Subject: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Anonymous Coward on Thu, 19 Oct 2006 12:30:37 GMT
View Forum Message <> Reply to Message

OpenVZ Linux kernel team has discovered the problem
with 32bit quota tools working on 64bit architectures.
In 2.6.10 kernel sys32_quotactl() function was replaced by sys_quotactl() with
the comment "sys_quotactl seems to be 32/64bit clean, enable it for 32bit"
However this isn't right. Look at if_dqblk structure:

```
struct if_dqblk {
        __u64 dqb_bhardlimit;
        __u64 dqb_bsoftlimit;
        __u64 dqb_curspace;
        __u64 dqb_ihardlimit;
        __u64 dqb_isoftlimit;
        __u64 dqb_curinodes;
        __u64 dqb_btime;
        __u64 dqb_itime;
        __u32 dqb_valid;
};
```

For 32 bit quota tools sizeof(if_dqblk) == 0x44.
But for 64 bit kernel its size is 0x48, 'cause of alignment!
Thus we got a problem.
Attached patch reintroduce sys32_quotactl() function,
that handles the situation.

Signed-off-by: Vasily Tarasov <vtaras@openvz.org>
Acked-by: Dmitry Mishin <dim@openvz.org>

In OpenVZ technology 32 bit Virtual Environments over
64 bit OS are common, hence we have customers, that complains on this bad quota
behaviour:

```
# /usr/bin/quota
quota: error while getting quota from /dev/sda1 for 0: Success
```

