۱-اجزای اصلی یک سرور:

- **مادربورد(Motherboard / System Board)**

مادربورد مثل اسکلت یا مرکز فرمان سروره. همه‌ی قطعات مثلCPU ، RAM، هارد و کارت شبکه روش نصب می‌شن و با هم از طریق این برد ارتباط برقرار می‌کنن.

- **پردازنده(CPU - Central Processing Unit)**

پردازنده یا مغز سروره. همه‌ی محاسبات، پردازش‌ها و اجرای برنامه‌ها توسط این قطعه انجام می‌شه.

- **حافظه موقت(RAM - Memory)**

RAMاطلاعات موقتی رو نگه‌داری می‌کنه که سرور در لحظه بهشون نیاز داره. سرعتش خیلی زیاده، ولی بعد از خاموش شدن سرور، اطلاعاتش پاک می‌شن.

- **دیسک ذخیره‌سازی(Storage Drives - HDD / SSD / NVMe)**

این قطعات اطلاعات رو ذخیره می‌کنن:

HDD: ارزان‌تر ولی کندتر.

SSD: سریع‌تر ازHDD

NVMe: فوق‌العاده سریع و مدرن، برای کارهای سنگین.

- **کنترلر(RAID Controller - Smart Array) RAID**

این قطعه چند هارد رو با هم ترکیب می‌کنه تا اطلاعات یا امن‌تر بشن (نسخه پشتیبان) یا سرعت دسترسی بهشون بیشتر بشه.

- **منبع تغذیه(Power Supply Unit - PSU)**

منبع تغذیه برق مورد نیاز تمام قطعات سرور رو تأمین می‌کنه. در سرورها معمولاً از دو منبع تغذیه استفاده می‌شه برای اطمینان از کارکرد مداوم.

- **کارت شبکه(NIC - Network Interface Card)**

کارت شبکه به سرور اجازه می‌ده به اینترنت یا شبکه داخلی وصل بشه. ممکنه یک یا چند پورت شبکه داشته باشه.

- **سیستم خنک‌کننده(Cooling System - Fans & Heat Sinks)**

برای اینکه سرور داغ نکنه، از فن‌ها و هیت‌سینک‌ها استفاده می‌شه که گرما رو پخش می‌کنن یا خارج می‌کنن.

- **اسلات‌های گسترش(Expansion Slots - PCIe)**

جای نصب کارت‌های اضافه مثل:

- کارت گرافیک
- کارت RAID اضافی
- کارت شبکه پرسرعت و...

- **بدنه یا شاسی(Chassis - Rack / Tower / Blade)**

نوع فیزیکی بدنه‌ی سرور:

- **Rack:** برای مراکز داده، داخل رک قرار می‌گیره.
- **Tower:** شبیه کیس کامپیوتر خانگی.
- **Blade:** خیلی فشرده، مناسب محیط‌های با فضای کم.

- **بایوس یا(UEFI (BIOS / UEFI Firmware**

یک نرم‌افزار پایه‌ای که قبل از اجرای سیستم‌عامل اجرا می‌شه و کار راه‌اندازی اولیه سخت‌افزارها رو انجام می‌ده.

- **بَک‌پلِین(Backplane)**

یک برد در قسمت داخلی بدنه سرور که به هاردها وصل می‌شه و اجازه می‌ده بدون نیاز به کابل جداگانه، دیسک‌ها راحت و سریع متصل یا جدا بشن. مخصوصاً در سرورهای رک‌مونت استفاده می‌شه.

۲- IPMI و iLO چی هستن و چه کاربردی دارن؟

IPMI (Intelligent Platform Management Interface) و iLO (Integrated Lights-Out) هر دو ابزارهای مدیریتی برای سرورها هستن که به مدیر سیستم اجازه می‌دن از راه دور و حتی وقتی سیستم خاموشه، سرور رو مدیریت کنه. باهاش می‌تونی از راه دور سرور رو روشن/خاموش کنی، وضعیت سخت‌افزار رو ببینی، دمای قطعات، سرعت فن‌ها و خطاها رو چک کنی.

iLO مخصوص سرورهای HP هست، ولی مشابه اون رو در برندهای دیگه مثل iDRAC برای Dell یا BMC برای Supermicroهم داریم.

۳ -IPMI یا iLO چه ارتباطی با BIOS یا UEFI دارن؟

این سیستم‌ها قبل از بوت شدن سیستم‌عامل و حتی مستقل از اون کار می‌کنن. بنابراین:

- می‌تونن از راه دور وارد **BIOS/UEFI**بشن و تنظیمات سخت‌افزاری رو تغییر بدن.
- درواقع، IPMI یا iLO یک لایه مدیریتی خارج از سیستم‌عامله که می‌تونه تنظیمات BIOS رو هم کنترل کنه.

