



2023

Performance Evaluation Of The Kvm Hypervisor Running On Arm-Based Single-Board Computers

Eric Gamess

Jacksonville State University

Mausam Parajuli

Jacksonville State University

Syed Shah

Jacksonville State University

Follow this and additional works at: https://digitalcommons.jsu.edu/fac_res



Part of the [Computer Sciences Commons](#)

Recommended Citation

Gamess, E., Parajuli, M., and Shah, S. (2023). Performance Evaluation of the KVM Hypervisor Running on ARM-Based Single-Board Computers. *International Journal of Computer Networks & Communications*, 15(2), 147-164.

This Article is brought to you for free and open access by the Faculty Scholarship & Creative Work at JSU Digital Commons. It has been accepted for inclusion in Research, Publications & Creative Work by an authorized administrator of JSU Digital Commons. For more information, please contact digitalcommons@jsu.edu.

PERFORMANCE EVALUATION OF THE KVM HYPERVISOR RUNNING ON ARM-BASED SINGLE-BOARD COMPUTERS

Eric Gamess, Mausam Parajuli, and Syed Shah

MCIS Department, Jacksonville State University, Jacksonville, Alabama, USA

ABSTRACT

Single-Board Computers (SBCs) were initially targeted for education and small projects with low power-processing needs. However, their computational power has increased dramatically in the last few years, and they are now used in more advanced developments. In this paper, a study of the feasibility of using ARM-based SBCs as hypervisors is done. The authors selected the Raspberry Pi 4 Model B and the ODROID-N2+ and assessed them as virtualization servers, when running up to four VMs simultaneously, with the Linux de facto hypervisor (KVM). The tests performed in this work include: reading and writing throughputs in different types of storage media, processing power assessment, memory performance, timed compilations of open-source software, and performance of encryption algorithms. The results of the experiments showed that the amount of memory available in these SBCs is a determinant factor about the maximum number of VMs that can be executed simultaneously. The performance of the ODROID-N2+ exceeded the Raspberry Pi 4 Model B. However, the community support received by the latter is huge compared to the one of the former, and this can be a game changer when selecting a viable platform.

KEYWORDS

Single-Board Computers, SBC, Raspberry Pi, ODROID, Performance Evaluation, Benchmarks, Virtualization, Kernel-based Virtual Machines, KVM.

1. INTRODUCTION

Virtualization [1][2] is the usage of software to make a reflection layer over the actual hardware. This enables the possibility of running several Virtual Machines (VMs), operating systems, and applications on a unique real server. Consequently, virtualization allows a more productive utilization of the actual computer hardware, and minimizes the cost of operation. Several hypervisors have been developed by the computing community (e.g., KVM [3][4], Xen [5], Proxmox VE [6], VMware ESXi [7], etc) and are wildly used in servers, worldwide.

On the other hand, there is a great interest in small power-efficient computers, with low prices, to develop all kinds of applications. This enthusiasm has been growing with the release of Single-Board Computers (SBCs) from several manufacturers. They were initially targeted for teaching purposes or to implement small projects that require low computing power, especially in the IoT world. However, since their initial release, their computing power has increased dramatically, and they have started to be used increasingly in more advanced projects.

In this research, several SBCs were selected and a study of the viability of using them as hypervisors was carried out. The experiments included the Raspberry Pi 4 Model B [8] (RPi 4B) and the ODROID-N2+ [9][10], for their wide acceptance in the Internet community. At the level of the hypervisor, Kernel-based Virtual Machine [3][4] (KVM) was chosen, since it is open-

source and has been part of Linux for some time now. The tests that were conducted on the selected SBCs show that the actual limitations to using them as hypervisors are more related to the constrained amount of memory, than the limited computing power. Furthermore, in most of the experiments, the ODROID-N2+ exceeded the RPi 4B. However, the limited support that is offered for the ODROID platform may be a fundamental limitation for many projects.

The rest of the paper is structured as follows. Section 2 discusses a number of peer-reviewed literature works conducted within this research area. Section 3 provides the characteristics of the chosen SBCs for the study. A description of the testbeds for the experiments is given in Section 4. In Section 5, the SBCs that are not suitable to be used as hypervisors are discarded. The benchmarking tests are presented, done, and discussed in Sections 6, 7, 8, 9, and 10. Finally, Section 11 concludes and summarizes the results of this paper as well as discusses future avenues for further research work within the area of study.

2. RELATED WORK

On the one hand, performance evaluation of SBCs has been made by several research teams. Some works are more general [11], while others focus on a specific field. For example, Lintermann, Pleiter, and Schröder [12] selected a cluster system (ODROID-MC1) consisting of 4 nodes and investigated the applicability of such a system to run scientific problems by evaluating its performance for specific applications. Allahi, Khan, Nagra, Idrees, and Masud [13] implemented the IEEE 1588 Precision Time Protocol on RPi 3 and analyzed its performance in LANs and WLANs. The authors of [14] did an empirical study and measured the network capabilities of several Raspberry Pi models (RPi Zero W, RPi Zero 2 W, RPi 3 Model B, RPi 3 Model B+, and RPi 4 Model B) in LANs and WANs, for IPv4 and IPv6. Ford, Gamess, and Ogden [15] evaluated the performance of different Raspberry Pi models (RPi Zero W, RPi Zero 2 W, and RPi 3 Model B) as MQTT servers and clients.

On the other hand, hypervisors have been benchmarked but mainly for servers based on Intel processors. For example, Đorđević, Furtula, and Timčenko [16] assessed VMware ESXi and Microsoft Hyper-V in a server with 2 Intel Xeon E7-2850. They reported results obtained with Filebench, about access to web and file servers. In [17], the authors benchmarked Citrix XenServer and KVM on Dell Power Edge R620 servers based on Intel Xeon Hexa-core CPUs. The work is focused on studying how the number of vCPUs can affect the performance of a VM. The authors of [18] assessed 3 type-2 hypervisors (Oracle VirtualBox, VMware Workstation Player, and Microsoft Hyper-V) running on a PC with an Intel Core i5-4590S. The measurements reported include random file access, and the performance of email and web servers. Lingayat, Badre, and Gupta [19] measured the impact of implementing containers (Dockers) in 2 scenarios: (1) directly on top of the operating system and (2) inside a VM. As hypervisor, they used KVM and studied the effects on deploying 20, 30, 40, and 50 web server container images, on a server based on Intel Core i7-4600U. In [20], the authors studied the impact of DDoS attacks on the performance of 3 hypervisors (Xen, KVM, and Oracle VirtualBox). For the experiments, they used 3 identical server machines, each one with an Intel Core i5 processor. They reported the network, CPU, and disk performances under 2 types of DDoS attacks.

There is little done in the area of performance evaluation of hypervisors running on ARM-based SBCs, and the authors could only find the work of Toumassian, Werner, and Sikora [21]. This work measured the performance overhead of 2 hypervisors (Xen and Jailhouse) on ARM processors in the context of heavy loads. The measurements were executed on a Banana Pi board from LeMaker, which contains an Allwinner A20 (dual-core ARM Cortex-A7 CPU).

3. CHARACTERISTICS OF THE CHOSEN SBCs

Nowadays, there is a plethora of manufacturers that are proposing SBCs. Some of the well-accepted SBCs include the ones distributed by the Raspberry Pi Foundation, Hardkernel, BeagleBoard.org, Nvidia, and UDOO. This paper focuses on the Raspberry Pi and ODROID families, and studies their suitability as hypervisors. They were selected since they seem to be the favorite SBCs used by hobbyists and professionals in the area. Table 1 shows the specifications of the Raspberry Pi 3 Model B [22] (RPi 3B), the Raspberry Pi 3 Model B+ [23] (RPi 3B+), the Raspberry Pi 4 Model B [8] (RPi 4B), and the ODROID-N2+ [9][10] (OD N2+). It is worth clarifying that the RPi 4B is sold with different sizes of RAM (1, 2, 4, and 8 GB), for US\$35, US\$45, US\$55, and US\$75, respectively. Similarly, the ODROID-N2+ has two possible configurations of RAM: 2 GB and 4 GB for a price of US\$66 and US\$83, respectively.

