# Introduction to Operating Systems, Part II

**CSD2180 Operating Systems**
BSc in Computer Science (IMGD / RTIS)
Singapore Institute of Technology / DigiPen Institute of Technology
September 2021

Adapted from Operating Systems Concepts - 10th Edition; Silberschatz, Galvin, and Gagne. ©2018

# Attendance Taking

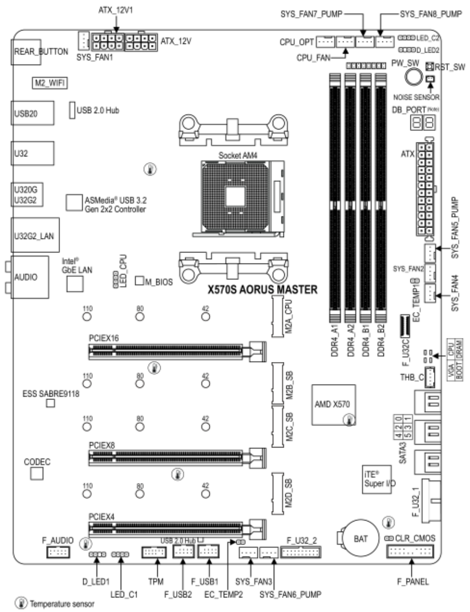
**https://forms.office.com/r/t6E6nkGXuA**

- Log in to your SIT account to submit the form
- You can only submit the form once
- The codeword is **durian**

# Operating System Operations

# Different Perspectives of the Same Computer Motherboard

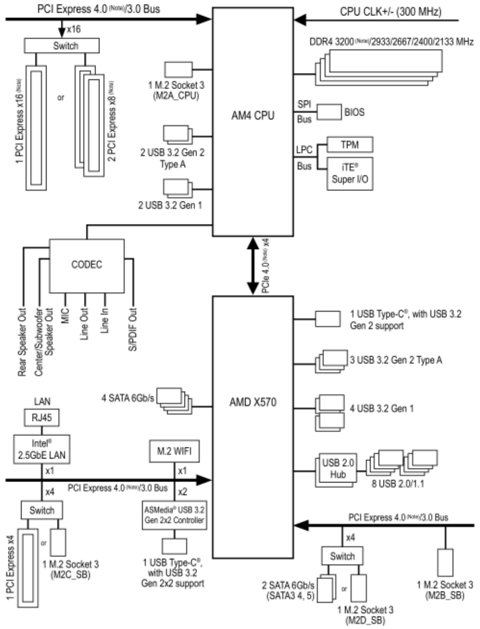
X570S AORUS MASTER Motherboard Layout



(Note) For debug code information, please refer to Chapter 6.

- 6 -

X570S AORUS MASTER Motherboard Block Diagram



(Note) Actual support may vary by CPU.

- 7 -

# How a Computer Boots Up

## Hardware powered on

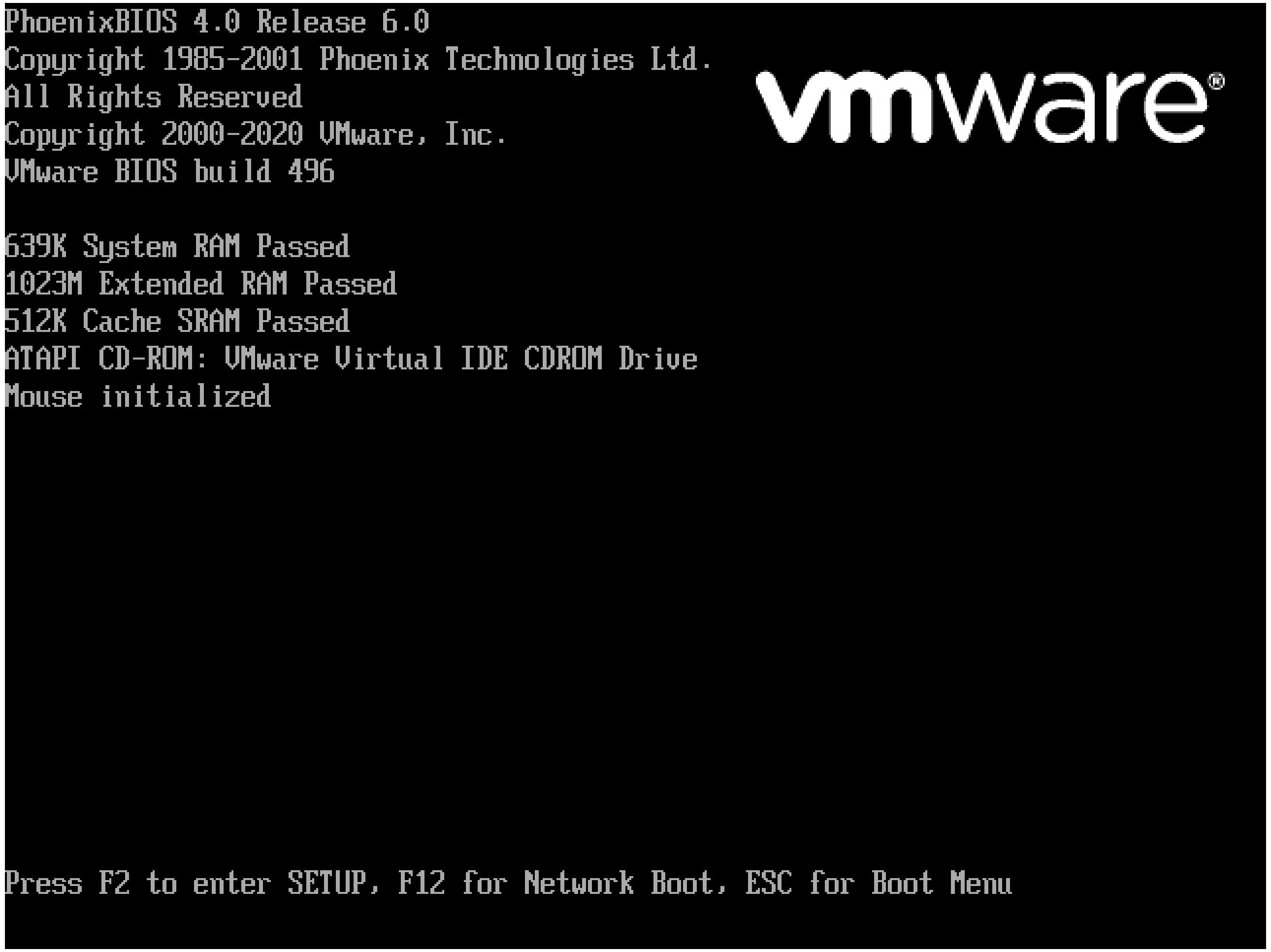
Press this button

# How a Computer Boots Up

**Hardware powered on**

**BIOS runs**

- Performs power-on self-test (POST)
- Initialize peripherals
- Locates secondary storage devices for bootable devices



```
PhoenixBIOS 4.0 Release 6.0
Copyright 1985-2001 Phoenix Technologies Ltd.
All Rights Reserved
Copyright 2000-2020 VMware, Inc.
VMware BIOS build 496

639K System RAM Passed
1023M Extended RAM Passed
512K Cache SRAM Passed
ATAPI CD-ROM: VMware Virtual IDE CDROM Drive
Mouse initialized




Press F2 to enter SETUP, F12 for Network Boot, ESC for Boot Menu
```

