
Subject: [PATCH] ecryptfs lower_file handling issues

Posted by [Dmitriy Monakhov](#) on Mon, 19 Feb 2007 14:33:51 GMT

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

1)Function ecryptfs_do_readpage() calls flush_dcache_page(lower_page), but lower_page wasn't changed here. So remove this line.

2)prepare_write ret val was ignored in ecryptfs_write_inode_size_to_header(). If error happens we can't call commit_write, just do cleanup and fail. It is issue easy to reproduce with full lower_fs, in this case prepare_write() will return ENOSPC.

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

```
diff --git a/fs/ecryptfs/mmap.c b/fs/ecryptfs/mmap.c
```

```
index 1e5d2ba..0cebb75 100644
```

```
--- a/fs/ecryptfs/mmap.c
```

```
+++ b/fs/ecryptfs/mmap.c
```

```
@@ -238,7 +238,6 @@ int ecryptfs_do_readpage(struct file *file, struct page *page,
    lower_page_data = kmap_atomic(lower_page, KM_USER1);
    memcpy(page_data, lower_page_data, PAGE_CACHE_SIZE);
    kunmap_atomic(lower_page_data, KM_USER1);
- flush_dcache_page(lower_page);
    kunmap_atomic(page_data, KM_USER0);
    flush_dcache_page(page);
    rc = 0;
```

```
@@ -454,6 +453,12 @@ static int ecryptfs_write_inode_size_to_header(struct file *lower_file,
    }
    lower_a_ops = lower_inode->i_mapping->a_ops;
    rc = lower_a_ops->prepare_write(lower_file, header_page, 0, 8);
+ if (rc) {
+   ecryptfs_printk(KERN_ERR, "Error preparing header page "
+     "write\n");
+   ecryptfs_release_lower_page(header_page);
+   goto out;
+ }
    file_size = (u64)i_size_read(inode);
    ecryptfs_printk(KERN_DEBUG, "Writing size: [0x%.16x]\n", file_size);
    file_size = cpu_to_be64(file_size);
```
