Cloud-Based Virtual Laboratory for Network Security Education

Le Xu, Dijiang Huang, *Senior Member, IEEE*, and Wei-Tek Tsai, *Member, IEEE*

***Abstract*—Hands-on experiments are essential for computer network security education. Existing laboratory solutions usually require significant effort to build, configure, and maintain and often do not support reconfigurability, flexibility, and scalability. This paper presents a cloud-based virtual laboratory education platform called V-Lab that provides a contained experimental environment for hands-on experiments using virtualization technologies (such as Xen or KVM Cloud Platform) and OpenFlow switches. The system can be securely accessed through OpenVPN, and students can remotely control the virtual machines (VMs) and perform the experimental tasks. The V-Lab platform also offers an interactive Web GUI for resource management and a social site for knowledge sharing and contribution. By using a flexible and configurable design, V-Lab integrates pedagogical models into curriculum design and provides a progressive learning path with a series of experiments for network security education. Since summer 2011, V-Lab has served more than 1000 students from six courses across over 20 experiments. The evaluation demonstrates that the platform and curriculum have produced excellent results and helped students understand and build up computer security knowledge to solve real-world problems.**

***Index Terms*—Collaborative learning, network security, virtual laboratory.**

I. Introduction

HANDS-ON experiments are essential when educating network security specialists. However, it is difficult for computer security education to keep pace with rapidly changing computer security issues to mimic real-world scenarios in a contained environment. This paper presents an innovative cloud-based virtual laboratory platform called V-Lab that utilizes open-source virtualization technologies such as Xen and KVM, and software defined networking (SDN) solutions such as OpenFlow switches to construct a scalable, reconfigurable, and contained experimental environment for network security education. The design of V-Lab is based on our previous work [1] with the following improved features:

1) a contained network security experimental environment providing dedicated virtual machines (VMs) and virtual networks to students;
2) a reconfigurable networking environment with the flexibility to mimic various real-world computer networks;
3) a collaborative laboratory environment with resource sharing and access control;
4) a Web portal for user-centric resource management with knowledge sharing.

V-Lab empowers a three-phase pedagogical model that is described in [2]:

- Phase I: learn basic knowledge;
- Phase II: acquire and practice skills and abilities;
- Phase III: collaborate and share knowledge.

This model is extended with six factors (*Motivation*, *Knowledge*, *Creativity*, *Collaboration*, *Demonstration*, and *Feedback*) to develop a series of progressive experiments to achieve the following pedagogical achievements:

1) a progressive curriculum that builds a private network system, from introductory to advanced levels;
2) a contained environment, in which all kinds of real-world network security experiments are performed without affecting external systems;
3) a collaborative model that encourages knowledge innovation and contribution through a Web-based social platforms and virtualized resource sharing approaches.

The rest of the paper is arranged as follows. Section II presents the related work. The V-Lab system architecture is presented in Section III. The applied pedagogical model and the corresponding case studies are presented in Sections IV and V, respectively. In Section VI, we discuss the educational results. Finally, the conclusion and future work are presented in Section VII.

II. Related Work

This section categorizes a few existing virtual laboratories for hands-on experiments as shown in Table I.

1) Virtual Application Laboratories: This type of laboratory uses desktop virtualization, in which the simulation and problem solving are restricted by predefined algorithms of the underlying software. Additionally, hands-on laboratories do not usually allow students to keep application data or states on remote servers, and as a result, this may require students to finish an experiment in a single session.

2) Shared-Host Laboratories: These laboratories are built on a fixed pool of computers with remote desktop accesses. Each computer can support several students logged in concurrently. However, a host is usually shared between multiple concurrent users at same time, which restricts the shared host to be used for different purposes. Moreover, the shared system may not

Manuscript received January 09, 2013; revised June 15, 2013; accepted July 23, 2013. Date of publication October 17, 2013; date of current version July 31, 2014. This work was supported by the NSF CCLI under Grant DUE-0942453. The infrastructure of V-Lab is based on support by the ONR Young Investigator Program (YIP) Award and the Ira A. Fulton School of Engineering, Arizona State University.

