
Subject: [PATCH] [RESEND] ext[34] orphan list check on destroy_inode

Posted by [vaverin](#) on Tue, 05 Jun 2007 10:19:53 GMT

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

Customers claims to ext3-related errors, investigation showed that ext3 orphan list has been corrupted and have the reference to non-ext3 inode. The following debug helps to understand the reasons of this issue.

Signed-off-by: Vasily Averin <vvs@sw.ru>

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

```
index 6e30629..e986376 100644
```

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

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

```
@@ -459,6 +459,13 @@ static struct inode *ext3_alloc_inode(struct super_block *sb)
```

```
static void ext3_destroy_inode(struct inode *inode)
{
+ if (!list_empty(&(EXT3_I(inode)->i_orphan))) {
+   printk("EXT3 Inode %p: orphan list check failed!\n",
+     EXT3_I(inode));
+   print_hex_dump(KERN_INFO, DUMP_PREFIX_ADDRESS,
+     EXT3_I(inode), sizeof(struct ext3_inode_info));
+   dump_stack();
+ }
  kmem_cache_free(ext3_inode_cachep, EXT3_I(inode));
}
```

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

```
index cb9afdd..935420d 100644
```

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

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

```
@@ -510,6 +510,13 @@ static struct inode *ext4_alloc_inode(struct super_block *sb)
```

```
static void ext4_destroy_inode(struct inode *inode)
{
+ if (!list_empty(&(EXT4_I(inode)->i_orphan))) {
+   printk("EXT4 Inode %p: orphan list check failed!\n",
+     EXT4_I(inode));
+   print_hex_dump(KERN_INFO, DUMP_PREFIX_ADDRESS,
+     EXT4_I(inode), sizeof(struct ext4_inode_info));
+   dump_stack();
+ }
  kmem_cache_free(ext4_inode_cachep, EXT4_I(inode));
}
```

Subject: Re: [PATCH] [RESEND] ext[34] orphan list check on destroy_inode

Posted by [Carsten Otte](#) on Tue, 05 Jun 2007 10:34:31 GMT

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

Vasily Averin wrote:

> Customers claims to ext3-related errors, investigation showed that ext3 orphan
> list has been corrupted and have the reference to non-ext3 inode. The following
> debug helps to understand the reasons of this issue.

This looks like it might be related to the -as far as I recall-
unfixed issue we've noticed some time ago:

<http://osdir.com/ml/file-systems/2004-02/msg00033.html>

I have'nt heard any reports of a reproduction on modern kernels on
390 lately, our issue seems to have disappeared.

so long,
Carsten
