THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF SECURITY AND RISK ANALYSIS

BENEFITS OF CYBER SECURITY LABS ON RASPBERRY PI'S

CARSON BROWN
SPRING 2019

A thesis
submitted in partial fulfillment
of the requirements
for baccalaureate degrees
in Information Science and Technology and Security and Risk Analysis
with honors in Security and Risk Analysis

Reviewed and approved* by the following:

Michael Hills
Associate Teaching Professor of Information Sciences and Technology
Thesis Supervisor

Edward J. Glantz
Teaching Professor of Information Sciences and Technology
Honors Adviser

* Signatures are on file in the Schreyer Honors College.

ABSTRACT

With the growing number of cyber-attacks occurring in the world today, the need for robust cyber security education has never been greater. With many universities putting together their own respective cyber security curriculums, it is imperative to establish the most efficient ways to teach cyber security. This study intends to guide the decision makers toward using hardware, in our case Raspberry Pis, to perform labs instead of on virtual machines, which are the current standard. The beginning chapters will dive deeper into the studies purpose and explain to the reader to the current cyber security landscape. The next chapter describes how the labs and surveys were devised and explain the decision to use a Raspberry Pi as the Hardware for the study. The following chapters examine the results of the study and how they may impact future studies and curriculum. The final chapter discusses how this study can be continued and what further steps should be taken by those who believe in Raspberry Pis in the classroom.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank the following people for supporting me through this process:

- Michael Hills
- Sarah Stager
- Tim and Louise Brown
- My friends in IST
- Ed Glantz

Chapter 1

Purpose

In recent years, cyber security has become one of the most discussed subject areas due to the increasing number of cyber-attacks and their increased damage and impact. Cyber security is beginning to account for a more substantial portion of individual company's budgets, and with every new attack, the global awareness increases. With entire nation-states starting to fund cyber-attacks, the growing need for cyber security professionals is becoming apparent.

Historically, many cyber security professionals have not come from security backgrounds, but instead, follow unique paths into the profession. For example, the head of Cyber Security at Equifax, who was the victim of one of the worst cyber-attacks in history, was a Music major in college. While this sounds ridiculous, it is not uncommon since cyber security professionals have often been rebellious types who end up learning about the subject through trying to hack and break things on their own (Wolff, 2018). This trend has led to many former criminals getting into the industry, and few professionals with backgrounds strictly tailored toward cyber security.

Examples of former criminals include Kevin Mitnick, who in the mid-nineties was on the FBI's most wanted list for hacking into 40 major corporations, is now one of the most in demand cyber security consultants in the industry. Although these criminal avenues are creating highly competent cyber security professionals, they are promoting nefarious methods of learning to new comers interested in the field. It is important that cyber security professionals come from a tailored education because if not, young enthusiasts will all want to commit crimes until they get

recognized for their ability. This touches on the major problem of balancing teaching students how to commit crimes while hoping they only use that knowledge to prevent the crimes from happening in the future.

The solution to this is stronger cyber security education providing students with a well-rounded understanding of the many issues that relate to the current cyber security landscape, while preventing them from using what they learn in class in nefarious ways. While different universities take different approaches to teach cyber security, all of them leverage virtual machines to give students exposure to many types of environments while minimizing the costs. Although virtual machines allow for lots of flexibility, they limit the student's exposure to many important aspects of cyber security such as computer setup and architecture, and in the end, help create unprepared professionals.

This study will look at the Raspberry Pi as a potential replacement for virtual machines in the cyber security classroom. Albeit virtual machines are incredibly dynamic and low cost, they do not expose a student's computer hardware or computer setup. By exposing students to these areas, they will become more aware of computer architecture and better understand the scope of cyber security. Due to the Raspberry Pi's low cost, they can be added to the classroom without impacting the budget too much and give students a more hands-on experience that they would not get without the device.

Chapter 2

Background on Cyber Security Education and Raspberry Pi's

Cyber Security and Cyber Security Education

To understand the history of cyber security education, it is first essential to understand the history of vulnerabilities and bugs. While terms like virus and worm only existed in Biology 40 years ago, they have become more associated with hacking and computers in more recent years. The first worm, a computer program that spreads itself across multiple devices, was created by a Cornell graduate student named Robert Morris who wanted to gauge the size of the internet. However, due to a coding error, the worm began to shut down systems, exposing how delicate computer security was. This hack, now known at the Morris Worm, was a major point in computer security and is covered in most cyber security curricula. Robert Morris was the first person convicted under the Computer Fraud and Abuse Act and is now a tenured professor at MIT. His profile on the MIT website makes no note of his development of the Morris Worm (Kelty, 2018).

After the fallout of the Morris Worm, the Computer Emergency Response Team was formed with the sole function of researching issues that might affect the internet in the future. On the other end of the security spectrum, malicious actors became inspired and started developing more impactful viruses and worms. This sparked the ongoing divide between the cyber criminals and researchers who battled on either side of the security spectrum, both benefiting from every new discovery made.

Since this first worm, technologists have been going back and forth discovering avenues to break into computers and subsequently patching those avenues or closing the doors the hackers used to gain access. While cyber security was left to computer scientists earlier on, most cyber security professionals were coming from hackers-turned-good-guy types. Currently, most cyber security professionals over the age of 30 do not have a cyber security related degree, and many don't even have a computer science degree. This problem has led to cyber security bachelor's degrees popping up at universities around the world. However, cyber security education was mainly just forums of people discussing hacking in the past, and universities have struggled to find a consistent curriculum that adequately educates all students (Wolff, 2018).

The Raspberry Pi

The Raspberry Pi is a credit card sized single-board computer that was invented to spread Computer Science education. It was invented by Eben Upton, who was trying to find unique ways to get students interested in his computer science class at the University of Cambridge. In an interview Upton stated, "You can do two things with Moore's Law: You can keep the price constant and add features, or you can keep the features set constant and lower the price… What we feel we're doing is using Moore's law to save people money" (Upton, 2015).

The Raspberry Pi has an ARM processor and can be used along with all standard computer peripherals (monitor, mouse, keyboard) (Benchoff, 2016). It was developed as a low-cost computer that could help the spread of Computer Science but was quickly adopted into many different domains, skyrocketing the computer into prominence for computer hobbyists of all types. The main advantages of the Raspberry Pi are its cost, accessibility, customizability, and

many open-source projects existing for new users. As of March 2018, there have been 25 million Raspberry Pi's sold (Pilch, 2019).

The Raspberry Pi is already being used by many people to learn cyber security. While the Pi mainly uses its own flavor of Linux, Raspbian, it can also run Kali Linux which is the go to operating system for cyber security professionals. Pi's have been used to build honeypots to lure in criminals, add DNS level ad-blocking to a network to prevent all ads, and sanitize USB's to block any malicious code that may be passed along by the USB. By having students complete the projects listed above, they learn about the different areas of security each project relates to. The Raspberry Pi has even been used by fictional hackers on TV in the show *Mr. Robot* (Buckley, 2018)

The Raspberry Pi was chosen for this study because it was cheap and easy to use during the labs. While this study could be extrapolated out to "The benefits of using Hardware in Cyber Security," it was easier to confine the study to a single device as it would be the easiest to adopt for future research or implementation by other organizations.

Chapter 3

Literature Review

According to data presented at the National Initiative for Cyber Security Education Conference in November 2018, to fill all current open Cyber Security positions, the current Cyber Security workforce would need to increase by around 40%. That's 313,735 open positions with a current workforce of approximately 715,000. The number of open positions is expected to climb into the millions in the next three years, and Universities are trying to capitalize on the need for educated Cyber Security professionals (Wolff, 2018).

When Fred Schneider, a Computer Science Professor from Cornell, discussed Cyber Security Education in an essay for the IEEE Security & Privacy Journal, he blamed Cyber Security educations shortfalls on the lack of an established structure around the field. Unlike fields like Finance, Biology, and others, Cyber Security does not have a generally accepted curriculum for the University level that can be implemented by the full gambit of participants. Universities are independently developing curriculum, and many are taking different directions, leading to inconsistent knowledge across the board of Cyber Security professionals. Schneider claimed Cyber Security in Universities lacked input from top technical researchers, and recommended an annual conference that would connect curriculum creators and top researchers so that all Universities could get on the same page (Schneider, 2013).

A group from George Washington University drew a similar conclusion to Schneider in their article *Holistically Building the Cyber Security Workforce.* In this article, they define the four areas identified as "Cyber Security," those being information technology management,

electronics engineering, computer engineering, and telecommunications. Just as Schneider discussed, the group identified the lack of a consistent curriculum as the largest reason education hasn't been able to put people in those roles at the rate needed. The George Washing University group also concluded oversight of national Cyber Security programs required development, but addressed the divide in areas encompassed by Cyber Security may require separate, specialized committees identifying necessary areas of improvement in their respective fields (Lance, Diana, & Costis, 2012).

A group from the University of Hawaii also took a look at the state of Cyber Security education and noted the gaps they found in a paper on *Re-Engineering Cyber Security Education*. They note that industry leaders have claimed graduates do not have enough hands-on experience. The DHS has recommended a stricter curriculum that would highlight hands-on labs to force Universities to increase the amount of hands-on work. This recommendation touches on the long-argued topic of Education vs. Training at Universities. The needs in the industry have highlighted the necessity of exposure to both education and training, but most Universities view themselves as above training (Conklin, Cline, & Roosa, 2014).

A methodology developed by Apple Computer's Inc. called Challenge Based Learning has proven to be a successful format for Cyber Security education through studies performed at the University of Massachusetts. This methodology splits into steps that guide a student to Cyber Security discoveries without handing them the information, and results have shown Students have higher perceived knowledge and interest in Cyber Security after participating in Challenge Based competitions about Cyber Security. The study split students into small groups consisting of a research portion and a competition portion all surrounding Cyber Security. These tests

require more time and attention than other education techniques but have results that justify the means (Cheung, Cohen, Lo, & Elia, 2011).

In a presentation titled *Toward Curricular Guidelines for Cyber Security* funded by the National Science Foundation, a group of Professors from multiple different Universities weighed in on what they felt were the most critical parts of developing future Cyber Security curriculum. They believed the breadth of Cyber Security needs requires progress in 2-year, 4-year, and Master's program levels to fill all the current Cyber Security positions with quality candidates. They also stressed the importance of developing the mindset of a professional along with just developing technical skills, which means curriculum should include subjects such as psychology, criminal justice, policy and economics and more. Beyond all this, however, proper technical development is the primary concern of developing curriculum (Impagliazzo, Dark, Cassel, McGettrick, & Hawthorne, 2014).

While many of the people referenced above believe there needs to a be an overarching curriculum for universities to follow, NIST, the National Institute of Standards and Technology, has been attempting to make this happen. The National Initiative for Cybersecurity Education, or NICE, assembled a framework of the all the skills needed to work in cyber security. The framework, which can be seen in Table 1, aims to highlight all the important subjects NIST feels a cyber professional should master.

