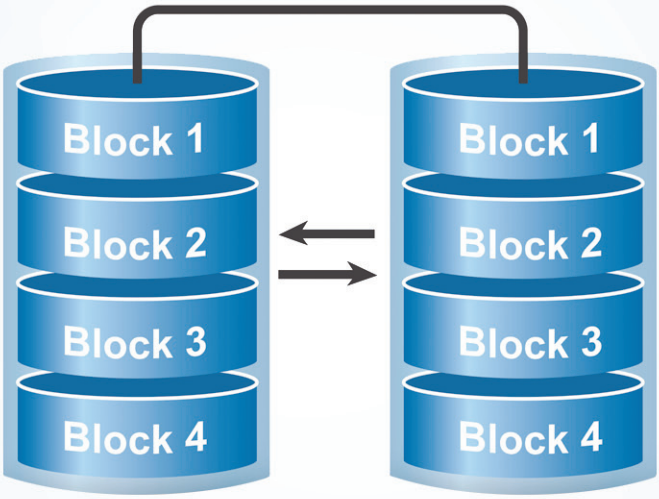


RAID
از ســـــیر تا پیـــــاز

هومن سیاری
Sayyari@ComputerNews.ir

RAID داده‌ها را در مقابل هرگونه ایراد سخت‌افزاری محافظت کرده و سرعت انتقال داده‌ها را افزایش می‌دهد.

حفظ و نگهداری اطلاعات همواره یکی از دغدغه‌های اصلی کاربران کامپیوتر بوده و هست. هر چه اهمیت اطلاعات بالاتر باشد نگرانی در جهت حفظ آنها بالاتر می‌رود. شـاید یکی از قدیمی‌ترین روش‌ها برای حفظ اطلاعات در مقابل آسـیب‌های احتمالی، گرفتن بکاپ باشد ولی در کاربردهای حساس و آنلاین مثل فعالیت‌های کامپیوتری بانک‌ها نمی‌توان از بکاپ اسـتفاده کرد چرا که اطلاعات مدام در حال تغییر هسـتند. لذا باید به دنبال راه‌حلی بود که بتواند در صورت هر گونه آسـیبی بلافاصله فعال شده و اثر آن آسـیب را بر طرف نماید. مثلا از ۲ هارددیسـک به صورت همزمان اسـتفاده کنیم و کلیه اطلاعات به صورت همزمان بر روی هر دو نوشته شوند. حال اگر یکی از آنها آسیب ببیند دیگری فعال شده و انگار نه انگار که اتفاقی اقتاده است. به روش‌هایی از این دست RAID گفته می‌شود.

RAID که مخفـف عبـارت Redundant Array of Independent Disks اسـت به معنای **آرایه افزونه‌ای از دیسـک‌های مسـتقل** اسـت. البته این ترجمه که بیشـتر در متون فارسی دیده می‌شـود جز پیچیده کردن موضوع کاربرد دیگـری ندارد! اگر راحت‌تر بگوییم در واقع RAID اسـتفاده از چند هارددیسـک به جای یک هارددیسک برای رسیدن به راندمان بالاتر، اطمینان بیشتر و یا ترکیبی از این دو مورد است.

RAID در سـال ۱۹۸۷ در دانشـگاه برکلی کالیفرنیا توسـط ۳ محقق به نام‌های David Patterson، Garth Gibson و Randy Katz تحت یک مقاله معرفی شـد. ایـن ۳ نفر نام ایـن تکنولوژی را RAID گذاشـته بودند ولـی RAID آنها سـر واژه عبـارت Redundant Array of Inexpensive Disks بود. تاکید بر واژه ارزان (Inexpensive) بـه دلیـل آن بود که تفاوت قیمت هارددیسـک‌های معمولی با هاردیسـک‌های تجاری که از لحاظ ظرفیت، پایداری و سـرعت خواندن و نوشـتن در حد بسیار بالاتری از هارددیسک‌های معمولی بودند بسیار فاحش بود. در واقع RAID می‌خواست که با استفاده از چند هارددیسک معمولی و ارزان‌قیمت به قابلیت‌های هارددیسـک‌های تجاری گران‌قیمت برسد و یا حداقل نزدیک شود. هارددیسک‌های تجاری گران‌قیمت را اصطلاحا SLED می‌نامند. SLED مخفف عبارت Single Large Expensive Disk است.

امروزه معمولا از واژه Independent به جای Inexpensive در RAID استفاده می‌کنند. البته باید اذعان داشـت که واژه Independent که معنای مسـتقل دارد چندان درسـت نیست و کمی گمراه‌کننده است چرا که در واقع این هارددیسک‌ها به هم وابسته هستند و نه مستقل!

از دیـد یک کاربر معمولی چند هارددیسـکی که RAID شـده‌اند به شـکل یک هارددیسک تنها دیده می‌شوند و فایل‌هایی که در RAID دیده می‌شوند مشخص نیست که به صورت فیزیکی در کدام هارددیسک قرار دارند.

RAID مزایـای زیادی دارد که بسـیاری را وسوسـه می‌نماید. یکـی از راه‌های دسـتیابی به این مزایا اسـتفاده از تجهیزات NAS و سیستم‌های اکسترنال USB اسـت که می‌توانند به سیستم شما متصل شـده و برخی از قابلیت‌های RAID را فراهم نمایند.

البتـه معمـولا کنترولر RAID بر روی مادربورد قرار می‌گیـرد و کاربر می‌تواند به راحتی با متصل کردن دو یا چند هارددیسـک بـه مادربورد و انجام تنظیمات لازم در بایوس مادربورد یک RAID را راه‌اندازی کند. در عین حال در صورتی که یک نسـخه حرفه‌ای از ویندوز داشته باشـید می‌توانید بدون داشتن هر گونه سخت‌افزار اضافی اقدام به راه‌اندازی RAID نرم‌افزاری نمایید.

