Evaluate the performance of a RAID System

1st Felipe Cadavid
Sistemas Operativos y Laboratorio.
Universidad de Antioquia
Medellin, Colombia
felipe.cadavidr@udea.edu.co

2nd Marfa Alejandra Kleber
Sistemas Operativos y Laboratorio.
Universidad de Antioquia
Medellin, Colombia
marfa.kleber@udea.edu.co

3rd Jaime Alexis Herrera Ruiz
Sistemas Operativos y Laboratorio.
Universidad de Antioquia
Medellin, Colombia
jalexis.herrera@udea.edu.co

***Abstract*— This project aims to evaluate the performance of a RAID (Redundant Array of Independent Disks) system by comparing different RAID levels in terms of read/write speeds, data redundancy, and rebuild times. RAID combines multiple physical disks into a single logical unit to provide data redundancy, improved performance, or a combination of both.**

Keywords—RAID Configurations, Data Redundancy, Disk Array Performance, Performance Benchmarking, Virtual Disk Simulation

I. Introduction

In today's data-intensive environment, efficient and reliable data storage solutions are critical for organizations to manage increasing volumes of information. RAID (Redundant Array of Independent Disks) technology offers a means to combine multiple physical disks into one logical unit, optimizing redundancy and performance.

The project is driven by the necessity to enhance data storage systems to accommodate escalating data requirements effectively. By evaluating the performance and reliability of various RAID levels—specifically RAID 0, 1, and 5—this study aims to identify optimal configurations that balance data availability, system performance, and storage efficiency

The methodology entails configuring RAID 0, 1, and 5 setups using virtual disk images to simulate physical disks. Performance assessments were conducted through workload simulations with the “fio” tool, providing a detailed analysis of each configuration under different operational conditions.

II. Background or Theoretical Framework

In accordance with the project's theme, we will analyze from theory concepts related to RAID systems and their different levels. Additionally, it is necessary to have knowledge of disk I/O operations, data striping, parity calculations, and fault tolerance mechanisms, as these topics are directly related to the content covered in Operating Systems, especially in terms of storage and data management [1].

RAID (Redundant Array of Independent Disks) is a data storage system that utilizes multiple units (hard disks or SSDs), among which data is distributed or replicated. At its simplest level, a RAID combines several hard disks into a single logical unit. Thus, instead of seeing multiple different hard disks, the operating system sees only one. RAID systems are commonly used in servers and typically (though not necessarily) implemented with disk units of the same capacity [1].

For the sake of this investigation, we will talk about the most common RAID levels:

A. RAID 0 (Striping):

Combines two or more hard disks to work together, multiplying the speed of data read and write. It functions as if you had a single giant hard disk with the performance of all the combined disks. It distributes data in blocks across the different disks, with the process occurring simultaneously on all disks, significantly accelerating access to information and offering higher performance. However, it lacks data redundancy [2].

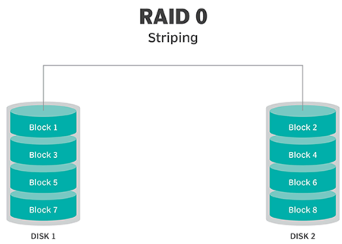

Fig. 1. RAID 0 Diagram

B. RAID 1 (mirroring)

Creates a mirror of your data on two hard disks, duplicating the information in real-time. This setup aims to provide maximum security, as if one disk fails, no data is lost; the other disk continues to function with the information intact. In addition to security, it also offers improved performance as reads are faster by accessing data from two disks simultaneously and writes perform similarly to a single disk. Lastly, the available capacity is that of the smaller disk in the array [2].

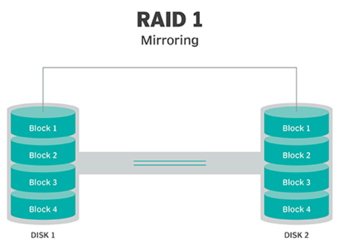

Fig. 2. RAID 1 Diagram

C. RAID 5 (striping with parity)

Combines three or more hard disks to provide a good balance of storage capacity, data protection, and performance. In this level, data is divided into blocks ("strips") and distributed across disks. It employs parity calculation for each block to recover data in the event of disk failure; the parity is distributed across the disks to avoid bottlenecks [3].

