International Journal of Innovative Computing, Information and Control
Volume **14**, Number **6**, December **2018**

ICIC International ©2018 ISSN 1349-4198
pp. 2245–2258

CYBER ATTACK AND DEFENSE TRAINING: USING EMULAB AS A PLATFORM

Cheng-Chung Kuo^{1,2}, Kai Chain^{3,4,*} and Chu-Sing Yang^{1,2}

^{1}Institute of Computer and Communication Engineering
^{2}Department of Electrical Engineering
National Cheng Kung University
No. 1, University Road, Tainan City 701, Taiwan

^{3}Department of Mechanical Engineering
^{4}Department of Computer and Information Science
R.O.C. Military Academy
No. 1, Wei-Wu Rd., Fengshan Dist., Kaohsiung City 83059, Taiwan
*Corresponding author: chainkai@mail2000.com.tw

Received February 2018; revised July 2018

Abstract. *Although the convenience of the Internet has changed contemporary society, lifestyles, and interpersonal communications, it has also enabled various network intrusions that attempt to seize fame or profits through manipulating Internet functionality, computers, software, or the social weaknesses of the users. Attackers have leveraged new technologies and developed new techniques to deploy an endless array of tactics and skills. However, conventional cybersecurity education often relies on classroom teaching instead of practical tasks using real machines, because actual machine practice often provokes attacks or abnormal network traffic. This article used Emulab as a testing platform to provide a controllable environment that enabled quick deployment, adjustment, and measurement of both offensive and defensive cybersecurity experiments. Through the integration of cyber-attack and defense maneuvers within predetermined scenarios, as well as related cybersecurity questions, this article compiled data regarding the operations and responses of the participants in the exercises. This enabled analysis and improvement of the attack and defense maneuvers, the scenarios, and the platform.*

Keywords: Cybersecurity, Training platform, Emulab, Attack and defense

1. **Introduction.** With the advent of the Internet and information technologies, an increasing number of Internet-based services have become established. Numerous people have come to rely on the Internet for information exchange and communications. However, the new technologies and changes in both network technology and user habits have also enabled an increasing number of attacks with ever-changing techniques. Cyberattacks are no longer restricted to predictable techniques; a salient example of an unpredictable cyberattack is the advanced persistent threat (APT) that has become quite common in recent years [1,2]. APT refers to a clandestine, prolonged, and continuous hacking process to infiltrate a targeted entity for specific gains. ATPs do not always use cutting-edge techniques; rather, they may rely on slow-paced and varied methods to find the most effective tactics against a specific entity.

In response, defenders have also developed numerous countermeasures, such as web application firewalls [3-5], and digital forensics [6,7], and must rely on multiple security equipment to mitigate the losses and risks incurred by attacks [8,9]. Therefore, effective analysis of attack techniques and their weaknesses, as well as the development of rational,

DOI: 10.24507/ijicic.14.06.2245

adequate, and minimally resource-intensive defense mechanisms, is an imperative research focus of information security. This emphasis on effective analysis also requires investment in information security education.

The prevalence of the Internet and the growing awareness of information security have prompted numerous countries to invest in network security education. For example, in 2010, South Korea launched its "Best of the Best" cybersecurity leader training program, in which governmental agencies and corporations recruited and trained outstanding students from schools nationwide, who contributed to the country's victory in Capture the Flag (CTF) at DEFCON 2015.

The United States has also announced a cybersecurity plan for the upgrading of information security in both the public and private sectors. Part of the budget is directed toward cybersecurity training, because the government has an enormous demand for cybersecurity personnel despite their acute shortage. To alleviate this issue, the United States has established a Cyber Corps Reserve program that offers scholarships to prospective students in exchange for their service in the federal government after graduation [10]. A Cybersecurity Core Curriculum has been formulated to ensure cybersecurity graduates who wish to join the federal government have the requisite knowledge and skills. Some cybersecurity experts who join the federal workforce may qualify for student loan forgiveness programs.

Several types of information security education were proposed in 2006 [16], such as academic qualifications, professional certifications (e.g., CISSP, SSCP, CISA, GISEC, and CEH) and vendor-specific certifications (e.g., MCSE, CCSP, CompTIA, Security+, and TISIA). Previous training programs often did not include adequate hands-on training. Because network technology has advanced, these types of information security education currently include considerable amounts of hands-on training. For example, the EC-Council provides a cloud-based environment called iLab for practical hands-on training [40]. Cisco offers the CCNA Cyber Ops certification program, in which virtual machines are provided for the exercises.

One training method is hands-on operation of real machines. Trainees practice, learn, and conduct their experiments on real machines, such as personal computers, switches or firewalls. During training sessions, trainees can touch, log in to, and use actual machines. The costs of labor, time, locations, and equipment pose a great challenge to training students in attack and defense exercises on real machines.

Security concerns have motivated some researchers to conduct attack defense, and honeypot exercises through virtual machines [41,42]. Building a training environment with virtual machines to enable hands-on operations is simple. With the growth of virtualization technologies, a substantial body of security research has been conducted on virtual machines. Virtual machines cost less than actual machines and the environment can be easily rebuilt. Any state of a virtual machine can be saved as a snapshot. Because security-related training is dangerous and may be harmful to the operation system, the trainer can take a snapshot of the machine's original state and return the machine to its original state after any training session.

This paper presented a training course with easy deployment and flexible construction. Users only need to log in to the VPN server, and act like real hackers to do the security training works legally. The training course can be adjusted according to the examinee from beginner to the expert. This paper also presented a step by step penetrate testing procedure for an example and examinee can realize how a hacker can get into my system and steal the important information. In the experiment, participants are trained with the hacking skill and also the behaviors of participants are recorded during the experiment.

According to the aforementioned assertions, this article designed and implemented a cyberattack and defense exercise with flexible scenarios, and collected the participants' attack and defense operations on the platform for the further analysis. This analysis facilitated improvements to the operability and efficiency of the attack and defense platform, attack and defense tactics, and scenario designs. Regarding the structure of the present article, Section 2 reviews relevant literature and best practices. Section 3 explains the proposed platform, Section 4 discusses the scenarios for the exercise and the actual outcomes of the exercise, and recommendations for future studies, possible improvements, and the conclusions are drawn from the exercise in Section 5.

