AJSE

American Journal of Science & Engineering

Volume 3 Issue 1

June 2022

American Journal of Science & Engineering (AJSE)
Society for Makers, Artists, Researchers and Technologists (SMART)
6408 Elizabeth Ave SE, Auburn 98092, Washington, USA
ISSN: 2687-9530 (Print) and 2687-9581 (Online)

Editor-in-Chief

Dr. Izzat Alsmadi

Texas A&M, San Antonio, USA

Research Interest: Cyber Intelligence, Cyber Security, Software Engineering, Social Networks

Bio: Izzat Alsmadi is an Assistant Professor in the department of computing and cyber security at the Texas A&M, San Antonio. He has his master and PhD in Software Engineering from North Dakota State University in 2006 and 2008. He has more than 100 conference and journal publications. His research interests include: Cyber intelligence, Cyber security, Software security, software engineering, software testing, social networks and software defined networking. He is lead author, editor in several books including: Springer The NICE Cyber Security Framework Cyber Security Intelligence and Analytics, 2019, Practical Information Security: A Competency-Based Education Course, 2018, Information Fusion for Cyber-Security Analytics (Studies in Computational Intelligence), 2016. The author is also a member of The National Initiative for Cybersecurity Education (NICE) group, which meets frequently to discuss enhancements on cyber security education at the national level.

A $49 Aerospace Cybersecurity Lab: RF Data Communications for Undergraduate Cybersecurity Education

Richard Hansen
Capitol Technology University, USA
rhhansen@captechu.edu

Zachary Klein
University of Maryland, USA
zklein@umd.edu

***Abstract* – Radio Frequency (RF) communications are essential for aircraft and satellite operation. The current generation of Software Defined Radios (SDRs) has increased the capabilities of equipment at a cost that brings this technology in reach for economically disadvantaged institutions and students. Labs are designed to provide immediate feedback to accelerate experimentation and learning. The novelty and technology associated with aerospace platforms can stimulate students' interest and motivation. These lessons can be applied to other attack surfaces such as automotive and 5G communications.**

Keywords
Radio Frequency, RF, Software Defined Radio, SDR, aviation, satellite, aerospace, data communications, cybersecurity, ADS-B, ACARS

1. Introduction

Radio frequency (RF) communications can be an attack surface for safety- and mission-critical systems and utilities [1] [2]. Cybersecurity programs often present RF communications through discussions of Wifi and cellular phone networks without providing instruction in RF fundamentals [3]. Undergraduate RF engineering courses are an alternative. They typically require a full semester and require advanced coursework in engineering mathematics which are not a part of many cybersecurity programs. [4] This paper proposes the use of software-defined radios (SDRs) to teach RF fundamentals through reception and exploitation of aerospace data communications. The aviation community has recognized the need for cybersecurity services [5] as has the United States Department of Defense (DoD) satellite community [6].

Figure 1: Software Defined Radio - USB "Dongle" form factor

Software Defined Radios (SDRs) provide an inexpensive and straightforward method of providing RF education to aspiring Cyber professionals. A USB "dongle" radio costing $25 can be used with a desktop or laptop computer to receive aircraft data feeds and weather images from weather satellites. Aircraft and weather satellites can be received regularly from almost any location in the United States. The computing requirements are low, allowing economically disadvantaged institutions to provide an effective education with a very modest investment. SDRs provide a graphical representation of the frequency spectrum which is useful for learning and experimentation.

SDRs digitally sample the amplitude of the energy in a range of frequencies millions of times each second. The output is a series of scalar measurements which are processed on the computer. Algorithms in software are used to display graphical representations of signals and for demodulation of voice and data communications. Inexpensive SDRs were originally developed to receive digital television (DTV) on laptops and PCs. Experimentation showed that they could be tuned to other frequencies and software could be used to demodulate many different kinds of signals. There are many software packages for Windows, MacOS and Linux available [7].

The graphical representation of signals is important. As shown in *Figure 2* below, students can observe a dynamic picture showing frequency and amplitude information for all signals in the selected range.

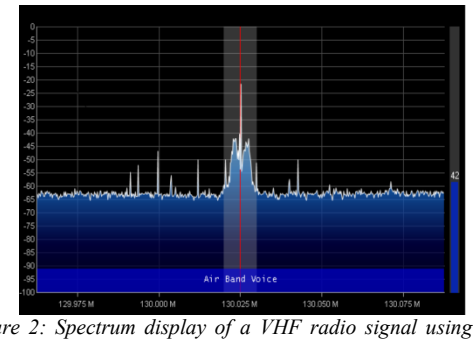


Figure 2: Spectrum display of a VHF radio signal using SDR# (SDR Sharp) software for Windows and a USB "dongle" SDR

The goal for exercises using SDRs is to develop competency at the first, second, and third levels (Remembering, Understanding, and Applying) of the Revised Bloom's [8]. Hands-on labs can teach students to classify RF signals by their characteristics, progressively realizing the goal of competency at Bloom's Level 3 (Application).

Jerome Bruner's Theory of Discovery proposes that students use their own past experiences and knowledge to discover new facts and relationships. As stated by McLeod [9], "Bruner proposes that learners' construct their own knowledge and do this by organizing and categorizing information using a coding system. Bruner believed that the most effective way to develop a coding system is to discover it rather than being told it by the teacher." The labs are designed to support and encourage experimentation with rapid feedback that progressively builds new knowledge.

Students can classify the components of their observations based on characteristics of signals such as frequency, bandwidth, modulation, and output. The sensory aspects, such as seeing real-time changes in frequency and amplitude and listening to the resulting output, help students rapidly discover the properties of different signal types and receiver settings.

On a wide scale the students can see separations between signals and relate this to existing knowledge of TV channels and FM station frequencies. Close-up views show changes in signals as they are modulated with information. This provides a method of learning RF fundamentals that is not based on mathematics. Some SDR software, such as GNU Radio [10], provide the ability to process different signal types by linking together code that can be represented as blocks on a diagram. Students can create new designs in a matter of minutes using software. This hands-on experimentation provides for rapid experimentation with immediate feedback on their actions.

RF can be further classified as a type of electromagnetic radiation (EM) that uses space itself as the medium for transmission, as does light. Other classifications for EM include electrical signals passing through an ethernet cable with metallic conductors, reflected light or radio signals carrying information about an object (size, distance) through space, and light carrying data signals in a fiber optic cable. When EM is used for communications its characteristics are classified as OSI Layer 1 properties, serving as a starting point for discussions of the OSI Model.

2. EXERCISE DESIGN

Receiving these signals is free of any requirements for licensing or registration. The activities below are legal anywhere in the United States and in many other countries. As mentioned above a key goal is development of competency up to Bloom's Level 4 through hands on experimental activities.

2.1 FM Broadcast Radio

FM broadcast radio is an example of analog communications and provides a good introduction to basic principles. The visual display of a range of frequencies shows how FM stations are grouped together in the same frequency space, a band, and they are separated by enough distance so their signals do not interfere, the separation is greater than the bandwidth.

The SDR# (pronounced SDR Sharp) [11] software can be used to display signals in the FM band and listen to the demodulated output. *Figure 3* below shows an example of the display. The X-axis shows the frequency of signals and the Y-axis shows relative strength on a logarithmic scale. The "waterfall" display at the bottom shows the strength of signals over time.

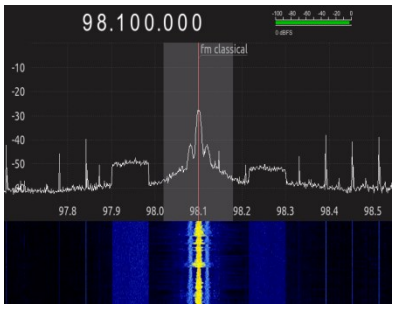


Figure 3: Spectrum display of an FM radio signal and adjacent signals using SDR# software and an RTL SDR Radio

Modest guidance can assist with developing classifications using this display. The term "band" refers to sets of adjacent frequencies that have common characteristics or share a common use. When receiving a signal in the FM broadcast band students can be encouraged to tune lower and higher in frequency until they could no longer find stations. They can also "zoom out" to larger amounts of spectrum to make the search easier. The term "band" and their own observations will help students create a new method of classifying information.

Bandwidth is another key characteristic of RF communications. It is generally related to quantity of information passing in a unit length of time (bits-per-second or a range of audio frequencies). The real-time display can be "zoomed in" to provide a very granular measurement of the upper and lower limits of spectrum used by the signal. Students may find the waterfall display is even more useful for this task.

Bandwidth is used to help define a channel. A channel has a center frequency as well as upper and lower frequency limits. Students can be told that in the US stations have 0.2 MHz/200 KHz channels. They can be asked to estimate the extra space left between channels and speculate on other features of the channel system.

2.2 Digital Aircraft Data

ACARS refers to the Aircraft Communications, Addressing and Reporting System [12]. It uses an analog voice channel to carry digital information. In the United States it can be received on a frequency of 130.45 & 131.5 MHz as shown in the left half of *Figure 4* below.

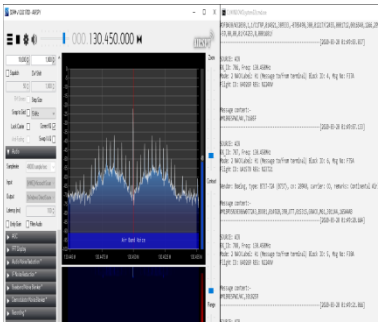


Figure 4: Spectrum display of ACARS signal and several decoded messages from a separate window

This image shows the use of SDR# software to receive the signal with AcarSDeco2 software [13] to demodulate the signal and provide the digital output. Students can hear "bursts" of digital information from SDR# and see the corresponding data output in another window.

The OSI model is applicable to both windows. SDR# is showing the characteristics of the OSI Layer 1 analog signal. The ACarsDeco2 window shows Layer 2&3 information (source address, sequence number, and contents of the message).

Students can be encouraged to tune away from the center frequency and see the effect on the signals. Other settings such as bandwidth and modulation can be changed and the effects observed.

After a period of experimentation, a Socratic dialogue can be held to ensure students are meeting the learning objectives of competency at Bloom's Level 3. They should compare and contrast their observations on these first two types of signals. Questions to stimulate discussions may include:

- Why are frequencies on one axis and power/strength on the other?
- How could color or other techniques be helpful in displaying the strength or other characteristics of a signal?
- What new insights do I have from the watching movement on the spectrum display and corresponding digital output?

A rule that proved effective was the "and" rule. After a student voices an observation, other students must precede their comments with the word "and." This creates a supportive feedback loop and encourages students to use their observations to build upon those of others.

The last activity is a "Point of View" exercise. It can be useful for students to view themselves as attackers (Red Team) and Defenders (Blue Team). Students can be challenged to modify network attack scenarios to the RF domain, such as Denial of Service (DoS) and spoofing. They can be challenged to suggest modifications to protocols that would provide resilience against these attacks.

2.3 Aircraft Location Information

Many aircraft are required to broadcast their location using the Automated Dependent Surveillance-Broadcast (ADS-B) technology on a frequency of 1090 MHz. ADS-B systems broadcast GPS-enhanced location and identification data to help air traffic controllers manage traffic. SDR# for Windows can be used with other packages to display location and identification information as shown in *Figure 5* below. On Linux, the Dump1090 [14] software can be used to receive and display information in textual format.

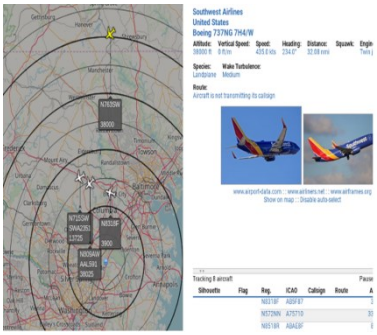


Figure 5: ADS-B display and aircraft information lookup

Figure 5 above shows the output from the Virtual Radar Server for Windows [15] software which displays aircraft identification, location and altitude information. Observation of the display can reveal from which directions and at what altitudes traffic can be received. *Figure 5* shows the coverage with an antenna placed in a north-facing window.

ADS-B is an excellent subject for a Red Team-Blue Team discussion. This service is critical for managing traffic and for collision avoidance. Spoofing and denial-of-service attacks could have severe consequences. An examination of the protocol in terms of the CIA triad will show students its lack of provisions for security. It also shows the need for cybersecurity professionals to be involved in designing these systems.

2.4 Satellite Data

The United States and other countries have a placed weather satellites in polar orbits approximately 500 miles above the earth. They take and transmit images showing cloud cover and other details using a format calls APT (Automatic Picture Transmission). The Heavens Above website [17] provides information for predicting satellite passes with detailed information. This is a challenging undertaking and may be impractical given the time constraints for undergraduate education.

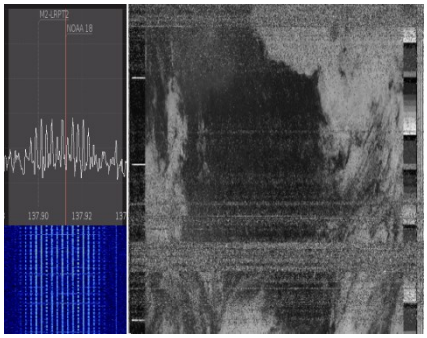


Figure 6: NOAA weather satellite signal and a decoded image

If *Figure 6* above, signals are displayed by using the GQRX [17] software for Linux to receive the signal and record the demodulated output to an audio file. The audio is then decoded with the NOAA-APT software [18]. Windows software such as SDR# can also be used with the Windows version of NOAA-APT.

The reception of satellite signals is much more challenging than the previous exercises. An unobstructed view of the sky is needed and the antenna and receiver should be outdoors. Signals can only be received from the time the satellite rises over the horizon (Acquisition of Signal or AOS) to when it

goes below the horizon (Loss of Signal or LOS). The Heavens Above website is one source for these details.

The challenges presented by satellite signals acts as a Capstone exercise. It provides opportunities to apply learning from the other exercises and to observe new phenomena such as doppler shift.

3. CONCLUSIONS & APPLICATION TO OTHER AREAS

The ability to see a representation of the RF spectrum in real-time is a powerful tool for teaching. Students receive immediate feedback on their actions and can quickly switch between macro and micro scales to examine frequencies and modulation. Students claimed that the ease of experimenting with SDRs made the exercises fun and interesting.

The SDRs can be used to view other unseen and unintentional communications. Electromagnet interference (EMI) is produced by digital circuitry in computer systems. It can interfere with the reception of radio signals and can also provide a means of covertly extracting information. Students can demonstrate this for themselves. An antenna can be created by wrapping one or more loops of wire around a laptop or desktop computer and connecting it to the SDR's antenna input. Students should then select AM modulation and reboot or power on the computer. By tuning through frequencies at the bottom of the SDR's range students will hear many different sounds as the system proceeds through the boot process. These signals have been exploited to extract encryption keys from a PC in under a minute [19].

RF-based telecommunications are used in many parts of our daily lives. Key fobs for automobiles and garage door openers often operate in the 315 MHz range. Inexpensive SDRs can be used to capture and investigate data transmissions used by these devices.

For a larger investment SDRs such as the HACKRF ONE, available on Amazon and Ebay, can both receive and transmit at frequencies up to 5 GHz. The RF output is low enough for legal use in the unlicensed spectrum used by these devices. In a shielded environment it can be used for more complex experiments with technologies such as GPS and cellular systems to include 5G.

4. EQUIPMENT LIST

These exercises are designed to be independent of the specific type of SDR used. A minimal set of equipment can be purchased for approximately $49. Included is a "BNC" cable adapter which provides a standard interface for use with experimental and ready-made antennas for specific purposes.

Software-Defined Radio, RTL-SDR R820T2, $24.99, Amazon
Purpose: Receives signals for display and demodulation on PC

RTL-SDR Dipole Antenna kit, $14.95, Amazon
Purpose: Captures electromagnetic energy for the receiver

BNC Female Adapter to MCX Male Connector, $8.99, Amazon
Allows us for use of home-made and commercial antennas.

5. ACKNOWLEDGEMENTS

The Maryland Space Grant Consortium (MDSGC) provided funding for a Computer Engineering student, Mr. Zachary Klein, to assist with research on use of Software Defined Radios and other technology for communications with High Altitude Balloons at the edge of space. The MDSGC has agreed to publication of information in this paper. Mr. Klein worked diligently and creatively to develop configurations and provide data. He assisted with input on the exercises and with creating graphics for this paper. Capitol Technology University was kind enough to allow the use of its campus and facilities for this research. Dr. Win Wenger provided valuable guidance on the works of Drs. Piaget and Burner and applying their methods. Drs. William Butler and Sandy Antunes provided much helpful input and encouragement.

REFERENCES

[1] Cyber Vulnerabilities & Mitigations in the Radio Frequency Domain. (n.d.). Retrieved April 3, 2020, from https://www.sbir.gov/node/1208173

[2] Sun, C.-C.; Liu, C.-C.; Xie, J. Cyber-Physical System Security of a Power Grid: State-of-the-Art. *Electronics* 2016, *5*, 40.