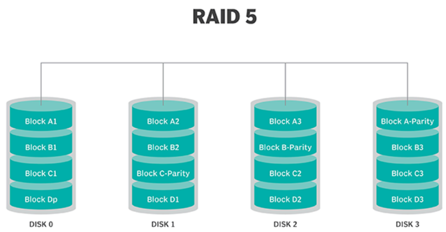

Fig. 3. RAID 5 Diagram.

Additionally, we will cover topics about file systems and the concept of loop devices which will allow us to create virtual environments so that it is possible to simulate the operations of a physical disk and for scalable testing.

III. Methodology

To comprehensively evaluate the performance of different RAID levels, virtual/logical disks were used and read/write speeds were monitored and reviewed as well as, data redundancy, and rebuild times, aiming to provide insights for informed decision-making in virtual storage system configuration. Once this information is gathered, a cross comparison was carried out using the aforementioned variables of performance.

The objectives of this evaluation are:

- Compare the read/write speeds of RAID 0, RAID 1, and RAID 5 configurations using virtual disks under varying workloads and data access patterns.
- Assess the data redundancy capabilities of RAID 1 and RAID 5 configurations using virtual disks by simulating disk failures and measuring the impact on data integrity and availability.
- Evaluate the rebuild times of RAID 5 arrays using virtual disks after simulated disk failure and recovery processes, analyzing the efficiency of data reconstruction and system restoration.
- Review the impact of different RAID configurations using virtual disks on overall system performance and resource utilization, including CPU and disk utilization metrics.
- Explore the trade-offs between performance, data redundancy, and rebuild times across various RAID levels using virtual disks to identify optimal configurations for different use cases and workload profiles.

A. *Setup and Configuration*

The following are configurations and preconditions for the configuration of the system. It is important to point out that for the sake of the testing, the same equipment was used to avoid contamination of the results caused by different system capabilities.

- Software Installation: Tools like ‘mdadm’ [4] for RAID management, qemu-utils for creating disk images, and ‘fio’ [5] for performance testing were installed.
- Virtual Disk Setup: Virtual disk images were created and configured as loop devices to simulate physical disks.
- RAID Configuration: Arrays for RAID 0, 1, and 5 were set up to test different levels of redundancy and performance.
- Performance Benchmarking: Systematic performance testing was conducted using ‘fio’ [5] to simulate realistic workloads.

B. *Data Collection and Analysis*

Data on IOPS, throughput, and latency were collected from ‘fio’ tests, analyzing how each RAID level performed under varied loads to assess their operational efficiency and suitability for different data storage requirements.

IV. Implementation

The implementation involved creating and testing various RAID configurations using virtual disks on an Ubuntu system to evaluate their performance, data redundancy, and rebuild times. By leveraging software-based RAID management tools like ‘mdadm’, we configured RAID 0, RAID 1, and RAID 5 arrays, simulating typical data access patterns with synthetic workloads using ‘fio’. This setup allowed us to benchmark the read/write speeds and observe the impact of disk failures on data redundancy and availability, providing comprehensive insights into the efficiency and reliability of each RAID level. Below is a description of the steps of implementation [7], [8], [11].

A. *Virtual Disk Setup*

Virtual disks were created to simulate physical drives, essential for configuring RAID arrays. The *‘dd’* command was used to generate empty disk images, which were then linked to loop devices using *‘losetup’*.