2. **Related Works.** Emulab [11-13] is a testbed that has been developed at the University of Utah since 1999. Emulab can provide unlimited experimental environments in which researchers can develop, debug and evaluate their systems. Emulab is also designed for education. Emulab can coordinate a composite system that includes numerous physical computers as nodes. Each computer has six network interface cards. The first network interface card manages user logins, power on signals, power off signals, and other control signals. The other five network interface cards are connected to a hardware switch. Emulab can manage various virtual local area networks (VLANs) and connect one or more VLANs to each computer in this system. Each user navigates to a web interface to request the resources or access the topology of the network. Emulab allocates all resources within this system. Emulab incorporates modern virtual technology into its new testbed, Cloudlab [14]. In Cloudlab, the nodes and network interfaces are generated by OpenStack [15].

Traditional training on cybersecurity mainly focuses on academic qualifications, professional certifications and vendor-specific certifications [16]. A professional takes courses and exams to obtain certifications that prove that the aforesaid professional has a particular set of cybersecurity skills. The trend has been to use certification courses and hands-on training to produce formal guarantees of cybersecurity professionalism. However, students must pay exorbitant fees to attend these types of professional training courses. Furthermore, because examinees can acquire standard solutions from previous exam questions, exams can become exercises in rote memorization rather than professional skill.

E-learning and massive open online courses (MOOCs) have effectively disseminated basic knowledge about cybersecurity. The most prominent MOOC websites offer numerous courses about cybersecurity, information security and cryptography [44-46]. In [43], the authors discussed the role of MOOCs in higher education and raised concerns regarding quality and completion rates. Similar concerns can be raised for information security education. In [22], a tutoring system was presented. Teachers were able to manage lessons, exercises, lectures and homework for students just like in a typical MOOC. However, online courses of this type always lack hands-on exercises.

Thus, numerous studies have used cloud environments with virtual machines or containers to build e-learning laboratories [17-21]. In [15], a security lab based on software-defined networks (SDNs) was presented with Cloudlab. Five SDN security experiments were presented for training. The main purpose of [15] was to encourage the students to conduct SDN security research; thus, the experiments were independent and suitable for coursework. VMs can easily be deployed in a few seconds; however, some experiments are not suitable for VMs such as distributed denial of service (DDoS) attacks and the implantation of malware into VMs [47-50].

Live exercises and hands-on training sessions are also solutions for training cybersecurity, because students who attend the courses can learn how to detect that a cyberattack is happening [23,24]. However, these types of exercises focus on the procedures of network attacks and tend to lack training regarding defense or monitoring mechanisms.

CTF [25-29] (Jeopardy) is a special information security competition akin to a war game. In the competition, numerous questions are released to the participants, who solve them with skills such as web technology, digital forensics, decompilation, and packet analysis. Points corresponding to the difficulty of the question are awarded for correct answers.

Another type of CTF is attack and defense. Each team is assigned to guard a machine or service and to attack the machines and services of the other teams. Targets can be hacked by impeding operations or gaining access authority. Students can receive the best training in both attack and defense because the exercise is run in real or emulated environments. However, these types of activities require considerable resources. In [17], the authors presented a CTF scenario generator for virtual machines. DEFCON [30,31] is presently the world's most famous hacker convention [32]; it is held annually in Las Vegas, with participants from all over the world. A participating team must pass a qualifying match known as the DEFCON CTF Qual, which is conducted in the CTF (Jeopardy) style, before proceeding to the final, in which 10 teams compete in the attack and defense style.

In the large scale cybersecurity training can be involved with several nations. Locked Shields is an international cybersecurity exercise initiated by NATO, which involves over 1500 virtualized system, over 1700 attacks and lasts over half a year from conception to conclusion [33]. Its major players are as follows. (1) Blue Teams: The Blue Teams play a leading role in the exercise, who are tasked to guard an established network with predetermined loopholes. Each of the Blue Teams is assigned one or two legal consultants. (2) Red Team: The Red Team is the antagonist that attempts to invade or disrupt the networks of the Blue Teams. The Red Team is made aware of the loopholes in the Blue networks in advance, and it is also allowed to probe loopholes in the Blue networks before the start of the exercise. (3) White Team: The White Team organizes the scenarios to serve the goals of the exercise. The White Team also develops attacking tactics and rules with the Red Team. (4) Green Team: The Green Team is in charge of the technical aspects of the exercise. (5) Yellow Team: The Yellow Team collects and analyzes information from the exercise, and delivers the latest developments to the control center.

Cyber Storm is a biennial cybersecurity exercise initiated by the United State Department of Homeland Security (DHS) in 2006 [34]. The exercise tests how well individual agencies in both public and private sectors are able to withstand attacks and ward off industrial espionage, for which purpose the participants must identify and defend hacking attacks. Over 100 agencies and organizations took part in the first Cyber Storm exercise, including the Central Intelligence Agency, National Security Agency, Microsoft, CERT Coordination Center, Federal Bureau of Investigation, US Secret Service, NORTHCOM, American Red Cross, and Public Safety and Emergency Preparedness Canada. At present, it has been held five times; a recent exercise was designed as a set of building block exercises that allowed the participants to delve deeper into particular electronic security issues, and continued to enhance the cyber incident response community's capabilities through further collaborations with the DHS. However, this kind of large scale will cost large resources for holding an event.

3. **Proposed Architecture.** The proposed cyber-attack and defense training program can be divided into the pre-training module, environment module, scoring module, and

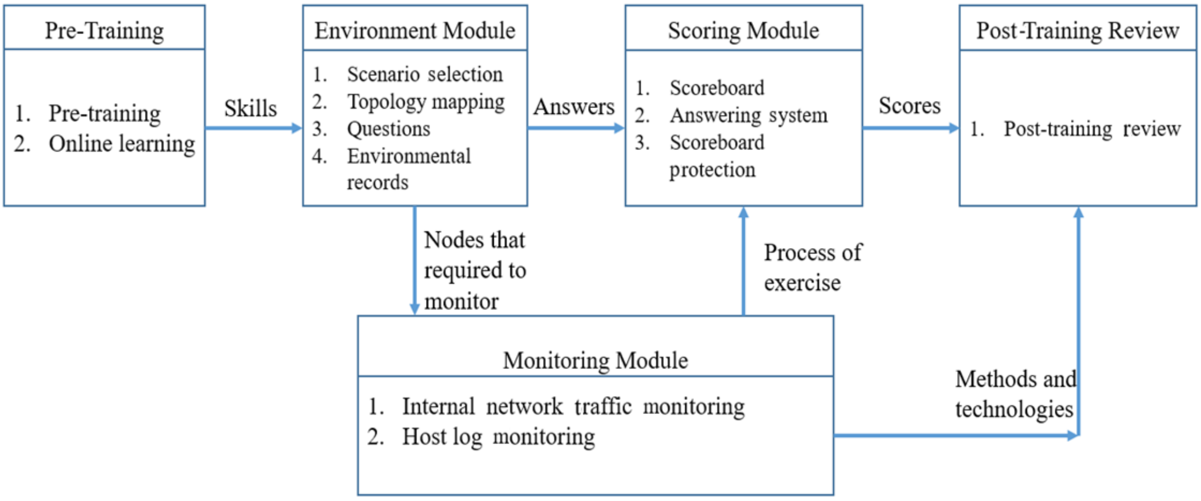


FIGURE 1. Structure of the program

the post-training review. A more in-depth description of the program follows. Based on the scenarios planned for the exercise, the program can be split into the modules shown in Figure 1.

