizeof (*name));

```

```

up_read(&uts_sem);
if (personality(current->personality) == PER_LINUX32)
    err |= copy_to_user(&name->machine, "i686", 5);
--- ./arch/xtensa/kernel/syscalls.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./arch/xtensa/kernel/syscalls.c 2006-02-07 01:18:50.000000000 +0300
@@ -129,7 +129,7 @@ out:

int sys_uname(struct old_utsname * name)
{
- if (name && !copy_to_user(name, &system_utsname, sizeof (*name)))
+ if (name && !copy_to_user(name, &uts_name, sizeof (*name)))
    return 0;
    return -EFAULT;
}
--- ./drivers/usb/core/hcd.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./drivers/usb/core/hcd.c 2006-02-07 01:18:50.000000000 +0300
@@ -317,8 +317,8 @@ static int rh_string (

    // id 3 == vendor description
    } else if (id == 3) {
- snprintf (buf, sizeof buf, "%s %s %s", system_utsname.sysname,
- system_utsname.release, hcd->driver->description);
+ snprintf (buf, sizeof buf, "%s %s %s", uts_name.sysname,
+ uts_name.release, hcd->driver->description);

    // unsupported IDs --> "protocol stall"
    } else
--- ./fs/cifs/connect.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./fs/cifs/connect.c 2006-02-07 01:18:50.000000000 +0300
@@ -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(uts_name.nodename, 15); i++) {
    /* 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(uts_name.nodename[i]);
    }
    vol->source_rfc1001_name[15] = 0;
    /* null target name indicates to use *SMBSEVR default called name
@@ -2062,7 +2062,7 @@ CIFSSessSetup(unsigned int xid, struct c
    32, nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bytes_returned =

```

```

-   cifs_strtoUCS((__le16 *) bcc_ptr, system_utsname.release,
+   cifs_strtoUCS((__le16 *) bcc_ptr, uts_name.release,
    32, nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bcc_ptr += 2;
@@ -2089,8 +2089,8 @@ CIFSSessSetup(unsigned int xid, struct c
    }
    strcpy(bcc_ptr, "Linux version ");
    bcc_ptr += strlen("Linux version ");
-   strcpy(bcc_ptr, system_utsname.release);
-   bcc_ptr += strlen(system_utsname.release) + 1;
+   strcpy(bcc_ptr, uts_name.release);
+   bcc_ptr += strlen(uts_name.release) + 1;
    strcpy(bcc_ptr, CIFS_NETWORK_OPSYS);
    bcc_ptr += strlen(CIFS_NETWORK_OPSYS) + 1;
    }
@@ -2329,7 +2329,7 @@ CIFSSpnegoSessSetup(unsigned int xid, st
    32, nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bytes_returned =
-   cifs_strtoUCS((__le16 *) bcc_ptr, system_utsname.release, 32,
+   cifs_strtoUCS((__le16 *) bcc_ptr, uts_name.release, 32,
    nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bcc_ptr += 2;
@@ -2354,8 +2354,8 @@ CIFSSpnegoSessSetup(unsigned int xid, st
    }
    strcpy(bcc_ptr, "Linux version ");
    bcc_ptr += strlen("Linux version ");
-   strcpy(bcc_ptr, system_utsname.release);
-   bcc_ptr += strlen(system_utsname.release) + 1;
+   strcpy(bcc_ptr, uts_name.release);
+   bcc_ptr += strlen(uts_name.release) + 1;
    strcpy(bcc_ptr, CIFS_NETWORK_OPSYS);
    bcc_ptr += strlen(CIFS_NETWORK_OPSYS) + 1;
    }
@@ -2619,7 +2619,7 @@ CIFSNTLMSSPNegotiateSessSetup(unsigned i
    32, nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bytes_returned =
-   cifs_strtoUCS((__le16 *) bcc_ptr, system_utsname.release, 32,
+   cifs_strtoUCS((__le16 *) bcc_ptr, uts_name.release, 32,
    nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bcc_ptr += 2; /* null terminate Linux version */
@@ -2636,8 +2636,8 @@ CIFSNTLMSSPNegotiateSessSetup(unsigned i
    } else { /* ASCII */
        strcpy(bcc_ptr, "Linux version ");

```

