EZSetup: A Novel Tool for Cybersecurity Practices Utilizing Cloud Resources

Yanyan Li
Department of Computer Science
University of Arkansas at Little Rock
Little Rock, AR, USA
yxli5@ualr.edu

Dung Nguyen
Department of Computer Science
University of Arkansas at Little Rock
Little Rock, Arkansas, USA
dvnguyen@ualr.edu

Mengjun Xie
Department of Computer Science
University of Arkansas at Little Rock
Little Rock, Arkansas, USA
mxxie@ualr.edu

ABSTRACT

Recent years have witnessed fast growth of interests and efforts in developing systems and tools for cybersecurity practices, especially with the rapid advancement of cloud computing. However, those systems either suffer from issues in scalability, customization and setup complexity or are constrained by a specific cloud technology and limited customization support. In this paper, we present a novel Web based tool called EZSetup that can create and manage user-defined virtual environments for various types of cybersecurity practices on demand and at scale. Distinct from previous cloud-based systems, EZSetup does not rely on a particular type of cloud platform or technology. It is able to interact with multiple clouds and instantiate virtual environments in multiple clouds simultaneously. EZSetup allows for easy customization and significantly reduces the overhead in creating and using practice environments through carefully designed Web interfaces. The experimental results are quite positive in general and indicate that EZSetup can also be applicable to other computer science and engineering subjects.

KEYWORDS

Cybersecurity Practice; Security Lab; Cloud Computing; Tool Development

1 INTRODUCTION

The gap between demand and supply for cybersecurity professionals is huge [10]. According to Forbes [8], the number of cybersecurity job openings is expected to rise to 6 million by 2019 and one quarter of them, roughly 1.5 million jobs, are projected to be unfilled. Meanwhile, cyber attack and data breach events occur constantly (e.g., [4, 6]). Therefore, training and enhancing current and next-generation cybersecurity professionals via effective, affordable, and scalable mechanisms is vital to fill in the gap.

Hands-on cybersecurity practices are indispensable to cybersecurity education and training. They can be applied in various forms such as regular course labs and extracurricular cyber defense competitions. The system settings of those practices can range from a simple virtual machine to a complex network with multiple subnets and server systems. Although a cybersecurity practice environment built with physical hardware is usually effective, the high cost and low scalability make it less popular and only adopted by well-funded activities (e.g., collegiate cyber defense competition or CCDC [12]). With the advancement of virtualization technologies, virtual machine (VM) based approaches to building cybersecurity practice environments become popular and there exist a number of VM-based solutions such as SEED Labs [3], Open Cyber Challenge Platform (OCCP) [13], ISERink [5], V-NetLab [11], and Platoon [7]. Those tools and systems have greatly facilitated the widespread of cybersecurity practices due to their much reduced cost in environment setup and scenario creation. However, they suffer from the following issues: 1) They are not designed for large-scale deployment of practices; 2) The deployment of those systems still requires nontrivial effort and beyond entry-level system and networking knowledge and skills, which becomes a barrier to their wide adoption; 3) They lack strong and user-friendly support for customization.

The rapid development of cloud computing also brings cybersecurity practices into clouds. Recent years have witnessed the flourish of cloud-based security practice platforms and systems such as the training system at the Center for Systems Security and Information Assurance (CSSIA) [2], NICE Challenge Project [9], and V-Lab [14]. Those systems are able to support large-scale deployment of security practice scenarios. However, they usually are built using a specific cloud technology and do not expose an interface to their users for customization.

In this paper, we propose a novel Web application called EZSetup, which is capable of creating a variety of user-defined cybersecurity practice environments (e.g., labs and competition scenarios) in one or multiple computing clouds (e.g., OpenStack and Amazon AWS). Distinct from previous cloud-based solutions, EZSetup does not rely on a particular type of cloud platform or technology. It is able to interact with multiple clouds and instantiate virtual environments for security practices in multiple clouds simultaneously, which makes EZSetup flexible and scalable. EZSetup provides a Web user interface for practice designers to visually create a practice scenario (drag and drop icons and link them) and separates the Web interface for practice designers, namely admin panel, from the one for practice executors (i.e., end users), namely user panel. By doing so, EZSetup allows for customization and at the same time significantly reduces the overhead in creating and using practice environments. Completely hidden from the complexity in creating practice environments, end users can enjoy a quick start and fully concentrate on security practices. We recruited nine volunteers to

SIGITE'17, October 4–7, 2017, Rochester, NY, USA

ACM ISBN 978-1-4503-5100-3/17/10…$15.00
https://doi.org/10.1145/3125659.3125699

evaluate the current prototype of EZSetup in both lab creation as a lab manager and lab exercise as a lab user. The experimental results are in general quite positive and indicate that EZSetup can also be applicable to other computer science and engineering subjects.

