

JOS Lab5 文档

杨健邦 515030910223

本文档描述了本次 LAB 各个 *Exercise* 实现的方法或者我自己的理解。至于对 *Question* 的解答以及 *Challenge* 的实现方法，请阅读 *answers-lab5.txt*。

Disk Access

Exercise 1

在 `env_create` 中判断 `type` 是否等于 `ENV_TYPE_FS`，如果等于的话，就将 `e->env_tf.eflags |= FL_IOPL_3`，这表示的是在这个 `env` 中，RING 3 及以上特权可以进行 IO 操作，以及用户态可以操作 IO。

The Block Cache

Exercise 2.

先用 `sys_page_alloc` 分配一块物理页，接着用 `ide_read` 将文件的内容读入物理页，最后 `sys_page_map` 清除 `dirty_bit`。

The Block Bitmap

Exercise 3.

从 0 开始一直到 `super->s_nblocks`，逐一搜索 `bitmap`，直到找到第一个为 0 的位，将其置为 1，并且立刻将 `bitmap block` flush，然后再返回这个位在 `bitmap` 中的位置（第几个 bit）。

File Operations

Exercise 4.

对于 `file_block_walk`，先判断 `filebno` 有没有超出 `NDIRECT` 以及 `NINDIRECT` 之和，如果是在 `NDIRECT` 之内的，直接返回 `f_direct` 数组中的结果。如果在 `NINDIRECT` 里面，则根据参数，如果有需要的话分配一个新的物理页来存放 `NINDIRECT` 数组，并且返回数组中的结果。

对于 `file_get_block`，其逻辑与 `file_block_walk` 类似，如果文件相对应的 `block` 不存在的话，还需要调用 `alloc_block` 的方法分配一个新的 `block`。

Client/Server File System Access

Exercise 5.

在 `serve_read` 中，先根据文件 `id` 找到相应的 `openfile` 对象，通过 `file_read` 的方法读出文件的内容，并修改 `fd` 中的 `offset`，要注意读取字符的数量等于 `MIN(req->req_n, PGSIZE)`，防止 `buf` 溢出。

在 `devfile_read` 中，将要读的文件 `id` 以及要读的字符数设置好，并将 IPC 的方法标识号设置为 `FSREQ_READ`，如果 IPC 的返回值大于等于 0，则将返回的内容 `move` 到 `buf` 中。

Exercise 6.

在 `serve_write` 中，先根据文件 `id` 找到相应的 `openfile` 对象，通过 `file_write` 将内容写入文件，并修改 `fd` 中的 `offset`，要注意写入的字符数量等于 `MIN(req->req_n, PGSIZE - (sizeof(int) + sizeof(size_t)))`，防止 `buf` 溢出。

在 `devfile_write` 中，将要写的文件 `id` 以及要写的字符数设置好，并将 `buf` 的内容 `move` 到 `fsipcbuf.write.req.buf` 中，`move` 的字符数量为 `MIN(n, PGSIZE - (sizeof(int) + sizeof(size_t)))`，防止 `buf` 溢出，最后将 IPC 的方法标识号设置为 `FSREQ_WRITE`，调用 IPC。

Client-Side File Operations

Exercise 7.

先调用 `fd_alloc` 得到一个未被使用的 `fd`，这里不需要为这个 `fd` 分配一块物理页。然后调用 IPC，方法标识号为 `FSREQ_OPEN`，这个 IPC 会自动为这个刚分配的 `fd` 分配一块物理页，并将相应文件打开，绑定到这个 `fd` 上。最后通过 `fd2num(fd_store)` 返回这个 `fd` 的 `index`。要注意 IPC 返回失败结果时，要关闭 `fd`，释放掉 IPC 为这个 `fd` 分配的物理页。

Spawning Processes

Exercise 8.

在 `sys_env_set_trapframe` 中，先检查 `tf` 的地址是否有效，然后将 `tf` 拷贝到对应的 `env` 的 `tf` 中，最后将 `env` 的 `tf` 的 `cs` 或上 `0x3`、`tf_flags` 或上 `FL_IF`，以确保新的 `env` 是运行在用户态并且中断是开启的。