3.1. **Pre-training.** The pre-training module is used for teaching students cybersecurity skills related to the pertinent scenarios and how to use the attack and defense platform. For example, if the scenario addresses how to use the exploit of an application to obtain system access, the teacher should provide pertinent knowledge about exploits. Students who receive such knowledge can understand the types of cyberattacks they will experience. In several scenarios, the pre-training procedure is not required, because the exercise can demonstrate the real practices that hackers follow when they must attack without sufficient reconnaissance information. Learning to think like a hacker is an important part of security training. Moreover, pre-training can be entirely completed through online learning; the users can take the exam at any time after the course. This convenient scheduling resembles the scheduling of most MOOCs.

3.2. **Environment module.** The purpose of the environment module is twofold: to configure the operational environment for the exercise (including the environmental settings, related operating systems, vulnerabilities of the host, and questions for the test) and to generate related environmental records to expedite processing of the procedures when the session is run a second time.

The main functions of this module are as follows.

1) Scenario selection: The user selects scenarios for the test, such as web infiltration attacks, host vulnerability scanning, distributed denial-of-service attacks, or other attack and defense scenarios.

2) Topology mapping: After choosing a topology layout, the user receives a default topology from the system. At this stage, the user is also allowed to add offensive or defensive nodes in the topology, lest the procedure degenerate into monotonous kit installation and instruction execution owing to a predictable topology.

3) Questions: The questions, with an emphasis on problem solving, are placed in the nodes during topology mapping as challenges for the users.

4) During each test, data related to the nodes, software, topologies, and so on are recorded by the system, so as to expedite the environment module when the user runs the session a second time.

Furthermore, the environment module also monitors the nodes in the topologies with a pre-installed agent for the test. This agent is responsible for sending the processes, services, files, and related instructions from the monitoring hosts back to the central monitoring host.

3.3. **Scoring module.** The primary role of the scoring module is to check the participants' scores and control the progress of the exercise. Participants can check the progress of their team on a designated webpage, and view the problems that are unsolved.

The main functions are:

1) Scoreboard: A web-based display interface that shows the scores and progress of the participants;

2) Question answering system: Upon obtaining the key to a question or fulfilling the objectives of the question, the participant can enter the key to the question answering system as proof, whereupon points are awarded;

3) Protection mechanism: The scoreboard is one of the most frequently attacked targets in the system because it controls the score and progress. For example, the participants may break the scoreboard's encryption to acquire the scoreboard key rather than the given questions. Hence, various protection mechanisms, such as a 60-second interval for key uploading or DDoS detection are necessary.

3.4. **Monitoring module.** The primary role of the monitoring module is to record relevant packets and logs during the test for review after the test and for use in follow-up tests as illustrated in Figure 2.

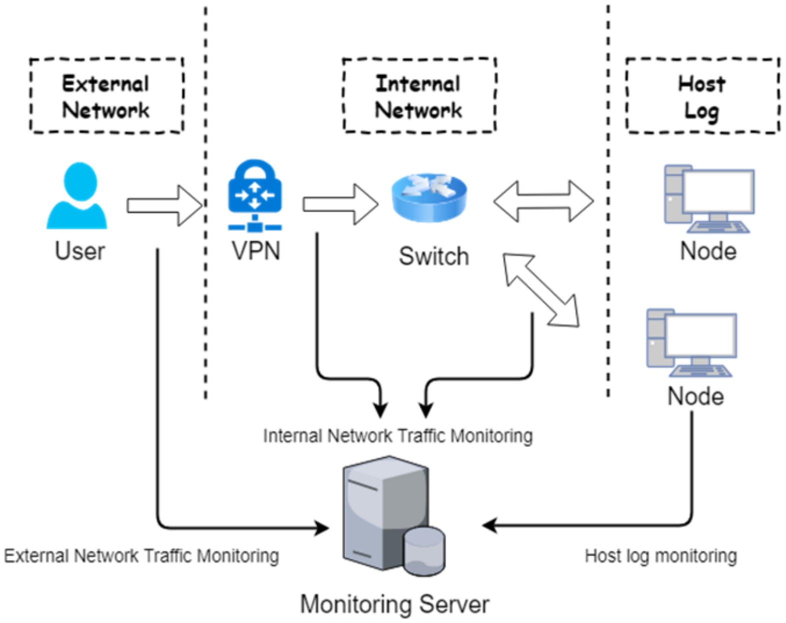


FIGURE 2. Architecture of monitoring module

The main functions are as follows.

1) External network traffic monitoring: The system records the packets transmitted between the participant's host and the platform. All packets transmitted by the participant after logging into the platform are recorded.

2) Internal network traffic monitoring: All of the communications between the nodes after the participant logs into the platform are recorded. The monitoring module reports all of the monitored packets to the central monitoring host. Any abnormal traffic is classified on the basis of predetermined rules, and if the rules must be modified on the spot

or later, the module can also be used to display traffic analysis under different rules. In the experiment, administrators must know what type of network behaviors the users are undertaking, so that the administrators can break the connection if the users' actions might harm the experimental system.

3) Host log monitoring: Host log monitoring is the set of actions required for monitoring and analyzing resources or user behavior on the hosts in the experiments. The monitoring should include system activities, logs, files/dictionary, port changes, and related items This monitoring can provide teachers with insights regarding the thought processes of students. If the host log indicates that the users did not attempt the right solution, the administrators can give hints or lower the difficulty.

