

实验思考题

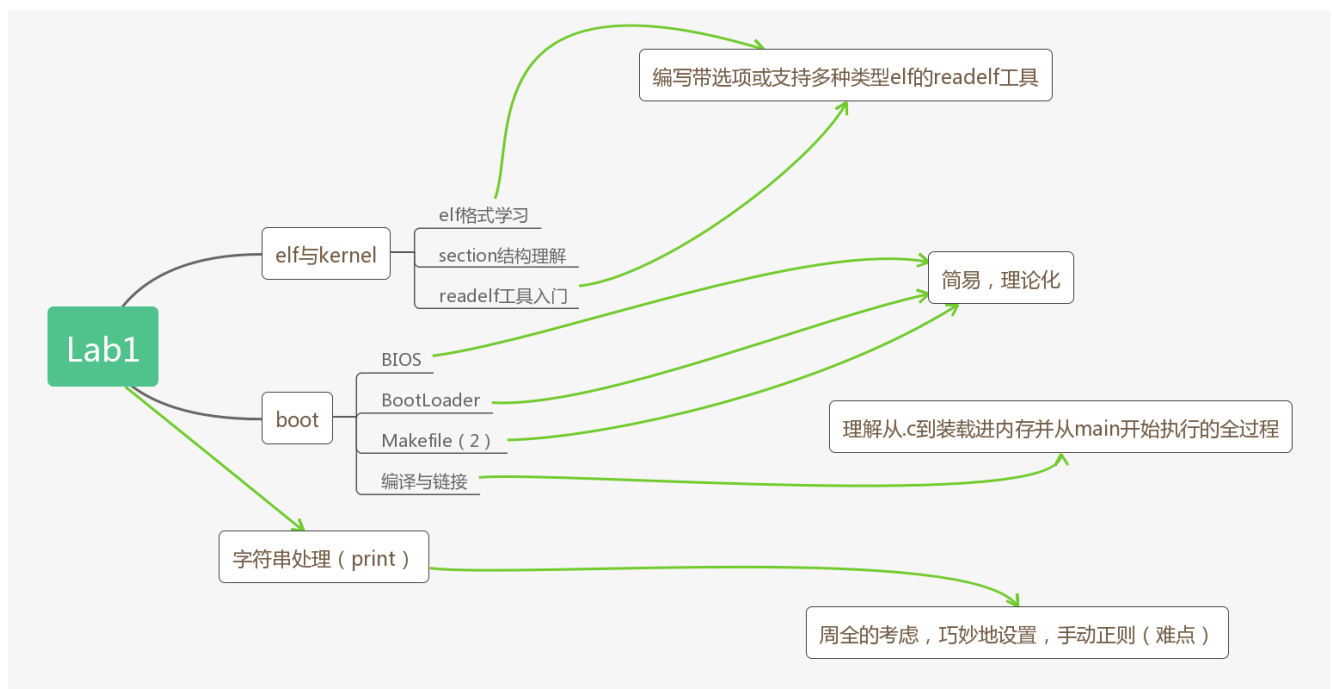
1.1

- vmlinux在ehdr中显示，这是大端存储的，因此课下生成的readelf无法解析

1.2

- main函数被装载到了某一段内存中
- 在执行jal main这一句话的时候，实际上“main”被当作了一个立即数。跨文件调用函数的前提是链接器已经将他们链接完毕并在jal xxx等位置填入了正确的地址

难点



难度评价

总体来说难度不大，相当于数据结构+状态机+计组+操作系统的大杂烩。真正属于操作系统核心的部分，比如编译链接的过程和启动的过程，要么被模拟器所简化，要么以script文件的方式呈现出来。

理论理解不太晦涩，实践起来好上手。大小端和汇编等内容几乎都是计组内容的重复与巩固，最难最坑的print题，难点也不在于了解操作系统本身的框架与流程，而是在于C的基本功和字符串处理。

花费时间

总体来说花费大约5个小时，还算比较多。其中相当一部分时间花在了理解elf格式和print思考上面

尤其是print，关于很多格式的处理感觉自己思考的并不透彻，处理起来也很狼狈

面对课上题目，尤其是gcc与Makefile相关，更是无从下手，最后只好靠cp或者rm等指令狼狈过关

指导书反馈

建议在有关elf的描述中，把elf结构的图片进行更改。作为一个第一次接触elf的人，我第一反应是，开头一个头，然后接着是ph头，然后是各个链接库的段，最后是shdr的头，以至于在一开始从shoff向低地址方向与寻找section的信息。后来查阅了网络资料，才看懂这幅图真正表达的意思。建议化成两张图，一张是ehdr，phtable，shtable，然后ph和sh部分伸出许多箭头，从两个侧面指向section/segment那一段，即把shdr花在紧邻phdr后面的位置，方便理解。

建议仿照lab0提交文件树/样例那样，给lab1提供更多的资料，或者在cscore上面放出更多的资料——(p.s. 如果课程组本意是让我们练习查看指导书（将来可能的说明书），并锻炼学习陌生事物的能力的话，那这条建议作废）

残留难点

如何有效的排列print检索各个符号的优先级