[3] Newhouse, W., Keith, S., & Scribner, B. (2017, August). National Initiative for Cybersecurity Education NIST 800-181, KSA K0274

[4] ENEE407: Design & Testing of RF and Microwave Devices. (n.d.). Retrieved March 20, 2020, from https://ece.umd.edu/course-schedule/course/ENEE407

[5] International Civil Aviation Organization, Addressing Cybersecurity in Civil Aviation, (n.d.). Retrieved March 20, 2020, from https://icao.int/cybersecurity/Documents/A40-10.pdf

[6] Barrett, B. (19AD, September 17). The Air Force Will Let Hackers Try to Hijack an Orbiting Satellite. Retrieved from https://www.wired.com/story/air-force-defcon-satellite-hacking/

[7] The BIG List of RTL-SDR Supported Software. Retrieved March 21, 2020, from https://www.rtl-sdr.com/big-list-rtl-sdr-supported-software/

[8] Mcdaniel, R. (2020, March 25). Bloom's Taxonomy. Retrieved from https://cft.vanderbilt.edu/guides-sub-pages/blooms-taxonomy/

[9] Mcleod, S. (2019). Bruner - Learning Theory in Education. Retrieved from https://www.simplypsychology.org/bruner.html

[10] The Free & Open Source Radio Ecosystem · GNU Radio. (n.d.). Retrieved March 25, 2020, from http://www.gnuradio.org/

[11] SDR# and Airspy Downloads. (n.d.). Retrieved April 5, 2020, from https://airspy.com/download/

[12] SKYbrary Wiki. (n.d.). Retrieved April 5, 2020, from https://www.skybrary.aero/index.php/Aircraft_Communications,_Addressing_and_Reporting_System

[13] Sergsero. (2018, December 8). AcarSDeco2. Retrieved April 5, 2020, from http://xdeco.org/?page_id=42

[14] ADS-B using dump1090 for the Raspberry Pi. (n.d.). Retrieved April 6, 2020, from https://www.satsignal.eu/raspberry-pi/dump1090.html

[15] Virtual Radar Server. (n.d.). Retrieved April 6, 2020, from http://www.virtualradarserver.co.uk/

[16] Heavens Above. (n.d.). Retrieved April 7, 2020, from https://www.heavens-above.com/

[17] GQRX (n.d.) Retrieved April 6, 2020, from https://gqrx.dk/download

[18] NOAA-APT (n.d.). Retrieved April 6, 2020, from https://noaa-apt.mbernardi.com.ar/download.html

[19] Thomson, I. (2017, June 24). AES-256 keys sniffed in seconds using €200 of kit a few inches away. Retrieved from https://www.theregister.co.uk/2017/06/23/aes_256_cracked_50_seconds_200_kit/

The development of a PPG and in-ear EEG device for application in fatigue measurement

John Robert Honiball
Stellenbosch University, South Africa
roberthoniballza@gmail.com

David Vandenheever
Mississippi State University, USA
davidvdh@abe.msstate.edu

***Abstract*-The need for proper fatigue detection and mitigation is made clear in research, with failure to detect fatigue resulting in significant societal health repercussions. Currently, there are limited hardware systems dedicated to the monitoring of fatigue-related biometrics. The devices that do attempt to provide this information are often impractical due to their size, required expertise and cost constraints. Access to these technologies by a broader population is therefore limited. Wearable health devices could provide a more practical solution. A data capture system was designed and implemented that records PPG and in-ear EEG information. The device was created to be inexpensive and portable. The in-ear EEG results obtained showed the detection of a statistically significant difference in alpha attenuation levels, which are closely associated with the state of alertness or drowsiness. While the acquired heart rate and blood oxygen saturation measurements showed a close correlation with an FDA approved pulse oximeter. Although the number of trials conducted was limited, the results show promising performance. This project is a stepping stone in the pursuit of an affordable fatigue monitoring solution that can mitigate the human-cost incurred on account of fatigue.**

I. INTRODUCTION

Fatigue can be described as a multidimensional phenomenon characterised by a deterioration of mental and physical performance. In modern industrial society, fatigue has become a common state in which people from all walks of life frequently find themselves. The most common fatigue symptoms include detrimental effects on reaction time, energy levels and the ability to focus [1]. High workplace demands, high stress levels, and insufficient sleep for extended periods increase the prevalence of an individual's fatigue and pose considerable societal health and safety risks [2], [3], [4], [5]. Over time, poor fatigue management increases the risk of work-related burnout, characterised by symptoms of extreme exhaustion, distress, and decreased effectiveness and communication [6]. Recently, job burnout has become an apparent cause of reduced working capacity in industrial and developing countries, with over 25% of employees exhibiting job burnout symptoms [7]. In dynamic and complex working environments involving extensive mental and physical demands, the need for fatigue management measures is clear [8].

Fatigue is a widespread societal problem and is currently one of the least researched human responses [9]. The lack of extensive research on fatigue is due to the inherent challenges found in fatigue research. These challenges include the absence of a well-defined theoretical framework, and a widely accepted definition of fatigue [10]. Therefore, fatigue has evolved into an umbrella term that comprises of various physiological, emotional, and behavioural factors that can result in chronic mental or physical states affecting an individual's capacity to perform tasks [11]. Fatigue has also become synonymous with terms such as drowsiness and sleepiness, which represent an intermediary stage between wakefulness and sleep [12].

The multifaceted nature of fatigue has made it difficult to measure and quantify, and as a result, there is no gold-standard measurement approach. The complex nature of fatigue requires measurements in different dimensions in the attempt to quantify it fully. The types of fatigue measurements can be divided into three groups: subjective, behavioural and objective measures [13]. Objective methods of fatigue measurement utilise well established diagnostic tools to monitor changes in vital signs. Commonly utilised methods include electroencephalogram (EEG), electrocardiogram (ECG) and photoplethysmography (PPG). For fatigue research, it is desirable to acquire neurological information in the time and frequency domain given the interconnection between fatigue and the brain [14]. EEG analysis has proven insightful in fatigue detection [15], [16], [17], [18]. Comparably, the body's cardiovascular function is also desirable given the connection between the autonomic nervous system and heart rhythms [19]. This relationship is the source of a group of useful biosignals such as heart rate (HR), heart rate variability (HRV), and average heart rate (AVR) [4]. Supplementary vital signs that can assist in fatigue detection include respiratory rate and blood oxygen

saturation, which a large variety of commercial sensors can measure[20]. Subjective measurements of fatigue have also been found to be useful in research [21], [22], [23]. These psychological self-report measures are dependent on the conditions being investigated. In the case of sleepiness, subjective sleepiness levels can be assessed by employing the Stanford Sleepiness Scale, and where fatigue is of interest, scales such as the Samn-Perelli checklist and Li's subjective fatigue scale can be used [24]. Other subjective measurements for fatigue that have been used include visual analogue scales (VAS), and the Brunel Mood Scale [17]. These subjective measurements provide meaningful insight into the way fatigue is experienced by individuals and has shown a positive correlation with objective measurements such as HRV and EEG [13], [14]. Behavioural or performance measurements are useful in fatigue research because they provide a means to evaluate fatigue's effect on an individual's decline in performance ability [25]. Simple reaction time measurements such as those utilised by the Psychomotor Vigilance Task (PVT) provide a repeatable metric used to evaluate decline in attention capacity [17], [26]. The accuracy of responses provided in tasks such as the AX-Continuous Performance Test (AX-CPT) and the Stroop Task also provide insight into the task's execution quality [27], [28], [23]. Along with objective and subjective measurements, these performance metrics should provide a reliable method to detect fatigue. These metrics would also be suitable for applications such as sleep monitoring, meditation, and general health [29], [30].

The process of measuring biometrics has often required excessive and large hardware setups and sufficient time away from daily responsibilities, but due to the large influx of wearable smart solutions, it has become much more possible to acquire the desired biometrics. Despite the increased popularity of wearable devices, there is yet to be a comprehensive monitoring system for fatigue-related biometrics. In this paper we present the development of an in-ear sensor that can potentially be used to monitor fatigue.

II. METHODS

A. EEG Hardware Design

Biosignals are often small in amplitude and contain undesired noise and interference. The accumulation of interference has an undesired effect of corrupting relevant information that may be of interest in the measured signal. Therefore, throughout the data acquisition pipeline, it is desirable that the data of the original biological signal remain uncontaminated. Since these signals contain vital information, the procedures of amplification, analogue filtering, and ADC are used for signal conditioning. The analog circuitry used to condition these analogue signals is often referred to as the analog front-end (AFE).

The design of a reliable, precision, high-accuracy ADC is not a trivial task, so commercial, off-the-shelf integrated circuits (IC) provide a decent base from which to start an AFE design. The use of ICs with integrated functions reduces the number of components required to design a data acquisition system. The use of ICs enables optimized and affordable designs with robust performance. The family of production-grade AFE produced by Texas Instruments is popular amongst researchers. Extensive evaluation of the AFE has been conducted in research [31], [32]. An initial prototype was developed in order to verify the schematic design and functioning of the IC before moving forward to a PCB design. The layout of the resulting PCB design was chosen such that the EMI is minimized by using bypass capacitors and separating analog and digital signal layers. The finished PCB design can be seen in figure 1. Three different in-ear electrode implementations were considered for the EEG signal acquisition. The features that were considered in order to determine the electrode feasibility were: Electrode contact; Comfort; Sensor material; Cost of production; Reusability; and Feasibility.

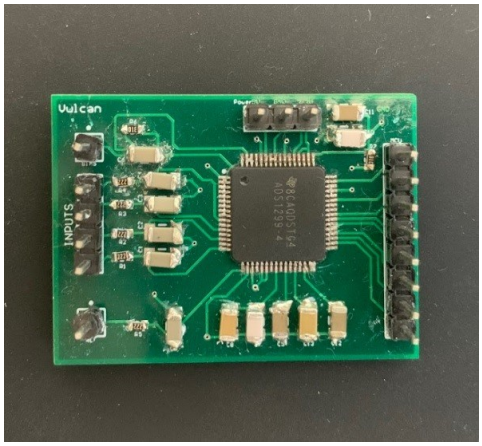


Figure 1: Finished PCB

With in-ear electrodes, there is no gold standard or design that works for everyone. The electrodes implemented need to be able to ensure reliable contact and low-impedance over time. The sensitive nature of the ear also requires that the electrodes be comfortable and safe to wear. There are two main approaches in electrode manufacturing: custom and generic. Custom electrodes provide the benefit that they are personally fitted for the specific participant and therefore ensure good contact, but they are expensive and difficult to manufacture.

Generic electrodes provide a more accessible and cheaper way to approach signal acquisition while sacrificing signal quality and comfort. Wet electrodes are often used to reduce the ESI by using conductive gels and improving signal acquisition. These conductive gels, however, dry over time, and their adhesion is easily lost during movement. Dry electrodes do not utilize an electrolytic substance and make direct contact with the skin. The main advantages of dry electrodes are that they are fast to place, do not require any additional instruments, and do not require extensive clean-up. There is also a clear distinction between active and passive electrodes. Active electrodes employ electronic circuitry between the sensor and the wire. The active electrodes provide a means to reduce possible line noise and therefore improve signal quality. The electronic elements used could add to noise if not correctly implemented. The cost of active electrodes is also high due to the circuitry's low noise and size requirements. Passive electrodes do not utilize circuitry between the sensor material and the wire. The last aspect of the electrode that needs to be considered is the sensor's material that will be the primary interface between the wire and the skin. The electrical and mechanical properties of the sensor material directly influence the design of the electrode.

The first electrode implemented was a custom electrode, as seen in figure 2, which was constructed by taking an impression of the ear to create an ear mould. The ear mould is then used to create a silicone-based earpiece. An audiologist often uses this process to create hearing aids. The ear mould is then used to create a silicone impression of the ear. Custom electrodes are desirable for their comfort and their ability to provide better contact over time. The downside of custom earpieces is that they are expensive to manufacture and that the time spent manufacturing them is also extensive. The electrode material selected for this implementation is conductive cloth. Cloth electrodes are highly flexible, conductive, soft, comfortable to wear and conform to changes in the earpieces' shape. The larger surface area of the custom earpiece makes it possible for the reference electrode and signal electrode to be located in the same ear. The reference electrode's location is targeted at the concha cymba, which has been shown to be an effective reference location in recent years [33], [34], [35]. The woven fabric has a low resistivity of 0.5 Ω/sq, making it highly conductive and formed into any shape required for the implementation.

The woven fabric is fixed to the earpiece substrate using a conductive adhesive. Soldering on fabric is infeasible; therefore, copper foil is implemented to bridge the wire and the fabric electrodes. Shielded cables were also used to enhance artifact rejection. The shielded wire's outer copper mesh was connected to the neutral part of the custom earpiece. It was found that the custom earpiece is comfortable to wear; however, the rigid nature of the earpiece made it prone to lose skin contact, which affects the signal quality.

The second earpiece that was considered was a foam earpiece design. Memory foam has the beneficial characteristic of being viscoelastic. The memory foam substrate's viscoelasticity ensures that an electrode can be safely and effectively inserted further into the ear canal than the custom earpiece. The earpiece can be compressed into a smaller shape and then inserted into the ear canal. Following insertion, the earpiece expands and redistributes pressure evenly along the entirety of its contact surface, thus providing a stable interface to the ear canal wall. The substrate's viscoelasticity also ensures that energy from abrupt motion such as pulsation is absorbed, thus minimally disturbing the electrode-skin contact. Sound-blocking earplugs utilize memory foam and are used for the viscoelastic substrate for this design. The electrodes must have similar flexible properties to accommodate the compression of the memory foam earplugs before insertion. A 6 *mm* x 10 *mm* square of conductive cloth is used as the electrode interface's top layer. Soldering directly onto conductive cloth would damage the material; therefore, the copper tape is used as a bridging material onto which wire can be soldered. A silicone ear hook was also implemented to assist in the contact functionality of the earpiece. An ear hook is a mechanical hook often found in hearing aids and sports earphones that improves an earpiece's stability during functioning. The implementation of the earpiece can be seen in figure 3.

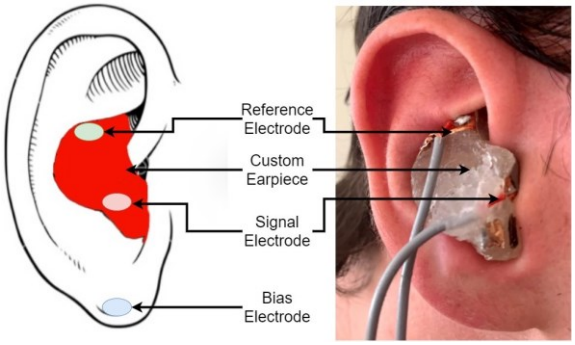
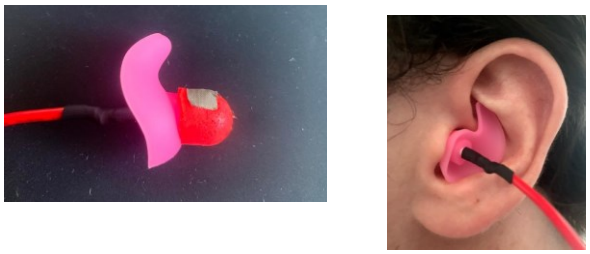


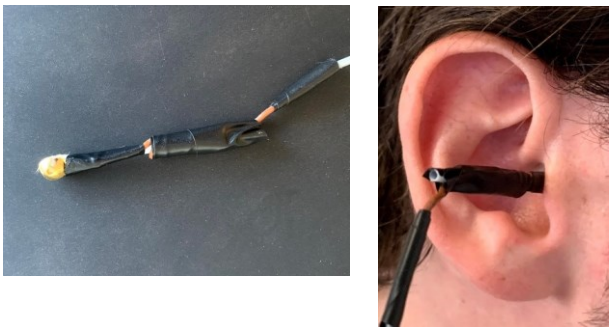
Figure 2: Custom Implemented earpiece conceptualization and Implementation with cloth electrodes



(a) (b)

Figure 3: (a) Foam earpiece implementation (b) Foam earpiece in ear

The final electrode that was implemented that provided the most consistently good results was the deep ear electrode. Utilising a smaller conductive area, the electrode can be inserted deeper into the ear canal. Inserting the electrode deeper into the ear canal provides an increase in signal quality and contact while possibly increasing the discomfort experienced. A gold-cup electrode was modified by reducing the overall size of the electrode and then positioned on an earbud to assist in positioning the electrode in the ear canal. The implemented deep ear electrode can be viewed in figure 4.



(a) (b)

Figure 4: (a) Deep ear electrode implementation (b) Deep electrode in ear

B. PPG Hardware Design

The typical PPG device contains a light source and a photodetector. The light source emits light onto the tissue, and the photodetector measures the light that is reflected or absorbed, depending on the configuration used. The most common PPG sensors are pulse oximeters used to measure blood oxygen saturation (SpO2). The choice of measuring mode would directly influence the viability of measuring sites. PPG sensors are commonly worn on the fingers in transmission mode due to the high signal amplitude that can be achieved compared to other sites [36]. This measuring site is not well-suited for extended sensing, as many activities of daily life involve the use of fingers. Earlobes are not compromised of cartilage and thus contain sufficient blood supply. Moreover, earlobes are far less vulnerable to the effects of motion artefacts compared to other extremities. Implementing a reflective PPG sensor with an ear clip on the earlobe was decided to be a suitable solution. The AFE that was selected for implementation is the MAX30102 from Maxim Integrated. The MAXREFDES117 is a reference design that utilizes the MAX30102 as an optical module and an integrated power supply and level translator. The MAXREFDES117 is an ideal sensor solution given its small size (12.7 *mm* x 12.7 *mm*) and low power design. A prototype was developed to use the earlobe as a measuring site. Clothes pegs provided an affordable solution to provide consistent contact to the earlobe for the MAXREFDES117 sensor module.

