# Course Introduction

Linux System Administration
Fall 2025

# Course Goals

- To help students gain an understanding of how to manage a UNIX or Linux environment, and provide a marketable set of technical skills.
  - Installing and configuring UNIX/Linux Systems
  - Routine maintenance
  - Security
  - **Troubleshooting**

# Course Objectives

- Demonstrate proficiency with the Linux command line, filesystem, and core components of UNIX
- Effectively operate a Linux system inside of a network environment
- Design and deploy Linux systems to accomplish tasks such as: serving web content, sharing files, firewalling, etc.

# Grading

| Assignment | Fraction |
|---|---|
| Labs | 60% |
| Final | 30% |
| Participation | 10% |

- Some assignments will have extra credit
- Assignments have extra tasks for grad students

| Letter | Numerical |
|---|---|
| A | 95-100 |
| A- | 90-94.99 |
| B+ | 85-89.99 |
| B | 80-84.99 |
| B- | 75-79.99 |
| C+ | 70-74.99 |
| C | 65-69.99 |
| C- | 60-64.99 |
| D+ | 55-59.99 |
| D | 50-54.99 |
| D- | 40-49.99 |
| F | 0-39.99 |

# Class Attendance

- **Attendance is expected**
  - Discuss solutions to the labs
  - Very useful for the final exam
- I will record lecture on an ad hoc basis if you there is a compelling reason
  - Email me by noon

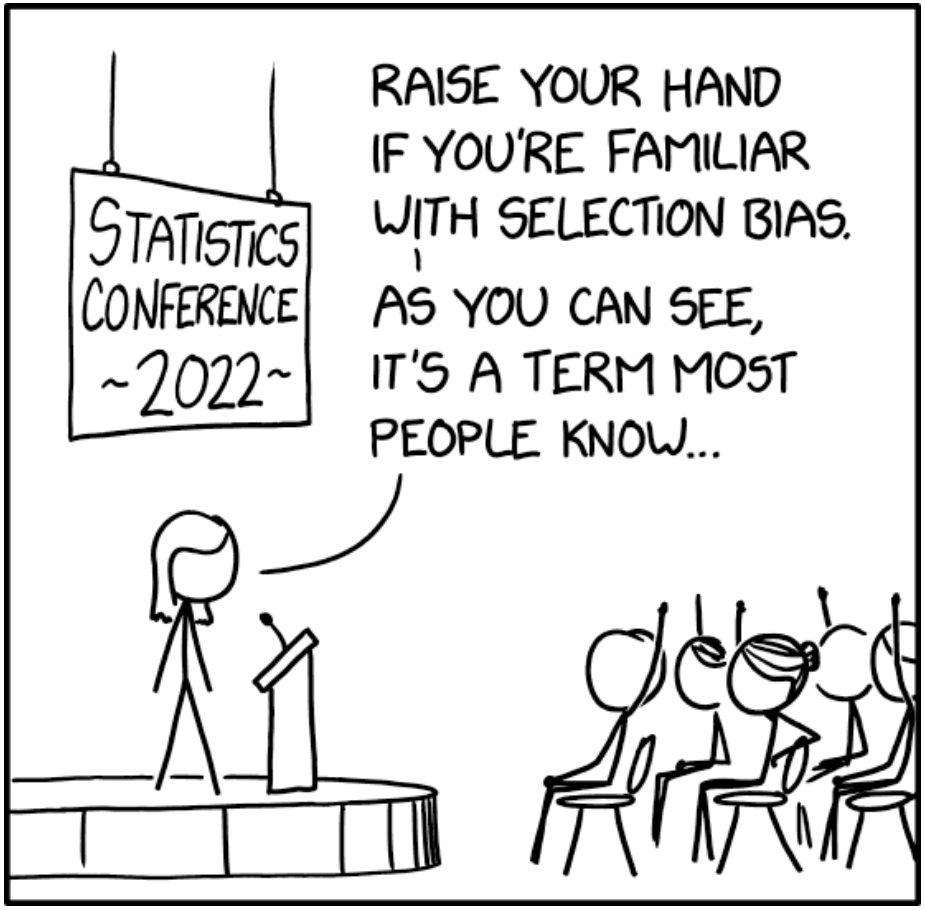
# Field Trip

- Each semester we go to touch some hardware
- Usually Wyoming Supercomputer Center

# My challenge

- System administration cannot be learned in a linear fashion
  - Some things may seem repetitive
- I discover commands all the time
- *TIMTOWTDI*

(You should read the jokes on the slides)



