Jurnal Mantik, 5 (2) (2021) 1708-1713
Published by:Institute of Computer Science (IOCS)
Jurnal Mantik
Journal homepage: www.iocscience.org/ejournal/index.php/mantik/index

Single-Board Computer For Affordable Personal Data Storage Server

Andika Candra Jaya

[1] Faculty of Computing, President University, Cikarang, Indonesia

E-mail: andika.candra@president.ac.id

ARTICLE INFO | **ABSTRACT**

Article history:
Received: 01 September 2021
Revised: 10 October 2021
Accepted: 01 November 2021

Keywords:
Raspberry Pi, Network Attached Storage, Personal cloud

The growing concerns of cloud storage privacy would necessitate local network storage. We are proposing to build do-it-yourself Data Storage server and test its cost-effectiveness. We are comparing homebrew data storage server against pre-built Network-Attached Storage and Cloud Storage providers. We conclude that using homebrew network storage server is slightly more cost effective albeit not as user-friendly to set up. Our recommendation is for user to build one and we provided the guide)

1. Introduction

With growing concerns of privacy, there has been issues about data stored in cloud. Particularly privacy and integrity. While local network storage has been primarily the domain of corporations, individuals began to see such network storage as alternative to cloud-based storage [1]. However, the inherent economical issues would leave individual users relying on the simplicity of cloud-based storage.

Instead of a cloud-based storage (in which one would have to routinely pay for rented data storage space), we develop our own local cloud-based storage scheme. Of course, we are not suggesting completely abandoning cloud storage services; we suggest to augment it and leave the cloud storage to store the data which may mislead any potential adversary. One way to achieve this is by building network storage system locally. Some use cases for local network storage are: (1) Home-based IoT to avoid surrendering data to outsiders [2], (2) Improving workflow where data sharing is frequent [3][4], (3) Small and medium enterprises [3][5], (4) Efficient Data Storage [6].

Thus we came up with the necessity to build our own storage scheme. The requirement is to build a simple and easy to replicate using existing off-the-shelf hardware. Thus, we are using SATA-based drives for the sake of affordability.We are also analyzing the monetary cost and concerns of running such a system. The cost will be compared on per-TB basis to ensure level comparison across systems. Because we are comparing it against cloud storage, we need to consider the recurring cost of rented cloud storage. When considering single-board computers for the core of our network storage, care should be given to ensure SATA compatibility because we are dealing with consumer or prosumer-grade hardware. Assuming we use Raspberry Pi, given the ubiquity [7], as the brain of our storage, we need some form of SATA add-on cards. Single-board computers are chosen because they are relatively power-eficient. Raspberry Pi can be powered by 3 A from USB or less for older generations and they can share power with add-on cards if necessary [8].

Potential bottleneck within the Raspberry Pi has been identified: USB interface, medium write speed, network connection speed. Using USB 2.0 and Raspberry Pi 3B, one can easily get away with using HDD due to the Pi's ethernet speed at 100 Mbit/sec (this will not clog the SATA nor the HDD bandwidth). Even on Pi 4 with Gigabit Ethernet at 1 Gbit/sec, the bottleneck of HDD was not as noticable given that HDD typical speed of roughly 100 MB/sec [8]. Once we exceed gigabit ethernet, however, the drive speed started to became a bottleneck. Hence, for the time being, we can safely use hard disk drive.

There are also similar work by Maguire in 2018 in which it was built using Raspberry Pi Zero which was a micro-controller in contrast to single-board computer. The key concern is that the performance of

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

Raspberry Pi Zero and throughput of the connection. Pi Zero tends to max out at 10 Mbps while Raspberry Pi 3b can run at 100 Mbps. The work was comparing Pi Zero versus Pi 2 [9].

2. Method

We are proposing to build the NAS, then address the common concerns regarding the do-it-yourself storage systems. The build was based on Raspberry Pi 3B and RockPi SATA HAT with dual drive to assess our software integration. As mentioned, we do not concern over transmission speed which had been bottlenecked in Raspberry Pi's 100 Mbit ethernet and USB 2.0 interface. Given that RockPi SATA HAT is built with Raspberry Pi 4 and USB 3.0 in mind, using USB 2.0 type A of Raspi 3 is plausible at significantly reduced speed (10 MBps for Pi 3-based versus 100 MBps for Pi 4-based). Additionally, we also need a router to allow easier administration and locating the NAS.