Table 1. NICE Cybersecurity Workforce Framework

Recommended Skills	***Specialty Areas***
Analyze	• All-Source Analysis • Exploitation Analysis • Language Analysis • Targets • Threat Analysis

Collect and Operate	• Collection Operations • Cyber Operational Planning • Cyber Operations
Investigate	• Cyber Investigation • Digital Forensics
Operate and Maintain	• Customer Service and Technical Support • Data Administration • Knowledge Management • Network Services • Systems Administration • Systems Analysis
Oversee and Govern	• Cybersecurity Management • Executive Cyber Leadership • Legal Advice and Advocacy • Program/Project Management and Acquisition • Strategic Planning and Policy • Training, Education, and Awareness
Protect and Defend	• Cyber Defense Analysis • Cyber Defense Infrastructure Support • Incident Response • Vulnerability Assessment and Management
Securely Provision	• Risk Management • Software Development • Systems Architecture • Systems Development • Systems Requirements Planning • Technology R&D • Test and Evaluation
Acquired from NICE	

With many universities beginning to align their curriculums to the NICE framework, the problem of inconsistent curriculums will begin to disappear. The NSA has started to identify academic institutions that align to this framework as Centers of Academic Excellence. Penn State, as well as many other universities, are on this list (NICE, n.d.).

Although the cyber security field struggles in developing curriculum, it has experienced great success in Capture the Flag competitions (CTFs). Research performed at the University of Birmingham considered how CTFs could work in the classroom. They addressed the three

biggest factors blocking this adoption; the large infrastructure needed for a CTF competition, the necessary continuous supervision by organizers, and necessity of attacking vulnerable servers; and created a solution that would allow CTFs to enter the classroom. They devised virtual machines that had unique flags that students would need to locate and submit for a grade. As the course continued, subjects covered in the class would reveal new possible vulnerabilities for students to attack and improve their score on the assignment. The virtual machines would run on the student's local machine so there would be no risk of them accidentally attacking the universities servers and course staff would not need to monitor the network. At the end of the course, students claimed they spent 6-hours a week on the assignment and ranked the class as one of the most challenging courses taken. The course was also extremely popular due to the excitement of the CTF aspect (Chothia & Novakovic, 2015).

Chapter 4

Methodology

This study attempted to gain information on students perceptions of labs performed on Raspberry Pi's to further understand whether students would prefer using the device in the future. Data were acquired through surveys given to the class after they completed different labs that would typically take place on a virtual machine. The surveys used both paper and Google Forms to retrieve data, and the subjects in the tests all came from Dr. Michael Hills' IST 451 Network Security class. Surveys and labs were distributed in class on paper and were available on the classes Canvas page for more accessibility.

The surveys and labs, which can be found in Appendix A, were based on the classes curriculum and labs already being performed in the class before the Pi and this study was introduced. The surveys had both numerical scale and open-ended questions to gain quantifiable data as well as qualitative feedback on the study.

The numeric scale questions ranked from Strongly Disagree (1) to Strongly Agree (5) with Neutral (3) being in the middle. These questions were used to gain insight into information such as lab quality, lab enjoyment, appropriate challenge, and certain specific points about each lab. The open-ended questions leaned toward retention checks (questions about the subject covered in the labs) and general opinion questions concerning the quality of the labs and possible areas of improvement.

There was also information gained from observations of the student's participation in the Labs. These points were recorded separately from the survey responses and provided information on how students participated in the labs as opposed to what the team expected. Individual conversations were also held with all students to pull more insight out of them.

The combination of these different avenues of data acquisition provided a well-rounded data set that was used to draw the conclusions covered in Chapter 6. In the class of 50 students, a total of 39 (78%) responses were received for the first survey, 30 (60%) responses for the second survey, and 24 (48%) for the third survey. While these numbers were not ideal for gaining optimal insight, they were all that could be acquired from only working with one class of students.

Chapter 5

Results

The survey results showed strong support for the labs performed on the Raspberry Pis. Survey results, which can all be found in Appendix B, all averaged between (3) Neutral and (5) Strongly Agree for all positive questions, especially questions about the value of performing the labs in the future. Questions related to the labs themselves also received favorable responses, however, some students who had gotten frustrated throughout the lab did not give favorable feedback.

While many of the questions asking about other students in general received very high averages, questions asking about the student's own knowledge gain did not receive as favorable reviews. This can be seen in Figure 1 which averaged a 4.5 out of 5 response, opposed to Figure 2 which averaged a 3.8 out of 5. Figure 1 depicts responses on whether other students would benefit from the lab, while Figure 2 depicts responses on whether the students themselves felt as if they gained knowledge. This difference could be due to students thinking others would perform better than them, or because students let other group members take up much of the work.

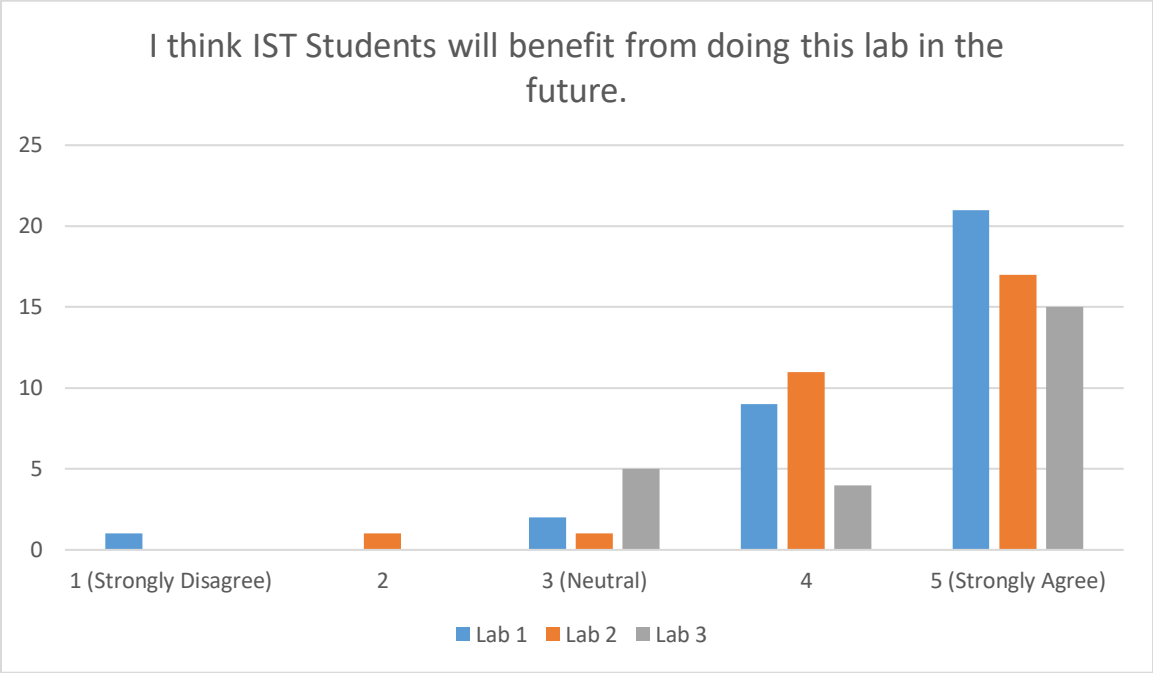

Figure 1

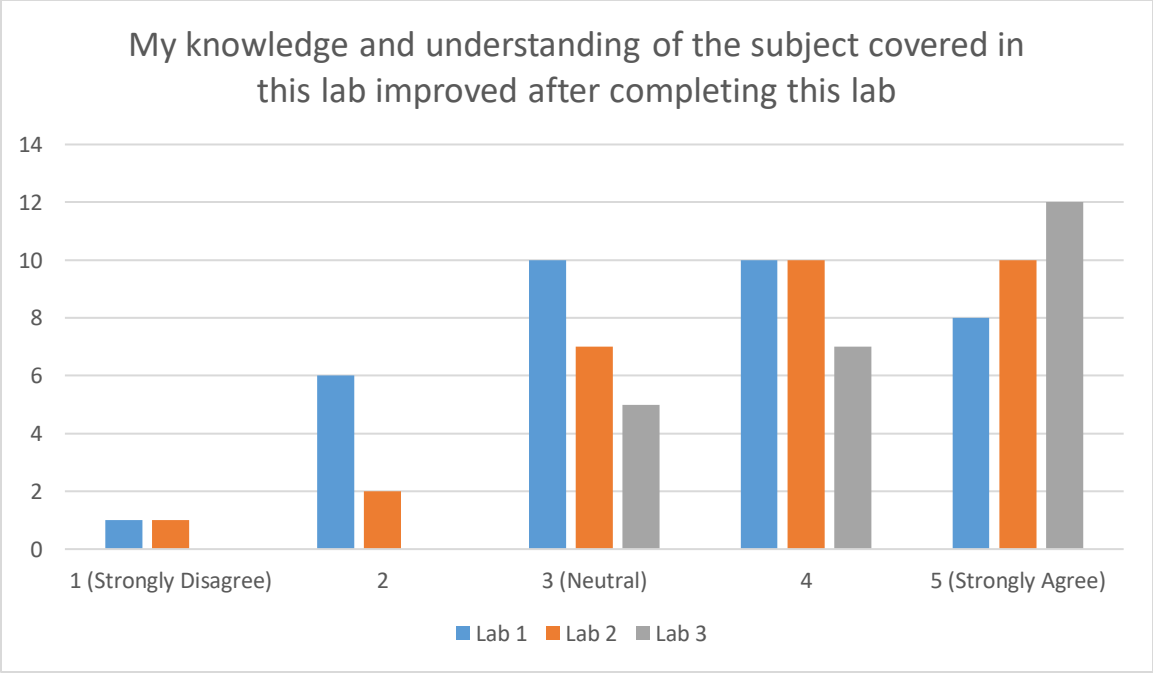

Figure 2

When observing the class participate in the labs, it became clear that while students seemed collectively very interested in the labs, there was always one student working the

keyboard and calling most of the shots for the group. For some groups, members who were not as involved would sit back and let a single group member complete the lab on their own. This problem seemed accentuated in groups that had highly skilled members as they would be able to complete the lab quickly on their own.

Lab 2 had some misleading instructions that frustrated some students but ironically seemed to cement concepts more effectively since the students were in charge of finding the correct command instead of using the one given to them. While survey responses for Lab 2 were less favorable because of this, it was clear when talking to students that they felt finding the correct solution was better than step by step instructions.

