



深圳大学  
Shenzhen University

# 操作系统

Lab1: xv6实验操作系统介绍  
谭舜泉  
计算机与软件学院

# XV6介绍

- xv6 是 MIT 开发的一个教学用的完整的类 Unix 操作系统，并且在 MIT 的操作系统课程 6.828 中使用。
- 通过阅读并理解 xv6 的代码，可以清楚地了解操作系统中众多核心的概念，对操作系统感兴趣的同学十分推荐一读！
- 在国内，包括清华大学等众多大学已经采用xv6作为实验操作系统。



深圳大学  
Shenzhen University

# 相关文档

- 网上有中文翻译的 MIT xv6 文档，是阅读代码过程中非常好的参考资料。
  - <https://www.gitbook.com/download/pdf/book/t-h0ar/xv6-chinese>
- 以下为对应的xv6 源代码，需要同上述文档一同阅读！
  - <http://pdos.csail.mit.edu/6.828/2012/xv6/xv6-rev7.pdf>



深圳大學  
Shenzhen University

# 应该选用哪种xv6版本？

- 最新的x86版本已经到了rev11（2018），后续的版本迁移到了RISC-V架构。
- 目前只有中rev7有对应的文版，建议大家还是参考rev7。
- 我们在centos7, ubuntu16.4上测试过，能顺利编译运行rev7和rev9。
- 在ubuntu18.4上，由于ubuntu18.4上qemu的一个bug，无法运行rev7，但能运行rev9。使用ubuntu18.4的同学可以使用rev9，但注意rev7和rev9的代码有细微的区别。



# xv6-rev9的相关文档可在以下链接下载

- <https://pdos.csail.mit.edu/6.828/2016/xv6/book-rev9.pdf>
- <https://pdos.csail.mit.edu/6.828/2016/xv6/xv6-rev9.pdf>



深圳大學  
Shenzhen University

# 环境搭建

- 首先，你需要一个Linux系统！最简单就是使用Ubuntu。如果不奉献一台机给Linux，那就使用虚拟机吧。（Virtualbox是免费的首选）
  -
- 参考这个文档：
  - <https://pdos.csail.mit.edu/6.828/2017/tools.html>
  - % sudo apt-get install -y build-essential gdb
  - % sudo apt-get install gcc-multilib



# 环境搭建

- 为什么一定要使用**Unix / Linux**家族的操作系统作为实验环境?
  - **Xv6**是一个**Unix**操作系统。
  - 编译**xv6**时，中间文件必须是**ELF**格式，而在**windows**系统下编译，产生的中间文件格式是**PE**格式，两者并不兼容。
  - **Windows**系统并不适于操作系统教学。



# 实验环境搭建——QEMU虚拟机

- 最简单的， 使用apt-get安装qemu。
- \*注意\* 6.828网站上提到：
  - Unfortunately, QEMU's debugging facilities, while powerful, are somewhat immature, so we highly recommend you use **\*our patched version\*** of QEMU instead of the stock version that may come with your distribution.
  - <https://pdos.csail.mit.edu/6.828/2017/tools.html> 最下方。



深圳大学  
Shenzhen University

# 下载并编译xv6!

- 下载（知道git是什么吗？）
  - \$git clone git://github.com/mit-pdos/xv6-public.git
- 2. 回退到rev7版本。
  - \$cd xv6-public
  - \$git checkout -b xv6-rev7 xv6-rev7
- 3. 打开**Makefile**文件。找到如下这一行，并添加qemu的可执行程序):
  - QEMU = qemu-system-x86\_64
- 4. 开始编译！
  - \$make
- 如果看到以下界面，恭喜你，成功了！



深圳大學  
Shenzhen University

# 成功的界面显示

```
./mkfs fs.img README _cat _echo _forktest _grep _init _kill _ln _ls _mkdir _rm _sh _stressfs _usertests _wc _zombie
used 29 (bit 1 ninode 26) free 29 log 10 total 1024
balloc: first 400 blocks have been allocated
balloc: write bitmap block at sector 28
dd if=/dev/zero of=xv6.img count=10000
10000+0 records in
10000+0 records out
5120000 bytes (5.1 MB) copied, 0.0077632 s, 660 MB/s
dd if=bootblock of=xv6.img conv=notrunc
1+0 records in
1+0 records out
512 bytes (512 B) copied, 6.5885e-05 s, 7.8 MB/s
dd if=kernel of=xv6.img seek=1 conv=notrunc
248+1 records in
248+1 records out
127268 bytes (127 kB) copied, 0.000254508 s, 500 MB/s
rm wc.o grep.o mkdir.o rm.o ln.o stressfs.o kill.o echo.o init.o usertests.o zombie.o cat.o sh.o ls.o
```



# 怎么运行呢？

- \$make qemu-nox
- 这就是你编译出来的xv6操作系统。**(C+a 再 x 退出)**

```
xv6...
cpul: starting
cpu0: starting
init: starting sh
$ ls
.
..
README      2 2 1929
cat          2 3 9684
echo         2 4 9217
forktest     2 5 5993
grep         2 6 10832
init         2 7 9538
kill         2 8 9205
ln           2 9 9199
ls           2 10 10779
mkdir        2 11 9278
rm           2 12 9267
sh           2 13 16563
stressfs    2 14 9753
usertests   2 15 37576
wc           2 16 10022
zombie      2 17 8991
console     3 18 0
$
```