As to the Raspberry operating system to run on top of the hardware, we are using Raspbian OS, the one favored by Raspberry Pi foundation as its "default" operating system. Raspbian OS is Debian-based Linux and had been validated to run with Raspberry Pi hardware by default. We are using headless installation because there is web interface for that, and we can keep our Raspberry Pi operate faster. We need install RockPi SATA HAT driver as well to allow the OS to recognize SATA drives later. After which we will need to run mdadm utility such that it will be able to manage multiple drives in RAID mode.

Software-wise, one can build it using OpenMediaVault due to its substantial throughput and better web interface allowing users to access from any client in the network [10]. With Raspbian, we can use OpenMediaVault because it is built around Debian OS and Raspbian OS is a further branch of Debian. OpenMediaVault is also having docker feature, allowing for future capability expansion by adding modular softwares without interfering with the OpenMediaVault core or Raspbian OS core. After installation, we need to take note what is the address of the OMV. Usually in the router, it had been named "raspberry". This means we can quickly locate the NAS and start accessing it.

From the build, we came up with common concerns in which we shall address in the Results and Discussion section.

3. Results and Discussion

There are five concerns around do-it-yourself storage system (or any other do-it yourself creations): set up difficulty, security, cost, appearance, and maintenance. We will discuss such concerns in the following sub-sections.

3.1 Set-up concerns

Primary problem is the difficulty of setting up due to its rather technical nature unlike pre-built NAS or cloud storage. While indeed there are quite a handful of instructions on the Internet, many users still find setting up a NAS difficult and prefer the simplicity given by cloud storage systems. This is one upside of cloud storage: they make operations by individuals as simple as drag and drop. However, the simplicity comes at a price of privacy in which users surrender their data to the cloud provider which can remove contents "in violation to terms of services" and companies have been known to retroactively impose new terms on existing contents. Similarly, pre-built NAS is also provides similar benefit of simplicity at a cost of right to repair and cost.

3.2 Security

Another issue of concern is security. This stems from assumption that the NAS would be exposed to the public internet. Concerning CIA triad, we need to consider the security in form of Confidentiality, Integrity, and Availability. Confidentiality and Integrity aspects can be dealt with together through limiting access only to authorized personnel. Using OpenMediaVault as reference model, we can build some form of access control list and only open a limited amount of services such as NFS, FTP, and SMB/CIFS. The access control list takes form of whitelist in which only pre-allowed users are allowed to access the NAS. Such features are already available by default on OpenMediaVault. On the other hand, availability would need to be considered in such a way that data inside the storage can be accessed virtually at all cost. One way to achieve this is to provide some form of redundancy. The problem is that it will need additional instruction to allow the SATA to also do RAID system [11]. Using quadruple drive set up, the preferred system is RAID 5 leaving us with 2 or 3 active drives plus 1 parity drive with up to 1 hot spare in case of emergency. Dual-drive system would only benefit from either JBOD or RAID 1. RAID 1, while offering maximum protection, halves usable capacity. JBOD, on the other hand, would offer limited protection from data loss (if one drive is, the other drive still works and we only lose the data in the lost drive).

Users would also need to weigh in what they need: redundancy or raw storage. If users choose redundancy, then RAID 1 or 01 is by large the most secure because one can mirror the entire stripe of data and this can withstand up to 50\% drive failure. On the other hand, if users need raw capacity storage, then JBOD is the recommended option because the total capacity is the sum of all disk capacity. RAID 0 is not recommended because any lost drive means the entire array is broken completely.

One may be tempted to use wireless networking because the convenience of less cable. This is not recommended due to interference, risk of unauthorized access attempt, and inconvenience of re-locating the NAS every time the NAS is shut down for maintenance. Thus, one should always try to use wired connection for their NAS with a wired router. Given that wireless routers also have several Ethernet ports, one can use such port to connect the NAS to the router then use the wireless to connect additional users as denoted in Figure 1. Routers can be configured to only allow specific IP or Mac address to connect to other specified address, further protecting the NAS.

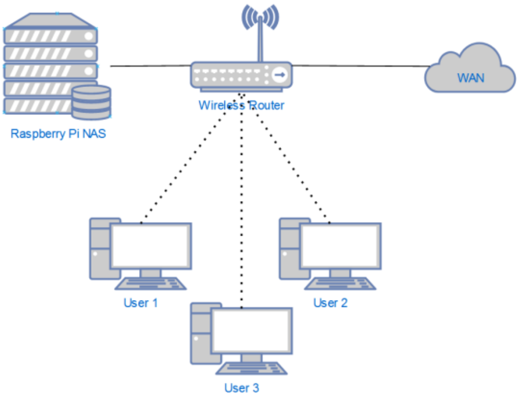

Figure. 1. Preferred connection strategy (WAN can be omitted for local storage not connected to the outside network).