Table 1. Specifications of Several SBCs

	RPi 3B	RPi 3B+	RPi 4B	ODROID-N2+
SoC Type	Broadcom BCM2837	Broadcom BCM2837B0	Broadcom BCM2711	Amlogic S922X
Core Type	Quad-core ARM Cortex-A53 @ 1.2 GHz	Quad-core ARM Cortex-A53 @ 1.4 GHz	Quad-core ARM Cortex-A72 @ 1.8 GHz	Quad-core ARM Cortex-A73 @ 2.4 GHz & Dual-core ARM Cortex-A53 @ 2.0 GHz
RAM	1 GB LPDDR2 SDRAM	1 GB LPDDR2 SDRAM	1/2/4/8 GB LPDDR4-3200 SDRAM	2/4 GB DDR4
USB Ports	4 x USB 2.0	4 x USB 2.0	2 x USB 2.0 & 2 x USB 3.0	4 x USB 3.0 & 1 x Micro USB 2.0 OTG
HDMI Port	1 x Full-size HDMI	1 x Full-size HDMI	2 x Micro HDMI	1 x Full-size HDMI
Data Storage	MicroSD	MicroSD	MicroSD	MicroSD/eMMC
Ethernet	10/100 Mbps	10/100/1000 Mbps over USB 2.0	10/100/1000 Mbps	10/100/1000 Mbps
WiFi	IEEE 802.11b/g/n	IEEE 802.11b/g/n/ac	IEEE 802.11b/g/n/ac	No
Price	US\$35	US\$35	US\$35/45/55/75	US\$66/83

4. TESTBEDS FOR THE EXPERIMENTS

For the testbeds, the following equipment was used: RPi 3B, RPi 3B+, RPi 4B with 1 GB, RPi 4B with 2 GB, RPi 4B with 4 GB, RPi 4B with 8 GB, ODROID-N2+ with 2 GB, and ODROID-N2+ with 4 GB.

Several operating systems are available for Raspberry Pi (Raspberry Pi OS, Debian, Ubuntu, Armbian, DietPi, Fedora, Manjaro, LibreELEC, RetroPie, etc). The Raspberry Pi OS (64-bit version) was selected for this research due to its popularity and good support by the community. It is developed by the Raspberry Pi Foundation and based on Debian Bullseye. The last version was released on September 22, 2022. Three options for this operating system are available: (1) Raspberry Pi OS Lite, (2) Raspberry Pi OS with Desktop, and (3) Raspberry Pi OS with Desktop and Recommended Software. The “Lite” option is a minimal image consisting of 564 packages without an X window manager, hence, more suitable for a server environment. The “Desktop” and “Desktop and Recommended Software” options consist of 1413 and 1767 packages,

respectively, including an X window manager and a desktop environment, therefore more appropriate for end-users. Thus, the “Lite” option was chosen for all the experiments since it is the best option to run a hypervisor.

At the operating system level, Hardkernel recommends Ubuntu or Android Pie for ODROID, although other operating systems are available (Armbian, DietPi, Archdroid, etc). This research group initially wanted to use the Ubuntu version distributed by Hardkernel (version 20.04 released in February 2022¹). However, the team had a hard time with this Linux distribution, since it has little support for USB WiFi adapters, and it does not include the KVM modules required for virtualization. Hence, after trying several options, the CLI version of Armbian was selected. It is based on Debian Bullseye, with version 5.19.10 for the kernel, and was released on October 2022².

Several hypervisors are available (e.g., KVM [3][4], Xen [5], Proxmox VE [6], VMware ESXi [7], etc) to implement a Virtual Machine (VM) server. They all have stable versions for the x86 architectures (e.g., Intel and AMD). For the ARM architecture, most of them do not exist or are still in beta versions. KVM [3][4] (Kernel-based Virtual Machine) is an open-source virtualization technology built into Linux. It is available for the ARM architecture, and allows running multiple VMs (isolated virtual environments) into an ARM-based SBC such as a Raspberry Pi or an ODROID. When choosing an operating system, selecting one with the kernel compiled with virtualization support is recommended. If it is not the case, VMs can still be created on the SBC, but they will use QEMU [24] instead of KVM [3][4], and the performance will be much lower.

For all the SBCs, a headless setup was done, that is, the boards were not connected to a keyboard and monitor. In each SBC, KVM [3][4] was installed and 4 identical minimal VMs based on Debian 10.13 (Buster) were created. In other words, the VMs had a basic operating system and no X Window capabilities. The only service that was installed and run in the VMs was an SSH server, so they could be accessed remotely.

Three types of storage were used for the experiments: (1) microSD cards, (2) SSDs, and (3) eMMC cards. Their specifications were:

- MicroSD cards: 64 GB SanDisk Extreme PRO microSDXC UHS-I Memory Card (SDSQXCY-064G-GN6MA). It is one of the fastest microSD cards on the market, and according to the manufacturer, it can achieve reading and writing speeds up to 170 MB/s and 90 MB/s, respectively.
- SSDs: 1 TB SanDisk Ultra 3D SSD (SDSSDH3-1T00-G25). According to the manufacturer, it can achieve reading and writing speeds up to 560 MB/s and 530 MB/s, respectively. When used, the SSDs were connected to the SBCs through a USB 3.0 port.
- eMMC cards: 64 GB Hardkernel eMMC 5.1. The eMMC cards were only used for the ODROID-N2+.

The chips of an SBC can overheat if they do not have enough cooling. Hence, CPU throttling (also known as dynamic frequency scaling) may occur to decrease the electrical energy being consumed, and in turn, to reduce the heat generation. To tackle this problem, the ODROID-N2+ has a heat sink in the bottom part. However, the default configuration of the Raspberry Pi SBCs (RPi 3B, RPi 3B+, and RPi 4B) used in this work does not have any overheating control mechanism. Hence, they were put inside cases that had small fans.

¹ https://wiki.odroid.com/getting_started/os_installation_guide

² <https://www.armbian.com/odroid-n2>

Each experiment in this paper was executed several times, and the reported results are an average of the repeated experimental runs. By repeating and averaging, the consistency of the empirical findings is ensured.

5. FEASIBILITY OF USING SBCs WITH LOW MEMORY

The goal of this experiment was to study the feasibility of using SBCs with a low amount of RAM as hypervisors, such as the RPi 3B [22], the RPi 3B+ [23], and the RPi 4B with 1 GB [8]. Hence, a CPU performance benchmark was run on all the SBCs in three scenarios: (1) natively, (2) within 1 VM, and (3) within 2 VMs at the same time. In the first scenario, no VMs were running, to minimize the load over the CPU, and to have the best possible native performance. In the second scenario, only 1 VM was started, and the benchmark was run inside this VM. In the last scenario, 2 VMs were started, and the benchmark was run simultaneously inside these VMs. The 7-Zip archiving tool [25] has three major functionalities: (1) pack and compress files into archives, (2) uncompress and extract files from archives, and (3) benchmark the power of a CPU through the LZMA [26] (Lempel–Ziv–Markov chain Algorithm) compression and decompression. The benchmarking tool reports how fast a CPU processes the compression and decompression instructions over dummy data, displaying the results in MIPS (Million Instructions Per Second). The number of threads to be used during the assessment can be specified in the command line. Figure 1 shows the instructions that were used to install and execute the 7-Zip benchmark. The line numbers were added for reference. Line 01 installed the tool from the repositories. Lines 02-03 run the benchmark for 1 and 2 threads, respectively.