Spanning

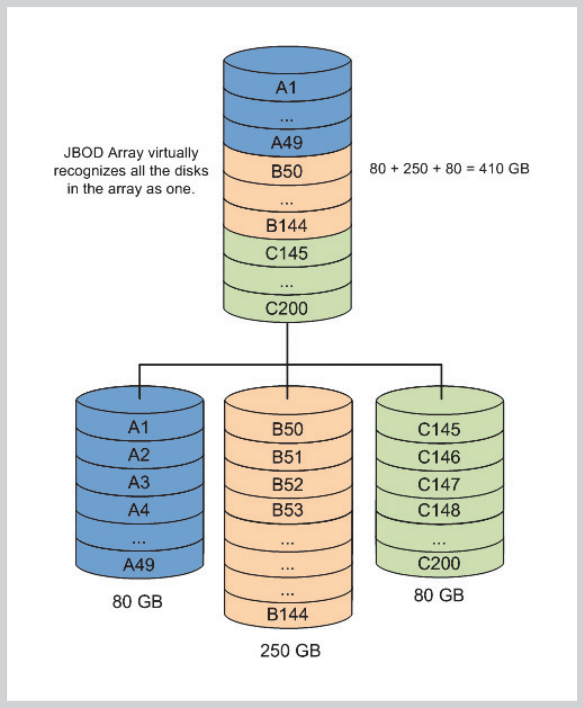
ساده‌ترین راه برای ترکیب چند هارددیسک آن است که آنها را به گونه‌ای در کنار هم قرار دهیم که همگی تشکیل یک درایو مجازی را بدهند و سیستم‌عامل فقط یک هارددیسک با ظرفیتی برابر مجموع آن هارددیسک‌ها را ببیند. این روش را اصطلاحا JBOD یا Pooling یا چسباندن هارددیسک‌ها می‌نامند.

چسباندن هارددیسک‌ها نه راندمان را افزایش می‌دهد و نه اطمینان بیشتری را به ارمغان می‌آورد بلکه تنها کاری که انجام می‌دهد آن است که موجب می‌شود که شما به جای چند هارددیسک فقط یک هارددیسک را ببینید و در نتیجه امکان پارتیشن‌بندی و مدیریت پارتیشن‌ها ساده‌تر می‌شود یعنی اگر ۲ هارددیسک ۵۰۰ و ۳۲۰ گیگابایتی داشته باشید با این روش فقط یک هارددیسک ۸۲۰ گیگابایتی خواهید دید و می‌توانید این هارددیسک را به هر روشی که دوست دارید پارتیشن بندی نمایید. یکی از مزایای این روش آن است که در صورتیکه بعد از مدتی این هارددیسک مجازی ۸۲۰ گیگابایتی پر شد فقط کافیست یک هارددیسک جدید مثلا ۵۰۰ گیگابایتی به سیستم اضافه کنید. در این صورت شما باز هم یک هارددیسک ولی با ظرفیت ۱۳۲۰ گیگابایتی خواهید دید و نیازی به ساخت پارتیشن‌های جدید نیست.

بسیاری از مادربوردها که دارای کنترولر RAID هستند این روش را به نام JBOD می‌شناسند. JBOD مخفف **J**ust a **B**unch **O**f **D**isks است که به معنای تنها گروهی از هارددیسک‌هاست.

برخی از ویندوزهای جدید مثل ویندوز۸ و ویندوز سرور خانگی۲۰۱۱ می‌توانند JBOD را به صورت نرم‌افزاری پیاده‌سازی کنند.

لازم به ذکر است که در صورتیکه یکی از هارددیسک‌ها دچار مشکل شود به احتمال زیاد کلیه اطلاعات از بین خواهند رفت. دلیل اینکه از واژه احتمال استفاده شده است آن است که برخی از کنترولرها به گونه‌ای عمل می‌کنند که در صورت از کار افتادن یک هارددیسک می‌توان اطلاعات سایر هارددیسک‌ها را ریکاوری کرد ولی اغلب آنها اینگونه نیستند.

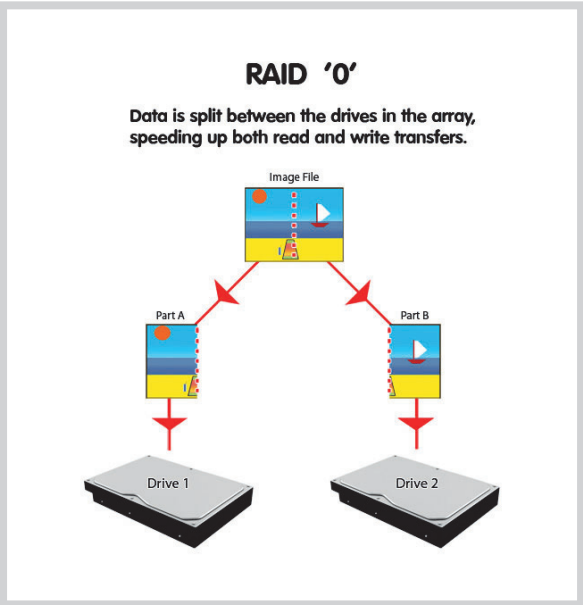


شکل ۱: نمایی شماتیک از روش Spanning

Striping

یک روش هوشمندانه‌تر برای استفاده ترکیبی از چند هارددیسک روش Striping است. به جای اینکه فقط هارددیسک‌ها را به هم بچسبانیم و وقتی هارددیسک اول پر شد به سراغ هارددیسک دوم برویم در این روش جدید از تمام هارددیسک‌ها به صورت همزمان استفاده می‌کنیم به طور مثال اگر ۲ بلاک داده داشته باشیم بلاک اول در هارددیسک اول و بلاک دوم در هارددیسک دوم به صورت همزمان ذخیره می‌شوند. در این صورت داده‌ها می‌توانند با سرعتی معادل ۲ برابر حالتی که از یک هارددیسک استفاده می‌شود، خوانده و یا نوشته شوند. سرعت خواندن و نوشتن را می‌توان با افزایش تعداد هارددیسک‌ها افزایش داد. به طور مثال می‌توان با افزایش تعداد هارددیسک‌ها به ۴ عدد، سرعت خواندن و نوشتن را تا ۴ برابر افزایش داد. تنها محدودیت در این روش حداکثر سرعت انتقال داده توسط کنترولر و یا حداکثر تعداد هارددیسک‌های قابل اتصال به مادربورد است.