- Created four 500MB virtual disks using the *‘dd’* command.
- Set up loop devices to simulate physical disks.

```
# Virtual Disk Creation
# For RAID 0
sudo dd if=/dev/zero of=/mnt/raid0-disk1.img bs=1M count=500
sudo dd if=/dev/zero of=/mnt/raid0-disk2.img bs=1M count=500

# For RAID 1
sudo dd if=/dev/zero of=/mnt/raid1-disk1.img bs=1M count=500
sudo dd if=/dev/zero of=/mnt/raid1-disk2.img bs=1M count=500

# For RAID 5
sudo dd if=/dev/zero of=/mnt/raid5-disk1.img bs=1M count=500
sudo dd if=/dev/zero of=/mnt/raid5-disk2.img bs=1M count=500
sudo dd if=/dev/zero of=/mnt/raid5-disk3.img bs=1M count=500


# Loop Setup
# For RAID 0
sudo losetup /dev/loop0 /mnt/raid0-disk1.img
sudo losetup /dev/loop1 /mnt/raid0-disk2.img

# For RAID 1
sudo losetup /dev/loop2 /mnt/raid1-disk1.img
sudo losetup /dev/loop3 /mnt/raid1-disk2.img

# For RAID 5
sudo losetup /dev/loop4 /mnt/raid5-disk1.img
sudo losetup /dev/loop5 /mnt/raid5-disk2.img
sudo losetup /dev/loop6 /mnt/raid5-disk3.img
```

B. *RAID Configuration*

The RAID arrays were configured using the ‘*mdadm’* tool. RAID 0, RAID 1, and RAID 5 configurations were established, formatted with the ext4 filesystem, and mounted for accessibility.

- Configured RAID 0, RAID 1, and RAID 5 arrays using *‘mdadm’*.
- Formatted the arrays with *ext4* and mounted them.

```
sudo mdadm --create --verbose /dev/md0 --level=0 --raid-devices=2 /dev/loop0 /dev/loop
sudo mkfs.ext4 /dev/md0
sudo mount /dev/md0 /mnt/raid0

sudo mdadm --create --verbose /dev/md1 --level=1 --raid-devices=2 /dev/loop2 /dev/loop
sudo mkfs.ext4 /dev/md1
sudo mount /dev/md1 /mnt/raid1

sudo mdadm --create --verbose /dev/md2 --level=5 --raid-devices=3 /dev/loop0 /dev/loop
/dev/loop2
sudo mkfs.ext4 /dev/md2
sudo mount /dev/md2 /mnt/raid5
```

C. Workload Generation

Synthetic workloads were generated using *‘fio’* to simulate typical read/write operations. Custom job files were created for each RAID level to ensure consistent and comparable performance metrics.

- Installed ‘fio’ to generate synthetic workloads.
- Created job files to simulate read and write operations on each RAID configuration.

```
# Example job file for RAID 0

[global]
ioengine=libaio
bs=4k
size=500M
direct=1
runtime=60
time_based
numjobs=4
group_reporting

[read]
rw=randread
filename=/mnt/raid0/testfile

[write]
rw=randwrite
filename=/mnt/raid0/testfile
```

D. Performance Measurement

The *‘fio’* tool was used to run the benchmarks on each RAID configuration. The performance data, including read and write IOPS, was collected and stored in output files for analysis [6].

```
sudo fio raid0.fio > raid0_result.txt
sudo fio raid1.fio > raid1_result.txt
sudo fio raid5.fio > raid5_result.txt
```

E. Data Redundancy Testing

Disk failures were simulated in RAID 1 and RAID 5 configurations to evaluate data redundancy and the system's ability to recover. This involved failing a disk, removing it, adding a new disk, and observing the rebuild process.

```
# Simulate failure RAID 1
sudo mdadm /dev/md1 --fail /dev/loop2
sudo mdadm /dev/md1 --remove /dev/loop2

# Add a new disk
sudo losetup /dev/loop4 /mnt/raid-disk4.img
sudo mdadm /dev/md1 --add /dev/loop4


# Simulate failure RAID 5
sudo mdadm /dev/md2 --fail /dev/loop1
sudo mdadm /dev/md2 --remove /dev/loop1

# Add a new disk
sudo losetup /dev/loop4 /mnt/raid-disk4.img
sudo mdadm /dev/md2 --add /dev/loop4
```