۴- سوکت CPU روی سرور چی هست و چه کاربردی داره؟

سوکت CPU محلیه روی مادربورد که پردازنده داخل اون قرار می‌گیره.

- سرورها معمولاً سوکت‌های بزرگ و قوی‌تری دارن و گاهی هم بیش از یک سوکت دارن (برای نصب چند پردازنده).
- نوع سوکت مشخص می‌کنه چه مدلCPU هایی می‌تونن روی مادربورد نصب بشن.

۵ -چرا در لینوکس pseudo file system معرفی شد؟

نکته: به فلسفه طراحی لینوکس توجه کن

لینوکس می‌گه: "همه‌چیز یک فایل است"!
برای همین، حتی چیزهایی که واقعاً فایل نیستن (مثل اطلاعات سخت‌افزاری یا تنظیمات سیستم)، به شکل فایل نشون داده می‌شن.

- **Pseudo file system**مثل `proc/`و `sys/`به این خاطر ساخته شدن که اطلاعات سیستمی مثل اطلاعاتCPU ، رم، فرآیندها رو به صورت فایل قابل خوندن دربیارن.
- این باعث می‌شه مدیریت سیستم آسون‌تر بشه چون می‌تونی با دستورات ساده مثل `cat`, `echo`, `grep` با اونها کار کنی.

۶ -تفاوت فایل‌سیستم معمولی با pseudo file system چیه؟

فایل‌سیستم شبه(Pseudo FS)	فایل‌سیستم معمولی	ویژگی
در حافظه موقت(RAM)	روی دیسک(HDD/SSD)	محل ذخیره
با ریستارت سیستم پاک می‌شه	ماندگار	ماندگاری
`/proc`, `/sys`	`/home`, `/var`	مثال
ارائه اطلاعات سیستمی به شکل فایل	ذخیره‌سازی داده‌ها	هدف
بیشترش فقط خواندنیه	اغلب قابل نوشتن	قابلیت نوشتن

۷- چه نوع اطلاعاتی در مسیر /sys/موجود است؟

دایرکتوری /sys/بخشی از **سیستم فایل مجازی sysfs** در لینوکس است که اطلاعات مربوط به **سخت‌افزار سیستم و کرنل** را به صورت ساختار دایرکتوری نمایش می‌دهد. این مسیر به کاربر و برنامه‌ها اجازه می‌دهد وضعیت سخت‌افزار را بررسی و در برخی موارد آن را تنظیم کنند. برای مثال:

- /sys/block/ — اطلاعات مربوط به دیسک‌ها و پارتیشن‌ها
- /sys/class/ — دستگاه‌ها بر اساس کلاس مانندnet ، sound، input
- /sys/devices/ — نمایش ساختار فیزیکی دستگاه‌ها
- /sys/module/ — ماژول‌های کرنل بارگذاری‌شده
- /sys/power/ — تنظیمات مربوط به مدیریت مصرف انرژی مانند suspend و hibernate

۸ -DMA (Direct Memory Access) چیست و چه کاربردی در لینوکس دارد؟

DMAمخفف **Direct Memory Access** به معنی "دسترسی مستقیم به حافظه" است. این مکانیزم به **دستگاه‌های سخت‌افزاری** مثل کارت شبکه، کارت صدا، یا هارد دیسک اجازه می‌دهد که **مستقیم به حافظه اصلی (RAM)** دسترسی پیدا کرده و داده بخوانند یا بنویسند، **بدون نیاز به دخالت CPU** در هر انتقال.

کاربرد در لینوکس:
در لینوکس، DMA برای افزایش سرعت عملکرد سیستم استفاده می‌شود. به عنوان مثال هنگام کپی فایل از دیسک به رم، سیستم می‌تواند از DMA استفاده کند تا CPU آزاد بماند و به وظایف دیگر رسیدگی کند.

۹ .دستور lsblkدر داخل لینوکس چه کاری انجام می‌دهد؟

آیا دستورهای lsusb، lspciو lshwهم عملکرد مشابهی دارند؟

- دستور lsblkاطلاعات مربوط به **دستگاه‌های ذخیره‌سازی (Block Devices)** مانند هارد دیسک، SSD، USBو پارتیشن‌ها را نمایش می‌دهد.
- به‌طور داخلی این دستور داده‌ها را از مسیرهای /sys/block/و /proc/partitions/می‌خواند.

عملکرد مشابه:

- lsusb: دستگاه‌های متصل به پورت USB را لیست می‌کند. اطلاعات را از /sys/bus/usb/و /proc/bus/usb/می‌گیرد.
- lspci: لیستی از دستگاه‌های متصل به گذرگاه PCI نمایش می‌دهد. داده‌ها را از /sys/bus/pci/و اطلاعات low-level از libpciمی‌گیرد.
- lshw: اطلاعات سخت‌افزاری سطح پایین را به صورت کامل و ساختاریافته نمایش می‌دهد. این ابزار از منابع مختلف مانند /proc, /sys,و دستورات low-level برای ساختار سیستم استفاده می‌کند.