An error was made in the survey for Lab 3 where “Strongly Disagree” and “Strongly Agree” were switched. This error is seen in Figure 18, where the amount of Strongly Disagree responses were higher than any other survey question given. Due to this, these two responses were not used to make conclusions. Lab 3 had more open-ended instructions that required a little more research by students, which resulted in better feedback about the lab.

Chapter 6

Analysis of Results

Based on the results from the surveys, students all seemed to find value in the labs performed on Raspberry Pis. While the team was unable to quantify any comparative data, conversations with student participants proved they preferred these labs over virtual machine labs. Many students seemed unsatisfied with the preconfigured machines as they made it too easy for them to complete the labs. Those who did not like the labs were often the ones struggling to get through them and were confusing easiness with quality.

Based on the feedback and results from Lab 2, students felt they developed a better grasp of the topic even though and error was made in one of the instructions on the lab. While this discovery was unintentional, it became clear that by forcing students to research the proper way to accomplish the goal on their own, they developed a better understanding of the subject being presented. This means that there needs to be a balance found in guidance provided to students; too much guidance decreases retention and understanding of the topic, but not enough guidance can lead students to dead ends or incorrect solutions. More research should be done into finding the correct balance since it could shape how labs are developed in the future.

When analyzing student's interactions with the Raspberry Pi, it was clear that physical hardware was better for engagement than virtual machines. The process of assembling the machine, installing the operating system, and booting up when compared to just logging into a virtual machine appeared to students more invested in the lab since the beginning was a more involved process. This setup process fostered more experimentation by the students since they

knew they had control over a fresh machine instead of a configured virtual machine. Examples of experimentation that occurred in class which included installing and testing different operating systems, such as Kali Linux and Ubuntu, and writing different custom scripts to $Path variables. Experimentation is the goal of every introductory lab because it fosters further learning and interest in the subject.

One of the major conclusions was that the group size requires heavy consideration when designing these labs. While 4-5 member groups are great for larger scale projects, they are too big for these single session labs. Groups of 2-3 would work better, or preferably an individual device for each student. The problem with this is the cost increase of decreasing group sizes since it would increase the number of Pis necessary for each class. This added cost is the reason for the allure of virtual machines; they can be spun up at zero cost and provide every student with a personal interface. Sometimes, however, it is more important to provide better experiences for students at the expense of more hardware. Therefore at $35, Raspberry Pi's are the perfect devices for these labs. They keep costs low but give students a more hands-on experience. On top of this, with text book prices rising into the hundreds, asking students to purchase their own Pi instead of buying a text book would save them money in most cases. Also, due to the plethora of functions able to be performed by the Pi, it could be used across multiple classes opposed to a text book that normally only applies to a single class.

Chapter 7

Conclusion

Based on the number of cyber-attacks happening around the world and the lack of a consistent cyber security curriculum across universities, improvement to the cyber security education process has become evident. While many different solutions are offered, the Raspberry Pi could emerge as the ideal platform for initial exposure to cyber security students. Its ability to give students a better understanding of Hardware and computer setup at a reasonable price makes it extremely practical for many levels of computer experience. While more research will need to be done, the Pi should be leveraged much more heavily in the beginner and intermediate levels of cyber security education. Other areas of further research should include different lab design styles, different Raspberry Pi peripherals, and open-source labs that could be downloaded by anyone in the world with a Pi.

Appendix A

Labs and Relating Surveys

Lab 1: Introduction and Installing Linux

Linux started as a small open source hobby project headed by Linus Torvalds over 20 years ago. Since then the Linux operating system is used everywhere from enterprise level cloud servers to common household devices like your phone or a media streaming device. In this lab, you will be exposed to 2 variants of Linux: Kali Linux and Raspbian. You will learn how to install these variants of Linux as well as be asked to compare and contrast the two at the end of the lab.

Part 1: Setting up your pi.

To assemble your Raspberry Pi 3 Kit, refer to the directions below. For additional help, please refer to this step-by-step instruction video (https://www.youtube.com/watch?v=tK-w-wDvRTg).

Your Raspberry Pi 3 kit includes:

A 6' Axis HDMI to HDMI cable, a premium case for Raspberry Pi 3 (7" Touchscreen), a power cord, a 16GB MicroSD card (pre-installed with NOOBS raspbian operating system), a Raspberry Pi 3 Model B, a Raspberry Pi 7" Touchscreen Display, Raspberry Pi Heat Sink Kit, a DSI ribbon cable, 4 colored ribbon cables, and 2 bags with 4 screws each.

**Important Materials Needed Not Included in RPi3 Kit: **

- Phillips head screwdriver
- Blank microSD card
- SD card reader (or a laptop with an SD card reader)
- Desktop Keyboard
- Mouse (recommended)

Before installing the physical Pi on the touchscreen, Follow the steps on the Raspberry Pi website for downloading Raspbian on an SD card for your specific machine here. Choose Raspbian Operating System (not NOOBS), and then under "RASPBIAN STRETCH WITH DESKTOP", click "Download ZIP". Extract the image file, then write it on your own, blank microSD card. To write the file on your SD card, download the program Etcher here. Once Etcher is open and running, it will prompt you to select a file and a drive. For the file, select the *"2017-11-29 Raspbian Stretch Zip File"*, and for drive select your SD card. Once Etcher is

finished writing the Raspbian image on the SD card it will say “Flash Complete”. Once this occurs, eject your microSD card from your card reader, and insert it into the microSD port on the underside of the Raspberry Pi 3 Model B.

Next, Remove the Raspberry Pi 7” Touchscreen Display from the packaging and place it screen side down, and with the microUSB port on the adaptor board facing towards you. Insert the DSI Ribbon Cable into the jack on the left-hand side (blue side down) by pulling out the black lock on the jack, inserting the ribbon cable in, and closing the black part back to secure the cable.

Take your Raspberry Pi 3 with your Kali Linux microSD card inside it (HDMI port facing towards you), and place it on top of the 4 pegs on the touch screen’s adaptor board so they line up with the 4 holes in the Raspberry Pi. Insert the 4 longer thinner screws into the holes in the Raspberry Pi and tighten them using a screwdriver to secure the Pi on the Touchscreen Display. Finally, bend the ribbon cable around and insert it in the upwards facing jack on the left-hand side of the Pi.

Take your Raspberry Pi as it stands now and turn it around so the ribbon cable you connected in Step 1 is now on your right-hand side. Take one of the 4 colored cables and insert one end onto the most bottom right upward facing peg on the Raspberry Pi 3. Take the other end of the cable and insert it onto the rightmost peg facing towards you on the adaptor board.

Take another ribbon cable and insert one end onto the 3rd peg from the right on the bottom row of upward facing pegs, and the other end on the leftmost peg facing towards you on the adaptor board.

Next take the black plastic case for your touchscreen display and insert your device into the case so that all of the inputs line up correctly. Insert the 4 shorter wider screws into the 4 holes on the case that line up with the 4 holes on the back of the touchscreen display and secure the case onto the device.

Finally, plug in your keyboard and mouse into the USB ports, and your power adaptor into the micro USB port on your Raspberry Pi 3 to power up your device.

Part 2: Raspbian

When you turn the Pi on, the screen will turn on and off while Raspbian OS is installing. If prompted with a log in, enter: **pi** as the username, and **raspberry** as the password. You should be shown a desktop view with Raspbian fully installed. Explore some of the functionalities of Raspbian and research why people use this specific distribution of Linux.

Perform 3 actions in Raspbian and provide screenshots:

1. Open and explore the Browser
2. Open and perform a function in the command line
3. Open an application or a game

Part 3: Installing the Virtual Keyboard

If you do not have access to a USB keyboard to use during these labs, a virtual keyboard can be installed and used on your pi by using a few simple commands in the terminal.

First, open up the command terminal in Raspbian to begin the installing the keyboard. Type the following commands to run the keyboard install:

sudo apt-get update
sudo apt-get upgrade
sudo apt-get install matchbox-keyboard
sudo reboot

Your pi should now be restarting. Once it is booted up again, to go the main menu (the Raspberry icon in the upper left-hand corner) and select “accessories,” and then “keyboard” from the drop down menu. This should activate the virtual keyboard you just installed.

Additional instructions on how to install operating system images and how to write image files on SD cards can be found here:
https://www.raspberrypi.org/documentation/installation/installing-images/

Part 4: Explore the Pi and Command Line

With the command terminal still open, start testing out some of the basic command line commands to explore the Pi. Use the commands cd, ls, and pwd to accomplish the following respectively: change directory, list the contents of the current directory, and see which directory you are in. You can use the “cd ..” command to go up a directory, and “cd [name of directory]” to change to a given directory. Use the cat command to read text files.

Linux Lab 1 Survey

Answer the following survey questions on the following scale:

Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
1	**2**	**3**	**4**	**5**

1. The instructions were clear and easy to follow, and sufficient learning resources were provided

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

2. I received a high quality of classroom support (from professors and TAs)

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

3. My knowledge and understanding of the Raspbian operating system improved after completing this lab

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

4. When I encountered problems during the lab, I felt I had sufficient resources and support to work through them

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

5. Completing this lab enhanced hardware skills and operating system knowledge that I see as useful for a potential career in security

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

6. I can use information learned in this lab in a potential job scenario

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

7. I think future IST students would benefit from learning how to set up and use a personal Raspberry Pi kit

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

8. I see value in participating in further labs of increased difficulty using my pi kit to further enhance my understanding and skills of the Linux command line, programming, networking, and other areas

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

9. I feel comfortable enough with my Raspberry Pi to continue to learn and improve my skills outside of the classroom without classroom support

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

10. I feel confident that I would be able to install a new OS onto a Raspberry pi without instruction.

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

11. I was able to get comfortable with the command line commands, and feel like I can navigate to any directory I need to.

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

12. Any other general comments about the lab or suggestions for improvement?

Lab 2: Creating Users and Assigning File Permissions

Scenario:

Congratulations! You have just accepted your first job as a data security analyst for Hills Technology Company. Unfortunately, Hills does not currently have a very sound data management system, and they are looking to organize and secure their data better. Your first project has been assigned, and the job is to organize and secure the company's data into 3 directories with different permissions on each so only authorized users may view and edit certain data.

Your Task:

- Create 3 user accounts on the Linux command line (Clerk, Manager, and Executive), in the system, and set passwords for each

- Create 3 directories for company data (SensitiveData, SalesData, and PriceData). Then put them into a larger directory called Lab 3

- Set the directory permissions so that the Executive Account has read/write/execute permission to SensitiveData and SalesData, but only read/execute for PriceData. The Manager Account has read/write/execute permission to the Sales Data and Price Data, but no permission for SensitiveData. Finally, the Clerk only has read permission to the PriceData.

(Note: Read Permission allows the user to read the contents of the directory, but nothing more. Write permission gives the user access to edit the contents of the directory, and execute permission allows the user to execute the contents of the directory)

Step 1:

Open the command line on your Raspberry Pi to get started.

