
Subject: [PATCH] ext3: fix ext3_fill_super group description initialization

Posted by [Dmitriy Monakhov](#) on Mon, 13 Aug 2007 09:09:34 GMT

[View Forum Message](#) <> [Reply to Message](#)

->s_group_desc have to be zero filled because if sb_read() failed
we jump to following error path.

failed_mount2:

```
for (i = 0; i < db_count; i++)
```

```
    brelse(sbi->s_group_desc[i]); << Bad things may happen here
```

Signed-off-by: Dmitry Monakhov <dmonakhov@openvz.org>

```
fs/ext3/super.c | 2 +-  
fs/ext4/super.c | 2 +-  
2 files changed, 2 insertions(+), 2 deletions(-)
```

diff --git a/fs/ext3/super.c b/fs/ext3/super.c

index f8ac18f..208738e 100644

--- a/fs/ext3/super.c

+++ b/fs/ext3/super.c

```
@@ -1718,7 +1718,7 @@ static int ext3_fill_super (struct super_block *sb, void *data, int silent)  
    / EXT3_BLOCKS_PER_GROUP(sb)) + 1;
```

```
    db_count = (sbi->s_groups_count + EXT3_DESC_PER_BLOCK(sb) - 1) /
```

```
    EXT3_DESC_PER_BLOCK(sb);
```

```
- sbi->s_group_desc = kmalloc(db_count * sizeof (struct buffer_head *),
```

```
+ sbi->s_group_desc = kzalloc(db_count * sizeof (struct buffer_head *),  
    GFP_KERNEL);
```

```
if (sbi->s_group_desc == NULL) {
```

```
    printk (KERN_ERR "EXT3-fs: not enough memory\n");
```

diff --git a/fs/ext4/super.c b/fs/ext4/super.c

index 8f1d2f6..fefffc0 100644

--- a/fs/ext4/super.c

+++ b/fs/ext4/super.c

```
@@ -1830,7 +1830,7 @@ static int ext4_fill_super (struct super_block *sb, void *data, int silent)
```

```
    sbi->s_groups_count = blocks_count;
```

```
    db_count = (sbi->s_groups_count + EXT4_DESC_PER_BLOCK(sb) - 1) /
```

```
    EXT4_DESC_PER_BLOCK(sb);
```

```
- sbi->s_group_desc = kmalloc(db_count * sizeof (struct buffer_head *),
```

```
+ sbi->s_group_desc = kzalloc(db_count * sizeof (struct buffer_head *),  
    GFP_KERNEL);
```

```
if (sbi->s_group_desc == NULL) {
```

```
    printk (KERN_ERR "EXT4-fs: not enough memory\n");
```

--

1.5.2.2

Subject: Re: [PATCH] ext3: fix ext3_fill_super group description initialization

Posted by [aneesh.kumar](#) on Mon, 13 Aug 2007 09:20:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

Dmitry Monakhov wrote:

```
> ->s_group_desc have to be zero filled because if sb_read() failed
> we jump to following error path.
> failed_mount2:
> for (i = 0; i < db_count; i++)
> brelse(sbi->s_group_desc[i]);<< Bad things may happen here
>
```

But the db_count is updated in the failure path to point to the number of successful sb_read.

-aneesh

Subject: Re: [PATCH] ext3: fix ext3_fill_super group description initialization

Posted by [Dmitry Monakhov](#) on Mon, 13 Aug 2007 09:24:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