The rest of this paper is organized as follows: We briefly describe the related work in Section II. We then present the design of EZSetup in Section III and its implementation in Section IV. We detail the evaluation of EZSetup in Section V and conclude this paper in Section VI.

2 RELATED WORK

Recent years have witnessed the growing interest in applying virtualization technologies to cybersecurity learning and practices. Many tools have been proposed and used in security education and training. Based on the architectural model of those tools, they can be roughly classified into the following categories: hosted hypervisor based (e.g., [3, 13]), bare metal hypervisor based (e.g., [5, 7, 11]), and cloud based (e.g., [1, 14]).

A hosted hypervisor refers to a hypervisor program running on top of an existing OS (e.g. Linux or Windows), e.g., Virtual Box, VMware Workstation, and Parallels Desktop. Tools relying on this type of hypervisor include SEED Labs and OCCP [3, 13]. SEED Labs are a collection of security labs designed primarily for security courses and individual practices. OCCP (Open Cyber Challenge Platform) is designed for various challenge scenarios such as network defense and penetration testing.

A bare metal hypervisor refers to a hypervisor that directly controls physical hardware, e.g., VMware ESXi , Citrix XenServer, and Microsoft Hyper-V. Tools relying on this type of hypervisor include V-NetLab, ISERink, and Platoon [5, 7, 11], all of which run on top of VMware ESXi. V-NetLab is a virtual network lab platform designed for networking courses and network related labs. ISERink is a virtual system that is initially designed to support cyber defense competitions. Platoon is a team-oriented cybersecurity exercise platform for both security labs and competitions.

Cloud-based platforms are the new trend where complex networks and labs can be created at scale using public or private cloud resources. Example systems of this type include V-Lab and DETER Lab [1, 14]. V-Lab is a cloud-based virtual lab education platform for hands-on course experiments. The DETER Lab (Cyber-Defense Technology Experimental Research Laboratory) is a research testbed mainly for cyber defense research on large-scale network attacks such as DDoS and botnet.

One uniqueness of our work is that EZSetup supports not only private clouds but also public clouds, which is not available in other security practice tools and systems. In addition, the capability of working with multiple clouds and on-demand lab creation and server auto-configuration features make EZSetup highly flexible and scalable.

3 SYSTEM DESIGN

The overview of EZSetup is shown in Figure 1, where EZSetup consists of a frontend (Web-based user interface (UI)) and a backend (the internal database not shown in the figure). EZSetup is a Web application. It can be deployed on a physical or virtual Web server. In order to use EZSetup to create a virtual cybersecurity practice environment in a computing cloud (e.g., an OpenStack cloud), a user is required to have access to the target cloud and have necessary cloud resources (such as computing, networking, and storage resources) for the environment.

EZSetup aims to make it easy to create and deploy a cybersecurity practice scenario (e.g., a lab or a competition) in a cloud by providing a user-friendly frontend and hiding all the technical complexity in the backend. A lab instructor/manager or a competition creator only needs to fill in the cloud access credentials and a few parameters (e.g., a specific practice scenario and the number of users) to instantiate a scenario. EZSetup automates the deployment process by utilizing the specified cloud resources. Once the deployment is finished, people seeking practice can log into EZSetup and select a scenario (called lab in the prototype) to join. By doing so, complexity in configuring a cybersecurity practice environment (e.g., setting virtual networks, routers and servers) is hidden to EZSetup users whose interest is in security practices instead of environment setup. In addition, the burden for instructors or managers is also significantly reduced. EZSetup is designed to provide the following features.