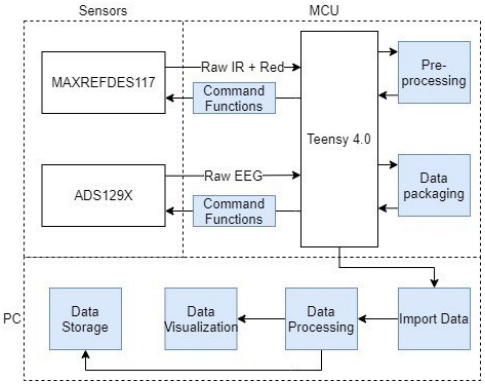


Figure 5: High-level Software Functional block diagram

Following successful prototyping, it was decided to implement a dedicated ear-clip design to ensure simpler reproducibility and improved sensor placement. The principle design of a clip is straight forward and ensures good contact. An initial clip design was implemented in Autodesk Inventor, featuring a dedicated sensor module location and a bias electrode location for the EEG setup.

C. Software Design

The implemented software was written for the MCU and the PC processing the data. The MCU software is based on *C++* and was developed using the Arduino Integrated Design Environment (IDE). The MCU software handles sensor communication, timing, selected processing functions and transmission of collected data. The PC software was developed in Python using Jupyter Notebook. The PC software reads the saved data, performs most of the processing required, and then displays the data stored

on the local hard drive. The high-level functional block diagram can be seen in figure 5.

Table I: Power supply verification by voltage probing

Voltage Probe	Expected Value	Measured Value
VREF	VREFP - VREFN = 4.5 *V*	4.496 *V*
VCAP1	GND + 1.2 V = 1.2 *V*	1.196 *V*
VCAP2	(AVDD + GND)/2 = 2.5 *V*	2.532 *V*
VCAP3	AVDD + 1.9 V = 6.9 *V*	6.941 *V*
VCAP4	VREF/2 = 2.25 *V*	2.249 *V*

Table II: Measured Noise Measurements compared to theoretical values for sample rate of 500 *SPS*

Input-referred noise Parameter	Datasheet (μV)	Measured value (μV)
V_{PP}	1.39	1.35
V_{RMS}	0.20	0.18

D. System Verification Tests

The design and implementation of the system were evaluated by performing incremental verification tests of the different subsystems. The tests are performed to identify the strengths and shortcomings of the designs. The AFE represents the intersection of analog and digital data. In the Software Design and Implementation section, successful communication between the AFE and MCU was verified by configuring the AFE and retrieving the Device ID. Verification of the analog data is still required to assess the functionality of the AFE. A suitable power supply to the AFE is crucial for acquiring analog data and is verified by measuring the internal voltage pins or the potential difference across the bypass capacitors. The bypass capacitors are essential for noise reduction in the AFE and, therefore, the output data quality. Table I shows the VREF voltage probe and the different bypass capacitors (VCAP1-VCAP4) and their expected voltages representing the appropriate power supply. The measured voltages across the bypass capacitors are also shown in table I. The resulting measurements show that the power distribution of the PCB design is satisfactory. Low noise levels from hardware improve signal acquisition for low-amplitude biosignals applications such as EEG. The noise measurements for the hardware was performed by shorting the inputs of the AFE and acquiring 5000 readings at a sample rate of 500 *SPS*. The test was repeated five times, and the results obtained were averaged over the number of tests. The resulting input-referred noise plot in the time domain can be seen in figure 6. The μV_{pp} value was calculated as 1.35 μV, and correspondingly the μV_{RMS} was calculated as 0.1856 μV. Table II compares the calculated noise measurements to the values stated in the ADS1299 datasheet.

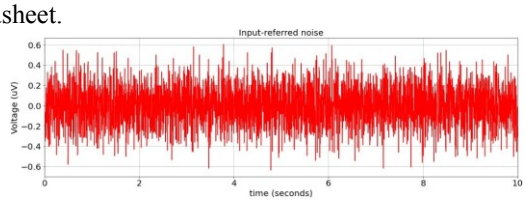


Figure 6: Averaged Input-referred Noise of AFE

E. EEG Measurements

The AAR was tested over twenty trials on one participant. Each trial's data was divided into the respective rhythm frequencies and then averaged over the twenty trials. The procedure was to keep the eyes open for 30 seconds and then close them for 30 seconds while in-ear EEG data is recorded.

F. PPG Measurements

For the testing of the SpO2 and HR parameters performance, data was simultaneously captured by an IMDK C101A3 pulse oximeter. The C101A3 is Food and Drug Administration (FDA) approved with an accuracy of ±2 % for SpO2 measurements and accuracy of ±2 *bpm* for heart rate data. The C101A3 uses fingers as measurings sites and utilises a transmissive PPG mode. For the measurements, the C101A3 was placed on the middle finger of the dominant hand. Trails were conducted to test the parameters' accuracy and consisted of a measurement period of 5 minutes per trail. The essential HR parameter tested is the instantaneous heart rate (IHR) or the Inter-Beat-Interval (IBI) HR-related values. This is because HRV is directly calculated from the IBI values. IHR uses the time difference between beats to estimate the heart rate.

III. Results

A. EEG Measurements

The resulting mean band amplitudes can be seen in figure 7 for eyes open and in figure 8 for eyes closed. Additionally, a single trail's frequency response can be seen in figure 9 to illustrate the increased activity in the alpha frequency range (8-13 *Hz*). Trails,

where excessive noise was present were discarded and repeated. The statistical analysis of the alpha band values over the twenty trails can be found in table III.

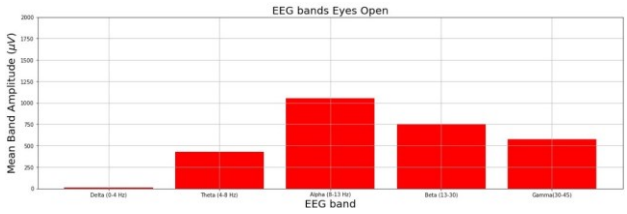


Figure 7: Mean band amplitude for EEG bands while eyes are open averaged over 20 trails.

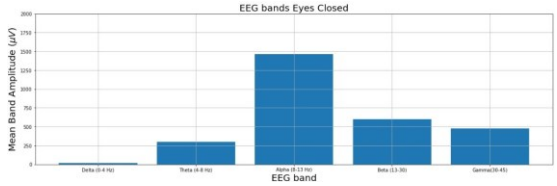


Figure 8: Mean band amplitude for EEG bands while eyes are closed averaged over 20 trails

Table III: Statistical analysis of EEG alpha band values over 20 trails

	Eyes Open	Eyes Closed
Mean	1466.47	1109.78
Standard deviation	600.14	399.99
p-value		0.0378

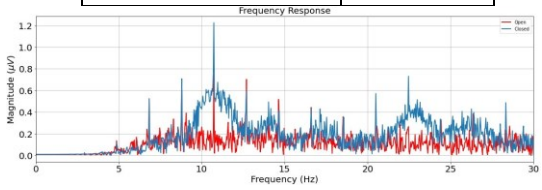


Figure 9: Frequency Response of an alpha attenuation trail

Figure 10: Comparison of raw IHR calculated values and HR values of C101A3 for trail period of 5minutes

B. PPG Measurements

Figure 10 compares the values between the raw IHR values obtained by the implemented PPG sensor and the calculated HR values of the C101A3 sensor for a single trail. While figure 11 shows the IHR values of the C101A3 sensor and the conditioned IHR values. The AHR for the trail illustrated in figures 10 and 11 was calculated as 65.2 *bpm*, while the C101A3 provided an AHR of 66.3 *bpm*. Five trials were performed and found similar results.

The typical range of SpO2 measurements for a healthy individual is between 95-100%. Measurements obtained from the IMDK pulse oximeter during the trials never dropped to under 95% as only healthy individuals participated in the trials. Similarly, the values obtained from the PPG sensor was also within the range of 95-100% and within a ±2 % error rate when compared with the IMDK pulse oximeter, as can be seen in table IV. The steps required for the proper inducing of hypoxia to achieve SpO2 levels under 95% are beyond the scope of this project and require additional equipment and ethical approval because of the associated health risks.

Figure 11: Comparison of conidtioned IHR values and HR values of C101A3 for trail period of 5 minutes

Table IV: Average SpO2 values over five trails

Trail	Average IMDK SpO2 (%)	Average PPG Sensor SpO2(%)
1	98.3	99.1
2	96.4	98.7
3	98.4	99.3
4	97.9	95.4
5	96.2	98.3

IV. DISCUSSION

A. *Interpretation of Results*

The results obtained provide insight into the system's functioning. The tests were performed to assess the validity of the designed systems' performance and its feasibility as an appropriate data capture device.

The first essential tests performed were the tests to verify the proper functioning and performance of the designed and implemented AFE. The first test was to verify the appropriate power supply to the AFE. The results obtained from the power supply verification was satisfactory and showed that the AFEs power requirements had been met. This is a necessary test to perform as improper power would affect the general functioning of the AFE, as the internal amplifiers, internal reference, and bias drive's performance would be compromised. After verifying the power supply, the noise performance of the AFE was evaluated. The system's input-referred noise was measured and compared to expected performance values, as stated in the ADS1299-x datasheet. The results obtained showed improved and comparable noise performance to that stated in the datasheet. The noise measurements show that the steps taken in the design process, such as using multiple PCB layers to minimize the noise between digital, analog and power signals, have ensured minimal system noise. The internal amplifiers' functioning was tested by generating an internal test signal fed into a selected channel's PGA. The internal test signal was successfully obtained and verified the connection between the PGAs and the internal ADC. The final verification test of the AFE consisted of supplying the system with a known external signal. The applied signal was successfully received and processed, verifying that the system setup, external input pins and analog filtering are functioning correctly. The collection of the results attained from the verification tests confirms that the AFE is working as intended and satisfies the performance requirements.

The two main components of the EEG system are the AFE and the in-ear electrodes. The AFE function has been verified; therefore, the focus shifted to electrodes and their performance. After extensive testing, it was found that the deep ear electrode provided the most consistent quality results. The custom electrodes did not provide constant contact over time and therefore yielded EEG signals contaminated with noise.

Similarly, it was found that the foam electrodes initially performed acceptably but suffered degradation of signal quality after extended use. The conductive cloth utilized as electrode material in the custom and foam electrode implementations provided more comfort but at the cost of signal quality. The gold material of the deep ear electrode implementation offers superior and consistent signal quality at the expense of user comfort. The deterioration in the conductive cloth's performance is likely due to the damaging of the conductive fibres, which increase the electrode's impedance and diminishes the signal quality. In contrast, the gold electrode's impedance is comparatively stable and therefore provides consistent quality signals.

When considering EEG analysis, the division is usually made between time and frequency domain based methods. After careful experimentation, it was found that the chosen approach did not compliment time-domain analysis methods such as the detection and analysis of MLAEP amplitude and latencies and was therefore not utilized. MLAEP responses are usually in the low (0-2) μV range and need to be captured 100 *ms* after the auditory stimulus. While MLAEP has seen promising results in monitoring anaesthesia levels, when considering the fatigue aimed application of the project, which inherently would require the participant to be awake, the MLAEP would too easily be contaminated with general physiological artifacts. The frequency analysis methods performed considerably better and have shown to be more resilient to artifacts. The alpha attenuation response is one of the methods chosen for detecting fatigue because alpha activity (8-13 *Hz*) increases when the eyes are closed and has been closely linked to the state of alertness or drowsiness. The tests performed yielded desirable results and showed an increase in alpha-band activity for the duration during which the eyes were closed. A t-test was performed on the alpha band data collected over twenty trials. It resulted in a p-value of 0.0378, which confirms that the device's results have a scientific, statistical significance.

The PPG measurements consisted of testing the IHR and SpO2 calculation of the implemented sensor configuration. The procedure consisted of determining the IHR and SpO2 values for 5 minutes while simultaneously recording the IMDK C101A3 pulse oximeter's HR and SpO2 values. This approach's motivation is that if the IHR values are accurately obtained, then the HRV values can easily be determined from the IHR values. For proper HRV implementation, a baseline measurement is required, and then the deviation from this baseline over time provides meaningful HRV data. The IHR values obtained from the trials were satisfactory and closely correlated with the IMDK pulse oximeter's HR values. Similarly, the SpO2 measurements of the implemented PPG sensor and the IMDK pulse oximeter were also very closely correlated. The utilization of the earlobe as a measuring site proved to be quite ideal as the site is less vulnerable to the effects of motion artifacts when compared to other measuring sites. This is because the earlobes do not contain cartilage, bone or

muscles and have sufficient blood supply. When considering the results obtained, the implemented earlobe PPG sensor would provide good HR and SpO2 biometrics for fatigue detection.

B. Comparison with Previous Literature

The overall project compares well with the findings of previous literature. Concerning the hardware development, there has been an increased interest in alternatives to conventional EEG systems. The most common implementations in recent EEG studies are off-the-shelf solutions such as the Open BCI biosensing boards and the more expensive medical grade bio-amplifiers. These EEG solutions offer adequate performance and have been shown to produce comparable results to conventional EEG systems.

The inherent dilemma of using a general EEG solution is that it is not application-specific. In-ear EEG studies also face this predicament and often use a general EEG solution, which leads to the underutilization of certain features and limits the use in specific research fields due to the obtrusive hardware. This is contradictory, given that the entire point of using in-ear EEG is to provide more flexibility to EEG research. Therefore, there is a clear need to develop dedicated hardware platforms to enable more mobile data acquisition. This project aims to contribute to this endeavour. There are very few dedicated hardware platforms for the in-ear EEG method as most in-ear EEG research focuses on electrode development and verification. Where in-ear fatigue, sleep, and drowsiness research is concerned, the utilized hardware is commercially available solutions [37], [33], [38], [27], [30]. The performance of the implemented EEG hardware in the project compares favourably with that found in the literature, exhibiting similar noise performance and can provide high-resolution EEG signals at a more than satisfactory sampling rate. Given the system's increased mobility, it is consequently more susceptible to environmental noise, which could degrade the system's performance if not adequately mitigated with shielding and filtering techniques.

The electrodes implemented in the project covered three different approaches that have been explored in research. The custom earpiece approach has proved to be a reliable method in the literature, given the excellent electrode contact and comfort. The custom earpiece in this project did not perform as well as the custom earpieces in the literature. This is mainly because the implemented custom earpiece featured some manufacturing limitations, such as not extending deeper into the ear canal. The utilization of conductive cloth as electrode material in literature has shown desirable results; however, the results obtained were inconsistent in the foam and custom earpieces. The most reliable electrode was the deep ear electrode and compared well with what was found in the literature. Inserting an electrode deeper into the ear canal provides better signal quality as it is closer to the source. The excellent conductivity of gold plated electrodes aids the signal quality obtained. The adequate electrode choice currently in research depends on the selected application and setting where the measurements will be taken. In-ear electrodes still need to be thoroughly researched in practice to determine a gold standard approach.

The results obtained from the selected EEG paradigms also compared well with tests conducted in research. It was shown that the AAR does indeed show an increase in alpha activity when the eyes are closed which is significant because the higher alpha band levels have been proven to be associated with a state of alertness or drowsiness [39]. This supports the use of the paradigm for fatigue detection application. Similarly, the ASSR results were desirable and illustrated the system's functionality and compare well with what has been found in other research studies. The system's shortcoming to detect MLAEP in a mobile setting is supported by research conducted by [40] and illustrates why MLAEPs are effective in idle applications such as an aesthesia monitoring.

The results obtained from the PPG measurements have also correlated well with the expected results from the literature. The choice of the earlobe as a measuring site for HR and SpO2 related measurements is suitable and is supported by similar findings in the literature [36]. The difficulty with verifying SpO2 measurements over an extended range (<95%) where hypoxia occurs is also a prevalent obstacle in SpO2 related studies.

C. Future Work and Improvements

From an EEG hardware perspective, the AFE is unmistakably one of the project's greatest strengths. The AFE provides high-quality EEG data in a small package, making it ideal for mobile applications, especially compared to off-the-shelf solutions and conventional EEG setups. The EEG hardware is also modular. It has a changeable sampling speed and support for up to four input channels giving it more versatility and ease of use if implemented in different applications. The weakness of the EEG hardware is the implemented electrodes. The variety of electrodes tested throughout the project emphasise the difficulty of implementing a reliable passive dry electrode for in-ear EEG data acquisition. The deep ear electrode that provided the most accurate and reliable results sacrificed comfort and would therefore not be convenient to wear for extended periods. The PPG hardware is also a distinct strength of the system.