به دلیل نوع تکنولوژی‌ای که در این روش استفاده می‌شود باید تمامی هارددیسک‌ها هم ظرفیت باشند؛ مثلا ۴ هارددیسک ۵۰۰ گیگابایتی داشته باشیم. در این حالت در صورتی که اندازه آنها متفاوت باشد از هر هارددیسک فقط به اندازه کوچکترین هارددیسک استفاده خواهد شد و مابقی فضا قابل استفاده نخواهد بود. به طور مثال اگر ۳ هارددیسک ۳۲۰ و ۵۰۰ و ۷۵۰ گیگابایتی داشته باشیم این سیستم آنها را به شکل ۳ هارددیسک ۳۲۰ گیگابایتی خواهد دید.



شکل ۲: نمایی از روش Striping

محدودیت دیگر این روش آن است که برای افزایش ظرفیت آن نمی‌توانید مثل روش Spanning به راحتی یک هارددیسک به آن اضافه نمایید بلکه مجبور

خواهید بود داده‌ها را از اول با اســتفاده از ســاختار جدید بازنویسی نمایید که زمان قابل توجهی نیاز دارد.

اما بزرگ‌ترین مشکل این روش در امنیت پایین آن است. در تئوری احتمال آسیب رسیدن به سیستمی که از ۳ هارددیسک استفاده می‌کند ۳ برابر سیستمی است که از یک هارددیسک استفاده می‌نماید. حال اگر به هر دلیلی یکی از هارددیسک‌های این سیستم آسیب ببیند به دلیل آنکه اطلاعات به شکل توزیع‌شده بین آنها ذخیره می‌گردد عملا کلیه اطلاعات را از دســت داده‌اید چرا که یک‌ســوم هر فایلی را از دست داده‌اید و با دو سوم بقیه نمی‌توان فایل را خواند و یا ریکاور کرد.

شاید به همین دلیل است که این روش که حرفه‌ای‌ترها به آن RAID0 می‌گویند کمتر مورد استفاده قرار می‌گیرد.

نقاط قوت

- سرعت بسیار بالا در خواندن و نوشتن اطلاعات
- استفاده بهینه از فضای ذخیره‌سازی

نقاط ضعف

- در صورتیکه یکی از هارددیسک‌ها آسیب ببیند، کلیه اطلاعات از بین می‌رود.

Mirroring

در ایــن روش هم ماننــد روش Striping دو یا چند هارددیســک به صورت یک هارددیسک مجازی دیده می‌شوند ولی با این تفاوت که یک کپی کامل از داده‌ها بر روی تک تک هارددیسک‌ها نوشته می‌شود یعنی تمامی دیسک‌ها دارای اطلاعات یکسانی هستند مثلا اگر ۳ هارددیسک داشته باشیم از هر داده ۳ نسخه وجود دارد. در این روش هم مانند روش Striping هارددیســک‌ها باید هم ظرفیت باشــند و در صورتی که اینگونه نباشد از هر هارددیسک فقط به اندازه کوچکترین هارددیسک استفاده خواهد شد. مثلا اگر ۲ هارددیسک ۵۰۰ گیگابایتی داشته باشیم در سیستم به صورت یک هارددیسک ۵۰۰ گیگابایتی دیده خواهند شد و داده‌ها بر روی هارددیسک دیگر هم کپی می‌شــود و چنانچه ۳ هاردیسک ۳۲۰ و ۵۰۰ و ۷۵۰ گیگابایتی داشته باشیم این سیستم آنها را به شکل یک هارددیسک ۳۲۰ گیگابایتی خواهد دید.

در این روش ســرعت خواندن و نوشــتن بهتر از یک هارددیســک تکی نیســت و حتی در بســیاری از مواقع کمتر از آن است چرا که فرآیند کپی داده‌ها بر روی چند هارددیسک می‌تواند وقت‌گیرتر از کپی داده‌ها بر روی یک هارددیسک باشد هرچند که این کپی به صورت همزمان انجام می‌شود.

اما تفاوت مهم با روش Striping آن اســت که چنانچه یکی از هارددیسک‌ها خراب شــود هیچ مشــکلی برای داده‌ها پیش نمی‌آید و به راحتی می‌توان آن هارددیسک را از سیســتم خارج کرد و دوباره به کار ادامه داد. حتی می‌توان هارددیســک خراب را با یک هارددیســک ســالم جایگزین کرد و از طریق برنامه کنترولر RIAD اقدام به اضافه کردن هارددیسک جدید به RAID نمود. در این فرآیند کلیه اطلاعات بر روی هارددیسک جدید هم نوشته می‌شوند. این روش را اصطلاحا RAID 1 می‌نامند.