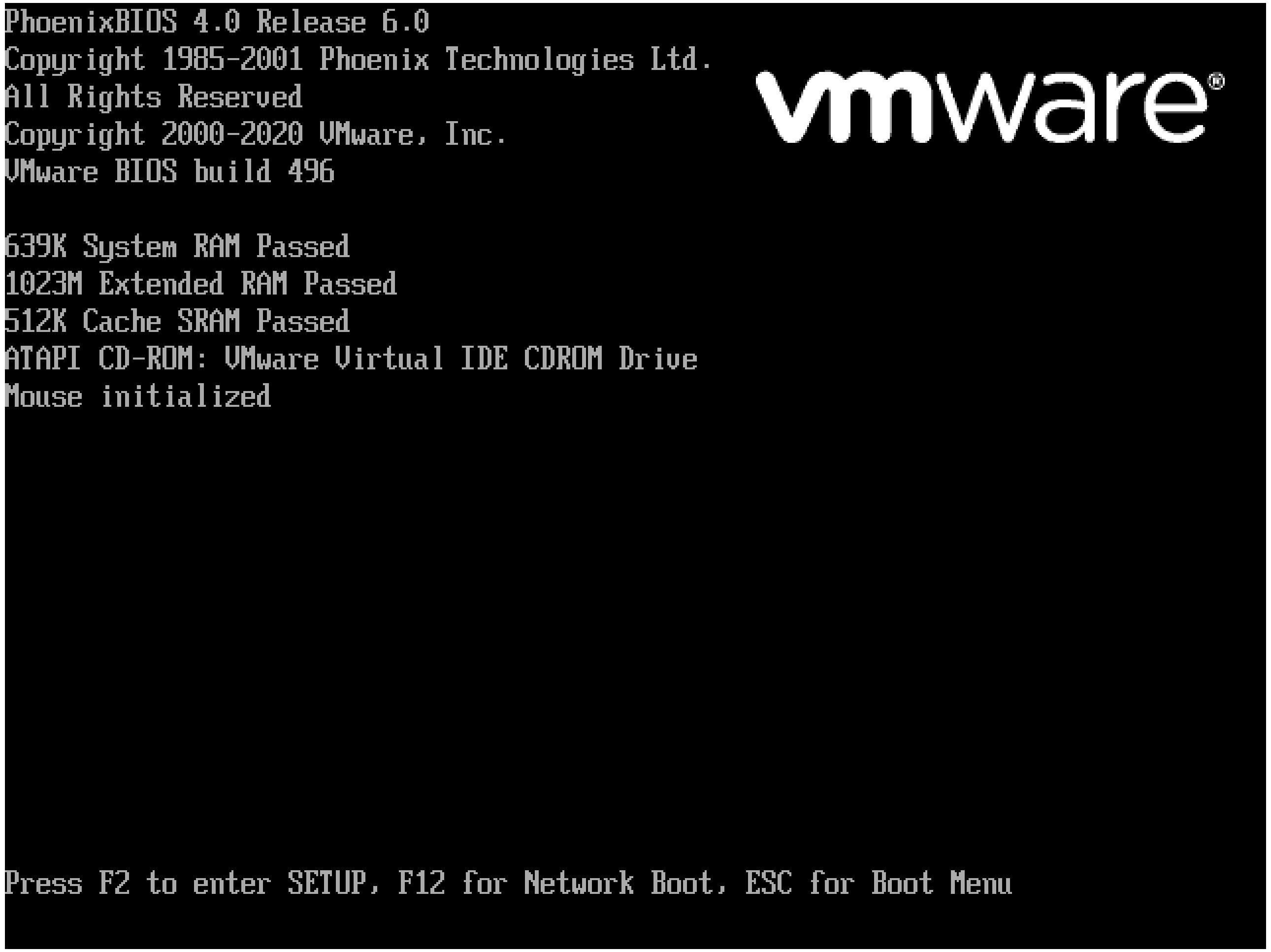
# How a Computer Boots Up

**Hardware powered on**

**BIOS runs**

**BIOS loads and runs Master Boot Record (MBR)**

- Located at first 512 bytes of first / designated boot device



```
PhoenixBIOS 4.0 Release 6.0
Copyright 1985-2001 Phoenix Technologies Ltd.
All Rights Reserved
Copyright 2000-2020 VMware, Inc.
VMware BIOS build 496

639K System RAM Passed
1023M Extended RAM Passed
512K Cache SRAM Passed
ATAPI CD-ROM: VMware Virtual IDE CDROM Drive
Mouse initialized








Press F2 to enter SETUP, F12 for Network Boot, ESC for Boot Menu
```

## How a Computer Boots Up

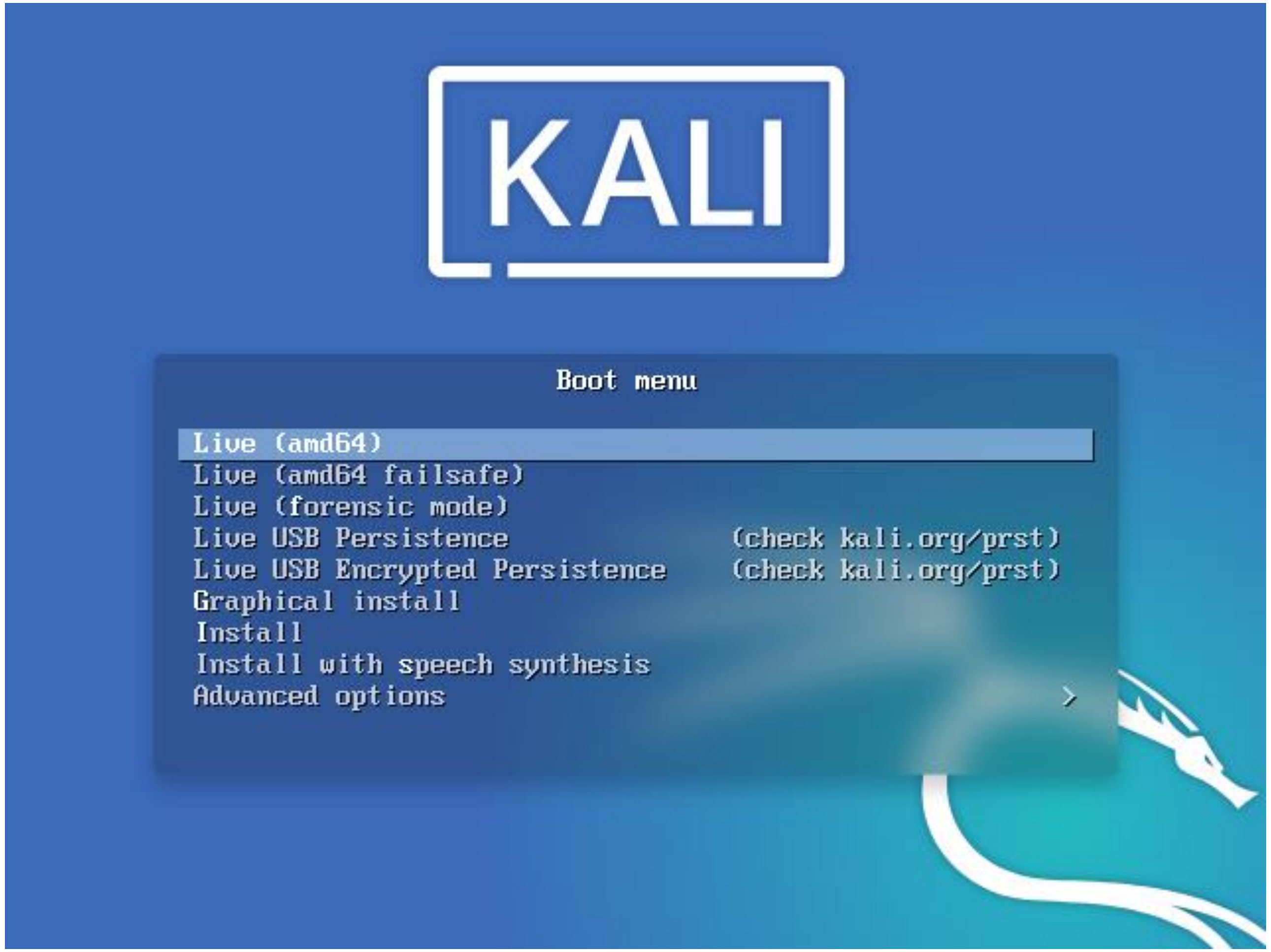
**Hardware powered on**

**BIOS runs**

**BIOS loads and runs Master Boot Record (MBR)**

**MBR loads bootloader**

- Bootloader may be loaded from e.g., partitions
- A bootloader may load subsequent bootloader stages, i.e., chain-loading



```
Boot menu

Live (amd64)
Live (amd64 failsafe)
Live (forensic mode)
Live USB Persistence                (check kali.org/prst)
Live USB Encrypted Persistence      (check kali.org/prst)
Graphical install
Install
Install with speech synthesis
Advanced options                                        >
```

## How a Computer Boots Up

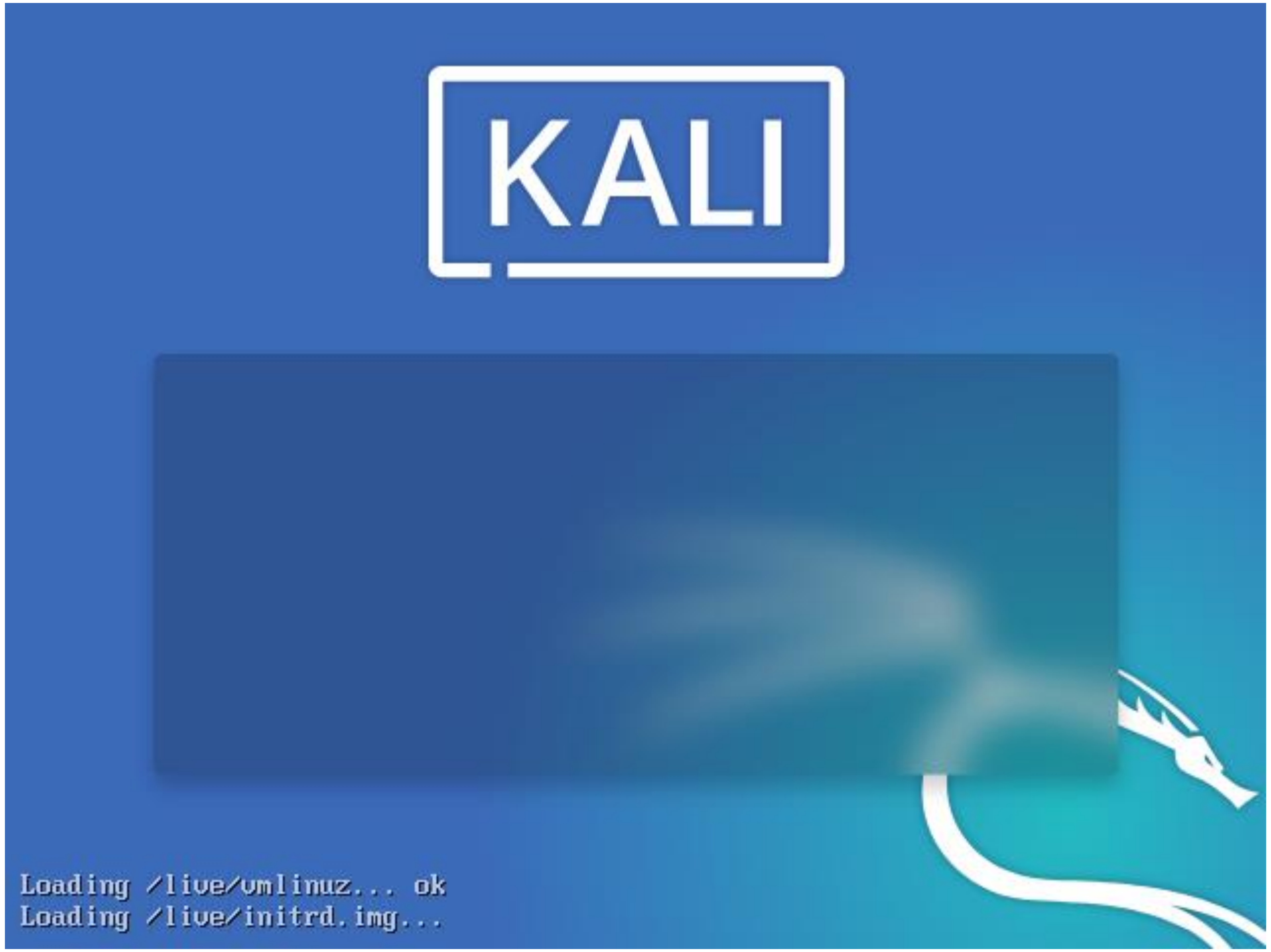
Hardware powered on

BIOS runs

BIOS loads and runs Master Boot Record (MBR)