Employing the earlobe as a measuring site with the implemented ear-clip provides comfortable and accurate measurement for extended periods. Some inherent disadvantages of most biometric acquisition systems are also featured in the hardware implementation, such as the susceptibility to noise and atrifacts despite passive noise reduction techniques. Another distinct advantage of the system is its cost. The entire system's hardware costs accumulate at around ± R2750 ($185), which is a decent reduction in price compared to the highly used 4-channel Open BCI ganglion board, which costs $250, and the project hardware also has PPG measurements. The system as a whole needs to be powered from a PC in its current state and does not feature onboard memory storage, which limits its existing range of applications.

The project achieved the objectives which it set out to do. However, there is the opportunity to improve the future versions of the device and similar devices. Some hardware improvements can be made to the system, which would significantly enhance the range of applications for which it can be used. These improvements include but is not limited to wireless connectivity, display for status monitoring, onboard memory and independent power supply. Implementing these features would significantly improve the ease of use and help isolate the system from the environment. The electrodes are a component where there is much room for improvement and innovation. Implementing a modular active electrode topology instead of passive electrodes would also assist in improving signal quality. Further optimisation and minimization of the hardware components would also help in promoting a mobile system. Additionally, more extensive testing can be performed. The complex nature of fatigue provides the opportunity to investigate multiple conditions associated with a state of drowsiness or sleepiness. These states' influences on the biometrics obtainable from the device could prove beneficial for future fatigue detection applications.

V. CONCLUSION

Fatigue is a condition developed as a result of extended wakefulness, increased workload, and sleep loss and is accompanied by the characteristic deterioration of mental and physical performance [13], [10]. In today's modern society, the demands of the workplace are extensive, and not enough attention is given to fatigue management. For some people, the improper management of fatigue has few societal consequences, while for others, such as doctors and drivers, the consequences can be severe. In order to effectively manage fatigue, it is first necessary to be able to detect it. Research has shown that fatigue is closely related to drowsiness and sleepiness, which can be identified with the correct combination of physiological and psychological biosignals [12], [41]. The tools required to measure these signals effectively are obtrusive and are impractical for daily monitoring. Furthermore, these tools are often expensive, limiting access to them [28]. The need for a low-cost and effective data capture device aimed at fatigue-related biosignals is evident. This project aimed to address this need. In-ear EEG provides an acceptable alternative to conventional EEG measures and has become increasingly popular due to its accessibility. Similarly, PPG technology has become prevalent in wearable health solutions and provides accurate cardiovascular and respiratory information. A data capture device was implemented utilizing in-ear EEG and PPG technology. The data capture device was designed with cost and size in mind to promote usability. Extensive testing was performed with the device to test its utility in the detection of fatigue-related biosignals. It was found that the device does measure the chosen fatigue-related biosignals with satisfactory accuracy. This is a significant result as it reinforces the feasibility of a fatigue monitoring device. Continued efforts to improve the understanding and detection of fatigue can ultimately lead to a device capable of mitigating the human cost of fatigue-related afflictions.

REFERENCES

[1] J. L. Axelsen, U. Kirk, and W. Staiano, "On-the-Spot Binaural Beats and Mindfulness Reduces the Effect of Mental Fatigue," *Journal of Cognitive Enhancement*, jan 2020.

[2] G. Borragán, C. Guerrero-Mosquera, C. Guillaume, H. Slama, and P. Peigneux, "Decreased prefrontal connectivity parallels cognitive fatigue-related performance decline after sleep deprivation. An optical imaging study," *Biological Psychology*, vol. 144, pp. 115–124, may 2019.

[3] C. Sugden, T. Athanasiou, and A. Darzi, "What Are the Effects of Sleep Deprivation and Fatigue in Surgical Practice?," *Seminars in Thoracic and Cardiovascular Surgery*, vol. 24, no. 3, pp. 166–175, 2012.

[4] S. Huang, J. Li, P. Zhang, and W. Zhang, "Detection of mental fatigue state with wearable ECG devices," *International Journal of Medical Informatics*, vol. 119, no. August, pp. 39–46, 2018.

[5] H. M. Gavelin, A. S. Neely, T. Dunås, T. Eskilsson, L. S. Järvholm, and C. J. Boraxbekk, "Mental fatigue in stress-related exhaustion disorder: Structural brain correlates, clinical characteristics and relations with cognitive functioning," *NeuroImage: Clinical*, vol. 27, jan 2020.

[6] B. B. Staples, A. E. Burke, M. Batra, K. J. Kemper, A. Schwartz, P. M. Wilson, C. J. Schubert, J. D. Mahan, and J. R. Serwint, "Burnout and Association With Resident Performance as Assessed by Pediatric Milestones: An Exploratory Study," tech. rep., 2020.

[7] L. Sokka, M. Huotilainen, M. Leinikka, J. Korpela, A. Henelius, C. Alain, K. Müller, and S. Pakarinen, "Alterations in attention capture to auditory emotional stimuli in job burnout: An event-related potential study," *International Journal of Psychophysiology*, vol. 94, no. 3, pp. 427–436, 2014.

[8] J. A. Caldwell, J. L. Caldwell, L. A. Thompson, and H. R. Lieberman, "Fatigue and its management in the workplace," jan 2019.

[9] T. C. Rosenthal, B. A. Majeroni, R. Pretorius, and K. Malik, "Fatigue: An overview," 2008.

[10] L. J. Tiesinga, W. N. Dassen, and R. J. G. Halfens, "Fatigue: A Summary of the Definitions, Dimensions, and Indicators," tech. rep., 1996.

[11] D. F. Dinges, "An overview of sleepiness and accidents," *Journal of Sleep Research*, 1995.

[12] M. Hirshkowitz, "Drowsiness," in *Encyclopedia of the Neurological Sciences*, 2014.

[13] J. F. Hopstaken, D. van der Linden, A. B. Bakker, and M. A. Kompier, "A multifaceted investigation of the link between mental fatigue and task disengagement," *Psychophysiology*, vol. 52, pp. 305–315, mar 2015.

[14] K. Yue, D. Wang, H. Hu, and S. Fang, "The correlation between visual fatigue and duration of viewing as assessed by brain monitoring," *Journal of the Society for Information Display*, vol. 26, pp. 427–437, jul 2018.

[15] R. Xu, C. Zhang, F. He, X. Zhao, H. Qi, P. Zhou, L. Zhang, and D. Ming, "How Physical Activities Affect Mental Fatigue Based on EEG Energy, Connectivity, and Complexity," *Frontiers in Neurology*, vol. 9, oct 2018.

[16] J. L. Taylor, M. Amann, J. Duchateau, R. Meeusen, and C. L. Rice, "Neural contributions to muscle fatigue: From the brain to the muscle and back again," *Medicine and Science in Sports and Exercise*, vol. 48, no. 11, 2016.

[17] M. R. Smith, R. Chai, H. T. Nguyen, S. M. Marcora, and A. J. Coutts, "Comparing the Effects of Three Cognitive Tasks

on Indicators of Mental Fatigue," *Journal of Psychology: Interdisciplinary and Applied*, vol. 153, pp. 759–783, nov 2019.

[18] E. Wascher, B. Rasch, J. Sänger, S. Hoffmann, D. Schneider, G. Rinkenauer, H. Heuer, and I. Gutberlet, "Frontal theta activity reflects distinct aspects of mental fatigue," *Biological Psychology*, vol. 96, no. 1, pp. 57–65, 2014.

[19] J. Allen, "Photoplethysmography and its application in clinical physiological measurement," mar 2007.

[20] A. Bestbier, P. Fourie, and W. Perold, "Development of a vital signs monitoring wireless ear probe," *University of Stellenbosch*, 2017.

[21] F. Gharagozlou, G. Nasl Saraji, A. Mazloumi, A. Nahvi, A. Motie Nasrabadi, A. Rahimi Foroushani, A. Arab Kheradmand, M. Ashouri, and M. Samavati, "Detecting Driver Mental Fatigue Based on EEG Alpha Power Changes during Simulated Driving.," *Iranian journal of public health*, vol. 44, no. 12, pp. 1693–700, 2015.

[22] J. M. Morales, J. F. Ruiz-Rabelo, C. Diaz-Piedra, and L. L. Di Stasi, "Detecting Mental Workload in Surgical Teams Using a Wearable Single-Channel Electroencephalographic Device," *Journal of Surgical Education*, pp. 1107–1115, 2019.

[23] W. Guo, J. Ren, B. Wang, and Q. Zhu, "Effects of Relaxing Music on Mental Fatigue Induced by a Continuous Performance Task: Behavioral and ERPs Evidence," 2015.

[24] J. Liu, C. Zhang, and C. Zheng, "EEG-based estimation of mental fatigue by using KPCA-HMM and complexity parameters," *Biomedical Signal Processing and Control*, vol. 5, no. 2, pp. 124–130, 2010.

[25] R. Azevedo, M. D. Silva-Cavalcante, B. Gualano, A. E. Lima-Silva, and R. Bertuzzi, "Effects of caffeine ingestion on endurance performance in mentally fatigued individuals," *European Journal of Applied Physiology*, vol. 116, no. 11-12, 2016.

[26] P. Alhola and P. Polo-Kantola, "Sleep deprivation: Impact on cognitive performance.," *Neuropsychiatric disease and treatment*, vol. 3, no. 5, pp. 553–67, 2007.

[27] Y. D. Alqurashi, T. Nakamura, V. Goverdovsky, J. Moss, M. I. Polkey, D. P. Mandic, and M. J. Morrell, "A novel in-ear sensor to determine sleep latency during the multiple sleep latency test in healthy adults with and without sleep restriction," *Nature and Science of Sleep*, vol. 10, pp. 385–396, 2018.

[28] J. W. Ahn, Y. Ku, and H. C. Kim, "A novel wearable EEG and ECG recording system for stress assessment," *Sensors (Switzerland)*, vol. 19, may 2019.

[29] M. Bigliassi, B. M. Galano, A. E. Lima-Silva, and R. Bertuzzi, "Effects of mindfulness on psychological and psychophysiological responses during self-paced walking," *Psychophysiology*, jan 2020.

[30] T. Nakamura, V. Goverdovsky, M. J. Morrell, and D. P. Mandic, "Automatic Sleep Monitoring Using Ear-EEG," *IEEE Journal of Translational Engineering in Health and Medicine*, 2017.

[31] D. Acharya, A. Rani, and S. Agarwal, "EEG data acquisition circuit system Based on ADS1299EEG FE," Institute of Electrical and Electronics Engineers Inc., dec 2015.

[32] E. Mastinu, M. Ortiz-Catalan, and B. Hakansson, "Analog Front-Ends comparison in the way of a portable, low-power and low-cost EMG controller based on pattern recognition EMBC 2015," vol. 2015-November, pp. 2111–2114, Institute of Electrical and Electronics Engineers Inc., nov 2015.

[33] R. Kaveh, J. Doong, A. Zhou, C. Schwendeman, K. Gopalan, F. Burghardt, A. C. Arias, M. Maharbiz, and R. Muller, "Wireless User-Generic Ear EEG," mar 2020.

[34] E. Kuatsjah, X. Zhang, M. Khoshnam, and C. Menon, "Two-channel in-ear EEG system for detection of visuomotor tracking state: A preliminary study," *Medical Engineering and Physics*, vol. 68, pp. 25–34, jun 2019.

[35] N. Merrill, M. T. Curran, S. Gandhi, and J. Chuang, "One-step, three-factor passthought authentication with custom-fit, in-ear EEG," *Frontiers in Neuroscience*, vol. 13, no. APR, 2019.

[36] S. K. Longmore, G. Y. Lui, G. Naik, P. P.

Breen, B. Jalaludin, and G. D. Gargiulo, "A comparison of reflective photoplethysmography for detection of heart rate, blood oxygen saturation, and respiration rate at various anatomical locations," *Sensors (Switzerland)*, vol. 19, apr 2019.

[37] T. Hwang, M. Kim, S. Hong, and K. S. Park, "Driver drowsiness detection using the in-ear EEG," 2016.

[38] C. Wessels, P. Fourie, and W. Perold, "Design of an in-ear EEG device to detect consciousness levels and be used in monitoring anaesthesia levels of a patient in a medical setting," *University of Stellenbosch*, 2018.

[39] D. Looney, P. Kidmose, C. Park, M. Ungstrup, M. Rank, K. Rosenkranz, and D. Mandic, "The in-the-ear recording concept: User-centered and wearable brain monitoring," *IEEE Pulse*, vol. 3, no. 6, pp. 32–42, 2012.

[40] K. B. Mikkelsen, S. L. Kappel, D. P. Mandic, and P. Kidmose, "EEG recorded from the ear: Characterizing the Ear-EEG Method," *Frontiers in Neuroscience*, vol. 9, no. NOV, 2015.

[41] M. Tanaka, "Effects of Mental Fatigue on Brain Activity and Cognitive Performance: A Magnetoencephalography Study," *Anatomy & Physiology*, vol. s4, pp. 0–4, 2015.

Towards a Model for Zero Trust Data

Jason M. Pittman
Booz Allen Hamilton, USA
pittman_jason@bah.com

Shaho Alaee
Booz Allen Hamilton, USA
alaee_shaho @bah.com

Courtney Crosby
Booz Allen Hamilton, USA
crosby_courtney @bah.com

Tom Honey
Booz Allen Hamilton, USA
honey_tom @bah.com

Geoffrey M. Schaefer
Booz Allen Hamilton, USA
schaeffer_geoffrey@bah.com

***Abstract*— The world has realized traditional cybersecurity models are flawed because users and systems behind the perimeter are implicitly trusted. The response has been to treat access requests and behaviors post-access as untrusted. Thus, the aim of such zero trust architecture is to establish a borderless access-control framework. Accordingly, existing research is centered around network perimeters and communications layers. That is, data access channels or endpoints and not data itself. Consequently, we conducted a systematic review of relevant literature and developed a model illustrating a potential application of zero trust tenets and principles to data objects instead of data access pathways based on the findings. Concurrently, given the rising popularity of employing artificial intelligence to zero trust frameworks, our zero trust data concept targets artificial intelligence training and real-world evaluation data segments.**

Keywords—zero trust, cybersecurity, data, artificial intelligence

I. Introduction

The modern enterprise is bereft of certainty in terms of operational security. Friends are foes and adversaries appear as friends. Further, the differentiation between use and misuse is touted as quantitative but remains qualitative at best. As a result, a shift in the cybersecurity paradigm has begun which seeks to position trust itself as a vulnerability. More specifically, the traditional cyber- security model of implicitly trusting users and systems once they are within the enterprise perimeter is eschewed in favor of not trusting at all [5].

This movement - zero trust architecture- was born from the observation [22] that traditional cybersecurity continually fails because of fundamental misappropriation of trust. Remarkably, 72% of organizations planned to implement zero trust capabilities in 2020 [6]. A year later, the number rose to 76% [18]. The remarkability of these percentages becomes clear when evaluated in the context of zero trust architecture becoming a defined cybersecurity concept barely a decade earlier [22]. We take this as evidence that zero trust architecture has momentum as a cybersecurity paradigm as well as a meaningful rate of adoption to substantiate the hype.

However, a general problem is zero trust architecture applies to data access points (i.e., endpoints) but data objects are not discussed throughout the literature [6]. There is a prima facie difference between access to data and data being accessed. As much as the difference is recognized in the zero trust architecture literature [5, 6, 37], we see no effort to work at the level of data. Instead, the operational focus continues to be on the network perimeter [12, 19, 34, 39] and adjacent access endpoints such as Internet of Things or Cloud [30, 8].

Coupled with this general problem, the same research suggests, "[l]ooking out further, generative adversarial networks will continuously verify the efficacy of zero trust protection by generating synthetic attacks and threats" [6, p. 113]. Additional literature [8, 19, 11, 12, 38] has likewise pointed towards artificial intelligence as having a critical role in the future of zero trust architecture. The same literature has yet to consider the role of data objects as a necessary input to the very artificial intelligence systems purported to play such a role in trust-based cybersecurity constructs. Such a gap in the research is indicative of a specific problem. Thus, the purpose of this work is to demonstrate a model for zero trust at the level of data objects within artificial intelligence training and evaluation operational segments.

II. Related Work

The essence and substance of zero trust data rests upon four existing knowledge domains. Foremost, we operationalize *trust* in two contexts: as trust relates to knowledge as a general case and as trust relates to technology as a specific case. Then, we introduce the concept of *approximate epistemology*. Zero trust architecture relies upon conditional reasoning and subjective logic. Approximate epistemology is a necessary bridge then between *trust* and the technological instantiation of zero trust. The third knowledge domain encapsulates zero trust architecture. We found two relevant categories in the zero trust literature: one category contains the fundamental elements of zero trust and the other details the growing role of AI in zero trust. Lastly, a consequence of involving AI in zero trust necessitates discussing adversarial AI.

A. Trust

Knowledge relies on trust. In fact, we can articulate four discrete components of epistemic trust: belief, communication, reliance, and confidence [25]. From there, we can subgroup the components into epistemic variables (belief and communication) and trust variables (reliance and confidence) [25]. For the purposes of this work, we are most interested in trust variables but recognize epistemic variables cannot be fully decoupled from our concepts. To that end, trust in this context, exemplifies the social aspect of knowledge insofar as we do not directly experience trust but hold trust as valid because of the collective position of validity.