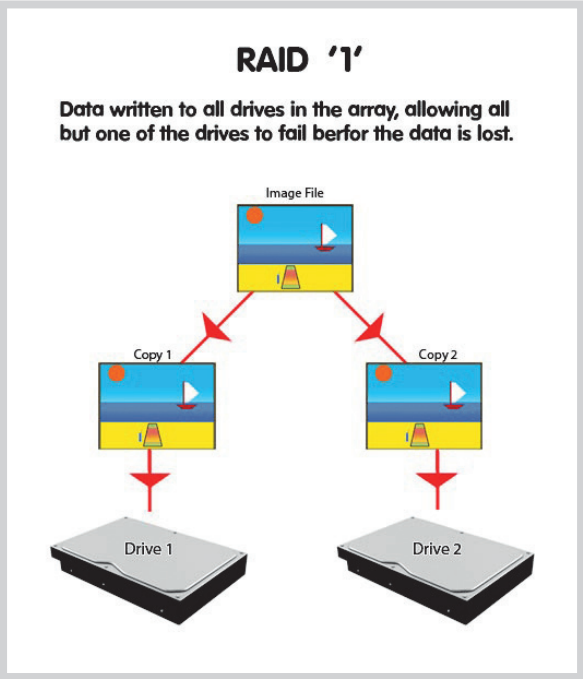
نکتــه مهمی که باید به آن توجه کنید آن اســت که RAID 1 مانند بکاپ گرفتن نیســت و جایگزینی برای آن هم محسوب نمی‌شود. مثلا اگر اشتباها فایلی را پاک کنید این فایل در همه هارددیســک‌ها پاک خواهد شد و اینگونه نیست که در یکی پاک شود و در بقیه پاک نشود تا بتوان آن را ریکاور کرد.

نقاط قوت

- در صورتیکه یکی از هارددیســک‌ها آسیب ببیند، اطلاعات از روی دیگری قابل استفاده هستند.

نقاط ضعف

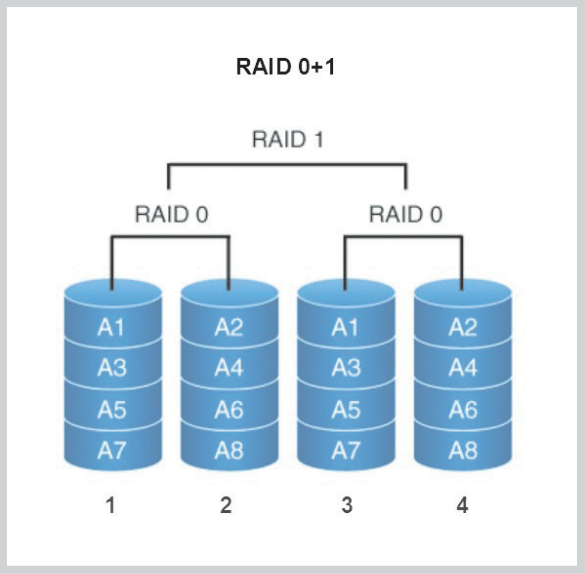
- استفاده ضعیف از فضای ذخیره‌سازی
- هزینه بالا



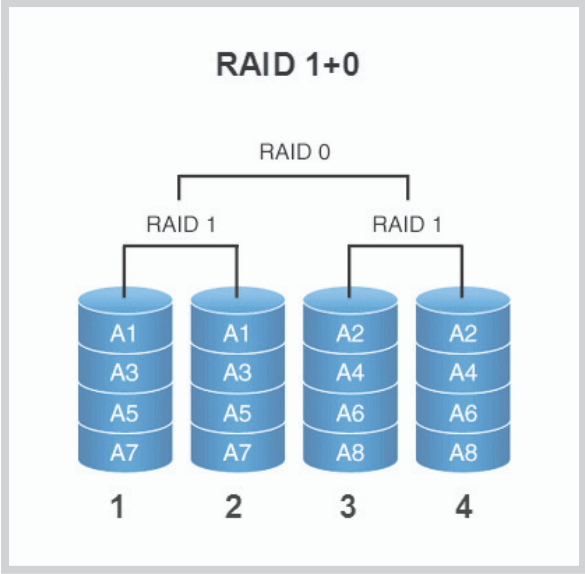
شکل ۳: نمایی از روش Mirroring

Raidهای ترکیبی

هــر چنــد RAID 0 و RAID 1 متفاوت به نظر می‌رســند اما می‌تــوان مزایای هــر دو را بــا ترکیــب آنها بدســت آورد. یکــی از روش‌های رایج آن اســت که از دو هارددیسک برای ساخت RAID 0 استفاده کنیم. سپس دو هارددیسک دیگر را برای ســاختن RAID 1 از روی ۲ هارددیسک قبلی اضافه نماییم. در این صورت اطلاعات بین هارددیســک ۱ و ۲ تقســیم می‌شــوند. حال هارددیسک ۳ یک کپی کامل از هارددیسک ۱ است و هارددیسک ۴ یک کپی کامل از هارددیسک ۲ است. در اینصورت چنانچه یکی از هارددیسک‌های ۱ یا ۲ خراب شوند، کل اطلاعات آن دو قابل اســتفاده نیست ولی در عوض از اطلاعات هارددیسک‌های ۳ و ۴ می‌توان



شکل ۴: نمایی از RAID 01



شکل ۵: نمایی از RAID 10

استفاده کرد. حتما دقت دارید که در این حالت هارددیسک‌های ۳ و۴ هم با یکدیگر بـه صورت RAID 0 عمل می‌کنند. به این روش RAID 0+1 می‌گویند. یادتان باشـد که عدد دوم همیشـه نشان دهنده RAIDای اسـت که در سطح بالاتر قرار دارد. در روش ذکر شده دو RAID 0 با هم RAID 1 شده‌اند و لذا RAID 1 در سطح بالاتر قرار دارد. با دیدن شکل متوجه قضیه خواهید شد.

RAID 0+1 را به اختصار RAID 01 هم نمایش می‌دهند.

نقاط قوت RAID 01

- سرعت خواندن بالا

نقاط ضعف RAID 01

- امنیت پایین
- استفاده از تعداد زیاد هارددیسک
- هزینه بالا

امـا روش معمول‌تری هم برای ترکیب RAID 0 و RAID 1 وجود دارد و آن بر عکس حالت قبل اسـت. در این حالت هارددیسـک‌های سـطح پایین‌تر ۲ به ۲ به صورت RAID 1 هسـتند و سـپس آنها با هم RAID 0 شـده‌اند. دلیل آنکه این روش پرکاربردتر است آن است که امنیت اطلاعات در آن بالاتر است. این روش را RAID 1+0 می‌نامند که به اختصار RAID 10 هم نمایش داده می‌شود.