3.5. **Post-training review.** The primary role of the post-training review module is for the participants to review their performance levels after the conclusion of the test, and one of the primary indicators of a participant's performance is the score. In addition, when the number of participants increases, this module can also be used to compare the methods and techniques of different participants.

Under this framework, the environment module provides questions and answers to the scoring module, which awards scores accordingly. The network traffic monitoring module is informed of the nodes and corresponding ports. The scoring module passes the scores to the post-training review module for review, and simultaneously receives process information from the network traffic monitoring module; the scoring module controls the progress of the exercise on the basis of that process information. The network traffic monitoring module sends information regarding the packets and the methods it recorded to the post-training review module, which determines what follow-up actions to take.

4. **Implementation.**

To verify the practicability of the proposed platform, a number of Taiwanese organizations were invited to conduct a training session with both attack and defense exercises and CTF competitions. The settings for both approaches are outlined as follows.

The training sessions were all conducted on the Testbed@TWISC [35] platform. Testbed@TWISC is a test platform that integrates the Emulab system authorized by the University of Utah (United States) and customized software kits and firmware [36]. It has undergone successive expansions and updates since its establishment in 2007. Using real machines to conduct cybersecurity emulations, Testbed@TWISC offers the isolation required by attack and defense training education, and can quickly deploy attack and defense scenarios, providing a topology and hosts environment for the training course. The Testbed@TWISC platform included remotely controlled nodes, allowing the users to conduct attack and defense maneuvers online, without being subjected to site restrictions [37,38].

4.1. **Scenario.** The scenario was that the participants were members of a hacker group that intended to break into the host computer of an adulterated food supplier, to steal the encrypted files within, to decrypt them, and then to publish them online. The topology and related processes of the attack and defense scenario were as shown in Figure 3.

In the initial, the users were only informed the initial environment. The scoreboard will show the webpage of the food company, and users were instructed to find some vulnerabilities in it and break into the webpage server. Once this had been achieved, the participants were entering the inside environment that infiltrate the company's Intranet and discreetly monitor the packets to extract confidential data, through which they could crack account names and passwords for the Intranet server. Subsequently, they could locate secret files in the computers of staff and bigboss, decrypt the files, and send the

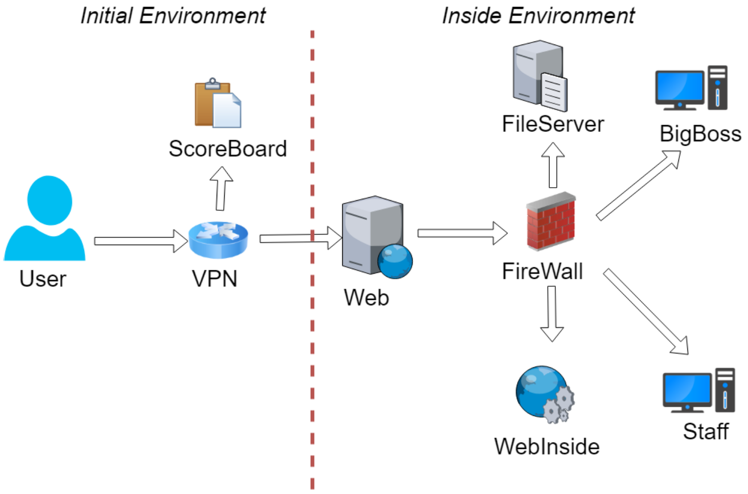
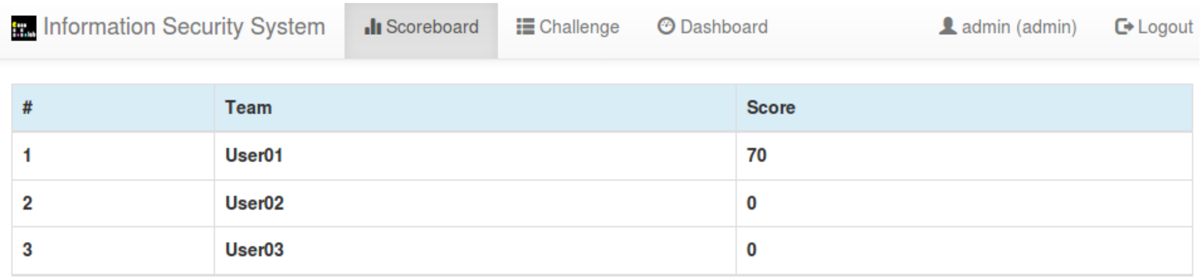


FIGURE 3. The topology of the scenario



#	Team	Score
1	User01	70
2	User02	0
3	User03	0

FIGURE 4. Appearance of scoreboard

key to the score board to get scores. This would bring the attack and defense session to an end.

4.2. **Scoreboard.** Users of the scoreboard consisted of the administrators and test participants, who employed the scoreboard in the following manners.

The administrators (admins) are key figures in charge of the whole process of attack and defense. They have full access to the system, and they are frequently updated regarding the progress of the participants. The major operations of the administrators are as follows.

Log in as "admin", and see "Scoreboard", "Challenge", and "Dashboard" options in the interface. As shown in Figure 4, the scoreboard displays the current scores of the participants (teams). It can also compare the scores with those of previous tests.

The challenge contains the questions for the test, and the administrators have access to all of them; the participants, however, must obtain a key to unlock a new question (Figure 5). To prevent any bottleneck in the exercise, the administrators are allowed to give out hints at certain milestones.

The dashboard consists of three subpages. Of the three, questions is the subpage on which all of the test questions and corresponding keys can be set (Figure 6).

To control the progress of the participants, attemption subpage can be used to list the information uploaded by the participants. If the information from a participant is found to be incompatible, the administrators can give the participant some concerned suitable hints (Figure 7).

The participants can access the scoreboard after logging in the VPN. In the scoreboard, the participants are allowed to check on the current score and test questions.