در کل، همه‌ی این دستورات از **sysfs (/sys)**و **procfs (/proc)**برای دریافت اطلاعات سخت‌افزاری استفاده می‌کنند.

۱۰- چگونه می‌توان عملیات خاموش کردن (shutdown) را از طریق فایل سیستم /sys/ شبیه‌سازی کرد؟

در لینوکس می‌توان عملیات shutdown یا خاموش‌کردن سیستم را با نوشتن مقادیر خاصی در فایل‌های مسیر /sys/class/power_supply/ یا دقیق‌تر از طریق:

echo o > /proc/sysrq-trigger

echo shutdown > /sys/power/state

۱۱- انواع مختلف کرنل چیست؟

- **Monolithic vs. Microkernel vs. Hybrid مزایا و معایب هر کدام چیست؟**

نوع کرنل	توضیح	مزایا	معایب
Monolithic Kernel	همه ماژول‌ها مانند مدیریت حافظه، فایل سیستم، درایورها، و ... در فضای کرنل اجرا می‌شوند.	سرعت بالا، عملکرد بهتر به خاطر نبود ارتباط بین پردازه‌ای زیاد.	پیچیدگی زیاد، احتمال بروز باگ و crash بیشتر به دلیل اجرای همه چیز در سطح کرنل.
Microkernel	فقط بخش‌های اصلی مانند مدیریت پردازه و حافظه در کرنل هستند. باقی سرویس‌ها (مثل درایورها) در سطح user space اجرا می‌شوند.	امنیت بیشتر، پایداری بهتر درایورها crash کنند، کل سیستم crash نمی‌کند.	کندتر است چون ارتباط زیاد بین user و kernel space دارد.
Hybrid Kernel	ترکیبی از monolithic و microkernel. برخی سرویس‌ها در user space و برخی در kernel space. مثل ویندوز NT ، macOS، و کرنل جدید لینوکس.	توازن بین عملکرد و پایداری.	پیچیدگی پیاده‌سازی بیشتر از monolithic.

۱۲- چرا سکتور اول دیسک برای MBR استفاده می‌شود؟

سکتور اول دیسک (بایت‌های ۰ تا ۵۱۱) مکان ویژه‌ای است که توسط سخت‌افزار (BIOS) به عنوان اولین نقطه برای بارگذاری سیستم‌عامل در نظر گرفته شده. این ناحیه به نام **MBR (Master Boot Record)** شناخته می‌شود و هنگام بوت، BIOS دقیقاً این سکتور را می‌خواند تا بداند کدام پارتیشن قابل بوت است.

۱۳- اگر MBR در ۵۱۲ بایت اول قرار دارد، چطور محل GRUB یا bootloader دیگر را می‌فهمد؟

MBR تنها **مرحله اول بوت (Stage 1)** را شامل می‌شود که بسیار کوچک است (حدود ۴۴۶ بایت برای کد اجرایی). این بخش:

- فقط وظیفه دارد مرحله بعدی مانند GRUB Stage 1.5 یا 2 Stage را از یک مکان مشخص روی دیسک بارگذاری کند.
- GRUBهنگام نصب، مکان دقیق 2 Stage را روی دیسک ذخیره می‌کند، و MBR آن آدرس را می‌داند.

در واقع، MBRخودش شامل کل GRUB نیست؛ فقط به اندازه‌ای هوشمند است که قسمت بعدی را از یک مکان مشخص لود کند.

۱۴-فایل‌های `efi`.چه هستند و در فرایند بوت چه نقشی دارند؟

فایل‌های `efi`.فایل‌های **اجرایی EFI (Extensible Firmware Interface)** هستند که توسط سیستم UEFI در زمان بوت اجرا می‌شوند. این فایل‌ها جایگزین MBR در سیستم‌های جدید هستند و شامل بوت‌لودرهایی مانند:

- `grubx64.efi`برای GRUB در سیستم‌های ۶۴بیتی
- `bootx64.efi`بوت‌لودر پیش‌فرض

وقتی سیستم UEFI راه‌اندازی می‌شود، وارد EFI System Partition می‌شود، فایل `efi`.را پیدا می‌کند، آن را اجرا می‌کند، و کرنل سیستم‌عامل را بارگذاری می‌کند.

۱۵ -ESP (EFI System Partition) در UEFI چیست و چگونه استفاده می‌شود؟

ESP مخفف **EFI System Partition** است. یک پارتیشن ویژه در دیسک است که توسط UEFI به عنوان مکان پیش‌فرض برای نگهداری بوت‌لودرها استفاده می‌شود.

