

PA0实验报告

231300027朱士杭

注意：必答题在最下面请耐心等待看完全部再去看必答题！！

2024.09.10 About VirtualMachine

早在上个学期，我就已经尝试过自己去安装虚拟机与Ubuntu22.04开始尝试熟悉shell命令行环境，但是最终无终而返，不管安装的是22.04版本的还是23.04版本的，在VM WorkStation上面最终都能成功安装，但是一到连接网络那一块就出问题了，根本就连接不上校园网，也STFW过很多遍但是都没有找到解决方案，最后没办法只能随它去了，

2024.09.10 About PA0 Pre

今天准备做完PA0，一开始没有什么问题，直到make menuconfig这一步出了问题，通过上网搜为什么没有make相关命令，结果一开始以为是自己没有安装linux内核，然后开始重新安装内核，结果又卡死在解压这一步，一直显示no sun file or directory 然后我又STFW，然后又有回答说这个可能是因为没有在linux内核源代码目录下面运行make命令，然后我就去/usr/src/目录下面但是还是需要先解压下载下来的linux-source.tar.gz压缩包，但是一直解压失败，一气之下给人家linux源代码文件夹直接全删除了，之后也没有找回来（不知道会不会对之后产生影响）。最后实在弄不好只能先去上课，结果在课上百思不得其解，

突然一想到好像需要进入nemu文件夹下再运行make menuconfig，结果自己一直在ics2024/这个文件夹下面，难怪会出问题 后来回来之后又重新进行make操作，确实少安装了[libncurses5]

{https://blog.csdn.net/Wanguang_/article/details/85229457}，于是进行必要的安装：

```
sudo apt-get install libncurses5-dev
```

然后发现自己还少了bison工具，进行相应的安装之后就可以正常compile了

```
sudo apt-get install bison sudo apt-get install flex
```

(其实这里自己还忘记安装flex因此会提示Error:flex:No such File or Directory但是和前面没啥关系所以暂时就不管了)

之后会出现一个图形化窗口，注意什么都不要做，向右进行移动然后exit退出即可（好像是因为这个是config配置文件不能随便修改，万一哪里改错了之后make编译不了了)

2024.09.10 About PA0 Pdf & Remote

在PA0 manual的最后有提到让我们自己去STFW如何进行pdf文件的查看，查到可以用evince命令查看pdf文件，但是目前我这个ubuntu虚拟机中还没有任何pdf文档可以试，这里先记着，等回头有机会了再试一试。

现在我又遇到了一个问题，我不只是想在本地存储文件，因为我有2台电脑嘛，所以我想把代码推送到远程github仓库中去，之后我可以方便访问，结果一git remote才发现本地git仓库已经和远程仓库连接了，但是我又怕如果断开这个连接之后做PA需要访问到远程内容不方便，因此我在这里先暂且保存一下远程仓库的URL，然后再remote连接到自己的URL当中保存，之后如果需要访问NJU的仓库再重新连接也未尝不可。

```
edzee3000@edzee3000-VMware-Virtual-Platform:~/ICS2024/ics2024$ git remote
show origin * remote origin Fetch URL: git@github.com:NJU-ProjectN/ics-
pa.git Push
URL: git@github.com:NJU-ProjectN/ics-pa.git HEAD branch: 2024
Remote branches:
    2017 tracked
    2018 tracked
    2019 tracked
    2020 tracked
    2021 tracked
    2022 tracked
    2023 tracked
    2024 tracked
Local branch configured for 'git pull': master merges with remote 2024
```

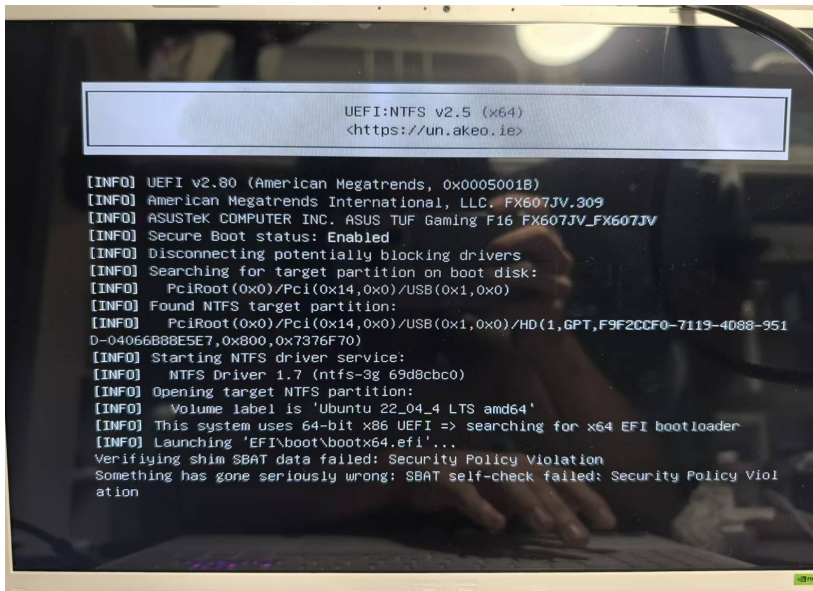
下面是一些与远程仓库进行连接的操作：

```
git remote add [远程仓库名] [远程仓库URL] //与远程仓库进行连接
git remote //显示远程仓库名称
git fetch master //抓取远程仓库数据
git branch -a //查看所有分支
git checkout master //合并远程仓库到本地
git remote rm [远程仓库名] //本地删除远程仓库
```

