

16 级计科 7 班: 操作系统原理实验 #5

Due on Friday, April 13, 2018

凌应标 周一 9-10 节

颜彬

16337269

Content

	Page
1 实验目的	3
2 实验要求	3
3 实验方案	3
3.1 基础原理	3
3.2 实验环境	3
3.2.1 系统与虚拟机	3
3.2.2 相关工具、指令	3
3.3 程序流程	3
4 实验过程	4
5 实验总结	4
5.1 亮点介绍	4
5.2 心得体会	4
5.3 BUG 汇总	4
A 参考文献	4
B 其他代码	5

1 实验目的

2 实验要求

3 实验方案

3.1 基础原理

3.2 实验环境

3.2.1 系统与虚拟机

- 操作系统
本实验在 Linux 下完成。采用 Ubuntu 16.04
- 虚拟机
bochs. 它是一款开源且跨平台的 IA-32 模拟器。

3.2.2 相关工具、指令

- 汇编器
NASM. NASM 是一个轻量级的、模块化的 80x86 和 x86-64 汇编器。它的语法与 Intel 原语法十分相似，但更加简洁和易读。它对宏有十分强大的支持。
- 编译器
GCC. GNU/GCC 是开源的 C 语言编译器。其产生的伪 16 位代码可以与 NASM 结合，混编生成伪 16 位程序。
- 镜像文件产生工具
bxiimage. 该命令允许生成指定大小的软件镜像。
- 二进制写入命令
dd. dd 允许指定源文件和目标文件，将源文件的二进制比特写入目标文件中的指定位置。
- 二进制文件查看命令
xxd. xxd 允许将二进制文件中的内容按地址顺序依次输出，可读性强
- 反汇编器
objdump. objump 可以查看目标文件和二进制文件的反汇编代码，还能指令 intel 或 at&t 格式显示。
- 代码生成脚本
makefile. makefile 脚本具有强大的功能，其可以识别文件依赖关系，自动构筑文件，自动执行 shell 脚本等。

3.3 程序流程

文件系统在虚拟软盘中的镜像如图??。全部引导的简要过程见图??。

表 1: GCC 编译指令

编译选项	作用
-o	D
-march=i386	X 产生原始 Itel i376 CPU 架构的代码
-m16	等价于汇编伪指令 <code>.code16gcc</code> 。 在所有必要的指令前加, 前缀 66, 67, 使得处理器能以 32 位的方式读取 GCC 产生的代码
-mpreferred-stack-boundary=2	以 2^i bytes 作为对齐量, 对齐栈的界限。 该选项默认为 4(即默认 2^4 字节对齐栈界限)。 为了节省空间, 这里设置成 2
-ffreestanding	产生“自立”的程序文件。 即告知 GCC 不使用(几乎)所有的库头文件。 保证产生的代码不依赖于 GCC 的自带库
-c	不产生可执行文件, 只执行“编译”操作产生目标文件
-Og	产生最小限度的优化。 即这个优化能化简汇编代码, 但又不影响汇编程序的可读性。
-O0	不产生任何优化。 本项目文件系统 FAT 表的构建使用 C 语言的 struct 完成。 为了保证产生的二进制表项和 C 代码中的 struct 严格一致 必须采用本优化等级。

4 实验过程

5 实验总结

5.1 亮点介绍

5.2 心得体会

5.3 BUG 汇总

附录 A 参考文献

sasadf

1. <https://blog.csdn.net/longintchar/article/details/50602851>
16 位和 32 位汇编指令的不同 (尤其是 push 指令)
2. <https://www.ibiblio.org/gferg/ldp/GCC-Inline-Assembly-HOWTO.html#s1>
GCC 内嵌汇编的书写。

3. <http://blog.51cto.com/dengqi/1349327> FAT32 文件系统讲解
4. <https://blog.csdn.net/yeruby/article/details/41978199> FAT16 文件系统讲解

附录 B 其他代码