The authors are with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ 85287 USA (e-mail: Le.Xu@asu.edu; Dijiang.Huang@asu.edu; wtsai@asu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TE.2013.2282285

TABLE I
VIRTUAL LABORATORY FEATURE COMPARISON

Lab Type	Resource Sharing	Virtualization Type	Remote Access	Lab Scheduling	Reconfiguration Level	Private Data	Fidelity	Existing Labs
Physical Lab	Not sharable	None	No	Restricted time & space	Limited & expensive	No	High	[3]–[6]
Simulation Lab	Partially sharable	Application-based	Yes	No restriction	Application-level	Yes	Low	[7]–[9]
Virtual Application Lab	In-campus sharable	Application-based	Yes	Restricted time	Application-level	No	Low	[10]–[12]
Shared-Host Lab	In-campus sharable	Session-based	Yes	Reservation-based & Restricted time	OS-level	No	Low	[13]–[17]
Single-VM Lab	Cross-Campus sharable	Single-VM	Yes	Reservation-based & Restricted time	VM-level	No	Medium	[18]–[23]
Multi-VM Lab	Cross-campus sharable	Multi-VM	Yes	Unrestricted time or space	VM-level	Yes	Medium	[24]–[27]
Multi-VM & Multi-Network Lab	Multi-site sharable	Dedicated multi-VM & virtual networks	Yes	Unrestricted time or space	VM and Network-level	Yes	High	[28], [29], V-Lab

support load balancing or may not provide sufficient isolation to prevent potential performance and security issues among users.

3) Single-VM Laboratories: These laboratories provide predefined VMs (or templates) for students. A VM can be requested and released by students, or students can establish their own VMs. The single-VM approaches usually do not have a management portal that creates virtual resources customized for each user. Moreover, VMs are usually running on common desktops or laptops, and they cannot support complicated multi-VM networking environments.

4) Multi-VM Laboratories: These laboratories provide multiple VMs that can either run in the cloud [25] or on a student's PC [24]. The multi-VM environment allows students to construct complex system configurations for experiments. However, these laboratories may not provide flexible networking, sufficient isolation, or reconfiguration capacities. Usually, these features are needed to perform network security experiments. Moreover, these systems often do not support isolated inter-server communication, and thus require all VMs to reside within one physical server.

5) Multi-VM and Multinetwork Laboratories: These laboratories fully utilize the virtualization capacities of cloud virtualization capabilities to provide dedicated and contained experimental environment with multiple VMs and multiple virtual networks. The system offers a Web-based management portal for instructors and students to manage and create virtual resources in a user-friendly fashion. The virtual resources can be reconfigured throughout the course to introduce new experiments. Compared to [28] and [29], the V-Lab system provides an interactive Web GUI for network constructions and reconfigurations and experiments deployments.

III. SYSTEM ARCHITECTURE

A. Overview

The overview of the V-Lab system architecture is shown in Fig. 1. Currently, the physical V-Lab system consists of a cluster of cloud servers with high-performance capabilities and virtualization support, an HP OpenFlow switch, an array of

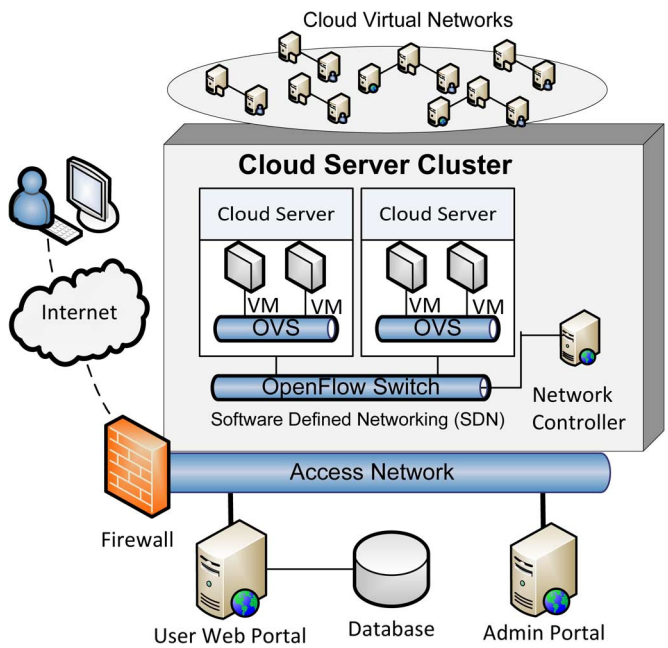

Fig. 1. Overview of V-Lab system architecture.

iSCSI storage area network (SAN) servers that provide VM storage and backup redundancy, and an uninterruptible power supply (UPS) system capable of 10 h of battery time for the entire system. The system allows up to 1000 VMs running various operating systems from Windows XP/7/Server to Ubuntu/CentOS/Redhat. The descriptions for each component follow.