```

    bcc_ptr += strlen("Linux version ");
-   strcpy(bcc_ptr, system_utsname.release);
-   bcc_ptr += strlen(system_utsname.release) + 1;
+   strcpy(bcc_ptr, uts_name.release);
+   bcc_ptr += strlen(uts_name.release) + 1;
    strcpy(bcc_ptr, CIFS_NETWORK_OPSYS);
    bcc_ptr += strlen(CIFS_NETWORK_OPSYS) + 1;
    bcc_ptr++; /* empty domain field */
@@ -2998,7 +2998,7 @@ CIFSNTLMSSPAuthSessSetup(unsigned int xi
    32, nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bytes_returned =
-   cifs_strtoUCS((__le16 *) bcc_ptr, system_utsname.release, 32,
+   cifs_strtoUCS((__le16 *) bcc_ptr, uts_name.release, 32,
    nls_codepage);
    bcc_ptr += 2 * bytes_returned;
    bcc_ptr += 2; /* null term version string */
@@ -3050,8 +3050,8 @@ CIFSNTLMSSPAuthSessSetup(unsigned int xi

    strcpy(bcc_ptr, "Linux version ");
    bcc_ptr += strlen("Linux version ");
-   strcpy(bcc_ptr, system_utsname.release);
-   bcc_ptr += strlen(system_utsname.release) + 1;
+   strcpy(bcc_ptr, uts_name.release);
+   bcc_ptr += strlen(uts_name.release) + 1;
    strcpy(bcc_ptr, CIFS_NETWORK_OPSYS);
    bcc_ptr += strlen(CIFS_NETWORK_OPSYS) + 1;
    bcc_ptr++; /* null domain */
--- ./fs/exec.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./fs/exec.c 2006-02-07 01:18:50.000000000 +0300
@@ -1326,7 +1326,7 @@ static void format_corename(char *corena
    case 'h':
        down_read(&uts_sem);
        rc = snprintf(out_ptr, out_end - out_ptr,
-           "%s", system_utsname.nodename);
+           "%s", uts_name.nodename);
        up_read(&uts_sem);
        if (rc > out_end - out_ptr)
            goto out;
--- ./fs/lockd/clntproc.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./fs/lockd/clntproc.c 2006-02-07 01:18:50.000000000 +0300
@@ -130,10 +130,10 @@ static void nlmclnt_setlockargs(struct n
    nlmclnt_next_cookie(&argp->cookie);
    argp->state = nsm_local_state;
    memcpy(&lock->fh, NFS_FH(fl->fl_file->f_dentry->d_inode), sizeof(struct nfs_fh));
-   lock->caller = system_utsname.nodename;
+   lock->caller = uts_name.nodename;
    lock->oh.data = req->a_owner;

```

```

    lock->oh.len = sprintf(req->a_owner, "%d@%s",
-   current->pid, system_utsname.nodename);
+   current->pid, uts_name.nodename);
    locks_copy_lock(&lock->fl, fl);
}

@@ -154,7 +154,7 @@ nlmclnt_setgrantargs(struct nlm_rqst *ca
{
    locks_copy_lock(&call->a_args.lock.fl, &lock->fl);
    memcpy(&call->a_args.lock.fh, &lock->fh, sizeof(call->a_args.lock.fh));
-   call->a_args.lock.caller = system_utsname.nodename;
+   call->a_args.lock.caller = uts_name.nodename;
    call->a_args.lock.oh.len = lock->oh.len;

    /* set default data area */
--- ./include/asm-i386/elf.h.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./include/asm-i386/elf.h 2006-02-07 01:18:50.000000000 +0300
@@ -108,7 +108,7 @@ typedef struct user_fxsr_struct elf_fpxr
    For the moment, we have only optimizations for the Intel generations,
    but that could change... */

-#define ELF_PLATFORM (system_utsname.machine)
+#define ELF_PLATFORM (uts_name.machine)