MBR loads bootloader

Bootloader loads and runs the operating system



```
Loading /live/vmlinuz... ok
Loading /live/initrd.img...
```

# How a Computer Boots Up

Hardware powered on

BIOS runs

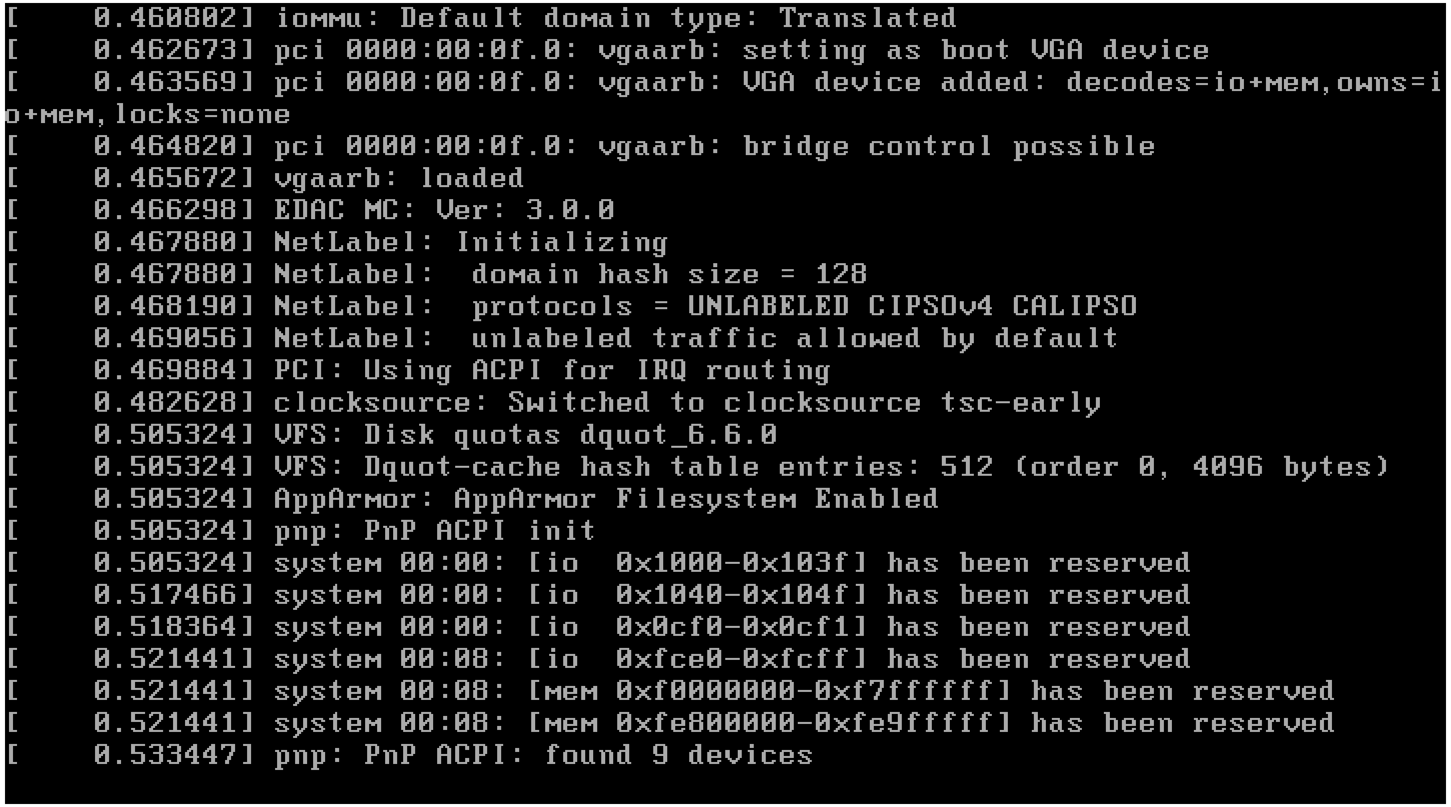
BIOS loads and runs Master Boot Record (MBR)

MBR loads bootloader

Bootloader loads and runs the operating system

Operating system starts up and runs

- ○ Kernel loads and executes
- ○ Services are started



```
[    0.460802] iommu: Default domain type: Translated
[    0.462673] pci 0000:00:0f.0: vgaarb: setting as boot VGA device
[    0.463569] pci 0000:00:0f.0: vgaarb: VGA device added: decodes=io+mem,owns=i
o+mem,locks=none
[    0.464820] pci 0000:00:0f.0: vgaarb: bridge control possible
[    0.465672] vgaarb: loaded
[    0.466298] EDAC MC: Ver: 3.0.0
[    0.467880] NetLabel: Initializing
[    0.467880] NetLabel:  domain hash size = 128
[    0.468190] NetLabel:  protocols = UNLABELED CIPSOv4 CALIPSO
[    0.469056] NetLabel:  unlabeled traffic allowed by default
[    0.469884] PCI: Using ACPI for IRQ routing
[    0.482628] clocksource: Switched to clocksource tsc-early
[    0.505324] VFS: Disk quotas dquot_6.6.0
[    0.505324] VFS: Dquot-cache hash table entries: 512 (order 0, 4096 bytes)
[    0.505324] AppArmor: AppArmor Filesystem Enabled
[    0.505324] pnp: PnP ACPI init
[    0.505324] system 00:00: [io  0x1000-0x103f] has been reserved
[    0.517466] system 00:00: [io  0x1040-0x104f] has been reserved
[    0.518364] system 00:00: [io  0x0cf0-0x0cf1] has been reserved
[    0.521441] system 00:08: [io  0xfce0-0xfcff] has been reserved
[    0.521441] system 00:08: [mem 0xf0000000-0xf7ffffff] has been reserved
[    0.521441] system 00:08: [mem 0xfe800000-0xfe9fffff] has been reserved
[    0.533447] pnp: PnP ACPI: found 9 devices
```

# How a Computer Boots Up

Hardware powered on

BIOS runs

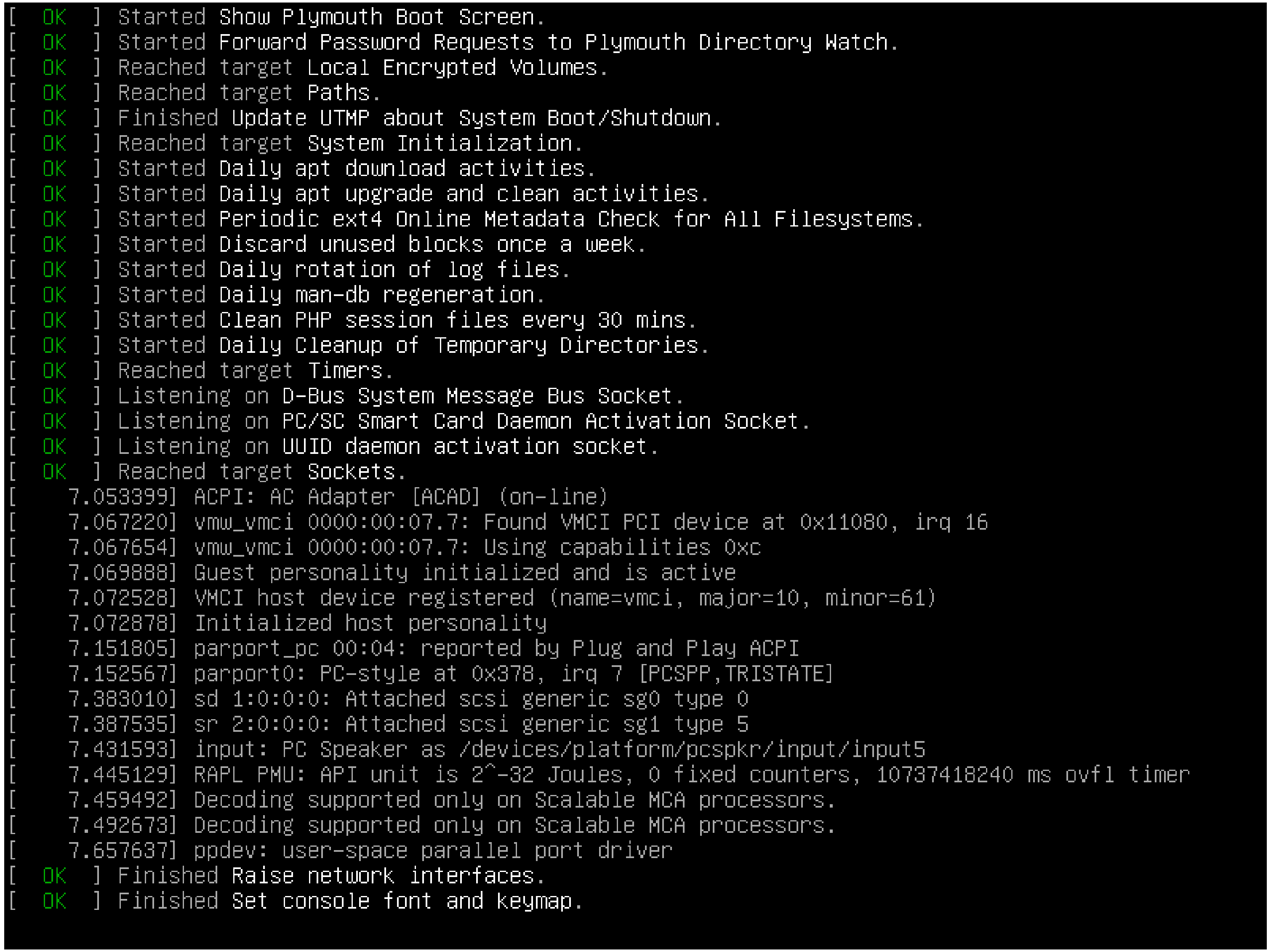
BIOS loads and runs Master Boot Record (MBR)