B. V-Lab Front-End User Web Portal

The front-end Web portal uses a real-time visual editor on the Web site to manage the virtual resources for experiments. Instructors can drag-n-drop multiple VM hosts into the canvas and configure them as various network devices. Once the configuration is complete, it can be submitted to the back-end virtual resource (VR) engine to allow enrolled students to perform experiments.

TABLE II
THREE-PHASE PEDAGOGICAL MODEL FOR COMPUTER NETWORK SECURITY CURRICULUM DESIGN

Phase	Pedagogical Goals	Sample Topics
Teaching I (Transfer Knowledge)	• Teaching and transferring factual knowledge; • Preparing students for experiments.	• Network Types and Uses; • Network Layers and Protocol Hierarchies; • Network Services; • Cryptography.
Teaching II (Practice Knowledge)	• Demonstrating and practicing knowledge to solve real problems in a collaborative fashion.	• Firewall Packet Filtering using IPTables; • Dual-Firewall Network with DMZ; • Intrusion Detection using Snort and OpenVAS; • Man-in-the-Middle Attack with SSL-Stripping.
Teaching III (Create Knowledge)	• Creating and researching knowledge while solving challenging projects; • Defining industrial design process with security measurements and collaborative working scenarios.	• Vulnerability Scanning & Attack Graph Analysis; • Proxy-based Mobile Anti-Phishing and Anti-Virus System; • Security testing; • Penetration testing.

C. V-Lab Back End

The major components of the V-Lab back-end systems are established based on Xen Cloud Platform (XCP) and OpenStack. Both XCP and OpenStack are open-source virtual computing platforms. The V-Lab platform allows students to work with special kernels or drivers of the VMs running in the cloud system. The system also uses open virtual switches (OVS) over generic routing encapsulation (GRE) through OpenFlow protocols with a network controller, e.g., a NOX/POX network controller [30], to provide isolated virtual networking experiments. The back end also contains various internal services for administration and management purposes.

IV. PEDAGOGICAL MODEL

To achieve the pedagogical goals, the paper presents a progressive three-phase teaching model, shown in Table II, which is evaluated by six factors, shown in Table III.

Teaching I focuses on transferring basic networking and cryptography knowledge to students and preparing them for advanced experiments. During this phase, students learn to use V-Lab and to set up the experimental environment with a small number of VMs and networks that will serve as building blocks for the next phase.

Teaching II allows students to build upon the knowledge gained previously and apply it in more realistic and complex experiments. This phase requires students to work in groups and utilize various applications and techniques to build up a working solution for experimentation. The solution will be demonstrated and shared with other students during or after experiments.

Teaching III provides a list of advanced-topic projects that require students to research existing network security systems and build their own systems. During the evaluation process, students are split into groups to challenge others' systems while defending their own. This allows students to increase and consolidate their own knowledge and contribute to that of others.

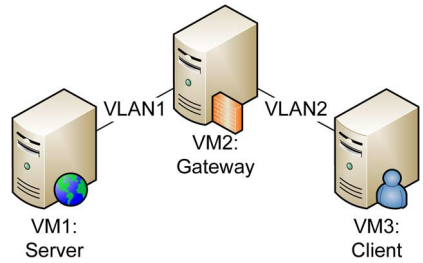

Fig. 2. Experimental environment for BNCE.