نقاط قوت RAID 10

- امنیت بالاتر

نقاط ضعف RAID 10

- استفاده از تعداد زیاد هارددیسک
- هزینه بالا

اما دلیل آنکه امنیت اطلاعات در RAID10 بالاتر از RAID 01 اسـت آن است کـه در صورتیکـه مثلا هارددیسـک ۱ از کار بیفتد، در RAID 01 بخش سـمت چـپ غیـر قابل اسـتفاده خواهد بود چـرا کـه در RAID 0 از کار افتادن یکی از هارددیسـک‌ها موجـب از بین رفتـن کل اطلاعات می‌شـد در نتیجه وقتی بخش سـمت چپ نباشد RAID 1 بالایی هم معنا ندارد و کار نخواهد کرد و فقط بخش سـمت راسـت باقی می‌ماند که آن هم RAID 0 اسـت. هر چند هنوز می‌توان با سیستم کار کرد ولی به دلیل آنکه RAID 0 باقی‌مانده امنیت پایینی دارد لذا ادامه کار با ریسک بالایی همراه است.

اما اگر از RAID 10 اسـتفاده کنیم و هارددیسـک ۱ از کار بیفتد هنوز اطلاعات هارددیسک ۱ را در هارددیسک ۲ داریم. هر چند RAID 1 بخش سمت چپ کار نخواهـد کرد اما همچنان RAID 0 بالایی بـه کار خود ادامه می‌دهد و اطلاعات را تقسـیم کرده و بخشی را برای هارددیسک ۲ می‌فرستد و بخشی را برای قسمت سـمت راسـت ارسـال می‌کند پس همچنان دو RAID فعال هسـتند. لذا امنیت RAID 10 بالاتر است.

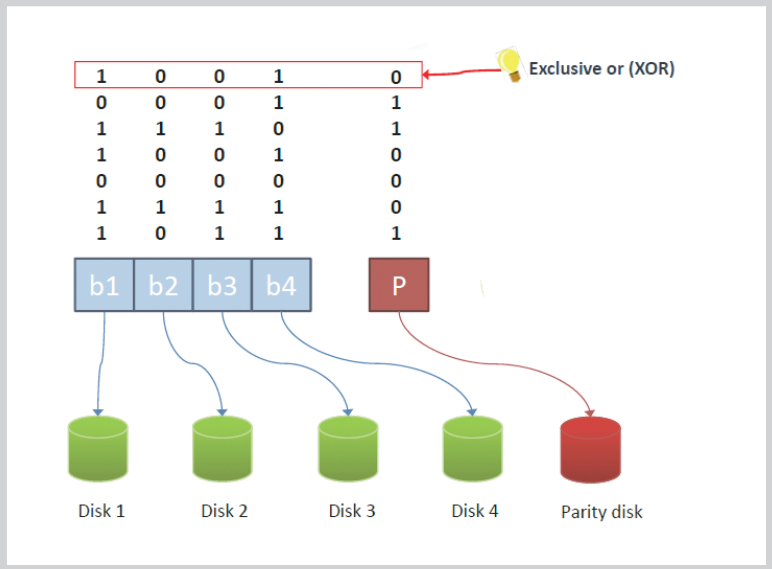
مقاومت در برابر خطا

RAID10 سـریع و قابل اطمینان اسـت امـا ظرفیت بالایـی را مصرف می‌کند. مثـلا در مورد بالا از ۴ هارددیسـک اسـتفاده کردیم تا ظرفیت ۲ هارددیسـک را بدسـت آوریم (استفاده از ۴ هارددیسک ۵۰۰ گیگابایتی منجر به ایجاد یک فضای ذخیره‌سازی یک گیگابایتی خواهد شد).

اما اگر بخواهید که کاملا ایمن باشید و در صورت بروز هر گونه مشکل سخت‌افزاری که منجر به از کار افتادن یکی از هارددیسک‌ها شود بتوانید مقاومت کنید نیاز است که از ۲ هارددیسـک دیگر هم اسـتفاده نمایید یعنی مجموعا ۶ هارددیسـک. این روش بسـیار مطمئن خواهد بود ولی به هیچ عنوان اقتصادی نیست و از جنبه‌های سخت‌افزاری، فضای اشغال‌شده و مصرف توان توجیهی ندارد.

خوشـبختانه روش‌های دیگری هم وجود دارد که بدون اضافه کردن هارددیسـک بتوان امنیت داده‌ها را تضمین کرد. این روش‌ها عمدتا نرم‌افزاری بوده و بر استفاده از سیستم‌های ریاضی پیشرفته‌تر تاکید دارند.

سیسـتمی که توسط اغلب RAIDها مورد استفاده قرار می‌گیرد توازن (Parity) نام دارد. برای اینکه بتوانیم عملکرد پریتی را توضیح دهیم به مثال زیر توجه کنید: ۲ عدد باینری ۸ بیتی 00010110 و 10010011 را در نظر بگیرید. برای ساختن



شکل ۶: نحوه کار RAID 5

پریتی از یک عملگر XOR استفاده می‌شود. این عملگر مقدار 0 را برای بیت‌های مشابه و مقدار 1 را برای بیت‌های متفاوت تولید می‌کند. نتیجه 10000101 خواهد شد.

A	0	0	0	1	0	1	1	0
B	1	0	0	1	0	0	1	1
XOR								
Parity	1	0	0	0	0	1	0	1

این عدد جدید به تنهایی معنایی ندارد ولی در صورتیکه یکی از اعداد اصلی به هر دلیلی از بین برود توسط عدد دوم و پریتی می‌توان عدد اول را بازسازی کرد. فرض کنید که عدد B از بین رفته باشد. بیت اول عدد A مقدار 0 دارد و بیت اول پریتی هم مقدار 1 است. مشخص است که بیت اول عدد B مقدار 1 داشته است چرا که اگر مقدار 0 می‌داشت پریتی آن هم 0 می‌شد. همین استدلال برای سایر بیت‌ها اعمال می‌شود تا کل عدد B ساخته شود.

