Teaching Web-Attacks on a Raspberry Pi Cyber Range

Sang Keun Oh
sangkeun.oh.mil@mail.mil
United States Military Academy
West Point, NY

Nathaniel Stickney
nathaniel.stickney@westpoint.edu
United States Military Academy
West Point, NY

Daniel Hawthorne
daniel.hawthorne@westpoint.edu
United States Military Academy
West Point, NY

Suzanne J. Matthews*
suzanne.matthews@westpoint.edu
United States Military Academy
West Point, NY

ABSTRACT

Cyber ranges are an important tool for teaching cyber security techniques. However, setting up a cyber range for classroom use can be costly. Prior work on lowering the cost of cyber ranges focuses on open source solutions and virtual machines. Yet, these solutions do not reduce the cost of physical components – namely, the underlying hardware used to build the range. In this paper, we describe a prototype cyber range built out of Raspberry Pis, a type of inexpensive single board computer. To illustrate the functionality of the range, we use Docker and Docker Swarm to deploy a vulnerable web server across four Raspberry Pi nodes and assess it in an undergraduate classroom. Our cyber range costs under $250.00 to build and consumes less than 25 Watts of power. We open-source our materials and provide pre-built Docker images on Docker Hub to enable others to use our work. Our results suggest that cyber ranges built using Raspberry Pi clusters can lower cost and enhance cyber security education.

CCS CONCEPTS

• **Applied computing** → **Education**; • **Security and privacy** → *Vulnerability management*; • **Computer systems organization** → *Embedded and cyber-physical systems.*

KEYWORDS

Raspberry Pi, Cyber range, Docker, Cyber Security, Education

ACM Reference Format:
Sang Keun Oh, Nathaniel Stickney, Daniel Hawthorne, and Suzanne J. Matthews. 2020. Teaching Web-Attacks on a Raspberry Pi Cyber Range. In *The 21st Annual Conference on Information Technology Education (SIGITE '20), October 7–9, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3368308.3415364

1 INTRODUCTION

Cyber ranges are valuable for cyber security education and are commonly used by universities, network security analysts, and hacking competitions. They offer students a sandboxed environment to safely explore and play with vulnerable applications and exploitation techniques. Several universities and organizations maintain persistent ranges [14] for cyber competitions and other educational activities. Well-known competitions making use of cyber ranges include iCTF [43, 44] and NCCDC [5, 38]. Cyber ranges help develop individual knowledge and teamwork in network security classes and hacking competitions [47]. More recently, cybersecurity education is extending into secondary school with summer camps such as GenCyber [15].

Classrooms commonly gain access to a cyber range by either connecting to an existing range or attempting to set up one of their own. For example, the U.S. Cyber Range (or Virginia Cyber Range) [35, 42] is used by students at James Madison University and Virginia Tech and is available for free for K-12 students in Virginia. Access to the range is available to U.S. classrooms outside of Virginia for a monthly fee. A course for 20 students needs to spend approximately $400.00 per month to access this range (see https://www.uscyberrange.org/pricing).

Setting up and maintaining cyber ranges can be very expensive; some universities invest millions into their ranges [42] [25]. For colleges and high schools looking to inject *some* cyber security concepts into their classrooms, the cost of building or gaining access to a cyber range can be prohibitive. Budget can also be an overriding concern for lower-income classrooms and individuals looking to build a range to learn cyber security concepts on their own [17].

In this paper, we discuss the development of a cyber range created with Raspberry Pi single board computers. Our use of the Raspberry Pi and Docker [9] helps lower the hardware and maintenance cost of the range. The prototype range described in this work consists of four Raspberry Pi 3 Model B+ (3B+) computers and costs approximately $234.00 to build. We have currently successfully tested and verified the functionality of thirteen modules of the Damn Vulnerable Web App (DVWA) [30] on our range. We also provide detailed instructions for replicating our work online [28, 29]. Lastly, we evaluate the effectiveness of the range to teach web attack techniques in an undergraduate cybersecurity course.

The rest of this paper is organized as follows. Section 2 describes related work. Section 3 gives an overview of the Raspberry Pi cyber range and currently available applications. Section 4 discusses the educational use of our range. We conclude in Section 5 with a discussion of perceived advantages and limitations of the Raspberry Pi cyber range.

*Corresponding author.