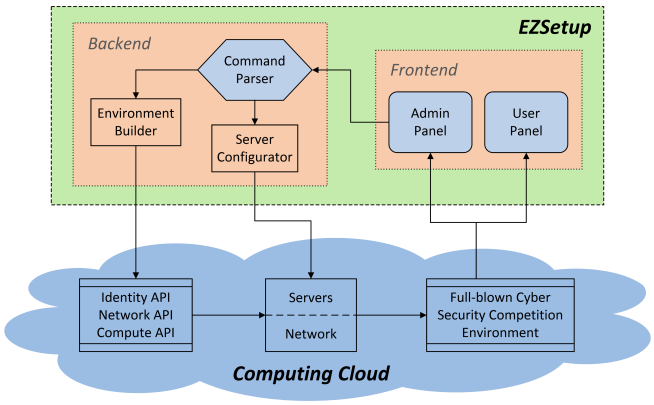

Figure 1: System Overview of EZSetup

- **Scalable and on-demand service.** EZSetup is able to deploy security practice scenarios at scale (e.g., creating labs for a few students as well as hundreds of students) by leveraging elasticity of computing clouds. Moreover, EZSetup deployment is on demand, making it flexible and cost effective.
- **Customizable scenarios.** EZSetup uses YAML based scenario templates, in which the network topology and system configurations can be clearly defined. With the Web-based visualization UI, a scenario can be easily created by dragging and dropping network and server icons onto a canvas and making specific settings on them.
- **Multi-cloud support.** EZSetup is not designed for a particular cloud. Instead, it can work with multiple cloud providers, e.g., OpenStack, Amazon AWS, Google Cloud Platform. Lab managers and competition organizers can choose the most appropriate cloud for their needs.
- **Open source.** EZSetup is an open source tool, aiming to benefit the cybersecurity education community and facilitate anyone interested in cybersecurity.

• **Portability.** EZSetup is a Web application and supports all major Web browsers.

• **Flexible user management.** EZSetup adopts a multi-layer user management scheme, which is illustrated in Figure 2. All users can be divided into three categories, i.e., EZSetup manager, lab manager, and lab user (lab here refers to a general practice scenario). Lab managers and lab users are based on group and managed by a EZSetup manager. Groups are created based on course offerings or competition needs. For example a computer security course can have its own group with all registered students as lab users for that group and the course instructor or teaching assistant who create and manage security labs serving as the lab manager. A lab manager is able to create, modify, and delete one or more labs. A lab user can only join one or multiple existing labs but cannot create a lab. A EZSetup manager is able to create, modify, and delete lab managers and lab users. For a cyber defense competition scenario, competition organizers can act as the lab manager, and each participating team joins as a lab user, with all team members sharing the same lab user account. In addition, a lab user can join in multiple groups if he or she registers multiple courses. Similarly, a person can serves as multiple lab managers (e.g., a TA for multiple courses).

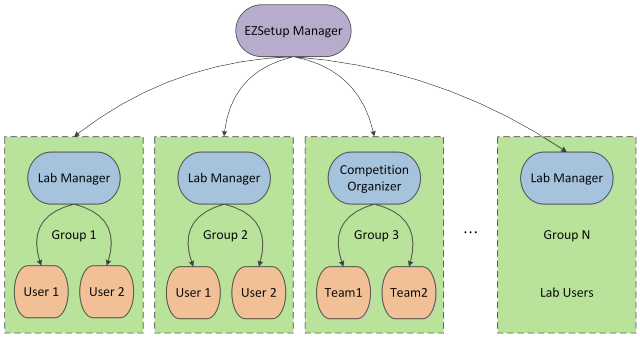

Figure 2: Multi-layer user management structure

3.1 Frontend Functions and Design

The frontend UI contains two different webpages: admin panel and user panel. The admin panel is for EZSetup manager and lab managers where they can manage groups and labs. Several components are used in the admin panel including scenario designer, deployment management, and lab management. The Scenario designer component helps a lab manager design lab scenarios through a visualization UI (i.e., via drag and drop on a canvas). The deployment management UI helps deploy labs onto a cloud by providing a webpage for deployment parameters and interacting with the specified cloud. The lab management UI receives network and server status information from the target cloud and displays the status of each created slice in EZSetup. A slice is a collection of the resources dedicated to a user account. The number of slices in a lab depends on the number of user accounts in need to access that lab.