01: apt-get install p7zip
02: 7zr b -mmt1
03: 7zr b -mmt2

Figure 1. Installation and Execution of 7-Zip

For this experiment, the microSD cards were used for storage. That is, the operating system of the SBCs, the hypervisor (KVM), and the VMs were installed/created in the microSD cards mentioned in Section 4 (64 GB SanDisk Extreme PRO). At the level of the VMs, the number of vCPUs was set to 2. At the level of the memory of the VMs, two types of assessments were done: 192 MB and 256 MB of RAM. These values were selected following the guideline of the Debian project [27]. Figures 2 and 3 depict the results obtained for compression and decompression, respectively, when the VMs had 192 MB of RAM. Each figure consists of 8 groups of 6 bars. Each group represents one architecture: (1) RPi 3B, (2) RPi 3B+, (3) RPi 4B with 1 GB, (4) RPi 4B with 2 GB, (5) RPi 4B with 4 GB, (6) RPi 4B with 8 GB, (7) ODROID-N2+ with 2 GB, and (8) ODROID-N2+ with 4 GB. For each group, the bars are (1) native performance with 1 thread, (2) native performance with 2 threads, (3) performance with 1 VM and 1 thread, (4) performance with 1 VM and 2 threads, (5) performance with 2 VMs and 1 thread, and (6) performance with 2 VMs and 2 threads. From this experiment, it can be seen that the RPi 3B and the RPi 3B+ have similar performances. As pointed out in Table 1, the major differences between the two models are upgraded-network capabilities. All the RPi 4B models significantly outperformed the RPi 3B and the RPi 3B+. With the increasing amount of RAM in the RPi 4B and the ODROID-N2+, the performance did get a slight boost. The ODROID-N2+ showed the best performance.

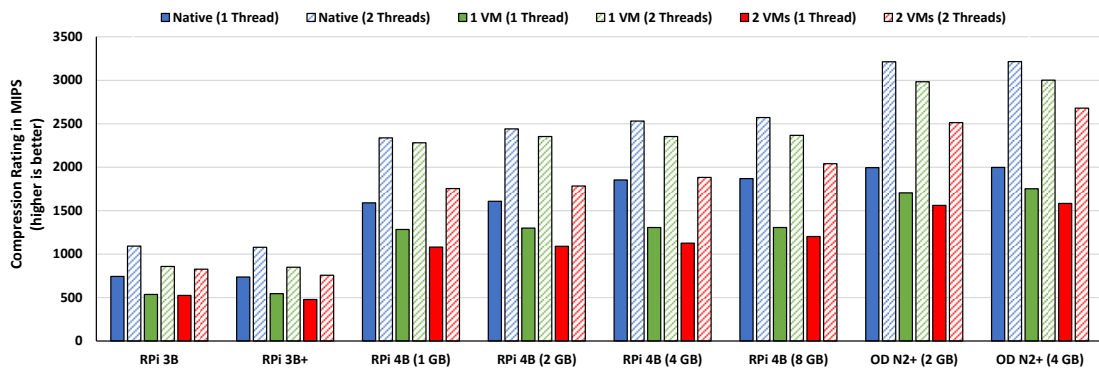


Figure 2. Compression Rating using 7-Zip with 1 or 2 Threads (VMs with 192 MB of RAM)

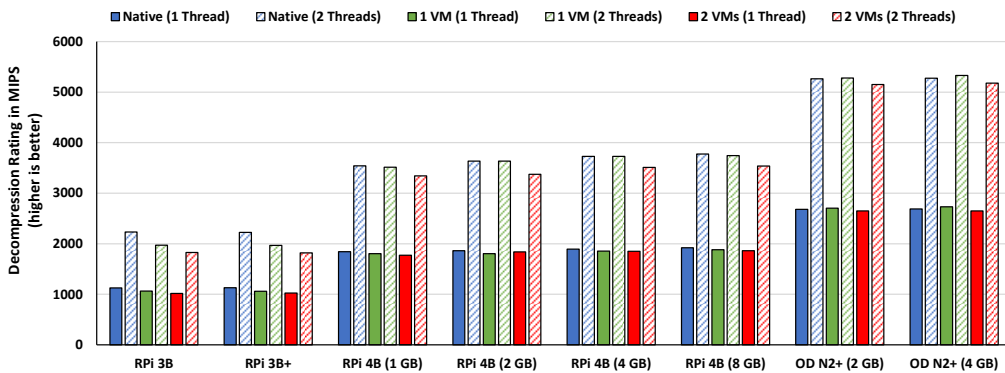


Figure 3. Decompression Rating using 7-Zip with 1 or 2 Threads (VMs with 192 MB of RAM)

Figures 4 and 5 depict the results obtained for compression and decompression, respectively, when the VMs had 256 MB of RAM. For the RPi 3B, the RPi 3B+, and the RPi 4B with 1 GB, the figures only have results for the native and 1 VM scenarios (4 bars instead of 6 in those cases). For the 2 VMs scenario, the SBCs simply crashed and had to be reset. This experiment shows that using an SBC as a hypervisor is not viable if it has just 1 GB of memory, even in contexts with low memory requirements for the VMs (e.g., only 1 VM with 256 MB of RAM can be run at a time). Hence, the RPi 3B, the RPi 3B+, and the RPi 4B with 1 GB were discarded from the remaining experiments.

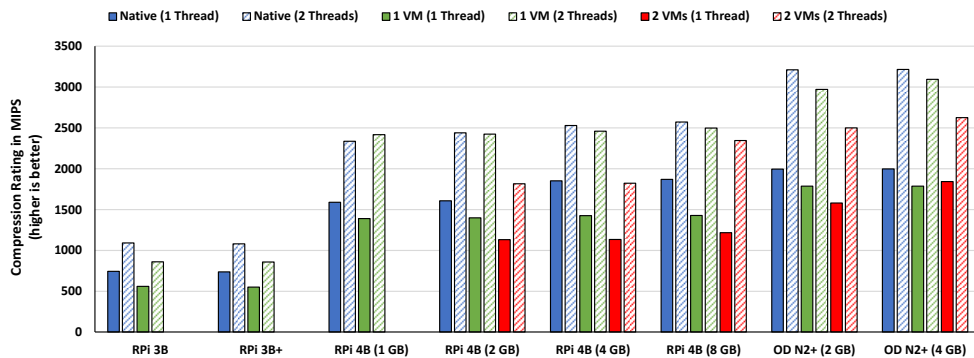


Figure 4. Compression Rating using 7-Zip with 1 or 2 Threads (VMs with 256 MB of RAM)

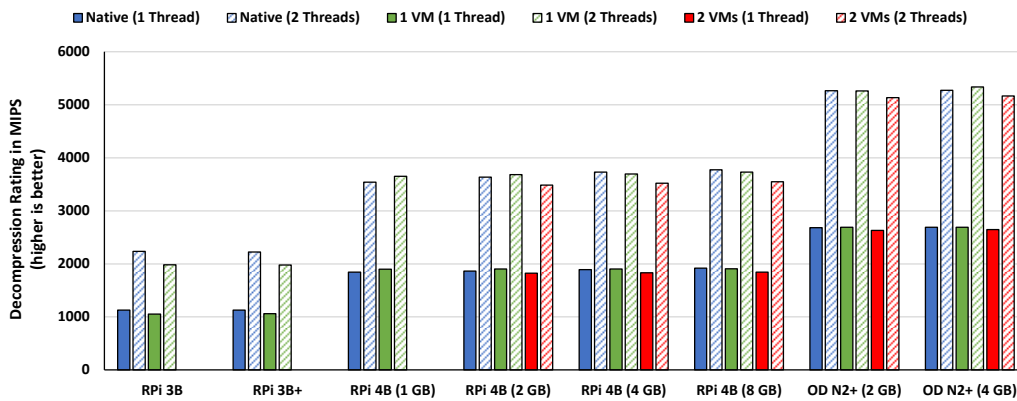


Figure 5. Decompression Rating using 7-Zip with 1 or 2 Threads (VMs with 256 MB of RAM)

For the rest of this paper, the Debian VMs were configured with 512 MB of RAM as recommended in [27]. It means that due to limitations of memory, only 2 VMs could be run simultaneously in the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB. Hence, in Figures 7, 8, 9, 11, 12, 13, 14, 15, 17, 20 and 21, there are no bars/results in the case of 4 VMs for the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB.