SIGITE '20, October 7–9, 2020, Virtual Event, USA
2020. ACM ISBN 978-1-4503-7045-5/20/10.
https://doi.org/10.1145/3368308.3415364

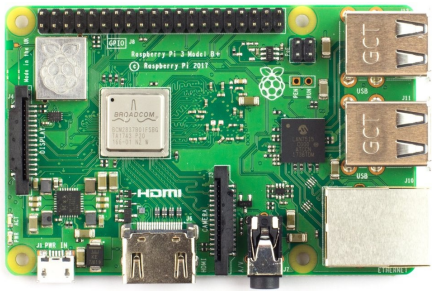

Figure 1: Raspberry Pi 3B+ single board computer

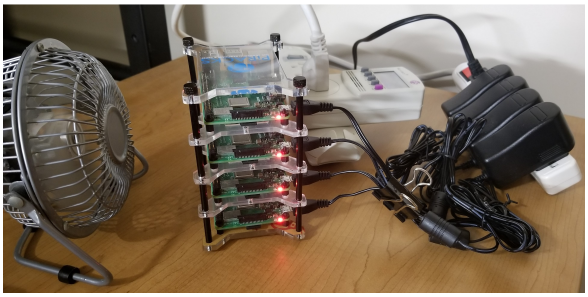

Figure 2: Raspberry Pi 3B+ Cyber Range Cluster

2 BACKGROUND & RELATED WORK

Local cyber ranges usually require considerable money, time and expertise. Most efforts in lowering the cost of locally-deployed cyber ranges attempt to defer the (often significant) time investment required for setup. A plethora of research and development exists around open-source software for cyber ranges; a sampling of the space includes CyberVAN [4], CyRIS [33], CyTrONE [3], KYPO [46] and CYRAN [19]. Each allows the creation of a set of virtual machines, typically hosted within a single environment (i.e. a workstation, server or cloud). However, virtual machines tend to be memory-intensive; adding additional VMs often requires incurring the cost of additional hardware. Most local ranges require separate hardware configured into a stand-alone network, to minimize the operational risk to campus networks [8].

A key novelty of our work is that we build our range out of inexpensive credit-card sized Raspberry Pi single board computers (SBCs). The Raspberry Pi (Figure 1) is widely used for computing education, including introducing students to C programming [48], ARM assembly [22], IoT [49] and parallel computing [27]. Due to the limited computing power of early models of the Raspberry Pi, several researchers also began creating Raspberry Pi clusters [7, 23, 40]. Researchers also use Raspberry Pi clusters to create inexpensive test-beds for various applications, including IDS [26], controls education [21], and software-defined networking [24].

Recently, researchers have begun to explore the Raspberry Pi for teaching cyber security in a hands-on manner. For example, Villa [45] employed Raspberry Pis to teach students cyber security concepts through hands-on labs. The ScriptKitty project [2] introduced middle school and early high school students to cyber security concepts using the Raspberry Pi. A recent SIGCSE workshop [6] also discussed how to use Raspberry Pis to integrate security concepts into high school and college-level curricula, including NSA GenCyber summer camps.

Another key novelty of our work is the use of Docker [9] containers to deploy cyber security exercises. We note that none of the aforementioned cyber security exercises on the Raspberry Pi utilized Docker. Unlike VMs, whose memory-requirements prohibit their use on Raspberry Pis, containers are "light-weight" environments that provide some level of resource isolation and do not require a full hypervisor. Other researchers have used Docker in conjunction with Raspberry Pis to create energy-efficient edge clouds [18, 32]. Some projects (such as Labtainers [39] and InCTF [36] use Docker to create easily deployable cyber security exercises and challenges that lower infrastructure requirements.

To the best of our knowledge, we are the first to propose the combined use of Docker and Raspberry Pis to create inexpensive cyber ranges. Our work supports prior work on the efficacy of Pi for teaching cyber security in a hands-on manner, and for creating educational testbeds. We believe the Raspberry Pi cyber range complements existing efforts and can be easily integrated into classrooms or summer cyber security workshops like GenCyber.

Table 1: Example Cluster Components

Description	Quantity	Cost/Unit	Total
Raspberry Pi 3B+	4	$35.00	$140.00
8 GB microSD cards	4	$4.99	$19.96
PiRacks Cluster Case	1	$29.95	$29.95
Power Supply	4	$8.99	$32.94
Fan	1	$10.99	$10.99
Grand Total			$233.84

3 CYBER RANGE OVERVIEW

Our prototype range consists of four Raspberry Pi 3B+ nodes running Raspbian Stretch (a Debian-based release) each with an 8 GB microSD card. The Raspberry Pi 3B+ [37] features a quad-core 1.4 GHz ARM Cortex A53 System-on-a-Chip (SoC) with 1 GB of RAM, integrated wireless and Bluetooth, and retails for $35.00 (Figure 1). While our test range is composed of 4 nodes, larger ranges can easily be built to support a greater number of simulated machines or active users. Next, while cyber ranges can theoretically be created out of any set of SBCs (such as Odroids or Tinkerboards), we focus specifically on the Raspberry Pi due to its widespread popularity and low cost.

Our prototype range is depicted in Figure 2. The cluster is assembled using the PiRacks [34] cluster case. A small fan assists with the cooling of the cluster. The test range utilizes separate power adapters for the cluster, but there are alternate cluster setups that use fewer wires and plugs; see [41] for one example. Table 1 shows the cost breakdown of the different components of the cluster. Optional materials for the range include a switch and Ethernet cables for wired connectivity. However, they are not necessary for our cluster, as the nodes communicate wirelessly with each other.

3.1 Container Management

We use Docker [9], a popular container platform that the Raspberry Pi natively supports, to facilitate the rapid deployment of test applications to our cyber range. Each container image hosts specific challenges or vulnerabilities for students to explore. Docker Swarm [11], a container orchestration system that automatically load-balances new containers on different cluster nodes, is used to deploy containers over the multiple nodes of the cluster. Docker Swarm is able to wirelessly manage the cluster nodes, simplifying setup. Users can initialize a Swarm Visualizer [12] to graphically display information about each node and container in the Swarm. The Swarm interface is especially useful for administrators or instructors trying to troubleshoot errors and manage the range.

Lastly, Docker Hub [10] expedites the setup of scenarios on the cyber range. Docker Hub is Docker's image repository service. Images pushed to Docker Hub can be downloaded by anyone, streamlining the process of starting one or more containers for individual or classroom use. Detailed instructions for setting up our prototype range are available in our GitHub repository [29]. The repository also includes a script that automatically installs Docker (Docker Swarm is included in that installation) and sets up the Swarm cluster. After the initial setup of the nodes and the Swarm, the instructor can start a range with a single command [29].

Our setup stands in stark contrast to the extensive setup required by some cyber ranges. CyberVan, for example, requires libvirt and Open vSwitch to be installed in addition to the CyberVAN management framework as a preliminary step [4]. Next, since CyberVan uses traditional VMs, a user needs to find a VM application online or build their own. Building a custom virtual machine an extremely time-consuming process that can take several hours to complete. In contrast, building a Docker image takes only minutes.

3.2 Sample Applicaton: DVWA

To assess the effectiveness of of the Rapberry Pi cyber range, we searched for existing Docker-based images that were deployable on the Raspberry Pi. We were unable to find any. Next, we looked for Docker-based cyber security exercises that made their Dockerfiles publicly available. Dockerfiles specify the set of instructions that Docker uses to build an image. While it is not uncommon for projects to share their Dockerfiles, we had limited success in identifying projects that publish cyber challenges or exercises that also publish their Dockerfiles. For example, while Labtainers [39] and InCTF [36] are Docker-based, they do not make the Dockerfiles of their exercises publicly available. Since the DockerHub images associated with these projects are based on the x86 architecture, they will not run as-is on the ARM-based Raspberry Pi.

Our search did reveal two cyber security exercise projects that did publish Dockerfiles. The first is the Damn Vulnerable Web App (DVWA) [30] which enables students to investigate vulnerabilities in MySQL and PHP websites. It is an intentionally vulnerability-ridden web application designed to be used as a teaching aid and a tool-testing environment without endangering production systems [30]. The second identified project was the Bali-based Cyber Jawara CTF competition [13]. Our Docker Hub repository [28] currently contains ARM-based images for both applications.

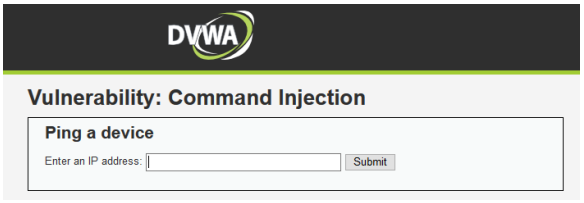

Figure 3: Student View on the DVWA for the Command Injection module

While we were able to successfully build and deploy the Cyber Jawara images on the Raspberry Pi cyber range, we could not fully verify the CTF's functionality, due to the solution set being written in a foreign language that was unavailable for translation through Google translate. As such, we focus on DVWA for the rest of this paper. Detailed instructions for setup are available [29]; the repository contains a script that deploys a visualizer app and a single DVWA container. The base image files for both applications are available [28] through DockerHub. The provided installation script automatically downloads the DVMA image and fully deploys it on the cluster in just 7 minutes.

Students connect their workstations to this WiFi network to access the range and browse to their assigned instance of DVWA, which is a website. Figures 3 depicts one sample DVWA module students can access. This module exposes students to command injection — and the necessity of input sanitization — and is just one of 13 modules that students can access through the DVWA containers in our prototype range.

While DVWA is the focus of this paper, any containerized application suitable for ARM architecture could be run on our range. It is sufficient to create a Dockerfile for the associated application, compile it using Docker, and deploy it to the range. Traditional cyber ranges require the same level of effort if the deployed applications are containerized, and more effort if range requires virtual machine images. Using the Pi range saves the institutional cost of maintaining persistent server hardware and the instructor overhead of having their exercises deployed on that hardware.

3.3 Curricula Standards Alignment

The Raspberry Pi Cyber Range with DVWA supports the Software Security ad other knowledge areas in CSEC2017 [20]. Using the range, students receive hands-on experience with topics including least privilege, minimizing trust, input validation, and exception handling. These topics also align with ABET student outcomes for computing programs [1], specifically evaluation of computing-based solutions and application of security principles and practices.

DVWA on the Pi cyber range can also be used to meet the objectives of the NSA GenCyber program [15]. With DVWA, students experience the importance of data hiding, least privilege, process isolation, and other cyber security first principles. It allows them to execute hands-on experiments to understand attack models and countermeasures for web applications, helping them to "Think Like an Adversary", one of the GenCyber cyber security concepts.

4 CLASSROOM USE AND ASSESSMENT

Prior to classroom use, the instructor sets up the range using instructions highlighted in our repository [29]. Given a set of Raspberry Pis running Raspbian, setup involves running our setup script on each Pi, initializing the Docker Swarm, and running the scenario startup script on the manager node. While many existing ranges require dedicated support personnel and extensive preparation for use, the Pi range can be set up by individual instructors, owing to this software simplicity. Our provided scenario script sets up multiple instances of DVWA, each hosted in a separate Docker container. Testing revealed that each Raspberry Pi 3B+ can host at least 3 separate instances of DVWA without performance issues.

4.1 Preliminary Testing

The prototype range measures 4.75" x 4.25" x 3" and weighs roughly 1.5 pounds. Using a KillAWatt [31], we estimate the range uses 10-11 Watts of power when idle, and 22-23 Watts during use. Setup is expedited by the use of our container image [28] provided on Docker Hub. On a university wireless connection (measured at 11 MBps download speed), it took less than 5 minutes to download and extract the DVWA image onto an individual Raspberry Pi. Deploying DVWA over all four nodes of our prototype cluster took 2 minutes. Furthermore, Docker Swarm makes it effortless to add additional Pis to the range.

We verified that the DVWA is operational by stepping through each of the fourteen modules and running them to completion. All but one module (CAPTCHA module) were verified working. The CAPTCHA module did not run due to an expired Google API key being embedded into the application. However, we believe this can be overcome if each student signs up with Google's API service to gain access to a new key.

4.2 Classroom Assessment

We assess the efficacy of the Raspberry Pi cyber range during a laboratory exercise given to five sections of a cyber security course at West Point. Sections at West Point are capped to 19 students. In the Test1 section, each student had access to their own container. In the Test2 section, students worked in pairs. Note that Test1 is a smaller section of more "advanced" students. All statistical analysis was conducted using the R package version 3.4.

We were fortunate that assessment was completed during in-person labs that took place prior to the COVID-19 crisis. All students learned the same set of web-attack principles during the in-person labs. Three sections learned the material through the traditional VM setup at West Point (control sections). The other two learned the material by using DVWA on the Raspberry Pi cyber range (test sections). During the exercise, students were unaware that that they were using the Pi range; at the end of the lab, the Pi range was revealed to the students.

The lab was scored out of 35 possible points. Table 2 shows the mean, median, and standard deviation for each section. Figure 4 shows the distribution of scores. Unsurprisingly, the advanced section (Test1) had the highest scores and lowest standard deviations.

To assess the significance of the differences of the means, we first performed a Bartlett test to see if the variances were equal. The extreme nature of the Test1 section caused the variances to be unequal. As a second check, we performed a Bartlett test on just the Test2 and control sections. The second check showed that the variances of the populations were not significantly different ($p = 0.093$). Since the data is normally distributed and the populations are of roughly equal size, we perform a Fisher's one-way ANOVA statistical test and Tukey's HSD to compare the means in the controls sections to Test2. Since the variances between Test1 and the control populations are unequal and the populations are of unequal size, we use a Welsh Two Sample t-test to perform pairwise comparison of the means between Test1 and the control sections.

Table 3 shows the final results. In all cases, students in the Test1 section performed significantly better on the lab activity than the control sections. Students in the Test2 section performed significantly better than students in the Ctrl1 section. However, there is no significant difference in the performance of students in the Test2

Table 2: Summary of Student Performance

Section	*N*	Mean	Median	StdDev
Test1	8	34.69	35.00	0.883
Test2	19	31.45	31.50	2.58
Ctrl1	16	27.88	28.50	4.64
Ctrl2	18	31.29	32.50	2.98
Ctrl3	17	32.18	32.00	3.72

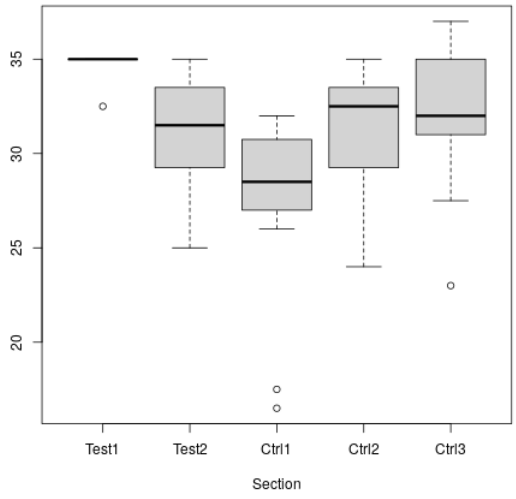

Figure 4: Distribution of Scores Across Sections

Table 3: Significance Testing

P-value	Ctrl1	Ctrl2	Ctrl3
Test1	$2.74e^{-5}$	0.0002	0.0164
Test2	0.0198	0.9991	0.9250

section compared to the Ctrl2 and Ctrl3 sections. These results suggest that students were able to learn web attack principles equally well (if not a little better) on the Raspberry Pi range compared to a traditional VM. We note however, that these results are preliminary, owing to the small sample size in the experiment. Further testing will help fully ascertain the efficacy of the Raspberry Pi cyber range.

4.3 Student and Faculty Perspectives

As a second point of data collection, students and faculty in the test sections were asked to share their experiences using the Raspberry Pi cluster. In accordance with institutional IRB policies, students were invited to take a voluntary survey about their experiences and were offered an opportunity to earn a nominal amount of extra credit. Only 7 of the 27 students (25%) opted to take the survey. The received survey responses were generally positive. Students were amazed by how small the Raspberry Pi cyber range was and were surprised that it was capable of running the exercise. *I didn't even know I used the Raspberry Pi Cyber Range until my teacher told me we used it* said one student. Three of the seven student respondents indicated that the things they liked most about the range was how inexpensive it was. Six of the seven students indicated that using Raspberry Pi Cyber range helped them better understand topics, and wished they had more exercises with the cyber range. All students expressed a desire to spend more time on the range.

Two of the authors were instructors for the cyber security course. Both instructors had prior experience with a traditional VMs for the laboratory exercise, which was traditionally conducted using a VMware vSphere cluster. When reflecting on their experiences using the Raspberry Pi range, the instructors were most struck by how little time it took to set up the Raspberry Pi cyber range (less than 15 minutes) compared to the traditional VM. As a form of resiliency testing, one instructor decided to "destroy" and rebuild the range several times; he was able to redeploy the range successfully each time without any issues. Furthermore, running the lab on the Pi range allowed us to conduct the lab without allocating resources of our vSphere cluster. Both instructors are interested in using the Raspberry Pi range again in future iterations of the course.

5 CONCLUSIONS

In this paper, we describe a Raspberry Pi cyber range and its use to teach basic web attack principles in a cyber security course. Our cyber range costs under $250.00 to build and consumes less than 25 Watts of power, significantly less than a typical range. To allow others to replicate our range, we open-source our materials [28, 29] and include simple instructions on how to build a Raspberry Pi cyber range and deploy the Damn Vulnerable Web App (DVWA).

To assess the efficacy of teaching web attacks using the Raspberry Pi cyber range, we assessed student performance on a laboratory exercise over several sections of a cyber security course. Our results suggest that students learned concepts equally well on the Raspberry Pi cyber range as they did on a traditional VM, if not better. Students also reported interest in spending more time on the Raspberry Pi range. Furthermore, instructors had a positive experience using the Raspberry Pi cyber range in their course.

There are several advantages of a Raspberry Pi-based cyber range over traditional cyber ranges, the largest being the inexpensiveness and ease of the setup. The Raspberry Pi's microSD card storage simplifies backup and recovery from mishaps. In the rare instances when nodes become corrupted, it suffices to re-image the microSD card. In the extreme case where microSD cards must be replaced, doing so is still a cost-effective option. Hard-drive replacement or reformatting is a much more involved process for traditional systems. All of these advantages make the Raspberry Pi cyber range an inexpensive and viable option for classroom use.

We acknowledge that there are limitations of the Raspberry Pi cyber range. Unlike VMs, containers use a shared kernel space, which means they are not always appropriate for creating sandbox environments. For memory-intensive applications that require a perfect sandboxed environment, a VM on a traditional x86 server is perhaps preferable. However, this may not matter for many applications typically found in an educational setting. Labtainers [39] is a prominent example of successfully using containers for cyber security education.

We also note that there is a distinct lack of ARM-based Docker containers available for cyber security education. The reason for this is two-fold. First, VMs are still the most common way to deploy cyber security exercises; the use of Docker for such exercises is still a relatively novel concept. Second, the use of the Raspberry Pi for cyber security exercises is also relatively novel. As a short term solution, instructors wishing to deploy separate cyber security exercises will either need to write their own custom Dockerfiles or modify existing ones. In the long term, cyber security exercise designers should release the Dockerfiles associated with their projects; doing so will expedite the porting of containers to the Raspberry Pi. Lastly, we note that container creation need only be a one-time cost; once a container for a particular exercise is successfully created, it can be uploaded to Docker Hub for use by anyone.

While our Raspberry Cyber range is still a prototype, our preliminary results are encouraging. We believe that our range will make it easy for organizations conducting cyber security camps and workshops (like NSA GenCyber) to incorporate Raspberry Pis into their curricula, or augment existing curricula that utilize Raspberry Pis. We anticipate that the low cost and ease of deployment of our cyber range will make it an asset for cyber education and make it possible for every classroom to have its own educational range.

Lastly, the technology around containers and the Raspberry Pi also continues to evolve and improve. The recently announced Raspberry Pi 4 can support up to 4 GB of RAM, making it possible to host more memory-intensive applications or a greater number of containers on each Pi. Innovations like gVisor [16] help improve the resource isolation of container-based applications. Future work will concentrate on exploring these technologies.

ACKNOWLEDGMENTS

Funding for this project is provided by U.S. Army Futures Command, CCDC Armaments and the DOD High Performance Computing Modernization Program (HPCMP). This paper was authored by employees of the U.S. Government. The views expressed in this paper are those of the authors and do not reflect the official policy or position of the Department of the Army, Department of Defense or the U.S. Government.

REFERENCES

[1] ABET. 2019. Criteria for Accrediting Computing Programs 2019 – 2020. https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-computing-programs-2019-2020/

[2] Ovidiu-Gabriel Baciu-Ureche, Carlie Sleeman, William C. Moody, and Suzanne J. Matthews. 2019. The Adventures of ScriptKitty: Using the Raspberry Pi to Teach Adolescents about Internet Safety. In *Proceedings of the 20th Annual SIG Conference on Information Technology Education (SIGITE'19)*. Association for Computing Machinery, New York, NY, USA, 118–123. https://doi.org/10.1145/3349266.3351399

[3] Razvan Beuran, Cuong Pham, Dat Tang, Ken-ichi Chinen, Yasuo Tan, and Yoichi Shinoda. 2017. Cytrone: An integrated cybersecurity training framework. In *Proceedings of the 3rd International Conference on Information Systems Security and Privacy (ICISSP 2017)*. SCITEPRESS–Science and Technology Publications, Japan, 156–166. https://doi.org/10.5220/0006206401570166

[4] R. Chadha, T. Bowen, C. J. Chiang, Y. M. Gottlieb, A. Poylisher, A. Sapello, C. Serban, S. Sugrim, G. Walther, L. M. Marvel, E. A. Newcomb, and J. Santos. 2016. CyberVAN: A Cyber Security Virtual Assured Network testbed. In *MILCOM 2016 - 2016 IEEE Military Communications Conference*. IEEE, Baltimore, MD, 1125–1130. https://doi.org/10.1109/MILCOM.2016.7795481

[5] Art Conklin. 2005. The Use of a Collegiate Cyber Defense Competition in Information Security Education. In *Proceedings of the 2Nd Annual Conference on Information Security Curriculum Development (InfoSecCD '05)*. ACM, New York, NY, USA, 16–18. https://doi.org/10.1145/1107622.1107627

[6] Andreea Cotoranu and Li-Chiou Chen. 2019. Using Raspberry Pi As a Platform for Teaching Cybersecurity Concepts. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*. ACM, New York, NY, USA, 1237–1237. https://doi.org/10.1145/3287324.3287534

[7] Simon J. Cox, James T. Cox, Richard P. Boardman, Steven J. Johnston, Mark Scott, and Neil S. O'Brien. 2014. Iridis-pi: a low-cost, compact demonstration cluster. *Cluster Computing* 17, 2 (01 Jun 2014), 349–358. https://doi.org/10.1007/s10586-013-0282-7

[8] Jon Davis and Shane Magrath. 2013. *A survey of cyber ranges and testbeds*. Technical Report DSTO-GD-0771. Defense Science and Technology Organisation Edinburgh (Australia) Cyber and Electronic Warfare Div.

[9] Docker Inc. 2019. Docker: Enterprise Application Container Platform. https://www.docker.com/

[10] Docker Inc. 2019. Docker Hub. https://hub.docker.com/

[11] Docker Inc. 2019. Swarm mode overview. https://docs.docker.com/engine/swarm/

[12] Docker Samples. 2019. Docker Swarm Visualizer. https://github.com/dockersamples/docker-swarm-visualizer

[13] farisv. 2018. Cyber Jawara 2018 Final - Attack & Defense CTF services environments based on Docker. https://github.com/farisv/CJ2018-Final-CTF

[14] Curtis Franklin. 2017. 7 University-Connected Cyber Ranges to Know Now. https://www.darkreading.com/cloud/7-university-connected-cyber-ranges-to-know-now/d/d-id/1331224

[15] GenCyber. 2019. 2019 GenCyber Call for Proposal. https://www.gen-cyber.com/proposals/rfp/gc-2019/)

[16] Google. 2019. google/gvisor. https://github.com/google/gvisor

[17] R. Guiler. 2018. Building a Cyber Training Range on a Budget. https://www.youtube.com/watch?v=NQaVqN1HFPs

[18] Wajdi Hajji and Fung Po Tso. 2016. Understanding the Performance of Low Power Raspberry Pi Cloud for Big Data. *Electronics* 5, 2 (2016). https://doi.org/10.3390/electronics5020029

[19] Bil Hallaq, Andrew Nicholson, Richard Smith, Leandros Maglaras, Helge Janicke, and Kevin Jones. 2018. CYRAN: a hybrid cyber range for testing security on ICS/SCADA systems. In *Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications*. IGI Global, UK, 622–637. https://doi.org/10.4018/978-1-5225-5634-3.ch033

[20] Joint Task Force on Cybersecurity Education. 2017. *ACM/IEEE/AIS SIGSEC/IFIP Cybersecurity Curricular Guideline*. Technical Report CSEC2017. ACM, IEEE, AIS, IFIP. 121 pages. https://cybered.hosting.acm.org/wp/

[21] Martin Kalúz, L'uboš Čirka, Richard Valo, and Miroslav Fikar. 2014. ArPi Lab: A Low-cost Remote Laboratory for Control Education. *IFAC Proceedings Volumes* 47, 3 (2014), 9057 – 9062. https://doi.org/10.3182/20140824-6-ZA-1003.00963 19th IFAC World Congress.

[22] Jalal Kawash, Andrew Kuipers, Leonard Manzara, and Robert Collier. 2016. Undergraduate Assembly Language Instruction Sweetened with the Raspberry Pi. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16)*. ACM, New York, NY, USA, 498–503. https://doi.org/10.1145/2839509.2844552

[23] Joshua Kiepert. 2013. *Creating a raspberry pi-based beowulf cluster*. Technical Report. Boise State University. 1–17 pages.

[24] H. Kim, J. Kim, and Y. Ko. 2014. Developing a cost-effective OpenFlow testbed for small-scale Software Defined Networking. In *16th International Conference on Advanced Communication Technology*. 758–761. https://doi.org/10.1109/ICACT.2014.6779064

[25] Alex Knisely. 2019. UA joins Ohio Cyber Range in $1.18M agreement. https://www.uakron.edu/im/news/ua-joins-ohio-cyber-range-in-1-18m-agreement/

[26] A. K. Kyaw, Yuzhu Chen, and J. Joseph. 2015. Pi-IDS: evaluation of open-source intrusion detection systems on Raspberry Pi 2. In *2015 Second International Conference on Information Security and Cyber Forensics (InfoSec)*. 165–170. https://doi.org/10.1109/InfoSec.2015.7435523

[27] Suzanne J. Matthews, Joel C. Adams, Richard A. Brown, and Elizabeth Shoop. 2018. Portable Parallel Computing with the Raspberry Pi. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education (SIGCSE '18)*. ACM, New York, NY, USA, 92–97. https://doi.org/10.1145/3159450.3159558

[28] Sang Oh. 2019. Docker Hub Cyber Range Image for Raspberry Pi. https://hub.docker.com/r/sko9370/rpi

[29] Sang Oh. 2019. sko9370/CyberRangePi. https://github.com/sko9370/CyberRangePi

[30] Opsxcq. 2018. opsxcq/docker-vulnerable-dvwa. https://github.com/opsxcq/docker-vulnerable-dvwa

[31] P3 International. 2019. Kill A Watt Meter - Electricity Usage Monitor. http://www.p3international.com/products/p4400.html

[32] C. Pahl, S. Helmer, L. Miori, J. Sanin, and B. Lee. 2016. A Container-Based Edge Cloud PaaS Architecture Based on Raspberry Pi Clusters. In *2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*. 117–124. https://doi.org/10.1109/W-FiCloud.2016.36

[33] Cuong Pham, Dat Tang, Ken-ichi Chinen, and Razvan Beuran. 2016. CyRIS: A Cyber Range Instantiation System for Facilitating Security Training. In *Proceedings of the Seventh Symposium on Information and Communication Technology (SoICT '16)*. ACM, New York, NY, USA, 251–258. https://doi.org/10.1145/3011077.3011087

[34] PiRacks. 2019. PiRacks Raspberry Pi (3, 2, 1 A , 1 B , Zero) Clear Acrylic 4-Stacker Rack Enclosure Box Storage System Case. https://www.amazon.com/dp/B077D4J3M5

[35] Nicole M Radziwill. 2017. Virginia Cyber Range. *Software Quality Professional* 19, 4 (2017), 46.

[36] Arvind S Raj, Bithin Alangot, Seshagiri Prabhu, and Krishnashree Achuthan. 2016. Scalable and Lightweight {CTF} Infrastructures Using Application Containers (Pre-recorded Presentation). In *2016 {USENIX} Workshop on Advances in Security Education ({ASE} 16)*.

[37] Raspberry Pi Foundation. 2019. Raspberry Pi 3 Model B+. https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/

[38] Paul Sroufe, Steve Tate, Ram Dantu, and Ebru Celikel Cankaya. 2010. Experiences During a Collegiate Cyber Defense Competition. *Journal of Applied Security Research* 5, 3 (2010), 382–396. https://doi.org/10.1080/19361611003601280

[39] Michael F Thompson and Cynthia E Irvine. 2018. Individualizing Cybersecurity Lab Exercises with Labtainers. *IEEE Security & Privacy* 16, 2 (2018), 91–95.

[40] F. P. Tso, D. R. White, S. Jouet, J. Singer, and D. P. Pezaros. 2013. The Glasgow Raspberry Pi Cloud: A Scale Model for Cloud Computing Infrastructures. In *2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops*. IEEE, Philadelphia, PA, 108–112. https://doi.org/10.1109/ICDCSW.2013.25

[41] Tyler. 2019. How to run a Raspberry Pi cluster with Docker Swarm. https://howchoo.com/g/njy4zdm3mwy/how-to-run-a-raspberry-pi-cluster-with-docker-swarm

[42] Viginia Business. 2017. Virginia Cyber Range to grow under new agreement. http://www.virginiabusiness.com/reports/article/virginia-cyber-range-to-grow-under-new-agreement

[43] Giovanni Vigna. 2004. iCTF. https://ictf.cs.ucsb.edu/

[44] Giovanni Vigna, Kevin Borgolte, Jacopo Corbetta, Adam Doupé, Yanick Fratantonio, Luca Invernizzi, Dhilung Kirat, and Yan Shoshitaishvili. 2014. Ten Years of iCTF: The Good, The Bad, and The Ugly. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*. USENIX Association, San Diego, CA, 7. http://www.usenix.org/conference/3gse14/summit-program/presentation/vigna

[45] Adam H. Villa. 2016. Hands-on Computer Security with a Raspberry Pi. *J. Comput. Sci. Coll.* 31, 6 (June 2016), 4–10. http://dl.acm.org/citation.cfm?id=2904446.2904447

[46] Jan Vykopal, Radek Ošlejšek, Pavel Čeleda, Martin Vizvary, and Daniel Tovarňák. 2017. Kypo cyber range: Design and use cases. In *Proceedings of the 12th International Conference on Software Technologies (ICSOFT)*, Vol. 1. SciTePress, Czech Republic, 310–321. https://doi.org/10.5220/0006428203100321

[47] J. Vykopal, M. Vizvary, R. Oslejsek, P. Celeda, and D. Tovarnak. 2017. Lessons learned from complex hands-on defence exercises in a cyber range. In *2017 IEEE Frontiers in Education Conference (FIE)*. IEEE, Indianapolis, IN, USA, 1–8. https://doi.org/10.1109/FIE.2017.8190713

[48] Michael Wirth and Judi McCuaig. 2014. Making Programs With The Raspberry Pi. In *Proceedings of the Western Canadian Conference on Computing Education (WCCCE '14)*. ACM, New York, NY, USA, Article 17, 5 pages. https://doi.org/10.1145/2597959.2597970

[49] Xiaoyang Zhong and Yao Liang. 2016. Raspberry Pi: An Effective Vehicle in Teaching the Internet of Things in Computer Science and Engineering. *Electronics* 5, 3 (2016), 9. https://doi.org/10.3390/electronics5030056