published as: J. Hu, Christoph Meinel: Tele-Lab IT-Security: A Means to Build Security Laboratories on the Web;
In Proceedings of the 18th International Conference on Advanced Information Networking and Applications (AINA 2004);
Fukuoka (Japan), 2004, pp. 285-288; 29-31 March 2004; isbn: 0-7695-2051-0; doi: 10.1109/AINA.2004.1283804.

Tele-Lab IT Security: A Means to Build Security Laboratories on the Web

Ji Hu, Christoph Meinel
Department of Computer Science, University of Trier, Germany
{hu, meinel}@ti.uni-trier.de

Abstract

Providing hands-on experience by live exercises is essential for current IT security education. Therefore, Tele-Lab IT security, a web-based training system, is being developed at the University of Trier, Germany. It attempts to integrate a security laboratory on the Internet using well-managed virtual machines which allow students to gain experiences of security technologies and tools in a reliable and secure way. In this paper, its user interface and architecture as well as some security considerations are described.

1. Introduction

Providing hands-on experience by live exercises is essential for current IT security education. Many universities therefore developed security laboratories, from which realistic experience of security technologies and tools become available for IT students. Facilitated by today's Internet technologies, it is not a difficult task to present security courses on the web, but to move conventional security laboratories onto the Internet is hard because many requirements for conventional laboratories are difficult to be satisfied in an open and sharable environment. E.g. conventional laboratories usually are implemented on an isolated network which allows students to exercise attacks. The dangerous operations in exercises are restricted by physical isolation from production networks. What would happen if we connect such a laboratory to the Internet and continue to allow those attacks? The other example is that system failures can be recovered manually if they take place in a conventional laboratory. How to deal with such a situation if failures take place online without human interference? In order to conquer the issues above, the University of Trier, Germany, is currently developing a web-based training system for security education. This system is called Tele-Lab IT security that familiarizes students with IT security technologies and tools. Based on virtual machine technology, it integrates a security laboratory on the web. Its well-managed virtual machines allow students to gain real experience of security technologies and tools in a reliable and secure way. Thus, both basic requirements and functions of security laboratories can be fulfilled on-line.

This paper will discuss the Tele-Lab concept and its architecture in detail. The rest of the paper is organized as follows: Section 2 discusses related work. Section 3 describes concepts about Tele-Lab. Section 4 is about how Tele-Lab is organized. Its architecture and components are described. In Section 5, some security considerations are discussed. Section 6 concludes the paper.

2. Related work

We have investigated two categories of work with similar nature. First, many projects address the development of off-line security laboratories, e.g. the U.S. Military Academy, West Point, provides an isolated laboratory for Information Assurance education [4]. In such an environment, students can familiarize themselves with computer exploits and exercise to employ technical measures to defend their network against such exploits. George Washington University has built the Portable Education Network to assist in the study of computer security [2]. The both laboratories give us security requirements for isolation and recovery of the laboratory infrastructure. However, their tasks have to be prepared and maintained by hand, and bring a heavy administration burden. Second, we try to find very close projects on online security laboratories. However, only a few simulation training systems are found. ID-Tutor [8] and the intelligent tutoring system (ITS) described in [9] familiarize users with intrusion detection. With both tools, users perform their exercises in a simulation environment. Unfortunately, for practical reasons, such simulators can model a real system only to a very limited degree. For example, a user can not apply software tools. The feedback from real procedures is impossible either.

3. Concepts

Before the Tele-Lab project, we have developed a computer-based security training system, E-learning Platform IT Security (LPF) [3]. The LPF system is

In Proceedings of The 18th International Conference on Advanced Information Networking and Applications (AINA 2004), March 29 - March 31, 2004, Fukuoka, Japan, pp. 285-288.

installed on a Linux PC and integrated with a web interface. Students can use a browser to follow its security course, and use a shell terminal, or X applications to complete real exercises (e.g. cracking passwords or encrypting email). The web server evaluates results by executing scripts to compare answers, or to examine system changes made by the students.

To facilitate tele-learning, we decided to move the LPF onto the web. The concept is described as follows. We use a main web server, called Tutor and multiple subsidiary web servers, called Lab Server. The tutor presents lecture contents and manages laboratory environments. If students are required to perform exercises, current web connections will be redirected to lab servers which offer students working environments where exercises are executed. After students finish exercises on lab servers, the web connections will be return to the tutor server. Lab servers are built with virtual machines which are software machines run on a host PC, and can be networked. In this way, the entire laboratory infrastructure can be cloned on a PC and connected to the Internet. However, here exists a risk, i.e. some exercises require students to have a privileged access for performing system-level operations. Lab servers including operating systems and file systems would be destroyed if they misuse their privilege rights. In this case, failed servers must be manually recovered using a backup copy. In addition, the tutor has to manage many connection transitions from/to lab servers, which involve complex shared authentication. So it has to take much managerial effort for in parallel running multiple lab servers.

The Tele-Lab design developed a new architecture (shown in Figure 1) which manages virtual lab server in a more effective way that eliminate that risk and improve running performance. The idea is all virtual machines are supervised. In case of a crash, the failed machine will be recovered by using a clean file system, and restarted in the background. The detection and recovery procedures are carried automatically without human interference. Also, there is no web connection transition required any more. To perform exercises, a student can login to a virtual machine using a remote desktop access tool. Meanwhile, the tutor uses a SSH connects to it and interact with the student. Considering the risk that students misuse the lab servers for compromising Internet hosts, we carefully configure its network and firewall which effectively prevent system operations from reaching production networks.

4. System architecture

The infrastructure of Tele-Lab is based on a Linux platform which provides a lot of free software and security tools. Therefore, it is a self-satisfied system that does not require additional, commercial software. Tele-Lab consists of three major components (shown in Figure 1): a web server, a virtual laboratory and user interfaces.

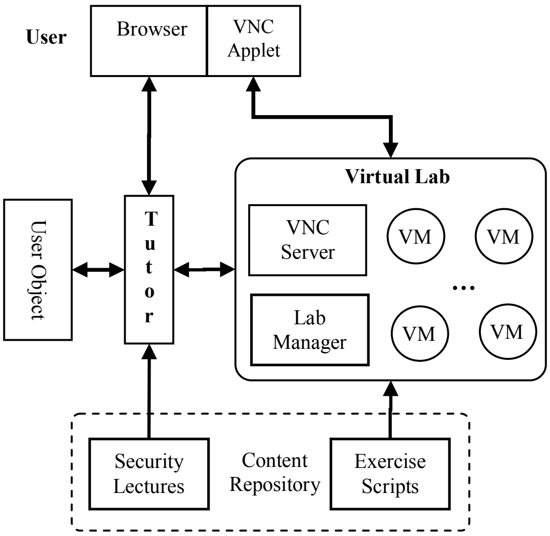

Figure 1. System architecture

1. The web server contains a tutor, a content repository and user objects. The tutor is responsible for navigation. Its tasks include managing students' learning records, presenting security lectures, preparing exercises and assigning them to students, as well as evaluating results. The content repository is an IT security knowledge base represented as a collection of web pages and scripts. A user object keeps track of a user's knowledge at every stage in a learning process.
2. The virtual laboratory implements a security laboratory which provides students with a real working environment. It consists of a virtual machine pool and a lab manager. The pool runs virtual machines which can be assigned to students. The lab manager is responsible for maintenance of the virtual machine pool. Its major tasks include monitoring virtual machines and recovering failed machines.
3. The applet-supporting web browser provides students with a uniform and user-friendly interface from which they are able to remotely access web contents and the virtual laboratory.

4.1 Web server

Content repository: it is a knowledge base that consists of three types of materials: descriptions of security concepts, descriptions of security tools, and exercise scripts. The first two types represent declarative knowledge that is presented to a student by web pages. The exercises reflect how a security task is performed in the form of scenarios that require step-by-step interactions with the student. General IT security topics (including cryptography, digital certificates, secure email, authentication and security scanning) are designed in the form of chapters, and integrated into the repository.

User object: It is used to record and analyze the individual student's performance. The user object contains three types of data: one is the personal information, such as accounts and profiles. The second records user performance, e.g. completed sections, the time spent on each section. These data can be use to analyze user performance and to present statistics on the current status. The third type of information is related to the setting of the user's virtual machine, e.g. its IP addresses and current section.

Tutor: Its main function is to present materials in the content repository in a structured manner. The tutor knows exactly which pages belong to a section and which sections belong to a specific chapter. Therefore, it can create correctly linked pages for web contents. It also decides whether a user has finished a section successfully and where to continue at the end of a section. In addition, the tutor is responsible for managing user objects and analyzing theirperformance .

Before a user start learning, he or she must register or log into Tele-Lab with a valid account. At the same time, the user object is restored from a database and his or her settings in the laboratory become ready for use. Then the tutor provides a list of available chapters from which the user can choose one. If the user is required to perform exercises, it prepares exercises, and assigns them to the user. After the user submits answers, it contacts the laboratory and evaluates results.

4.2 Virtual laboratory

The virtual laboratory plays a core role in the system. It has two functional components: a virtual machine pool and a lab manager (see Figure 1).

Virtual machine pool: The pool maintains many virtual machines which can be dynamically assigned to users. The virtual machine (VM) from User-Mode-Linux (UML) [1] provides a separated working environment with a low cost. It is run as processes in the memory on the host PC. Its file system is contained in an ordinary file on the hard-disk. It is easy to start, shutdown, and recover virtual machines because their processes can be killed and its file system can be refreshed with a backup file. Furthermore, if the file system of a virtual machine becomes small, less CPU and memory resources are needed and a host PC can accommodate more virtual machines and have a better performance, e.g. a fast recovery from a failure. Therefore, we decide to separate a virtual machine's file system into two parts: a local file system and an external file system. The very compact local file system is contained in a file which includes only a kernel and essential system files. The external file system provides needed software and user profiles and is imported from a NFS server on the host PC. In addition, each virtual machine has a remote access agent which enables a remote desktop access for a user. For the tutor, a remote execution interface, e.g. a Secure Shell (SSH) server, is run on the virtual machine. From that the tutor is able to contact user's virtual machine and manage exercises on it.

Lab manager: It is responsible for monitoring and maintaining virtual machines in the pool, as well as managing some supporting services, including NIS, NFS and E-mail services. The lab manager records virtual machines' running status (e.g. idle, assigned, crashed or restarting) and their users. It is up to the lab manager to update records in time based on monitored information. In order to monitor failures in the pool, the manager periodically tests the connections to each assigned machine. If this test passed, essential services (e.g. SSH service and the remote access service) are scanned. If all checks passed, the virtual machine is proved to be still alive. Otherwise, the lab manager asserts it failed. In this case, a recovery procedure is invoked in the background: the manager will replace its local file system with a clean copy and restart it. The interrupted exercise will continue on a new virtual machine. After a user logs out from Tele-Lab, his or her machine is released to the pool. If the tutor or a user finds the virtual machine crashed, the lab manager will be informed to directly recover it. In this way, the assignment, reclamation and recovery of the virtual machines can be carried out without any manual interference.

4.3 User interfaces

Besides a standard browser, the user interface of the Tele-Lab employs a solution from VNC [7]. VNC implements a remote access to a user's desktop. Tele-Lab installs a VNC desktop viewer on the client. It is a Java applet that is embedded in the browser. The viewer interacts with a VNC server installed on a virtual machine by a remote frame buffer protocol (RFB). This protocol is used to collect user input from the viewer, encode desktop displays, and send display information to the viewer. In this way, a thin user interface is implemented.

5. Security considerations

A user may use the Tele-Lab software or systems in incorrect way or for a malicious purpose. Also, some special exercises (e.g. security scanning or attack simulation) have to assign users privilege rights. If this system is connected to the Internet, it would become an attack station which may compromise to production network by a malicious user. Second, there is a possibility that a user internally interrupts or corrupts our infrastructure (e.g. other virtual machines). Therefore, user behaviors must be restricted tightly within the scope of Tele-Lab. This means any irrelevant connections derived from a user are not allowed to reach external networks. The user's working environment should also be isolated against internal attacks.

In order to run Tele-Lab in a secure way, a virtual machine is not allowed to be in parallel shared by multiple users. Moreover, a Netfilter firewall [6] is installed on the host PC to control network traffic. Netfilter implements packet filtering and network address translation (NAT) in Linux 2.4.x kernel. We build the virtual machine pool on a virtual network. This virtual network is configured in a special way: each virtual machine is attached to the host PC directly, instead of connected to a broadcasting network. Any traffic including those among virtual machines is routed by the host PC. The firewall on the host PC is able to control all traffic in Tele-Lab, and enforce control policy effectively. The policy includes two major firewall rules: first, the traffic between the pool and the Internet are controlled. Any connections to the pool, which are irrelevant to the VNC service, are banned. Second, any packets, whose source address and destination address are inside the virtual lab, will be dropped.

6. Conclusions

In this paper, we have presented a security training architecture which implements a security laboratory on the web. The system has the following important features. First, it offers students with a real working environment (a Linux virtual machine) instead of a simulation environment. Second, a thin user interface is implemented. The tools and programs needed in exercises are available via the browser. Third, Tele-Lab is run in a safe manner based on virtual machine technology though some privileged operations are allowed. Its administrative work is carried out without manual interference. Last, User activities in Tele-Lab are limited in a secure scope. The risk that a user misuses the laboratory infrastructure is eliminated.

7. References

[1] J. Dike, "A User-mode Port of the Linux Kernel", In *Proceedings of the 4th Annual Linux Showcase & Conference*, Usenix, Atlanta, USA, 2000.

[2] L.J. Hoffman, R. Dodge, T. Rosenberg and D.J. Ragsdale, "Information Assurance Laboratory Innovations", Present at 7th Colloquium for Information Systems Security Education, Washington, DC, USA, 2003.

[3] J. Hu, M. Schmitt, Ch. Willems and Ch. Meinel, "A Tutoring System for IT Security", In *Proceedings of the 3rd World Conference in Information Security Education*, Monterey, USA, 2003, pp. 51-60.

[4] S. Lathrop, G. Conti and D.J. Ragsdale, "Information Warfare in the Trenches: Experiences from the Firing Range", In *Proceedings of the 3rd World Conference in Information Security Education*, Monterey, USA, 2003.

[5] W. McEwan, "Virtual Machine Technologies and Their Application in the Delivery of ICT", In *Proceedings of the 15th Annual NACCQ*, Hamilton, New Zealand, 2002.

[6] Netfilter, "Netfilter and Iptables" , available at http://www.netfilter.org/, 2003.

[7] T. Richardson, Q. Stafford-Fraser, K.R. Wood and A. Hopper, "Virtual Network Computing", *IEEE Internet Computing*, Vol.2 No.1, Jan/Feb 1998, pp. 33-38.

[8] N.C. Rowe and S. Schiavo, "An Intelligent Tutor for Intrusion Detection on Computer System", *Computers and Education*, 1998, pp. 395-404.

[9] C. Woo, J. Choi and M. Evens, "Web-based ITS for Training System Managers on the Computer Intrusion", In *Proceedings of the 6th International conference on Intelligent Tutoring Systems*, Biarritz, France and San Sebastian, Spain, 2002, pp. 311-319.