EZSetup provides a user panel for each lab user helping him or her interact with all available and joined labs within the group. The user panel aims to provide necessary information for users and make them concentrate on practices instead of lab management. Two main components–lab listing and lab viewer–are included in the user panel. Lab listing is able to retrieve the lab user's group information from the internal database and to display all the labs created within the group. Joining in one lab essentially make a user get one slice from the lab's reserved resources. After joining a lab, the lab user can see the lab's network topology as well as detailed network and server information through the lab viewer.

3.2 Lab Scenario

In EZSetup, we use scenarios to describe and manage different types of security exercises such as labs and competitions. A scenario consists of a scenario template file describing necessary networking and computing resources and a number of server configuration files specifying the software packages to be installed and the system settings to be applied. Scenario creation and customization is supported through EZSetup Web GUI to cater to the needs of different users and to encourage creation of new scenarios. EZSetup is aimed to support a wide spectrum of security labs and competitions to save its users' time in building practice environments. For instance, the scenarios under development include network security labs (e.g., SYN Flooding attack and DNS attack scenarios), Web security labs (e.g., Cross-Site Scripting attack and SQL Injection attack scenarios), and a CCDC-style cyber defense competition scenario. The list of scenarios is expected to grow quickly once EZSetup is officially released.

3.3 Backend Functions and Design

The main function of the backend is to parse the input specified on the admin panel by a lab manager and to interact with the target cloud through specific cloud API to instantiate the scenario(s) in the cloud. The backend consists of three types of modules: a command parser, environment builders, and a server configurator. The command parser is able to not only select an appropriate environment builder based on the specified cloud provider, e.g., OpenStack or Amazon AWS, but also reconstruct the scenario template based on the specified number of users or teams (slices) so that the reconstructed scenario template defines all the resources required by users or teams. Essentially, a scenario template created on EZSetup Web GUI only includes the resources needed for one slice. The total resources that are actually needed are calculated by the command parser based on the slice information filled in by a lab manager.

An environment builder generates the required networks and VM instances for the specified scenario. Several environment builders are provided in EZSetup, each responsible for interacting with one particular cloud. For example, there is one builder for OpenStack and one for Amazon AWS. The environment builders essentially enable EZSetup to support multiple clouds. A separate session is created and maintained for each lab deployment request, which makes multiple simultaneous deployments possible. To fulfill a deployment request, one environment builder needs to employ at least three types of APIs provided by a cloud: identity API, network API, and compute API. The identity API is needed because the initiator's identity has to be verified before resources are granted. The network and compute APIs are for requesting for the networking and computing resources from the cloud.

The server configurator is mainly used to set up servers based on the specified configuration, e.g., installing a particular software