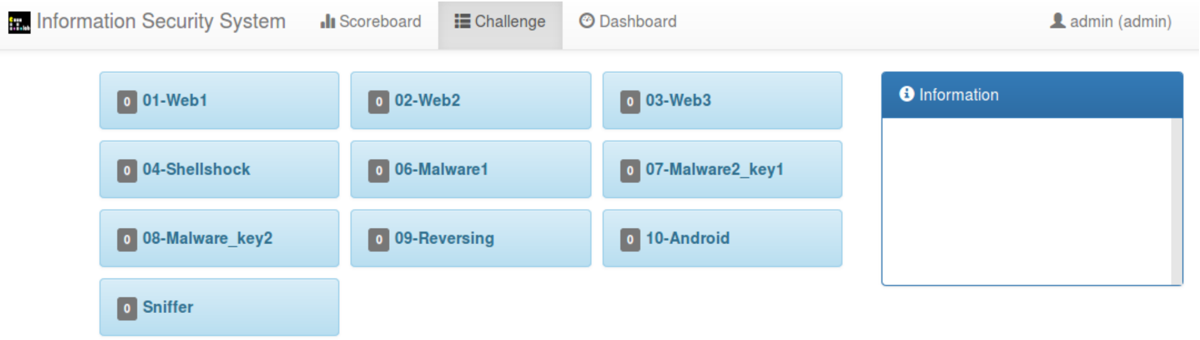
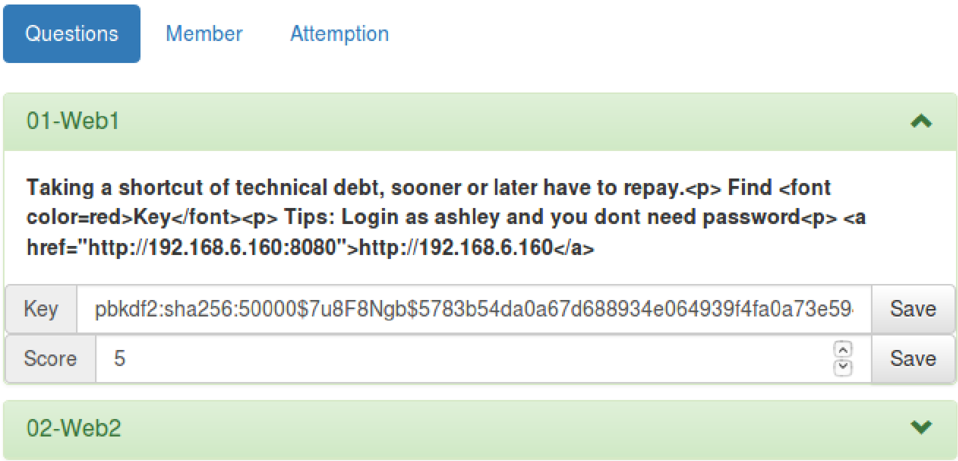


FIGURE 5. Appearance of the challenge



01-Web1

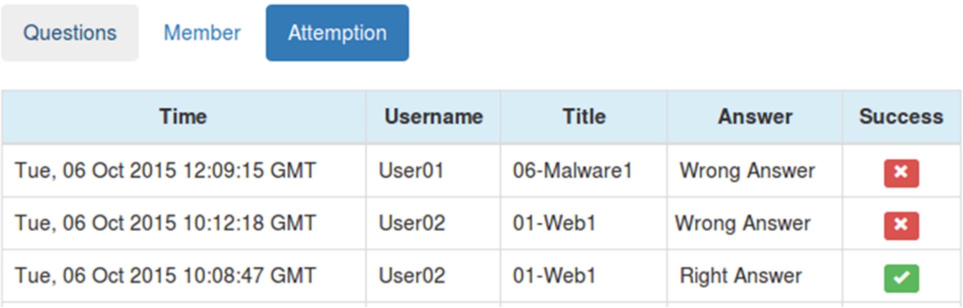
Taking a shortcut of technical debt, sooner or later have to repay.<p> Find <font color=red>Key</font><p> Tips: Login as ashley and you dont need password<p> <a href="http://192.168.6.160:8080">http://192.168.6.160</a>

Key `pbkdf2:sha256:50000$7u8F8Ngb$5783b54da0a67d688934e064939f4fa0a73e59` Save

Score 5 Save

02-Web2

FIGURE 6. Layout of the question subpage



Time	Username	Title	Answer	Success
Tue, 06 Oct 2015 12:09:15 GMT	User01	06-Malware1	Wrong Answer	✖
Tue, 06 Oct 2015 10:12:18 GMT	User02	01-Web1	Wrong Answer	✖
Tue, 06 Oct 2015 10:08:47 GMT	User02	01-Web1	Right Answer	✓

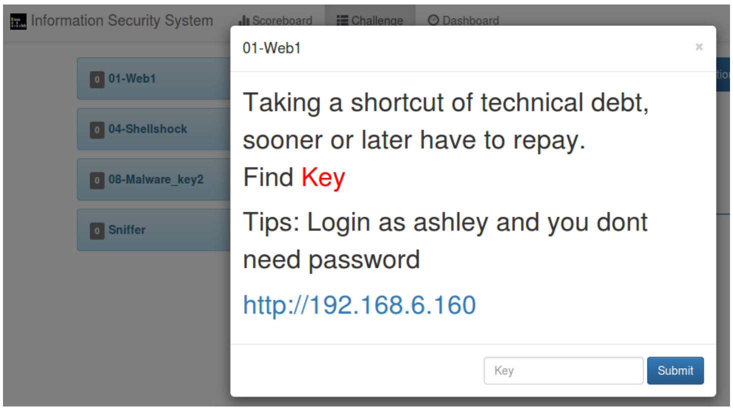
FIGURE 7. Information uploaded by the participants

Upon entering the challenge subpage, the participants are shown of the questions available to them, each of which can be displayed by clicking as shown in Figure 8.

The participants must upload the key to a question. A "failed" message will pop up if the participants fail to do so; conversely, successfully uploading the key will solve the question, earn score points, and unlock a new question.

4.3. **Network traffic monitoring.** Three tools, namely Netflow, Netflow Inside Topology, and Packet Dumping, record the operations of the participants and display it through an interface.

The Netflow tool concerns the original readings of the network traffic, and it records all the traffic of a host.



