OS Kernels

Today

- Computer Boot Sequence
- HW09: xv6 system call
  - You'll be implementing a system call in an OS kernel.
  - Due next Wednesday – so you only get 6 days.
- The xv6 operating system
  - This is what you'll be modifying

Boot Sequence

For any computer, we've got the same general boot sequence:

- Power on
- Hardware runs firmware
- Firmware runs bootloader
- Bootloader runs OS Kernel
- OS Kernel starts userspace

Traditional PC Booting Linux

- Power On
- Hardware loads BIOS (firmware)
- BIOS loads Grub (bootloader)
  - Scans disks for “bootable” one.
  - Has configurable boot order
- Grub loads Linux (OS kernel)
- Linux runs PID 1 (“init”), which runs the rest of userspace programs.

Modern Cellphone Boot

- Power On
- Hardware loads firmware, checks signature
- Firmware loads bootloader, checks signature
- Bootloader loads OS, checks signature

- If signature check fails, soft-brick.
- Who’s signature? (...)

Modern PC Boot

- Power on of ME CPU, loads ME OS
- ME OS boots main CPU
- Main CPU loads UEFI (firmware), checks signature
- UEFI loads bootloader, checks signature
- Bootloader selects and loads OS, checks signature

- If signature check fails, soft-brick.
- Who’s signature? (...)

Who’s signature?

- Modern systems verify signatures at several stages during the boot process, and stop if the verification fails.
- Who’s signature?
  - The manufacturer, the OS vendor
- Why?
  - Conceptually could prevent an early rootkit or unauthorized USB boot.
  - More common case is simply to prevent unauthorized OS installs / modifications. See “iPhone Jailbreak”.
  - It’s not entirely clear that this is actually a security measure for the benefit of the user of the machine.

Management Engine

- Intel processors initiate booting not through the main CPU, but through a secondary processor running user-inaccessible code.
- This code then starts the main CPU.
- This guarantees that admins of large deployments can do remote administration of machines regardless of their software state.
- But… not all machines want to have a pre-software remote access feature.
- AMD and some ARM systems have similar features.

Bootloader

- The bootloader selects the operating system to run, loads the initial kernel stage from disk, and then jumps to executing kernel code.
- Amusingly, we start in 16 bit kernel code...

xv6 background

- The instructors at MIT wanted an OS to teach a systems course
- They thought that existing teaching OSes weren't "real" enough
- But "real" OSes are too complicated.
- So they ported version 6 UNIX – from 1975, for the PDP 11 – to Intel x86.
- So this is real, and still applicable, but not modern.