On 13:09 Mon 13 Aug , Dmitry Monakhov wrote:

```
> ->s_group_desc have to be zero filled because if sb_read() failed
> we jump to following error path.
> failed_mount2:
> for (i = 0; i < db_count; i++)
> brelse(sbi->s_group_desc[i]);<< Bad things may happen here
OOPs i'm sorry. I've lost db_count changing after sb_read() error,
so error path is ok, and this patch is not needed.
>
> Signed-off-by: Dmitry Monakhov <dmonakhov@openvz.org>
> ---
> fs/ext3/super.c | 2 +-
> fs/ext4/super.c | 2 +-
> 2 files changed, 2 insertions(+), 2 deletions(-)
>
> diff --git a/fs/ext3/super.c b/fs/ext3/super.c
> index f8ac18f..208738e 100644
> --- a/fs/ext3/super.c
> +++ b/fs/ext3/super.c
> @@ -1718,7 +1718,7 @@ static int ext3_fill_super (struct super_block *sb, void *data, int
> silent)
>     / EXT3_BLOCKS_PER_GROUP(sb)) + 1;
>     db_count = (sbi->s_groups_count + EXT3_DESC_PER_BLOCK(sb) - 1) /
>     EXT3_DESC_PER_BLOCK(sb);
> - sbi->s_group_desc = kmalloc(db_count * sizeof (struct buffer_head *),
> + sbi->s_group_desc = kzalloc(db_count * sizeof (struct buffer_head *),
```

```

> GFP_KERNEL);
> if (sbi->s_group_desc == NULL) {
>   printk (KERN_ERR "EXT3-fs: not enough memory\n");
> diff --git a/fs/ext4/super.c b/fs/ext4/super.c
> index 8f1d2f6..feffc0 100644
> --- a/fs/ext4/super.c
> +++ b/fs/ext4/super.c
> @@ -1830,7 +1830,7 @@ static int ext4_fill_super (struct super_block *sb, void *data, int
> silent)
>   sbi->s_groups_count = blocks_count;
>   db_count = (sbi->s_groups_count + EXT4_DESC_PER_BLOCK(sb) - 1) /
>     EXT4_DESC_PER_BLOCK(sb);
>   - sbi->s_group_desc = kmalloc(db_count * sizeof (struct buffer_head *),
>   + sbi->s_group_desc = kzalloc(db_count * sizeof (struct buffer_head *),
>     GFP_KERNEL);
>   if (sbi->s_group_desc == NULL) {
>     printk (KERN_ERR "EXT4-fs: not enough memory\n");
>   --
> 1.5.2.2
>
>
> -
> To unsubscribe from this list: send the line "unsubscribe linux-ext4" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at http://vger.kernel.org/majordomo-info.html

```

Subject: Re: [PATCH] ext3: fix ext3_fill_super group description initialization

Posted by [Mariusz Kozlowski](#) on Mon, 13 Aug 2007 13:19:38 GMT

[View Forum Message](#) <> [Reply to Message](#)

```

> ->s_group_desc have to be zero filled because if sb_read() failed
> we jump to following error path.
> failed_mount2:
> for (i = 0; i < db_count; i++)
>   brelse(sbi->s_group_desc[i]); << Bad things may happen here
>
> Signed-off-by: Dmitry Monakhov <dmonakhov@openvz.org>
> ---
> fs/ext3/super.c | 2 +-
> fs/ext4/super.c | 2 +-
> 2 files changed, 2 insertions(+), 2 deletions(-)
>
> diff --git a/fs/ext3/super.c b/fs/ext3/super.c
> index f8ac18f..208738e 100644
> --- a/fs/ext3/super.c
> +++ b/fs/ext3/super.c
> @@ -1718,7 +1718,7 @@ static int ext3_fill_super (struct super_block *sb,

```

```
> void *data, int silent) / EXT3_BLOCKS_PER_GROUP(sb)) + 1;
> db_count = (sbi->s_groups_count + EXT3_DESC_PER_BLOCK(sb) - 1) /
> EXT3_DESC_PER_BLOCK(sb);
> - sbi->s_group_desc = kmalloc(db_count * sizeof (struct buffer_head *),
> + sbi->s_group_desc = kzalloc(db_count * sizeof (struct buffer_head *),
> GFP_KERNEL);
```

kcalloc?

```
> if (sbi->s_group_desc == NULL) {
> printk (KERN_ERR "EXT3-fs: not enough memory\n");
> diff --git a/fs/ext4/super.c b/fs/ext4/super.c
> index 8f1d2f6..feffc0 100644
> --- a/fs/ext4/super.c
> +++ b/fs/ext4/super.c
> @@ -1830,7 +1830,7 @@ static int ext4_fill_super (struct super_block *sb,
> void *data, int silent) sbi->s_groups_count = blocks_count;
> db_count = (sbi->s_groups_count + EXT4_DESC_PER_BLOCK(sb) - 1) /
> EXT4_DESC_PER_BLOCK(sb);
> - sbi->s_group_desc = kmalloc(db_count * sizeof (struct buffer_head *),
> + sbi->s_group_desc = kzalloc(db_count * sizeof (struct buffer_head *),
> GFP_KERNEL);
```

kcalloc?

```
> if (sbi->s_group_desc == NULL) {
> printk (KERN_ERR "EXT4-fs: not enough memory\n");
```

Regards,

Mariusz
