Future Generation of Communication and Internet of Things (FGCIOT)

Journal homepage: http://fgciot.semnaniau.ac.ir/

PAPER TYPE: Research Paper

Compare RIAD, Replication and Erasure Code in Data Protection

Fattah Tavakkoli Nia

Department of Electrical and Computer Engineering, Electronic Branch, Islamic Azad University, Tehran, Iran

Article Info

Article History:
Received December 4, 2022
Revised January 22, 2023
Accepted March 4, 2023

Keywords:
Data Protection
Data Engineering
RAID
Erasure Code
Replica
Replication

HamidNami110@gmail.com

Abstract

Data Protection is all about prevent data lost or ability to restoring data in a certain time. It is really important to have a fault tolerance system that insure data lost would not happen. Some of these systems are RAID, Replication and Erasure Code. RIAD is a set of disks combined to represent as a logical disk. It has multiple levels that most of them copy data in some way that if a disk failed you don not loose any data. But RAID purpose is not necessarily duplicating data. Replication is a process of copying and storing data in different locations to improve data availability and accessibility across a network. It is a key component of disaster recovery. Data backup plan should be scheduled and periodically to have best restore point in case of a failure. Erasure Code is another method of Data Protection that is has some similarity to RAID. Erasure Code break data into fragments, expand and encode them with redundant data pieces and store data across a set of different locations or storage media.

Introduction

A. What is Data Protection?

Data protection is the process of safeguarding important information from corruption, compromise or loss.

The importance of data protection increases as the amount of data created and stored continues to grow at unprecedented rates. There is also little tolerance for downtime that can make it impossible to access important information.

Consequently, a large part of a data protection strategy is ensuring that data can be restored quickly after any corruption or loss. Protecting data from compromise and ensuring data privacy are other key components of data protection.

The coronavirus pandemic caused millions of employees to work from home, resulting in the need for remote data protection. Businesses must adapt to ensure they are protecting data wherever employees are, from a central data center in the office to laptops at home.

In this article, explore what data protection entails, key strategies and trends, and compliance requirements to stay in front of the many challenges of protecting critical workloads.

B. What is the purpose of data protection?

Storage technologies for protecting data include a disk or tape backup that copies designated information to a disk-based storage array or a tape cartridge. Tape-based backup is a strong option for data protection against cyber-attacks. Although access to tapes can be slow, they are portable and inherently offline when not loaded in a drive, and thus safe from threats over a network.

Organizations can use mirroring to create an exact replica of a website or files so they're available from more than one place.

Storage snapshots can automatically generate a set of pointers to information stored on tape or disk, enabling faster data recovery, while continuous data protection

Doi:

(CDP) backs up all the data in an enterprise whenever a change is made.

C. Differences between data protection, security and privacy

Although some businesses use the terms data protection, data security and data privacy, they have different purposes:

Data protection safeguards information from loss through backup and recovery.

Data security refers specifically to measures taken to protect the integrity of the data itself against manipulation and malware. It provides defense from internal and external threats.

Data privacy refers to controlling access to the data. Organizations must determine who has access to data. Understandably, a privacy breach can lead to data security issues. [1]

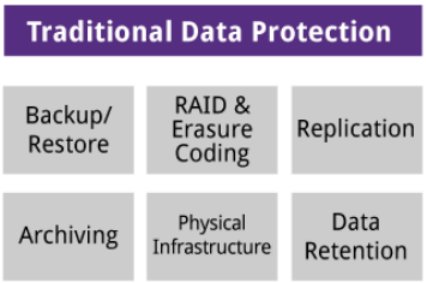

Figure 1: Traditional Data Protection items [2]

In this article, we will discuss RAID, Replication and Erasure Code:

RAID

A. What is RAID?

RAID (redundant array of independent disks) is a way of storing the same data in different places on multiple hard disks or solid-state drives (SSDs) to protect data in the case of a drive failure. There are different RAID levels, however, and not all have the goal of providing redundancy.

B. How RAID works

RAID works by placing data on multiple disks and allowing input/output (I/O) operations to overlap in a balanced way, improving performance. Because using multiple disks increases the mean time between failures, storing data redundantly also increases fault tolerance.

RAID arrays appear to the operating system (OS) as a single logical drive.