6. ASSESSING THE ACCESS TO THE FILESYSTEM USING DIFFERENT STORAGE TECHNOLOGIES

This experiment aims to evaluate the reading and writing performance of the filesystem when stored in different technologies. As mentioned in Section 4, three types of storage were used: (1) microSD cards, (2) SSDs, and (3) eMMC cards. The operating system of the SBCs, the hypervisor (KVM), and the 4 VMs were installed/created in each media. At the level of the VMs, the number of vCPUs was set to 2 and the RAM was configured to 512 MB. For the assessment, the standard Unix/Linux “dd” tool was selected, which reports throughputs in MB/sec. Figure 6 shows the instructions that were used. For the experiment with SSD, the write-caching was disabled with Lines 01-04. Line 01 got the list of block devices in the SBC so the SSD could be identified. Line 02 checked the status of the write-caching feature (it is enabled by default). Line 03 disabled it, and a verification of its new state was done in Line 04. Notice that this was not done for the microSD and eMMC cards, since they do not have write-caching feature. The sequential read access was benchmarked with Line 05, where an existing testfile of 1 GB was read through 256 read operations of 8 MB each. The sequential write was evaluated with Line 06, where a 1 GB file was created by making 256 write operations, using blocks of 8 MB.

```

01: lsblk -p
02: hdparm -W /dev/<deviceName>
03: hdparm -W 0 /dev/<deviceName>
04: hdparm -W /dev/<deviceName>
05: dd if=<testfile> of=/dev/null bs=8M count=256 iflag=direct
06: dd if=/dev/zero of=<testFile> bs=8M count=256 oflag=direct
    
```

Figure 6. Instructions Used to Assess the Reading and Writing Performance of the Filesystem

Figures 7 and 8 show the results obtained for the microSD and SSD storages, respectively. Each figure consists of 8 groups of 5 bars. The groups are: (1) native read, (2) native write, (3) read

with 1 VM, (4) write with 1 VM, (5) read with 2 VMs, (6) write with 2 VMs, (7) read with 4 VMs, and (8) write with 4 VMs. For each group, the bars are (1) RPi 4B with 2 GB, (2) RPi 4B with 4 GB, (3) RPi 4B with 8 GB, (4) ODROID-N2+ with 2 GB, and (5) ODROID-N2+ with 4 GB. Figure 9 depicts the results obtained for the eMMC card. Since the RPi 4B does not have support for this type of storage (see Table 1), there are only results for the ODROID-N2+. It is worth remembering that, as mentioned in Section 5, there are no results in Figures 7, 8, and 9 for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations.

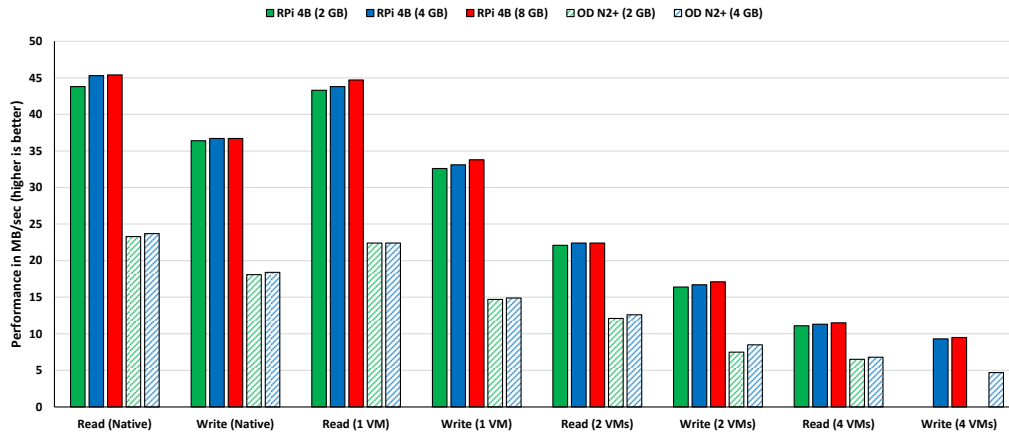


Figure 7. Read and Write Performance for the MicroSD Card

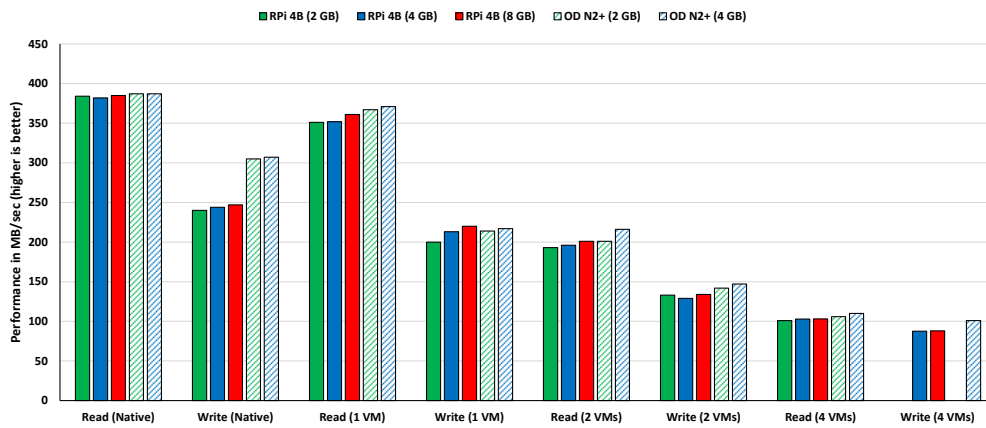


Figure 8. Read and Write Performance for the SSD

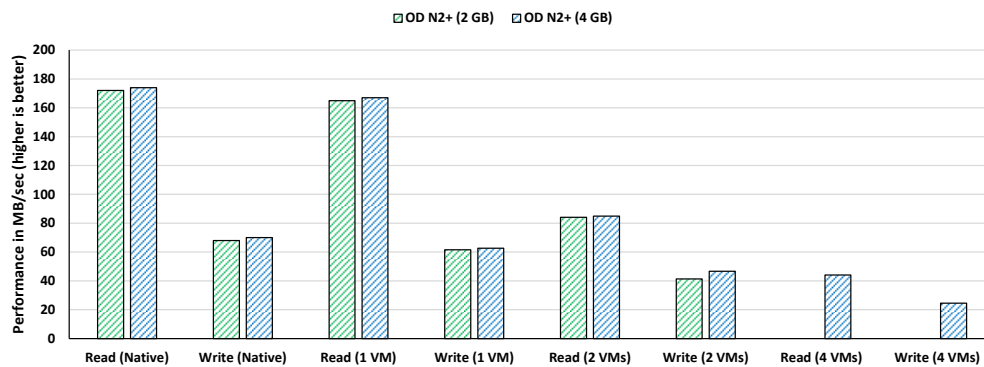


Figure 9. Read and Write Performance for the eMMC Card (ODROID-N2+ Only)

Figures 7, 8, and 9 confirm that the read performance is superior to the write performance. The microSD card has the poorest results, while the SSD significantly outperformed the other two storage media. The eMMC has much better results than the microSD card (only applies to the ODROID-N2+). The amount of RAM of an SBC slightly affected the results. The RPi 4B did much better when accessing the microSD cards than the ODROID-N2+ (see Figure 7). However, when using SSDs, the SBCs of the 2 manufacturers have similar achievements (see Figure 8), except for the native write, where the ODROID-N2+ did significantly better. In light of these results, the microSD and eMMC cards were discarded for the remaining tests, since it is more probable that the implementation of a hypervisor will use an SSD for storage.

7. EVALUATING THE PROCESSING POWER

The objective of this experiment was to assess the processing power natively and when varying the number of vCPUs assigned to the VMs. PerformanceTest [28], a benchmarking solution developed by PassMark Software, was used for the performance. It is multiplatform and supports Windows (x86 and ARM), Linux (x86 64-bit, ARM 32-bit, and ARM 64-bit), macOS, Android, and iOS. To use PerformanceTest on a Windows device, a license should be bought. However, the product is free for the other architectures. The Windows version is the most comprehensive, and has tests for CPU, memory, disk, 2D graphics, and 3D graphics. For Linux, PerformanceTest is limited to CPU and memory tests. The CPU assessment includes 9 tests: (1) Integer Math, (2) Floating Point Math, (3) Prime Numbers, (4) Sorting, (5) Encryption, (6) Compression, (7) CPU Single Threaded, (8) Physics, and (9) Extended Instructions. In addition to these 9 specific performance values, PerformanceTest reports an aggregated result named “CPU Mark”. Figure 10 shows the steps that were followed to download, install, and execute the tools in the SBCs under test. Line 01 installed the dependencies from the repositories. Lines 02-03 downloaded the archive file from PassMark Software and unzipped it. Line 05 executed the binary file.

```

01: apt-get install libncurses5 libcurl4
02: wget https://www.passmark.com/downloads/pt_linux_arm64.zip
03: unzip pt_linux_arm64.zip
04: cd PerformanceTest
05: ./pt_linux_arm64

```

Figure 10. Installation and Execution of PerformanceTest

Figures 11, 12, and 13 depict the “CPU Mark” (aggregated results for the 9 CPU tests) natively and when varying the number of vCPUs in the VMs. Each figure consists of 4 groups of 5 bars.