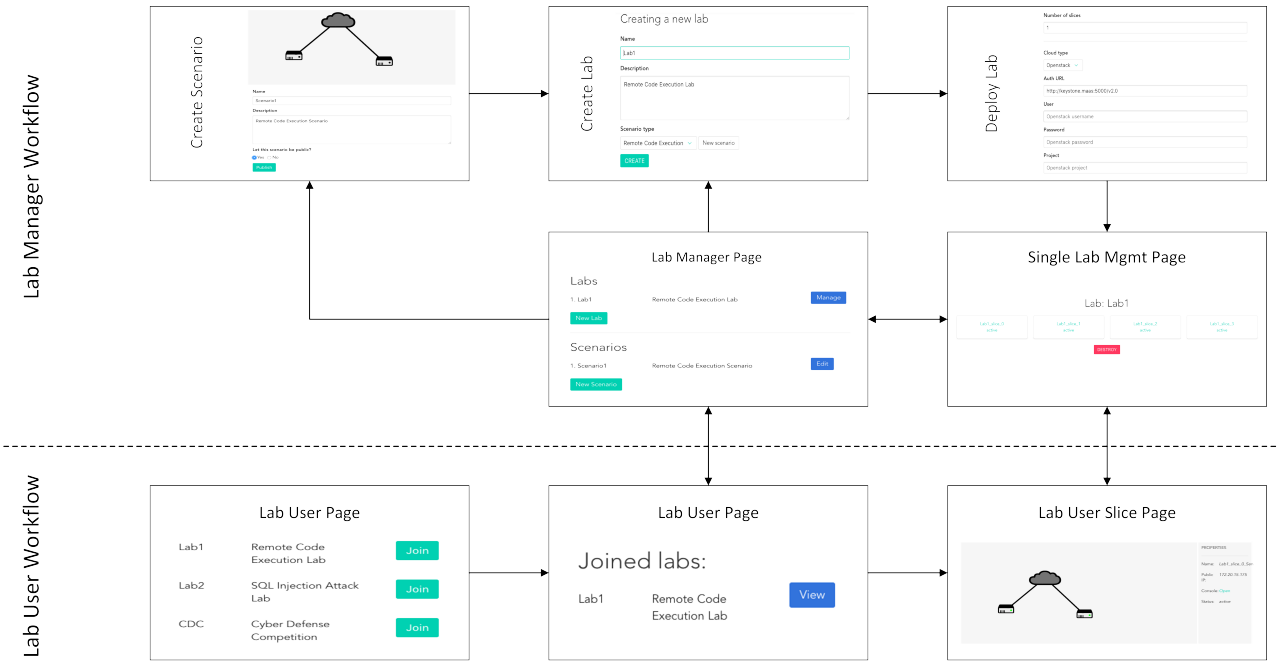

Figure 3: Workflows for a lab manager (upper) and for a lab user (lower)

package or adjusting an application or system setting. The server configuration is conducted using Ansible playbooks, which define all the actions to be performed on the server so that the configuration is automated without human intervention.

Last but not least, an SQL database is internally employed in the backend to bookkeep user and system information including user and manager account information, group information, and lab settings.

3.4 Usage

Similar to other Web applications, EZSetup is accessed through a web browser and a user has to pass the login page for access. Once logged in, the admin panel will be displayed for a lab manager or admin and the user panel will be displayed for a normal end user. A lab manager on the admin panel can either choose an existing scenario from a a drop-down list or create a custom one on the scenario design canvas. After that, a lab manager can create a lab based on the selected scenario and then deploy it to a cloud (e.g., OpenStack or Amazon AWS) by supplying their cloud credentials. The last step is to define the number of slices for the lab. A separate slice will be created for each user afterwards.

After logging in, regular lab users can find the groups they belong to and the labs they have joined. They can find the network topology information of their registered lab such as IP addresses and SSH command. Console access and snapshot creation are realized via the compute APIs provided by the different cloud services. The workflows for both lab managers and lab users are illustrated in Figure 3.

```
- network: LAN
  cidr: 192.168.0.0/24
- network: DMZ
  cidr: 192.168.1.0/24
- network: WAN
  cidr: 172.32.1.0/24
- server: Email
  interfaces:
    - network: LAN
      ip: 192.168.0.11
    playbook: email_server.yml
- server: Web server
  intefaces:
    - network: DMZ
      ip: 192.168.1.11
  playbook: web_server.yml
- router: Shorewall
  interfaces:
    - network: LAN
      ip: 192.168.0.1
    - network: DMZ
      ip: 192.168.1.1
    - network: WAN
      ip: 172.32.0.1
```