F. Data Analysis

The performance metrics were compiled into a comprehensive table, enabling a comparative analysis of the different RAID configurations in terms of read/write speeds, data redundancy, and rebuild times.

```
echo "RAID Level,Read IOPS,Write IOPS,Rebuild Time" >
performance_data.csv
grep -e "READ" -e "WRITE" raid0_result.txt | awk '{print "RAID
0,"$3","$4",N/A"}' >> performance_data.csv
grep -e "READ" -e "WRITE" raid1_result.txt | awk '{print "RAID
1,"$3","$4","$5}' >> performance_data.csv
grep -e "READ" -e "WRITE" raid5_result.txt | awk '{print "RAID
5,"$3","$4","$5}' >> performance_data.csv
```

V. RESULTS

The evaluation of RAID 0, RAID 1, and RAID 5 configurations involved measuring their performance, assessing their data redundancy capabilities, and observing their behavior during disk failure and rebuild processes. This section presents a detailed analysis of the performance metrics obtained from synthetic workload simulations using *‘fio’*, as well as insights into the redundancy and recovery mechanisms of each RAID level. The findings highlight the trade-offs between speed, data integrity, and fault tolerance inherent to each RAID configuration, providing a comprehensive understanding of their suitability for different operational scenarios.

The following table represents the 3 virtualized RAID systems after following the configuration steps specified in the methodology:

Attribute	/dev/md0	/dev/md1	/dev/md2
Version	1.2	1.2	1.2
Raid Level	raid0	raid1	raid5
Array Size	1019904	510976	1019904
Used Dev Size	N/A	510976	509952
Raid Devices	2	2	3
Total Devices	2	2	3
Persistence	Superblock is persistent	Superblock is persistent	Superblock is persistent
State	clean	clean	clean

Active Devices	2	2	3
Working Devices	2	2	3
Failed Devices	0	0	0
Spare Devices	0	0	0
Layout	-unknown-	N/A	left-symmetric
Chunk Size	512K	N/A	512K
Consistency Policy	none	resync	resync
Events	0	17	18
Devices			
Number 0	7:0 (active sync /dev/loop0)	7:2 (active sync /dev/loop2)	7:4 (active sync /dev/loop4)
Number 1	7:1 (active sync /dev/loop1)	7:3 (active sync /dev/loop3)	7:5 (active sync /dev/loop5)
Number 2	N/A	N/A	7:6 (active sync /dev/loop6)

Fig. 4. RAID Systems Details

A. *Performance Metrics*

- RAID 0: This system exhibited a read IOPS of 82.9k and a read throughput of 340 MB/s. The high read performance is due to the striping mechanism of RAID 0, which allows data to be read from multiple disks simultaneously. Also, it achieved a write IOPS of 81.0k and a write throughput of 332 MB/s. Similar to the read performance, the striping of data across multiple disks allows for high write speeds. However, RAID 0 offers no redundancy, so any disk failure would result in data loss.

- RAID 1: It showed a read IOPS of 62.1k and a read throughput of 254 MB/s. Although the read performance is lower compared to RAID 0 and RAID 5, RAID 1 still provides decent read speeds due to its mirroring, which allows data to be read from either of the mirrored disks. On the other hand, it achieved a write IOPS of 43.7k and a write throughput of 179 MB/s. The write performance is impacted by the need to write data to both disks in the mirror, ensuring data redundancy. RAID 1 is ideal for environments where data reliability and redundancy are critical, despite the moderate write performance.

- RAID 5: This system had the highest read performance with a read IOPS of 95.1k and a read throughput of 389 MB/s. The striping with parity allows for efficient read operations, making RAID 5 suitable for read-intensive workloads. Moreover, it had a write IOPS of 26.4k and a write throughput of 108 MB/s. The need to calculate and write parity information impacts the write performance, making it lower than RAID 0 and RAID 1. Despite this, RAID 5 offers a good balance of performance and redundancy.

RAID Level	Read IOPS	Write IOPS	Read Throughput (MB/s)	Write Throughput (MB/s)
RAID 0	82.9k	81.0k	340 MB/s	332 MB/s
RAID 1	62.1k	43.7k	254 MB/s	179 MB/s
RAID 5	95.1k	26.4k	389 MB/s	108 MB/s

Fig. 5. RAID I/O results

B. *Rebuild Process Observations*

- RAID 1: Upon failing and removing one disk from the RAID 1 array, the array continued to operate in a degraded state with the remaining disk.Then, after adding a new disk to the array, the RAID 1 array successfully rebuilt, restoring full redundancy. The array remained accessible throughout the rebuild process, ensuring continuous data availability. However, the rebuild process impacted the overall performance, but the array remained functional, highlighting RAID 1's resilience and suitability for environments where data redundancy and availability are paramount.