The groups are: (1) Native, (2) one VM, (3) two VMs, and (4) four VMs. The RAM of each VM was set to 512 MB. SSDs were used for storage in the SBCs. It is noticeable that the ODROID-N2+ outperformed the RPi 4B in all these experiments. For the native performance, all the cores of the SoC (System on a Chip) were available (4 cores for the RPi 4B and 6 cores for the ODROID-N2+ as specified in Table 1) during the execution of the benchmark, which explains the much higher performance in this case.

For the experiments with 1 vCPU per VM (see Figure 11), there were enough cores in the SoC so that each vCPU was assigned to one core (a maximum of 4 cores were needed when having 4 VMs, since each VM had a single vCPU). Hence, Figure 11 reflects this by having the “same” performance when 1, 2, or 4 VMs executed the benchmark simultaneously. It can also be noted that the native performance of the RPi 4B is almost 4 times higher than the one of a VM, since the 4 cores were used in the native evaluation and only 1 core was used in the VM assessment. However, the native performance of the ODROID-N2+ is 4 to 6 times greater than the one of a VM, and this is due to the asymmetric nature of the 6 cores of this SBC (Quad-core ARM Cortex-A73 @ 2.4 GHz & Dual-core ARM Cortex-A53 @ 2.0 GHz). Also, it is worth remembering that, as mentioned in Section 5, there are no results for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations.

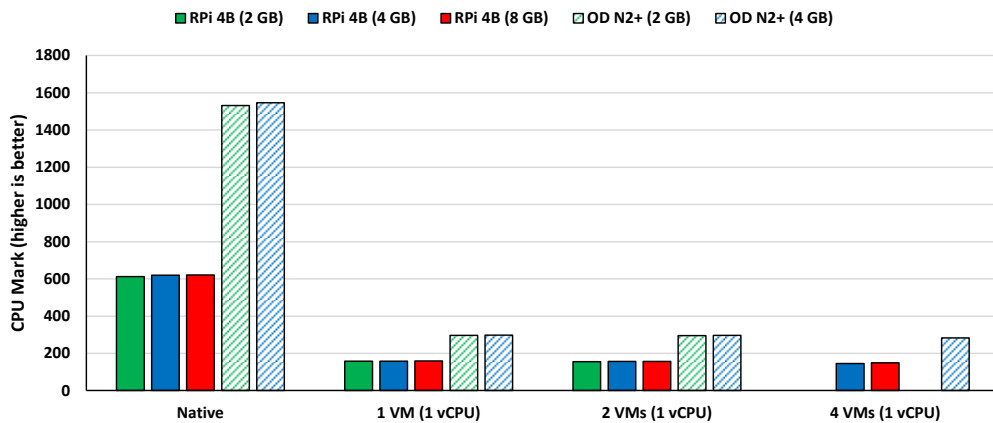


Figure 11. CPU Performance when the VMs had 1 vCPU

For the experiments with 2 vCPUs per VM (see Figure 12), there were enough cores in the SoC to assign 1 vCPU per core, up to 2 VMs. However, this was not the case with 4 VMs, where 8 cores would be required to assign 1 vCPU per core. Hence, the results are the “same” for 1 and 2 VMs (second and third groups of bars). However, the results for 4 VMs (fifth group of bars) are smaller since several vCPUs were assigned to each core. Also, it is worth remembering that, as mentioned in Section 5, there are no results for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations.

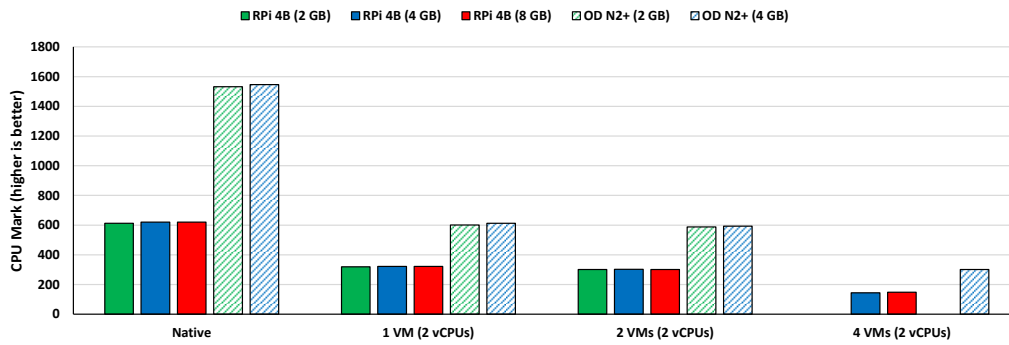


Figure 12. CPU Performance when the VMs had 2 vCPUs

For the experiments with 4 vCPUs per VM (see Figure 13), there were enough cores in the SoC to assign 1 vCPU per core, only for the tests with 1 VM (second group of bars). The required numbers for 2 and 4 VMs would be 8 and 16 cores, respectively, to assign 1 vCPU per core. For the RPi 4B, the experiments gave the “same” result for the native execution and for 1 VM, since the 4 available cores were used in both assessments. In the case of the ODROID-N2+, the native results are better than the ones obtained for 1 VM, since the 6 cores were used in the native case, while 4 cores were used in the 1 VM experiments. For the 2 VMs experiments of the RPi 4B, the results are about “half” the values of the 1 VM tests, since in this case, there were 2 vCPUs assigned per core. For the 4 VMs experiments of the RPi 4B, the results are about “one-fourth” the values of the 1 VM tests, since in this case, there were 4 vCPUs assigned per core. Also, it is worth remembering that, as mentioned in Section 5, there are no results for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations.

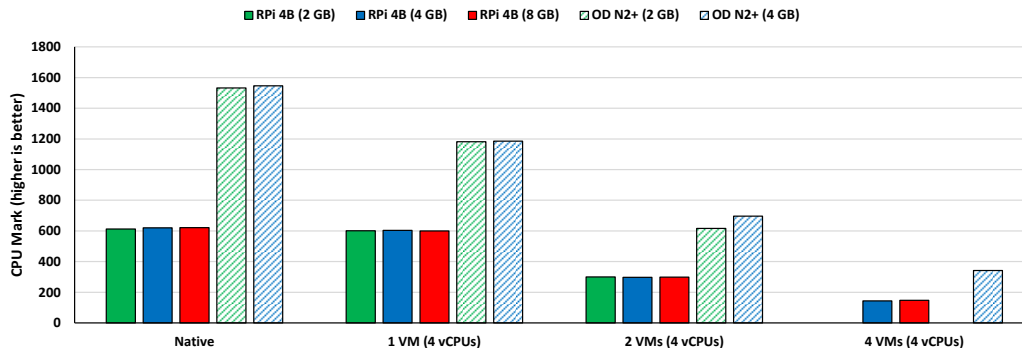


Figure 13. CPU Performance when the VMs had 4 vCPUs

8. MEMORY PERFORMANCE

This experiment focused on measuring the reading and writing performance of the RAM memory. The memory test of PerformanceTest [28] was used and consists of 7 benchmarks: (1) Database Operations, (2) Memory Read Cached, (3) Memory Read Uncached, (4) Memory Write, (5) Available RAM, (6) Memory Latency, and (7) Memory Threaded.

SSDs were used for storage in the SBCs. At the level of the VMs, the number of vCPUs was set to 2 and the RAM was configured to 512 MB. Figures 14 and 15 display the results obtained for the “Memory Read Uncached” and the “Memory Write”, and consist of 4 groups of 5 bars, where the groups are: (1) native, (2) one VM, (3) two VMs, and (4) four VMs. The ODROID-N2+

surpassed the RPi 4B, especially in the native and 1 VM tests. Also, it is worth remembering that, as mentioned in Section 5, there are no results for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations.