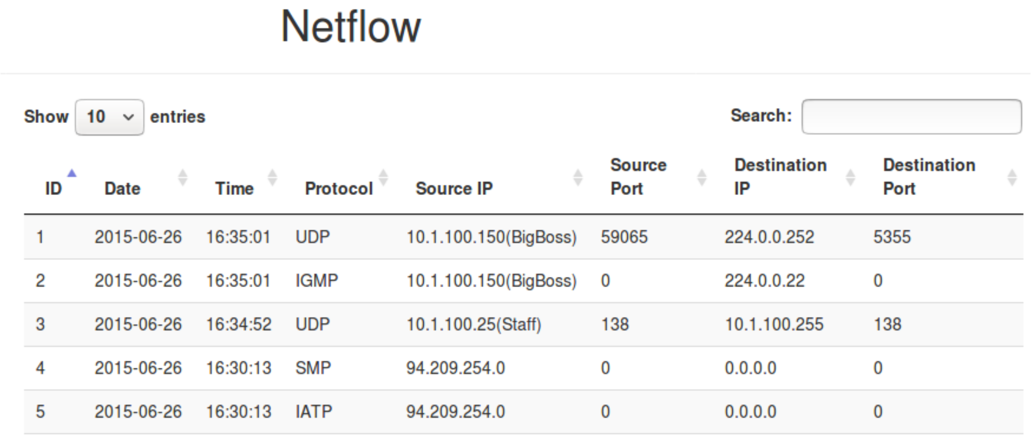
01-Web1

Taking a shortcut of technical debt, sooner or later have to repay.
Find Key

Tips: Login as ashley and you dont need password

http://192.168.6.160

FIGURE 8. Example of a question

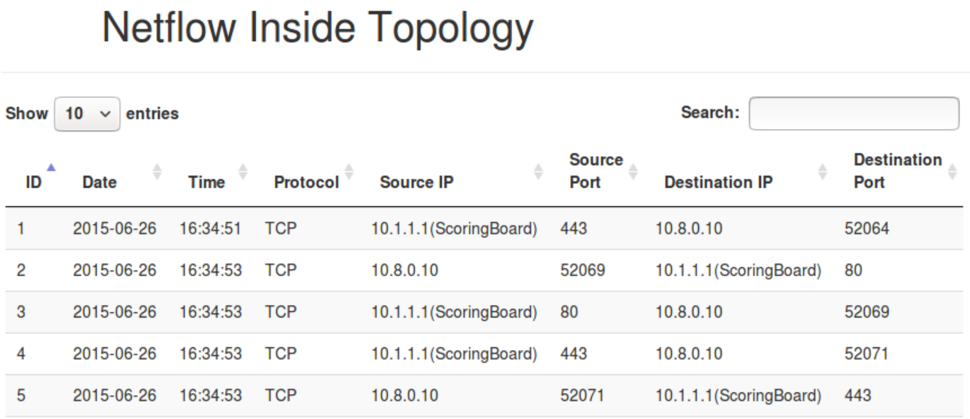


Netflow

Show 10 entries Search:

ID	Date	Time	Protocol	Source IP	Source Port	Destination IP	Destination Port
1	2015-06-26	16:35:01	UDP	10.1.100.150(BigBoss)	59065	224.0.0.252	5355
2	2015-06-26	16:35:01	IGMP	10.1.100.150(BigBoss)	0	224.0.0.22	0
3	2015-06-26	16:34:52	UDP	10.1.100.25(Staff)	138	10.1.100.255	138
4	2015-06-26	16:30:13	SMP	94.209.254.0	0	0.0.0.0	0
5	2015-06-26	16:30:13	IATP	94.209.254.0	0	0.0.0.0	0

FIGURE 9. Interface of the Netflow tool



Netflow Inside Topology

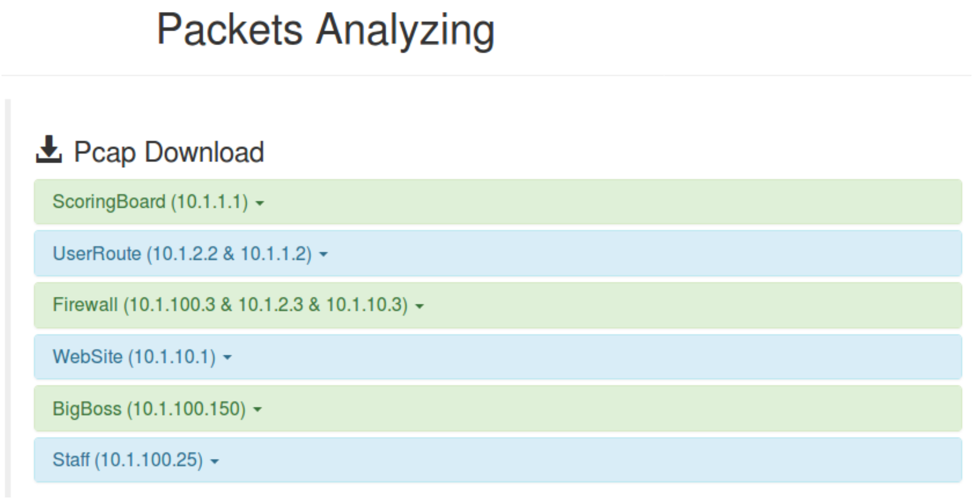
Show 10 entries Search:

ID	Date	Time	Protocol	Source IP	Source Port	Destination IP	Destination Port
1	2015-06-26	16:34:51	TCP	10.1.1.1(ScoringBoard)	443	10.8.0.10	52064
2	2015-06-26	16:34:53	TCP	10.8.0.10	52069	10.1.1.1(ScoringBoard)	80
3	2015-06-26	16:34:53	TCP	10.1.1.1(ScoringBoard)	80	10.8.0.10	52069
4	2015-06-26	16:34:53	TCP	10.1.1.1(ScoringBoard)	443	10.8.0.10	52071
5	2015-06-26	16:34:53	TCP	10.8.0.10	52071	10.1.1.1(ScoringBoard)	443

FIGURE 10. Netflow Inside Topology interface

As shown in Figure 9, all of the communications between the participants and the host are recorded by the Netflow tool.

During the test, participant traffic is directed to the topology through a virtual private network; the real traffic of the participants is recorded by Netflow Inside Topology (Figure 10).



