PAVILION

WHITE PAPER

NVMe and RAID

Table of Contents

Introduction

Redundant Arrays of Independent Disks (RAID) is a standard technology used in all modern data centers. From the largest enterprise to even some consumer devices RAID, while not a replacement for a proper backup strategy, is used to protect against the loss of data due to the failure of an individual drive.

RAID was popularized as a solution to multiple problems with legacy hard disk drives - reliability, capacity and performance. Legacy disk drives operated by magnetically storing data on platters. These platters would rotate and an actuator arm would move back and forth across the platter to read or write data.

The reliability problem with hard disk drives (HDDs) is that anything with moving parts will eventually wear out and fail. Data backups were performed on a regular basis, such as nightly, but a way to recover the data that was written in between those backup events was also needed.

Any individual hard disk drive is going to have a physical capacity limit which, particularly in enterprise organizations, may not be large enough to store a single data set. While HDDs have grown in capacity over time, it seems that the amount of that needs to be stored is growing even faster.

Finally, HDDs are limited in the rate at which they can transfer data. HDDs were connected using a bus, such as SCSI or ATA (and later SAS or SATA). While both HDDs and those busses each got faster over time, no individual HDD was able to fully saturate the bus it was attached to.

The RAID Solution

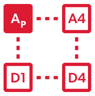
RAID is able to help solve these challenges. With RAID, multiple drives are groups together logically to provide greater capacity, performance, and/or reliability. There are many RAID levels and each provides a different level of those capacity, performance, and reliability benefits. Over the years, there have been many different RAID levels, including multi-level RAID and proprietary implementations. Some of the more common levels include:

RAID 0

Data is striped across all disks, providing the highest performance for both read and write operations, but also the greatest risk as the failure of any one drive would result in the loss of all data on the array.

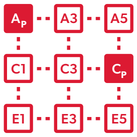
RAID 1

Mirroring. Limited to two drives, all data written to one drive is also written to the other. Provided a high level of data protection and some minor read performance improvement, but at a 50% capacity cost.



RAID 5

Striping with rotating parity. Parity is calculated on all data written and striped across all drives in the array. If a drive fails, the data can be recovered using parity, but at a significant performance penalty. Has a capacity cost equivalent to one drive in the array.



RAID 6

Striping with dual parity. Similar to RAID 5, but with a second parity function that provided the ability to survive the loss of 2 drives. Has a capacity cost equivalent to two drives in the array.

Implementation

RAID could be implemented through a variety of methods including the use of a specialized controller card in a DAS system, as a feature within an external, dual controller array, or through specialized software. Certain operating systems even include some RAID functionality.

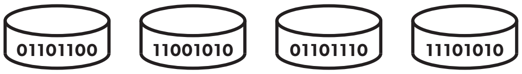
Most RAID solutions support the inclusion of a hot spare drive, which is a drive that is physically installed and fully configured in the system, but not in active use. In the event of a drive failure, the data from the failed drive can be recovered and written to the hot spare drive. Once all the recovered data has been written, the hot spare will become an active part of the array replacing the failed drive.

RAID 5 has long been one of the most common levels used by organizations, as it offers a good mix of performance, capacity, and reliability. For any data written to a RAID 5 array, the data would be broken down in blocks that were written to each drive simultaneously. A parity calculation was also performed on that data and was written the drive stripe. So a single data stripe would consist of five bytes (four data bytes and one parity byte) with each drive in the array writing one byte. Since each drive is writing only one byte, the operation could be completed faster then any single drive could write the same four blocks one at a time.

Preventing Data Loss

In the event of a drive failure, data from the lost drive could be recovered by performing a calculation on the data on the remaining drives and the parity. The only problem was that the impact of performing those calculations on a production server could be significant.

The reason it was so impactful is that RAID 5 typically uses an Exclusive OR (XOR) parity calculation. In this calculation, each bit in a byte is compared to the bits in another byte to generate the parity. For each bit in the first byte that is the same as in the second byte, a 0 is calculated. For each bit that is different, a 1 is calculated. Let's look at that in a simple four drive array.



Disk 1 data:	01101100	**Parity data:**	10100110
Disk 2 data:	11001010	**Disk 3 data:**	01101110
Parity data:	10100110	**Parity data:**	11001000

In this example, the byte that is being written to the first two drives is compared to generate a parity byte. That parity byte is then compared to the data being written to the third drive. The resulting parity byte is then written to the fourth drive. To recover data from a failed drive, the process is reversed so that the data can be recovered.

Large-Scale Example

Now, let's extrapolate this to a 15 drive array with hundreds of gigabytes of data written to each drive. In a 15 drive array, that parity calculation would be performed 14 times on each data stripe. These calculations are all being performed in the memory of the RAID controller or depending on the configuration, within the memory of the host system.

In an active production server, that is reading and writing data to the array, parity calculations to rebuild lost data typically take secondary priority to application reads and writes, but overall the entire system is slowed down. The RAID array will operate in degraded mode until the array is fully recovered. The more drives in the array and the larger those drives are, the longer the rebuild will take. It was not uncommon for large arrays to take hours, or in some cases days, to recover from a single drive failure.

With RAID 5, a second drive failure before the array is fully rebuilt will result in the loss of all data on all drives in the array. This will necessitate a full data recovery from backup.

Catastrophic Loss Scenario

Catastrophic data loss during the RAID 5 rebuild process is a significant concern. Given that one drive has failed, it is probable that the other drives in the array are similarly worn. Further, the action of rebuilding an array from parity will stress the remaining drives more than during normal operation, increasing the likelihood of a second drive failure.

RAID 6 uses a second parity calculation to provide an additional layer of security. With RAID 6, an array can survive the loss of up to 2 drives before there is catastrophic data loss requiring a recovery from backup. While RAID 6 does provide a higher level of data availability than RAID 5, it also has two drawbacks. First, there is a performance penalty for that second RAID calculation. The second penalty is that using a second parity checksum consumes an additional amount of available capacity. RAID 5 parity consumes the equivalent of one drive in any array, while RAID 6 uses the capacity of two drives. If a hot spare is used, then the capacity of a third drive is then utilized. The potential consumption of the capacity of three drives in every array within a datacenter quickly becomes significant.

For many applications, solid state drives (SSDs) provide a significant benefit over HDDs due to their significantly greater performance. The benefits of SSDs also include reliability as SSDs, with no moving parts, can have a significantly higher MTBF than HDDs with physical moving components.

> *For more information about NVMe and RAID solutions, visit our website or contact us directly.*
> *Website: http://www.pavilion.io/*
> *Email: info@pavilion.io*
> *Phone: (669) 263-6900*