In this way, trust is a non-Boolean proxy for human behavior [36]. Furthermore, technological mediation is the embodiment of that proxy. Meaning, humans trust humans but pass trust judgement vis-vis the technologies created and used by people. Furthermore, perceived trust to be integral to society [35]. That is, trust as a knowledge construct, exists in many disciplines and permeates our cognitive existence [27]. Additionally, there is an argument to be made that, by using

technology, we implicitly place trust in such technology [23]. Nonetheless, trust we do. Certainly, part of such trust is due to the mediation technology provides. As well, trust in technology and trust from technology are integral functions. At the same time, we must be cautious in establishing concepts leading to technological trust, especially when trust is first positioned as a vulnerability. Such caution is warranted insofar as research [16, 17] has suggested that technological trust first-and-foremost stems from our relation to the technology.

B. Approximate Trust

Furthermore, research [20, 21, 2] suggests approximate trust is an extension of technological trust because modern technology operates within environments harboring high degrees of uncertainty. In the face of such un- certainty, trust is an emergent judgement of action based on assumed truth [29, 9]. If we treat data with zero trust, then data have zero truth. Therefore, it follows data must be treated as always false. If we accept such a conclusion as following from the premises, when combined with the notion of trust existing as a continuous or non-Boolean value, we can pose meaningful questions relative to the overarching topic. For instance, an operational question is how do we derive an approximate truthful (i.e., trusted) conclusion from known false premises?

Research in adjacent areas [33, 2, 26] demonstrates implementations of approximate or *fuzzy* logic. Notably, the implementations represented in the literature are con- fined to authentication and networking-based evaluations of trust. This extends naturally into zero trust architecture wherein traditional binomial conditional reasoning is employed. For instance, just as a common perimeter technology such as a firewall relies on binomial conditional reasoning [5, 20], so too does zero trust architecture. The difference being the former assumes trust whereas the latter assumes untrust.

Moreover, insofar as these architectures attempt to *diagnose* trust, probabilistic reasoning is not included as a rational foundation. This compounds potential issues since especially where human operators are involved since, "there is the possibility in the inability to take into account the analyst's levels of confidence in the probability arguments and the inability to handle the situation when the analyst fails to produce probabilities for some of the input arguments" [21, p. 462].

C. Zero Trust Architecture

Abstractly, NIST [31], codifies zero trust architecture implementation as (a) enhanced governance; (b) micro-segmentation; (c) and software-defined network perimeters. Succinctly, the emphasis of zero trust architecture is on borderless access-controls [1]. As an architecture, the goal of is, "fine-grained identify-based access control" [1, p. 9] to prevent lateral movement across the enterprise. Functionally, zero trust architecture functions in alignment with this goal by forcing the explicit verification, authentication, authorization, and continuous monitoring of access to data [6, 31, 1, 5]. Notably, existing re- search does not demonstrate a means of applying zero trust architecture *to data*.

Nonetheless, the literature does maintain a consistent articulation of the technological components essential for any zero trust architecture. Overarchingly, zero trust implementation requires a centralized controller which verifies access requests [40, 5]. Subordinately, the same re-search details how a policy enforcement point acts as a proxy service for these requests and communicates internally with a zero trust engine component. In turn, the engine cross-references access policies in its allocated pol- icy storage and communicates a *trust, no trust* semantic back upstream to the policy enforcement point [40, 5]. In this way, the default assumption of untrusted is equivalent to a default deny all in a firewall.

In addition to the implementation goal and overall architecture, there is consensus across the literature concerning the tenets and principles zero trust architecture embodies. For instance, common tenets [1] are (a) segmentation of access; (b) authentication for all access; (c) end-to-end encryption; (d) least privilege always; and (e) continuous monitoring of all endpoints. Stated differently but with the same intent [5], it can be said that zero trust (a) must apply to all data and services; (b) all access must be secured; (c) trust is never a default state; (d) it must be characteristics, behaviors, and environmental attributes which earn trust and not identity credentials; (e) and access is always temporary.

Despite the rising popularity and rate of adoption, zero trust architecture is not foolproof. Adversaries can by- pass zero trust architecture controls [1] if they are able to sufficiently alter the underlying policy or present them-selves as conforming to the policy as a form of trojan horse appearance. The potential issues are compounded by the necessity to construct zero trust models as self- regulating and immutable. To this end, the literature is pointing towards incorporation of AI into the PEP and engine layers.

Indeed, an AI agent is an appropriate technology to handle the complexity of mediating untrusted access [38]. The addition of AI is not without its own perils though. Chiefly, AI is tightly coupled to data and the standard AIOPs workflow includes two segments vulnerable to manipulation [7, 14]. The level of irony associated with employing AI to zero trust while not securing trust within the AI itself cannot be overstated.

D. Adversarial AI

The relation of adversarial AI to zero trust architecture may seem tenuous at first. However, an outstanding challenge in adversarial AI is to mitigate threats originating from areas of uncertainty [3, 4]. Meanwhile, zero trust architecture research [6, 8, 19, 11, 12, 38] is calling for deeper incorporation of AI. Critically, data are not capable of proving trust as might be the case for a user accessing a network segment. Thus, we understand some portion of the uncertainty to be related to data used during AI model training and, separately but coupled to the same idea, data ingested by the AI during operative evaluation.

In brief, adversarial training encompasses a set of, "intentionally worst-case perturbations to examples from the dataset" [10, p. 1]. A variety of examples exist in the research, most notably methods to perturb or *poison* AI with the intent of corrupting classification modalities [10, 24]. While such perturbative techniques can be effective, the research community is actively developing countermeasures [3]. Unfortunately, the countermeasures appear effective only within spaces governed by certainty.

Based on this, we argue for reconceptualizing adversarial training attacks within the framework of zero trust. In other

words, training data are not *poisoned* as much as such data are *untrusted*. In doing so, the evolving re- search in the adversarial AI space remains adjacent while also allowing for zero trust data to co-evolve as a distinct area of investigation.

III. Method

The goal of this research was to demonstrate a model for zero trust data. To affect this goal, we followed a systematic literature review methodology [15, 28]. Specifically, we (1) searched for literature; (2) selected results from the search based on inclusion criteria; (3) extracted relevant features; (4) and synthesized those features into findings.

Further, to guide and organize the work, we developed three research questions:

Q1: What zero trust architecture tenets or principles are applicable to zero trust data?

Q2: What tenets or principles do not exist in zero trust architecture that are necessary for zero trust data?

Q3: How do the responses to questions 1-3 converge into a zero trust data model?

A. Literature Search and Selection

We operationalized a series of literature searches as input to the systematic review methodology. The searches were conducted against prominent research databases such as ACM, IEEE, EBSCOHost, Springer, dblp, Microsoft Re- search, and Arxiv. We also leveraged Google Scholar to search these and other academic indices. Specific search strings included, but were not limited to, *zero trust*, *zero trust architecture*, *trust and technology*, *zero trust and (tenets or principles)*, *adversarial training approximate trust and technology*.

Overall, the search uncovered more than 874,000 articles. As with all literature searches, a smaller subset was created through a manual review of title and abstract. Then, the operational corpus was selected based on a reading of specific article sections such as introduction and results or findings.

B. Literature Inclusion Criteria

As with all literature searches, not all results are relevant or of sufficient quality to be valuable to the systematic review. Therefore, a definitive protocol to guide include (or exclude for that matter) returned literature becomes operational vital. In view of this, we included zero trust architecture and adversarial AI relevant research from the past ten years. The fields are significantly new and thus date criteria did not negatively impact the literature searches. On the other hand, trust and approximate truth relevant research spans a wide timeline. Therefore, it did not make sense to apply date criteria.

Furthermore, we did not actively include or exclude based on stated research methodology. In all cases we included previous reviews as well because of the leverage provided by such literature. Against the background of existing literature review, we focused on theoretical re- search given the exploratory nature of this work. While we had a primary interest in research demonstrating zero trust architecture models or prototypes of models, we included any relevant research attempting to situate such theory in application (e.g., Internet of Things, Cloud, and so forth).

Likewise, the discriminatory criteria applied to the ad- jacent literature categories foremost operationalized the theoretical utility of the topic. For instance, there is theoretical utility in developing a basis for trust insofar as trust is utilized in zero trust without a need to consider the applicable situation of the former.

IV. Findings

The following sections present our findings based on the outcomes of our systematic review. The findings are grouped and presented according to our guiding research questions for organizational purposes, not to imply state or prioritization in any way.

A. Question 1 - Tenets and Principles

Tenets and principles form the core of any exploratory effort with a goal of developing a model. To that end, we found nine articles [38, 32, 31, 22, 8, 6, 5, 1, 13] which explicitly asserted a set of tenets or principles for zero trust architecture. The set of tenets or principles ranged from three to seven assertions with three being common. Overall, we found the following applicable to our work:

1) trust is not a default state

Whereas the literature applies trust to users or systems accessing services and endpoints, the same applies to data objects.

2) access must be segmented

The literature conceptualizes access in terms of network communications. We refactor this principle to apply in terms of programmatic access between internal software components.

3) activity must be continuously monitored

Unlike with the previous two principles, activity is a general concept with unaltered applicability to our research goal. We do contextualize activity within the specific bounds of AI model training and evaluation (Fig. 1, Appendix A).

The remaining tenets and principles in the literature were related to authentication, least privilege, or to non-data objects. Consequently, those are not viable within the context of our research purpose. However, this is not to say there are no other elements potentially relevant.

B. Question 2 - Missing Elements

Indeed, given the identified problem stated in the intro- duction, we understood from the beginning of the study there might be missing elements from the set of tenets and principles. Accordingly, a concurrent phase of the systematic review was to infer absent features or characteristics in relation to the specific context for the potential zero trust data model.

1) data as an object

Without any doubt, the first missing element from existing zero trust architecture principles are data. Instead of relegating data to a motivation for implementing conventional zero trust architecture, this work positions data as the focal point of the architecture. Moreover, it may be important to distinguish between individual data elements

(e.g., a specific feature) and complete datasets (e.g., a data model).

2) internal components

The existing literature does not specifically outline architectural concepts or implementations relative to internal software components. This is rational when discussing zero trust at the network layer. However, particularly in the context of AI, internal components are critical elements to incorporate as such are simultaneously the end- point and the access mechanism in relation to data.

C. Question 3 - Converged Model

Perhaps we can demonstrate the converged model of tenets and principles with missing elements by leveraging the logic of approximate trust. In this manner, we use a contrapositive quantifier: assume values are false and compute the following:

$$\forall x(\neg Q \rightarrow \neg P)$$

Fortunately, the adversarial AI literature demonstrates the traditional paradigm of: If an object is red, is an octagon, and has the text "STOP", then the object is a stop sign.

We can take the contraposition of the traditional paradigm as: If the object is not a stop sign, then it is not red, not an octagon, and does not have the text STOP.

On one hand, this is a classification problem: trust versus untrusted or true versus false. On the other hand, this is a regression problem: spectrum or degree of truth (i.e., certainty). Thus, our assertion is the concept of how the zero trust data model operates can be described as a function with two inputs as U or the untrusted set of data and T as the trusted set of data. The delta between the intersection of the lower approximation of these sets is deltaed against the intersection of the lower approximation of the same:

$$f(U, T) \rightarrow (\underline{U} \cap \underline{T}) \Delta (\underline{U} \cap \underline{T})$$

The intended outcome is output approximately more trusted or approximately less poisoned. A simple example embodying this expression is the case of a poisoned stop sign artifact. If the zero trust data system assumes all road signs are untrusted (i.e., poisoned), it can use the approximation of (1) an object appearing as a sign and (2) approximating a stop sign given a stop signs unique features contained in the trusted data, the system ought to approximate stop sign from the intersection road sign features and observed features.

Doing so elevates categorization beyond trust and to truth. Put simply, the zero trust data model asserts it is true enough an object appears like a stop sign to be a stop sign. This is in contrast to trusting an object is or is not a stop sign.

With the stop sign example in mind, we contend a zero trust data solution must include two components, one for each segment of the overall AIOPs pipeline (Fig. 2, Appendix A). On the training segment, we see the potential for employing a Bayesian Network with an embedded subjective logic gate. The Bayesian Network would quantify approximate truth through two mechanisms (see Appendix B, Fig. 3). On the evaluation segment, the zero trust data model takes a queue from existing research [4] in leveraging a GAN [4].

V. Conclusion

Zero trust is a state, not a discrete technology, pol- icy zone, or protocol layer. Despite the brief time since its inceptions [22], zero trust has evolved a stable set of core tenets and principles governing its architecture. While specific assertions vary across the literature [38, 32, 31, 22, 8, 6, 5, 1, 13], there is consistency in implementations across a variety of technological plat- forms [12, 30, 36]. However, research [1] suggests zero trust architecture may be imparting a false sense of security because the dominant architecture focuses on end- points.

The purpose of this work was to rebalance the traditional paradigm wherein systems and data within a de- fined boundary are implicitly trusted. The rebalancing is achieved by inverting trusted to untrusted at the level of data rather than at the level of access to such data. In this sense, there is applicability of zero trust data in protecting against adversarial AI manipulations at the data layer.

While conducting the systematic review, we realized zero trust data may be useful outside the scope of cybersecurity. For instance, in general, machine learning depends heavily on unbiased, normalized data. It may be possible to apply a zero trust data solution to guard against such data-based issues. The result would be machine learning models with higher accuracy and lower false positives during evaluation periods. Furthermore, additional future work should include development of an applied prototype leveraging the zero trust data function demonstrated in our findings. In doing so, the inevitable convergence between the related work may be realized. Finally, an important factor not considered thus far is the human-computer interaction consequences of implementing zero trust both in the traditional context as well as the zero trust data framework.

References

[1] Alevizos, L., Ta, V. T., and Hashem Eiza, M. Augmenting zero trust architecture to endpoints using blockchain: A state-of-the-art review. Security and Privacy 5, 1 (2022).

[2] Arora, G., Mathur, I., and Gandhi, S. Quantifying trust evaluation based on approximate rea- soning. In 2015 2nd International Conference on Computing for Sustainable Global Development (IN-DIACom) (2015), IEEE, pp. 1448–1451.

[3] Bai, T., Luo, J., Zhao, J., Wen, B., and Wang, Q. Recent advances in adversarial train- ing for adversarial robustness. arXiv preprint arXiv:2102.01356 (2021).

[4] Bai, T., Zhao, J., Zhu, J., Han, S., Chen, J., Li, B., and Kot, A. Towards efficiently evaluat- ing the robustness of deep neural networks in iot systems: A gan-based method. IEEE Internet of Things Journal (2021).

[5] Buck, C., Olenberger, C., Schweizer, A., Volter, F., and Eymann, T. Never trust, always verify: A multivocal literature review on current knowledge and research gaps of zero-trust. Com- puters & Security 110 (2021), 102436.

[6] Campbell, M. Beyond zero trust: trust is a vulnerability. Computer 53, 10 (2020), 110–113.

[7] Dang, Y., Lin, Q., and Huang, P. Aiops: real- world challenges and research innovations. In 2019 IEEE/ACM 41st International Conference on Soft- ware Engineering: Companion Proceedings (ICSE- Companion) (2019), IEEE, pp. 4–5.

[8] Dimitrakos, T., Dilshener, T., Kravtsov, A., La Marra, A., Martinelli, F., Rizos, A., Rosett, A., and Saracino, A. Trust aware continuous authorization for zero trust in consumer in- ternet of things. In 2020 IEEE 19th International Conference on Trust, Security and Privacy in

Computing and Communications (TrustCom) (2020), IEEE, pp. 1801–1812.

[9] Douven, I., and Kelp, C. Truth approximation, social epistemology, and opinion dynamics. Erkenntnis 75, 2 (2011), 271–283.

[10] Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014).

[11] Hale, B., Van Bossuyt, D. L., Papakonstantinou, N., and O'Halloran, B. A zero-trust methodology for security of complex systems with machine learning components. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (2021), vol. 85376, American Society of Mechanical Engineers, p. V002T02A067.

[12] Hireche, O., Benza¨ıd, C., and Taleb, T. Deep data plane programming and ai for zero-trust self- driven networking in beyond 5g. Computer Net- works (2021), 108668.

[13] Horne, D., and Nair, S. Introducing zero trust by design: Principles and practice beyond the zero trust hype.

[14] Hornik, K., Stinchcombe, M., and White, H. Multilayer feed forward networks are universal approximators. Neural networks 2, 5 (1989), 359–366.

[15] Hunt, H., Pollock, A., Campbell, P., Est- court, L., and Brunton, G. An introduction to overviews of reviews: planning a relevant research question and objective for an overview. Systematic reviews 7, 1 (2018), 1–9.

[16] Ihde, D. Technology and the lifeworld: From garden to earth.

[17] Ihde, D. Technics and praxis: A philosophy of technology, vol. 24. Springer Science & Business Media, 2012.

[18] Invanti. 2021 zero trustprogress report. Tech. rep., 2021.