ایده خوب آن است که از پریتی در کنار RAID استفاده کنیم تا امنیت اطلاعات را بالاتر ببریم و در عین حال از هارددیسک‌های کمتری هم استفاده نماییم.

یکی از روش‌ها آن است که در یک سیستم RAID 0 از یک هارددیسک جداگانه برای نگهداری پریتی استفاده نماییم. در این حالت مجموعا از ۳ هارددیسک استفاده شده است. حال اگر به هردلیلی یکی از هارددیسک‌ها خراب شود، از روی ۲ هارددیسک باقیمانده می‌توان هارددیسک سوم را مجددا بازسازی کرد. توجه داشته باشید که اگر پریتی نبود مجبور بودیم از روش‌های RAID01 یا RAID10 استفاده کنیم که به ۴ هارددیسک نیاز دارند. بنابراین یک هارددیسک کمتر استفاده شد.

حتی می‌توان بهتر از این هم عمل کرد. پریتی فقط از روی دو مقدار بدست نمی‌آید بلکه روش‌های وجود دارد که می‌توان پریتی را از روی ۳، ۵ و یا حتی ۱۰۰۰ مقدار تولید کرد. تنها کاری که باید بکنیم آن است که در صورتی که تعداد 1های متناظر زوج بود مقدار 0 و در صورتیکه فرد بود مقدار 1 را در بیت پریتی قرار دهیم.

A	0	0	0	1	0	1	1	0
B	1	0	0	1	0	0	1	1
C	0	1	1	1	1	0	0	1
XOR								
Parity	1	1	1	1	1	1	0	0

در این حالت هم می‌توان در صورتیکه یکی از اعداد A یا B یا C و یا حتی خود پریتی از بین رفت به راحتی توسط سایر داده‌ها آن را بازسازی کرد. این نتیجه جالب منجر به استفاده از یک روش بسیار کارآمد خواهد شد. ما می‌توانیم از یک دیسک اختصاصی پریتی در کنار ۳، ۵ و ... هارددیسک که به صورت RAID 0 چیده شده‌اند استفاده کرد. مثلا در یک RAID 0 که از ۵ هارددیسک استفاده می‌کند (یعنی کلیه اطلاعات ۵ تکه می‌شوند و هر کدام در یکی از هارددیسک‌ها ذخیره می‌شوند) اگر از پریتی استفاده نشود نیاز به ۵ هارددیسک دیگر برای RAID کردن آنها داریم که مجموعا ۱۰ هارددیسک می‌شود در حالیکه اگر از پریتی استفاده شود تنها به ۶ هارددیسک نیاز است.

لازم به ذکر است که سیستم فوق از لحاظ امنیت به پای RAID 1 نمی‌رسد. حالتی را در نظر بگیرید که یکی از هارددیسک‌ها خراب شده باشد و هنوز بازسازی آن تمام نشده است که هارددیسک دیگری هم خراب می‌شود! در این حالت کلیه اطلاعات شما از دست خواهد رفت چرا که نمی‌توان اطلاعات ۲ هارددیسک را از روی سایر هارددیسک‌ها بازسازی کرد. با پریتی فقط می‌توان اطلاعات یک هارددیسک را از روی بقیه بازسازی کرد و نه بیشتر.

بنابراین می‌توان گفت که RAID 0 در کنار پریتی بسیار ایمن‌تر از RAID 0 بدون پریتی است و از نظر هزینه هم بسیار مقرون به صرفه است.

RAID 2 و RAID 3 و RAID 4

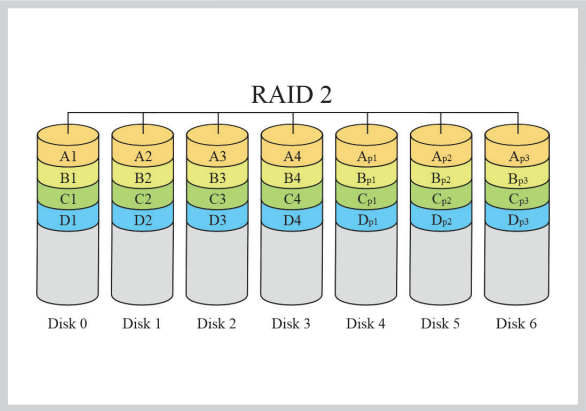
تقریبا تمامی RAIDهای بالاتر از RAID 1 از پریتی استفاده می‌کنند. البته تنها RAID 2 از روش پیشرفته‌تر دیگری به نام کد همینگ استفاده می‌کند. کد همینگ علاوه بر اینکه می‌تواند داده‌های از بین رفته را از روی سایر داده‌ها بازسازی کند می‌تواند تشخیص دهد که کدام هارددیسک دچار اشکال شده است و داده‌های ازبین‌رفته مربوط به کدام هارددیسک بوده‌اند. این قابلیت تا زمانی که RAID اختراع نشده بود تنها راه حل برای تشخیص هارددیسک خراب بود. البته در حال حاضر تمامی کنترولرهای هارددیسک می‌توانند هارددیسک خراب را به طور خودکار تشخیص دهند. با وجود این کنترولرها دیگر نیازی به استفاده از کد همینگ نیست و شاید به همین دلیل باشد که امروزه RAID 2 را نمی‌بینید.

