

1. Please summarize the following server components:

- **Motherboard (System Board):**
The main circuit board connecting all components, providing communication pathways and power distribution.
- **CPU (Central Processing Unit):**
The processor executing instructions, handling computations, and managing tasks.
- **RAM (Memory):**
Temporary, high-speed storage for active data and processes, enabling quick CPU access.
- **Storage Drives (HDD/SSD/NVMe):**
Persistent storage; HDDs use spinning disks, SSDs use flash memory, and NVMe offers faster data transfer via PCIe.
- **RAID Controller (Smart Array):**
Manages multiple drives for redundancy or performance, enabling RAID configurations.
- **Power Supply Unit (PSU):**
Converts AC to DC power, supplying stable voltage to components.
- **Network Interface Card (NIC):**
Enables network connectivity, facilitating data transfer over LAN or internet.
- **Cooling System (Fans and Heat Sinks):**
Dissipates heat from components to maintain optimal temperatures.
- **Expansion Slots (PCIe):**
Slots for adding cards (e.g., GPUs, NICs) to enhance functionality.
- **Chassis (Rack/Tower/Blade):**
The physical enclosure; rack for data centers, tower for standalone, blade for compact, high-density setups.
- **BIOS/UEFI Firmware:**
Firmware initializing hardware during boot and providing a basic interface for OS loading.
- **Backplane:**
A board connecting multiple drives or modules, simplifying cabling and enhancing scalability.

2. What are IPMI and iLO, and what are their functions?

- **IPMI (Intelligent Platform Management Interface):**
A standardized interface for out-of-band server management, enabling tasks like power control, monitoring, and logging.
- **iLO (Integrated Lights-Out):**
HP's proprietary implementation of IPMI with additional features (web GUI, remote console,

virtual media, etc.).

3. How do IPMI or iLO relate to the BIOS or UEFI firmware?

IPMI or iLO works independently of BIOS/UEFI, using a dedicated chip. However, they can interact with BIOS/UEFI settings, allowing remote firmware configuration, updates, or resets.

4. What are CPU sockets on a server, and what is their purpose?

These are physical interfaces on the motherboard where CPUs are installed. Some servers have multiple sockets to support multiple processors for higher performance.

5. why was the pseudo file system introduced in Linux?

Linux follows the "everything is a file" philosophy. Pseudo filesystems (like `/proc`, `/sys`) expose kernel and system information as files/directories, enabling easier access without needing separate tools or APIs.

6. What are the differences between a pseudo file system and a normal file system?

A pseudo file system is virtual, in-memory, for kernel data, unlike a normal file system storing data on disk.

7. What kind of information is available in the `/sys/` directory?

The `/sys/` directory shows kernel and hardware details, like devices and power settings.

8. What is DMA (Direct Memory Access), and what is its use case in Linux?

DMA lets devices access RAM directly, used in Linux for fast disk and network data transfers.

9. What does the `lsblk` command do internally when executed in Linux? Do `lsusb`, `lspci`, and `lshw` function similarly?

The `lsblk` command lists block devices from `/sys/block`; `lsusb`, `lspci`, and `lshw` do similar for USB, PCI, and all hardware.

10. How can we simulate a shutdown operation via the `/sys` file system?

Simulate shutdown by writing `"reboot"` to `/sys/power/state`.

11. What are the different types of kernels?

- Monolithic vs. Microkernel vs. Hybrid
- What are the advantages and disadvantages of each?

Kernel Type	Description	Pros	Cons
Monolithic	Entire OS runs in one address space	Fast, efficient	Bugs can crash system
Microkernel	Only essential parts in kernel; rest in user space	More stable and secure	Slower due to IPC overhead
Hybrid	Combines both	Better modularity	Complex design

12. Why is the first sector of a disk used for the MBR?

The BIOS /legacy firmware loads the first 512 bytes of the disk (sector 0) during boot. This sector contains:

- Bootloader code (first-stage)
- Partition table
- Boot signature

13. If the MBR is located in the first 512 bytes, how does it know the location of GRUB or another bootloader to load the kernel?

The MBR's small bootloader code loads the GRUB stage 1.5 or stage 2 from a known disk offset (usually immediately after MBR or in a dedicated partition like `/boot`).

14. What are `.efi` files, and what is their role in the boot process?

`.efi` files are UEFI executables that start bootloaders or kernels from the ESP .

15. What is the ESP (EFI System Partition) in UEFI, and how is it used?

A special FAT32 partition that UEFI firmware reads. It stores `.efi` bootloaders and configuration files. OS installers write their `.efi` files here for booting.

16. Please explain the following section from `/etc/grub/grub.conf` :

A special FAT32 partition that UEFI firmware reads. It stores `.efi` bootloaders and configuration files. OS installers write their `.efi` files here for booting.