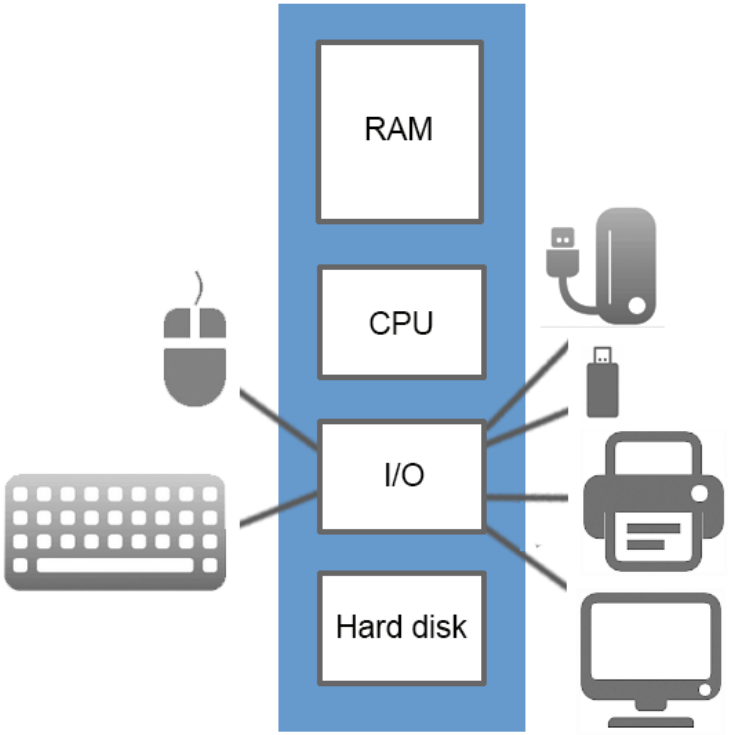
Chapter 15: Hardware Configuration

Core Hardware

- CPU
- RAM
- Firmware
- Mass storage devices



The CPU

- Processes data and performs calculations
- Two common types:
  - 32 bit x86
  - 64 bit x86_64
- To view CPU information
  - For detailed info: **$**cat `/proc/cpuinfo`
  - For summary: **$**`lscpu`

RAM

- Stores system and program data
- Max for 32 bit processors is 4GiB
- Max for 64 bit processors is 16EiB
- Swap space:
  - Hard drive space used when memory is low
  - Also called virtual memory or swap
- To display RAM and swap space:
  - For detailed info: **$**cat `/proc/meminfo`
  - For summary: **$**`free`

Firmware

- Software provided by the computer hardware:
  - BIOS: Basic Input Output System
  - UEFI: Unified Extensible Firmware Interface
- Initiates the boot process
- Helps the kernel recognizes devices
- Not standard on all systems
- Accessed using a function key (F2 or F12), ESC, or DEL depending on the computer

Mass storage devices

- Used to store programs, data, etc.
- Several types available:
  - SCSI: Small Computer System Interface
  - IDE: Integrated Drive Electronics
  - SATA: Serial Advanced Technology Attachement
  - USB: Universal Serial Bus
- User either serial (SCSI, SATA, USB) or parallel interfaces
  - Use **$**df –h to determine drive type

Plug and Play

- Coldplug
  - When system needs to be turned off to connect the device
  - Example: IDE hard drive
- Hotplug
  - Also called "plug and play"
  - When system can be running when device is connected
  - Example: a USB thumb drive

Hardware resources

- In order for a device to work correct, certain resources must be allocated
- Resources used to have to be manually configured
- Now resources are automatically configured

Hardware resources

- Four hardware resources:
  - io ports
  - io memory
  - interrupts
  - Direct Memory Access channels
- Resource files are locted in `/proc`:
  - `dma`, `iomem`, `ioports` files
  - `irq` subdirectory

View hardware devices

- View internal devices with `lspci`
  - Use `-v` (verbose) for more information
  - Also supports `-vv` and `-vvv` for even more verbose output
  - Use `-nn` to show vendors name and device code

lspci Example

```
$lspci
00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX Host bridge (
rev 01)
00:01.0 PCI bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX AGP bridge (re
v 01)
00:07.0 ISA bridge: Intel Corporation 82371AB/EB/MB PIIX4 ISA (rev 08)
00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
00:07.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:07.7 System peripheral: VMware Virtual Machine Communication Interface (rev 1
0)
00:0f.0 VGA compatible controller: VMware SVGA II Adapter
00:11.0 PCI bridge: VMware PCI bridge (rev 02)
00:15.0 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.1 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.2 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.3 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.4 PCI bridge: VMware PCI Express Root Port (rev 01)
<Output omitted>
03:00.0 Serial Attached SCSI controller: VMware PVSCSI SCSI Controller (rev 02)
0b:00.0 Ethernet controller: VMware VMXNET3 Ethernet Controller (rev 01)
```

View hardware devices

- View USB devices with `lsusb`
- First device is the USB hub (the ports)
- Second devices is an actual device connected to a USB port

View hardware devices

- Use `-v` (verbose) for more information:

```
$lsusb -v -d 0e0f:0003 | head -20
Bus 001 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Couldn't open device, some information will be missing
Device Descriptor:
  bLength                18
  bDescriptorType         1
  bcdUSB               1.10
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass         0
  bDeviceProtocol         0
  bMaxPacketSize0         8
  idVendor           0x0e0f VMware, Inc.
  idProduct          0x0003 Virtual Mouse
  bcdDevice            1.02
  iManufacturer           1
  iProduct                2
  iSerial                 0
  bNumConfigurations      1
  Configuration Descriptor:
    bLength               9
    bDescriptorType       2
```

Hardware subsystems

- Components that help the kernel manage hardware:
  - udev
  - sysfs
  - dbus

udev

- Maintains `/dev` pseudo filesystem
- Files under `/dev` represent devices, like hard drives
- Which files are created, along with ownership and permissions, can be administered by files in `/etc/udev/rules.d` directory

Kernel modules

- Software designed to help the kernel communicate with a device
- List active modules with the `lsmod` command:

```
$ lsmod |sort
Module                  Size  Used by
ablk_helper            12939  1 aesni_intel
aes_x86_64             16766  1 aesni_intel
aesni_intel           148509  0
ata_piix               32931  1
autofs4                33893  0
bridge                100280  0
btrfs                 837676  1
button                 12771  0
cdrom                  46399  1 sr_mod
```

Output of lsmod

- First column - the name of the loaded module
- Second column - shows the size in bytes of the module
- Third column - indicates how many “things” depend on the current module being loaded
- Fourth column - shows the name of the module that depends on the current module

Using modprobe

- Modules are normally loaded automatically
- To load manually, use the `modprobe` command:

```
$modprobe vfat
```

- Loads module and any dependent modules
- Use `–r` to remove a module:

```
$modprobe –r vfat
```