RAID employs the techniques of disk mirroring or disk striping. Mirroring will copy identical data onto more than one drive. Striping partitions help spread data over multiple disk drives. Each drive's storage space is divided into units ranging from a sector of 512 bytes up to several megabytes. The stripes of all the disks are interleaved and addressed in order. Disk mirroring and disk striping can also be combined in a RAID array.

In a single-user system where large records are stored, the stripes are typically set up to be small (512 bytes, for example) so that a single record spans all the disks and can be accessed quickly by reading all the disks at the same time.

In a multiuser system, better performance requires a stripe wide enough to hold the typical or maximum size record, enabling overlapped disk I/O across drives.

C. RAID controller

A RAID controller is a device used to manage hard disk drives in a storage array. It can be used as a level of abstraction between the OS and the physical disks, presenting groups of disks as logical units. Using a RAID controller can improve performance and help protect data in case of a crash.

A RAID controller may be hardware- or software-based. In a hardware-based RAID product, a physical controller manages the entire array. The controller can also be designed to support drive formats such as Serial Advanced Technology Attachment and Small Computer System Interface. A physical RAID controller can also be built into a server's motherboard.

With software-based RAID, the controller uses the resources of the hardware system, such as the central processor and memory. While it performs the same functions as a hardware-based RAID controller, software-based RAID controllers may not enable as much of a performance boost and can affect the performance of other applications on the server.

If a software-based RAID implementation is not compatible with a system's boot-up process and hardware-based RAID controllers are too costly, firmware, or driver-based RAID, is a potential option.

Firmware-based RAID controller chips are located on the motherboard, and all operations are performed by the central processing unit (CPU), similar to software-based RAID. However, with firmware, the RAID system is only implemented at the beginning of the boot process. Once the OS has loaded, the controller driver takes over RAID functionality. A firmware RAID controller is not as pricey as a hardware option, but it puts more strain on the computer's CPU. Firmware-based RAID is also called hardware-assisted software RAID, hybrid model RAID and fake RAID.

D. RAID levels

RAID devices use different versions, called levels. The original paper that coined the term and developed the RAID setup concept defined six levels of RAID -- 0 through

5. This numbered system enabled those in IT to differentiate RAID versions. The number of levels has since expanded and has been broken into three categories: standard, nested and nonstandard RAID levels.

I) Standard RAID levels

RAID 0. This configuration has striping but no redundancy of data. It offers the best performance, but it does not provide fault tolerance.

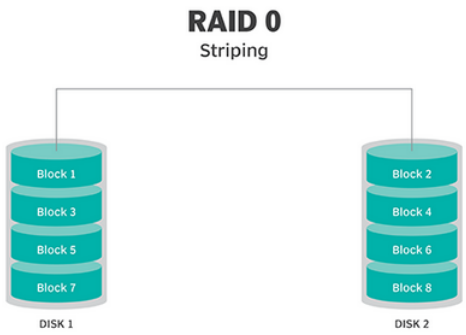

Figure 2: RAID 0.

RAID 1. Also known as *disk mirroring*, this configuration consists of at least two drives that duplicate the storage of data. There is no striping. Read performance is improved, since either disk can be read at the same time. Write performance is the same as for single disk storage.

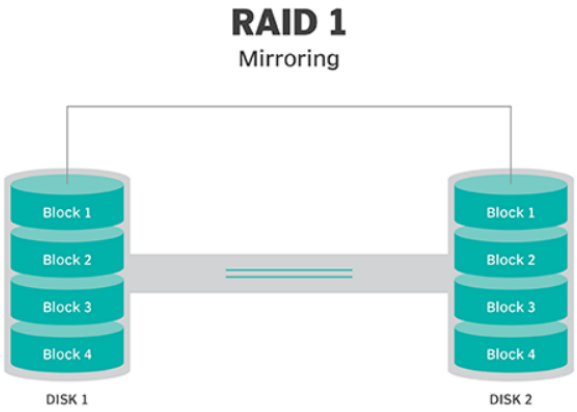

Figure 3: RAID 1.

RAID 2. This configuration uses striping across disks, with some disks storing error checking and correcting (ECC) information. RAID 2 also uses a dedicated Hamming code parity, a linear form of ECC. RAID 2 has no advantage over RAID 3 and is no longer used.