MBR loads bootloader

Bootloader loads and runs the operating system

Operating system starts up and runs

- ○ Kernel loads and executes
- ○ Services are started



```
[  OK  ] Started Show Plymouth Boot Screen.
[  OK  ] Started Forward Password Requests to Plymouth Directory Watch.
[  OK  ] Reached target Local Encrypted Volumes.
[  OK  ] Reached target Paths.
[  OK  ] Finished Update UTMP about System Boot/Shutdown.
[  OK  ] Reached target System Initialization.
[  OK  ] Started Daily apt download activities.
[  OK  ] Started Daily apt upgrade and clean activities.
[  OK  ] Started Periodic ext4 Online Metadata Check for All Filesystems.
[  OK  ] Started Discard unused blocks once a week.
[  OK  ] Started Daily rotation of log files.
[  OK  ] Started Daily man-db regeneration.
[  OK  ] Started Clean PHP session files every 30 mins.
[  OK  ] Started Daily Cleanup of Temporary Directories.
[  OK  ] Reached target Timers.
[  OK  ] Listening on D-Bus System Message Bus Socket.
[  OK  ] Listening on PC/SC Smart Card Daemon Activation Socket.
[  OK  ] Listening on UUID daemon activation socket.
[  OK  ] Reached target Sockets.
[    7.053399] ACPI: AC Adapter [ACAD] (on-line)
[    7.067220] vmw_vmci 0000:00:07.7: Found VMCI PCI device at 0x11080, irq 16
[    7.067654] vmw_vmci 0000:00:07.7: Using capabilities 0xc
[    7.069888] Guest personality initialized and is active
[    7.072528] VMCI host device registered (name=vmci, major=10, minor=61)
[    7.072878] Initialized host personality
[    7.151805] parport_pc 00:04: reported by Plug and Play ACPI
[    7.152567] parport0: PC-style at 0x378, irq 7 [PCSPP,TRISTATE]
[    7.383010] sd 1:0:0:0: Attached scsi generic sg0 type 0
[    7.387535] sr 2:0:0:0: Attached scsi generic sg1 type 5
[    7.431593] input: PC Speaker as /devices/platform/pcspkr/input/input5
[    7.445129] RAPL PMU: API unit is 2^-32 Joules, 0 fixed counters, 10737418240 ms ovfl timer
[    7.459492] Decoding supported only on Scalable MCA processors.
[    7.492673] Decoding supported only on Scalable MCA processors.
[    7.657637] ppdev: user-space parallel port driver
[  OK  ] Finished Raise network interfaces.
[  OK  ] Finished Set console font and keymap.
```

# How a Computer Boots Up

- Hardware powered on
- BIOS runs
- BIOS loads and runs Master Boot Record (MBR)
- MBR loads bootloader
- Bootloader loads and runs the operating system
- Operating system starts up and runs

# Multiprogramming and Multitasking

Problem

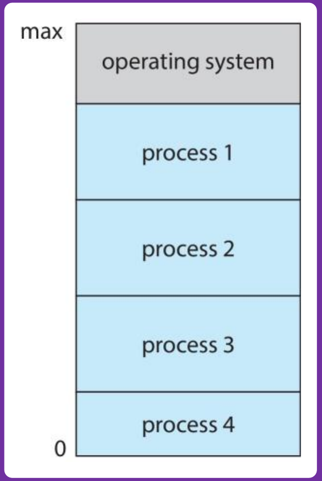
- A single program cannot always keep the CPU or I/O devices busy
- Users want to run more than one program at a time

**Multiprogramming** is needed for efficiency, by organizing programs so that the CPU always has one to execute

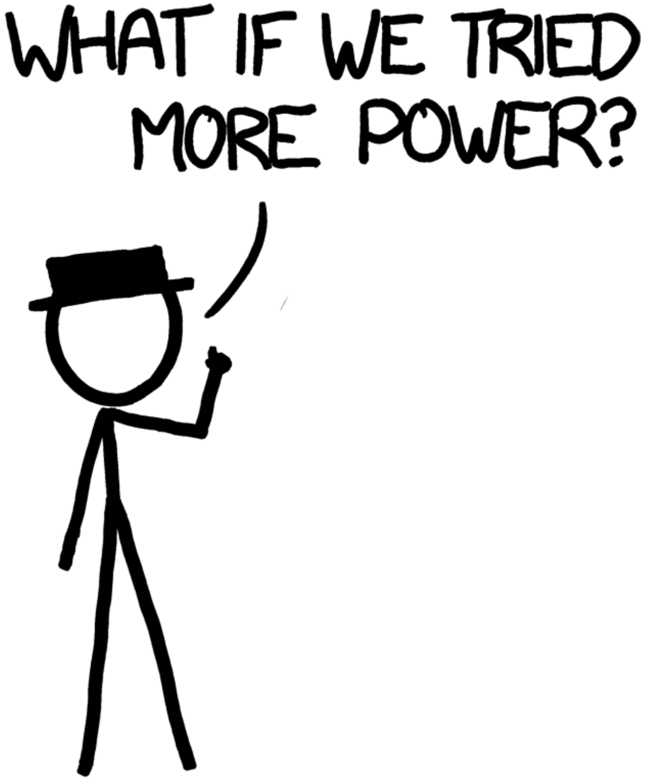
General idea

- The OS keeps several processes in memory
- One process is selected and run via **process scheduling**
- When the process has to wait (e.g., for I/O), the OS switches to another process

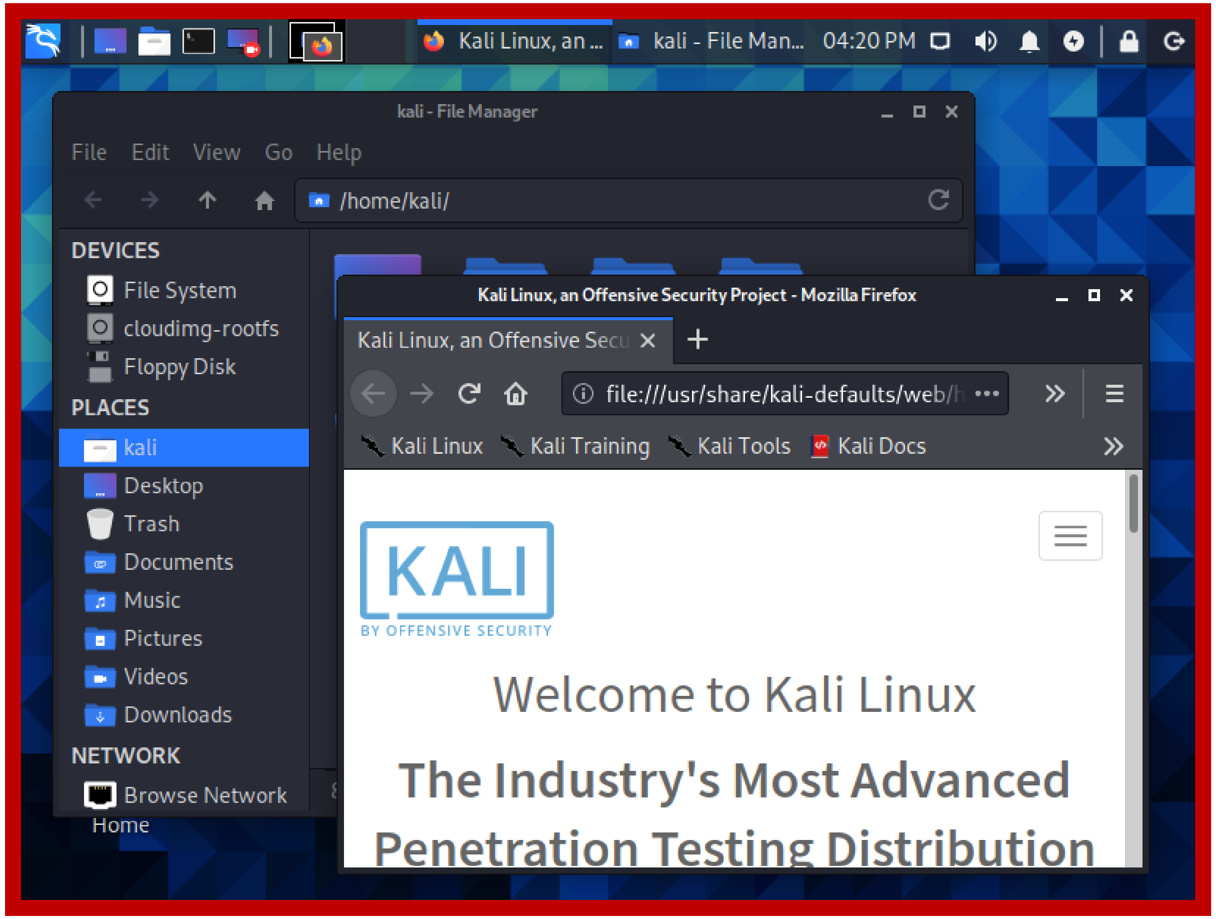
**Protip: A *program* in execution is called a *process***