Packets Analyzing

Pcap Download

ScoringBoard (10.1.1.1)

UserRoute (10.1.2.2 & 10.1.1.2)

Firewall (10.1.100.3 & 10.1.2.3 & 10.1.10.3)

WebSite (10.1.10.1)

BigBoss (10.1.100.150)

Staff (10.1.100.25)

FIGURE 11. Pcap download

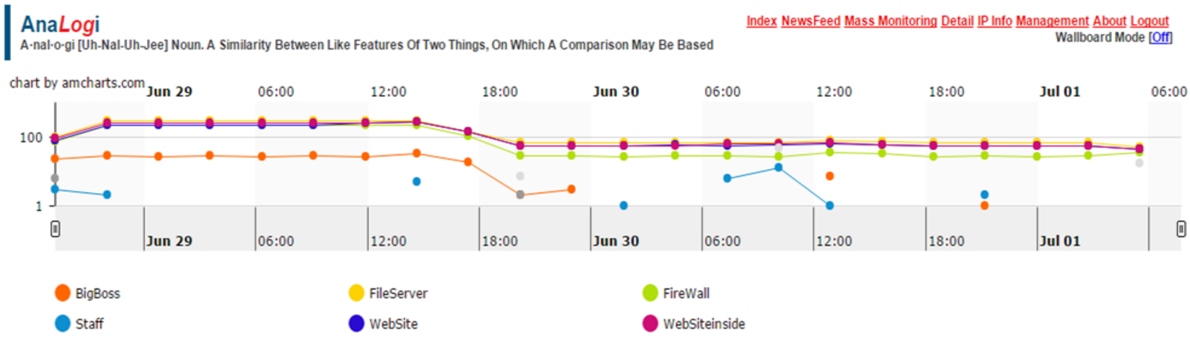


FIGURE 12. Interface of the detail screen

After the attack and defense test, if the network traffic of the test must be analyzed, the system's Pcap files can be downloaded for that analysis (Figure 11).

4.4. **Host monitoring.** The monitoring is conducted by OSSEC [39], an open-source host-based intrusion detection system that has a client installed on each host to record a log for each host system. When a monitored action matches the default rules of OSSEC, an alert is issued, and a complete log is recorded in the database.

The interface displays recent critical alerts in chronological order; the icons at the lower part of the interface represent the rules and present a brief description for each of them. Clicking on the detail button at the upper right corner reveals more details (Figure 12).

The screenshot clearly shows the attempts of participants who sought to access the web host. Aside from these actions, login and logout actions, root access applications, monitoring of ports, the addition and removal of files are also recorded and can be accessed by changing the choices in source and category. In the event that a massive amount of information is required, a comma-separated value file can also be downloaded for further analysis.

5. **Conclusion.** A platform based on Emulab was planned and designed. This platform was dedicated to training and experience exchange. The cybersecurity attack and defense exercises provided through this platform trained the participants regarding the methods, techniques, and mechanisms used in attack and defense maneuvers. Furthermore, the experiences and data obtained through the exercises are also instrumental for the testing

and improvement of the platform's practicability for cybersecurity tests and exercises. The research team generated numerous operating system image files, attack and defense databases, and related training materials from the Testbed@TWISC, and integrated them into the platform.

In the experiment, a fixed cybersecurity attack and defense scenario was used in an environment designed for a three-party network. Through this experiment, the Emulab environment of Testbed@TWISC has been proved to be capable of supporting all types of cybersecurity attack and defense exercises, so long as the scenarios used have been well planned and designed.

Additionally, testing the proposed platform revealed that to collect attack and defense data accurately under different scenarios – without hampering the attackers and defenders – is a highly challenging undertaking. Therefore, future development focus will entail an improved monitoring and data collection mechanism.

In the future, the research team will continue to collect and analyze suitable tools for the platform. Moreover, the team will pay due attention to the development of scenarios and environment settings. A future training platform based on an improved Testbed@TWISC is expected to provide adequate education and training for cybersecurity personnel.

Acknowledgment. The authors would like to thank the Ministry of Science and Technology of Taiwan, for financially supporting this research under Contract Nos. MOST 105-2221-E-145-002-, MOST 106-2221-E-145-002- and MOST 107-2218-E-006-036-.

REFERENCES

[1] H. Gao, Y. Peng, Z. Dai and H. Li, Techniques and research trends of network testbed, *The 10th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Kitakyushu, pp.537-541, 2014.

[2] P. Chen, L. Desmet and C. Huygens, A study on advanced persistent threats, in *Communications and Multimedia Security. Lecture Notes in Computer Science*, B. de Decker and A. Zúquete (eds.), vol.8735, 2014.

[3] S. Prandl, M. Lazarescu and D.-S. Pham, A study of web application firewall solutions, in *International Conference on Information Systems Security. Lecture Notes in Computer Science*, S. Jajoda and C. Mazumdar (eds.), vol.9478, 2015.

[4] D. Appelt, A. Panichella and L. Briand, Automatically repairing web application firewalls based on successful SQL injection attacks, *IEEE the 28th International Symposium on Software Reliability Engineering (ISSRE)*, pp.339-350, 2017.

[5] D. Appelt, C. D. Nguyen and L. Briand, Behind an application firewall, are we safe from SQL injection attacks?, *IEEE the 8th International Conference on Software Testing, Verification and Validation (ICST)*, Graz, pp.1-10, 2015.

[6] D. Lillis, B. Becker, T. O'Sullivan and M. Scanlon, *Current Challenges and Future Research Areas for Digital Forensic Investigation*, CoRR, 2016.

[7] SANS, *The SANS Survey of Digital Forensics and Incident Response, A SANS Whitepaper*, SANS Institute, Austin, 2013.

[8] M. M. Masud, T. Al-khateeb, L. Khan, B. Thuraisingham and K. W. Hamlen, Flow-based identification of botnet traffic by mining multiple log files, *The 1st International Conference on Distributed Framework and Applications*, Penang, pp.200-206, 2008.

[9] D. H. Jeong, B. K. Jeong and S. Y. Ji, Designing a hybrid approach with computational analysis and visual analytics to detect network intrusions, *IEEE the 7th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, pp.1-7, 2017.

[10] E. Eide, L. Stoller and J. Lepreau, An experimentation workbench for replayable networking research, *Proc. of the 4th USENIX Conference on Networked Systems Design & Implementation*, USENIX Association, Berkeley, CA, USA, 2007.

[11] D. S. Anderson et al., Automatic online validation of network configuration in the Emulab network testbed, *IEEE International Conference on Autonomic Computing*, pp.134-142, 2006.

[12] M. Berman et al., GENI: A federated testbed for innovative network experiments, *Computer Networks*, vol.61, pp.5-23, 2014.

[13] R. Ricci, Precursors: Emulab, in *The GENI Book*, R. McGeer, M. Berman, C. Elliott and R. Ricci (eds.), Springer, Cham, 2016.

[14] L. Xu, D. Huang and W.-T. Tsai, Cloud-based virtual laboratory for network security education, *IEEE Trans. Education*, vol.57, no.3, pp.145-150, 2014.

[15] Y. Park, H. Hu, X. Yuan and H. Li, Enhancing security education through designing SDN security labs in CloudLab, *Proc. of the 49th ACM Technical Symposium on Computer Science Education (SIGCSE'18)*, New York, NY, USA, pp.185-190, 2018.