[19] Jin, Q., and Wang, L. Zero-trust based distributed collaborative dynamic access control scheme with deep multi-agent reinforcement learn- ing. EAI Endorsed Transactions on Security and Safety 8, 27 (2020).

[20] Josang, A. Conditional reasoning with subjective logic. Journal of Multiple-Valued Logic and Soft Computing 15, 1 (2008), 5–38.

[21] Jøsang, A. Generalising bayes' theorem in subjective logic. In MFI (2016), pp. 462–469.

[22] Kindervag, J., Balaouras, S., et al. No more chewy centers: Introducing the zero trust model of information security. Forrester Research 3 (2010).

[23] Kiran, A. H., and Verbeek, P.-P. Trusting our selves to technology. Knowledge, Technology & Pol- icy 23, 3 (2010), 409–427.

[24] Kireev, K., Andriushchenko, M., and Flammarion, N. On the effectiveness of adversar- ial training against common corruptions. arXiv preprint arXiv:2103.02325 (2021).

[25] McCraw, B. W. The nature of epistemic trust. Social epistemology 29, 4 (2015), 413–430.

[26] Miao, T., Shen, J., Lai, C.-F., Ji, S., and Wang, H. Fuzzy-based trustworthiness evaluation scheme for privilege management in vehicular ad hoc networks. IEEE Transactions on Fuzzy Systems 29, 1 (2020), 137–147.

[27] Origgi, G. Is trust an epistemological notion? Episteme 1, 1 (2004), 61–72.

[28] Pollock, A., Campbell, P., Brunton, G., Hunt, H., and Estcourt, L. Selecting and implementing overview methods: implications from five exemplar overviews. Systematic reviews 6, 1 (2017), 1–18.

[29] Ramsey, J. L. Towards an expanded epistemology for approximations. In PSA: Proceedings of the bien nial meeting of the philosophy of science association (1992), vol. 1992, Philosophy of Science Association, pp. 154–164.

[30] Rodigari, S., O'Shea, D., McCarthy, P., Mc- Carry, M., and McSweeney, S. Performance analysis of zero-trust multi-cloud. In 2021 IEEE 14th International Conference on Cloud Computing (CLOUD) (2021), IEEE, pp. 730–732.

[31] Rose, S., Borchert, O., Mitchell, S., and Connelly, S. Zero trust architecture. Tech. rep., National Institute of Standards and Technology, 2020.

[32] Sanders, G., Morrow, T., Richmond, N., and Woody, C. Integrating zero trust and devsecops. Tech. rep., 2021.

[33] Schmidt, S., Steele, R., Dillon, T. S., and Chang, E. Fuzzy trust evaluation and credibility development in multi-agent systems. Applied Soft Computing 7, 2 (2007), 492–505.

[34] Sengupta, B., and Lakshminarayanan, A. Distritrust: Distributed and low-latency access validation in zero-trust architecture. Journal of Informa- tion Security and Applications 63 (2021), 103023.

[35] Simmel, G. The philosophy of money. Routledge, 2004.

[36] Sood, A. K., Huang, Y., Simon, R., White, E., and Cleary, K. Zero trust intrusion containment for telemedicine. Tech. rep., 2002.

[37] Teerakanok, S., Uehara, T., and Inomata, A. Migrating to zero trust architecture: reviews and challenges. Security and Communication Networks 2021 (2021).

[38] Walker-Roberts, S., and Hammoudeh, M. Artificial intelligence agents as mediators of trustless security systems and distributed computing appli- cations. In Guide to Vulnerability Analysis for Computer Networks and Systems. Springer, 2018, pp. 131–155.

[39] Wang, L., Ma, H., Li, Z., Pei, J., Hu, T., and Zhang, J. A data plane security model based on zero-trust architecture.

[40] Yan, X., and Wang, H. Survey on zero-trust network security. In International Conference on Artificial Intelligence and Security (2020), Springer, pp. 50–60.

Appendix A

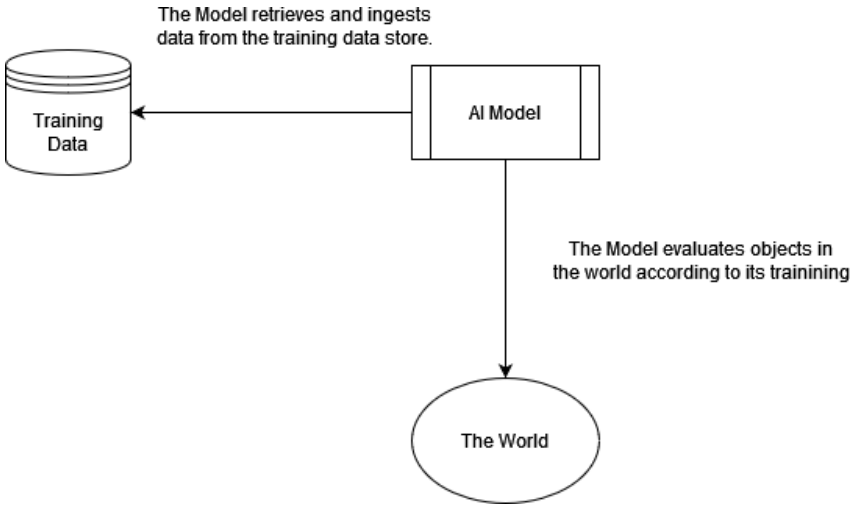


Fig. 1. The Standard AIOPs Pipelines

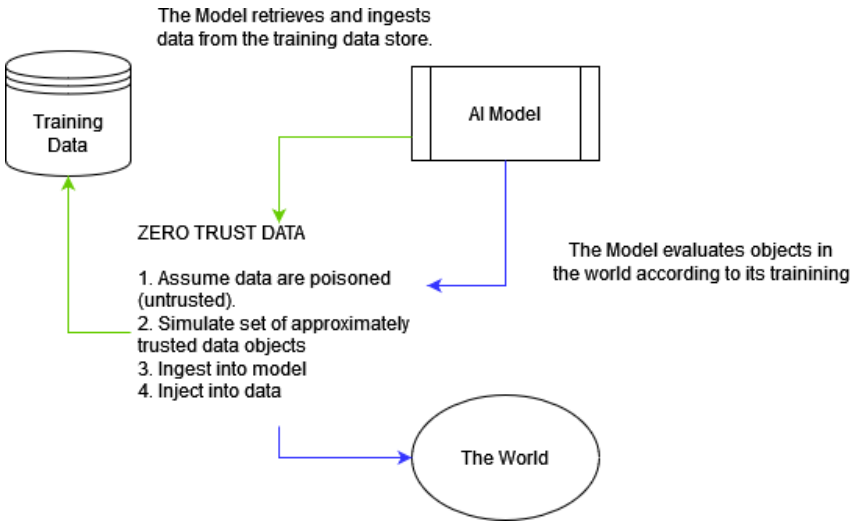


Fig. 2. The Standard AIOPs Pipelines with ZTD

Appendix B

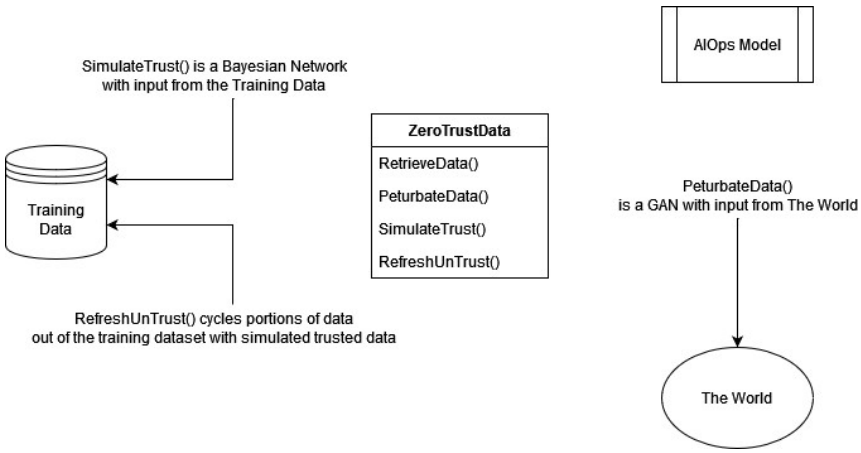


Fig. 3. The Methods for Zero Trust Data

Implementing Classical Logic in a Quantum Environment

Christopher T. Dunne
Capitol Technology University, USA
cdunne@captechu.edu

***Abstract*— This paper is meant to serve as an introductory guide on how to implement simple logic gates in a quantum environment. Uncompressed circuits for each statement can be found in the appendix.**

Keywords—quantum, logic, classical

I. IF STATEMENTS

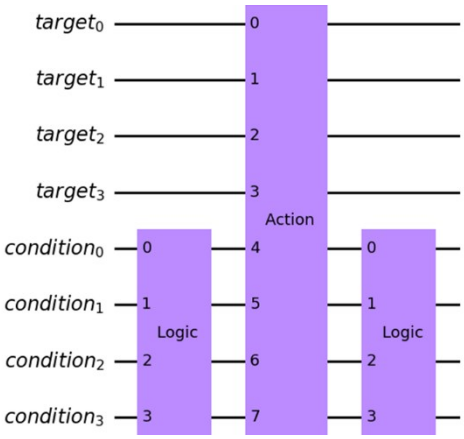


Fig. 1. The general structure of an IF statement in a quantum environment.

The target register in Fig. 1 will store the results of the Action gate and the condition register will be used in the Logic gate to determine if the Action gate runs. The Logic gates are used to set each qubit in the condition register to |1⟩, or true.

A. LOGIC GATE EXAMPLE

If the Action gate is only supposed to run if a given value is 1001 the Logic gate would look like Fig. 2. This works by running an X-gate on each qubit that should be in the state |1⟩ if the desired state was |1001⟩. In this example, the qubits being set to |1⟩ are qubits 1 and 2 of the condition register. The second Logic gate will then set the condition register back to its original value of |1001⟩.

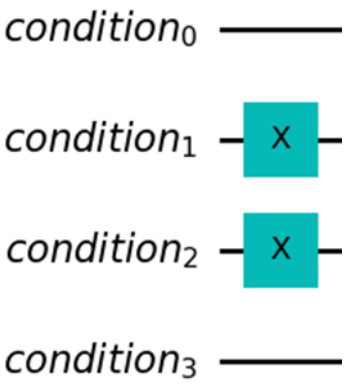


Fig. 2. An example of a Logic gate testing to see if the condition register is in the state |1001⟩.

B. ACTION GATE EXAMPLE

The Action gate will consist of the normal gates one would use to reach a desired outcome but said gates will be modified to be controlled gates. Controlled gates are gates with a base gate and a control state, wherein the base gate will only run if the control is true (ControlledGate — Qiskit 0.36.1 documentation, 2021). Fig. 3 displays an example of this if one wanted to invert qubits 1 and 2 of the target register if the condition register was true.

In this scenario the Action gate consists of two controlled gates wherein a X-gate serves as the base gate and the condition register serves as the control state. This controlled gate is functionally identical to the MCX (multi-controlled X) gate (MCXGate — Qiskit 0.36.1 documentation, 2021). If a multi-controlled gate has multiple targets it will be referred to as a MCMT (multi-controlled multi-target) gate.

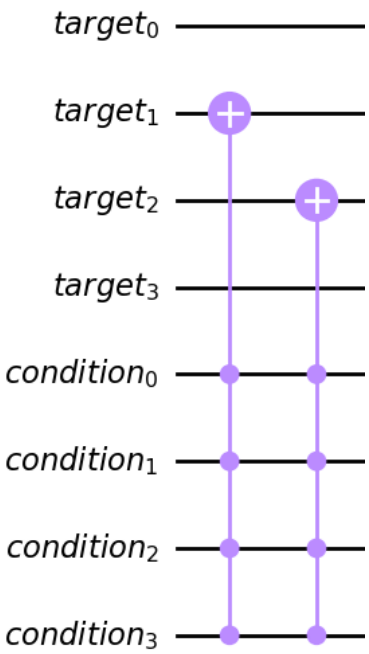


Fig. 3. An example of an Action gate that will inverts qubits 1 and 2 of the target register.

II. IF AND STATEMENTS

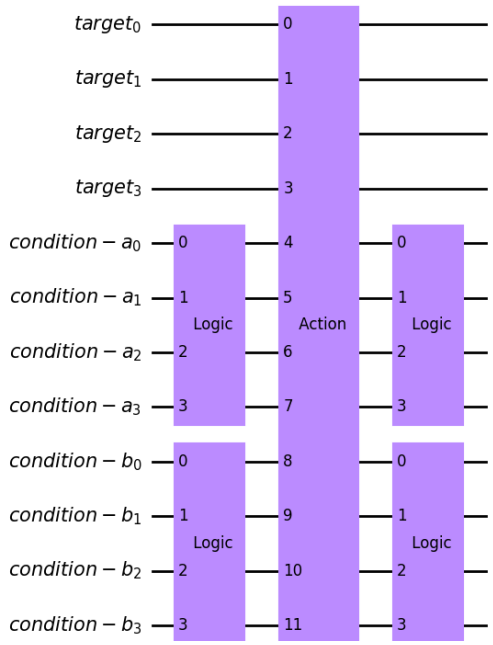


Fig. 4. The general structure of an IF AND statement in a quantum environment.

IF AND statements in a quantum environment are identical to IF statements, but the multi-controlled gates in the Action gate will use the qubits in both condition registers. This ensures that the Action gate will only run if the condition registers are in their desired state.

III. IF OR STATEMENTS

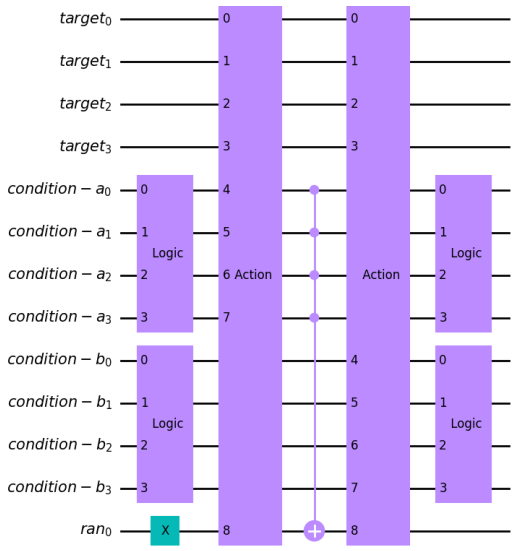


Fig. 5. The general structure of an IF OR statement in a quantum environment.

In an IF OR statement the Action gates will use the condition register as well as the ran register as the control state. The ran register is a single qubit register that is initialized to the state $|1\rangle$. After the first Action gate runs a MCX gate is used using the condition register for the first condition as its control state. This ensures that the Action gate will not be ran twice if both conditions are true.

IV. IF ELSE STATEMENTS

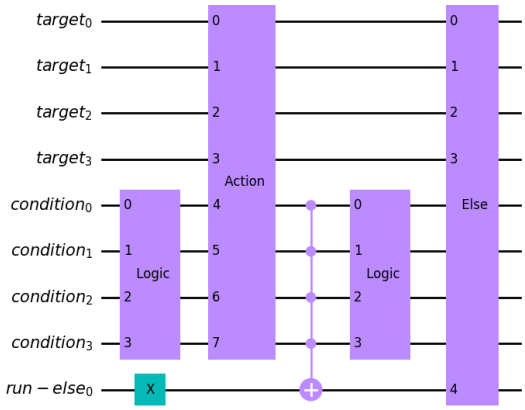


Fig. 6. The general structure of an IF ELSE statement in a quantum environment.

An IF ELSE statement uses an additional register (run-else) to determine whether to run the else condition. Similar to the ran register in the IF OR circuit, the run-else register is also initialized to the state $|1\rangle$. After the Action gate is ran a MCX gate is used using the condition register as the control state. Afterwards the Else gate will run if the run-else register is in the state $|1\rangle$.

V. APPENDIX

A. IF CIRCUIT

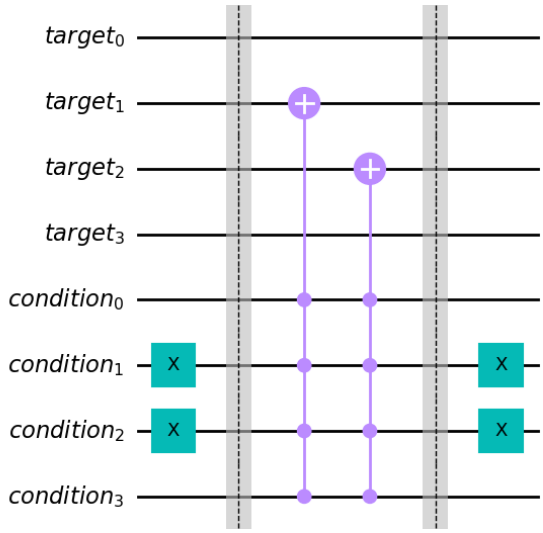


Fig. 7. A circuit that will invert qubits 1 and 2 of the target register if the condition register has the initial state |1001⟩.