# Final Exam

- The final is a practical exam
  - Open access to textbook, notes and internet, but NO help from other people or ChatGPT
- Timed task
  - Recover hung machines or services
  - Very difficult unless you do the labs yourself
  - Learn to troubleshoot when you do the labs

# Deadlines

- Most labs are due at 23:59 on Monday.
- No penalty grace period until 08:00 on Tuesday
  - *If you leave this until Monday evening, it will typically not turn out well*
- Extensions can be granted if requested **before** the deadline, but cannot be later than the start of the next class period

# Grading

- Grading is done using a grading script that verifies that all the requirements are satisfied
  - The script is ***very*** picky about exact naming of scripts, user and group names, directories, etc.
  - Watch for adding extensions (e.g. .py or .sh - that's a Windows thing - Linux uses a `#!` line)
- *An important part of the system administration is to develop your own test and verification scripts*

# Textbook

- **Unix and Linux System Administration Handbook 5ed**
  - Nemeth, Snyder, Hein, Mackin & Whaley
  - ISBN-13: 978-0134277554
- Required for readings.

# About Evi

- Evi Nemeth
  - 6/7/1940 – 6/?/2013
  - CU Boulder 1980-2001
- Lead author of textbook
- Pioneer in establishing system administration as a field of study



# Contact Information

- Willem A (Vlakkies) Schreüder
  - vlakkies@colorado.edu
  - ECOT 732
  - Piazza
- Office Hours
  - Monday 3-4pm by Zoom
  - Thursday 2-3pm in person or by Zoom
  - Email me to meet by Zoom or in person at other times

# System Administration Principles

- As shown when using sudo the first time:
  - Respect the privacy of others.
  - Think before you type.
  - With great power comes great responsibility.
- Our Motto:  D.F.I.U.
  - Don’t Mess It Up



# Don’t be this guy

- Never make changes shortly before leaving
  - *Do not try to do the homework an hour before the deadline*
- Reboot the system after touching critical configs



# System administration tasks

- Design new systems
- Provision hardware
- Add/remove hardware
- Install/upgrade software
- Provision users
- Perform backups
- Monitor systems
- Document systems
- Monitor security
- Support operations

# UNIX Principles

- *UNIX is simple. It just takes a genius to understand its simplicity* -Dennis Ritchie
- Everything is a file
  - Files are streams of bytes
  - Programs consume and produce text streams
  - Commands are executable files
    - #! sets interpreter for scripts

# UNIX Philosophy (Doug McIlroy)

- Write programs that do one thing and do it well.
- Write programs to work together.
- Write programs to handle text streams, because that is a universal interface.

# A brief history of Linux

- 1964 MULTICS (MIT, GE, AT&T/Bell Labs)
- 1969 Unics/UNIX (Bell labs)
- 1983 GNU (Richard M. Stallman)
- 1987 MINIX (Tanenbaum/Vrije Universiteit)
- 1991 Linux (Linus Torvalds)