2024.09.11 About Real Machine

昨天尝试了一下使用虚拟机ubuntu完成PA0，确实挺顺利的，但是因为以后可能用到虚拟机的地方很多，因此不妨一劳永逸直接装了一个Windows11+Ubuntu24.04LTS的双操作系统（不要问我为什么不按照要求安装22.04版本因为以前安装过22.04版本但是觉得它比24.04丑太多了，而且ubuntu一般也会向下兼容吧，那应该就不会出现什么大问题）

安装windows和ubuntu双系统最痛苦的就是等待了，一个是要等待驱动盘制作，一个是要等待ubuntu重启从驱动盘读取数据开始安装，这些都不是大问题，有个致命的问题就是我当时从驱动盘进入的时候会出现"Verifying shim SBAT data failed:Security Policy Violation"的错误：



百思不得其解，最后决定使用同学已经做好的24.04 LTS版本的驱动盘就可以重新安装了，并且最后也成功安装，给真机留了大概200G的存储空间（应该是顶够了）

估计安装过程中的bug是因为bois里面某个东西没关，因为我的舍友也是出现了类似的问题，他关了某个东西就解决了，但是既然已经装好了那就算了。

其实安装过程中还出现了一个问题导致ubuntu安装失败，就是手动分区那里，看了网上的很多教程结果跟自己的电脑不一样，结果手动分区导致安装失败，重新安装

2024.09.11 About Clone

但是今天从github上clone下来昨天完成的的部分的时候发现~/.bashrc下面的环境变量好像需要自己手动配置但是我不会，因此只好把前面做好的给删掉然后重新source一下，然后发现可以echo出对应的路径名称了。

接下来查看.bashrc下的内容，发现多了以下两行：

```
export NEMU_HOME=/home/edzee3000/ICS2024/ICS2024PA/nemu
export AM_HOME=/home/edzee3000/ICS2024/ICS2024PA/abstract-machine
```

好了，接下来就可以放心地将自己的代码仓库上传到github当中去了，因为已经学会配置~/.bashrc文件了，如果需从nju的github上获取相关内容也就很容易了

2024.09.13 Answers to Some Questions in PA0

为什么在某些 Linux 发行版中执行“poweroff”命令需要超级用户权限？

因为poweroff特别危险，涉及到很多linux内核的操作，而且需要操作系统介入去做一些“善后工作”，稍有不慎很可能导致数据丢失。在执行poweroff命令时，系统执行以下操作：

1. 发送关机信号：当执行poweroff命令时，Shell会向系统发送一个关机信号。这个信号通常是SIGTERM，它告诉系统需要进行关机操作。
2. 关闭所有进程：系统收到关机信号后，会开始关闭所有正在运行的进程。首先，系统会向所有运行的进程发送SIGTERM信号，要求它们优雅地终止。如果进程没有响应SIGTERM信号，系统会发送SIGKILL信号

(9号信号) 来强制终止这些进程。

3. 卸载文件系统：在关闭进程后，系统会卸载所有挂载的文件系统。这包括根文件系统和其他挂载的文件系统，如/home /var等。卸载文件系统是为了确保数据的完整性，并防止在关机过程中出现错误。
4. 关闭设备：在卸载文件系统后，系统会关闭硬件设备。这包括关闭磁盘驱动器、网络接口、USB设备等。关闭设备是为了确保系统在关机过程中不会出现任何问题，并保护硬件设备的完整性。
5. 最后才是进行关机操作：当所有进程、文件系统和设备都已关闭后，系统会执行最后的关机操作。这包括停止CPU的运行、断电等操作，以完全关闭系统。

为什么不要在root账户中做PA

安全！！！！

以 root 身份登录 相当于 所有应用程序都以 root 权限运行，Firefox ·Flash ·OpenOffice 等中的每个漏洞现在都可以摧毁整个操作系统，问问自己真的有那个能力去把握整个操作系统不会出问题吗？误操作可能会无意中损坏系统文件，导致系统无法启动，自己有能力去解决这个问题吗？

当自己是root身份时，可以执行任何操作并且系统不会询问！是否要格式化此磁盘？好的，只需单击一下就完成了，因为自己是 root，并且您知道自己在做什么.....