B. IF AND CIRCUIT

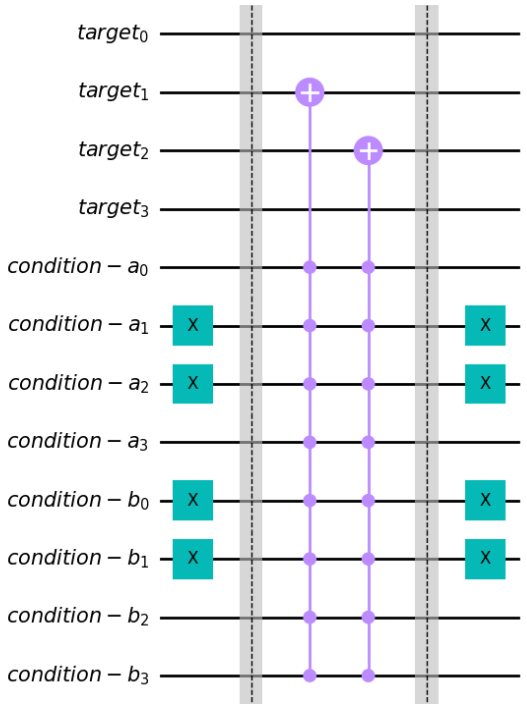


Fig. 8. A circuit that will invert qubits 1 and 2 of the target register if the condition registers have the initial states of |1001⟩ and |0011⟩.

C. IF OR CIRCUIT

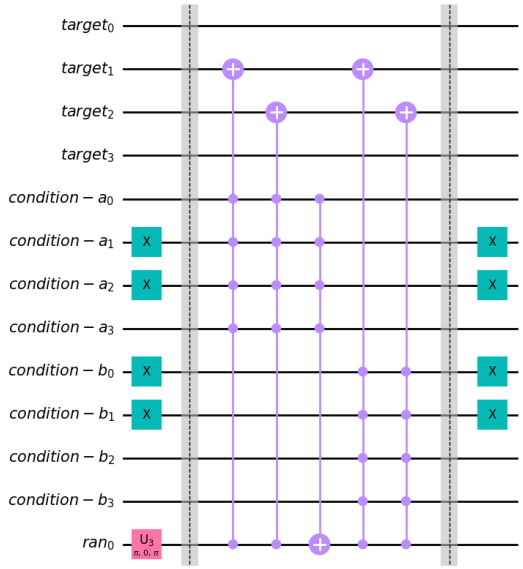


Fig. 9. A circuit that will invert qubits 1 and 2 of the target register if the condition registers have the initial states of |1001⟩ or |0011⟩.

D. IF ELSE CIRCUIT

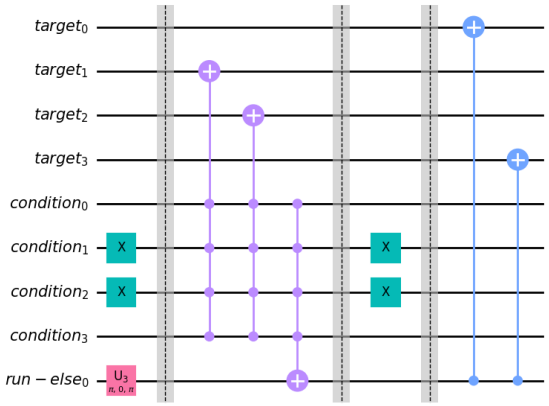


Fig. 10. A circuit that will invert qubits 1 and 2 of the target register if the condition register has the initial state |1001⟩. Otherwise, it will invert qubits 0 and 3 of the target register.

REFERENCES

ControlledGate — Qiskit 0.31.0 documentation, 2021, accessed October 8, 2021, at Qiskit.org at https://qiskit.org/documentation/stubs/qiskit.circuit.ControlledGate.html.

MCXGate — Qiskit 0.31.0 documentation, 2021, accessed October 8, 2021, at Qiskit.org at https://qiskit.org/documentation/stubs/qiskit.circuit.library.MCXGate.html.

Hardware Utilization by using Docker

Marshia Mostafiz Mim
American International University-Bangladesh (AIUB)
mostafizmim24@gmail.com

Joydeb Karmakar
American International University-Bangladesh (AIUB)
krmkjoy@gmail.com

Mrinmoy Karmakar
American International University-Bangladesh (AIUB)
mrinmoy602@gmail.com

Moshfiq-Us-Saleheen Chowdhury
Military Institute of Science and Technology (MIST)
moshfiqussaleheenchowdhury@gmail.com

Jannatun Nayim Supti
American International University-Bangladesh (AIUB)
jannatun.supti@gmail.com

***Abstract*—Developers and system administrators may use Docker to construct, ship, and operate distributed applications. Docker's main advantage is that it enables code to be quickly tested and deployed into production across a variety of applications. This article looks at the performance of Docker containers. They are judged on the effectiveness of their system. This is predicated on making the most of the system's resources. Docker Swarm is an open-source project that lets you build, deploy, and operate applications in a virtualized container environment. The goal of this research is to use the host computer's resources to spread web server traffic across a Docker swarm. Using this method, a single point of failure in a web server cluster is less probable**

Keywords—Benchmark, Container, Docker, Load balancing, Resource Usage, Swarm, Web Cluster

I. INTRODUCTION

Docker is a new concept in Computer science by using this we can utilize hardware that can be helpful to all of us and society. Learning about the docker system very deeply is one of my motivations. I want to improve the system and I want to make it easier and more efficient. Improvement of this system is my main concern. Research can help us to develop the existing system. I think in our upcoming development and deployment world it will be very promising. In our personal computer or servers, there are a lot of unused memory storage left and a small part of it is used. Docker's key benefit is that it allows code to be tested and pushed into production as quickly as feasible. Docker is a free and open platform that allows developers and system administrators use Docker Hub, a cloud service for sharing programs and automating processes, and Docker Engine, a portable, lightweight runtime and packaging tool, to create, ship, and execute distributed applications. The primary benefit of Docker is that code can be tested and deployed as soon as possible into production in various applications are possible. Be executed in a language-independent way across Docker containers. This article investigates the performance of Docker containers. They are evaluated based on the performance of their system. This is founded on the Utilization of system resources Various benchmarking tools are available. This is what I used. The performance of the file system is measured. Bonnie++ is being used. Other system resources, such as CPU usage, Benchmarking is used to evaluate memory utilization and other factors. Python code (using psutil) was created. Obtaining specific results. This document also includes the findings of all of these testing. CPU utilization, memory usage, CPU count, and CPU times, as well as disk partition, network I/O counter, and so on, are the results. (Preeth, 2015)

Virtualization using containers is becoming increasingly common. It's a kind of mild virtualization that takes use of the Linux kernel's ability to separate applications and control resource allocations. Docker is a virtualization system based on containers that is extensively used today. Docker is an open-source software that allows you to create, ship, and run applications in a containerized environment. Docker may be used to serve millions of users by deploying several web application containers. It can decrease a lot of chances of the architecture which will have one of the failures. Arranging a few containers to form one service, on the other hand, is a challenging undertaking. Docker Swarm, a container cluster management solution, solves this problem. The internal load balancing mechanism in Docker Swarm was designed to distribute requests amongst workers in a fair and equitable manner based on the user's request. It has no means of knowing how much of each host system's resources are being used. It's concerning since it might result in an imbalanced load distribution across host systems. On each host system, we focused on memory consumption. We propose a technique for monitoring and distributing web traffic depending on memory use on each host computer. Our experiment turned out to be a success. A balanced load distribution is applied to each worker node. In a web server cluster, this strategy may assist to less the chances of one point of failure. (Bella, 2018)

Web servers have become a critical component of the internet's architecture. The web application runs on top of the most popular internet tools, such as Google, YouTube, and Gmail. Building a reliable web server is essential for safeguarding the company from system risks such as outages, message processing errors, insufficient data, and data loss. (M.Waliullah, 2014). Container-based virtualization, rather than virtual machines, has lately acquired traction, especially in the deployment of microservice architecture. (O.Sallou, 2015) (M.Villari, 2016). Container-based virtualization is a kind of virtualization that divides and regulates process resource allocations using Linux kernel capabilities. It doesn't need or

include its own operating system; instead, it makes advantage of the host kernel's functionality and resource isolation (CPU, memory, block I/O, network, and so on) to completely isolate the application's operating system. (Preet, 2015) (S.Julian, 2018).

Docker is an open-source software that allows you to create, ship, and run applications in a containerized environment. A Docker container isolates the application from the host infrastructure. By packaging the program and its dependencies into a container, it may be possible to minimize the time between creating and deploying code. (Preet, 2015).

There are two sorts of nodes in the Docker Swarm: management nodes and worker nodes. A Docker host computer may act as both a worker and a management node. The administration node uses its own IP address and port to provide the swarm's services to the outside world. (Docker, 2018).

II. Literature Review

Docker is an open platform that provides container-based virtualization. Container-based virtualization is also called OS-level virtualization. It delivers software in packages called Container. Containers are isolated spaces, and they are separated from one another. It is a client-side application program. It is also can act as a service and can be deployed onto any server. Portability is one of the absolute charms of the docker (M.Waliullah, 2014) (M.Chae, 2017) (Docker, 2018). Docker is advanced level virtualization.

Docker allows developers to package the application into a container. Containers also contain host OS configuration, networking information, also code or application dependencies. The main purpose of Docker is like one application or program is running on a developer pc or server perfectly but as soon as it deploys on another pc or environment it might not act like the same (Mochamad, 2018) (M. Villari, 2016) (J. Liu, 2006). Docker can help to solve this problem.

With Docker, you'll oversee your framework within the same ways you oversee your applications. In the Docker container, every dependency related to the application will be packeted up. There is another option in virtualization which is Virtual Machines (VM). Virtual Machine works like a fully new OS machine. It (VM) consumed hardware resources because it is hardware-based virtualization (Preeth, 2015) (Mochamad, 2018). There are some bad sides of virtual machines like consuming more hardware and creating a new VM is time-consuming. This problem can be solved by using OS-based virtualization Docker. On tracking data and operating strategies specified by the end-user, application developer and/or administrator, per container, features of the resources needed by the host can be allocated. Preeth et al. evaluated Docker container efficiency with a focus on device resource optimization. The experimental scope has been broadened to include researching container scalability to reduce resource waste and increasing prediction accuracy on a web application using a load balancer. In terms of virtualization technology performance, the authors suggest that container-based virtualization may be compared to a memory-based OS operating on bare metal (2016). Docker has a lower resource usage than virtual machines. (Preeth, 2015) (Mochamad, 2018) (S. Julian, 2018).

Docker, on the other hand, has several flaws.

In the Docker system, there are three types of networking hosts. Bridge networking for single hosts, overly networking for multi-host communication, and Macvlan networking are all used in docker containers. Docker containers and services are very powerful because they can easily be integrated with other Docker containers and services, as well as non-Docker workloads. Containers and services in Docker don't need to know whether their neighbors are Docker workloads. Docker will manage your Docker hosts whether they are running Linux, Windows, or a mix of the two. Docker may be more powerful than a virtual machine in certain situations (VM). Hardware is disabled in the virtual machine to make room for a new virtual OS that functions similarly to a real computer. However, since Docker is an OS-based virtualization, it uses fewer resources. Docker is a modular system. The framework is built on ECo-Ware, which defines self-resource adaptation as a meta-workflow strategy for dealing with applications. It comes with the TOSCA library to make infrastructure easier, and it's been tested on Amazon EC2 to look at resource metrics like average response time. Despite the fact that, resource management is embedded into the container. Virtual machines have a major issue that will influence cloud computing. That is resource squandering. In the massive cloud system, this will be a major issue. However, certain testing show that Docker's CPU management isn't always reliable. (Mochamad, 2018) (M. Waliullah, 2014) (O. Sallou, 2015). This is one of the drawbacks of docker. But in Ram management docker is better than VMs. Docket CPU management problems will be a big problem in the long run. In some research, CPU management is not good for some situations (Mochamad, 2018) (M.Villari, 2016) (M. Fazio, 2016). This particular issue has room for improvement. The phrase is most often used to describe information centers that are available to many users over the internet. If a CPU issue occurs in cloud computing, the whole process will be disrupted. As a result, hardware usage is critical.

III. Methodology

A. Proposed Research Methodology

According to my research topic, I need to follow two methods and they are formal method & the model method. In my research, I went to utilize the hardware of the Docker system (Preeth, 2015). It will help the upcoming user experience. How the system improvement helps the new user that we can measure by using model method (G Pierre, 2020). In Docker, there is some ram issue that makes upgrade hassles, performance-critical applications, multiple operating systems, security is a critical factor critical with some ram and CPU issue (O. Sallou, 2015) (S. Julian, 2018) (G Pierre, 2020). We can design a bespoke model to comprehend and address this problem on a small scale in this kind of situation. When working on a huge project, the model might be useful. Modelling is the process of reducing an actual or projected model to a small, but representative, collection of components and interactions that allows its

attributes to be expressed qualitatively and quantitatively. In big data analysis and workload can be understood by model method (M. Fazio, 2016) (Preeth, 2015) (N. Thoai, 2016). With the expanding prominence of holder advances, many exploration endeavors have been made to investigate the points of interest of holders, improve compartment execution and security, just as to contrast the holders and the VMs (Mochamad, 2018), (N. Thoai, 2016). Carl (Docker, 2018) addressed the typical challenges avoided by the reproducibility of a vast number of research projects and the evaluation of how they can be overcome by the Docker container technique. (M. Data, 2018) Roberto provided a performance assessment of container utilization in the Internet of Things industry in terms of container performance and security. Roberto demonstrated that the containerized layer had no effect when compared to native execution utilizing a single board computing system like the Raspberry Pi 2. To further understand why there is such a huge disparity between Docker and VM bootup speed, we must study the real system's memory use using the model technique (Docker, 2018), (M. Data, 2018). Then, to improve the system Formal method is very important. The formal method basically works with algorithms analysis, space complexity analysis, time complexity analysis which is important for hardware utilization of Docker. To improve the system complexity analysis is important and very relevant to think. There was some algorithmic error that occurs this kind of hardware utilization problem (M. Waliullah, 2014), (O. Sallou, 2015), (M. Fazio, 2016). Container virtualization is gaining popularity as a technology. As a result of IBM's mainframe implementation of virtualization in the 1970s (i.e., the hypervisor), container-based technologies have been developed (for example, Docker and Linux Container). It was newly introduced. The key benefit in containers is that near-native efficiency is attained. It arises from a review of the literature on performance measurement of container runtime environments that study works can be divided into workload distributed database studies and other concurrent workload studies (A. M. Joy, 2015). The space and time complexity should be reduced in order to reduce resource consumption (O. Sallou, 2015), (M. Villari, 2016). The formal procedure should be used for this. Formal techniques are often employed in computing science to verify truths about algorithms and systems. (S. Shirinbab, 2018), (C. Boettiger, 2015) and (Dua R, 2016). Researchers may be interested in formal speciation of a software component in order to automate the variation of that component's implementation. Alternatively, researchers may be interested in an algorithm's time or space complexity, as well as the accuracy or quality of the answers it generates. (M. Luthfi, 2017), (R. Morabito, 2016), (Dua R, 2016). The main reason for this research is to give a better system and the research will be successful when the hardware and fully optimized by following my research methodologies- model and formal methods.

B. A Survey on Docker Container and its use cases

The criteria and explanation of Docker Container and use cases are given below:

a) Simplifying Configuration-

This is the most common use case; Docker setups may be reused in different contexts several times. Virtual machines have an advantage when operating any platform with its configurations; Dockers have the same advantage, but without the Virtual Machine overhead, and allow users to put the configurations and environment into the code and distribute it. The application environment and infrastructure needs are separate. In terms of real-world use, it allows businesses to speed up project setup by allowing them to go right into work without going through the tedious process of setting up environments and configuration processes.

b) Code Pipeline Management-

A few minor differences can be observed as code written in a developer environment progresses through various stages (each of which uses different platforms/environments) and approaches the production stage; however, Dockers provide a consistent environment throughout all stages from development to production, allowing for a simple development and deployment pipeline. The steady nature of the Docker image and the ease with which it may be launched can help with the pipeline mentioned above management.

c) Docker for Production Efficiency-

In a development environment, Docker makes it simpler to meet two goals: keeping a developer near to production while still allowing them to work remotely. The second need is an interactive development environment, which Docker meets by making application code from the host OS accessible to the container through shared volumes. This provides a number of advantages, including the developer's ability to make changes to the source code from his preferred platform and see the results. Multi-Tenancy - Docker is utilized in multi-tenant systems since it helps to avoid significant application rewrites. The codebases of multi-tenant systems are far more complicated, stiff, and difficult to manage. Redesigning an application takes a significant amount of effort and money. Docker simplifies creating limited environments to execute numerous apps for each tenant easier and more cost-effectively. Docker's simple API makes it possible to spin up containers programmatically.

d) Debugging Tools-

Docker offers many tools that work well with the notion of containers. One of its features is the ability to checkpoint containers and their versions, distinguish two containers, and quickly repair applications when necessary.

e) Improved Disaster Recovery-

In the event of a disaster, a Docker image, also known as a snapshot, may be saved and retrieved at a specific point in time. A Docker image may be

used to transition between two distinct versions of the same software, and a file can be duplicated to new hardware using Docker.

f) Increased DevOps Adoption-

To standardize confined deployment, the DevOps community has developed foundations on Docker. Docker's relationship with DevOps has been established via CI/CD, and Docker ensures consistency in both testing and production settings. Docker simplifies machine setup by standardizing the configuration interface. Docker might be utilized to assist the company improve its DevOps.