Figure 4: A Sample scenario template

4 IMPLEMENTATION

The core of EZSetup was implemented in Python 3. The Web UI is based on Vue, a modern Javascript framework. The internal database uses PostgreSQL. Certain frameworks and libraries have to be installed before running EZSetup, e.g., Flask, Flask-Login, psycopg2, argon2-cffi, boto3, cloud sdk, and openstacksdk. Flask is a python microframework for developing Web applications. Flask-Login helps manage user sessions for Flask. psycopg2 is a PostgreSQL adapter for Python that allows using Python code to manage PostgreSQL databases. argon2-cffi provides an efficient password hashing function. The three Python-based SDKs boto3, cloud sdk, and openstacksdk are provided by Amazon AWS, Google Cloud Platform and OpenStack respectively, and they allow for EZSetup to communicate with their respective cloud platform.

```
- name: Install nginx
  apt: name={{item}} state=installed
  with_items:
    - nginx
- name: Copy index.html file
  template: src=index.html.j2 dest=/usr/share/nginx/html/index.html
- name: Create the nginx configuration file (non-SSL)
  template: src=site.conf.j2 dest=/etc/nginx/sites-available/{{prj_n
ame}}
  when: not use_ssl
- name: Ensure that the default site is removed
  file: path=/etc/nginx//sites-enabled/default state=absent
- name: Ensure that the application site is enabled
  file: src=/etc/nginx/sites-available/{{prj_name}} dest=/etc/nginx/
sites-enabled/{{prj_name}} state=link
  notify: reload nginx
- name: Ensure nginx service is started, enable service on restart
  service: name=nginx state=restarted enabled=yes
- name: Stop nginx for local dev, disable service
  service: name=nginx state=stopped enabled=no
  notify: stop nginx
  when: not enabled
- name: Allow incomming http request
  ufw: rule=allow port=http
```

Figure 5: Ansible playbook file (web_server.yml)

Our scenario file uses a YAML-based domain specific language, an example of which is shown in Figure 4. This example defines a blue team network for a cyber defense competition scenario. There are three subnets defined in a blue team network, i.e., LAN, DMZ, and WAN. Inside each subnet, there are servers with basic networking information. The configuration of these servers is defined in separate Ansible playbooks.

One example of an Ansible playbook for an nginx Web server is given in Figure 5. The installation and configuration process involves multiple tasks, each of which has a name and a specific action or a state check, e.g., using apt to install nginx, copying a pre-defined web page to a specific location and enabling it, and removing the default webpage. With its own Ansible playbook, each server can be quickly and automatically configured to the desired state.

5 EVALUATION

In this section, we first conduct a comparison between EZSetup and other virtual solutions to cybersecurity practices and then present users' feedback on using EZSetup both as a lab manager and as a lab user.

5.1 Feature Comparison

We compare EZSetup with other similar solutions that offer virtual machine (VM) based security practices including SEED Labs, V-Lab, OCCP, ISERink, Platoon, V-NetLab and DETER Lab. Those solutions are evaluated against the features important for hosting cybersecurity practices. The detailed comparison results are shown in Figure 6.

From Figure 6, we can see that not all the systems support security competitions. This is due to the fact that competition environments are usually more complex and demand more resources and efforts to build. All the platforms that support competitions also support teamwork. Self-hosting is desirable in that it allows users to deploy environments themselves and have full control. Based on the complexity of installing and configuring the system itself, only Platoon and EZSetup have "easy deployment" marked in that both provide installation scripts for automating system deployment, which is critical for self-hosting. On-demand creation is an

	Support Security Labs	Support Security Competitions	Support Teamwork	Can be self-hosted	Easy deployment	Deployed in public cloud	Work with multi-cloud	On-demand creation	Scenarios can be customized	Easy Customization	Servers auto-configured
SEED Labs	✓	×	×	✓	×	×	×	×	×	×	×
V-NetLab	✓	×	×	×	×	×	×	×	×	×	×
V-Lab	✓	×	×	×	×	×	×	×	×	×	×
OCCP	✓	✓	✓	✓	×	×	×	×	✓	×	×
ISERink	✓	✓	✓	✓	×	×	×	×	×	×	×
Platoon	✓	✓	✓	✓	✓	×	×	×	×	×	×
DETER Lab	✓	×	×	×	×	×	×	✓	✓	✓	✓
EZSetup	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Figure 6: Comparison of systems for security practices

appealing feature of EZSetup as it gives users a fine-grained control over scenario creation and allows them to flexibly create and delete scenarios. The level of scenario customization is important for instructors and scenario creators. EZSetup gains the easy scenario customization feature by using scenario template and visualization UI. Another attractive feature of EZSetup is that the configuration of the systems specified in a scenario can be automated through Ansible playbooks.

EZSetup has two unique features: the ability to be deployed in public clouds and multi-cloud support. The ability to deploy in public clouds empowers users in short of hardware resources to host security practice labs or competitions at scale. Working with multiple clouds gives users a great flexibility to select appropriate target cloud(s) for deployment from different clouds so as to maximize their benefits.