The reason is caused above.

```
--- linux-2.6.18/arch/ia64/ia32/sys_ia32.c.quot32 2006-09-20 07:42:06.000000000 +0400
+++ linux-2.6.18/arch/ia64/ia32/sys_ia32.c 2006-10-19 11:17:50.000000000 +0400
@@ -2545,6 +2545,54 @@ long sys32_fadvise64_64(int fd, __u32 of
         advice);
 }
```

```
+asmlinkage long sys32_quotactl(unsigned int cmd, const char __user *special,
+      qid_t id, void __user *addr)
+{
+ long ret;
+ unsigned int cmds;
+ mm_segment_t old_fs;
+ struct if_dqblk dqblk;
+ struct if32_dqblk {
+  __u32 dqb_bhardlimit[2];
+  __u32 dqb_bsoftlimit[2];
+  __u32 dqb_curspace[2];
+  __u32 dqb_ihardlimit[2];
+  __u32 dqb_isoftlimit[2];
+  __u32 dqb_curinodes[2];
+  __u32 dqb_btime[2];
+  __u32 dqb_itime[2];
+  __u32 dqb_valid;
+ } dqblk32;
+
+ cmds = cmd >> SUBCMDSHIFT;
+
+ switch (cmds) {
+  case Q_GETQUOTA:
+   old_fs = get_fs();
+   set_fs(KERNEL_DS);
+   ret = sys_quotactl(cmd, special, id, &dqblk);
+   set_fs(old_fs);
+   memcpy(&dqblk32, &dqblk, sizeof(dqblk32));
+   dqblk32.dqb_valid = dqblk.dqb_valid;
+   if (copy_to_user(addr, &dqblk32, sizeof(dqblk32)))
+    return -EFAULT;
+   break;
+  case Q_SETQUOTA:
+   if (copy_from_user(&dqblk32, addr, sizeof(dqblk32)))
+    return -EFAULT;
+   memcpy(&dqblk, &dqblk32, sizeof(dqblk32));
+   dqblk.dqb_valid = dqblk32.dqb_valid;
+   old_fs = get_fs();
+   set_fs(KERNEL_DS);
+   ret = sys_quotactl(cmd, special, id, &dqblk);
+   set_fs(old_fs);
+   break;
+  default:
+   return sys_quotactl(cmd, special, id, addr);
+ }
+ return ret;
+}
+
```

```
 #ifdef NOTYET  /* UNTESTED FOR IA64 FROM HERE DOWN */

 asmlinkage long sys32_setreuid(compat_uid_t ruid, compat_uid_t euid)
--- linux-2.6.18/arch/ia64/ia32/ia32_entry.S.quot32 2006-09-20 07:42:06.000000000 +0400
+++ linux-2.6.18/arch/ia64/ia32/ia32_entry.S 2006-10-19 11:15:52.000000000 +0400
@@ -341,7 +341,7 @@ ia32_syscall_table:
  data8 sys_ni_syscall /* init_module */
  data8 sys_ni_syscall /* delete_module */
  data8 sys_ni_syscall /* get_kernel_syms */  /* 130 */
- data8 sys_quotactl
+ data8 sys32_quotactl
  data8 sys_getpgid
  data8 sys_fchdir
  data8 sys_ni_syscall /* sys_bdflush */
--- linux-2.6.18/arch/x86_64/ia32/ia32entry.S.quot32 2006-09-20 07:42:06.000000000 +0400
+++ linux-2.6.18/arch/x86_64/ia32/ia32entry.S 2006-10-18 10:05:53.000000000 +0400
@@ -526,7 +526,7 @@ ia32_sys_call_table:
  .quad sys_init_module
  .quad sys_delete_module
  .quad quiet_ni_syscall  /* 130  get_kernel_syms */
- .quad sys_quotactl
+ .quad sys32_quotactl
  .quad sys_getpgid
  .quad sys_fchdir
  .quad quiet_ni_syscall /* bdflush */
--- linux-2.6.18/arch/x86_64/ia32/sys_ia32.c.quot32 2006-09-20 07:42:06.000000000 +0400
+++ linux-2.6.18/arch/x86_64/ia32/sys_ia32.c 2006-10-19 11:00:18.000000000 +0400
@@ -915,3 +915,50 @@ long sys32_lookup_dcookie(u32 addr_low,
  return sys_lookup_dcookie(((u64)addr_high << 32) | addr_low, buf, len);
 }

+asmlinkage long sys32_quotactl(unsigned int cmd, const char __user *special,
+       qid_t id, void __user *addr)
+{
+ long ret;
+ unsigned int cmds;
+ mm_segment_t old_fs;
+ struct if_dqblk dqblk;
+ struct if32_dqblk {
+  __u32 dqb_bhardlimit[2];
+  __u32 dqb_bsoftlimit[2];
+  __u32 dqb_curspace[2];
+  __u32 dqb_ihardlimit[2];
+  __u32 dqb_isoftlimit[2];
+  __u32 dqb_curinodes[2];
+  __u32 dqb_btime[2];
+  __u32 dqb_itime[2];
+  __u32 dqb_valid;
```

```
+ } dqblk32;
+
+ cmds = cmd >> SUBCMDSHIFT;
+
+ switch (cmds) {
+  case Q_GETQUOTA:
+   old_fs = get_fs();
+   set_fs(KERNEL_DS);
+   ret = sys_quotactl(cmd, special, id, &dqblk);
+   set_fs(old_fs);
+   memcpy(&dqblk32, &dqblk, sizeof(dqblk32));
+   dqblk32.dqb_valid = dqblk.dqb_valid;
+   if (copy_to_user(addr, &dqblk32, sizeof(dqblk32)))
+    return -EFAULT;
+   break;
+  case Q_SETQUOTA:
+   if (copy_from_user(&dqblk32, addr, sizeof(dqblk32)))
+    return -EFAULT;
+   memcpy(&dqblk, &dqblk32, sizeof(dqblk32));
+   dqblk.dqb_valid = dqblk32.dqb_valid;
+   old_fs = get_fs();
+   set_fs(KERNEL_DS);
+   ret = sys_quotactl(cmd, special, id, &dqblk);
+   set_fs(old_fs);
+   break;
+  default:
+   return sys_quotactl(cmd, special, id, addr);
+ }
+ return ret;
+}
```

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Alan Cox on Thu, 19 Oct 2006 13:03:58 GMT
View Forum Message <> Reply to Message

Ar Iau, 2006-10-19 am 16:32 +0400, ysgrifennodd Vasily Tarasov:
> OpenVZ Linux kernel team has discovered the problem
> Signed-off-by: Vasily Tarasov <vtaras@openvz.org>
> Acked-by: Dmitry Mishin <dim@openvz.org>

Acked-by: Alan Cox <alan@redhat.com>

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by rdunlap on Thu, 19 Oct 2006 15:20:56 GMT
View Forum Message <> Reply to Message