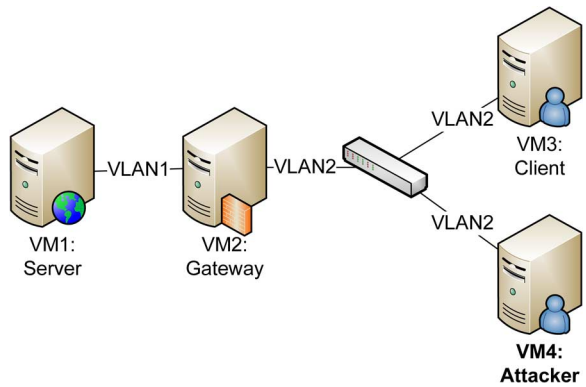

Fig. 3. Experimental environment for man-in-the-middle attack.

V. V-LAB CASE STUDIES

A. Teaching I: Basic Network Configuration Experiments (BNCE) With Knowledge Learning and Sharing

In the Teaching I phase, the V-Lab curriculum offers a series of experiments that cover a wide range of basic networking knowledge, such as using SSH and VNC to access remote hosts, configuring IP addresses, subdividing networks, using network commands such as Ping and Ifconfig, and configuring Web servers, DNS servers, and IPTables rules. In these experiments, a student is provided with three VMs interconnected with two virtual networks, as shown in Fig. 2. After each experiment, a lab teaching assistant (TA) can login remotely to each student's environment to perform grading, without requiring the student and the TA to be physically present in the laboratory. This flexible grading process allows TAs to manage their TA hours more efficiently and focus on answering questions and debugging the students' systems remotely. When the student encounters a problem during the experiment, the TA can remotely login and see the screen of the student's VMs. Using the V-Lab embedded peer-to-peer video conferencing capability, the TA and the student can work on the lab experiments remotely without needing to meet in person.

B. Teaching II: Intermediate Network Security Experiments (INSE) With Collaborations and Demonstrations

The Teaching II phase begins by expanding and reconfiguring the three-VM system from the Teaching I phase to solve more complex networking security problems, such as SSL-strip-based MITM attack (Fig. 3), IPTables-based packet filtering, and Snort-based intrusion detection. For each experiment, the V-Lab system dynamically changes the current experimental

TABLE III
SIX-FACTOR PEDAGOGICAL MODEL FOR EXPERIMENT EVALUATION AND GRADING (SAMPLE SIZE: 212)

Factors	Experiment Elements	Evaluation Methods	Evaluation Results
Motivation	Goals & usefulness of experiments	Student survey	82% found experiments useful even beyond the class.
	Tutorials, demos and samples	Student feedback	73% were satisfied with materials provided.
	Fair grading criteria and bonus points	Student survey	78% were satisfied with grading.
Knowledge	Background and factual knowledge	Quiz and exams	94% completed the exams, assignments and quizzes.
	Industrial design and development processes	Student survey	69% found experimental process similar to industrial standard.
	Setup and configure experimental environments	Checked by Teaching Assistant	96% completed the tasks on time and the environments were checked by TAs.
	Perform and practice experiment tasks	Assignments and projects	Majority were satisfied with experimental task difficulty.
	Experiment results verification and validation	Peer-evaluation, presentation	See Table IV.
Collaboration	Teamwork in group experiments	Peer-evaluation within group	94% were satisfied performing experiments in a shared environment. 82% were satisfied with teamwork and workload distribution. 77% found it easy to share environment with other groups.
	Knowledge sharing and discussion participation on V-Lab wiki and social sites	Grading on knowledge contribution, peer-evaluation	12% shared knowledge on V-Lab WiKi site.
Creativity	Innovative approaches for experiment tasks	Instructor grading, peer-evaluation	5% of materials found in students' demonstrations are innovative and contributed to the creativities of experiments. 95% of materials are from web searching and on-line knowledge base.
	Composition of research surveys and articles	Instructor grading	15% produced research articles or implementations.
Demonstration	Documentation and presentation for experiment design and results	Grading	83% of the documentation and presentations are acceptable and well-organized.
	Demonstration in class	Grading, peer-evaluation	82% found it easy to demonstrate experiment using V-Lab.
Feedback	Platform and curriculum feedback	Student survey, on-line discussion	45% encountered reliability or quality-of-service issues with V-Lab.
	Instructor and TA feedback	Student survey	93% were satisfied with support from instructor and TA.

environment by adding new VMs and VLANs to fit new requirements. In addition to discussing the knowledge base of the experiment, instructors also need to give students a comprehensive lecture about the problem at hand, such as the various attack or defense mechanisms, with appropriate references. To gain a good grade for the experiment, students must not only achieve the expected results, but also write a report presenting the techniques and applications used in the experiment.