# Multiprogramming and Multitasking



# Multiprogramming and Multitasking



Welcome to Kali Linux

The Industry's Most Advanced Penetration Testing Distribution

**Multitasking:** The CPU switches jobs so frequently that users can *interact* with each job while it is running, creating **interactive computing** ☺

In general

- Response time should be *fast* (e.g., sub-second)
- User have >1 program executing in memory
- If >1 jobs ready to run at the same time, then **scheduling** takes place
- If processes don't fit in memory, **swapping** moves them in and out to run
- **Virtual memory** allows execution of processes not completely in memory

# Dual-Mode Operation

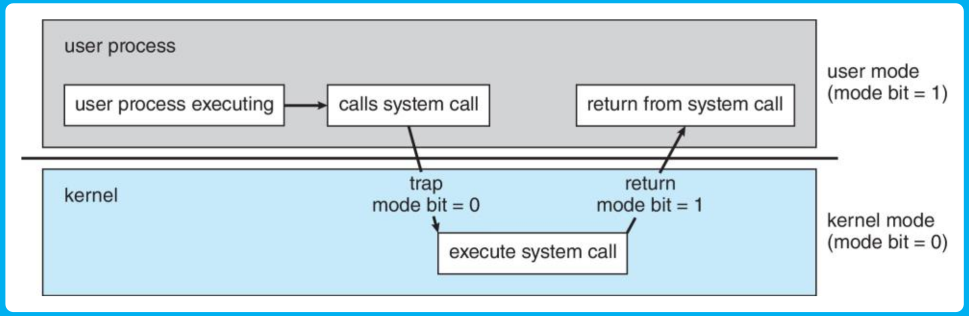
**User mode** and **kernel mode**

- **Mode bit** provided by hardware
- Provides ability to distinguish when system is running user code or kernel code

Some instructions are designated as **privileged**, only executable in kernel mode

How do we guarantee that user does not explicitly set the mode bit to "kernel"?

- System call sets mode to kernel, and on return from call resets it back to user
- Instruction to switch to kernel mode is handled specially (sometimes via a trap, sometimes as a unique instruction)



Dual-mode operation allows OS to protect itself and other system components

**Protip: Kernel mode is also called *supervisor mode*, *system mode*, or *privileged mode***

# Timer

Timer to prevent infinite loop or process hogging resources

Timer is set to interrupt the computer after some time period

Keeps a counter that is decremented by the physical clock

The operating system set the counter (privileged instruction)

When counter reaches zero, an interrupt is generated

Set up before scheduling process to regain control or terminate program that exceeds allotted time

# Resource Management

# Recap

Memory management

Filesystem management

Mass storage management

Caching

***We have discussed some aspects of operating system resource management...***

# Why Are We Talking About Storage?

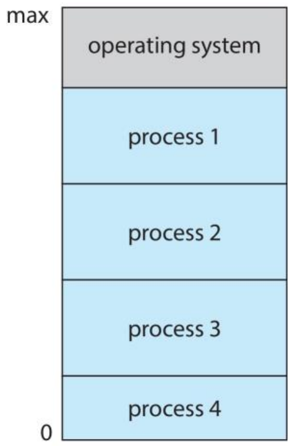
## Memory Management

To execute a program all (or part) of the instructions must be in memory

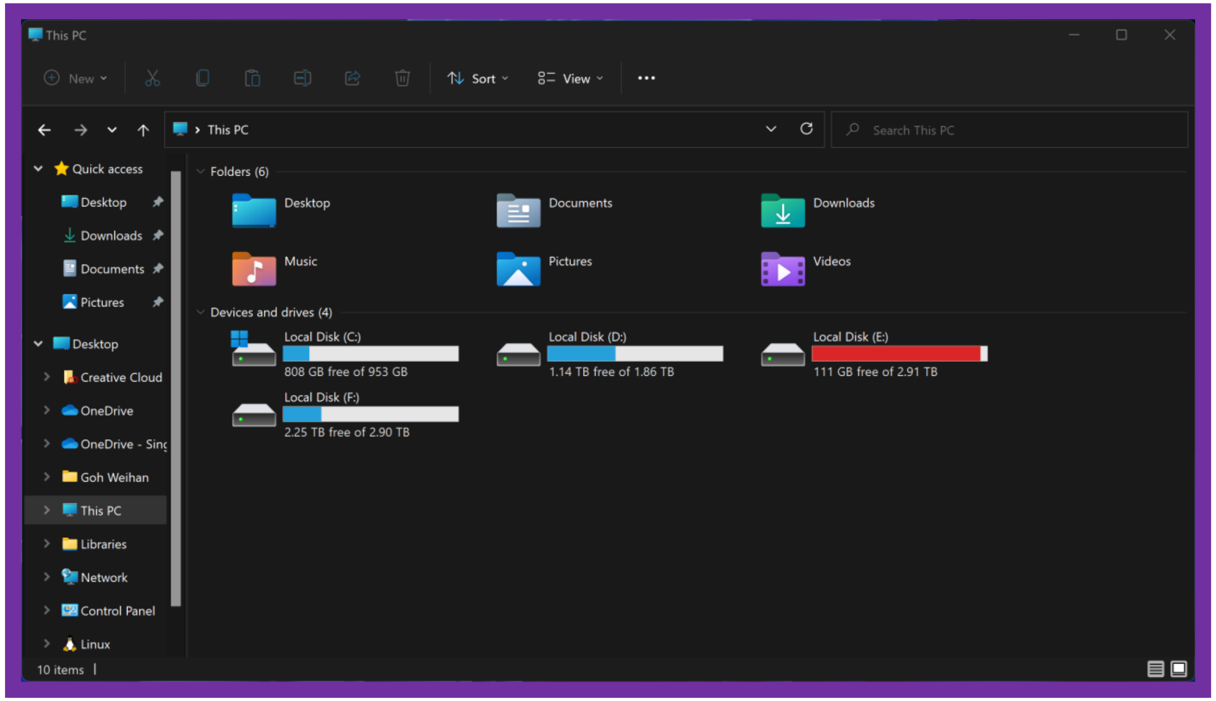
Memory management determines what is in memory and when

Memory management activities

- ○ Keeping track of which parts of memory are currently being used and by whom
- ○ Deciding which processes (or parts thereof) and data to move into and out of memory
- ○ Allocating and deallocating memory space as needed



# Why Are We Talking About Storage?



## Filesystem Management

Operating system provides uniform, logical view of information storage

Filesystem management

- ○ Files usually organized into directories
- ○ Access control on most systems to determine who can access what
- ○ Operating system activities include
  - ○ Creating and deleting files and directories
  - ○ Primitives to manipulate files and directories
  - ○ Mapping files onto secondary storage
  - ○ Backup files onto stable (non-volatile) storage media

# Why Are We Talking About Storage?

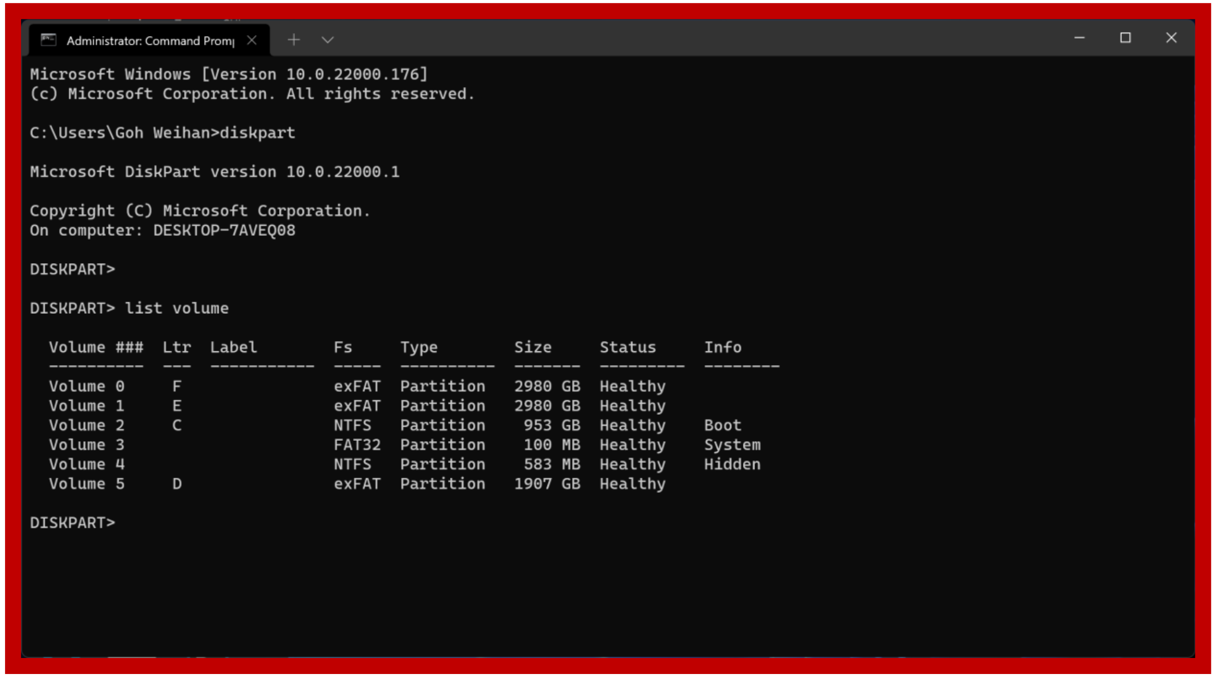
### Mass-Storage Management

Usually, disks are used to store data that does not fit in main memory, or data that must be kept for a long period of time