- RAID 5: Upon failing and removing one disk from the RAID 5 array, the array continued to function in a degraded state with the remaining disks, leveraging the parity information. After adding a new disk to the array, the RAID 5 array successfully rebuilt, restoring full redundancy. The array remained accessible throughout the rebuild process. Nevertheless, the rebuild process impacted write performance due to the need to recalculate parity information. However, read performance was less affected, demonstrating RAID 5's ability to balance performance and redundancy, especially for read-heavy workloads.

VI. CONCLUSIONS

The evaluation of RAID 0, RAID 1, and RAID 5 configurations highlighted the trade-offs between performance and redundancy:

- RAID 0: Provides the highest read and write performance due to data striping but lacks any form of redundancy. It is best suited for applications requiring high-speed data access without the need for data protection.

- RAID 1: Offers moderate read and write performance with excellent data redundancy. The mirroring of data ensures that the array can withstand a single disk failure without data loss, making it ideal for environments where data integrity and availability are critical.

- RAID 5: Delivers the best read performance among the tested configurations, with a balanced approach to redundancy. While write performance is impacted due to parity calculations, RAID 5 is suitable for applications that require high read speeds and can tolerate slower write speeds, offering a compromise between performance and data protection.

The results demonstrate that the choice of RAID configuration should be based on specific workload requirements, balancing the need for performance and data redundancy to meet the operational demands of the environment.

References

[1] D. A. Patterson, G. Gibson, and R. H. Katz, "A Case for Redundant Arrays of Inexpensive Disks (RAID)," in Proc. SIGMOD Conf., 1988.

[2] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson, "RAID: High-Performance, Reliable Secondary Storage," ACM Computing Surveys (CSUR), vol. 26, no. 2, pp. 145-185, June 1994.

[3] A. Leventhal, "Flash Storage Memory," Communications of the ACM, vol. 51, no. 7, pp. 47-51, July 2008.

[4] Linux Kernel, "A guide to mdadm," Linux Kernel RAID Wiki. [Online]. Available: https://raid.wiki.kernel.org/index.php/A_guide_to_mdadm. [Accessed: June 9, 2024].

[5] J. Axboe, "Fio documentation," Read the Docs. [Online]. Available: https://fio.readthedocs.io/en/latest/fio_doc.html. [Accessed: June 9, 2024].

[6] Arcserve, "Understanding RAID Performance at Various Levels," Arcserve Blog. [Online]. Available: https://www.arcserve.com/blog/understanding-raid-performance-various-levels. [Accessed: May 02, 2024].

[7] ComputerNetworkingNotes, "How to Configure RAID in Linux Step by Step Guide," ComputerNetworkingNotes. [Online]. Available: https://www.computernetworkingnotes.com/linux-tutorials/how-to-configure-raid-in-linux-step-by-step-guide.html. [Accessed: March 16, 2024].

[8] LinuxBabe, "How to Set Up Software RAID 1 on an Existing Linux Distribution," LinuxBabe. [Online]. Available: https://www.linuxbabe.com/linux-server/linux-software-raid-1-setup. [Accessed: May 02, 2024].

[9] Intel, "Definición de volúmenes RAID para la tecnología Intel® de," Intel Support. [Online]. Available: https://www.intel.la/content/www/xl/es/support/articles/000005867/technologies.html. [Accessed: March 17, 2024].

[10] PCWorld, "Understanding RAID: What it is and how to use it effectively," PCWorld. [Online]. Available: https://www.pcworld.com/article/512178/raid-made-easy.html. [Accessed: May 02, 2024].

[11] T. Perumal, "RAID Tutorial," University of Cape Town, 2004. [Online]. Available: https://pubs.cs.uct.ac.za/id/eprint/131/1/perumal2004_RAIDTutorial.pdf. [Accessed: May 02, 2024].