# MULTICS (MIT,GE,AT&T/Bell Labs)

**Mult**iplexed **I**nformation and **C**omputing **S**ervice

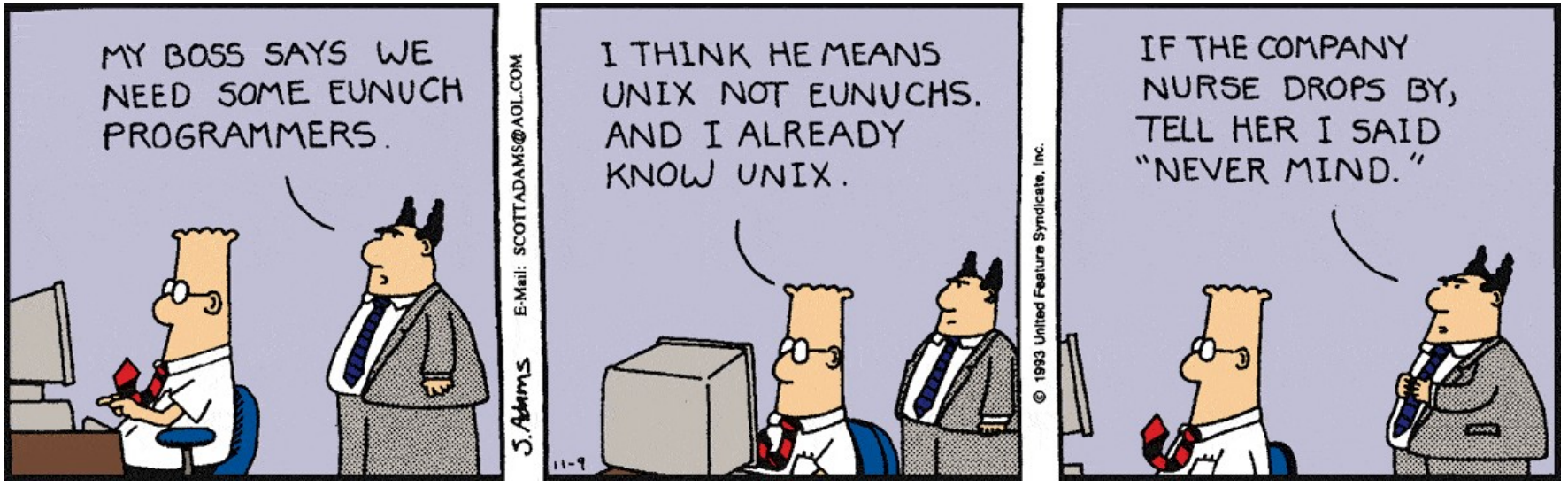


- Unics (Bell labs)

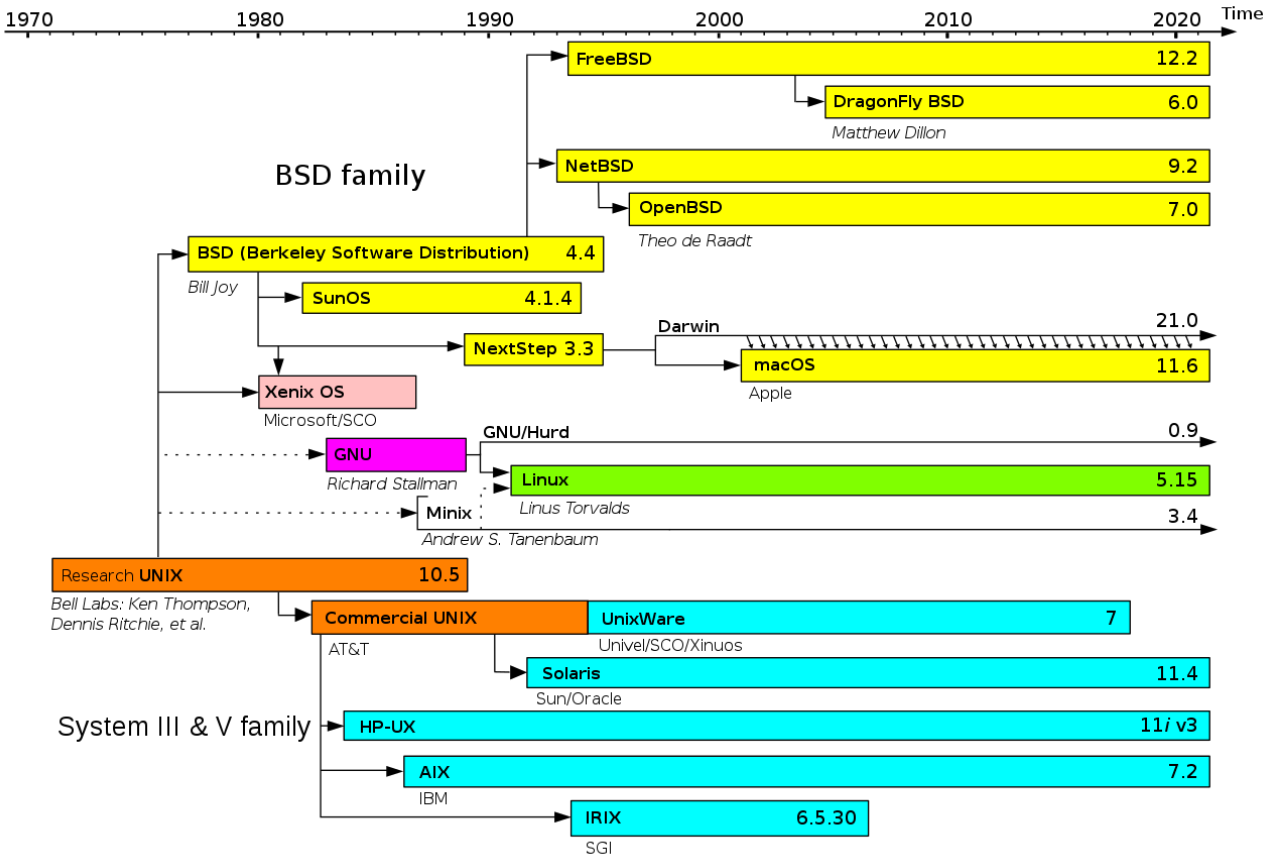
**Uni**plexed **I**nformation and **C**omputing **S**ervice