ویژگی‌ها:

- معمولاً با فایل‌سیستم FAT32 فرمت می‌شود.
- شامل پوشه‌هایی مانند `/EFI/ubuntu/` ,`/EFI/Microsoft/`, `/EFI/boot/`و غیره است.
- هر سیستم‌عامل یا بوت‌لودر فایل `efi`.مخصوص خود را در این پارتیشن قرار می‌دهد.

نقش:
در زمان بوت، فریم‌ور UEFI وارد ESP می‌شود و فایل `efi`.مربوط به سیستم‌عامل را اجرا می‌کند. این پارتیشن الزامی است برای سیستم‌هایی که از UEFI استفاده می‌کنند.

۱۶-لطفاً بخش زیر از فایل /etc/grub/grub.conf را توضیح دهید:

توضیح کامل هر خط از menuentry در GRUB:

```
menuentry 'Ubuntu' --class ubuntu --class gnu-linux --class gnu --class os
```

◆تعریف یک گزینه‌ی منو با عنوان **Ubuntu**.
◆کلاس‌ها برای تنظیم استایل گرافیکی در صفحه بوت (مثلاً آیکون سیستم‌عامل) هستند.

```
$menuentry_id_option 'gnulinux-simple-3e3d2181-a1f5-4456-867c-a69f52c910e6'
```

◆شناسه‌ی منحصربه‌فرد برای این ورودی GRUB .از این شناسه برای ارجاع داخلی یا استفاده در دستورات auto-boot استفاده می‌کند.

```
{
recordfail
```

◆اگر بوت قبلی با خطا متوقف شده باشد، این دستور آن را ثبت می‌کند (برای تصمیم‌گیری در بوت بعدی).

```
load_video
gfxmode $linux_gfx_mode
```

◆برای بارگذاری قابلیت گرافیکی و تنظیم رزولوشن صفحه گرافیکی بوت.

```
insmod gzio
if [ x$grub_platform = xxen ]; then insmod xzio; insmod lzopio; fi
```

◆بارگذاری ماژول فشرده‌سازی .gz
◆اگر پلتفرم روی Xen باشد، ماژول‌های فشرده‌سازی اضافی نیز بارگذاری می‌شوند.

```
insmod part_gpt
insmod ext2
```

◆بارگذاری ماژول‌های مربوط به نوع پارتیشن **GPT** و فایل‌سیستم **ext2/ext3/ext4**.

```
set root='hd0,gpt2'
```

◆ست کردن ریشه سیستم‌عامل) جایی که GRUB فایل کرنل را جستجو می‌کند .(یعنی پارتیشن دوم از دیسک اول.

```
if [ x$feature_platform_search_hint = xy ]; then
search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt2 \
--hint-efi=hd0,gpt2 --hint-baremetal=ahci0,gpt2 \
۴۹□□۸□۴□-۷□۱۳-۴۵۵□-□۲۸۷-۴۸۸□۳۳۲۸۶□۳۰
else
search --no-floppy --fs-uuid --set=root \
۴۹□□۸□۴□-۷□۱۳-۴۵۵□-□۲۸۷-۴۸۸□۳۳۲۸۶□۳۰
```

```
fi
```

◆جستجوی پارتیشن بوت بر اساس) UUID کد یکتا)، که امن‌تر و دقیق‌تر از ارجاع به hd0,gpt2است.
◆بسته به تنظیمات سخت‌افزاری و وجود یا عدم وجود search hint، مسیر با راهنمای دقیق یا بدون آن تنظیم می‌شود.

```
linux /vmlinuz-5.4.0-65-generic root=/dev/mapper/vg0-root ro maybe-ubiquity
```

◆بارگذاری فایل کرنل لینوکس از مسیر /vmlinuz-5.4.0-65-generic
◆پارامتر rootمشخص می‌کند سیستم‌عامل روی کدام پارتیشن نصب شده. اینجا اشاره به LVM (/dev/mapper/vg0-root)دارد.
◆ro یعنی به صورت Read-Only بوت شود.
◆maybe-ubiquity برای نمایش نصب‌کننده‌ی Ubuntu در Live mode است.

```
initrd /initrd.img-5.4.0-65-generic
```

◆بارگذاری تصویر initrd (initial ramdisk) که برای راه‌اندازی اولیه سیستم مورد نیاز است.

این ورودی GRUB مشخص می‌کند که چطور سیستم Ubuntu را بوت کند:

- با استفاده از کرنل نسخه مشخص،
- از یک پارتیشن خاص LVM) روی(GPT ،
- همراه با ماژول‌های موردنیاز و پشتیبانی از گرافیک،
- با initrd مشخص.