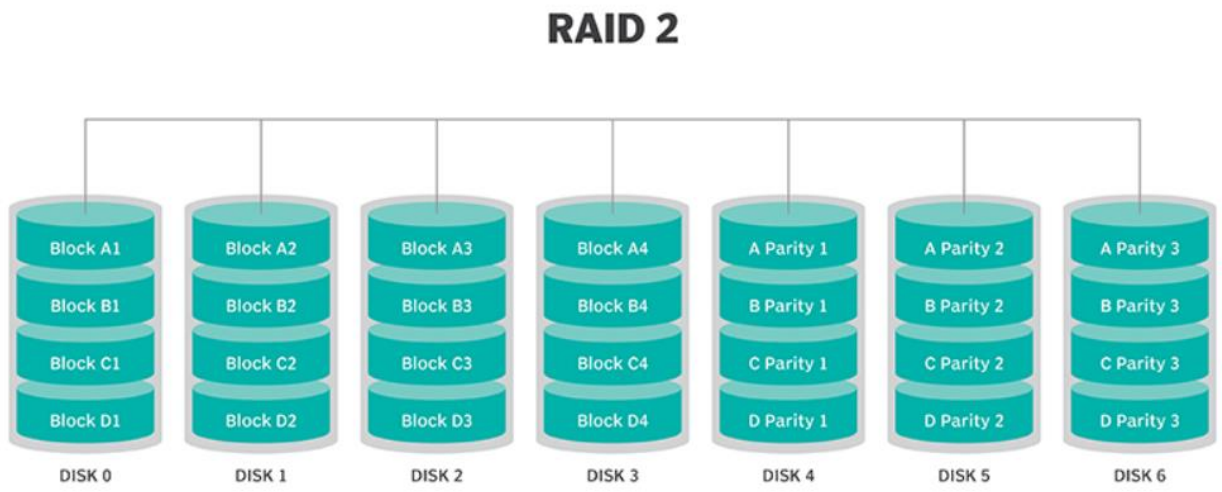

Figure 4: RAID 2.

RAID 3. This technique uses striping and dedicates one drive to storing parity information. The embedded ECC information is used to detect errors. Data recovery is accomplished by calculating the exclusive information recorded on the other drives. Because an I/O operation addresses all the drives at the same time, RAID 3 cannot overlap I/O. For this reason, RAID 3 is best for single-user systems with long record applications.

Future Generation of Communication and Internet of Things (FGCIOT)

Journal homepage: http://fgciot.semnaniau.ac.ir/

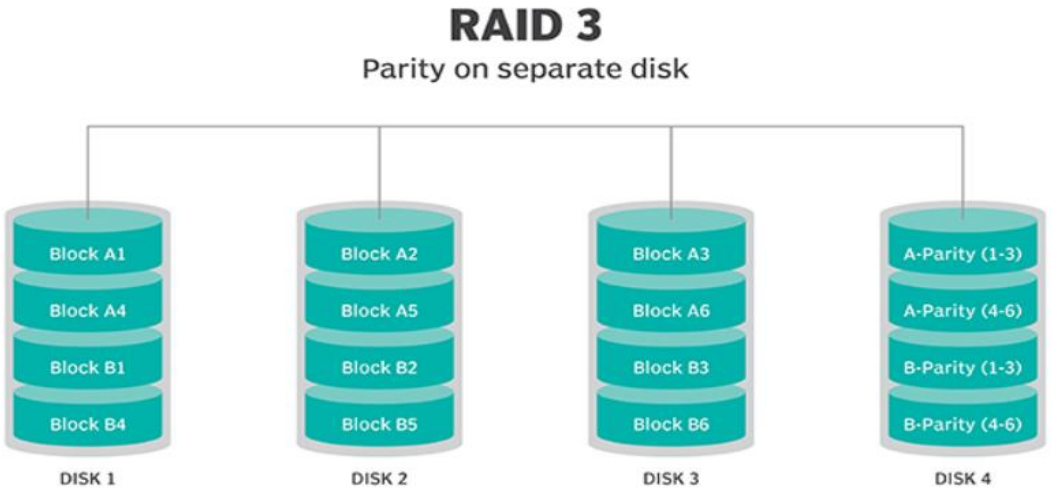

Figure 5: RAID 3.

RAID 4. This level uses large stripes, which means a user can read records from any single drive. Overlapped I/O can then be used for read operations. Because all write operations are required to update the parity drive, no I/O overlapping is possible.