Proper management is of central importance

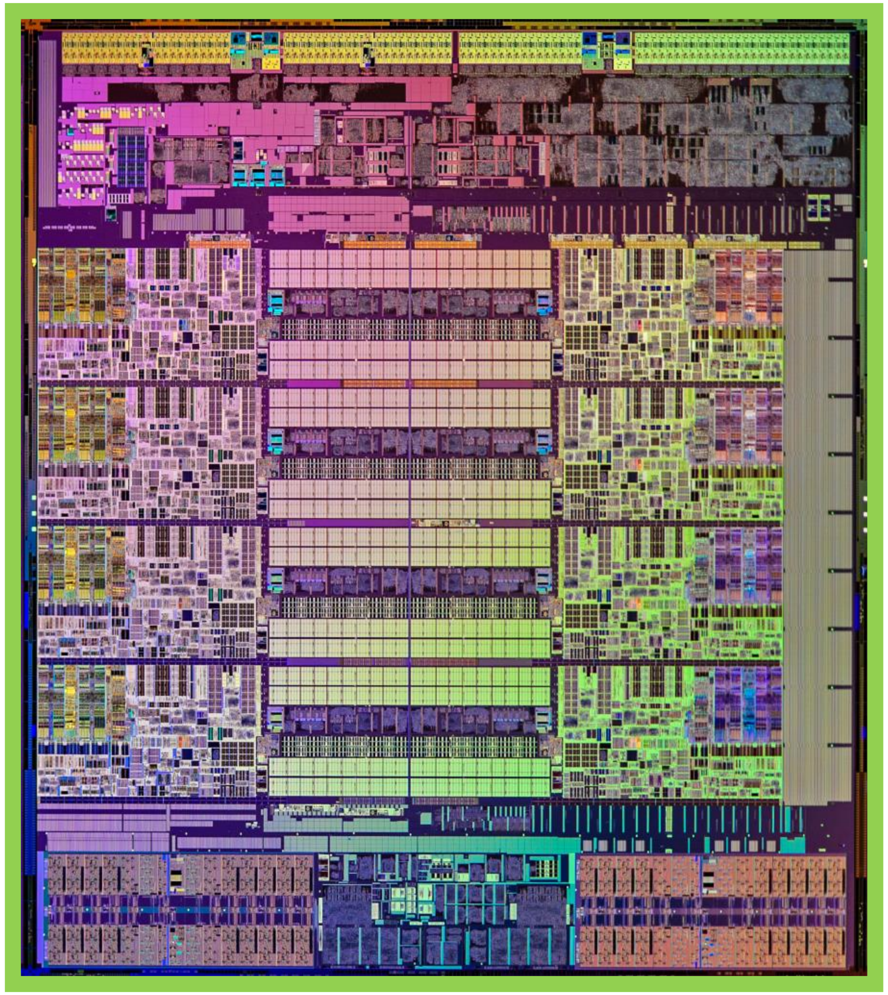
Operating system activities

- ○ Mounting and unmounting (i.e., safe removal)
- ○ Free-space management
- ○ Storage allocation
- ○ Disk scheduling
- ○ Partitioning
- ○ Protection



```
Microsoft Windows [Version 10.0.22000.176]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Goh Weihan>diskpart

Microsoft DiskPart version 10.0.22000.1

Copyright (C) Microsoft Corporation.
On computer: DESKTOP-7AVEQ08

DISKPART>

DISKPART> list volume

  Volume ###  Ltr  Label        Fs     Type        Size     Status     Info
  ----------  ---  -----------  -----  ----------  -------  ---------  --------
  Volume 0     F                exFAT  Partition   2980 GB  Healthy
  Volume 1     E                exFAT  Partition   2980 GB  Healthy
  Volume 2     C                NTFS   Partition    953 GB  Healthy    Boot
  Volume 3                      FAT32  Partition    100 MB  Healthy    System
  Volume 4                      NTFS   Partition    583 MB  Healthy    Hidden
  Volume 5     D                exFAT  Partition   1907 GB  Healthy

DISKPART>
```

# Why Are We Talking About Storage?



## Caching

Information in use copied from slower to faster storage temporarily

Faster storage (cache) checked first to determine if information is there

- If it is, information used directly from cache (fast)
- If not, data copied to cache and used there

Cache smaller than storage being cached

- Cache management important design problem
- Cache size and replacement policy

# Process Management

Process needs resources to accomplish its task

- CPU, memory, I/O, files
- Initialization data

Process termination requires reclaim of any reusable resources

Typically, a system has many processes, some user, and the operating system running *concurrently* on one or more CPUs

- Concurrency by multiplexing the CPUs among the processes / threads

**A program in execution is called a process**

- Program = passive entity
- Process = active entity

**Single-threaded process has one *program counter* specifying location of next instruction to execute**

- Process executes instructions "sequentially", one at a time, until completion

**Multi-threaded process has one program counter *per thread***

# Process Management Activities

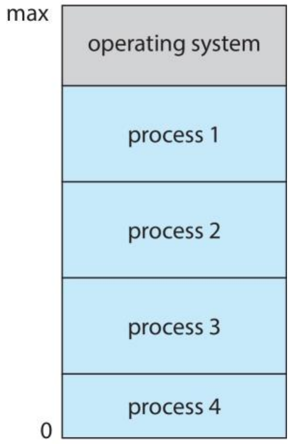
Create / delete user and system processes

Suspend / resume processes

Provide mechanisms for process synchronization

Provide mechanisms for process communication

Provide mechanisms for deadlock handling



# I/O Subsystem Management

**Hide peculiarities of hardware devices from the user**

**Memory management of I/O, including**

- **Buffering:** Storing data temporarily while it is being transferred
- **Caching:** Storing parts of data in faster storage for performance
- **Spooling:** the overlapping of output of one job with input of other jobs

**General device-driver interface**

**Drivers for specific hardware devices**

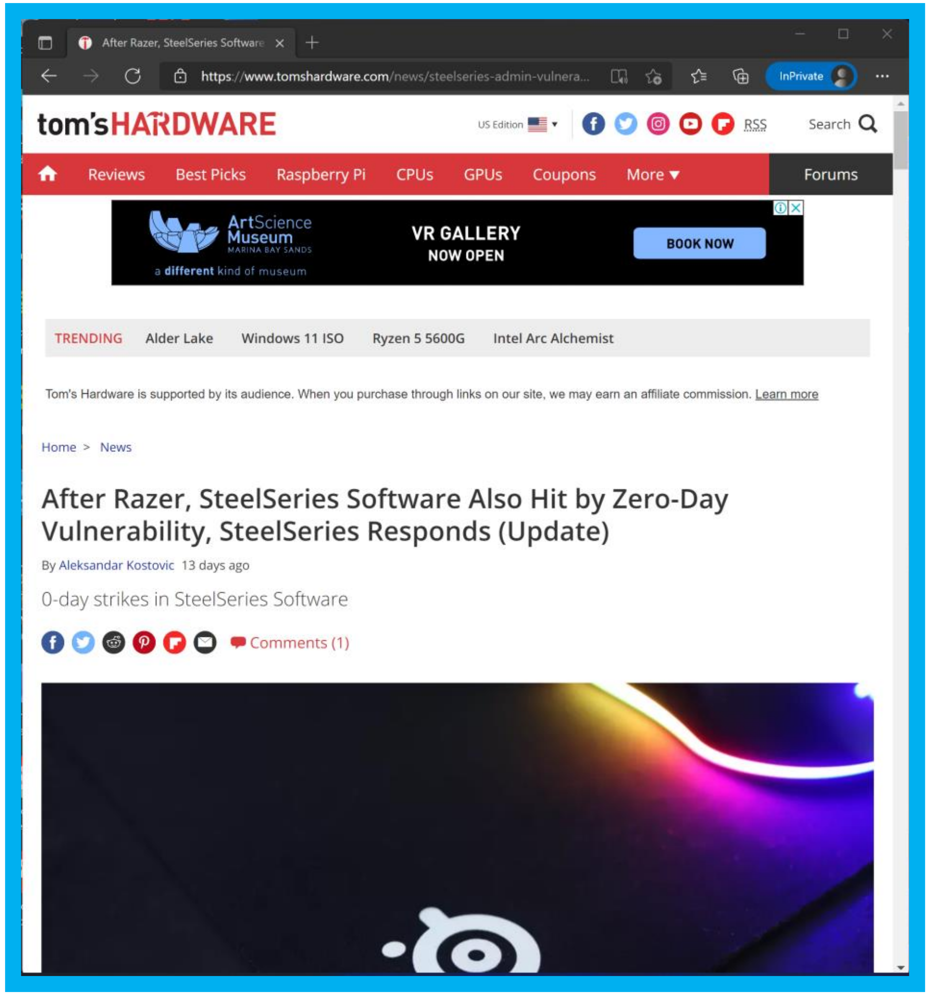
# Security and Protection

# Security and Protection

**Systems generally distinguish among users to determine who can do what**

- User identities (user IDs, security IDs) include name and associated identifier, one per user
- User identifier then associated with all files and processes of that user to determine access control
- Group identifier (group ID) allows sets of users to be defined and controls managed, then also associated with each process and file

**Privilege escalation allows user to access resources of another user with *more* rights**



Tom's Hardware is supported by its audience. When you purchase through links on our site, we may earn an affiliate commission. Learn more