First, you must install an Access Control List to…? By inputting the following commands:

"sudo apt update"
"sudo apt Install acl"

Create 3 user accounts using the "useradd" command: Clerk, Manager, and Executive, and then assign passwords for each user using the "passwd" command. Make the password for Clerk, "Clerk", Manager, "Manager", and Executive, "Executive".

(Note: To return to the default user at any time, input the command "su pi", and enter "raspberry" for the prompted password. The "su" command can be used to switch between users)

Step 2:

Create 3 groups using the 'groupadd' command, and name the groups "Cashiers", "Managers", and "Executives".

Next, using the 'userMod' command, add each of the 3 users you created to the appropriate group.

Step 3:

Next, you must give permissions for each directory. You can do this in several ways, however for the purpose of this exercise you will use the 'SetFacl' command.

For example, to give the Cashiers read and execute permission for PriceData, imput the following command:

Sudo setfacl -a -G Cashiers: rx -R/PriceData

(Note: r = read permission, w = write permission, x = execute permission)

Finally, change the directory permissions using the reference above to complete the following tasks (For each of the following, provide a brief explanation of what you have done as well as the Linux commands you inputted.)

1. Give the Executive permission to Read, Write and Execute within the SensitiveData directory, and no one else any permission.

2. Give the Executive and the Manager permission to Read, Write, and Execute within the SalesData directory, and no one else any permission.

3. Give the Manager permission to read, write, and execute within the PriceData, the Executive read and execute permission, and the Clerk only permission to read it.

Linux Lab 2 Survey

Answer the following survey questions on the following scale:

Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
1	**2**	**3**	**4**	**5**

1. The instructions were clear and easy to follow, and sufficient learning resources were provided

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

2. I received a high quality of classroom support (from professors and TAs)

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

3. My knowledge and understanding of Linux file and directory permissions improved after completing this lab

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

4. When I encountered problems during this lab, I felt I had sufficient resources and support to work through them

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

5. Completing this lab enhanced programming skills and Linux OS knowledge that I see as useful for a potential career in security

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

6. Completing this lab helped me to better understand how Linux programming can be applied in a real world scenario

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

7. I think future IST students would benefit from learning file permission skills in Linux

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

8. The Linux Commands practiced in this lab challenged my programming skills and knowledge

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

9. I feel comfortable enough with the Linux OS after completing this lab to continue to enhance my skills and perform more advanced commands in Linux without classroom support

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

9. I feel comfortable enough with the Linux OS after completing this lab to continue to enhance my skills and perform more advanced commands in Linux without classroom support

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

10. What is the point of ACLs?

11. What does the sudo command do?

12. How would you create a user group called “Students”?

13. Any other general comments about the lab or suggestions for improvement?

Lab 3: System Administrator Commands and Tasks

There are many different commands that will be used regularly by any Linux user, and are necessary to be understood before entering the workforce. The following lab will cover some of the most important commands, and provide some resources to continue diving deeper into Linux command line tools.

Section 1: Writing to Files, $PATH variables

Part 1: Creating a file (vi)

To start off, you will be writing some code that we will learn how to run from anywhere on the device. To start, we will write a C++ Hello World programming using the vim command line text editor. Vim comes preinstalled on all linux distributions, and is the chosen editor for many Linux users since it offers a wide range of functionality. Type the command “vi hello.c”, which will put you in an insert prompt. Type “i” and the bottom left should display “**-- INSERT --**” (when tested, this is outside the view on the Pi, but will be there on other devices), then type the following.

```
#include <stdio.h>
```

```
int main() {
        printf(“Hello, world!\n”);
        return 0;
}
```

To exit out of the the vim editor, press the escape key then type “ZZ” (hold shift and press z twice). Type ls to confirm that the file was created.

Part 2: Adding the executable to the $PATH (gcc, chmod, echo, export)

With the file created, it is now time to compile the code. Since this is C++ code, the GCC compiler will be used. Type the follow command “gcc -o hello hello.c” to compile the code into an executable called “hello”. Type “chmod a+x hello” then “./hello” and you should see the output “Hello, world!”.

With the code compiled, it is time to export the code to the PATH. PATH variables are file paths that lead to folders containing binaries and executables that are run command line tools like cd, ls, python, and others you may install. Type “echo $PATH” to see all the different PATH variable. You can also type “printenv” to view all of the environment variables on the machine. Now that you have your executable in hello, type the following command: “export PATH=<*path to directory with hello*>:$PATH. You should now be able to type “hello” from any location in the file system and the code should run.

Section 2: Exploring files

- **Uptime & W**

In this section we will be exploring other command commands you may find yourself using, starting with “uptime”. Type “uptime” to see how long the machine has been running. Next, use the “w” command to get more detail on all the processes being run on the machine. Both of these are good tools to use when initially checking out a machine.

- **Ls, ls -l, ls -ltr, find**

Next, we will be exploring some navigation commands. While you have used the “ls” command many times, there are some flags you may not have used. Try “ls -l” to get more detail on the files in the directory, and use “ls -ltr” to sort them by last modified time. To explore some more “ls” commands, check out this link. While “ls” can help explore the directory you are currently in, you can use the “find” command to explore the entire device. The find command works with the following format “find <*base search directory*> -name <*name to search for*>”. Type “find / -name cd” to find where the “cd” executable is stored. This is where the $PATH variable looks every time you use the “cd” command. For more uses of the find command, check out this link. (if the find takes too long, use ctrl+c to exit the process).

- **Less, grep**

Looking for files is nice but sometimes you want to look inside files. There are many different ways to accomplish this in Linux. Let’s look at the passwd file. The best way to read a file is using the “less” command. Use the find command to find the location of the “passwd” file, the use “less <*path to file*>” to view the contents of the file. When using less you can use the

arrow keys to go up and down in the file, the “q” key to exit out of the view. If you want to search the file for a certain word, use the “grep” command. Type the line “grep root /etc/passwd”. This will only print the line containing the searched string.

Section 3: Handling Processes

- **top**

Most Sys Admins are going to need to have a strong grasp of what tasks are running on the machine and how to manage them. To get an idea of what processes are running on the machine, use the “top” function. You should see an output of the processes being run and lots of important information. This is often the first step to debugging a system.

- **Bash, kill**

With the “top” command understood it is important to also understand the “kill” command. To start, type the command “`for i in 1 2; do while : ; do : ; done & done`” and run the “top” command. Look at the CPU% column, notice anything? Now that we have processes eating CPU, it is time to kill them. Find the PID (Process ID) of the processes created from the for statement, and use “kill <*PID*>” to force the command to exit. Use “top” one final time to confirm the change has been made.

- **Free, last, ps**

There are other commands that can be used to debug a system and get vital information on the machine. The “free” command gives an overview of the memory usage of the machine and the “last” command shows the last processes to run. The “ps” command is similar to the “top” command with some minor differences. Utilizing these commands is necessary for any SysAdmin and should be understood fully.

Linux Lab 3 Survey

Answer the following survey questions on the following scale:

Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
1	**2**	**3**	**4**	**5**

1. The instructions were clear and easy to follow, and sufficient learning resources were provided

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

2. I received a high quality of classroom support (from professors and TAs)

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

3. My knowledge and understanding of Linux file and directory permissions improved after completing this lab

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

4. When I encountered problems during this lab, I felt I had sufficient resources and support to work through them

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

5. Completing this lab enhanced programming skills and Linux OS knowledge that I see as useful for a potential career in security

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

6. Completing this lab helped me to better understand how Linux programming can be applied in a real world scenario

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

7. I think future IST students would benefit from learning file permission skills in Linux

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

8. The Linux Commands practiced in this lab challenged my programming skills and knowledge

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

9. I feel comfortable enough with the Linux OS after completing this lab to continue to enhance my skills and perform more advanced commands in Linux without classroom support