On Thu, 19 Oct 2006 16:32:07 +0400 Vasily Tarasov wrote:

```
> --- linux-2.6.18/arch/ia64/ia32/sys_ia32.c.quot32 2006-09-20 07:42:06.000000000 +0400
> +++ linux-2.6.18/arch/ia64/ia32/sys_ia32.c 2006-10-19 11:17:50.000000000 +0400
> @@ -2545,6 +2545,54 @@ long sys32_fadvise64_64(int fd, __u32 of
>           advice);
>  }
>
> +asmlinkage long sys32_quotactl(unsigned int cmd, const char __user *special,
> +      qid_t id, void __user *addr)
> +{
> +
> + switch (cmds) {
> +  case Q_GETQUOTA:
> +   old_fs = get_fs();
> +   set_fs(KERNEL_DS);
> +   ret = sys_quotactl(cmd, special, id, &dqblk);
> +   set_fs(old_fs);
> +   memcpy(&dqblk32, &dqblk, sizeof(dqblk32));
> +   dqblk32.dqb_valid = dqblk.dqb_valid;
> +   if (copy_to_user(addr, &dqblk32, sizeof(dqblk32)))
> +    return -EFAULT;
> +   break;
> +  case Q_SETQUOTA:
> +   if (copy_from_user(&dqblk32, addr, sizeof(dqblk32)))
> +    return -EFAULT;
> +   memcpy(&dqblk, &dqblk32, sizeof(dqblk32));
> +   dqblk.dqb_valid = dqblk32.dqb_valid;
> +   old_fs = get_fs();
> +   set_fs(KERNEL_DS);
> +   ret = sys_quotactl(cmd, special, id, &dqblk);
> +   set_fs(old_fs);
> +   break;
> +  default:
> +   return sys_quotactl(cmd, special, id, addr);
> + }
> + return ret;
> +}
```

Please align the switch and case/default source lines.
We prefer not to "double-indent" each case block inside a switch.

I suppose I should try to add this to CodingStyle since it's
not there.

```
> --- linux-2.6.18/arch/x86_64/ia32/sys_ia32.c.quot32 2006-09-20 07:42:06.000000000 +0400
> +++ linux-2.6.18/arch/x86_64/ia32/sys_ia32.c 2006-10-19 11:00:18.000000000 +0400
> @@ -915,3 +915,50 @@ long sys32_lookup_dcookie(u32 addr_low,
```

```
>   return sys_lookup_dcookie(((u64)addr_high << 32) | addr_low, buf, len);
>  }
>
> +asmlinkage long sys32_quotactl(unsigned int cmd, const char __user *special,
> +       qid_t id, void __user *addr)
> +{
> +
> + switch (cmds) {
> +  case Q_GETQUOTA:
> +   old_fs = get_fs();
> +   set_fs(KERNEL_DS);
> +   ret = sys_quotactl(cmd, special, id, &dqblk);
> +   set_fs(old_fs);
> +   memcpy(&dqblk32, &dqblk, sizeof(dqblk32));
> +   dqblk32.dqb_valid = dqblk.dqb_valid;
> +   if (copy_to_user(addr, &dqblk32, sizeof(dqblk32)))
> +    return -EFAULT;
> +   break;
> +  case Q_SETQUOTA:
> +   if (copy_from_user(&dqblk32, addr, sizeof(dqblk32)))
> +    return -EFAULT;
> +   memcpy(&dqblk, &dqblk32, sizeof(dqblk32));
> +   dqblk.dqb_valid = dqblk32.dqb_valid;
> +   old_fs = get_fs();
> +   set_fs(KERNEL_DS);
> +   ret = sys_quotactl(cmd, special, id, &dqblk);
> +   set_fs(old_fs);
> +   break;
> +  default:
> +   return sys_quotactl(cmd, special, id, addr);
> + }
```

~Randy

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Christoph Hellwig on Thu, 19 Oct 2006 17:29:49 GMT
View Forum Message <> Reply to Message

