



Operating Systems Tutorial 2:Pintos Compilation and Installation

Nguyen Trung Nam (tutor)

Hanoi University of Science and Technology

Compiled with reference to other presentations



Guide







Download Qemu Emulator Software & Pintos Source Code

Install Pintos and Run Examble Program



Introduction



- Qemu is a free and open-source emulator and virtualizer that can perform hardware virtualization.
- Pintos is a simple operating system framework for the 80x86 architecture.
- Pintos support kernel threads, loading and running user programs and a file system.



Requirement



- Ubuntu 18.04 or 20.04 LTS.
- 5GB ROM.
- 2 4GB RAM.



Download



- 1. Download Qemu Emulator : sudo apt-get install qemu-system
- 2. Download latest Pintos source code from: https://github.com/WyldeCat/pintos-anon

Note: Do not download the old version.





- 3. Unzip Pintos .tar.gz
- 4. Open file /utils/pintos-gdb with Vim.
- Edit **GDBMACROS** variable, assign your own Pintos full path to /src/misc/gdb-macros.
- 5.In /utils/, open the **Makefile** with Vim and **LOADLIBES** variable name is edited as **LDLIBS**.





6. Open terminal in /src/utils and type "make" to compile utils.

For error "stropts.h not found" (e.g. Ubuntu 20.04 LTS), see https://github.com/huawenyu/pintos to correct codes. Specifically, you must remove #include <stropts.h> in squish-pty.c and squish-unix.c, and statements that call isastream

7. Edit /src/threads/Make.vars

Line 7 : change **bochs** for **qemu**.

8.In /src/threads, type "make" in terminal to compile the thread directory.





9. Edit /utils/pintos

Line 103: Replace bochs with qemu.

Line 257: Replace kernel.bin with the full path

(...src/threads/build/kernel.bin).

Line 621: Replace qemu-system-i386 with qemu-system-x86_64

10. Edit /utils/Pintos.pm

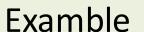
Line 362: Replace loader.bin for the full path

(...src/threads/build/loader.bin)





- 11. Edit ~/.bashrc in /home/ directory
 Add "export PATH=/home/.../pintos/src/utils:\$PATH" at the last line.
- 12. Type "source ~/.bashrc" in the terminal.
- 13. Run "pintos run alarm-multiple"







Turn off or Reboot Emulator:

- Edit /src/threads/init.c :
 Line 136 : edit "shutdown()" to "shutdown_power_off()"
- In directory /src/threads, type "make" to recompile the thread directory.
- Run "pintos run alarm-multiple" .