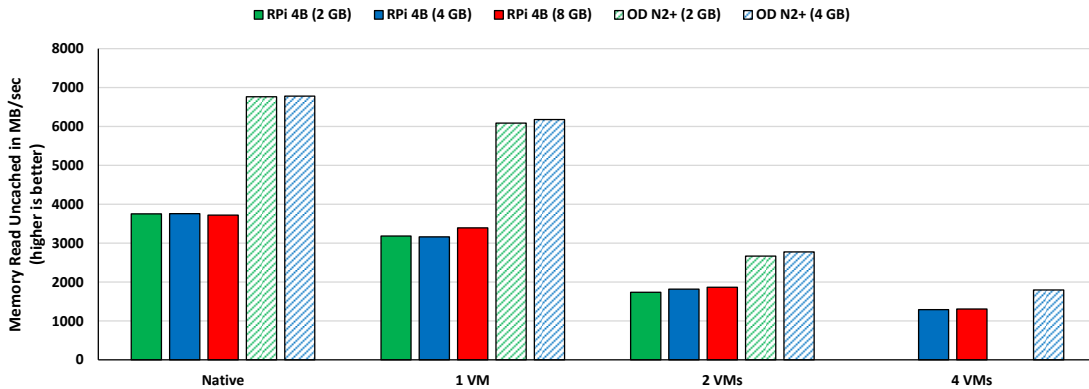


Figure 14. Memory Read Uncached

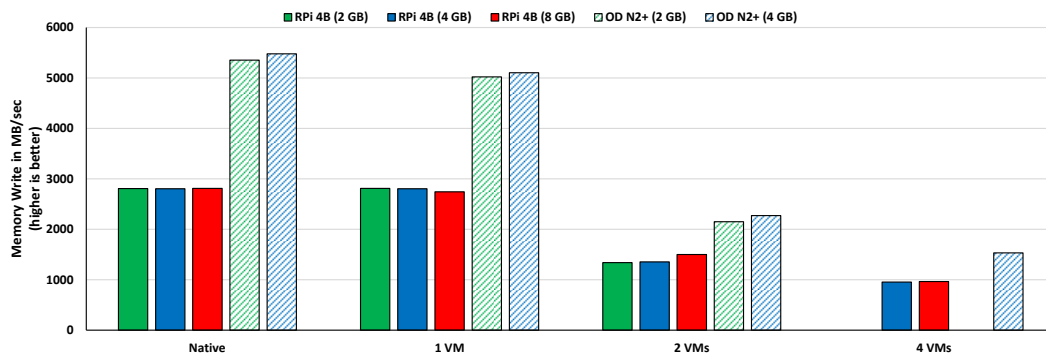


Figure 15. Memory Write

9. TIMED APACHE COMPILATION

The total time required to compile open-source software is considered a good benchmark to measure the performance of computers since it evaluates the system’s processor, memory, and filesystem. Common open-source applications that the Internet community has used for timed compilations include Apache [29] (a famous webserver), FFmpeg (a solution to record, convert, and stream audio and video), GCC (the GNU Compiler Collection), GDB (the GNU Debugger), ImageMagick (an application to create, edit, compose, or convert digital images), LLVM (a collection of modular and reusable compiler and toolchain technologies), Mesa (an implementation of OpenGL, Vulkan, and OpenCL), MPlayer (a movie player), PHP (a popular general-purpose scripting language that is especially suited to web development), and the Linux Kernel. In these experiments, the time required to compile the last version of Apache (version 2.4.54) [29] is reported.

Figure 16 shows the steps that were followed to compile Apache from the source files. The dependencies were installed from the repositories in Line 01. Apache, APR (Apache Portable Runtime), and APR-Util were downloaded with Lines 02-04. Lines 05-07 uncompressed and unpacked the source codes. The directory of APR and APR-Util were moved inside the directory of Apache with Lines 08-09. Line 11 was used for the configuration, while Line 13 did the timed

compilation. It is worth clarifying that the number of jobs (commands) that the utility “make” can run simultaneously is specified with the option “-j <numJobs>”.

```

01: apt-get install libpcre3-dev
02: wget https://dlcdn.apache.org/httpd/httpd-2.4.54.tar.bz2
03: wget http://archive.apache.org/dist/apr/apr-1.7.0.tar.bz2
04: wget http://archive.apache.org/dist/apr/apr-util-1.6.1.tar.bz2
05: tar -jxvf httpd-2.4.54.tar.bz2
06: tar -jxvf apr-1.7.0.tar.bz2
07: tar -jxvf apr-util-1.6.1.tar.bz2
08: mv apr-1.7.0 httpd-2.4.54/src/lib/apr
09: mv apr-util-1.6.1 httpd-2.4.54/src/lib/apr-util
10: cd httpd-2.4.54
11: ./configure --with-included-apr
12: make clean
13: time make -s -j <numJobs>
    
```

Figure 16. Instructions for Timed Apache Compilation

SSDs were used for storage in the SBCs. At the level of the VMs, the number of vCPUs was set to 2 and the RAM was configured to 512 MB. Figure 17 presents the obtained results that consist of 5 groups of 5 bars, where the groups are: (1) native with the maximum number of jobs, (2) native with 2 jobs, (3) one VM with 2 jobs, (4) two VMs with 2 jobs, and (5) four VMs with 2 jobs. In the first group of bars, the timed compilations were run with 4 and 6 jobs in the RPi 4B and the ODROID-N2+, respectively, according to the number of cores available in their SoC (see Table 1). As can be seen, the ODROID-N2+ exceeded the RPi 4B (lower time required to compile). For the RPi 4B, there is not much difference between the native experiments with 2 jobs (second group of bars) and the experiment when using 1 VM (with 2 jobs also), indicating that the overload of the virtualization can be neglected in this case. Notice that it is not the case of the ODROID-N2+, where the virtualization’s impact can be seen. Also, it is worth remembering that, as mentioned in Section 5, there are no results for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations.