←--- 1 ---------- 2 ---------- 3 ---------- 4 ---------- 5 ---→

10. What problems did you have in this lab and how did you solve them?

11. What parts of this lab did you find interesting and what parts were not?

12. How did you feel about the format of the lab?

13. Any other general comments about the lab or suggestions for improvement?

Appendix B

Survey Responses

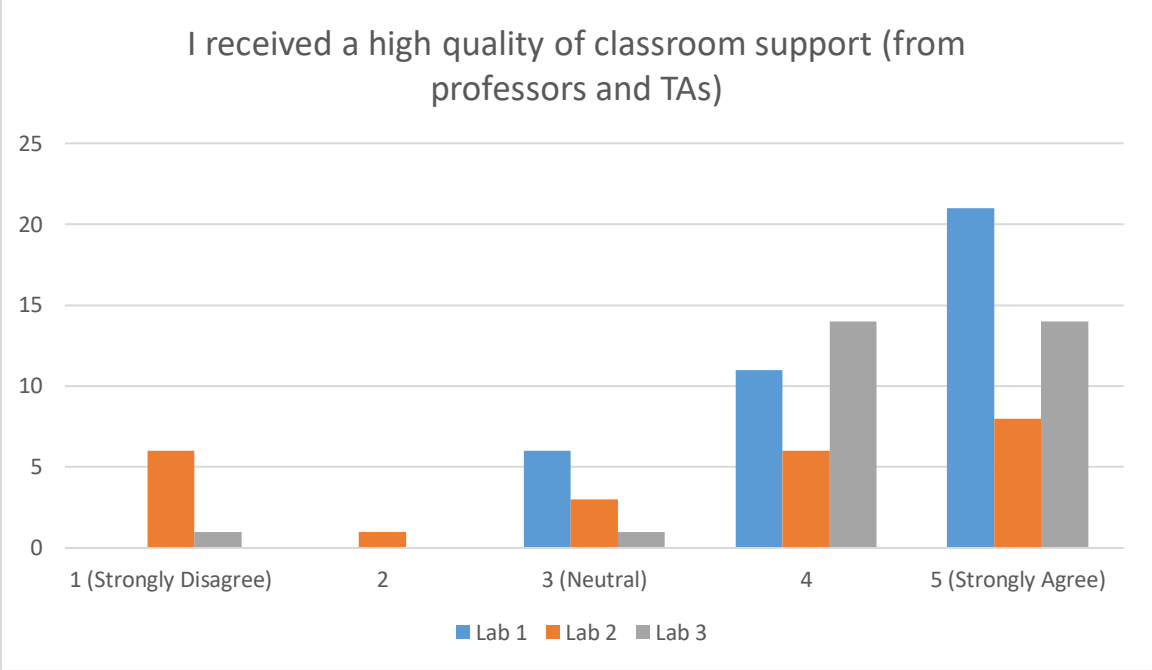

Figure 3

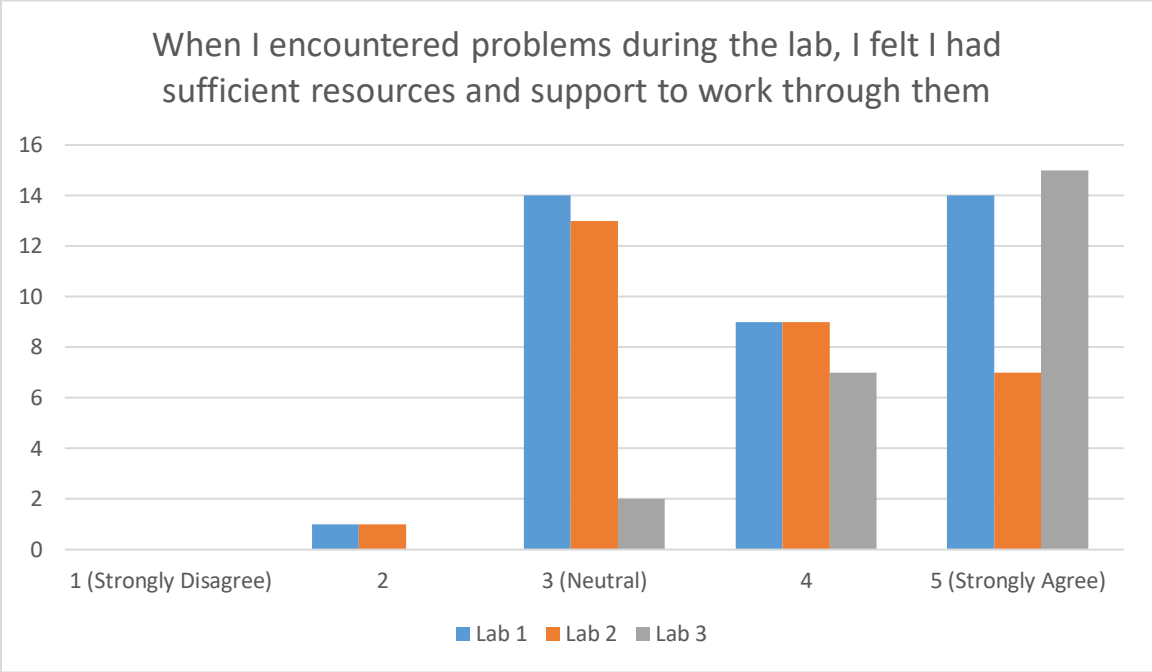

Figure 4

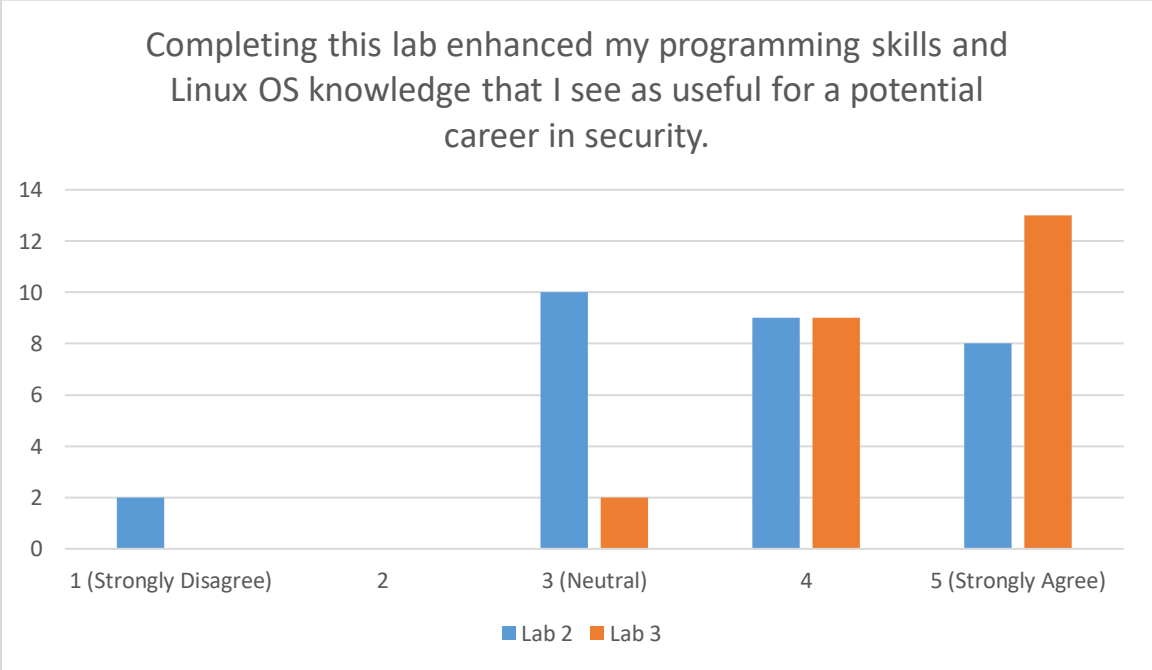

Figure 5

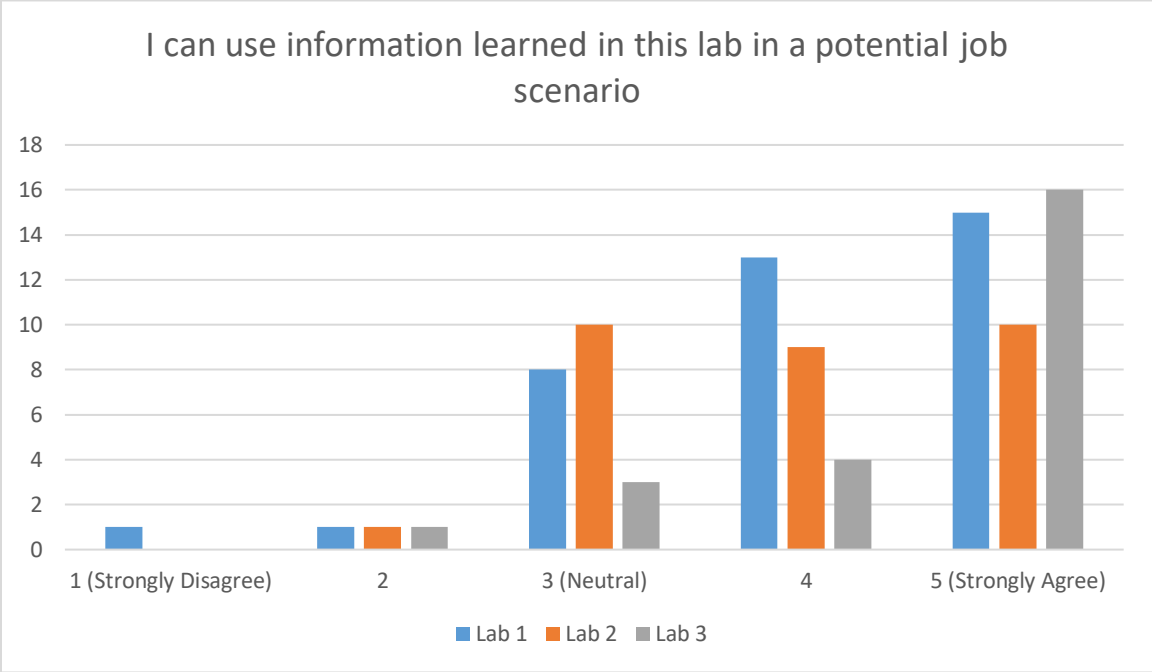

Figure 6

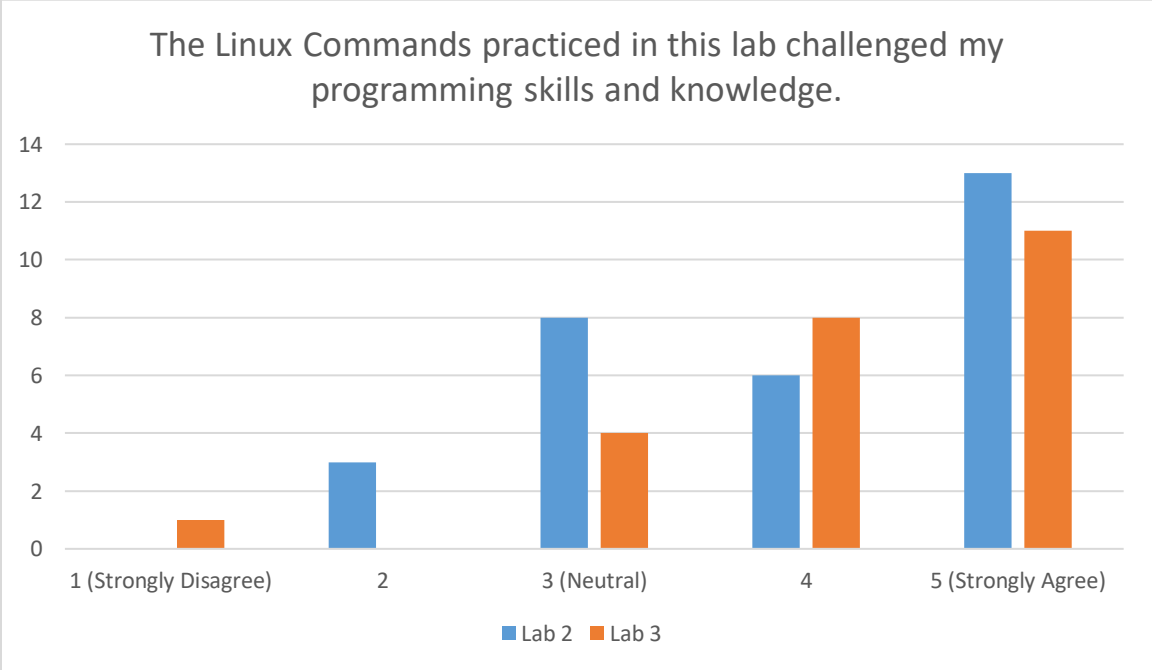

Figure 7

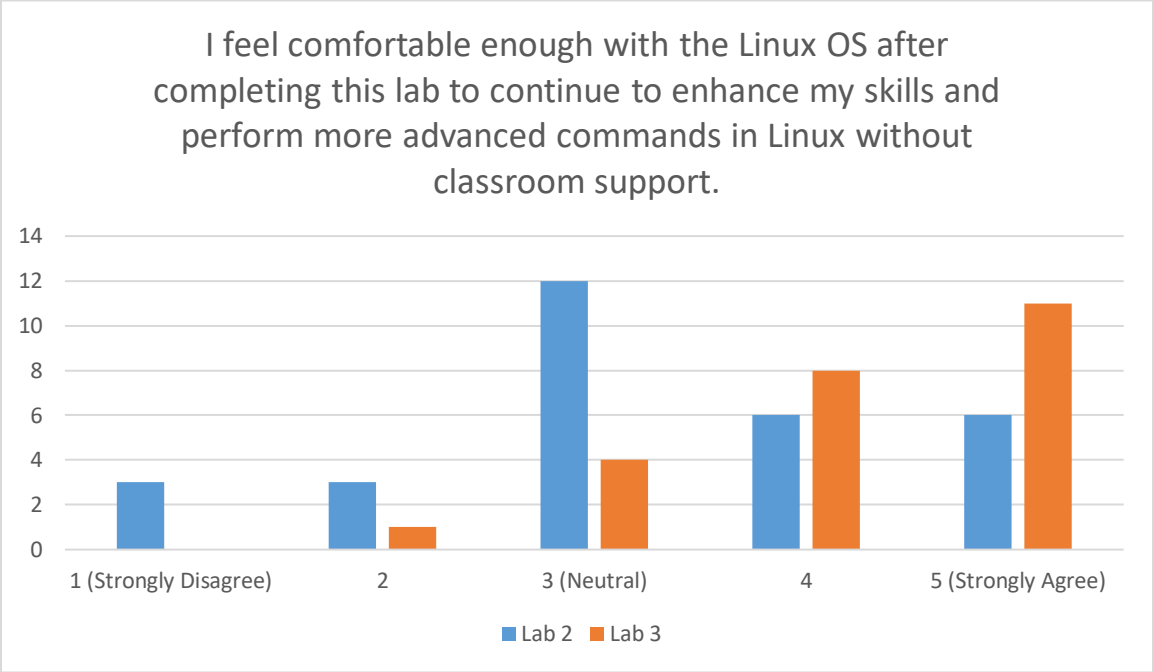

Figure 8

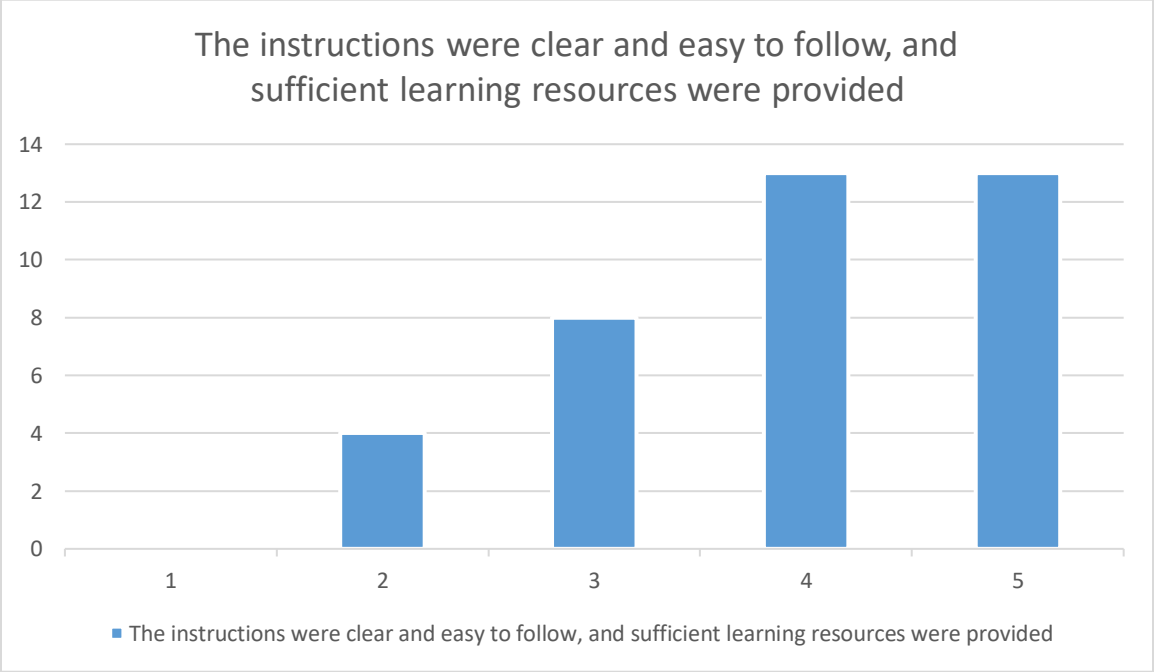

Figure 9

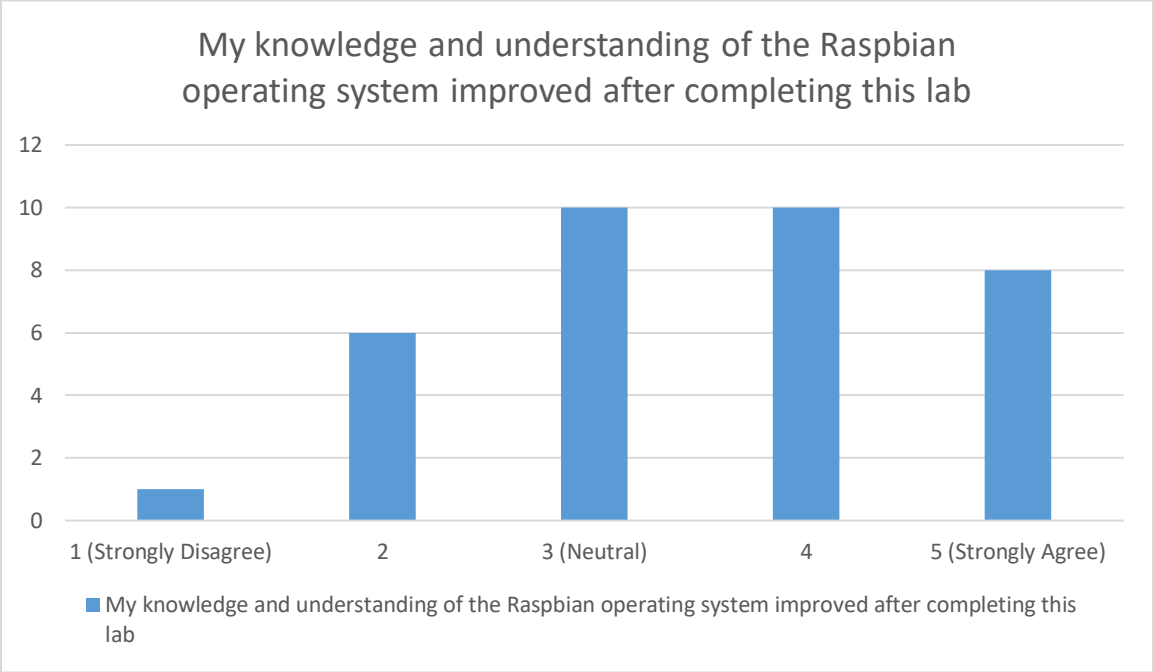

Figure 10

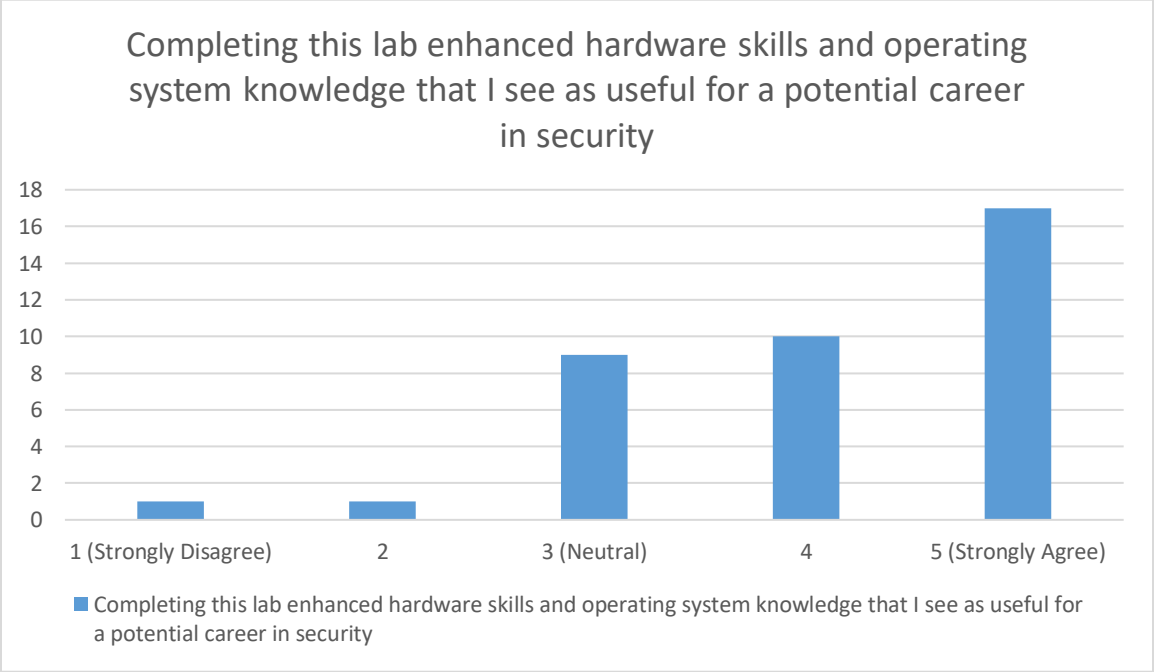

Figure 11

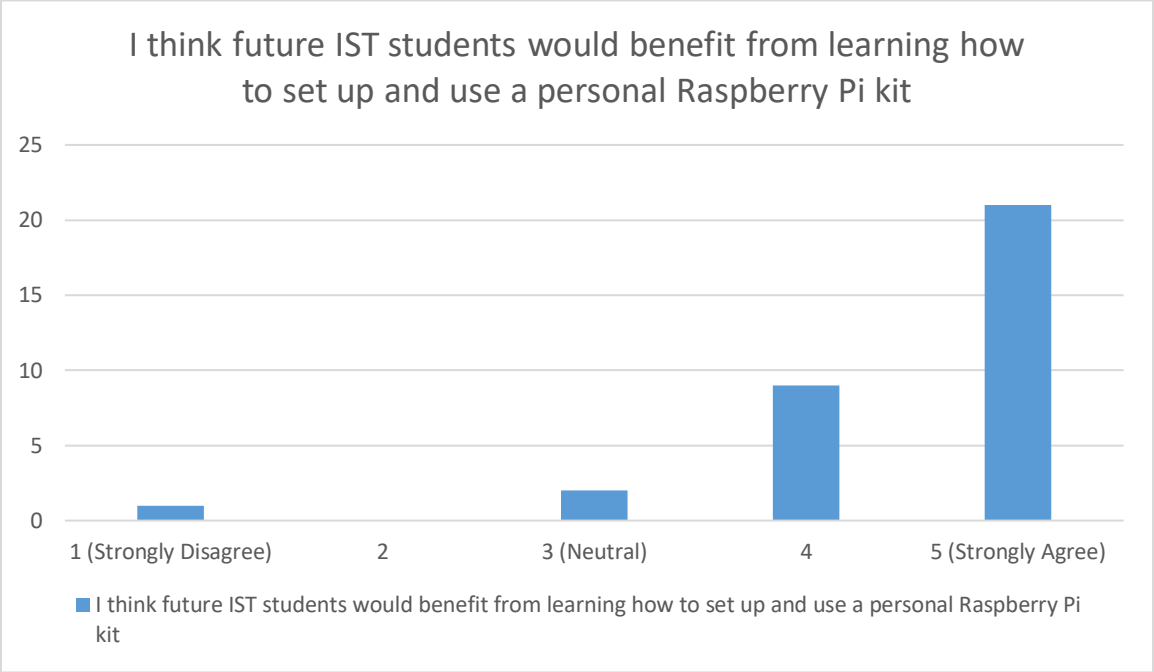

Figure 12

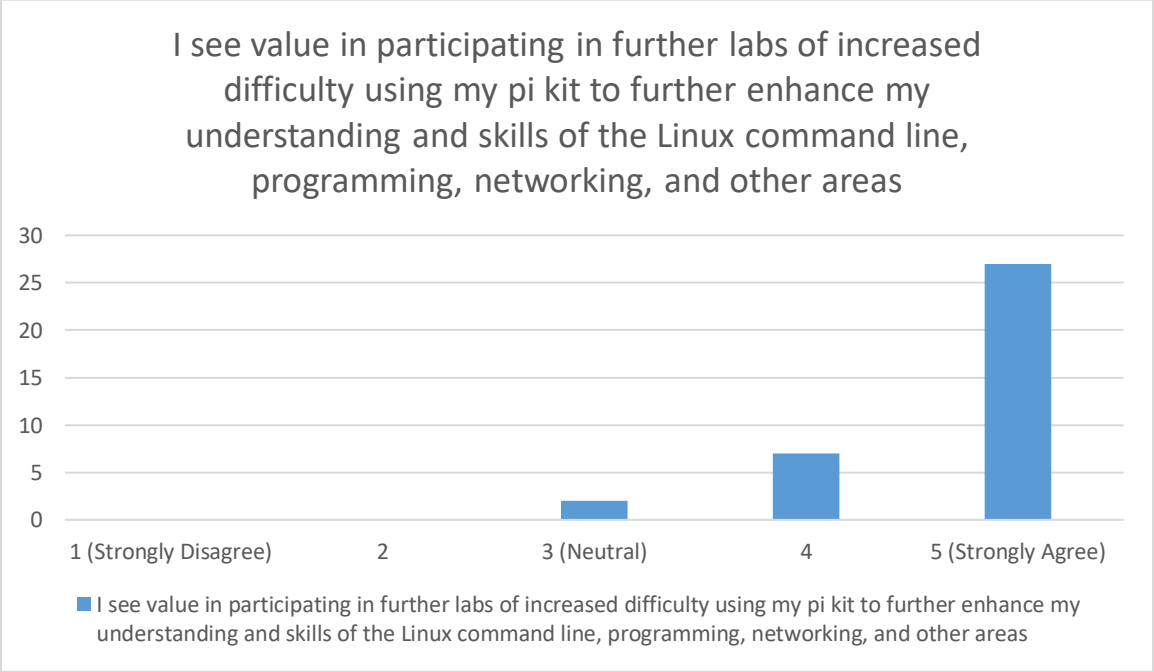

Figure 13

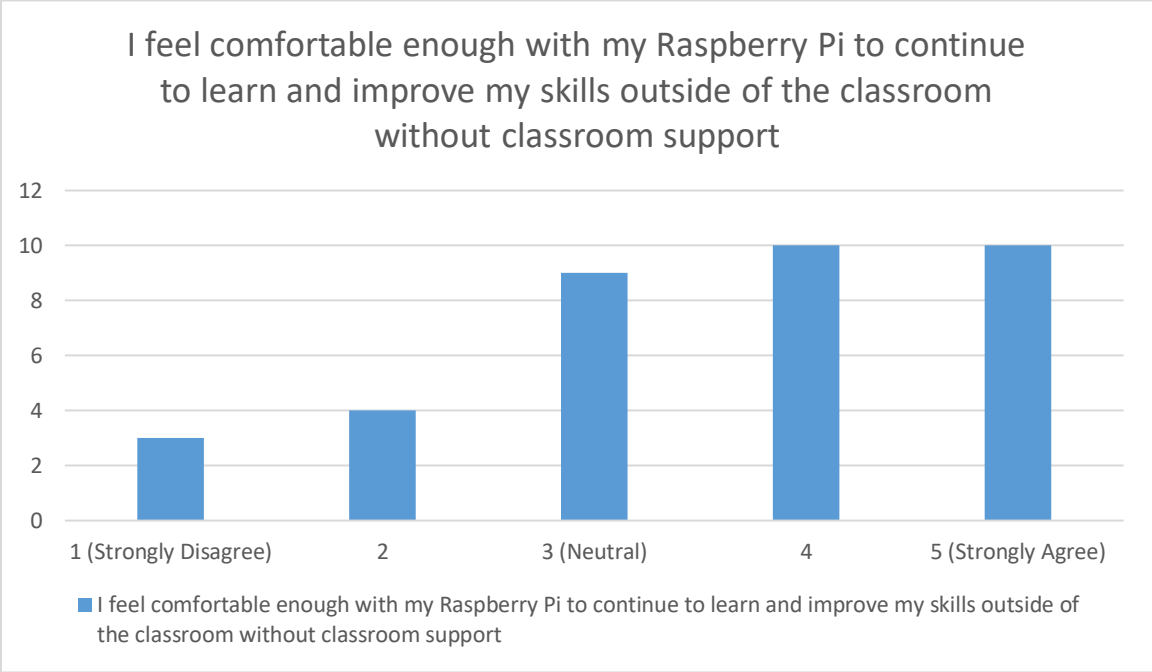

Figure 14

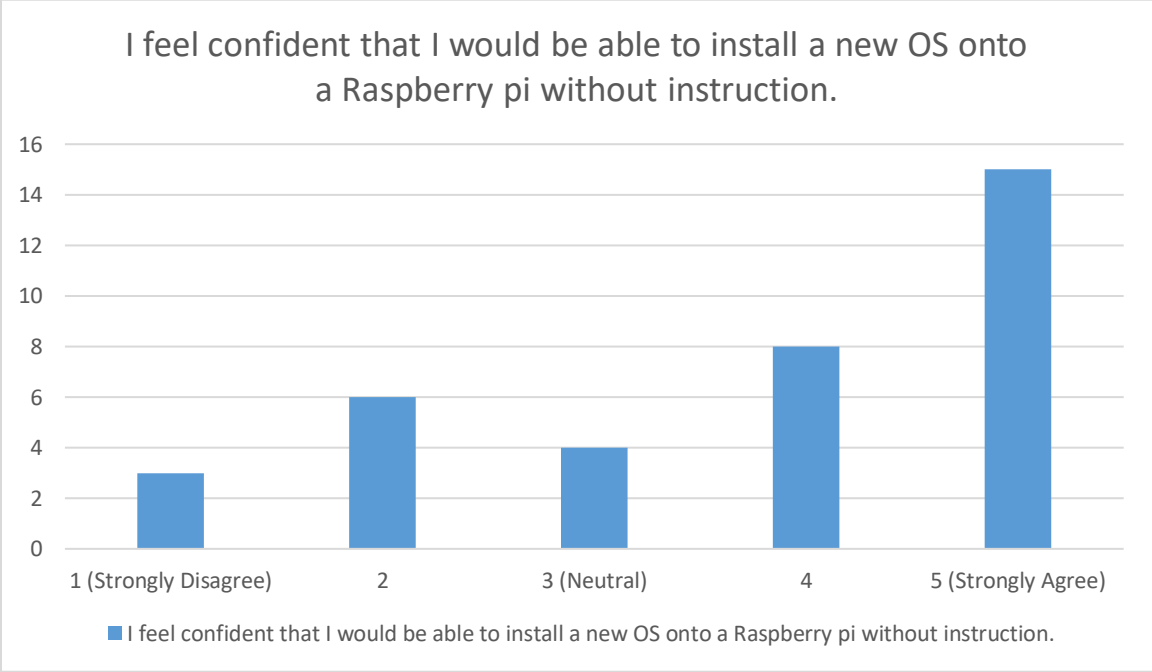

Figure 15

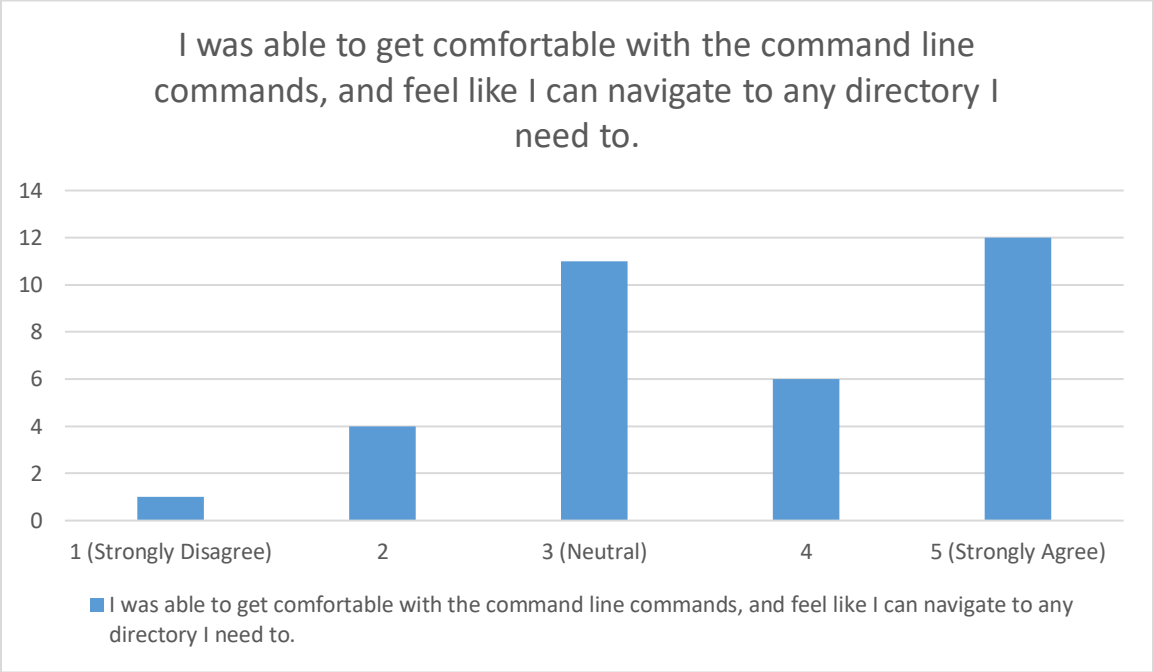

Figure 16

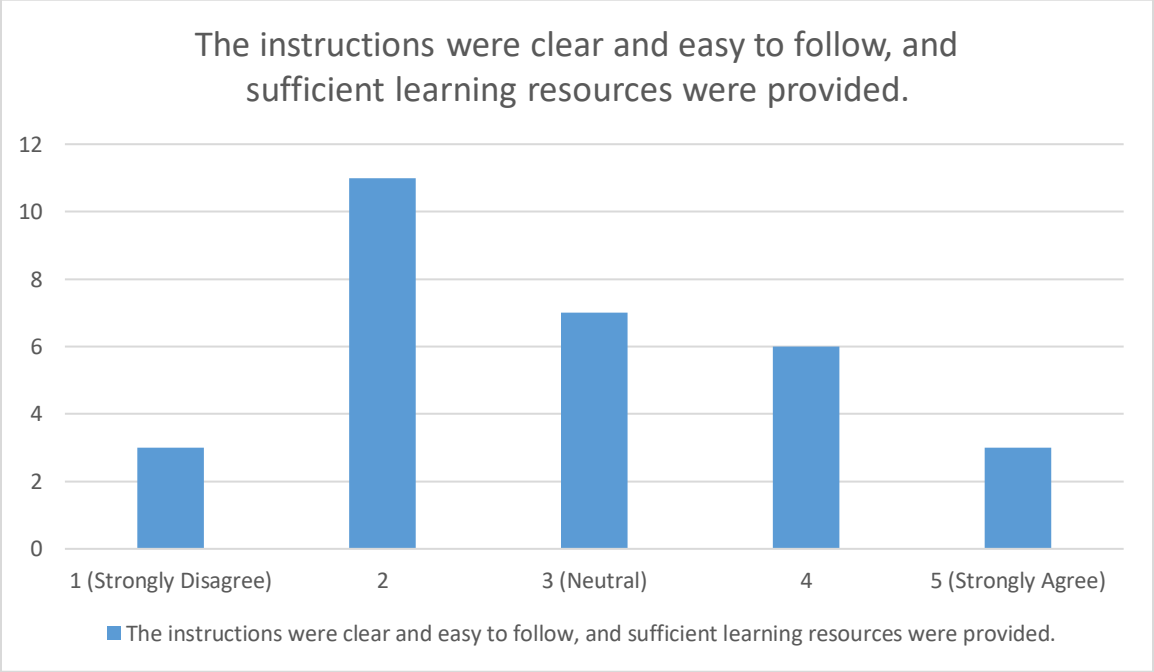

Figure 17

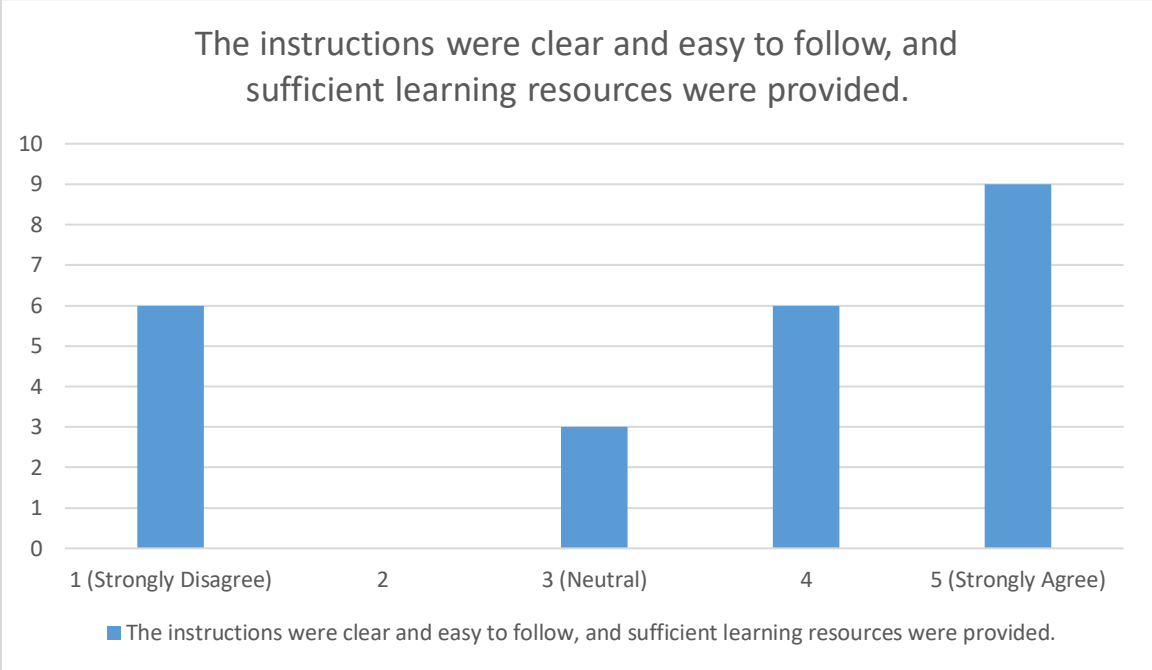
The instructions were clear and easy to follow, and sufficient learning resources were provided.
10
9
8
7
6
5
4
3
2
1
0
1 (Strongly Disagree)
2
3 (Neutral)
4
5 (Strongly Agree)
The instructions were clear and easy to follow, and sufficient learning resources were provided.

Figure 18

BIBLIOGRAPHY

Benchoff, B. (2016, February 29). Introducing the Raspberry Pi 3. Retrieved April 2, 2019, from https://hackaday.com/2016/02/28/introducing-the-raspberry-pi-3/

