



## Course Information and Policy

---

Yajin Zhou (<http://yajin.org>)

Zhejiang University



# Instructor

---

- Yajin Zhou (周亚金)
  - A Zhejiang University 100 Young Professor
  - A system security researcher, build and hack system
  - Published in all top 4: IEEE S&P, ACM CCS, USENIX Sec, NDSS
  - Research
    - Program analysis (source code/binary): how to find vulnerabilities
    - System security: how to make the system more secure
    - Hardware-software codesign: how to make the system more secure through hardware support?



# Course Information

---

- Office: 曹楼 412
- Office hour: Saturday 14:05 – 15:40
- Class website: <https://yajin.org/os2019fall/>
- TAs:
  - Jiaqi Li (教9 211)
  - Qiang Li (教9 211)
  - Dingding Wang (教9 211)
  - Pengfei Li (科工楼)



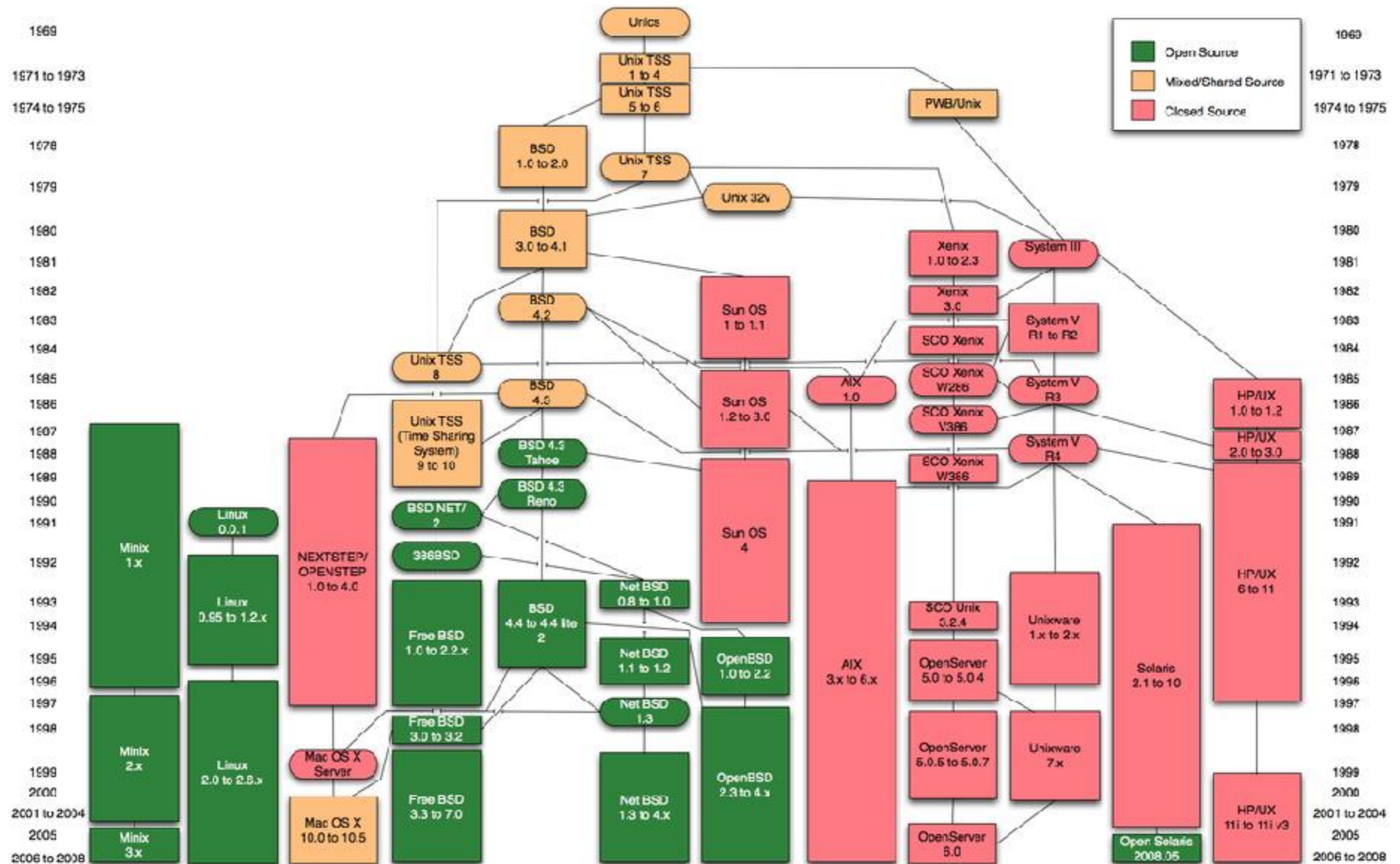
# Popular Operating Systems

---





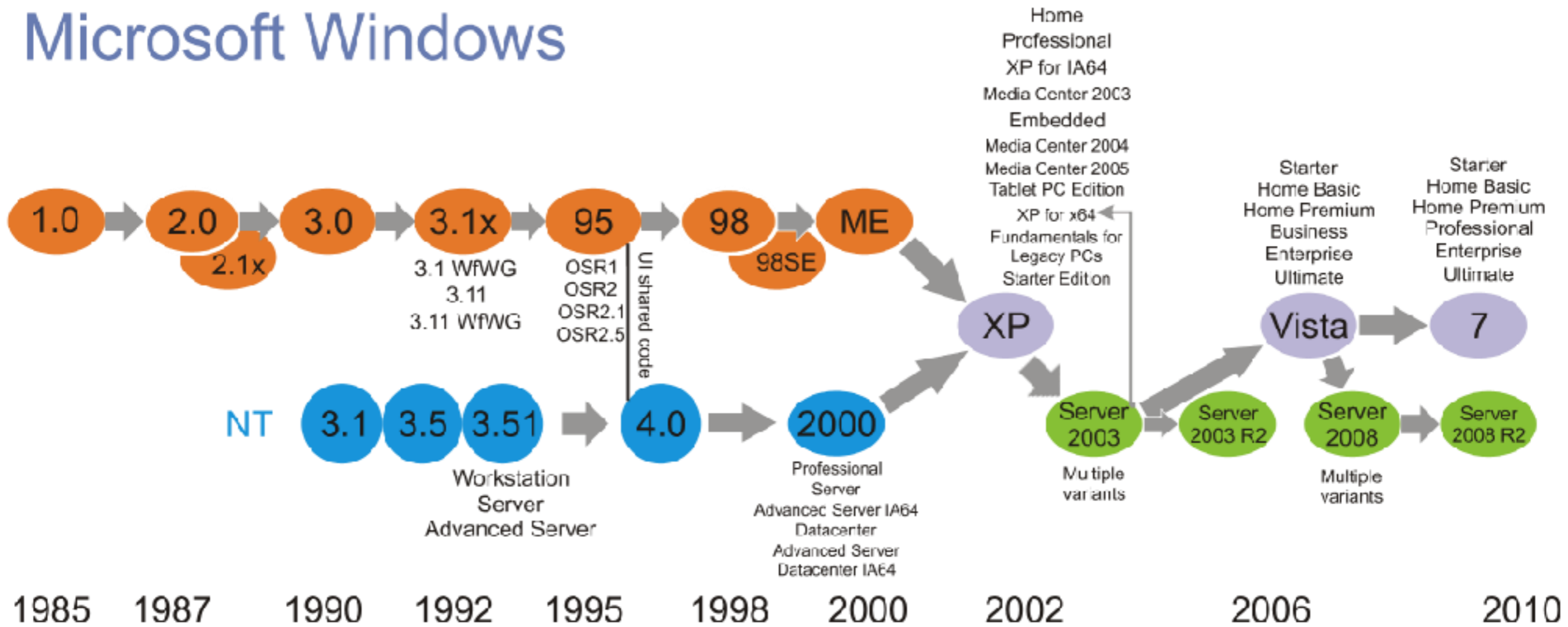
# UNIX Family Tree





# Windows Family Tree

## Microsoft Windows





# Why Study Operating Systems

- OS is highly complicated software running on most machines
  - Windows: 50M lines of source code
  - Linux: 15M lines of source code
- It contains many important system concepts
  - complexity hiding, performance tuning, resource allocation...
- Studying OS internals makes you a more **capable** programmer
  - know how it works, and how it works better



You need to restart your computer. Hold down the Power button until it turns off, then press the Power button again.