On Thu, Oct 19, 2006 at 04:32:07PM +0400, Vasily Tarasov wrote:

```
> +asmlinkage long sys32_quotactl(unsigned int cmd, const char __user *special,
> +       qid_t id, void __user *addr)
> +{
> + long ret;
> + unsigned int cmds;
> + mm_segment_t old_fs;
```

```
> + struct if_dqblk dqblk;
> + struct if32_dqblk {
> +  __u32 dqb_bhardlimit[2];
> +  __u32 dqb_bsoftlimit[2];
> +  __u32 dqb_curspace[2];
> +  __u32 dqb_ihardlimit[2];
> +  __u32 dqb_isoftlimit[2];
> +  __u32 dqb_curinodes[2];
> +  __u32 dqb_btime[2];
> +  __u32 dqb_itime[2];
> +  __u32 dqb_valid;
> + } dqblk32;
> +
> + cmds = cmd >> SUBCMDSHIFT;
> +
> + switch (cmds) {
> +  case Q_GETQUOTA:
> +   old_fs = get_fs();
> +   set_fs(KERNEL_DS);
> +   ret = sys_quotactl(cmd, special, id, &dqblk);
> +   set_fs(old_fs);
```

Please allocate the structure using compat_alloc_userspace and copy
with copy_in_user instead of the set_fs trick.

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Anonymous Coward on Fri, 20 Oct 2006 05:58:04 GMT
View Forum Message <> Reply to Message

Randy Dunlap wrote:

<snip>
> Please align the switch and case/default source lines.
> We prefer not to "double-indent" each case block inside a switch.
>
> I suppose I should try to add this to CodingStyle since it's
> not there.
>
> ---
> ~Randy
<snip>

Thank you, I'll do it!

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures

Posted by Anonymous Coward on Fri, 20 Oct 2006 06:08:57 GMT
View Forum Message <> Reply to Message

Christoph Hellwig wrote:

<snip>
> Please allocate the structure using compat_alloc_userspace and copy
> with copy_in_user instead of the set_fs trick.
<snip>

Good idea, thank you for your tip, I'll do it.

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Anonymous Coward on Fri, 20 Oct 2006 06:28:23 GMT
View Forum Message <> Reply to Message

Andi Kleen wrote:

<snip>
> Thanks. But the code should be probably common somewhere in fs/*, not
> duplicated.
<snip>

Thank you for the comment!
I'm not sure we should do it. If we move the code in fs/quota.c for example,
than this code will be compiled for _all_ arhitectures, not only for x86_64 and ia64.
Of course, we can surround this code by #ifdefs <ARCH>, but I thought this is
a bad style... Moreover looking through current kernel code, I found out that
usually code is duplicated in such cases.

However, if you insist I'll modify the code! :)

Thank you.

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Christoph Hellwig on Fri, 20 Oct 2006 07:12:04 GMT
View Forum Message <> Reply to Message

On Fri, Oct 20, 2006 at 10:30:04AM +0400, Vasily Tarasov wrote:
> Andi Kleen wrote:
>
> <snip>
> > Thanks. But the code should be probably common somewhere in fs/*, not
> > duplicated.
> <snip>
>

> Thank you for the comment!
> I'm not sure we should do it. If we move the code in fs/quota.c for example,
> than this code will be compiled for _all_ arhitectures, not only for x86_64 and ia64.
> Of course, we can surround this code by #ifdefs <ARCH>, but I thought this is
> a bad style... Moreover looking through current kernel code, I found out that
> usually code is duplicated in such cases.
>
> However, if you insist I'll modify the code! :)

I suspect a compat_x86.c file somehwere might make sense, as only x86 has
the wierd alignment rules, but we have two architectures that allow to run
x86 binaries with the compat subszstem. Now the big question: where should
we put this file?

>
> Thank you.
>
> -
> To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at http://vger.kernel.org/majordomo-info.html
> Please read the FAQ at http://www.tux.org/lkml/
---end quoted text---

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Andi Kleen on Fri, 20 Oct 2006 12:21:43 GMT
View Forum Message <> Reply to Message

On Friday 20 October 2006 08:30, Vasily Tarasov wrote:
> Andi Kleen wrote:
>
> <snip>
> > Thanks. But the code should be probably common somewhere in fs/*, not
> > duplicated.
> <snip>
>
> Thank you for the comment!
> I'm not sure we should do it. If we move the code in fs/quota.c for example,
> than this code will be compiled for _all_ arhitectures, not only for x86_64 and ia64.
> Of course, we can surround this code by #ifdefs <ARCH>, but I thought this is
> a bad style... Moreover looking through current kernel code, I found out that
> usually code is duplicated in such cases.

Well it doesn't hurt them even if not strictly needed and it's better to have common code for
this. BTW you have to convert over to compat_alloc_* for this as Christoph stated
because set_fs doesn't work on all architectures. Best you use the compat_* types too.