نقاط قوت RAID 2:

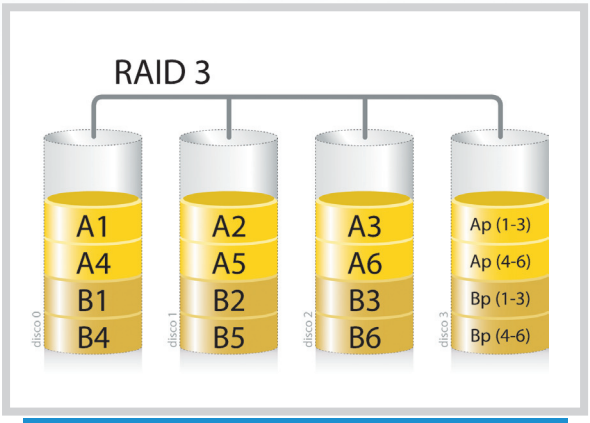
- سرعت بسیار در تشخیص خطا و تصحیح آن

نقاط ضعف RAID 2:

- استفاده ضعیف از فضای ذخیره‌سازی (تعداد زیاد هارددیسک‌های مورد نیاز)
- هزینه بالا



شکل ۷: نحوه کار RAID 2

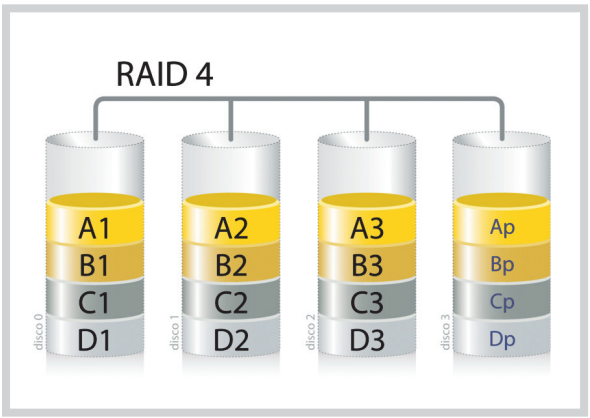


شکل ۸: نحوه کار RAID 4

RAID 3 و RAID 4 اطلاعات را بین ۲ یا چند هارددیسک به صورت RAID 0 کپی می‌کنند و از یک هارددیسک اضافه برای نگهداری پریتی هم استفاده می‌نمایند. تنها تفاوت بین RAID 3 و RAID 4 آن است که RAID 3 اطلاعات را به صورت بایت به بایت بین هارددیسک‌ها تقسیم می‌کند ولی RAID 4 از بلاک‌های بزرگتری استفاده می‌کند. نتیجه آن است که RAID 3 بسیار مناسب خواندن و نوشتن اطلاعات ترتیبی است و RAID 4 هم بسیار مناسب خواندن و نوشتن اطلاعات پراکنده است.

تفاوت دیگر آنها در نوع هارددیسک‌های مورد استفاده است. RAID 3 به شدت به ظرفیت و نوع هارددیسک‌های مورد استفاده حساس است. در این روش حتی توصیه می‌شود که برند و مدل هارددیسک‌ها دقیقا مثل هم باشد ولی در RAID 4 این حساسیت به مراتب کمتر است و می‌توان از هارددیسک‌هایی با ظرفیت یکسان ولی از برندها و مدل‌های متفاوتی استفاده کرد.

RAID 3 و RAID 4 از الگوریتم یکسانی برای ذخیره داده استفاده می‌کنند ولی نحوه پیاده‌سازی این الگوریتم همانگونه که اشاره شد کمی متفاوت است. RAID 3 و RAID 4 موجب افزایش راندمان می‌شوند و سرعت خواندن اطلاعات را بالا می‌برند ولی سرعت نوشتن اطلاعات گلوگاه این سیستم است و به سرعت دسترسی به هارددیسک پریتی و سرعت نوشتن آن وابسته است.



شکل ۹: نحوه کار RAID 4

نقاط قوت RAID 3 و RAID 4:

- سرعت بالا در خواندن اطلاعات
- استفاده بهینه از فضای ذخیره‌سازی

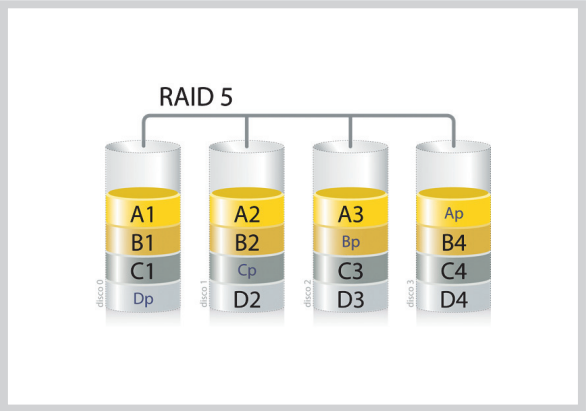
نقاط ضعف RAID 3 و RAID 4:

- پیچیدگی محاسباتی در صورت بروز مشکل برای بازگرداندن اطلاعات

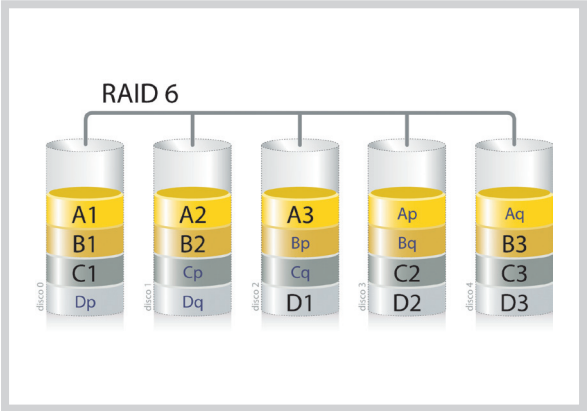
RAID 5 و RAID 6

RAID 5 یکی از پیشرفته‌ترین روش‌ها میان انواع RAIDها است. RAID 5 بر مبنای روش RAID 4 کار می‌کند با این تفاوت که به جای اینکه یک هارددیسک جداگانه برای پریتی در نظر بگیرد، پریتی‌ها را در همان هارددیسک‌ها ذخیره می‌کند و هارددیسک جداگانه‌ای به آن اختصاص نمی‌دهد. این تغییر کوچک بازدهی بالایی را به ارمغان می‌آورد. در این روش امکان نوشتن همزمان روی چند هارددیسک فراهم می‌شود مثلا در یک سیستم RAID 5 که دارای ۴ هارددیسک است می‌توان داده‌ها را بر روی هارددیسک اول کپی کرد و همزمان با آن پریتی همان داده را بر روی هارددیسک دوم آپدیت کرد. دوباره همزمان با آنها داده دیگری را بر روی هارددیسک سوم ذخیره کرد و پریتی آن را در هارددیسک چهارم آپدیت کرد. در این حالت هر ۴ هارددیسک به صورت همزمان در حال کار می‌باشند اما در روش RAID 3 یا RAID 4 در هر لحظه فقط یک هارددیسک می‌تواند کار کند چرا که همگی به هارددیسک پریتی محدود هستند و تا پریتی داده اول در هارددیسک پریتی ذخیره نشود داده دوم امکان نوشته شدن بر روی هارددیسک‌های دیگر را ندارد.

در عمل سرعت نوشتن اطلاعات در RAID 5 کمتر از RAID 0 یا RAID10 است و این موجب می‌شود که در کاربردهایی مثل دیتابیس‌ها که سرعت نوشتن اطلاعات بسیار حیاتی است کمتر مورد استفاده قرار بگیرند. اما RAID 5 از تمامی روش‌های باقیمانده سرعت بالاتری دارد و راندمان خوبی را هم در امنیت اطلاعات به ارمغان می‌آورد.



شکل ۱۰: نحوه کار RAID 5



شکل ۱۱: نحوه کار RAID 6

نقاط قوت RAID 5:

- سرعت خواندن بالا
- امنیت بالا

نقاط ضعف RAID 5:

- طراحی پیچیده مدار کنترولی

سرانجام نوبت RAID 6 است که بر مبنای RAID 5 ساخته می‌شود. در این روش حتی در صورتیکه ۲ هارددیسک به صورت همزمان خراب شوند امکان ادامه کار وجود دارد. این ویژگی با اضافه کردن یک هارددیسک پریتی و ساختن ۲ پریتی به ازای هر داده انجام می‌شود. در یک ساختار RAID 6 که از ۴ هارددیسک استفاده کرده است فضای ذخیره‌سازی به اندازه RAID 1 است اما همچنان که اندازه RAID بزرگتر می‌شود عملکرد RAID 6 معقول‌تر می‌شود. به عنوان مثال می‌توان با ۱۲ هارددیسک یک ترابایتی در RAID 6 به ظرفیت ۱۰ ترابایت رسید در حالیکه همان هارددیسک‌ها در RAID10 منجر به فضای ذخیره‌سازی ۶ ترابایتی خواهند شد.

ضمنا RAID 6 تنها RAIDی است که می‌تواند خرابی داده‌ها را تشخیص داده و تصحیح نماید. در این روش در صورتیکه داده‌ای به صورت اشتباه بر روی یکی از هارددیسک‌ها نوشته شود سیستم به طور خودکار تشخیص داده و آن را تصحیح می‌نماید. این ویژگی با استفاده از روش متفاوت محاسبه پریتی انجام می‌شود یعنی همانگونه که اشاره شد برای هر داده ۲ پریتی ساخته می‌شود اما روش ساختن این پریتی‌ها با هم متفاوت است و لذا با استفاده از ۲ پریتی متفاوت می‌توان داده اشتباه را تشخیص داد و اصلاح کرد.

لازم به ذکر است که به دلیل محاسبه همزمان پریتی‌ها به ۲ روش مختلف برای هر داده بار پردازشی RAID 6 بالاتر از سایر روش‌ها است و لذا سریع‌ترین روش نیست ولی یکی از مطمئن‌ترین روش‌هاست.

نقاط قوت RAID 6:

- سرعت خواندن بالا
- امنیت بسیار بالا

نقاط ضعف RAID 6:

- طراحی پیچیده مدار کنترولی
- پروسه زمانگیر بازگرداندن اطلاعات

جدول مقایسه‌ای انواع RAID

RAID Level	تعداد دیسک‌ها	ظرفیت	ظرفیت قابل استفاده	مقاومت در برابر خطا	راندمان در خواندن تصادفی	راندمان در نوشتن تصادفی	راندمان در خواندن ترتیبی	راندمان در نوشتن ترتیبی	هزینه
0	2,3,4,...	S*N	100%	none	★★★★	★★★★	★★★★½	★★★★	$
1	2	S*N/2	50%	★★★★	★★★	★★★	★★	★★★	$$
2	many	varies, large	~ 70-80%	★★	★★	★	★★★★	★★½	$$$$$
3	3,4,5,...	S*(N-1)	(N-1)/N	★★★	★★★	★	★★★★	★★½	$$
4	3,4,5,...	S*(N-1)	(N-1)/N	★★★	★★★★	★½	★★★	★★	$$
5	3,4,5,...	S*(N-1)	(N-1)/N	★★★	★★★★½	★★	★★★½	★★½	$$
6	4,5,6,...	S*(N-2)	(N-2)/N	★★★★½	★★★★½	★	★★★½	★★	$$$
01/10	4,6,8,...	S*N/2	50%	★★★★	★★★★½	★★★½	★★★★½	★★★½	$$$
03/30	6,8,9,10,...	S*N0*(N3-1)	(N3-1)/N3	★★★½	★★★★	★★	★★★★½	★★★	$$$$
05/50	6,8,9,10,...	S*N0*(N5-1)	(N5-1)/N5	★★★½	★★★★½	★★★	★★★★	★★★	$$$$
15/51	6,8,10,...	S*((N/2)-1)	((N/2)-1)/N	★★★★★	★★★★	★★★	★★★★	★★★	$$$$$