Redémarrez l'ordinateur. Enfoncez le bouton de démarrage jusqu'à l'extinction, puis appuyez dessus une nouvelle fois.

Debe reiniciar el ordenador. Mantenga pulsado el botón de arranque hasta que se apague y luego vuelva a pulsarlo.

Sie müssen den Computer neu starten. Halten Sie den Ein-/Ausschalter gedrückt bis das Gerät ausgeschaltet ist und drücken Sie ihn dann erneut.

コンピュータの再起動が必要です。電源が切れるまでパワーボタンを押し続けてから、もう一度パワーボタンを押します。



# Why Study Operating Systems

---

- OS concepts benefit whole life
  - OS concepts are re-usable when implementing other software
  - Lessons learned from OS study can be applied to complex software systems, such as map-reduce, DNS
- Foundation of ALL software
  - Better user-space software, including apps
    - Invoke proper kernel API: performance
    - What can and cannot be done: security/reliability
  - Better performance
    - Caching (memory cache/TLB)



# Why Study Operating Systems

---

- For Hacking
- The more you know OS, the better hacker you are
- Because the thing you are trying to hack into, probably is running an OS





# Why Study Operating Systems

- For Profit
  - Interview = coding + system design
  - Build your own company
    - Great system -> great product -> great company



Google

Google MapReduce

Google File System

Google Cloud

Fuchsia  
by Google



Wear OS by Google





# Learning Objectives

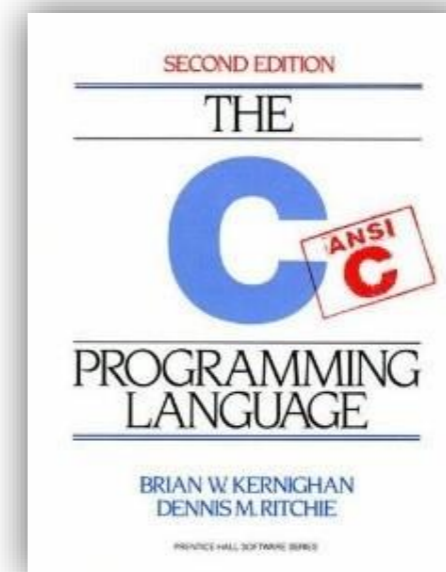
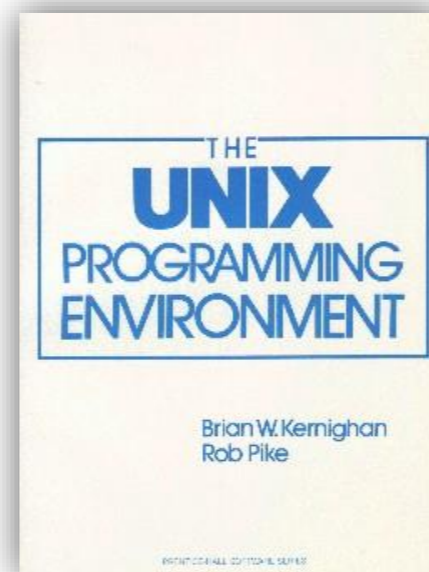
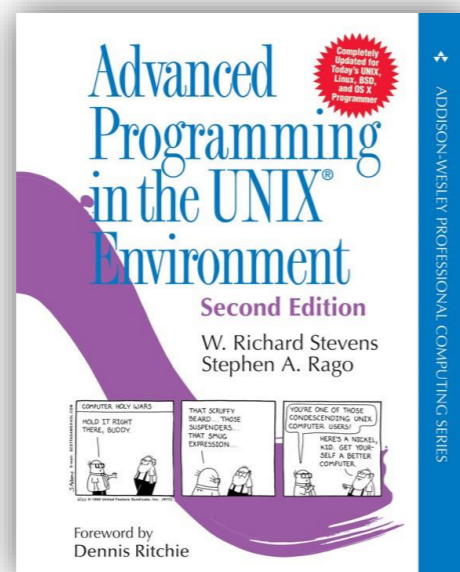
---