-Andi

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Arnd Bergmann on Sat, 21 Oct 2006 16:28:32 GMT
View Forum Message <> Reply to Message

On Friday 20 October 2006 08:10, Vasily Tarasov wrote:
> Christoph Hellwig wrote:
>
> <snip>
>
> > Please allocate the structure using compat_alloc_userspace and copy
> > with copy_in_user instead of the set_fs trick.
>
> <snip>
>
> Good idea, thank you for your tip, I'll do it.

I think it would be even better to integrate this into fs/quota.c
and get rid of the extra copy entirely. The only thing you need
to do differently in case of 32 bit Q_GETQUOTA is the size
of the copy_{from,to}_user.

On a related topic, I just noticed

```
typedef struct fs_qfilestat {
 __u64  qfs_ino; /* inode number */
 __u64  qfs_nblks; /* number of BBs 512-byte-blks */
 __u32  qfs_nextents; /* number of extents */
} fs_qfilestat_t;

typedef struct fs_quota_stat {
 __s8  qs_version; /* version number for future changes */
 __u16  qs_flags; /* XFS_QUOTA_{U,P,G}DQ_{ACCT,ENFD} */
 __s8  qs_pad;  /* unused */
 fs_qfilestat_t qs_uquota; /* user quota storage information */
 fs_qfilestat_t qs_gquota; /* group quota storage information */
 __u32  qs_incoredqs; /* number of dquots incore */
 __s32  qs_btimelimit;  /* limit for blks timer */
 __s32  qs_itimelimit;  /* limit for inodes timer */
 __s32  qs_rtbtimelimit;/* limit for rt blks timer */
 __u16  qs_bwarnlimit; /* limit for num warnings */
 __u16  qs_iwarnlimit; /* limit for num warnings */
} fs_quota_stat_t;
```

This one seems to have a more severe problem in x86_64 compat
mode. I haven't tried it, but isn't everything down from

gs_gquota aligned differently on i386?

Arnd <><

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by David Chinner on Mon, 23 Oct 2006 02:12:03 GMT
View Forum Message <> Reply to Message

On Sat, Oct 21, 2006 at 06:28:32PM +0200, Arnd Bergmann wrote:
> On a related topic, I just noticed
>
> typedef struct fs_qfilestat {
> __u64 qfs_ino; /* inode number */
> __u64 qfs_nblks; /* number of BBs 512-byte-blks */
> __u32 qfs_nextents; /* number of extents */
> } fs_qfilestat_t;
>
> typedef struct fs_quota_stat {
> __s8 qs_version; /* version number for future changes */
> __u16 qs_flags; /* XFS_QUOTA_{U,P,G}DQ_{ACCT,ENFD} */
> __s8 qs_pad; /* unused */
> fs_qfilestat_t qs_uquota; /* user quota storage information */
> fs_qfilestat_t qs_gquota; /* group quota storage information */
> __u32 qs_incoredqs; /* number of dquots incore */
> __s32 qs_btimelimit; /* limit for blks timer */
> __s32 qs_itimelimit; /* limit for inodes timer */
> __s32 qs_rtbtimelimit;/* limit for rt blks timer */
> __u16 qs_bwarnlimit; /* limit for num warnings */
> __u16 qs_iwarnlimit; /* limit for num warnings */
> } fs_quota_stat_t;

Ah, the XFS quota structures.....

> This one seems to have a more severe problem in x86_64 compat
> mode. I haven't tried it, but isn't everything down from
> gs_gquota aligned differently on i386?

Yes - this is just one of several interfaces into XFS that need compat
handling that don't have them right now.

Cheers,

Dave.
--
Dave Chinner
Principal Engineer
SGI Australian Software Group

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Anonymous Coward on Mon, 23 Oct 2006 10:50:24 GMT
View Forum Message <> Reply to Message

Hello,

Arnd Bergmann wrote:

<snip>
> On a related topic, I just noticed
>
> typedef struct fs_qfilestat {
> __u64 qfs_ino; /* inode number */
> __u64 qfs_nblks; /* number of BBs 512-byte-blks */
> __u32 qfs_nextents; /* number of extents */
> } fs_qfilestat_t;
>
> typedef struct fs_quota_stat {
> __s8 qs_version; /* version number for future changes */
> __u16 qs_flags; /* XFS_QUOTA_{U,P,G}DQ_{ACCT,ENFD} */
> __s8 qs_pad; /* unused */
> fs_qfilestat_t qs_uquota; /* user quota storage information */
> fs_qfilestat_t qs_gquota; /* group quota storage information */
> __u32 qs_incoredqs; /* number of dquots incore */
> __s32 qs_btimelimit; /* limit for blks timer */
> __s32 qs_itimelimit; /* limit for inodes timer */
> __s32 qs_rtbtimelimit;/* limit for rt blks timer */
> __u16 qs_bwarnlimit; /* limit for num warnings */
> __u16 qs_iwarnlimit; /* limit for num warnings */
> } fs_quota_stat_t;
>
> This one seems to have a more severe problem in x86_64 compat
> mode. I haven't tried it, but isn't everything down from
> gs_gquota aligned differently on i386?
<snip>

The problem indeed exists:

ia32:
sizeof(fs_qfilestat) = 0x14
sizeof(fs_quota_stat) = 0x44

x86_64:
sizeof(fs_qfilestat) = 0x18
sizeof(fs_quota_stat) = 0x50

Note, that the difference between sizes of fs_qfilestat on ia32 and
on x86_64 doesn't equal 8 bytes, as was expected (by me :)), but equals 12 bytes:
'cause of padding at the end of fs_quota_stat structure on x86_64.

I will add support of 32-bit XFS quotactl over 64bit OS in next patch.

Thank you!

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Mikael Pettersson on Fri, 15 Jun 2007 10:03:57 GMT
View Forum Message <> Reply to Message

On Fri, 15 Jun 2007 13:01:48 +0400, Vasily Tarasov wrote:
> OpenVZ Linux kernel team has discovered the problem
> with 32bit quota tools working on 64bit architectures.
> In 2.6.10 kernel sys32_quotactl() function was replaced by sys_quotactl() with
> the comment "sys_quotactl seems to be 32/64bit clean, enable it for 32bit"
> However this isn't right. Look at if_dqblk structure:

Your patch only converts ia32 on x86-64 or ia64.
What about ppc32-on-ppc64 and sparc32-on-sparc64?
And, I guess, mips32-on-mips64?

> --- linux-2.6.22-rc4-fixed/fs/quota.c.orig 2007-06-14 15:55:26.000000000 +0400
> +++ linux-2.6.22-rc4-fixed/fs/quota.c 2007-06-14 19:50:13.000000000 +0400
...
> +#if defined(CONFIG_X86_64) || defined(CONFIG_IA64)
> +/*
> + * This code works only for 32 bit quota tools over 64 bit OS (x86_64, ia64)
> + * and is necessary due to alignment problems.
> + */

The #ifdef looks way too arch-specific. And isn't there a shared
compat.c module somewhere that this should go into?

/Mikael

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Vasily Tarasov on Fri, 15 Jun 2007 10:41:49 GMT
View Forum Message <> Reply to Message

You can find a half-year back discussions of this patch:

First attempt: http://lkml.org/lkml/2006/10/19/123
Second attempt: http://lkml.org/lkml/2006/10/25/57

I think they will answer your questions.

Thank you,
Vasily

On Fri, 2007-06-15 at 12:03 +0200, Mikael Pettersson wrote:
> On Fri, 15 Jun 2007 13:01:48 +0400, Vasily Tarasov wrote:
> > OpenVZ Linux kernel team has discovered the problem
> > with 32bit quota tools working on 64bit architectures.
> > In 2.6.10 kernel sys32_quotactl() function was replaced by sys_quotactl() with
> > the comment "sys_quotactl seems to be 32/64bit clean, enable it for 32bit"
> > However this isn't right. Look at if_dqblk structure:
>
> Your patch only converts ia32 on x86-64 or ia64.
> What about ppc32-on-ppc64 and sparc32-on-sparc64?
> And, I guess, mips32-on-mips64?
>
> > --- linux-2.6.22-rc4-fixed/fs/quota.c.orig 2007-06-14 15:55:26.000000000 +0400
> > +++ linux-2.6.22-rc4-fixed/fs/quota.c 2007-06-14 19:50:13.000000000 +0400
> ...
> > +#if defined(CONFIG_X86_64) || defined(CONFIG_IA64)
> > +/*
> > + * This code works only for 32 bit quota tools over 64 bit OS (x86_64, ia64)
> > + * and is necessary due to alignment problems.
> > + */
>
> The #ifdef looks way too arch-specific. And isn't there a shared
> compat.c module somewhere that this should go into?
>
> /Mikael