[16] M. Hentea, H. S. Dhillon and M. Dhillon, Towards changes in information security education, *Journal of Information Technology Education*, vol.5, no.5, pp.221-233, 2006.

[17] Z. C. Schreuders, T. Shaw, M. Shan-A-Khuda, G. Ravichandran, J. Keighley and M. Ordean, Security scenario generator (SecGen): A framework for generating randomly vulnerable rich-scenario vms for learning computer security and hosting CTF events, *USENIX Workshop on Advances in Security Education (ASE 17)*, Vancou-ver, BC, 2017.

[18] S. Johannes, C. Willems and C. Meinel, *A Container-Based Virtual Laboratory for Internet Security e-Learning*, 2016.

[19] D. Appelt, C. Nguyen, L. Briand and N. Alshahwan, Automated testing for SQL injection vulnerabilities: An input mutation approach, *Proc. of the 2014 International Symposium on Software Testing and Analysis*, pp.259-269, 2014.

[20] R. S. Weiss et al., Teaching cybersecurity analysis skills in the cloud, *Proc. of the 46th ACM Technical Symposium on Computer Science Education*, 2015.

[21] M. S. Rubio, G. L. Civera and J. J. M. Herraiz, Automatic generation of virtual machines for security training, *IEEE Latin America Transactions*, vol.14, no.6, pp.2795-2800, 2016.

[22] A. O. Mahdi, M. I. Alhabbash and S. S. A. Naser, *An Intelligent Tutoring System for Teaching Advanced Topics in Information Security*, 2016.

[23] G. Vigna, Teaching network security through live exercises, in *Security Education and Critical Infrastructures. IFIP – The International Federation for Information Processing*, C. Irvine and H. Armstrong (eds.), Boston, MA, vol.125, 2003.

[24] G. Vigna, Teaching hands-on network security: Testbeds and live exercises, *Journal of Information Warfare*, vol.3, no.2, pp.8-25, 2003.

[25] L. McDaniel, E. Talvi and B. Hay, Capture the flag as cyber security introduction, *The 49th Hawaii International Conference on System Sciences (HICSS)*, Koloa, HI, pp.5479-5486, 2016.

[26] M. Nakaya, T. Abe and H. Tominaga, Implementation and trial practices for hacking competition CTF as introductory educational experience for information literacy and security learning, *The 5th International Conference on Informatics and Applications (ICIA2016)*, pp.57-62, 2016.

[27] A. Mansurov, A CTF-based approach in information security education: An extracurricular activity in teaching students at Altai State University, Russia, *Modern Applied Science*, vol.10, no.11, p.159, 2016.

[28] C. Eagle and J. L. Clark, *Capture-the-Flag: Learning Computer Security Under Fire*, 2004.

[29] K. Chung and J. Cohen, Learning obstacles in the capture the flag model, *USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*, San Diego, CA, 2014.

[30] C. Cowan, S. Arnold, S. Beattie, C. Wright and J. Viega, Defcon capture the flag: Defending vulnerable code from intense attack, *Proc. of DARPA Information Survivability Conference and Exposition*, vol.1, pp.120-129, 2003.

[31] E. Nunes, N. Kulkarni, P. Shakarian, A. Ruef and J. Little, Cyber-deception and attribution in capture-the-flag exercises, *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, Paris, pp.962-965, 2015.

[32] S. A. Cowan, S. Beattie, C. Wright and J. Viega, Defcon Capture the Flag: Defending vulnerable code from intense attack, *Proc. of DARPA Information Survivability Conference and Exposition*, vol.1, pp.120-129, 2003.

[33] *NATO Cooperative Cyber Defence Centre of Excellence*, https://ccdcoe.org/locked-shields-2016 .html, 2017.

[34] *Cyber Storm V: After Action Report July 2016*, https://www.dhs.gov/sites/default/files/publications /CyberStormV_AfterActionReport_2016vFinal-%20508%20Compliant%20v2.pdf, 2017.

[35] P.-W. Tsai, F. Piccialli, C.-W. Tsai, M.-Y. Luo and C.-S. Yang, Control frameworks in network emulation testbeds: A survey, *Journal of Computational Science*, vol.22, pp.148-161, 2017.

[36] *Emulab.Net – Emulab – Network Emulation Testbed Home*, https://www.emulab.net/.

[37] M. Y. Liao, J. H. Li, C. S. Yang, M. Chen, C. W. Tsai and M. C. Chang, Botnet topology reconstruction: A case study, *The 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Palermo, pp.529-534, 2012.

[38] C. Siaterlis, A. P. Garcia and B. Genge, On the use of Emulab testbeds for scientifically rigorous experiments, *IEEE Communications Surveys & Tutorials – Second Quarter*, vol.15, no.2, pp.929-942, 2013.

[39] *Home – OSSEC*, http://ossec.github.io/, 2017.

[40] *iLabs*, https://ilabs.eccouncil.org/, 2018.

[41] H. Zhang and J. Gong, Research and design of network attack and defense platform based on virtual honeynet, *International Conference on Computational and Information Sciences (ICCIS)*, pp.507-510, 2010.

[42] T. Zhang and L. Guo, Research and implementation of experimental platform for network attack and defence based on honeynet, *Advanced Materials Research*, vols.403-408, pp.2221-2224, 2011.

[43] L. Yuan and S. Powell, *MOOCs and Open Education: Implications for Higher Education*, Technical Report, JISC CETIS, 2013.

[44] *Cyberdegrees*, https://www.cyberdegrees.org.

[45] *Vouesera*, https://www.coursera.org.

[46] *Mooclist*, https://www.mooc-list.com/.

[47] X. Chen et al., Towards an understanding of anti-virtualization and anti-debugging behavior in modern malware, *IEEE International Conference on Dependable Systems and Networks with FTCS and DCC*, 2008.

[48] B. Lau and V. Svajcer, Measuring virtual machine detection in malware using DSD tracer, *Journal in Computer Virology*, vol.6, no.3, pp.181-195, 2010.

[49] M. Egele et al., A survey on automated dynamic malware-analysis techniques and tools, *ACM Computing Surveys (CSUR)*, vol.44, no.2, 2012.

[50] M. Lindorfer, C. Kolbitsch and P. M. Comparetti, Detecting environment-sensitive malware, *International Workshop on Recent Advances in Intrusion Detection*, 2011.