5.2 User Feedback

We also collected user feedback via a survey given to the voluntary testers immediately after they completed two tasks: creation of a security lab as a lab manager and the following lab exercise as a lab user. In the test, EZSetup was deployed on a virtual Web server and was able to interact with a small private OpenStack testbed that was built using 4 old Dell R410 servers.

The evaluation was conducted in June 2017. We recruited 9 science and engineering students for it. The task of lab creation is to create a remote code execution lab through EZSetup Web UI. Two virtual machines (VMs) are used in the lab, with one Kali VM serving as the attacking machine and one Ubuntu VM serving as the victim. Both VMs are in the same LAN, as illustrated in Fig. 3. The task of lab exercise is to conduct an attack against the Ubuntu VM where an old and vulnerable version of Java installed. In the evaluation, the volunteers played both roles of lab manager and lab user. Thus they experienced the entire life cycle of a lab in EZSetup. A detailed lab manual was provided for both tasks. As a lab manager, a participant is first instructed to create a lab using the pre-built scenario, and then asked to deploy the lab onto the OpenStack cloud with the given cloud credentials. Once the lab is deployed, the same participant is required to log in EZSetup as a lab user, join the newly-created lab, and launch a remote attack following the lab manual. Specifically, a user has to first log into the Kali VM through SSH or web-based console and then use the social engineering toolkit (SET) to launch a browser exploit attack. By doing so, whoever accesses the vulnerable webpage hosted on the

Kali VM will be compromised, and a reverse shell will be established from the Ubuntu victim VM to the attacking Kali VM.

A survey was given out to each participant after the test. In total, 15 questions were included in the survey with 9 questions for lab manager and 6 questions for lab user. Likert scale (strongly disagree to strongly agree) is used for all questions. The survey results are presented in Figures 7 and 8.

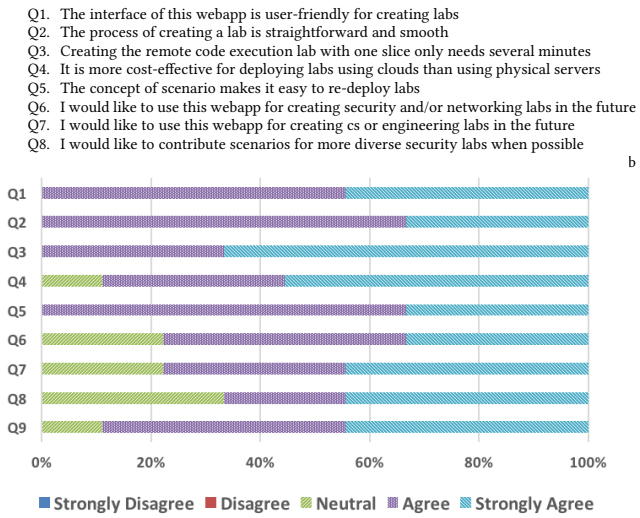

Figure 7: Results of survey questions for lab manager

From Figure 7, we can see that participants have very positive feedback about the interface, operation, efficiency of EZSetup in creating labs. In addition, the scenario concept and time-saving feature of EZSetup are widely recognized. Some participants gave neutral feedback about using EZSetup for creating security or engineering labs. A major reason is that in the test they used an existing scenario instead of creating one themselves. So they did not fully understand the capability and potential of EZSetup in other settings.

From Figure 8, we can see that the participants can complete the lab exercise smoothly and the security knowledge conveyed by the lab can be gained quickly. This is a highly positive sign given that most of the participants were security novice, having no prior experience of security practice. This survey also reveals that the participants are willing to use EZSetup in other computer science and engineering labs, which indicates that EZSetup can also be applied to other online practices other than cybersecurity ones.

6 CONCLUSION

In this paper, we have presented a new Web application called EZSetup that can create and manage virtual environments for various types of cybersecurity practices using one or more cloud computing platforms. EZSetup provides an easy and user-friendly mechanism to create virtual environments for cybersecurity practices on demand and at scale. EZSetup can be applied to both academic curriculum (e.g., security labs) and extracurricular activities (e.g., cyber defense competitions). Given its unique features and positive user feedback from experimental results, EZSetup is expected to further lower the barriers in promoting hands-on cybersecurity practices and help address the shortage of cybersecurity professionals.