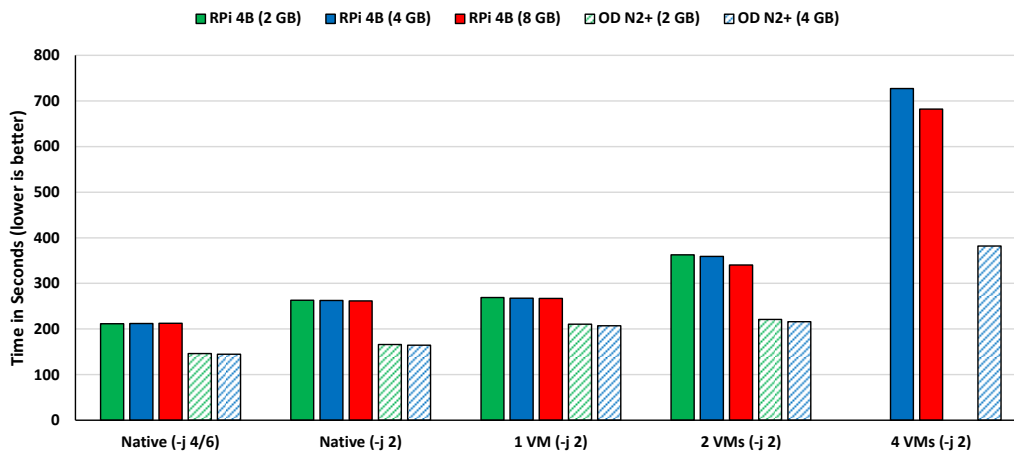


Figure 17. Timed Apache Compilation

10. PERFORMANCE OF ENCRYPTION ALGORITHMS

This experiment aims to assess the performance of standard symmetric encryption algorithms. Botan [30] is a cross-platform open-source C++ crypto library that supports the most publicly known cryptographic algorithms. For example, at the level of the symmetric encryption algorithms, the tool offers support for: AES-128, AES-192, AES-256, Blowfish, CAST-128, CAST-256, Camellia-128, Camellia-192, Camellia-256, DES, IDEA, TripleDES, Twofish, and others. Moreover, the goal of Botan is to provide the tools necessary to implement a range of practical systems, such as TLS protocol, X.509 certificates, modern AEAD ciphers, PKCS#11 and TPM hardware support, password hashing, and post-quantum crypto schemes. It is worth mentioning that Botan has the option to benchmark the implemented algorithms. Its last versions in the repositories for Raspberry Pi OS (for the RPi 4B) and Armbian (for the ODROID-N2+) are 2.17.3 and 2.19.2, respectively. Since they correspond to different versions that are not the last one, its last version (2.19.3) was installed in the testbed from the source code. Figure 18 displays the steps to download, install, and benchmark some symmetric encryption algorithms (Blowfish [31] and AES-256 [32]). Botan was downloaded in Line 01. Line 02 uncompressed and unpacked the source code. Lines 04-05 were used to configure and build Botan. Line 07 executed it in benchmarking mode for Blowfish and AES-256, which were selected due to their vast adoption by the community. In this case, Botan reports the encryption and decryption throughputs (rates) of both algorithms in MiB/sec.

```

01: wget http://botan.randombit.net/releases/Botan-2.19.3.tar.xz
02: tar -Jxvf Botan-2.19.3.tar.xz
03: cd Botan-2.19.3
04: python3 ./configure.py
05: make
06: export LD_LIBRARY_PATH=./:$LD_LIBRARY_PATH
07: ./botan speed --msec=15000 Blowfish AES-256

```

Figure 18. Instructions to Install and Assess Symmetric Encryption Algorithms with Botan

SSDs were used for storage in the SBCs. At the level of the VMs, the number of vCPUs was set to 2 and the RAM was configured to 512 MB. It is also worth mentioning that the RPi 4B [8] does not have hardware acceleration for AES, while the ODROID-N2+ [9][10] does. This can be verified with any of the commands of Lines 1-3 of Figure 19, which will show a list of “features” or “flags” that are supported by the CPU in hardware. Lines 04-05 display the features for the RPi 4B and ODROID-N2+, respectively.

```

01: cat /proc/cpuinfo
02: lscpu
03: hwinfo --cpu
04: Features      : fp asimd evtstrm crc32 cpuid
05: Features      : fp asimd evtstrm aes pmull sha1 sha2 crc32 cpuid

```

Figure 19. Verifying Hardware-acceleration Support for AES

Figure 20 depicts the encryption/decryption throughputs for Blowfish. As can be seen, the ODROID-N2+ has a slightly better performance than the RPi 4B. The amount of RAM of the SBCs did not make a substantial difference. As mentioned in Section 5, there are no results for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations. The assessment of Botan does not take advantage of the multicore CPU, hence the experiments gave the “same” results for native, 1, 2, or 4 VMs. Also, it is noted that encrypting

or decrypting requires a similar effort, since the results are very close to each other in these experiments.

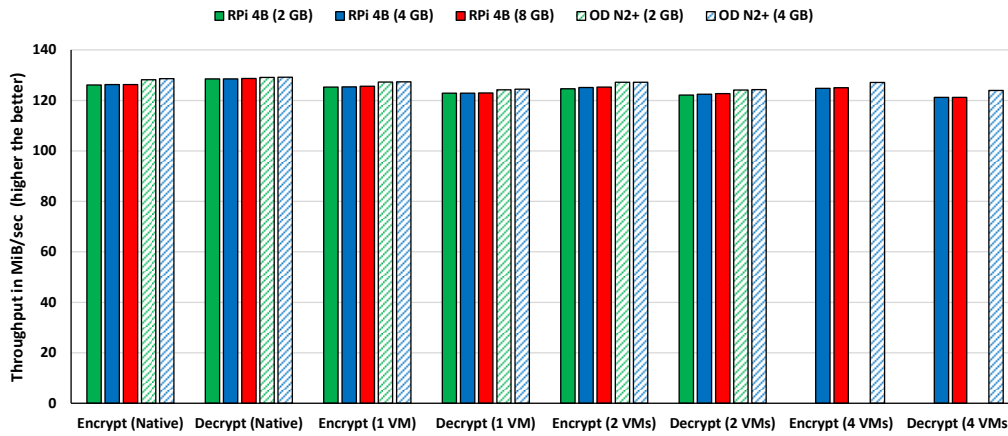


Figure 20. Encryption/Decryption Throughputs for Blowfish with Botan

Figure 21 depicts the encryption/decryption throughputs for AES-256. As can be seen, the ODROID-N2+ significantly outperformed the RPi 4B due to the hardware-accelerated support for AES. The amount of RAM of the SBCs did not make a significant difference. As mentioned in Section 5, there are no results for 4 VMs in the case of the RPi 4B with 2 GB and the ODROID-N2+ with 2 GB, due to memory limitations. The assessment of Botan does not take advantage of the multicore CPU, hence the experiments gave the “same” results with 1, 2, or 4 VMs. The differences between the native experiments and those done with VMs were noticeable only in the case of the ODROID-N2+.

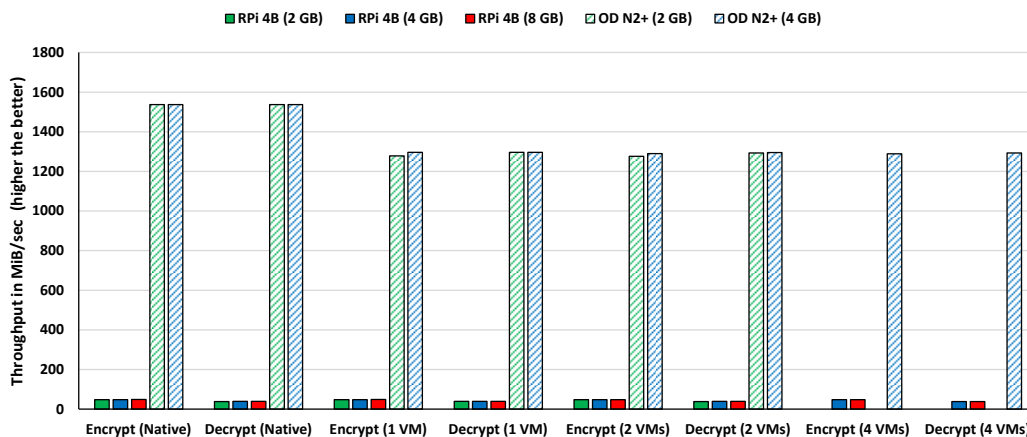


Figure 21. Encryption/Decryption Throughputs for AES-256 with Botan

11. CONCLUSIONS AND FUTURE WORK

This paper presented a study on using ARM-based SBCs as hypervisors. With the exception of the microSD card access, the ODROID-N2+ outperformed the RPi 4B. As reported in Table 1, the SoC of the ODROID-N2+ has higher specifications than the one of the RPi 4B, and its hardware-acceleration support for some crypto algorithms (AES, SHA-1, and SHA-256) makes it an attractive possibility for projects that require high performance. Furthermore, with the

problem of the global supply chain disruption since COVID-19, it is difficult to find SBCs of the Raspberry Pi Foundation, and so far, when they are available, they can only be bought at a much higher price (sometimes, even more than 3 times the price recommended by the manufacturer). These points may lead hobbyists and the industry toward the ODROID-N2+. However, the community's support is an essential factor in the equation. The support from the Raspberry Pi Foundation and the Internet community is huge for the Raspberry Pi SBCs, compared to the one that receives the ODROID platform. The authors found answers to their doubts with the RPi 4B by searching the Internet. However, it was not the case with the ODROID-N2+, and the research team had to find ways to solve them most of the time without assistance, significantly delaying the research.