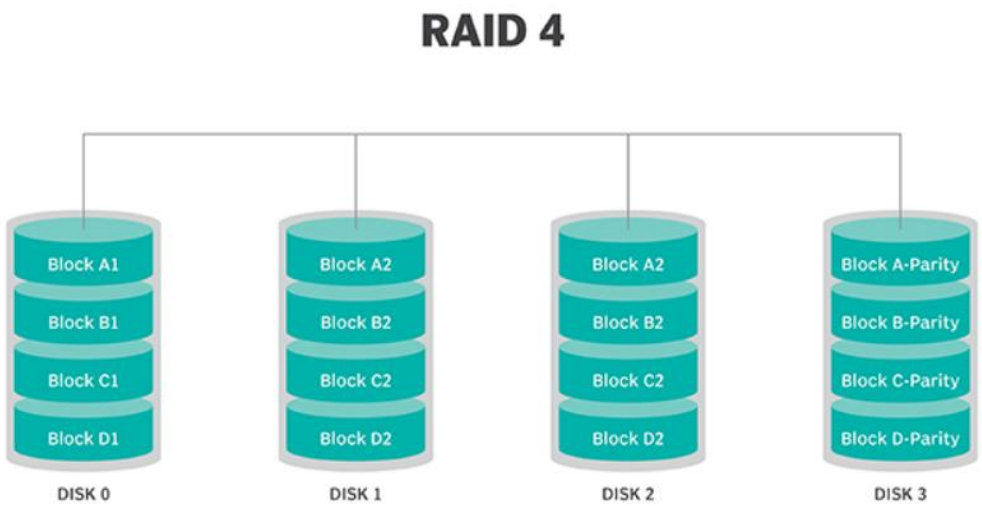

Figure 6: RAID 4

RAID 5. This level is based on parity block-level striping. The parity information is striped across each drive, enabling the array to function, even if one drive were to fail. The array's architecture enables read and write operations to span multiple drives. This results in performance better than that of a single drive, but not as high as a RAID 0 array. RAID 5 requires at least three disks, but it is often recommended to use at least five disks for performance reasons.

RAID 5 arrays are generally considered to be a poor choice for use on write-intensive systems because of the performance impact associated with writing parity data. When a disk fails, it can take a long time to rebuild a RAID 5 array.

Doi:

Future Generation of Communication and Internet of Things (FGCIOT)

Journal homepage: http://fgciot.semnaniau.ac.ir/

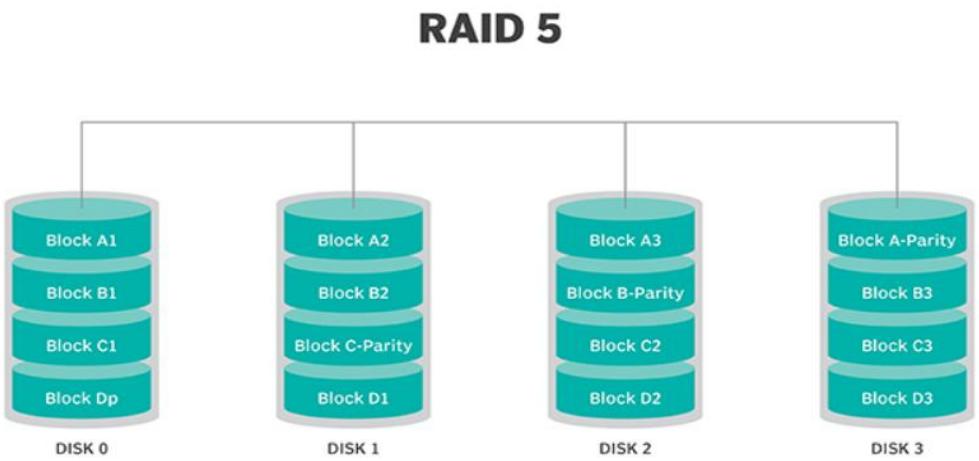

Figure 7: RAID 5.

RAID 6. This technique is similar to RAID 5, but it includes a second parity scheme distributed across the drives in the array. The use of additional parity enables the array to continue functioning, even if two disks fail simultaneously. However, this extra protection comes at a cost. RAID 6 arrays often have slower written performance than RAID 5 arrays.