# Unics, UNIX, Unix or *nix?



# UNIX family


1970
1980
1990
2000
2010
2020
Time
BSD family
FreeBSD 12.2
DragonFly BSD 6.0
Matthew Dillon
NetBSD 9.2
OpenBSD 7.0
Theo de Raadt
BSD (Berkeley Software Distribution) 4.4
Bill Joy
SunOS 4.1.4
NextStep 3.3
Darwin 21.0
macOS 11.6
Apple
Xenix OS
Microsoft/SCO
GNU/Hurd 0.9
GNU
Richard Stallman
Linux 5.15
Linus Torvalds
Minix 3.4
Andrew S. Tanenbaum
Research UNIX 10.5
Bell Labs: Ken Thompson, Dennis Ritchie, et al.
Commercial UNIX
AT&T
UnixWare 7
Univel/SCO/Xinuos
Solaris 11.4
Sun/Oracle
System III & V family
HP-UX 11i v3
AIX 7.2
IBM
IRIX 6.5.30
SGI

# GNU (**G**nu is **N**ot **U**nix)

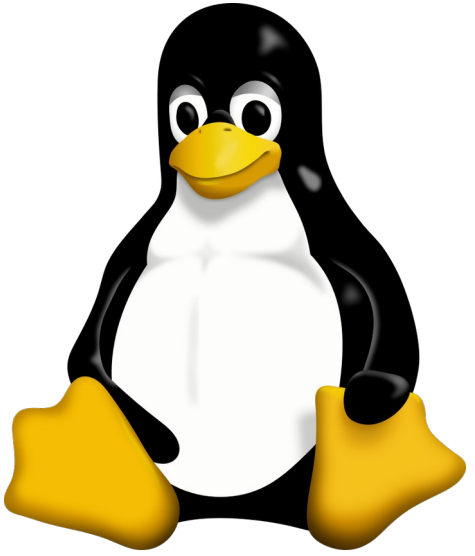


- Richard M Stallman (RMS) announced plans to create an **entirely free** UNIX like operating system in 1983
  - Free as in speech, not free as in beer
  - GNU Public License (GPL)
- Most of the core components, except for the kernel, were completed by 1987 (GNU hurd)
  - gcc, g++ and many UNIX utilities often used on commercial UNIX platforms like HP-UX and SunOS

# MINIX (mini-Unix)

- Microkernel implemention on IBM PC
  - call compatible with UNIX V7
  - Only about 12,000 lines of C
- Used to teach operating systems
  - Operating Systems: Design and Implementation
    Andrew S Tanenbaum, Vrije Universiteit
- Not free, but source code available
- Inspiration for Linux

