Subject: Re: [PATCH 3/6] fuse: rework fuse_retrieve()
Posted by Miklos Szeredi on Wed, 12 Sep 2012 16:23:55 GMT
View Forum Message <> Reply to Message

Maxim Patlasov <mpatlasov@parallels.com> writes:

> The patch reworks fuse_retrieve() to allocate only so many page pointers
> as needed. The core part of the patch is the following calculation:
>
> num_pages = (num + offset + PAGE_SIZE - 1) >> PAGE_SHIFT;
>
> (thanks Miklos for formula). All other changes are moslty shuffling lines.
>
> The patch also contains one-line fix suggested by Miklos: the loop iterating
> over pages from inode mapping should interpret 'offset' as offset in the very
> first page only.

That fix is already in -linus, so let's leave it out of this patch.

Thanks,
Miklos

> ---
> fs/fuse/dev.c | 26 ++++++++++++++++++----------
> 1 files changed, 16 insertions(+), 10 deletions(-)
>
> diff --git a/fs/fuse/dev.c b/fs/fuse/dev.c
> index 6e38af5..cf69df3 100644
> --- a/fs/fuse/dev.c
> +++ b/fs/fuse/dev.c
> @@ -1562,13 +1562,24 @@ static int fuse_retrieve(struct fuse_conn *fc, struct inode *inode,
>   unsigned int num;
>   unsigned int offset;
>   size_t total_len = 0;
> + int num_pages;
>
> - req = fuse_get_req_multipage(fc, FUSE_MAX_PAGES_PER_REQ);
> + offset = outarg->offset & ~PAGE_CACHE_MASK;
> + file_size = i_size_read(inode);
> +
> + num = outarg->size;
> + if (outarg->offset > file_size)
> + num = 0;
> + else if (outarg->offset + num > file_size)
> + num = file_size - outarg->offset;
> +
> + num_pages = (num + offset + PAGE_SIZE - 1) >> PAGE_SHIFT;
> + num_pages = min(num_pages, FUSE_MAX_PAGES_PER_REQ);

```
> +
> + req = fuse_get_req_multipage(fc, num_pages);
>   if (IS_ERR(req))
>    return PTR_ERR(req);
>
> - offset = outarg->offset & ~PAGE_CACHE_MASK;
> -
>   req->in.h.opcode = FUSE_NOTIFY_REPLY;
>   req->in.h.nodeid = outarg->nodeid;
>   req->in.numargs = 2;
> @@ -1577,14 +1588,8 @@ static int fuse_retrieve(struct fuse_conn *fc, struct inode *inode,
>   req->end = fuse_retrieve_end;
>
>   index = outarg->offset >> PAGE_CACHE_SHIFT;
> - file_size = i_size_read(inode);
> - num = outarg->size;
> - if (outarg->offset > file_size)
> -  num = 0;
> - else if (outarg->offset + num > file_size)
> -  num = file_size - outarg->offset;
>
> - while (num && req->num_pages < FUSE_MAX_PAGES_PER_REQ) {
> + while (num && req->num_pages < num_pages) {
>    struct page *page;
>    unsigned int this_num;
>
> @@ -1596,6 +1601,7 @@ static int fuse_retrieve(struct fuse_conn *fc, struct inode *inode,
>    req->pages[req->num_pages] = page;
>    req->num_pages++;
>
> +  offset = 0;
>    num -= this_num;
>    total_len += this_num;
>    index++;
```