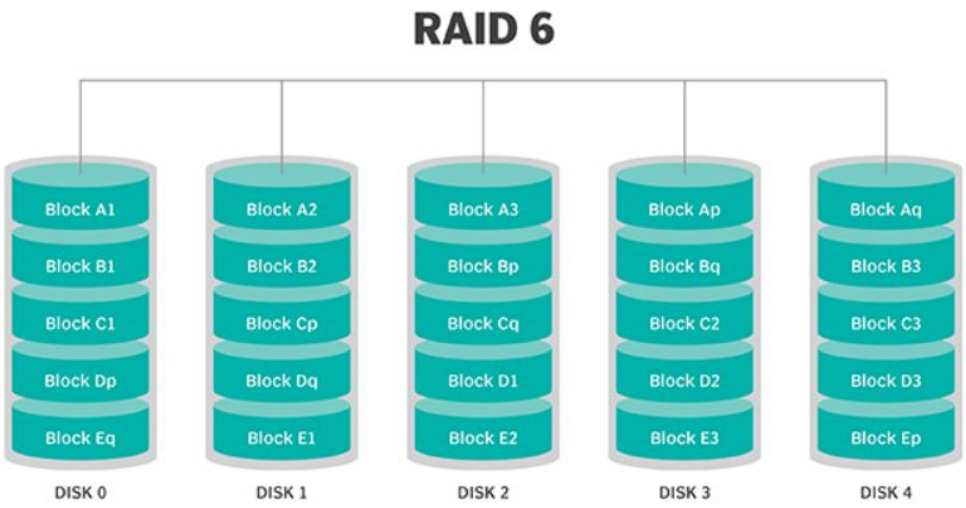

Figure 8: RAID 6.

II) Nested RAID levels

Some RAID levels that are based on a combination of RAID levels are referred to as *nested RAID.* Here are some examples of nested RAID levels.

RAID 10 (RAID 1+0). Combining RAID 1 and RAID 0, this level is often referred to as RAID 10, which offers higher performance than RAID 1, but at a much higher cost. In RAID 1+0, the data is mirrored and the mirrors are striped.

RAID 01 (RAID 0+1). RAID 0+1 is similar to RAID 1+0, except the data organization method is slightly different. Rather than creating a mirror and then striping it, RAID 0+1 creates a stripe set and then mirrors the stripe set.

RAID 03 (RAID 0+3, also known as RAID 53 or RAID 5+3). This level uses striping in RAID 0 style for RAID 3's virtual disk blocks. This offers higher performance than RAID 3, but at a higher cost.

Doi:

RAID 50 (RAID 5+0). This configuration combines RAID 5 distributed parity with RAID 0 striping to improve RAID 5 performance without reducing data protection.

III) Nonstandard RAID levels

Nonstandard RAID levels vary from standard RAID levels and are usually developed by companies or organizations for mainly proprietary use. Here are some examples.

RAID 7. A nonstandard RAID level based on RAID 3 and RAID 4 that adds caching. It includes a real-time embedded OS as a controller, caching via a high-speed bus and other characteristics of a standalone computer.

Adaptive RAID. This level enables the RAID controller to decide how to store the parity on disks. It will choose between RAID 3 and RAID 5. The choice depends on what RAID set type will perform better with the type of data being written to the disks.

Linux MD RAID 10. This level, provided by the Linux kernel, supports the creation of nested and nonstandard RAID arrays. Linux software RAID can also support the creation of standard RAID 0, RAID 1, RAID 4, RAID 5 and RAID 6 configurations.

E. When should you use RAID?

Instances where it is useful to have a RAID setup include:

When a large amount of data needs to be restored. If a drive fails and data is lost, that data can be restored quickly, because this data is also stored in other drives.

When uptime and availability are important business factors. If data needs to be restored, it can be done quickly without downtime.

When working with large files. RAID provides speed and reliability when working with large files.

When an organization needs to reduce strain on physical hardware and increase overall performance. As an example, a hardware RAID card can include additional memory to be used as a cache.

When having I/O disk issues. RAID will provide additional throughput by reading and writing data from multiple drives, instead of needing to wait for one drive to perform tasks.

When cost is a factor. The cost of a RAID array is lower than it was in the past, and lower-priced disks are used in large numbers, making it cheaper.[3]

Replication

A. What Is Data Replication?

Data replication is the process of copying and storing data in multiple locations to improve data availability and accessibility across a network. The result is a distributed environment that enables local users to access the data they need faster, and without disrupting other users.

Data replication is a key component of disaster recovery (DR) strategies, as it makes sure an accurate and up-to-date copy of data always exists in case of a system failure, cybersecurity breach, or other disaster—whether naturally occurring or through human error.

Copies of replicated data can be stored within the same system, in onsite or off-site servers, or in multiple clouds.