What Happened in Make ?

有makefile的存在，我们可以直接make就行了，但是make究竟是什么？我自己认为make相当于一个自动化程序帮你自动执行编译汇编链接的操作，通过一个生成树根据每个文件之间的依赖关系自动进行链接，但是当我看过makefile里面的内容时说实话我并没有看懂，所以也只能这样了，比如进行make run命令的时候，我只需要知道抽象地表达的含义即可（即开始运行程序），但是其背后的逻辑依赖关系就不需要我再去关心了（因为我也看不懂），避免给我弱小的心灵带来打击。

2024.09.13 End for PA0

好的，那PA0就到此为止啦，这也是我第一次使用markdown写实验日志（应该叫实验报告），还有很多有关于vim ·shell等命令学习，我不知道从何开始写起，还有好多当时做PA0过程中想要记录下来的东西，可惜现在都忘了（比如在ubuntu上安装微信QQ输入法edge浏览器等等）其实很多都可以写，但是跟实验报告关系不大，我就把它们都放在ICS2024/Experiment_Log/Attention.txt下面了，类似于一种实验日志随笔吧，有兴趣可以去看一看

2024.09.20 必答题

PA0的训练让我真切感受到了“好的提问”和“搜索查找”的重要性

“好的提问”和“通过STFW（Search The Fing Web）和RTFM（Read The Fing Manual）独立解决问题”是技术和学术领域中常见的两个概念，它们对于提高个人解决问题的能力 and 促进有效的沟通都非常重要。

在提问之前，应该先尝试自己解决问题，比如通过搜索、阅读文档、尝试不同的解决方案等。确保你的问题具体、明确，并且提供了足够的信息，以便他人能够理解并帮助你。即使是在寻求帮助时，也应该保持礼貌和尊重。认识到回答者是在自愿地花费自己的时间来帮助你。并且说话的时候尽量使用清晰、准确、语法正确的语言。提供详细的背景信息和问题描述，包括你已经尝试的步骤和遇到的具体问题。选择合适的论坛或邮件列表提问，避免在不相关的地方提问，避免在多个地方重复提问。如果问题得到解决，应该更新问题状态，让其他人知道问题已经解决，对于提供帮助的人表示感谢。

好的提问

我个人认为一个“好的提问”首先要遵循“提问的智慧”和“别像弱智一样提问”中提到的相关建议，除此之外，我自己认为一个好的提问还需要

1. 明确具体：一个好的问题应该是明确和具体的，这样回答者才能准确理解问题的核心，避免给出模糊或不相关的回答。
2. 背景信息：在提问时提供足够的背景信息和上下文，比如你已经尝试过的解决方法、遇到问题的环境或条件，这些都有助于他人更快地定位问题。
3. 简洁清晰：尽管提供背景信息很重要，但问题本身应该尽量简洁，避免冗长和复杂，这样更容易吸引他人回答。
4. 尊重他人时间：认识到回答者是在花费自己的时间帮助你，因此提问时应该表现出尊重和感激。
5. 避免重复：在提问之前，先搜索看看是否已经有现成的答案，这样可以避免重复提问，节省大家的时间。

通过STFW和RTFM独立解决问题

在提问之前，自己先进行深入的研究和尝试，这是对自己和他人时间的尊重。清晰、准确地描述问题，能够让回答者更快地理解问题并提供帮助。遵守社区的规则和文化，比如在正确的论坛提问，使用合适的语言等。对于那些愿意花时间帮助我的人，我学会了表达感激之情。通过提问和回答，学习是一个双向的过程，我也可以成为帮助他人的人。

通过STFW和RTFM独立解决问题帮助很大（真的超级有意义!!!）：

1. 培养自主能力：STFW和RTFM鼓励个人自主寻找解决方案，这有助于培养解决问题的能力 and 自我学习能力。
2. 提高效率：通过自己搜索和阅读手册，可以更快地找到问题的解决方法，而不是等待他人回答。
3. 减少依赖：减少对他人帮助的依赖，可以在没有外部帮助的情况下也能独立完成任务。
4. 促进社区发展：当每个人都能通过自己的努力解决问题时，可以减少社区中重复问题的数量，使得社区资源可以更集中地解决新问题或更复杂的问题。
5. 避免信息过载：在没有进行充分搜索和尝试之前就提问，可能会导致信息过载，因为可能已经有很多类似问题被讨论过。

总的来说，好的提问是一种艺术，它需要提问者具备清晰的表达能力和对问题背景的深入了解。而STFW和RTFM则是一种自我提升的过程，它们鼓励个人在寻求帮助之前先尽自己最大的努力。这两者都是技术社区中推崇的做法，有助于建立一个更加高效和互助的环境。