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Arnd Bergmann on Fri, 15 Jun 2007 10:43:01 GMT
View Forum Message <> Reply to Message

On Friday 15 June 2007, Mikael Pettersson wrote:

> ...
> > +#if defined(CONFIG_X86_64) || defined(CONFIG_IA64)
> > +/*
> > + * This code works only for 32 bit quota tools over 64 bit OS (x86_64, ia64)
> > + * and is necessary due to alignment problems.
> > + */
>
> The #ifdef looks way too arch-specific. And isn't there a shared
> compat.c module somewhere that this should go into?
>

Only x86_64 and ia64 have this particular problem, the other architectures,
and hopefully all future 64 bit platforms with 32 bit user space use
the same alignment rules in elf32 and elf64.

Still, the patch should be converted to use the compat_u64 type and not
add an 'attribute((packed))' so that you _can_ use the same code on all
architectures. See my 'Introduce compat_u64 and compat_s64 types' patch
that I just posted in another thread.

Arnd <><

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Vasily Tarasov on Fri, 15 Jun 2007 11:00:32 GMT
View Forum Message <> Reply to Message

On Fri, 2007-06-15 at 12:43 +0200, Arnd Bergmann wrote:
> On Friday 15 June 2007, Mikael Pettersson wrote:
> > > > --- linux-2.6.22-rc4-fixed/fs/quota.c.orig 2007-06-14 15:55:26.000000000 +0400
> > > > +++ linux-2.6.22-rc4-fixed/fs/quota.c 2007-06-14 19:50:13.000000000 +0400
> > ...
> > > > +#if defined(CONFIG_X86_64) || defined(CONFIG_IA64)
> > > > +/*
> > > > + * This code works only for 32 bit quota tools over 64 bit OS (x86_64, ia64)
> > > > + * and is necessary due to alignment problems.
> > > > + */
> > >
> > The #ifdef looks way too arch-specific. And isn't there a shared
> > compat.c module somewhere that this should go into?
> >
>
> Only x86_64 and ia64 have this particular problem, the other architectures,
> and hopefully all future 64 bit platforms with 32 bit user space use
> the same alignment rules in elf32 and elf64.
>
> Still, the patch should be converted to use the compat_u64 type and not
> add an 'attribute((packed))' so that you _can_ use the same code on all
> architectures. See my 'Introduce compat_u64 and compat_s64 types' patch
> that I just posted in another thread.
>
> Arnd <><
>

Agree, it'll be the most clean solution.

Is it ok, if I'll send a patch against (current kernel + Arnd's patch)?

Thanks,
Vasily

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Mikael Pettersson on Fri, 15 Jun 2007 11:08:11 GMT
View Forum Message <> Reply to Message

On Fri, 15 Jun 2007 12:43:01 +0200, Arnd Bergmann wrote:
> On Friday 15 June 2007, Mikael Pettersson wrote:
> > > --- linux-2.6.22-rc4-fixed/fs/quota.c.orig=A0=A0=A0=A02007-06-14 15:55:=
> 26.000000000 +0400
> > > +++ linux-2.6.22-rc4-fixed/fs/quota.c=A02007-06-14 19:50:13.000000000 +=
> 0400
> > ...
> > > +#if defined(CONFIG_X86_64) || defined(CONFIG_IA64)
> > > +/*
> > > + * This code works only for 32 bit quota tools over 64 bit OS (x86_64,=
> ia64)
> > > + * and is necessary due to alignment problems.
> > > + */
> >=20
> > The #ifdef looks way too arch-specific. And isn't there a shared
> > compat.c module somewhere that this should go into?
> >=20
>
> Only x86_64 and ia64 have this particular problem, the other architectures,
> and hopefully all future 64 bit platforms with 32 bit user space use
> the same alignment rules in elf32 and elf64.

Ah yes, alignof(u64) is the same in 32- and 64-bit modes on !x86,
thus they don't have a problem here.

Thanks for explaining that. I consider this is an essential
piece of information that should be included in the patch.
(In a comment in the code, not buried in some commit log.)