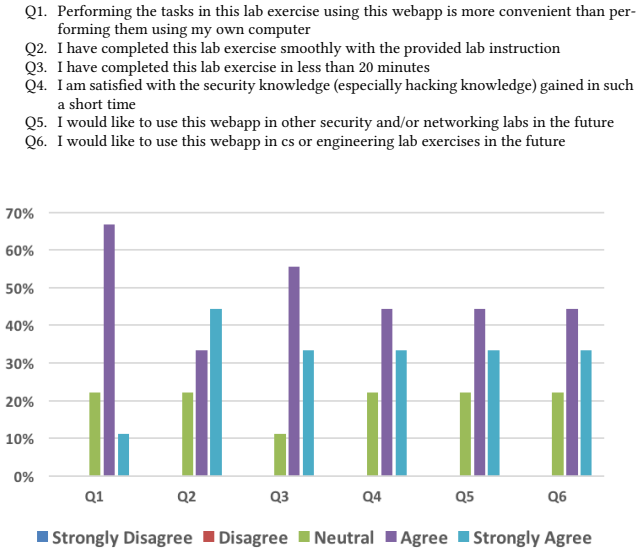

Figure 8: Results of survey questions for lab user

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation (Grant Numbers: 1338102 and 1623628), National Security Agency (Grant Number: H98230-17-1-0273), and Amazon with an AWS in Education Research grant. Development of EZSetup also used the Chameleon testbed supported by the National Science Foundation.

REFERENCES

[1] Terry Benzel. 2011. The Science of Cyber Security Experimentation: The DETER Project. In *Proc. 27th ACSAC*. 137–148.
[2] CSSIA. n.d.. National Center for Systems Security and Information Assurance. http://www.cssia.org. (n.d.).
[3] Wenliang Du and Ronghua Wang. 2008. SEED: A Suite of Instructional Laboratories for Computer Security Education. *J. Educ. Resour. Comput.* 8, 1 (March 2008), 3:1–3:24.
[4] Jim Finkle and Dustin Volz. 2015. Database of 191 million U.S. voters exposed on Internet: researcher. http://www.reuters.com/article/us-usa-voters-breach-idUSKBN0UB1E020151229. (December 2015).
[5] ISU. 2015. ISERink. http://www.iserink.org. (2015).
[6] Kif Leswing. 2016. Yahoo confirms major breach - and it could be the largest hack of all time. http://www.businessinsider.com/yahoo-hack-by-state-sponsored-actor-biggest-of-all-time-2016-9. (September 2016).
[7] Yanyan Li and Mengjun Xie. 2016. Platoon: A Virtual Platform for Team-oriented Cybersecurity Training and Exercises. In *Proceedings of the 17th Annual Conference on Information Technology Education*. ACM, 20–25.
[8] Steve Morgan. 2016. One Million Cybersecurity Job Openings In 2016. https://www.forbes.com/sites/stevemorgan/2016/01/02/one-million-cybersecurity-job-openings-in-2016. (January 2016).
[9] National Initiative For Cybersecuriy Education. n.d.. NICE Challenge Project. https://nice-challenge.com. (n.d.).
[10] Jon Oltsik. 2016. High-demand cybersecurity skills in 2017. http://www.networkworld.com/article/3152023/security/high-demand-cybersecurity-skills-in-2017.html. (December 2016).
[11] W. Sun, V. Katta, K. Krishna, and R. Sekar. 2008. V-NetLab: An Approach for Realizing Logically Isolated Networks for Security Experiments. In *Proc. USENIX CSET*. 5:1–5:6.
[12] UTSA. n.d.. CCDC. http://www.nationalccdc.org/. (n.d.).
[13] Richard H. Wagner. 2013. *Designing a Network Defense Scenario Using the Open Cyber Challenge Platform*. MS thesis. University of Rhode Island.
[14] Le Xu, Dijiang Huang, and Wei-Tek Tsai. 2013. Cloud-Based Virtual Laboratory for Network Security Education. *IEEE Trans. Educ.* 57, 3 (Oct. 2013), 145–150.