IV. Related Work

As container technologies have grown in popularity, a number of research projects have been established to better understand container boundaries, increase container performance and security, and compare containers to virtual machines. (N. Thoai, 2016). Carl examined the common obstacles that prevent many research projects from becoming replicable, as well as how Docker container technology may assist. Many projects, for example, need certain requirements in order to recreate the same findings as the original researchers, but due to the project's diverse underlying OS and hardware, it's impossible to just provide an installation script. In order to tackle this issue, Docker was created, which delivers a tiny binary image that includes all of the necessary software. In Paolo's company and with his other pals You may use Docker to link the implementation of many containers and run a channel application in Genomic pipelines. (C. Boettiger, 2015) In the company of Douglas' buddies One of the benefits of using containers in HPC systems is their versatility and reproducibility, according to the research. Several scholars have investigated the advantages of cloud computing container technology. According to the research team, cloud-based platforms such as Platform-as-a-Service (PaaS) may help containers because of the simplicity with which they may be used, set up, and deployed. You can count on Roberto for container efficiency and safety. (Raho M, 2015) In the Internet of Things area, a performance assessment of employing containers was presented. Roberto proved that the containerized layer has a minor effect compared to native execution by managing containers on top of a single board computing device like the Raspberry Pi 2. The use of container technologies like Linux Containers, Linux VServer, and OpenVZto has been shown by Miguel and his colleagues to provide a very low overhead HPC environment when compared to native systems. Thanh (Docker, 2018) Docker's internal agents offer security, and how they interact with the Linux kernel's security features was investigated. Sergei et al. (J. Liu, 2006) used Intel SGX authorized execution technology to solve container security and prevent outside attacks. Many individuals have been interested to studies comparing receptacles and virtual machines. (A. M. Joy, 2015), (Z. Kozyev, 2017), (Lars, 2019), (Docker, 2018), (S. Shirinbab, 2018), (Preeth , 2015), (Nguyen, 2016), (Goldberg, 1974), (Chae, 2019).

There was just one physical machine used in Janki et al. study, which does not reflect the normal cloud architecture when looking at the performance of Spark processes operating inside a container cluster vs a virtual machine group. It was also endorsed by Claus (Z. Kozyev, 2017). Virtualization is used by containers and virtual machines alike (VMs). Using containers to package and manage applications and the PaaS cloud, the author claims, is a superior solution since containers better support microservices. Using these tools, we were able to measure how much CPU, RAM, and the network were being consumed by various containers and virtual machines. When random disk access is necessary, containers outperform VMs for disk and network I/O-intensive workloads, even though both have moderate CPU and memory performance overhead. Containers and VMs were permitted to produce 100000 factorials through I/O. Kyoung-Taek et al. (Nguyen, 2016) for this research and their boot uptime and computation performance were examined. VMs and containers cost the same amount of energy under idle, CPU/Memory stress, and network-intensive workloads, according to Roberto (Goldberg, 1974). However, containers require less power for these workloads.

V. Docker Usage

A. When to Use :

a) Learning About New Technologies:

Docker enables individuals to use a new tool in a disposable and isolated environment without spending time installing and configuring it. Several projects save docker images containing apps that have already been installed and configured.

b) Main Use Cases:

Docker Hub is a cloud registry service that allows customers to get Docker images created by other communities; it is a convenient way to get images for either Basic or Standard applications.

c) Program Isolation:

Managing each application component in its container eliminates dependency on other apps on a different server.

d) Dev Groups:

For developers working in various settings, Docker provides a close match between local development and production environments.

B. When Not to Use :

a) Sophisticated Apps:

Unlike simple applications, complicated applications will benefit from using pre-obtained docker files or fetching images from Docker Hub since modifying, constructing, and managing requests and replies across several containers on multiple servers is time-consuming.

b) Behaviour of Applications:

Docker outperforms VMs since Containers share the host kernel and imitate the complete operating system when it comes to performance. Dockers can be avoided to get the most remarkable performance from the server since processes operating on a native OS are quicker than processes running within a container.

c) Upgrade Troubles:

Docker is still a developing technology that needs regular upgrades to take advantage of new features.

VI. Performance Evaluation

For the purposes of this essay, a four-core LINUX machine running Docker was used, which did not need the use of a hypervisor. The host computer has 4GB of RAM, a 512GB hard drive, and a Core i5 processor running LINUX Ubuntu 12.04 64-bit OS.

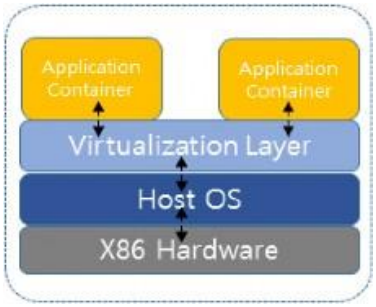


Fig. 1. Virtualization at Operating System Level

TABLE I. Table Type Styles

Virtual Machine	Docker
Virtual machines run on virtual hardware, and the guest operating system is loaded into memory.	All visitors have access to the physical memory of the Host OS.
Guests communicate via network devices, which may or may not be software.	Pipes, sockets, bridges, and other means of communication are used to connect visitors.
The hypervisor is in charge of security.	There are no security measures in place.
Because of its complexity, it has a higher overhead.	Because the containers are light, there is less overhead.
It is not feasible to share libraries or files.	File sharing is available (for example, using LINUX's SCP command).
It takes a long time to boot up.	Faster start-up.

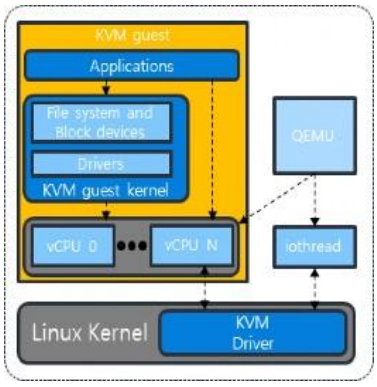


Fig. 2. KVM Architecture

A. *Bonnie ++ :*

Hard drives and file systems may be tested using the C++ Bonnie++ (O. Sallou, 2015) benchmarking suite. The following picture shows a Docker container running Bonnie++. Several I/O performance tests are included in Bonnie++. A file system's performance may be gauged by looking at how long it takes to read or write data. Data with a fixed size of 40MB was used to evaluate the container's reading and writing speeds. Bonnie++'s execution can be tracked down using the container id: 111c5ab491fe. There were precisely 40 million putc () operations per second in Bonnie's putc () call output rate of 507 k per second. Windows 98 was the most popular operating system.

TABLE II. Bonnie++ On Docker

Container ID	File size	Sequential output		Sequential input	
-	-	Output rate	CPU time	Input rate	CPU time
256e3sh643rg	46Mb	502Kb/s	96%	1342Kb/s	95%

On an Ubuntu 12.04 machine, the following is the result of executing Bonnie++. To highlight the performance differences between Docker and the host operating system, this experiment has been set up to do just that. The host and guest operating systems run at different speeds. The Host OS utilizes much less of the computer's resources than a container running on top of it. As a consequence, the Host OS is a little bit quicker than the Guest OS.

TABLE III. BONNIE++ ON HOST OS

Container ID	File size	Sequential output		Sequential input	
-	-	Output rate	CPU time	Input rate	CPU time
User-OptiPlex	46Mb	576Kb/s	96%	5784Kb/s	93%

B. Psutil :

A Python package known as (M.Villari, 2016) psutil shows data about running processes and how the system is being used (CPU, memory, storage, and network). Monitoring, profiling, assigning resources and controlling processes are the most common uses of this tool. The 32-bit and 64-bit architectures support a wide range of operating systems, including LINUX, Windows, FreeBSD, Sun Solaris, and many more options as well.

a) Memory management:

Retrieving the memory used by containers is a memory management feature: According to the table, memory use will be given in numerous areas, with each field represented by a byte.

• Entire: The whole amount of physical memory on the host system is shown here.

• Used: Previous apps have used up all of this RAM.

• Memory that isn't being used but is still accessible is referred to as "free memory." In addition to what has already been said.

• Active memory refers to memory that is now or recently in use. This is a command in the UNIX operating system.

• Buffers: Metadata from the file system is cached in memory.

• Cache: A memory that is used to keep track of a large number of objects. This is utilized to increase the performance of the system. A total of 20.2 percent of RAM was utilized. The memory usage test for the Docker container yielded the following results.

b) CPU times:

Returns the number of times the CPU has been used. Time spent in various modes is shown on the screen. The following table lists many aspects that denote the passage of time, each having its own unique qualities based on the platform on which it is shown.

TABLE IV. CPU TIMES OF DOCKER AND HOST OS

Machine	User	Nice	System	Idle	Iowait	Irq
Docker	24.22	20.41	18.20	13312.87	87.88	3.51
Host	24.93	20.41	18.64	13656.47	91.24	3.87

TABLE V. MEMORY USAGE OF DOCKER AND HOST OS

Machine	Total	Used	Free	Active	Buffers	Cached
Docker	3915586415	0965808211	2420113432	6468111184	1213435235	7513455412
Host	3915586415	1252315764	2623308546	5746200357	1213555360	7512332342

VII. FUTURE WORKS

The switch from virtual serves to container-based infrastructures in large-scale computing infrastructures allows for faster and more efficient software deployment. Fog computing systems, on the other hand, are generally made up of incredibly tiny machines like Raspberry PIs, and even launching a simple Docker container might take several minutes. The slowest of the three basic resources: network connection, CPU, or disk I/O, now affects deployment time, depending on the hardware. As hardware improves, the bottleneck may shift from one to the other. Docker-pi, on the other hand, will make the most of all available hardware to the greatest degree feasible, regardless of the system's characteristics. Our findings could aid practitioners and academics in making better decisions about how to set up cloud infrastructure and big data applications for optimal performance and resource use. In the future, we will examine many more features of the container and virtual machine environments as part of our research. Another approach is to investigate how changes in image file structure and design effect application performance across containers and virtual machines.

VIII. CONCLUSION

Docker is a container-based software development platform that is free and open source. Before you can comprehend Docker, you must first grasp the concept of containers. Containers are a "lightweight, stand-alone, executable piece of software that contains everything essential to run it," according to Docker. Docker can operate on both Windows and Linux systems since containers are platform agnostic. Docker may even be run within such a virtual machine if necessary. Docker's main goal is to enable client-server applications to operate in a distributed environment. Docker seems to perform well in Bonnie++ and the psutil performance tool, and its performance may be comparable to that of a bare-metal operating system. Docker promises to be faster than a virtualized node, however this has yet to be shown in practice. We expect the Docker community to address the main security problems in Docker container technology in the future.

REFERENCES

Preeth, E. N., Mulerickal, F. J. P., Paul, B., & Sastri, Y. (2015, November). Evaluation of Docker containers based on hardware utilization. In *2015 international conference on control communication & computing India (ICCC)* (pp. 697-700). IEEE.

Bella, M. R. M., Data, M., & Yahya, W. (2018, November). Web server load balancing based on memory utilization using Docker swarm. In *2018 International Conference on Sustainable Information Engineering and Technology (SIET)* (pp. 220-223). IEEE.

Moniruzzaman, A. B. M., Waliullah, M., & Rahman, M. (2014). A High Availability Clusters Model Combined with Load Balancing and Shared Storage Technologies for Web Servers. *arXiv preprint arXiv:1411.7658*.

Sallou, O., & Monjeaud, C. (2015). Go-Docker, A batch scheduling system with Docker containers. IEEE International Conference of Cluster Computing, pp. 514-515.

Villari, M., Fazio, M., Dustdar, S., Rana, O., & Ranjan, R. (2016). Osmotic Computing: A New Paradigm for Edge/Cloud Integration, IEEE Cloud Computing, 3 (6), 76–83.

Preeth, E. N., Mulerickal, F. J. P., Paul, B., & Sastri, Y. (2015, November). Evaluation of Docker containers based on hardware utilization. In *2015 international conference on control communication & computing India (ICCC)* (pp. 697-700). IEEE.

Julian, S., Shuey, M., & Cook, S. (2016, July). Containers in research: initial experiences with lightweight infrastructure. In *Proceedings of the XSEDE16 Conference on Diversity, Big Data, and Science at Scale* (pp. 1-6).

Docs, D. (2018). Swarm mode key concepts. *Disponıvel em https://docs. docker. com/engine/swarm/key-concepts*.

Fazio, M., Celesti, A., Ranjan, R., Liu, C., Chen, L., & Villari, M. (2016). Open issues in scheduling microservices in the cloud. *IEEE Cloud Computing*, *3*(5), 81-88.

Chae, M., Lee, H., & Lee, K. (2019). A performance comparison of linux containers and virtual machines using Docker and KVM. *Cluster Computing*, *22*(1), 1765-1775.

Huang, W., Liu, J., Abali, B., & Panda, D. K. (2006, June). A case for high performance computing with virtual machines. In *Proceedings of the 20th annual international conference on Supercomputing* (pp. 125-134).

Docs, D. (2018). Swarm mode key concepts. *Disponıvel em https://docs. docker. com/engine/swarm/key-concepts*.

Ahmed, A., & Pierre, G. (2020). Docker-pi: Docker container deployment in fog computing infrastructures. *International Journal of Cloud Computing*, *9*(1), 6-27.

Preeth, E. N., Mulerickal, F. J. P., Paul, B., & Sastri, Y. (2015, November). Evaluation of Docker containers based on hardware utilization. In *2015 international conference on control communication & computing India (ICCC)* (pp. 697-700). IEEE.

Chung, M. T., Quang-Hung, N., Nguyen, M. T., & Thoai, N. (2016, July). Using docker in high performance computing applications. In *2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)* (pp. 52-57). IEEE.

Bella, M. R. M., Data, M., & Yahya, W. (2018, November). Web server load balancing based on memory utilization using Docker swarm. In *2018 International Conference on Sustainable Information Engineering and Technology (SIET)* (pp. 220-223). IEEE.

Joy, A. M. (2015, March). Performance comparison between linux containers and virtual machines. In *2015 international conference on advances in computer engineering and applications* (pp. 342-346). IEEE.

Data, M., Luthfi, M., & Yahya, W. (2017, November). Optimizing single low-end LAMP server using NGINX reverse proxy caching. In *2017 International Conference on Sustainable Information Engineering and Technology (SIET)* (pp. 21-23). IEEE.

Shirinbab, S., Lundberg, L., & Casalicchio, E. (2020). Performance evaluation of containers and virtual machines when running Cassandra workload concurrently. *Concurrency and Computation: Practice and Experience*, *32*(17), e5693.

Boettiger, C. (2015). An introduction to Docker for reproducible research. *ACM SIGOPS Operating Systems Review*, *49*(1), 71-79.

Morabito, R. (2016, April). A performance evaluation of container technologies on internet of things devices. In *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)* (pp. 999-1000). IEEE.

Dua, R., Kohli, V., Patil, S., & Patil, S. (2016, December). Performance analysis of union and cow file systems with docker. In *2016 International Conference on Computing, Analytics and Security Trends (CAST)* (pp. 550-555). IEEE.

Goldberg, R. P. (1974). Survey of virtual machine research. *Computer*, *7*(6), 34-45.

Raho, M., Spyridakis, A., Paolino, M., & Raho, D. (2015, November). KVM, Xen and Docker: A performance analysis for ARM based NFV and cloud computing. In *2015 IEEE 3rd Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE)* (pp. 1-8). IEEE.

Kozhirbayev, Z., & Sinnott, R. O. (2017). A performance comparison of container-based technologies for the cloud. *Future Generation Computer Systems*, *68*, 175-182.

Preeth, E. N., Mulerickal, F. J. P., Paul, B., & Sastri, Y. (2015, November). Evaluation of Docker containers based on hardware utilization. In *2015 international conference on control communication & computing India (ICCC)* (pp. 697-700). IEEE.

Chung, M. T., Quang-Hung, N., Nguyen, M. T., & Thoai, N. (2016, July). Using docker in high performance computing applications. In *2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)* (pp. 52-57). IEEE.

Zhang, Q., Liu, L., Pu, C., Dou, Q., Wu, L., & Zhou, W. (2018, July). A comparative study of containers and virtual machines in big data environment. In *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)* (pp. 178-185). IEEE.

Ahmed, A., & Pierre, G. (2018, July). Docker container deployment in fog computing infrastructures. In *2018 IEEE International Conference on Edge Computing (EDGE)* (pp. 1-8). IEEE.

Shirinbab, S., Lundberg, L., & Casalicchio, E. (2020). Performance evaluation of containers and virtual machines when running Cassandra workload concurrently. *Concurrency and Computation: Practice and Experience*, *32*(17), e5693.

Chae, M., Lee, H., & Lee, K. (2019). A performance comparison of linux containers and virtual machines using Docker and KVM. *Cluster Computing*, *22*(1), 1765-1775.