Hands-on experiments can greatly expand students' knowledge. For instance, instructors can show that an MITM attack can only happen within a client network by assigning a new VM on the server's network and allowing students to attempt the MITM attack with the new VM. Experiments in Teaching II usually blend multiple levels of networking knowledge and thus are good test beds for practicing and exploration.

C. Teaching III: Advanced Network Security Experiments (ANSE) With Researches and Creativities

The Teaching III phase allows students to collaborate on research into real-world network system design. Students also learn how to follow a requirement-driven industrial design and development process and to construct the system with evaluation, attack model, and risk analysis. For example, some research conducted in V-Lab was published in [31] and [32].

VI. RESULTS AND DISCUSSIONS

Since the summer of 2011, V-Lab has hosted 2892 VMs to serve 604 graduate and 530 undergraduate students across six computer science and engineering courses (as shown in Table IV) in over 20 hands-on experiments. Two types of student surveys were collected from 212 of a total of 278 students, for a participation rate of 76.3%. The first survey evaluated the experiment elements (Table III), and the second one focused on each teaching phase (Table V) with satisfactory results. Compared to the same set of courses for each semester year since 2009, the V-Lab curriculum and resources helped produce more hands-on experiments [Fig. 4(a)], reduced training hours [Fig. 4(b)], and resulted in a higher completion rate [Fig. 4(c)]. The results also showed a positive feedback in Fig. 5. The paper

TABLE IV
COURSES USING V-LAB

Course Name	Description
Service-Oriented Computing and Information Management	Use C#, ASP.NET and XML and learn to develop web services, web sites and databases.
Software Security	Use Holodeck and IE6.0 to simulate various failures in a software system and test whether the software can securely handle failures.
Computer Network Security	Use Apache, DNS Server, VNC, Iptables, SSH, SSL, Snort, Syslog, OpenVAS and etc. and learn to create, configure and protect computer networks.
Information Assurance	Use Apache to configure various access control models on a website with authentication, encryption and policies.
A Web-based Document Management System (WDMS) Lab	Develop a WDMS to facilitate the management and access of all the documents of an organization.
Capstone	1-year project on mobile, cloud and security.

TABLE V
V-LAB EXPERIMENT SURVEY RESULTS (SAMPLE SIZE: 212)

Teaching Phase	Grading Criteria	Grading Results	Participation	Survey Results
Basic Experiments	• Environment Setup, configuration and basic experiment tasks • Exams and assignments	Covers basic knowledge in Cryptography, IP Address, IPTables Package filtering, IP Security, Networking Protocols, Intrusion Detection and etc.	96% finished the basic experiment tasks in time 94% completed the exams, assignments and quiz.	2% felt this phase difficult or challenging
Intermediate Experiments	• Different from traditional approaches • Documentations and presentations • Knowledge sharing and contribution • Workload distribution in the group and feedback from teammates	83% Documentations and presentations are reasonable and well-organized. 95% of materials are from web searching and on-line knowledge base. 5% of materials are innovative and creative from experiments.	76% finished the experiment tasks in time	23% felt this phase difficult or challenging
Advanced Experiments	• Creative elements • Security, Performance • Research survey and feedback	15% produced research articles or implementations	17% participated in this phase	81% found this phase difficult and challenging