B. Why Is Data Replication Important?

Data replication is key to business resiliency because data drives decision-making. It feeds into and informs mission-critical processes, analytics, systems, and—ultimately—business insights. You want to ensure that it is always available and accessible to users in as close to real-time as possible. Data replication can help you achieve this.

These are just some of the many benefits of a strategic approach to data replication:

Ensure business continuity and disaster recovery (BCDR) – By copying your data and storing it across multiple machines, you are assured that an up-to-date version will always be available no matter what hardware malfunction, ransomware attack, or other disaster occurs

Improve app and data performance – By storing your data in multiple places, you can reduce latency since the data is located closer to the user or where the transaction is occurring—even if it's at the very edge of the network

Enhance analytics capabilities – When you replicate data to a shared system such as a data warehouse or to the cloud, analysts working from anywhere can collaborate on projects to power more accurate business intelligence, faster

C. What Are the Types of Data Replication?

Organizations often put in place data replication in Oracle, data replication in SQL Server, or data replication in MySQL strategies to mitigate downtime risk.

Common types of data replication include:

Snapshot replication – Like a picture, this is a single point in time replication

Transactional replication – You get a full copy of the data and are continually sent updates every time they happen, in the order they happen, in real time

Merge or heterogeneous replication – This type of replication happens when two or more data sources are combined into one singular source

D. What Is the Difference Between Synchronous and Asynchronous Replication?

I) Synchronous Replication

Data is copied to a secondary site as new data is written or updated on the primary site. Multiple sites thereby have current copies of the data, which enables rapid failover-based disaster recovery.

With synchronous replication, data is written first to the primary site array and then immediately to the secondary site array. The writes are considered completed only after the host receives acknowledgement that the write process completed on the arrays at both sites. While synchronous replication ensures little-to-no discrepancy between the data on the primary and secondary sites, the process may tax overall performance and may also be negatively impacted if the distance between the primary and secondary sites is significant.

II) Asynchronous Replication

Data is written to the primary site and then replicated periodically to a secondary site, which may occur hourly, daily, or weekly. When the secondary site has been updated, an acknowledgement is sent to the primary site.

Since data is written asynchronously, users can schedule replication at times when network performance will be least impacted. The secondary site can be used for disaster recovery with the understanding that the primary and secondary sites may not be fully synchronized.

E. What are Data Replication Techniques?

The three most popular data replication techniques are:

Full-table replication – The process of copying everything from the data storage source to another source (e.g., every existing, new, and updated row)

Key-based incremental replication – The process of scanning keys or indexes to see what's changed and then copying only what's different

Log-based incremental replication – During this process, software scans the log files of the source to determine what has changed and then makes only those changes in the source copy. [4]

Erasure Code

Erasure coding (EC) is a method of data protection in which data is broken into fragments, expanded and encoded with redundant data pieces and stored across a set of different locations or storage media.

If a drive fails or data becomes corrupted, the data can be reconstructed from the segments stored on the other drives. In this way, EC can help increase data redundancy, without the overhead or limitations that come with different implementations of RAID.

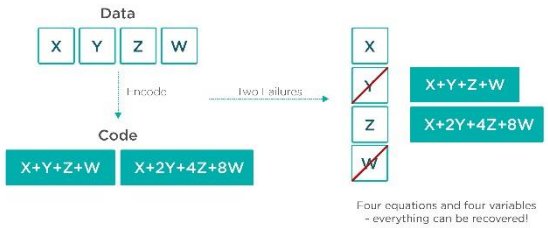

Figure 9: Erasure code calculation

A. How does erasure coding work?

Erasure coding works by splitting a unit of data, such as a file or object, into multiple fragments (data blocks) and then creating additional fragments (parity blocks) that can be used for data recovery. For each parity fragment, the EC algorithm calculates the parity's value based on the original data fragments. The data and parity fragments are stored across multiple drives to protect against data loss in case a drive fails or data becomes corrupted on one of the drives. If such an event occurs, the parity fragments can be used to rebuild the data unit without experiencing data loss.

For example, a storage system might use a 5+2 encoding configuration to distribute data across multiple physical drives. In this configuration, the EC algorithm breaks each data unit into five data fragments and then adds two parity fragments, which are calculated from the original data. Each fragment is stored on a different physical drive. As a result, the storage system must include at least seven drives.

In a 5+2 configuration, the parity data consumes 40% of raw capacity. The configuration can also tolerate up to two disk failures, whether the disks contain data fragments or parity fragments. However, EC is flexible enough to support a wide range of configurations. For example, a 17+3 encoding would split each data unit into 17 segments and then add three parity segments. Although this configuration requires at least 20 physical drives, it can support up to three simultaneous disk failures, while reducing the parity overhead to less than 18%.

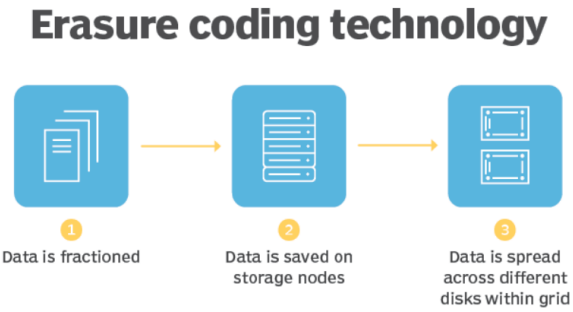

Figure 10: Erasure coding technology

Erasure coding makes it possible to protect data without having to fully replicate it because the data can

be reconstructed from parity fragments. For instance, in a simple 2+1 configuration, a data unit is split into two segments, with one parity fragment added for protection. If an application tries to retrieve data from either of the data segments and those segments are available, the operation proceeds as normal, even if the parity segment is unavailable.

However, if the first data fragment is available but the second data fragment isn't, or vice versa, data is read from the first data fragment and the parity fragment. Together these two fragments are used to reconstruct the data that was in the second fragment, making it possible to continue data operations while the disk is being rebuilt.

B. Erasure coding vs. RAID

Erasure codes, also known as forward error correction codes, were developed more than 50 years ago to help detect and correct errors in data transmissions. The technology has since been adopted to storage to help protect data in the event of drive failure or data corruption. More recently, EC has been gaining popularity for use with large object-based data sets, particularly those in the cloud. As data sets continue to grow and object storage is more widely implemented, EC is becoming an increasingly viable alternative to RAID.

I) RAID

RAID relies on two primary mechanisms for protecting data: mirroring and striping with parity. Mirroring is one of the most basic forms of data protection. When used alone, it's referred to as RAID 1. In this configuration, multiple copies of the data are stored on two or more drives. If one drive fails, the data can be retrieved from one of the other drives, without interruption to service. Mirroring is easy to implement and maintain, but it uses a large amount of storage resources, just like any form of replication.

Striping with parity, referred to as RAID 5, stripes data across multiple hard disks and adds parity blocks to protect the data. If a drive fails, the missing data can be reconstructed using the data on the other disks. However, RAID 5 can support only one disk failure at a time. For this reason, some vendors offer RAID 6 storage systems, which can handle up to two simultaneous disk fails. Different RAID configurations can also be combined, as in RAID 10, which uses disk mirroring and data striping without parity to protect data.

The various RAID configurations have been integral to data center operations for many years because the technology is well understood and has proven a reliable form of data protection for a wide range of workloads. However, RAID comes with significant challenges. For example, mirroring is inefficient when it comes to resource utilization, and striping with parity can protect against only two simultaneous disk failures at best.

Another issue with RAID is related to capacity. As disk drives become larger, it takes much more time to rebuild a drive if it should fail. Not only can this affect application performance, it can also increase the risk of losing data. For example, if a drive fails in a RAID 5 configuration, it might take days to rebuild that drive, leaving the storage array in a vulnerable position until the rebuild is complete. An incapacitated disk can also affect application performance.

II) Erasure coding

In some cases, erasure coding can be used in place of RAID to address its limitations. Erasure coding can exceed RAID 6 in terms of the number of failed drives that can be tolerated, increasing the level of fault tolerance. In a 10+6 erasure coding configuration, 16 data and polarity segments are spread across 16 drives, making it possible to handle up to six simultaneous drive failures.

Erasure coding is also much more flexible than RAID, whose configurations are fairly rigid. With EC, organizations can implement a storage system to meet their specific data protection requirements. In addition, EC can reduce the amount of time it takes to rebuild a disk that has failed, depending on the configuration and number of disks.

Despite these benefits, EC has a serious drawback: its effect on performance. Erasure coding is a processing-intensive operation. The EC algorithm must run against all data written to storage, and the data and parity segments must be written across all participating disks. If a disk fails, rebuild operations put an even greater strain on CPU resources because the data must be reconstructed on the fly. RAID configurations, whether mirroring or striping with parity, have much less of an effect on performance and can often improve it.

C. Why is erasure coding useful?

Major cloud storage services such as Amazon Simple Storage Service (S3), Microsoft Azure and Google Cloud use erasure coding extensively to protect their vast stores of data. Erasure coding has proven especially beneficial for protecting object-based storage systems, as well as distributed systems, making it well suited to cloud storage services. That said, erasure coding has also been making its way into on-premises object storage systems, such as the Dell EMC Elastic Cloud Storage (ECS) object storage platform.

Erasure coding can be useful with large quantities of data and any applications or systems that must tolerate failures, such as disk array systems, data grids, distributed storage applications, object stores and archival storage. Most of today's use cases revolve around large data sets for which RAID isn't a practical option. To support EC, the infrastructure must be able to deliver the necessary

performance, which is why its predominant use case has been with major cloud services.

Erasure coding is often recommended for storage such as backups or archive -- the types of data sets that are fairly static and not write-intensive. That said, erasure coding is finding its way into a variety of systems trying to avoid the high costs of replication. For example, many Hadoop Distributed File System (HDFS) implementations now use EC to reduce the overhead associated with storing redundant data across data nodes. In addition, object storage platforms such as Hitachi Content Platform now support erasure coding for protecting data.

D. What are the benefits of erasure coding?

Although RAID can still be a useful tool for data protection, EC offers several important benefits that should be considered when planning data storage:

Better resource utilization. Replication techniques such as RAID 1 mirroring use a high percentage of storage capacity for data copies. Erasure coding can significantly reduce storage consumption, while still protecting data. The exact amount of capacity saving will depend on the encoding configuration, but whatever it is, it will still translate to greater storage efficiency and lower storage costs.

Lower risk of data loss. When a RAID array is made up of high-capacity disks, rebuilding a failed drive can take an extremely long time, which increases the risk of data loss should another drive fail before the first one can be rebuilt. Erasure coding can handle many more simultaneous disk failures, depending on the encoding configuration, which means that there is a lower risk of data loss if a drive goes down.

Greater flexibility. RAID tends to be limited to fairly fixed configurations. Although vendors can implement proprietary RAID configurations, most RAID implementations are fairly standard. Erasure coding provides far more flexibility. Organizations can choose the data-to-parity ratio that best fits their specific workloads and storage systems.

Greater durability. Erasure coding enables an organization to configure a storage system that offers a high degree of availability and durability. For example, Amazon S3 is designed for 99.999999999% object durability across multiple Availability Zones. Unlike RAID 6, which can sustain only two simultaneous disk failures, an EC-based system can be configured to handle substantially more.

When planning their storage strategies, organizations must consider several factors, including how to protect against data loss and provide disaster recovery. Straightforward replication is one approach and RAID is another. Erasure coding is yet one more.

Each strategy comes with advantages and disadvantages. However, with the growing amount of data and continued move to object storage, EC is destined to gain momentum. Erasure coding enables organizations to meet their scalability needs and still protect their data, without incurring the high costs of full replication. Even so, no technology can flourish without adapting to industry changes, and the EC in service today could look much different five years down the road. [5]

Conclusion

Best Data Protection method differ for each projects and use cases. Traditional project usually used replication or mirroring. However, with the foundation of the Big Data era and rise of data production rate RIAD became more suitable solution because it can provide Data Protection with usage of 40 present of disk. Erasure Code also has many benefits from RIAD and now a day it is most used in corporates such as MINIO company. It has its own algorithm that reduce the overhead and limitations from RAID.

References

[1] Paul Crocetti, Senior Site Editor ,Stacey Peterson,Senior Managing Editor Kim Hefner, Managing Editor. What is data protection and why is it important. Available: https:/www.techtarget.com/searchdatabackup/definition/data-protection

[2] SNIA. What is Data Protection. Available: https://www.snia.org/education/what-is-data-protection

[3] Alexander S. Gillis, Technical Writer and Editor ,Erin Sullivan, Site Editor ,Brien Posey. RAID (redundant array of independent disks). Available: https://www.techtarget.com/searchstorage/definition/RAID

[4] COHESITY. Data Replication. Available: https://www.cohesity.com/glossary/data-replication/

[5] Robert Sheldon. erasure coding. Available: https://www.techtarget.com/searchstorage/definition/erasure-coding