# And then there was Linux

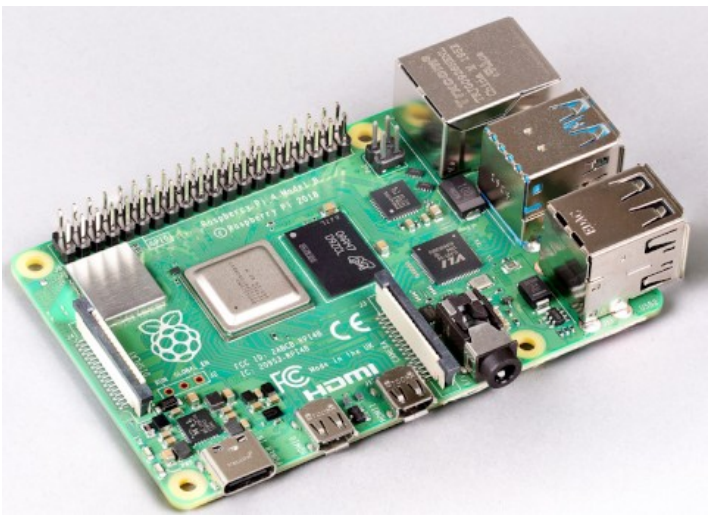


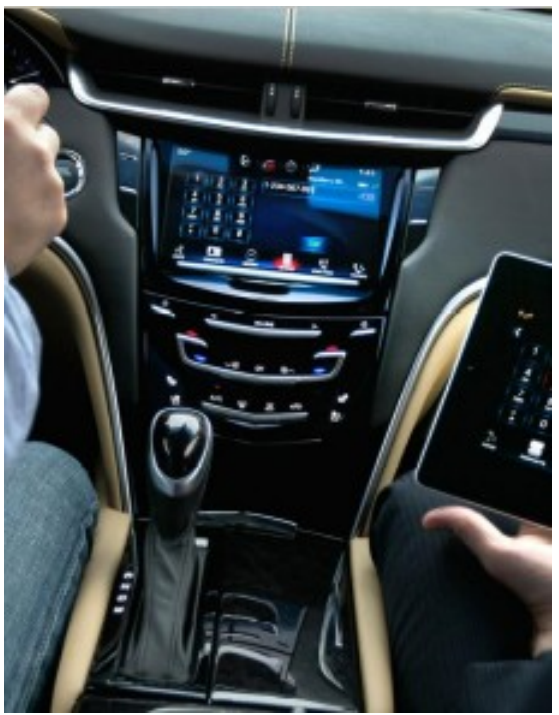
- Linus Torvalds (1991)
  - New kernel released under GPL
  - GNU utilities to complete OS (GNU/Linux)
- Quickly adopted from desktop to Big Iron to IoT
  - Distributions provide installer and package manager
  - Initially CDs, Internet make downloads possible
- Desktop matured with KDE & GNOME (late 90s)
  - Android largest installed base of any OS




LINKSYS


Featured
Movies
TV Shows

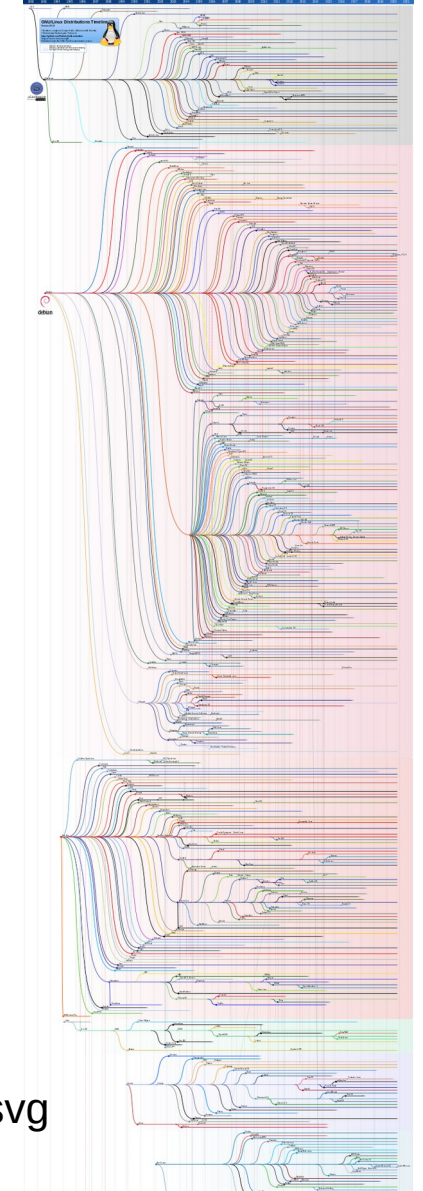

HDMI





# Linux Distributions

- Slackware
  - SuSE
- Debian
  - Ubuntu
- Red Hat
  - CentOS
  - Rocky
  - Alma



https://upload.wikimedia.org/wikipedia/commons/1/1b/Linux_Distribution_Timeline.svg

# Choosing a Distibution

- debian
  - Stresses free software
  - Base for Ubuntu, Raspbian, etc
  - Upgrades across major revisions
  - Fairly quick updating of packages to latest
- RedHat
  - Stresses stability
  - Commercial product
    - CentOS, Rocky, Alma are free versions
  - Major revision upgrades are often new installs
  - Not bleeding edge

# Choosing a Distibution

- SuSE
  - Popular in Europe
  - SuSE - Commercial with support
  - OpenSuSE - free version
  - Open SuSE MicroOS - transactional updates
    - Geared to edge computing

# Berkeley Software Distribution

- Early derivative of Research Unix
  - University of California Berkeley
- Pioneered networking
- Permissive license
  - Basis for commercial products
  - SunOS, macOS
- FreeBSD
  - Stresses ports
- OpenBSD
  - Stresses security
- NetBSD
  - Stresses hardware support
- Many others

# Lab 1

- Access the CS vSphere machines
- Change the root password on all machines
- Learn how to ssh and scp from your local machine (laptop or desktop)