Buckley, I. (2018, November 14). 8 Ways a Raspberry Pi Can Help You Learn Online Security Skills. Retrieved April 2, 2019, from https://www.makeuseof.com/tag/raspberry-pi-learn-online-security/

Cheung, R. S., Cohen, J. P., Lo, H. Z., & Elia, F. (2011). Challenge Based Learning in Cybersecurity Education. *2011 International Conference on Security and Management*, 1–6.

Chothia, T., & Novakovic, C. (2015). An Offline Capture The Flag-Style Virtual Machine and an Assessment of its Value for Cybersecurity Education. *USENIX Summit on Gaming, Games, and Gamification in Security Education*.

F. B. Schneider, "Cybersecurity Education in Universities," in IEEE Security & Privacy, vol. 11, no. 4, pp. 3-4, July-Aug. 2013.
doi: 10.1109/MSP.2013.84

Impagliazzo, J., Dark, M., Cassel, L. N., McGettrick, A., & Hawthorne, E. K. (2014). *Toward curricular guidelines for cybersecurity*. (0), 81–82.
https://doi.org/10.1145/2538862.2538990

Kelty, C. M. (2018, January 26). The Morris Worm. Retrieved April 2, 2019, from https://limn.it/articles/the-morris-worm/

McGettrick, A., Cassel, L. N., Dark, M., Hawthorne, E. K., & Impagliazzo, J. (2014). Toward curricular guidelines for cybersecurity. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*. Retrieved March 31, 2019, from https://dl.acm.org

NICE Cybersecurity Workforce Framework. (n.d.). Retrieved April 2, 2019, from https://niccs.us-cert.gov/workforce-development/cyber-security-workforce-framework#wcm-survey-target-id

Piltch, A. (2019, March 01). Raspberry Pi Celebrates 25 Millionth Sale as 7th Anniversary Arrives. Retrieved April 2, 2019, from https://www.tomshardware.com/news/raspberry-pi-25-million-sold,38724.html

SentinelOne. (2019, March 21). The History of Cyber Security - Everything You Ever Wanted to Know. Retrieved from https://www.sentinelone.com/blog/history-of-cyber-security/