- Understand operating system concepts
  - process management, CPU scheduling, synchronization, file systems...
- Comprehend OS concepts through programming
  - multi-threading and synchronization, system call, kernel modules...
- Get ~~an overall~~ a deep understanding of how the real-world operating systems work
  - You can never truly understand a concept unless you **implemented (CODE) it**



# Prerequisites

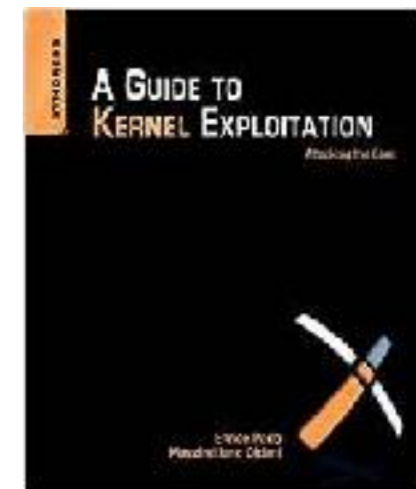
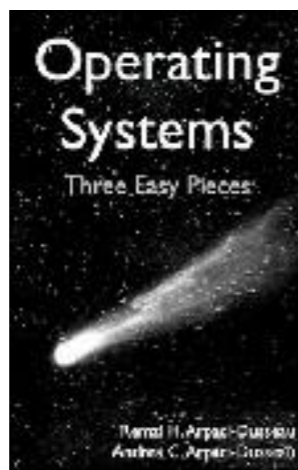
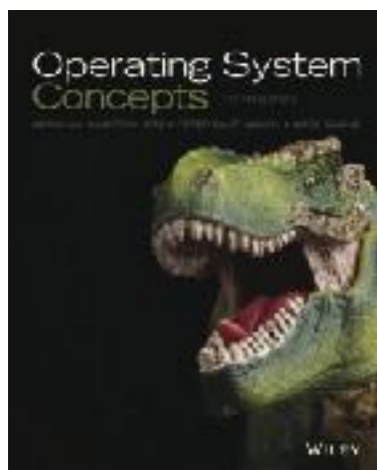
- Data Structures
- Programming skills:
  - proficiency in UNIX(Linux) programming and debugging
  - proficiency in the C programming language





# Course Material

- Lecture notes (posted at the class website)
- Textbook:
  - Operating System Concepts
  - Operating Systems: Three Easy Pieces, v1.0





# Grading

---

- Final Exam – 50%
- Homework – 10%
- Survey/Presentation – 12%
- Class Quiz – 7%
- Project – 21%
- **No bonus!**

## **Advanced Track**

- Final Exam – 50%
- Homework – 10%
- Project – 40%+ **bonus**



# Homework

---

- Homework assignments are **individual** efforts
- Submission **MUST** be typed, no hand-written submission
- Late submissions are accepted **after the deadline**
  - a **10%** penalty will be applied for **each day** of late submission
- Disputes of grade **MUST** be resolved within **one week** of receiving it



# Exams

---

- No midterm exam
- One final exam, close-booked
- Final exam is comprehensive/cumulative





# Your Responsibilities

---

- Understand lecture & reading materials
- Ask for extra help, if needed
  - if the class is too hard or you do not have necessary backgrounds
- Uphold academic integrity
- Turn in your assignments on time
- Check class web page regularly



# Dos and Don'ts

---

- Do share debugging experiences, knowledge of tools
- Do acknowledge help from others
- Do acknowledge sources of information from books and web pages
  
- Don't cheat or help others cheat
- Don't paraphrase code from others
  - e.g., changing variable names or indentation
- Don't post code to the discussion board