#ifdef __KERNEL__
#define SET_PERSONALITY(ex, ibcs2) do { } while (0)
--- ./include/linux/container.h.utsnamex 2006-02-07 01:18:40.000000000 +0300
+++ ./include/linux/container.h 2006-02-07 01:18:53.000000000 +0300
@@ -7,6 +7,7 @@

struct task_struct;
struct list_head;
+struct new_utsname;

struct container {
    u32 id;
@@ -14,6 +15,7 @@ struct container {
    atomic_t refcnt;

    struct list_head *c_uid_hash;
+   struct new_utsname *c_uts_name;
};

extern struct container init_container;
--- ./include/linux/init_task.h.utsnamex 2006-02-07 01:18:40.000000000 +0300
+++ ./include/linux/init_task.h 2006-02-07 01:18:53.000000000 +0300
@@ -133,6 +133,7 @@ extern struct group_info init_groups;
}

```



```

#define INIT_CONTAINER(cont) \
- .refcnt = ATOMIC_INIT(1)
+ .refcnt = ATOMIC_INIT(1) \
+ .c_uts_name = &system_utsname,

#endif
--- ./include/linux/utsname.h.utsnamex 2006-02-07 01:18:40.000000000 +0300
+++ ./include/linux/utsname.h 2006-02-07 01:18:53.000000000 +0300
@@ -1,6 +1,8 @@
#ifndef _LINUX_UTSNAME_H
#define _LINUX_UTSNAME_H

+#include <linux/container.h>
+
#define __OLD_UTS_LEN 8

struct oldold_utsname {
@@ -31,6 +33,11 @@ struct new_utsname {
};

extern struct new_utsname system_utsname;
+#ifdef CONFIG_CONTAINER
+#define uts_name (*(econtainer()->c_uts_name))
+#else
+#define uts_name system_utsname
+#endif

extern struct rw_semaphore uts_sem;
#endif
--- ./kernel/sys.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./kernel/sys.c 2006-02-07 01:18:50.000000000 +0300
@@ -1518,7 +1518,7 @@ 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,&uts_name,sizeof *name))
    errno = -EFAULT;
up_read(&uts_sem);
return errno;
@@ -1536,8 +1536,8 @@ asmlinkage long sys_sethostname(char __u
down_write(&uts_sem);
errno = -EFAULT;
if (!copy_from_user(tmp, name, len)) {
- memcpy(system_utsname.nodename, tmp, len);
- system_utsname.nodename[len] = 0;
+ memcpy(uts_name.nodename, tmp, len);

```

```

+ uts_name.nodename[len] = 0;
  errno = 0;
}
  up_write(&uts_sem);
@@ -1553,11 +1553,11 @@ asmlinkage long sys_gethostname(char __u
  if (len < 0)
    return -EINVAL;
  down_read(&uts_sem);
- i = 1 + strlen(system_utsname.nodename);
+ i = 1 + strlen(uts_name.nodename);
  if (i > len)
    i = len;
  errno = 0;
- if (copy_to_user(name, system_utsname.nodename, i))
+ if (copy_to_user(name, uts_name.nodename, i))
    errno = -EFAULT;
  up_read(&uts_sem);
  return errno;
@@ -1582,8 +1582,8 @@ asmlinkage long sys_setdomainname(char _
  down_write(&uts_sem);
  errno = -EFAULT;
  if (!copy_from_user(tmp, name, len)) {
- memcpy(system_utsname.domainname, tmp, len);
- system_utsname.domainname[len] = 0;
+ memcpy(uts_name.domainname, tmp, len);
+ uts_name.domainname[len] = 0;
  }
  errno = 0;
}
  up_write(&uts_sem);
--- ./net/sunrpc/clnt.c.utsnamex 2006-02-07 01:18:42.000000000 +0300
+++ ./net/sunrpc/clnt.c 2006-02-07 01:18:50.000000000 +0300
@@ -168,10 +168,10 @@ rpc_new_client(struct rpc_xprt *xprt, ch
}

/* save the nodename */
- clnt->cl_nodename = strlen(system_utsname.nodename);
+ clnt->cl_nodename = strlen(uts_name.nodename);
  if (clnt->cl_nodename > UNIX_MAXNODENAME)
    clnt->cl_nodename = UNIX_MAXNODENAME;
- memcpy(clnt->cl_nodename, system_utsname.nodename, clnt->cl_nodename);
+ memcpy(clnt->cl_nodename, uts_name.nodename, clnt->cl_nodename);
  return clnt;

```

out_no_auth:
