2021 ASEE ANNUAL CONFERENCE
Virtual Meeting | July 26–29, 2021 | Pacific Daylight Time
Paper ID #33589

A Call to Create an Open-source Project Initiative for Cybersecurity Virtual Labs

Dr. Radana Dvorak, City University of Seattle

Dr. Dvorak received her Ph.D. in computer science from the University of London, Queen Mary College and Master's in AI from the University of Sussex. Dr. Dvorak has been working in IT, higher education, academic industry and program development for over 25 years. She has served as a researcher, university professor and Dean in the US, UK, and the Cayman Islands. Currently, Dr. Dvorak is an associate professor and program manager at the School of Technology & Computing at City University of Seattle managing degree and certificate programs and teaching various CS courses. Her current research interests are related to teaching in STEM fields. She advises the cyber security club, and is a member of several organizations including OWASP-Portland Chapter. Dr. Dvorak is passionate about teaching, technology, career pathways and student success.

Mr. John L. Whiteman, University of Portland

John L. Whiteman is a security researcher for Intel Corporation and a part-time adjunct cybersecurity instructor for the University of Portland. He also teaches the UC Berkeley Extension's Cybersecurity Boot Camp. John received a Masters of Science in computer science from Georgia Institute of Technology, a Bachelors of Science in computer science from Portland State University and a Bachelors of Arts in Asian studies from the University of Maryland University College. John holds multiple security certifications, including Certified Information Systems Security Profession (CISSP), Certified Cloud Security Professional (CCSP), Certified Ethical Hacker (CEH), and CompTIA Security+. John is a U.S. Navy veteran who honorably served as a surface sonar technician and shipboard/classroom instructor. He is a member of the OWASP leadership team for the Portland, Oregon chapter and hosts a popular security podcast for them. John has over 20 years of experience in high tech with 10 years focused on security, working at startups, fortune 500 companies and government institutions.

A Call to Create an Open-Source Project Initiative for Cybersecurity Virtual Labs

Radana Dvorak Ph.D. & John L. Whiteman MSCS

Abstract

Cybersecurity classes present challenging problems to engineering and computer science departments. Having to negotiate with overstretched IT departments to set up specialized labs to support the curriculum and purchasing third-party cyber labs are not an option for many departments due to reduced budgets. Setting up environments is often left to the instructor after finding difficulties with the limited IT support/or lack of support. Instructors having to create labs is a problematic option since it is a very time-consuming overhead added to the regular activities involved in teaching a curriculum. COVID-19 has recently compounded this problem due to universities having to restrict access to labs.

Creating virtual labs for cybersecurity classes has been given attention in the last few years, and ASEE has published papers on the topic [1- 4]. Some universities are creating labs, while others are using the NSF-funded SEED Labs Project. [5-7]. The authors are proposing an opensource project initiative that allows universities, students, and others to contribute their lab work to a public repository hosted by an entity like GitHub. The work can be shared globally without costs or dependence on funding.

This presentation is divided into two parts. First, the success of developing hands-on virtual labs and their importance for cybersecurity classes is described. Second, the opensource initiative in greater detail is discussed outlining what was developed, and the authors call on universities to pilot our framework and invite interested parties to contribute to an opensource collaborative infrastructure platform currently under construction. The authors believe the success of this project has great potential for community colleges and universities.

Introduction

This paper addresses the challenging problems computer science departments undergo to prepare and set up specialized labs to support the cybersecurity curriculum. The paper first reports the success of delivering a cybersecurity curriculum designed based on a framework consisting of introducing a topic, followed by a real-life application practice lab. The authors also describe the platform developed to support the framework. These labs were deployed outside the university IT infrastructure and successfully supported hands-on-practice, and practical experience the industry expects of graduates. [8]

In teaching cybersecurity classes for the last two years, the authors found that students learn best by doing hands-on exercises immediately after a security concept is introduced in the lecture. The more traditional lecture followed by a different day for labs, often with only TAs present, has shown to be less effective. Furthermore, the authors found that the more realistic the scenarios, the more students became engaged and excited about the topics. The excitement and

further interest in the topic led to many students making decisions to pursue cybersecurity internships, apply for cybersecurity jobs or apply to Master's programs in cybersecurity. This initiative continues to be a work in progress as the labs are expanded and updated. The authors did not carry out empirical research to evaluate the outcomes. The outcomes reported are only from university-wide end-of-semester course evaluations; these evaluations are standardized across all courses. A survey will be designed, data will be collected, and research outcomes reported in the second phase of this work.

The realistic scenarios also provided students with the confidence to discuss real-life cases and demonstrate their knowledge and practical know-how in technical interviews when applying for internships and jobs. Documented complaints from the industry report that students have the 'theoretical' knowledge and can talk about how to *approach* the problem; however, when it came to actual demonstration of their skills, they were not competent and generally lacked hands-on experience. The lack of students' practical experience often stems from difficulties university departments have setting up environments, restricting students to practice skills such as pentesting[1], sniffing networks[2], or analyzing malware[3]- skills covered in cybersecurity courses.

The second part of the paper reports our solution to developing practical virtual labs based on real-world cybersecurity scenarios students will encounter at the workplace and outlines the framework for the labs' deployment. Lastly, the authors call on the community to pilot the proposed framework, invite interested parties to contribute to an opensource collaborative infrastructure platform, and contribute to developing and adding cybersecurity lab activities.

Background

Three cybersecurity classes were delivered in a classroom/lab setting before COVID-19 restrictions. One course was held in Spring 2020, half in the classroom and the remainder online because the University closed in-person classes. In Fall 2020, the course was delivered entirely online. Having the infrastructure set up before the University classes were entirely online made the transition seamless for students and the instructor compared to what other engineering courses experienced.

Lectures were designed around specific topics, taking to more than 20-30 minutes each. A short discussion followed. The labs were created based on highly publicized security incidents, like data breaches. After the breaches were introduced, students discussed the incidents and shared their thoughts on the ethics of the organizations' actions and/or legal outcomes. Initial work started on introducing ethical components to cybersecurity modules in 2019 [4]. The lab sections followed right after the discussion. The first lab is designed to teach students how to set up their personal laptop environments. Although students worked individually, they were able to chat and share in an online public forum. Instructors stressed the importance of figuring out the problem

[1] Pentesting (penetration testing), also referred to as ethical hacking, is an 'authorized' cyberattack. Can be simulated or real-life. Pentesters are hired to find security vulnerabilities in systems.

[2] Network sniffing, also known as packet sniffing (or sniffers) are snapshot copies of data flowing over a network without altering it or redirecting it.

[3] Malware is software designed to damage, disrupt or gain access to a computer system.

by themselves. If students requested help, instructors would first quiz the students on what was tried and guide them to discover the solution.

Here are several examples of the typical problem-solving scenario given to the students where each lab created a simulation of a real-world cybersecurity event:

- Extracted and analyzed malware from a binary image using opensource forensic tools. It was the infamous WannaCry ransomware that affected over 200,000 computers in 2017.
- Found a famous fugitive, John McAfee, by extracting coordinates from pictures taken of him while on the lam in Central America. [5]
- Created an encryption and decryption C program for one assignment and have it be continuously bombarded with garbage data to see if any security vulnerabilities exist. If so, students learn how to write more secure code to fix the problems.
- Ran a capture-the-flag event that simulated a vulnerable website that sold juice.
- Students participated in the National Cyber League (NCL) [6] competition during the term. This competition is only for college students. Over 4,700 students participated.

Course Outcomes

No formal survey was designed and administered to evaluate the outcomes. The instructors used the university-wide end-of-semester formal student course evaluation, compared grades from their previous offerings, and the improved result of students participating in the National Cyber League (NCL).

Students were very engaged in the labs; they reported enjoying them and felt it was a valuable learning experience. Grades were higher than prior offerings of similar content, which contributed to the engagement in being active participants in 'real-world' cybersecurity scenarios. As mentioned above, we did not carry out empirical research to report student performance and engagement. However, the end of the term evaluation showed that using real-life scenarios helped students understand and apply their cybersecurity concepts. Furthermore, students had the opportunity to experience what cybersecurity teams are face when confronted with a breach.

A few students used the open response area to provide more general feedback on the course. Most of the responses were positive, with students offering appreciation for this assignments' complexity compared with the more traditional lab elements.

Sample of Students' Comments:

- *I do think that in some cases it's good to get your hands dirty. I wish the course could be expanded or divided into more electives, as the material here is really important.*
- *Labs are fun and relatable to course content -- homework is derived from lectures, which I like -- the course gives great background on cybersecurity -- I really enjoyed the extra credit assignments*
- *I liked the interactive lessons*
- *Each of the topics taught was very interesting and the labs were also just as interesting as well.*

Virtual Lab Infrastructure

Creating virtual labs for cybersecurity classes has been given attention in the last few years, and ASEE has published papers on the topic. [1-4] Some universities create their labs while others use the NSF-funded SEED Labs Project [6-8]. The drawbacks of instructors creating labs are addressed above. Relying on graduate students is hit or miss, often not sustainable due to graduate students' priorities to graduate, often not replaced to support what was developed.

The solution to the ongoing challenges CS departments experience is to use a virtual lab framework as designed and developed for our courses. The authors propose to share what they have started to develop and launch an opensource initiative for the academic community.

First Class Offering

Using a cloud-based hosting environment for the labs was initially considered, but the costs had to be approved because of the lab fees already added to the course. Initial budget estimates seemed high too. It was decided to create a localize lab environment instead. Every student had a laptop, but the operating systems (OS) were not the same. Some students used macOS, others used Windows, and the Linux environments Instructors had to standardize the lab environment starting with the OS. One way to accomplished this was to use virtualization technology. Students installed a special software application called a hypervisor on their OS. A hypervisor runs a virtual machine (VM). VM is a virtual computer running inside the host OS. Each VM has its own configurable CPU, memory, networking capabilities and disk space. Instructors took advantage of this technology to install a Linux based OS to host the lab environment. As long as students installed a compatible hypervisor, they could run the same VM regardless of what host OS they used. Diagram 1 below shows a simplified virtualization configuration on a computer.

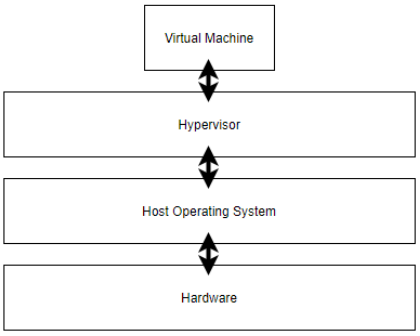

Diagram 1: Virtualization

During the first week of class, instructors asked students to install the hypervisor software on their laptops. This exercise was called *Lab 0*. Students also had to create three VMs that were needed to support all of the labs. Students learned the basics of virtualization while instructors also took the opportunity to teach basic virtualization security. For example, students learned why some types of network connections are less secure than others. Students also prepared special cryptographic keys that were later used to digitally sign homework to prevent cheating. Students later came to understand the purpose of these keys during the cryptography sections of the class. Finally, students were tested on these concepts during the midterm.

A hypervisor can run multiple VMs. To create a VM, students started with the following prepackaged VMs: Ubuntu, Kali Linux, and Metasploitable (Diagram 2)

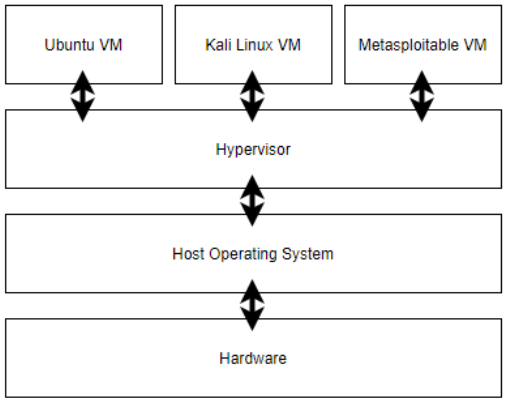

Diagram 2: Multiple virtual machines

After installing the operating systems (OS), students downloaded files from a public source code repository hosted on GitHub that contained our lab materials. Students were provided helper scripts to automate the setup process for each VM. Diagram 3 below shows the first VM framework, including some examples of the security applications, tools and samples that were used for future labs, and the network connections between the VMs and GitHub.

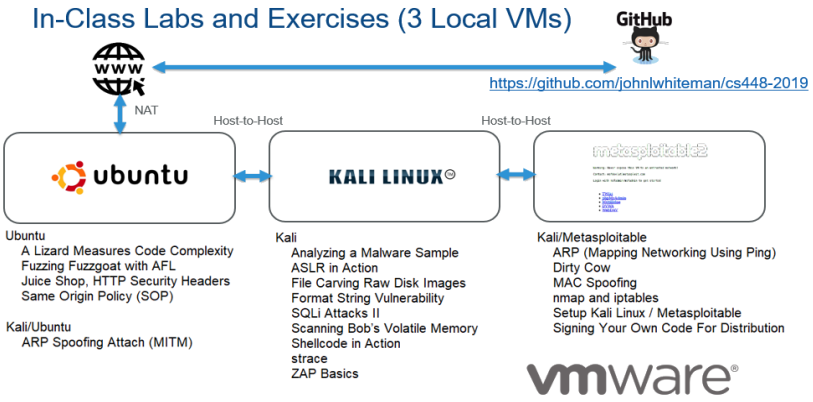

Diagram 3: First class offering

Another advantage of using VMs is isolation from the host OS. For example, one of the assigned labs involved reverse engineering a famous malicious computer worm called Stuxnet. It can only be executed on the Windows OS. Since all the VMs are Linux-based, there was no concern about the worm activating and spreading to the host environment.

Lessons Learnt of First Class Offering:
Advantages

- Standardize and secure lab environments
- Automated setup and teardown
- Students took ownership of the labs, not dependent on the IT department
- Instructors taught students about basic VM security principles
- All software was free

Disadvantages

- Running multiple VMs concurrently caused less powerful laptops to bog down or freeze
- Storage size of all of VMs took significant disk space, up to 75 GB
- Misconfiguration of network connects between VMs could disrupt the IT network
- Some students were not prepared to work in Linux command line environments

Second Class Offering

The number of VMs was reduced to only one. Kali Linux 2020 was chosen since it already came pre-packaged with many of the security tools used during the first iteration; this also decreased the storage footprint size to less than 25 GB.

Instructors initially worked with the IT department to host the VM on the school's network. Many advantages came from this approach:

1. Students could use the institution's computer resources instead of their laptops.
2. Instructors were also given administrative access to students' VMs in case students needed assistance with running the labs.
3. IT could also properly isolate the VM network from the rest school's main network to prevent accidental disruptions.
4. Only one master VM image was needed to be created by the instructors.

The VM was then handed-off to the IT department where each student got a unique copy.

The second offering of the course introduced the use of containers. Containers also perform virtualization but are limited to the OS. Consider each container as a component used to build a lab. Diagram 4 depicts an example of an architecture that uses one container as a web application server and another container as a security tool that tests the web server for potential hacking vulnerabilities.

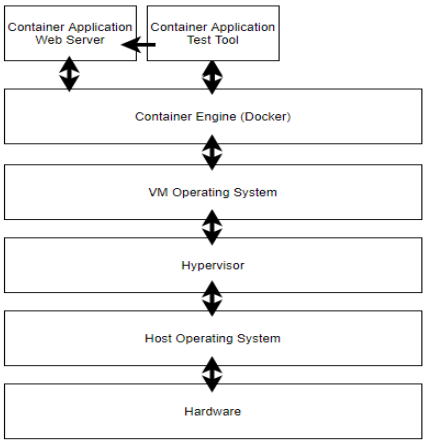

Diagram 4: Containerization with VM

By packaging individual labs into containers enabled the environments to be standardized. All students had the same recipe for each lab with performance and security in mind. Using this framework enabled the instructors to add more labs and class exercises than before through the

use of containerizations. Because many of the labs were built based on currently existing ones, labs did not have to be rewritten - containers were used as building blocks.

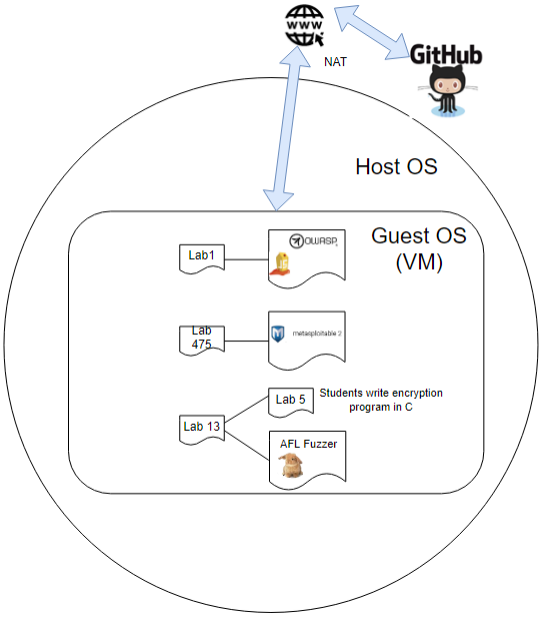

Diagram 5: Container interconnectivity

The diagram above (Diagram 5) shows a simplified overview of potential containers running inside the VM's guest OS. Instructors demonstrated that each lab could connect to other containers without exposing the IT network. This provided additional security.

- *Lab 1 is a container that connects to an intentionally vulnerable website called JuiceShop. The lab is a capture-the-flag (CTF) event that allows students to learn how to hack and, in doing so, collect points. They are free to use offensive tools such as fuzzers to find these vulnerabilities. Everything is done securely in containers inside the VM.*
- *Lab 475 is a container that connects to an intentionally vulnerable OS called Metasploitable. It's completely isolated from the host including its antimalware tools.*
- *Lab 13 is an example of a lab that combines two other labs. Lab 5 is an older lab completed by the students who were supposed to write a fully functional encryption and decryption program in the C language. Later in the term the instructor assigns this lab to see if the AFL fuzzing tool can find vulnerabilities in their program by continuously bombarding it with garbage data.*

Third Class Offering

The third time the course was offered, instructors had minimal technical issues experienced in the initial offerings. For example, Lab 13 described above can be pieced together to provide continuity during a course. Below is an extreme example of how multiple labs can be stitched together to create a compelling and exciting assignment that uses real-world artifacts:

1. *Students are given a file containing recorded wireless traffic from an overseas hotel in Asia.*

2. *They are requested to analyze it using a few opensource tools such as Wireshark and tcpreplay. The traffic is encrypted except at the start where a cryptographic handshake takes place to establish the secret key.*
3. *Students are asked questions about this exchange including the random values that are passed in the clear between the client and server. Later the students are prompted to find anomalous traffic. This is next to impossible since the traffic is encrypted, so the instructor helps by providing the actual secret key to decrypt it.*
4. *The students quickly observe that a suspicious Windows executable is sent from the server to the client.*
5. *They use a forensics tool to extract it from the rest of the file and with a free malware analyzer, they find that it contains the infamous WannaCry ransomware worm that affected hundreds of thousands of computers across 150 countries in just days, including hospitals and other critical infrastructure. The damage could have been worse though except that a young British security researcher found a hidden "kill switch" hardcoded in the program to stop the virus from spreading.*
6. *The students download the Ghidra reverse engineering tool provided by the National Security Agency (NSA). They find the same kill switch as the security researcher with excitement because they are looking at the real thing.*
7. *Instructor delivers a lecture about legal and ethical issues associated with computer security. Instructor informs students that the security research was also wanted for alleged security crimes he committed in the past. All of this was happening at the same time.*
8. *Students are given an assignment to debate the issue and make a decision about criminal activities – court room scenario. Students need to decide whether the hacking hero should be exonerated of his alleged previous criminal past.*

Next Class Offerings

After offering the same course for two years, the instructors believe using a cloud service provides the best experience for the students. A cloud service has successfully been used for similar training at Oregon Health and Science University (OHSU). One of the instructors taught a web security course where Amazon Web Services hosted the labs. The model is similar to the IT networking effort, but, in this case, the students had complete control of the setup and teardown of the labs while the instructors fully managed the VM updates.

Cost is the biggest factor when considering using the cloud to host labs. Charges take into account the software installed, the hardware that it uses, how much data is transferred, how many instances per student, and how long each remains active. These costs add up fast and can exceed any reasonable lab fee for students.

Creating an Opensource Community to Provide Free Content

Successful opensource projects [10] [11] have a strong, active, transparent community behind them. Our goal is to create an opensource project that designs and leverages trusted and reliable content from anyone who wants to contribute. We start with a free public Git repository like GitHub, GitLab, or BitBucket. We use the industry's best practices to organize the content.

Contributions

Any member in the opensource community can contribute, whether new lab exercises, bug fixes, testing, and documentation. New lab content is first peer-reviewed by the community to ensure quality and security before submission. If bugs are identified, they can be reported and tracked in the repo's ticketing database. All labs require documentation using a well-known formatting language such as Markdown. Students can even contribute to the labs. They are the end-users who can add the most significant value to them. Instructors strongly encourage this. Making contributions to an opensource project focused on security are relevant experiences students can showcase and add to their resumes. The author goal is to reach a wider academic community, excite them about what is being created, and contribute to a successful project. Since this is an opensource project, it should use the best-known processes in place. At the heart of contribution is something called a Pull Request on GitHub, described below.

The quality control is maintained by restricting members from adding their content until they first go through a pull request (PR) process. Each PR contains a description of the proposed changes along with the content itself. Any member of the community can review the PR, often adding observations along the way. Once a PR is thoroughly reviewed, a core project member can accept the PR by merging it or reject it. The PR process is the same used by nearly every other opensource project that uses Git. Community peer review helps keep the integrity of the project intact as well as encourages collaboration and transparency. Who are the core members? The community can decide that, but, in most cases, it is usually someone (or a core group) who has the most experience with the project. This can also be an experienced industry volunteers who are interested in the initiative. Everything is tracked via the Git logging system. Everyone is encouraged to contribute to the opensource repository, including development and testing.

Maintenance

Diagram 6 below proposes a well-known branch flow used by many other organizations. Starting from the bottom, the new branch contains cutting-edge content not officially released. Once mature, it gets merged to the Dev (Development) branch, where it gets integrated with existing content. Eventually, the Dev branch is tagged and merged into the Release branch. How often a release occurs is determined by the community; either by quarter or semester might make the most sense in the academic community. Releases are merged into the Master branch. A Path branch houses emergency fixes in case critical bugs are reported after a release. All other bugs are tracked in a bug database, which is then dispatched to developers to fix in the Dev branch.

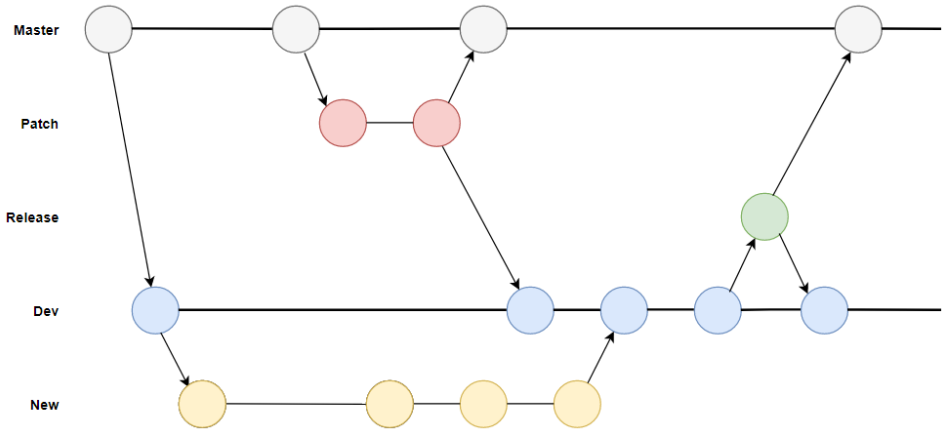

Diagram 6: Git repository

Discussion

The delivery model has been deemed successful based on students' engagement, improved grades and end of the course evaluations. Both instructors will continue to teach using the opensource lab framework. Of the students who completed the courses, twenty percent (20%) decided to pursue cybersecurity fields from our junior and senior standing students. Two students were admitted to the Carnegie Mellon University Cyber Security Master's program; other students successfully secured internships. This is a personal success for the instructors dedicated to exciting students to pursue careers in cybersecurity to help fill the extreme technical gap currently experienced in the industry. Recently reported by "The Cybersecurity Talent Gap Is An Industry Crisis", stating 3.5 M jobs worldwide will not be filled by 2021 [12]. A recent workforce study published in 2019 by (ICS)2, [13] the world's largest non-profit association that certifies cybersecurity professionals, reported that to meet the needs of the cybersecurity workforce globally, skilled cybersecurity professionals need to grow by 145% to meet the demand. The report further stated that in order to meet American businesses' crucial needs, the US cybersecurity workforce needs to grow by 62%!

Conclusion - future work

The success of opensource projects is well documented [14] [15]. Most recently, GitHub, acquired by Microsoft Inc. 2018 launched GitHub Learning in 2019 [16], fully supporting education lab classrooms; however, there is still *not one* opensource project specifically aimed at supporting educators to teach cybersecurity curriculum.

The proposed work with the virtual lab framework has demonstrated initial success with just several modules; future work will continue to build on the initial structure and ideally be tested by partnering universities. Processes need to be in place to ensure quality and security. We ask the community whether they are interested in this project and help with the seed development. In addition to continued development to expand the virtual lab environments, we are looking for:

- Cloud support – partner with Microsoft, Amazon or IBM
- Grant – submit grants to help fund the development

- Licenses – work with opensource licensing agencies

We are proposing and calling for an opensource project initiative that allows universities, students, and others to contribute their lab work to a public repository hosted by an entity like GitHub. Our goal is to provide a centralized, trusted platform where tools are vetted and cataloged beforehand. We also would like to showcase the work of contributors by proposing an annual security conference. Students and instructors can present their work as conference speakers and workshops. Partners can also attend these conferences to recruit college graduates and internships for those still attending.

As we continue to add to the labs, we plan to formalize our research, collect data on the program's success utilizing questionnaires, and research whether the outcomes led to continued interest in pursuing cybersecurity internship, jobs and enrolling in graduate programs in cybersecurity. The authors are also interested in collecting demographic data to study non-represented populations in cybersecurity. The research outcomes will be reported in future conferences.

References

[1] Mendlein, A., & Nguyen, T., & Rege, A. (2020, June), *Cybersecurity Awareness and Training through a Multidisciplinary OSINT Course Project* Paper presented at 2020 ASEE Virtual Annual Conference Content Access, Virtual On line . 10.18260/1-2—34367

[2] Carpenter, A. (2018, June), *A Hardware Security Curriculum and its Use for Evaluation of Student Understanding of ECE Concepts* Paper presented at 2018 ASEE Annual Conference & Exposition , Salt Lake City, Utah. 10.18260/1-2—29688

[3] Whipple, A., & Smith, K. B., & Rowe, D. C., & Moses, S. (2015, June), *Building a Vulnerability Testing Lab in an Educational Environment* Paper presented at 2015 ASEE Annual Conference & Exposition, Seattle, Washington. 10.18260/p.23640

[4] Dvorak, R., Dillon H., Ralston, N., Welch, J. (2020, June*), Exploring Ethical Hacking from Multiple Viewpoints,* Paper presented at 2020 ASEE Virtual Annual Conference Content Access, Computing and Information Technology, https://peer.asee.org/34640.

[5] Cluley, Graham (2012), "*Fugitive John McAfee's location revealed by photo meta-data screw-up".* Available at, https://nakedsecurity.sophos.com/2012/12/03/john-mcafee-location-exif/

[6] W. Du, "SEED: Hands-On Lab Exercises for Computer Security Education," in IEEE Security & Privacy, vol. 9, no. 5, pp. 70-73, Sept.-Oct. 2011, doi: 10.1109/MSP.2011.139.

[7] Wenliang Du. 2015. SEED Labs: Using Hands-on Lab Exercises for Computer Security Education (Abstract Only). In Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15). Association for Computing Machinery, New York, NY, USA, 704. DOI:https://doi.org/10.1145/2676723.2678290

[8] SEED Labs; https://seedsecuritylabs.org/

[9] Nuryake Fajaryati, Budiyono, Muhammad Akhyar and Wiranto. The Employability Skills Needed To Face the Demands of Work in the Future: Systematic Literature Reviews. Open Eng. 2020; 10:595–603. Available from: DOI: https://doi.org/10.1515/eng-2020-0072

[10] G. DeKoenigsberg, "How Successful Open Source Projects Work, and How and Why to Introduce Students to the Open Source World," 2008 21st Conference on Software Engineering Education and Training, Charleston, SC, 2008, pp. 274-276, doi: 10.1109/CSEET.2008.42.

[11] Onitiu, Daria, 'The Case for Open Source' a Report on Matt Wells' Seminar by NINSO, Northumbria Internet & Society Research Group (June 19, 2020). Available at SSRN: https://ssrn.com/abstract=3630841 or http://dx.doi.org/10.2139/ssrn.3630841

[12] The Cybersecurity Talent Gap Is An Industry Crisis Forbes Technology Council, Brian NeSmith Forbes Councils Member Forbes Technology Council, https://www.forbes.com/sites/forbestechcouncil/2018/08/09/the-cybersecurity-talent-gap-is-an-industry-crisis/#26548da6b367, Aug 9, 2018,

[13] "Strategies for Building and Growing Strong Cybersecurity Teams" (ISC)2 Cybersecurity Workforce Studey, https://www.isc2.org (2019)

[14] https://techcrunch.com/2019/01/12/how-open-source-software-took-over-the-world/ January 12, 2019 Mike Volpi.

[15] Webber, S., (2009), "The Success of Open Source", Harvard University Press.

[16] GitHub Learning Labs; https://lab.github.com/