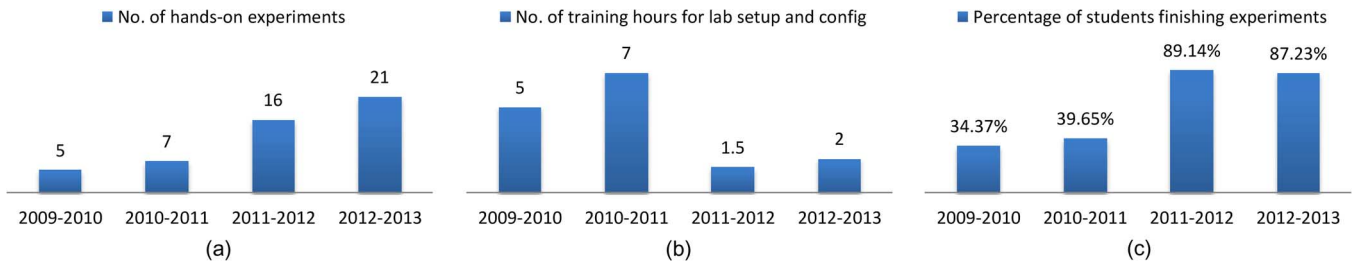

Fig. 4. Benefits of using V-Lab system: (a) increased number of hands-on experiments, (b) reduced hours for lab setup and configure, and (c) increased percentage of students finishing the experiments on time.

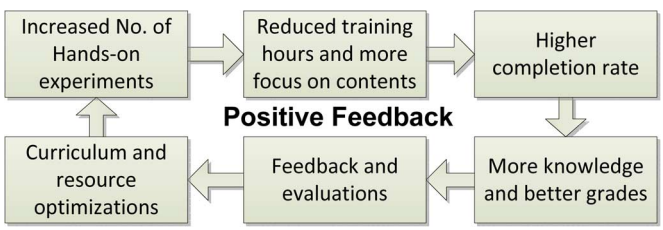

Fig. 5. Positive feedback of V-Lab benefits.

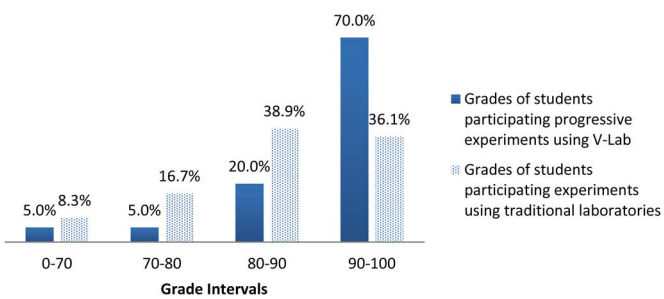

Fig. 6. Comparison of grade distributions for students participating traditional experiments versus progressive V-Lab experiments.

compares student grades from the network security course before and after using V-Lab. The results showed that students using V-Lab achieved better grades (Fig. 6), compared to traditional laboratories. Grade data for 42 students from a network security course in 2012 showed that course performance in

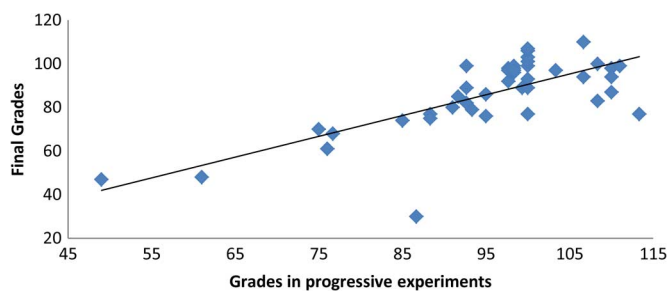

Fig. 7. Correlation between grades in progressive experiments and final grades.

progressive experiments also affected their grades (Fig. 7). Both grades were on a range from 0 to 100 plus a 20-point bonus. The Pearson correlation on these two sets of data is 0.72 and is statistically significant at the 0.01 level. Thus, students who performed better in progressive experiments also get better grades with 99% confidence.

VII. CONCLUSION AND FUTURE WORK

This paper has presented V-Lab, a cloud-based virtual laboratory education platform that provides a contained and private experiment environment for each student using the Cloud Platform and SDN approaches. V-Lab also provides an interactive Web GUI for virtual resource management and a social site for knowledge sharing and contribution. The virtual resources created can be securely accessed through OpenVPN. The system transcends the time and space limits of traditional laboratories

and provides experiments that not only allow flexible schedules, but also enable students to focus on content rather than the setting up of the environment. The system incorporates a three-phase teaching model with progressive hands-on experiments that encourage collaboration and sharing and help students gain more knowledge and better grades.

In the future, the system can deploy high-availability and redundancy features to provide a more reliable platform. By incorporating crowdsourcing from communities, the system can benefit from user-contributed curricula and feedbacks and eventually become an open crowd-learning ecosystem.

REFERENCES

[1] L. Xu, D. Huang, and W. T. Tsai, "V-Lab: A Cloud-based Virtual Laboratory Platform for Hands-on Networking Courses," in *Proc. 17th Annu. ACM ITiCSE*, 2012, pp. 256–261.

[2] P. Baumgartner and F. I. Hagen, "The Zen art of teaching communication and interactions in education," in *Proc. Interactive Conf. Comput. Aided Learning*, 2004, pp. 1–18.

[3] D. Ramalingam, "Practicing computer hardware configuration and network installation in a virtual laboratory environment: A case study," in *Proc. 37th Annu. Frontiers Educ. Conf.*, 2007, pp. F3G-21–F3G-24.

[4] Y. Liu, L. Zhang, and F. Jiao, "Teaching computer networking experiment in the realistic network laboratory," in *Proc. Int. Conf. Comput. Intell. Softw. Eng.*, Dec. 2009.

[5] T. A. Yang and T. A. Nguyen, "Network security development process: A framework for teaching network security courses," *J. Comput. Small Coll.*, vol. 21, pp. 203–209, April 2006.

[6] Rochester Institute of Technology (RIT), Rochester, NY, USA, "Rochester Institute of Technology (RIT) NSSA Labs," Apr. 2012 [Online]. Available: http://www.rit.edu/gccis/computingsecurity/

[7] L. DeLooze, P. McKean, J. Mostow, and C. Graig, "Incorporating simulation into the computer security classroom," in *Proc. 34th Annu. Frontiers Educ. Conf.*, Oct. 2004, vol. 3, pp. S1F/13–S1F/18.

[8] Y. Tateiwa, K. Kurachi, J. Zhang, T. Yasuda, and S. Yokoi, "LiNeS: Virtual network environment for network administrator education," in *Proc. 3rd Int. Conf. Innov. Comput. Inf. Control*, Jun. 2008, pp. 1–4.

[9] A. Ferrero and V. Piuri, "A simulation tool for virtual laboratory experiments in a www environment," in *Proc. IEEE Instrum. Meas. Technol. Conf.*, 1998, pp. 102–107.

[10] State University of New York, Geneseo, NY, USA, "State University of New York Geneseo Virtual Computer Lab," Apr. 2012 [Online]. Available: http://www.geneseo.edu/cit/virtual_computer_labs

[11] ASU, Tempe, AZ, USA, "ASU My Apps," Apr. 2012 [Online]. Available: http://www.asu.edu/myapps

[12] Duke University, Durham, NC, USA, "Duke University Virtual Computing Lab," Apr. 2012 [Online]. Available: http://oit.duke.edu/comp-print/labs/vcl/index.php

[13] Pennsylvania State University, University Park, PA, USA, "Penn State University Virtual Lab," April 2012 [Online]. Available: http://psbeh8b.psu-erie.bd.psu.edu/academicservices/virtuallab.htm

[14] Illinois Security Lab, Urbana, IL, USA, "Illinois Security Lab," Apr. 2012 [Online]. Available: http://seclab.illinois.edu/

[15] I. Gustavsson, K. Nilsson, J. Zackrisson, J. Garcia-Zubia, U. Hernandez-Jayo, A. Nafalski, Z. Nedic, O. Gol, J. Machotka, M. Pettersson, T. Lago, and L. Hkansson, "On objectives of instructional laboratories, individual assessment, and use of collaborative remote laboratories," *IEEE Trans. Learning Technol.*, vol. 2, no. 4, pp. 263–274, Oct.–Dec. 2009.

[16] M. Chirico, A. Scapolla, and A. Bagnasco, "A new and open model to share laboratories on the Internet," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 3, pp. 1111–1117, Jun. 2005.

[17] J. Prieto-Blazquez, J. Arnedo-Moreno, and J. Herrera-Joancomarti, "An integrated structure for a virtual networking laboratory," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2334–2342, Jun. 2008.

[18] W. Du and R. Wang, "Seed: A suite of instructional laboratories for computer security education," *J. Educ. Resources Comput.*, vol. 8, no. 1, 2008, Art. no. 3.

[19] H. E. Schaffer, S. F. Averitt, M. I. Hoit, A. Peeler, E. D. Sills, and M. A. Vouk, "NCSU's virtual computing lab: A cloud computing solution," *Computer*, vol. 42, no. 7, pp. 94–97, Jul. 2009.

[20] W. Sun, V. Katta, K. Krishna, and R. Sekar, "V-NetLab: An approach for realizing logically isolated networks for security experiments," in *Proc. Conf. Cyber Security Exper. Test*, 2008, pp. 1–6.

[21] V. J. H. Powell, C. T. Davis, R. S. Johnson, P. Y. Wu, J. C. Turchek, and I. W. Parker, "VLabNet: The integrated design of hands-on learning in information security and networking," in *Proc. 4th Annu. InfoSecCD*, 2007, pp. 9:1–9:7.

[22] University of New Mexico (UNM), Albuquerque, NM, USA, "University of New Mexico (UNM) IA Lab," Apr. 2012 [Online]. Available: http://ia.mgt.unm.edu/labintro.asp

[23] Idaho University, Moscow, ID, USA, "Idaho University VRAD Lab," Apr. 2012 [Online]. Available: http://seniordesign.engr.uidaho.edu/index.html

[24] P. Li and T. Mohammed, "Integration of virtualization technology into network security laboratory," in *Proc. 38th Annu. Frontiers Educ. Conf.*, Oct. 2008, pp. S2A-7–S2A-12.

[25] M. Wannous and H. Nakano, "NVLab, a networking virtual Web-based laboratory that implements virtualization and virtual network computing technologies," *IEEE Trans. Learning Technol.*, vol. 3, no. 2, pp. 129–138, Jun. 2010.

[26] M. Anisetti, V. Bellandi, A. Colombo, M. Cremonini, E. Damiani, F. Frati, J. Hounsou, and D. Rebeccani, "Learning computer networking on open paravirtual laboratories," *IEEE Trans. Educ.*, vol. 50, no. 4, pp. 302–311, Nov. 2007.

[27] A. Kara, E. Aydin, M. Ozbek, and N. Cagiltay, "Design and development of a remote and virtual environment for experimental training in electrical and electronics engineering," in *Proc. 9th ITHET*, 2010, pp. 194–200.

[28] C. Yan, "Build a laboratory cloud for computer network education," in *Proc. 6th ICCSE*, Aug. 2011, pp. 1013–1018.

[29] USC ISI, Los Angeles, CA, USA, U. C. Berkeley, Berkeley, CA, USA, "The DETER Project," Apr. 2012 [Online]. Available: http://www.deter-project.org/

[30] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, "NOX: Towards an operating system for networks," *Comput. Commun. Rev.* vol. 38, no. 3, pp. 105–110, Jul. 2008.

[31] D. Huang, Z. Zhou, L. Xu, T. Xing, and Y. Zhong, "Secure data processing framework for mobile cloud computing," in *Proc. IEEE INFOCOM Workshop Cloud Comput.*, 2011, pp. 614–618.

[32] C. Chung, P. Khatkar, T. Xing, J. Lee, and D. Huang, "NICE: Network intrusion detection and countermeasure selection in virtual network systems," *IEEE Trans. Depend. Secure Comput.*, vol. 10, no. 4, pp. 198–211, Jul. 2013.

Le Xu is currently pursuing the Ph.D. degree at the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA.

His research areas include service-oriented computing, network security, and education.

Dijiang Huang (M'00–SM'11) received the B.S. degree in telecommunications from Beijing University of Posts and Telecommunications, Beijing, China, in 1995, and the M.S. and Ph.D. degrees in computer science and telecommunications from the University of Missouri–Kansas City, Kansas City, MO, USA, in 2001 and 2004, respectively.

He is currently an Associate Professor with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA. His current research interests are computer networking, security, and privacy.

Dr. Huang is an Associate Editor of the *Journal of Network and System Management* and an Editor of IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. His recent research is supported by the ONR, ARO, NSF, and HP.

Wei-Tek Tsai (M'13) received the Ph.D. degree in computer science from the University of California, Berkeley, CA, USA, in 1985.

He is a Professor with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA. He is interested in service-oriented computing, education, testing and simulation and has written four books in related fields.