/Mikael

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Vasily Tarasov on Fri, 15 Jun 2007 14:48:02 GMT
View Forum Message <> Reply to Message

On Fri, 2007-06-15 at 12:43 +0200, Arnd Bergmann wrote:
> On Friday 15 June 2007, Mikael Pettersson wrote:
> > > --- linux-2.6.22-rc4-fixed/fs/quota.c.orig   2007-06-14 15:55:26.000000000 +0400

> > > +++ linux-2.6.22-rc4-fixed/fs/quota.c 2007-06-14 19:50:13.000000000 +0400
> > ...
> > > +#if defined(CONFIG_X86_64) || defined(CONFIG_IA64)
> > > +/*
> > > + * This code works only for 32 bit quota tools over 64 bit OS (x86_64, ia64)
> > > + * and is necessary due to alignment problems.
> > > + */
> >
> > The #ifdef looks way too arch-specific. And isn't there a shared
> > compat.c module somewhere that this should go into?
> >
>
> Only x86_64 and ia64 have this particular problem, the other architectures,
> and hopefully all future 64 bit platforms with 32 bit user space use
> the same alignment rules in elf32 and elf64.
>
> Still, the patch should be converted to use the compat_u64 type and not
> add an 'attribute((packed))' so that you _can_ use the same code on all
> architectures. See my 'Introduce compat_u64 and compat_s64 types' patch
> that I just posted in another thread.
>
> Arnd <><
>

Hello,

I just noticed that we can not avoid the addition of packed attribute.
Look, for example:

```
struct if_dqblk {
	__u64 dqb_bhardlimit;
	__u64 dqb_bsoftlimit;
	__u64 dqb_curspace;
	__u64 dqb_ihardlimit;
	__u64 dqb_isoftlimit;
	__u64 dqb_curinodes;
	__u64 dqb_btime;
	__u64 dqb_itime;
	__u32 dqb_valid;
};
```

sizeof(if_dqblk) = 0x48
On 32 bit: 0x44

If I replace __u64/__u32 with compat equivalents - it will not help!
alligned attribute can _only_ _increase_ the size of structure, but not
decrease it.

So we should use packed or just use the array of ints: int[2].

Vasily

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Arnd Bergmann on Fri, 15 Jun 2007 15:24:16 GMT
View Forum Message <> Reply to Message

On Friday 15 June 2007, Vasily Tarasov wrote:
> I just noticed that we can not avoid the addition of packed attribute.
> Look, for example:
>
> struct if_dqblk {

> };
>
> sizeof(if_dqblk) = 0x48
> On 32 bit: 0x44
>
> If I replace __u64/__u32 with compat equivalents - it will not help!
> alligned attribute can _only_ _increase_ the size of structure, but not
> decrease it.

No, the gcc documentation isn't quite clear there, see the discussion about compat_u64 and compat_s64 types. It actually does the right thing when you use 'typedef __u64 __attribute__((aligned(64))) compat_64', as my patch does.

Arnd <><

Subject: Re: [PATCH] diskquota: 32bit quota tools on 64bit architectures
Posted by Vasily Tarasov on Mon, 18 Jun 2007 07:41:12 GMT
View Forum Message <> Reply to Message

On Fri, 2007-06-15 at 17:24 +0200, Arnd Bergmann wrote:
> On Friday 15 June 2007, Vasily Tarasov wrote:
> > I just noticed that we can not avoid the addition of packed attribute.

> > Look, for example:
> >
> > struct if_dqblk {
> > __u64 dqb_bhardlimit;
> > __u64 dqb_bsoftlimit;
> > __u64 dqb_curspace;
> > __u64 dqb_ihardlimit;
> > __u64 dqb_isoftlimit;
> > __u64 dqb_curinodes;
> > __u64 dqb_btime;
> > __u64 dqb_itime;
> > __u32 dqb_valid;
> > };
> >
> > sizeof(if_dqblk) = 0x48
> > On 32 bit: 0x44
> >
> > If I replace __u64/__u32 with compat equivalents - it will not help!
> > alligned attribute can _only_ _increase_ the size of structure, but not
> > decrease it.
>
> No, the gcc documentation isn't quite clear there, see the discussion about
> compat_u64 and compat_s64 types. It actually does the right thing when
> you use 'typedef __u64 __attribute__((aligned(64))) compat_64', as my
> patch does.
>
> Arnd <><
>

Wow!... Thank you for the explanation.
I'll resend the patch soon.

Vasily