Home > News

After Razer, SteelSeries Software Also Hit by Zero-Day Vulnerability, SteelSeries Responds (Update)

By Aleksandar Kostovic 13 days ago

0-day strikes in SteelSeries Software

# Computing Environments

# Traditional | Emulated | Virtualized | Mobile | Cloud Computing | Real-Time Embedded

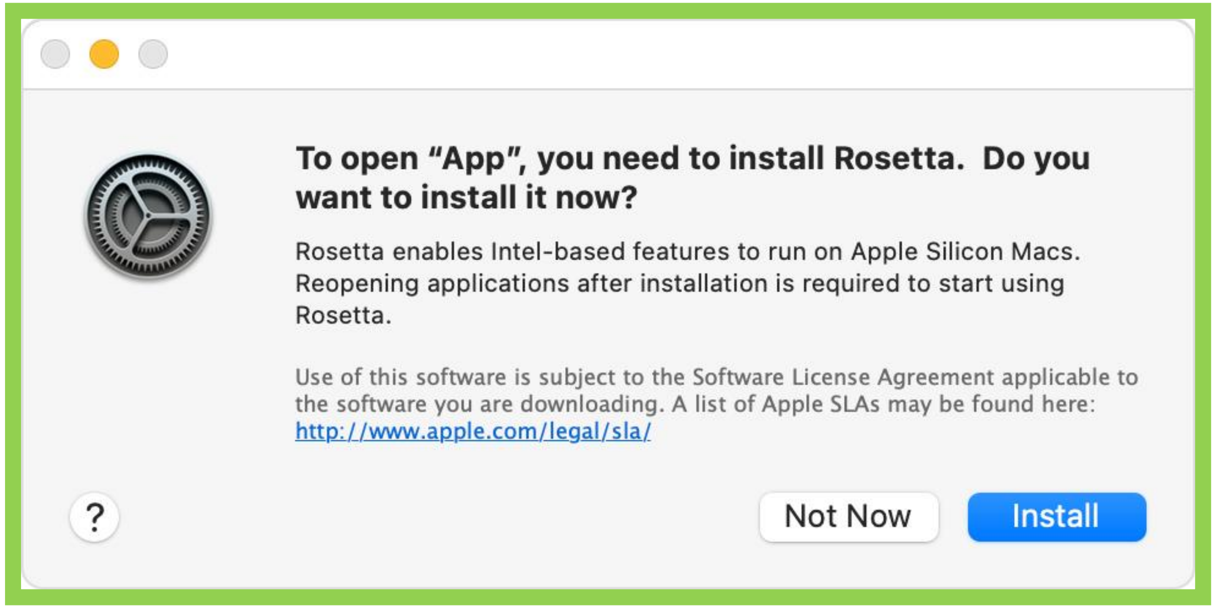
# Traditional Computing

Stand-alone, general-purpose machines

Definition kind of blurred these days as most computer systems **interconnect** with others (e.g., over the Internet)

Networking is now ubiquitous

# Emulation



**To open "App", you need to install Rosetta. Do you want to install it now?**

Rosetta enables Intel-based features to run on Apple Silicon Macs. Reopening applications after installation is required to start using Rosetta.

Use of this software is subject to the Software License Agreement applicable to the software you are downloading. A list of Apple SLAs may be found here: http://www.apple.com/legal/sla/

**Emulation** involves simulating computer hardware in software

Typically used for running applications when source CPU type different from target CPU type (e.g., running ARM applications on x86)

Can be (and generally) slow

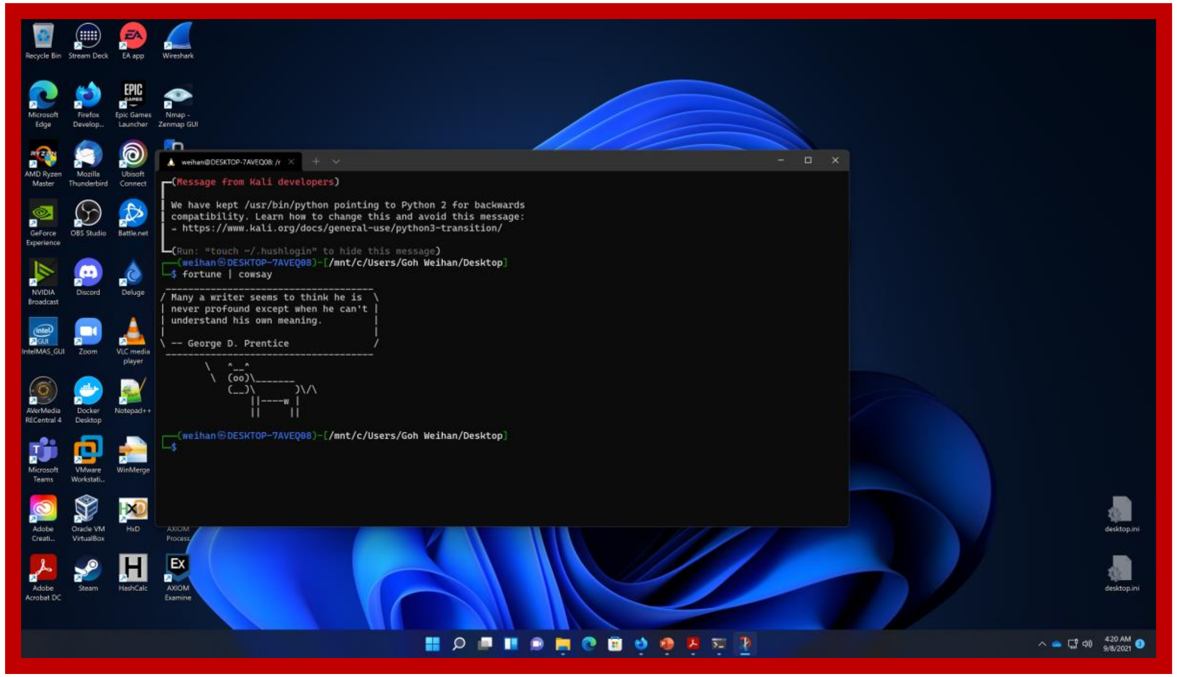
- Every machine-level instruction that runs natively on the source system must be **translated** to the equivalent on the target system

**Where has this been used before?**

# Virtualization

**Operating system natively compiled for CPU, running guest operating systems that are also natively compiled for CPU**

- For example, using VMware Workstation to run Windows XP guests on a Windows 10 host system
- Virtual machine manager (VMM) provides virtualization services



```
┌─(Message from Kali developers)
│
│ We have kept /usr/bin/python pointing to Python 2 for backwards
│ compatibility. Learn how to change this and avoid this message:
│ ⇒ https://www.kali.org/docs/general-use/python3-transition/
│
└─(Run: "touch ~/.hushlogin" to hide this message)
┌──(weihan㉿DESKTOP-7AVEQ08)-[/mnt/c/Users/Goh Weihan/Desktop]
└─$ fortune | cowsay
 _____________________________________
/ Many a writer seems to think he is  \
| never profound except when he can't |
| understand his own meaning.         |
|                                     |
\ -- George D. Prentice               /
 -------------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||

┌──(weihan㉿DESKTOP-7AVEQ08)-[/mnt/c/Users/Goh Weihan/Desktop]
└─$
```

# Mobile Computing

Handheld smartphones, tablets, etc.

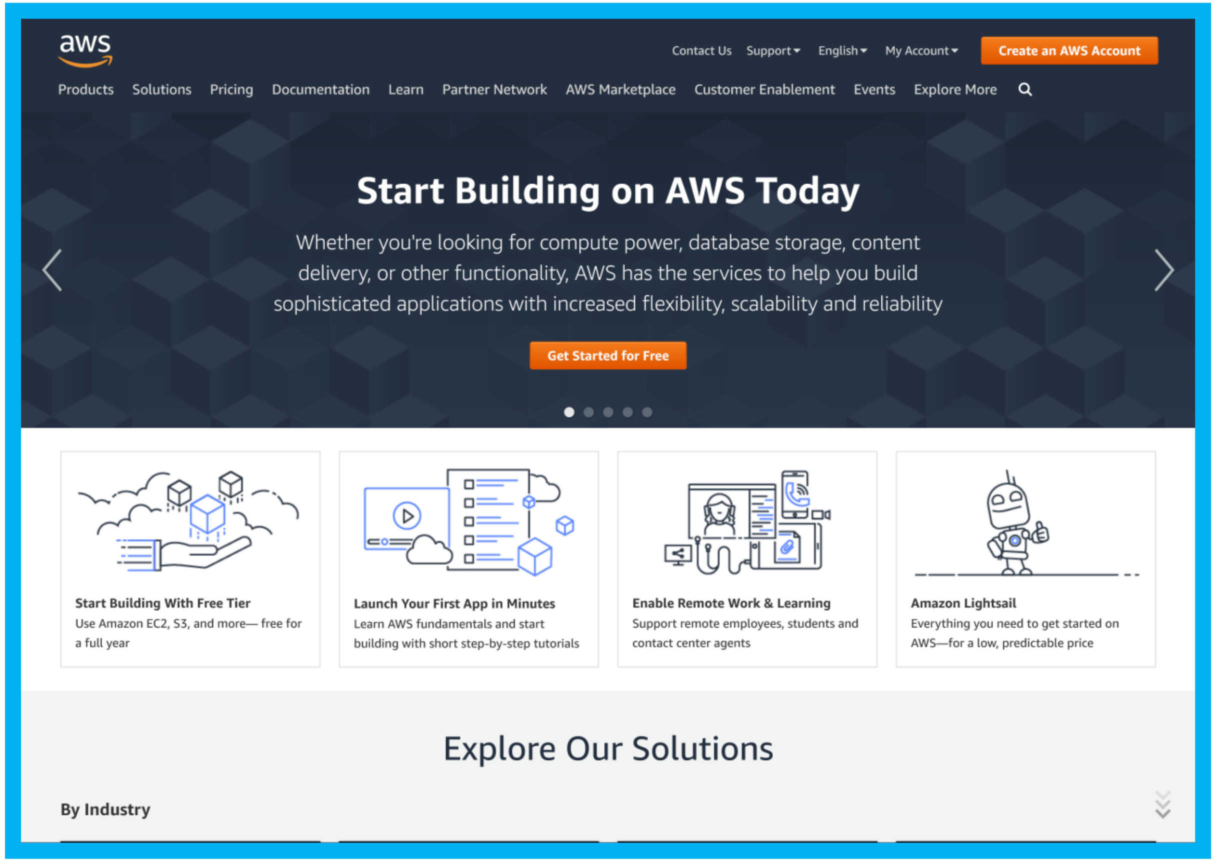
**More features**, such as GPS, gyroscope, etc.