Having a large community behind an SBC is essential. After this experience, it is clear to the authors that most of the support of the Internet community is for the x86-based architecture. The ARM-based SBCs are still far behind. Withing the ARM-based SBCs, the ones developed by the Raspberry Pi Foundation have a much better attention, especially regarding drivers and modules. In this research work, many USB WiFi adapters were tried. Just a few of them work out of the box with the Linux versions that were tested (Raspberry Pi OS, Debian, Ubuntu, Armbian, DietPi, etc). Hence, the authors faced the compilation problem. For most of the USB WiFi adapters, the source code for the Linux module is available on the Internet. Almost all of them did compile and run without problems on x86-based architectures. The Raspberry Pi devices are also well covered. However, the authors faced numerous issues when compiling them for the ODROID-N2+, ranging from unsuccessful compilations to modules that presented some type of inconvenience. For example, the modules compiled well but did not work at all, or performed much lower than expected, or frequently disconnected from the wireless routers, or only supported the 2.4 GHz band when the NIC did have both bands. To boot from different media, ODROID uses Petitboot. The authors faced issues when booting with other media on the ODROID-N2+, while it was straightforward for the RPi 4B.

This study seems to indicate that using the devices under test as hypervisors with KVM is feasible. A total RAM of 2 GB in the SBCs appears to work for 2 VMs configured with 512 MB. For 4 VMs with a RAM set to 512 MB, the experiments tend to indicate that a total RAM of 4 GB in the SBCs should be enough. For storage, using an SSD will significantly improve the performance. If an SSD is unavailable, an eMMC should be the second choice for the ODROID-N2+. A microSD card might only be used as the last possibility for storage.

In future work, the authors plan to evaluate the suitability of using other hypervisors and containers (Dockers, LXC, Podman, etc) in SBCs. Another avenue for research that the authors consider is assessing the network performance when using VMs and containers in SBCs.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

ACKNOWLEDGMENTS

We are grateful to “Faculty Commons” and the “College of Science & Mathematics” at Jacksonville State University for partially funding this project.

REFERENCES

- [1] Edouard Bugnion, Jason Nieh, and Dan Tsafir, *Hardware and Software Support for Virtualization (Synthesis Lectures on Computer Architecture)*, Morgan & Claypool Publishers, February 2017.
- [2] Shashank Mohan Jain, *Linux Containers and Virtualization: A Kernel Perspective*, First Edition, Apress, October 2020.
- [3] Vedran Dakic, Humble Devassy Chiramal, Prasad Mukhedkar, and Anil Vettathu, *Mastering KVM Virtualization: Design Expert Data Center Virtualization Solutions with the Power of Linux KVM*, Second Edition, Packt Publishing, October 2020.
- [4] Konstantin Ivanov, *KVM Virtualization Cookbook: Learn How to Use KVM Effectively in Production*, Packt Publishing, June 2017.
- [5] Sander van Vugt, *Practical Guide to XEN High Availability: Configuring Enterprise Virtualization on SUSE Linux Enterprise Server*, Books4brains, March 2010.
- [6] Wasim Ahmed, *Mastering Proxmox: Build Virtualized Environments using the Proxmox VE Hypervisor*, Third Edition, Packt Publishing, November 2017.
- [7] Mike Brown, Hersey Cartwright, Martin Gavanda, Andrea Mauro, Karel Novak, and Paolo Valsecchi, *The Complete VMware vSphere Guide: Design a Virtualized Data Center with VMware vSphere 6.7*, Packt Publishing, November 2019.
- [8] Raspberry Pi Foundation, *Raspberry Pi 4 Model B*, <https://www.raspberrypi.com/products/raspberrypi-4-model-b>.
- [9] Hardkernel, *ODROID-N2+ with 2 GB RAM*, <https://www.hardkernel.com/shop/odroid-n2-with-2gbyte-ram-2>.
- [10] Hardkernel, *ODROID-N2+ with 4 GB RAM*, <https://www.hardkernel.com/shop/odroid-n2-with-4gbyte-ram-2>.
- [11] Eric Gamess and Sergio Hernandez, "Performance Evaluation of Different Raspberry Pi Models for a Broad Spectrum of Interests", *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 2, pp. 819-829, 2022.
- [12] Andreas Lintermann, Dirk Pleiter, and Wolfgang Schröder, *Performance of ODROID-MC1 for Scientific Flow Problems*, *Future Generation Computer Systems*, vol. 95, June 2019, pp. 149-162.
- [13] Irfan Allahi, Bilal Khan, Aamir Sohail Nagra, Rabbia Idrees, and Shahid Masud, *Performance Evaluation of IEEE 1588 Protocol Using Raspberry Pi over WLAN*, in *Proceedings of the 2018 IEEE International Conference on Communication Systems (ICCS 2018)*, Chengdu, China, December 2018.
- [14] Eric Gamess and Syed Shah, *Network Performance Evaluation of Several Raspberry Pi Models for IPv4 and IPv6*, in *Proceedings of the 2022 ACM Southeast Conference (ACMSE 2022)*, Virtual Event, USA, April 2022.
- [15] Trent Ford, Eric Gamess, and Christopher Ogden, *Performance Evaluation of Different Raspberry Pi Models as MQTT Servers and Clients*, *International Journal of Computer Networks & Communications (IJCNC)*, vol.14, no.2, March 2022.
- [16] Borislav Đorđević, Rade Furtula, and Valentina Timčenko, *VMware ESXi and Microsoft Hyper-V Hypervisor Performance Comparison*, in *Proceedings of the 2020 28th Telecommunications Forum (TELFOR 2020)*, Belgrade, Serbia, November 2020.
- [17] Hafiz ur Rahman, Guojun Wang, Jianer Chen, and Hai Jiang, *Performance Evaluation of Hypervisors and the Effect of Virtual CPU on Performance*, *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovations*, Guangzhou, China, October 2018.
- [18] Borislav Đorđević, Iva Jovičić, Nenad Kraljević, and Valentina Timčenko, *Comparison of Type-2 Hypervisor Performance on the Example of VirtualBox, VMware Workstation Player and MS Hyper-V*, in *Proceedings of the 2022 International Conference on Electrical, Electronic and Computing Engineering (IcETRAN 2022)*, Novi Pazar, Serbia, June 2022.
- [19] Ashish Lingayat, Ranjana R. Badre, and Anil Kumar Gupta, *Performance Evaluation for Deploying Docker Containers on Baremetal and Virtual Machine*, in *Proceedings of the 2018 International Conference on Communication and Electronics Systems (ICCES 2018)*, Coimbatore, India, October 2018.
- [20] Pezhman Sheinidashtegol and Michael Galloway, *Performance Impact of DDoS Attacks on Three Virtual Machine Hypervisors*, in *Proceedings of the 2017 IEEE International Conference on Cloud Engineering (IC2E 2017)*, Vancouver, British Columbia, Canada, April 2017.

- [21] Sebouh Toumassian, Rico Werner, and Axel Sikora, Performance Measurements for Hypervisors on Embedded ARM Processors, in Proceedings of the 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI 2016), Jaipur, India, September 2016.
- [22] Raspberry Pi Foundation, Raspberry Pi 3 Model B, <https://www.raspberrypi.com/products/raspberry-pi-3-model-b>.
- [23] Raspberry Pi Foundation, Raspberry Pi 3 Model B+, <https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus>.
- [24] XiaoXiao Bian, Implement a Virtual Development Platform Based on QEMU, in Proceedings of the 2017 International Conference on Green Informatics (ICGI 2017), Fuzhou, China, August 2017.
- [25] 7-Zip, <https://www.7-zip.org>.
- [26] André Rosete, Kenneth Baker, and Yao Ma, “Using LZMA Compression for Spectrum Sensing with SDR Samples,” in Proceedings of the 2018 9th IEEE Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON 2018), November 2018, pp. 282–287, DOI: 10.1109/UEMCON.2018.8796574.
- [27] Debian Project, Meeting Minimum Hardware Requirements, <https://www.debian.org/releases/buster/amd64/ch03s04.en.html>
- [28] PassMark Software, PerformanceTest, <https://www.passmark.com/products/performancetest>.
- [29] Darren James Harkness, Apache Essentials: Install, Configure, Maintain, Second Edition, Apress, August 2022.
- [30] Botan: Crypto and TLS for Modern C++, <https://botan.randombit.net>.
- [31] Massimo Bertaccini, Cryptography Algorithms: A Guide to Algorithms in Blockchain, Quantum Cryptography, Zero-knowledge Protocols, and Homomorphic Encryption, Packt Publishing, March 2022.
- [32] Announcing the Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, United States National Institute of Standards and Technology (NIST), November 2001.