Figure 1 shows three users and Raspberry Pi NAS connected to a router in which the users within the network are connected wirelessly with the router and the NAS are wired to each other. This approach would allow the users to conveniently connect to the NAS while offering the NAS protection from interference.

3.3 Cost

Table 1 shows the typical pricing of cloud storages, excluding packages for corporates, lifetime payment, and family packages. We also assume the yearly package to be more economical than the monthly ones. Thus, we use yearly package unless they only offer monthly.

TABLE 1
CLOUD STORAGE PRICE COMPARISON [14]

Vendor	List Price	Capacity	Monthly price per TB
Google	USD 100/year	2 TB	USD 4.17
Sync	USD 20/month	6 TB	USD 3.33
pCloud	USD 95.88/year	2 TB	USD 3.85
IceDrive	USD 180/year	5 TB	USD 3.00
Mega.io	EUR 299/year	16 TB	USD 1.83

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

Microsoft 365	USD 69.99/year	1 TB	USD 5.83
Dropbox	USD 199/year	3 TB	USD 5.56

Cost issues are fairly on Raspberry Pi's side: SBC and SATA HAT together costs about 150 USD, add in quadruple 3.5" tray and associated cables, we can easily expect the cost to be 190 USD per system excluding HDDs. One can cut costs by using dual-SATA HAT and older Pi (which together does not exceed 100 USD). Compared to pre-built NAS storage which could start at 250 USD for 2-bay, 350 for 4-bay models, the pricing looks appealing. Then let us consider comparing quad HDD for equivalent cloud-based storage.

Our question then evolves to how long until we break even from cloud against Pi-based NAS. Quadruple 8 TB will yield 32 TB with 24 TB usable spaces, let us consider making 21 TB to account for operating system overhead. At 300 USD per HDD for NAS-grade ones, total cost for 21 TB is 1400 USD against 460 to 1470 USD per year for 21 TB. In many cases, by third or fourth year, the rented cloud storage started to break even against home-brew NAS.

With the advent of Shingled Magnetic Recording drives, we can consider using those drives to cut costs for the benefit of NAS. While the weakness of random write performance should be acknowledged, but using such drives in mostly read operations is not cause for concern. Using RAID array, we can slightly increase the performance of the array [15].

3.4 Appearance

Some may found the look of SBC-based NAS rather disagreeable, due to ill-organized cables and visible circuit boards. This could be alleviated by building the system with some sort of case and fan which further adds to the cost. On the other hand, open look would maximize cooling performance because hot air could be dissipated away passively through radiation.

We need to also keep in mind that the pre-built NAS has more simplified look. We have options either to buy a fitting case or to 3D print a case based on what we need. Our recommendation here is to 3D print the case to accommodate future expansion such as adding a router functionality or scaling the NAS up.

3.5 Maintenance

Maintenance of the Raspberry Pi NAS would require users to routinely re-format the microSD card and repeat the entire software installation process. While this is inconvenient to many people, the frequency of maintenance can be reduced throught the means of shutting down when not in use and only use it as NAS with no additional functionality.

We will conservatively assume 100,000 hours of mean time between failure (MTBF) per drive. We are also able to compute the annualized failure rate (AFR) which are a parameter of the chances a drive to fail in a given year. Through our conservative assumption and equation(1), we can calculate that the chance of a drive failing in a given year is about 8.4%. From this number, we can estimate as well how often our drives need to be replaced depending on risk tolerance.

$$\text{AFR} = 1 - e^{\frac{-8766}{\text{MTBF}}} \quad (1)$$

At 8.4% failure rate per year per drive, and quadruple HDD setup, we conclude that there is a 30% chance of a drive failure in a year. As HDD prices tends to decline, we can economically replace drives as we need. In the future, when solid state drives' prices are competitive to HDD, replacing entire HDD array in favor of SSD. We can as well prolong the life of the HDD slightly by generalizing equation (1) by replacing the number 8766 (the number of hours in a year) into the annual active hours (how many hours in a given year the drive is on). Thus, we can calculate the longevity of our NAS and plan the upgrade accordingly. In limited circumstances, we can defer upgrading in favor of replacing on failure-basis. This is not recommended.When the time comes to replace the entire HDD array, the suggested plan would involve building a second NAS and backing up to the second NAS. That way, we can maximize the longevity of the storage medium we are using.

Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

4. Conclusion

To summarize, we are suggesting users to build their own NAS using Raspberry Pi and configure it due to cost and privacy concerns. While configuring it could take time, users can be assured that the data is theirs and they know how to fix certain troubles.The future for this project is to augment the NAS with sneakernet, improved scalability, and simplicity for complete beginner to tech. In internet-constrained household, NAS is the preferred method because of no internet cost. For users accustomed to plug and play external hard drives, we can suggest adding a router inside the system and have 8 port switch with rear-facing WAN port facing the router. Said router then plugs to the NAS.We can also suggest to add scalability using Kubernetes and added Raspberry Pi as the master node and work nodes. The configuration is one master node without added drives and all other Raspberry Pi act as worker nodes with all the drives shared between them (3 drives + 1 parity per node). This should be further reinforced with nested RAID array between nodes. The communication with those outside the local network could be handled over through sneakernet. Using properly encrypted network is also accepted, but this will open vulnerability of someone on the other side accessing what should not be accessed. Thus, moving a copy of the medium is preferred for the sake of security and, in many instances, throughput [17]. Another concern of using sneakernet is internet transmission cost against cost of moving the physical medium.

References

[1] R. Revanth, “Creating a personal cloud storage using NAS (network attached storage),” *Software Engineering*, vol. 11 , no. 2 , pp. 17-22 , 2019.
[2] R. Gurunath, M. Agarwal, A. Nandi, and D. Samanta, “An overview: Security issue in IoT network,” in *2018 2nd International Conference on ISMAC (IoT in Social, Mobile, Analytics and Cloud)*, pp. 104-107, 2018.
[3] A. S .-Y. Lai and A. M .-S . Ma, “Designing Network-attached storage architecture for small and medium enterprise applications,” in *Advances in Computer Science and Ubiquitous Computing* (J. J. Park, V. Loia, G. Yi, and Y. Sung, eds.), (Singapore), pp. 274-279, Springer Singapore, 2018.
[4] R. N. Halim, ”Penerapan network attached storage (NAS) berbasis Raspberry Pi di LP3SDM Azra Palembang,” Jurnal Teknologi Informasi dan Ilmu Komputer, vol. 6, no. 3, pp. 309-314, 2019.
[5] C. Suharyanto and A. Maulana, ”Perancangan network attached storage
(NAS) menggunakan raspberry pi untuk usaha mikro kecil dan menengah (UMKM),” *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, vol. 5, pp. 271-278, Feb. 2020.
[6] A. Andreev and D . E. Koo, ”Practical guide to storage of large amounts of microscopy data,” *Microscopy Today*, vol. 28, no. 4, p. 4245 , 2020.
[7] S . J . Johnston, P. J . Basford, C . S . Perkins, H . Herry, F . P. Tso, D. Pezaros, R. D . Mullins, E. Yoneki, S. J. Cox, and J. Singer, ”Commodity single board computer clusters and their applications,” *Future Generation Computer Systems*, vol. 89, pp. 201-212 , 2018.
[8] Raspberry Pi Foundations, ”Raspberry pi 4 tech specs.” https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/, retrieved on August 21, 2021
[9] C. Maguire, ”Developing a network storage device using a single board computer.” Masters’ Thesis: University of Bedfordshire, 2018
[10] A. Kurniawan, A. H. Hendrawan, and B. A. Prakosa, ”Analisis perbandingan kinerja freeNAS, NAS4Free dan open media vault (OMV) sebagai sistem operasi network attached storage (NAS),” in *Seminar Nasional Teknologi Informasi*, vol. 1 , pp. 74-83, 2018.
[11] P. M . Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D . A. Patterson, ”RAID: High-performance, reliable secondary storage,” ACM Comput.Surv.,vol. 26, p. 145-185 , June 1994.
[12] A. Leventhal, ”Triple-parity raid and beyond,” *ACM Queue*, vol. 7, no. 11, 2009 .
[13] Y. Li, N. Wang, C . Tian, S . Wu, Y. Zhang, and Y. Xu, ”A hierarchical RAID architecture towards fast recovery and high reliability,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 4, pp. 734-747, 2018.
[14] M. Prinzlau, ”Cloud storage price comparison 2021.” https://www.cloudwards.net/comparison/, retrieved on August 28, 2021
[15] Q. Le, A. Amer, and J. Holliday, ”Raid4SMR: Raid array with shingled magnetic recording disk for mass storage systems,” *Journal of Computer Science and Technology*, vol. 34, pp. 854868, 2019.

Accredited “Rank 4”(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

[16] J . Lienig and H. Bruemmer, *Reliability Analysis*, pp. 45-73. Cham: Springer International Publishing, 2017.
[17] A. C. Jaya, C. Safitri, and R. Mandala, ”Sneakernet : A technological overview and improvement,” in *2020 International Conference on Sustainable Engineering and Creative Computing*, 2020.