Allow new types of **use cases** like augmented reality

Use **wireless or cellular data** networks for connectivity

Ecosystem led by Apple iOS and Google Android

# Cloud Computing



## Start Building on AWS Today

Whether you're looking for compute power, database storage, content delivery, or other functionality, AWS has the services to help you build sophisticated applications with increased flexibility, scalability and reliability

**Start Building With Free Tier**
Use Amazon EC2, S3, and more— free for a full year

**Launch Your First App in Minutes**
Learn AWS fundamentals and start building with short step-by-step tutorials

**Enable Remote Work & Learning**
Support remote employees, students and contact center agents

**Amazon Lightsail**
Everything you need to get started on AWS—for a low, predictable price

Explore Our Solutions

**By Industry**

Concept of *almost-anything-as-a-service*

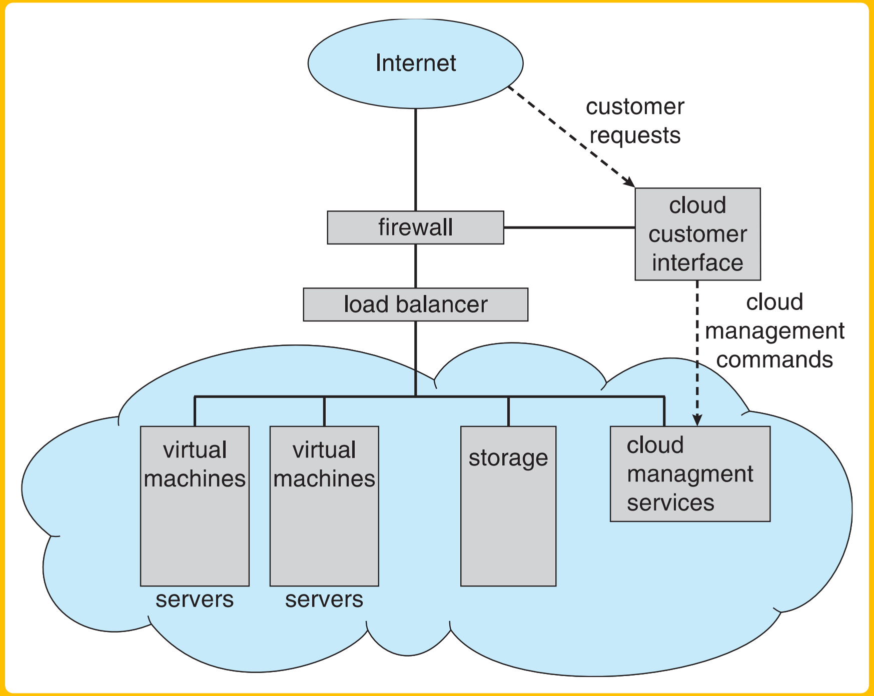
Delivers computing, storage, apps, etc. as a service across a network

Can be thought as an extension of virtualization as they use virtualization as base for their functionality

- Amazon EC2 has thousands of servers, **millions of virtual machines**, petabytes of storage available across the Internet, where users pay based on usage

# Cloud Computing

- Cloud computing environments composed of traditional operating systems + VMMs + cloud management tools



# Cloud Computing

## Almost-anything-as-a-service

**Software-as-a-Service (SaaS):** One or more applications available via the Internet (e.g., Google Docs, Dropbox, etc.)

**Platform-as-a-Service (PaaS):** Software stack ready for application use via the Internet (e.g., AWS Elastic Beanstalk, Google App Engine, etc.)

**Infrastructure-as-a-Service (IaaS):** Servers or storage available over Internet (e.g., AWS EC2, DigitalOcean, etc.)

## Cloud types

**Public cloud:** Available via Internet to anyone willing to pay

**Private cloud:** Run by a company for the company's own use

**Hybrid cloud:** Includes both public and private cloud components

# Embedded Systems

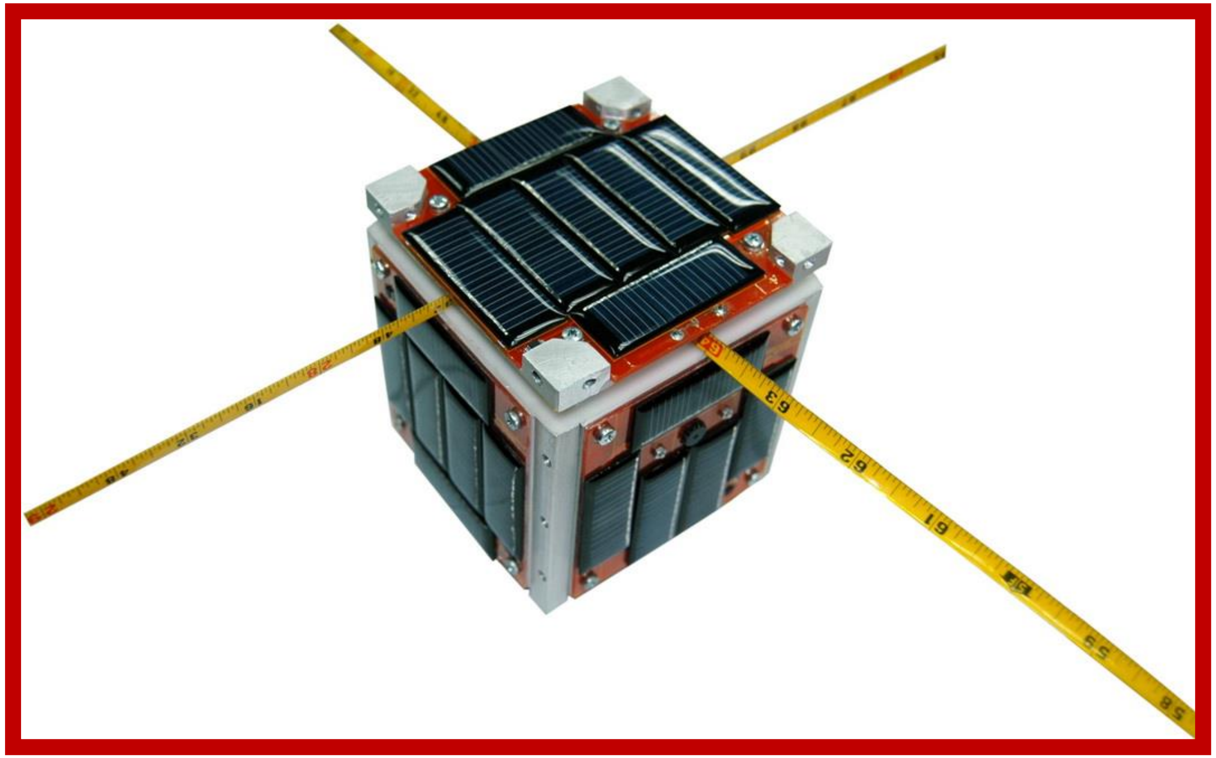
**Most prevalent** form of computers

- Vary considerably depending on purpose
- Operating system may be limited-purpose, etc.

Some systems may have operating system, some perform tasks without

**Real-time operating system** has well-defined fixed time constraints

- Processing must be done within constraint
- Correct operation only if constraints met



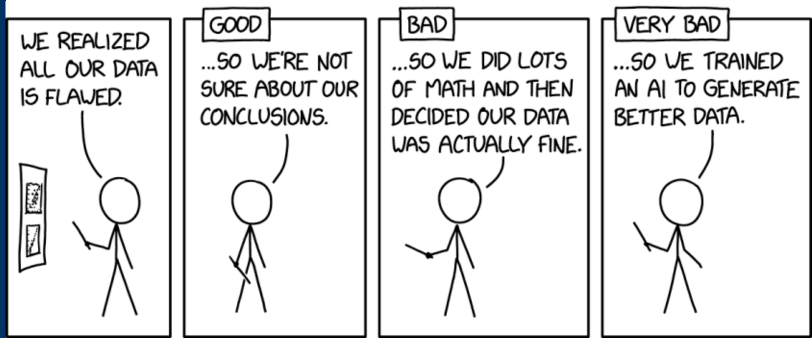
# 👋 Questions?

# Thank You!

📨 Weihan.Goh {at} Singaporetech.edu.sg

🔗 https://www.singaporetech.edu.sg/directory/faculty/weihan-goh

🔗 https://sg.linkedin.com/in/weihan-goh


WE REALIZED ALL OUR DATA IS FLAWED.
GOOD
...SO WE'RE NOT SURE ABOUT OUR CONCLUSIONS.
BAD
...SO WE DID LOTS OF MATH AND THEN DECIDED OUR DATA WAS ACTUALLY FINE.
VERY BAD
...SO WE TRAINED AN AI TO GENERATE BETTER DATA.