Toregas, C., Hoffman, L. J., & Burley, D. L. (2012, March). Holistically Building the Cybersecurity Workforce. *IEEE Security & Privacy*. doi:10.1109/MSP.2011.181

Upton, E. (2015, May 21). How The Raspberry Pi Sparked A Maker Revolution [Interview by M. Nuñez]. *Popular Science*. Retrieved April 3, 2019, from https://www.popsci.com/how-raspberry-pi-sparked-maker-revolution.

W. A. Conklin, R. E. Cline and T. Roosa, "Re-engineering Cybersecurity Education in the US: An Analysis of the Critical Factors," 2014 47th Hawaii International Conference on System Sciences, Waikoloa, HI, 2014, pp. 2006-2014.

Wolff, J. (2018, November 14). How Do You Get Students to Think Like Criminals? *New York Times*. Retrieved from https://www.nytimes.com/2018/11/14/opinion/cybersecurity-education-skills.html

Academic Vita

Carson Brown

EDUCATION

Bachelor of Science in Information Sciences and Technology
Bachelor of Science in Security and Risk Analysis

WORK EXPERIENCE

Ernst & Young **New York, NY**
Financial Services Technical Advisory Intern *June 2018 – August 2018*

- Advised a large institution on their Anti Money Laundering practices
- Helped develop risk scoring model utilizing R and Credit Risk Scoring formulas to identify high risk customer accounts
- Identified necessary rules within Actimize needed for proper Transaction Monitoring

GE Power **Redmond, WA**
Grid IQ Software Engineer Digital Technology Leadership Program Intern *May 2017 – August 2017*

- Developed chaos monkey to test a distributed system based on Chaos Engineering ideals; identified 5 defects in the system
- Configured HTTPS and SSL for API access and internal machine communication
- Wrote a packer script to automatically generate a configured VM in 15 minutes, which previously took up to 4 days
- Participated on an Agile Development team to develop a large scale distributed system utilizing Docker, Kafka, and Zookeeper

Professional Development Committee Leader *May 2017 – August 2017*

- Organized pitchout presentations for interns nationwide
- Organized a roundtable discussion on cyber security for all DTLP interns

ACTIVITES

Pennsylvania State University **State College, PA**

College of Information Sciences & Technology Student Government *April 2017 – Present*
Executive President

- Elected President by student body
- Involved in developing new ways to handle a quickly growing student body
- Developed new communication strategies for student clubs & orgs as well as the general student body

Executive Vice President

- Elected Vice President by student body
- Responsible for a committee comprised of the presidents of all IST clubs
- Meet regularly with the dean and other college officials to provide feedback from students as a representative of the student body

Nittany Data Labs *January 2017 – Present*

- Participated in a semester-long data science boot camp to educate members on data science concepts and tools
- Traveled to San Francisco to meet with multiple different companies, go to meetups, and attend a conference on disruptive technology
- Mentored a group of students coming out of our training program on developing a cryptocurrecy trading bot
- Involved in helping design training program for a group around 150 students
- Involved in corporate projects improving supply chain of major companies through data science

THON OPP *September 2017 – Present*

- Part of the committee in charge of managing operations at THON and THON related events