

Enhance Learning through Developing Network Security Hands-on Lab for Online Students



¹Jianhua YANG, ²Thomas Reddington

¹TSYS School of Computer Science

Columbus State University

4225 University Ave

Columbus, GA 31907, U.S.A.

²New York University

Polytechnic School of Engineering

6 MetroTech Center

Brooklyn, NY 11201, U.S.A.

yang_jianhua@ColumbusState.edu, treddington@nyu.edu

ABSTRACT: *Online teaching and learning have been becoming more and more popular because students can study at anywhere and anytime. The reality is that it is hard to ask online students to conduct hands-on labs which normally require students to be on-campus. Computer network security is a course in which hands-on labs play a key role in enhancing students learning performance. In this paper we first, discuss the challenges to design online hands-on lab; second, describe the contents covered in computer network security course; third, design a lab platform which can make online students convenient to do hands-on exercise; and finally, propose eight hands-on labs which are properly designed for computer network security online students. Our survey results show that the students are satisfied by the eight hands-on labs.*

Keywords: Network security, Virtual lab, Computer Security, Online hands-on lab

Received: 4 January 2015, Revised 13 February 2015, Accepted 18 February 2015

© 2015 DLINE. All Rights Reserved

1. Introduction

1.1 Demand on Network Security Education

Network security has been one of the hottest areas in computer industry since the emergence of the Internet, especially since the fast growth of e-commerce. Expertise on this new and dynamic technology has become a necessity for many IT professionals working in government agencies, educational organizations, industry and other businesses. Some online investigations (Information, n.d.) show that the need of information security skilled workers has largely increased from 2007: 6K to 2008: 15K despite the shrinking of the whole computer job market, and the median salary for a computer security job in 2008 in Georgia is as high as 84.71K which is higher than other computer areas. It is clear that almost all the aspects of national infrastructure depend upon the correct operation of computers and networks, thus the security of these systems is imperative to the health and protection of our national infrastructure and information assets (Hill et al., 2001).

In addition, a great deal of research is called for to address the challenges existing in information assurance, and to further advance the technology. Meanwhile, many computer networks suffer from all kinds of security threats. It is a fact that current computer networks are vulnerable to a large variety of attacks, such as denial of service, malicious hackers, malicious code, session hijacking, SQL injection, and so on. Attacks resulting from these threats, if being successful, place networks at big risks. Therefore, it is necessary that IT professionals of computer network also have expertise on designing, deploying, and maintaining security mechanism to ensure computer network secured.

1.2 Challenges for Online Learning

Pedagogical research has shown that engaging students in a significant amount of hands-on exercises can greatly enhance students' learning. This is an indisputable fact especially for computer science (CS) education. For example, CS students have been engaged in coding various scales of programs, building compilers and operating systems to learn programming language, software engineering, etc. However, those experiences have not yet been extended to networking and information assurance education. Though socket programming has been a popular way of hands-on exercises to familiarize students with developing network applications, students still have not much chance to gain practical experience on transport, routing, MAC layer protocols, and the related security threats/countermeasures, which are the core of network security design.

Some online students can remember some definitions, theories and techniques to prevent computers or network from being attacked, but they do not know how to design such systems, or even deploy them. They memorize exactly from the textbook how to handle virus, yet they have no experience in writing a virus program, as well as having no idea about designing an antivirus program/system. Our student may have learned how to hijack an *http* session from text book, but they may have no chance to try it. This situation is mainly attributed to two facts: high cost on security tools and equipment, and security reasons. Most existing security-purpose labs are offered for on-site students, and need high cost equipment and/or simulation software. Our students cannot use public networks as their experimental environment because of security reasons. The unavailability of suitable online hands-on exercise has impeded undergraduate and graduate students from receiving information assurance education.

2. Online Lab Design Challenges

As the demand for online education increases, more educators have been focusing their attention on course design. One of the most important course designs is to incorporate critical hands-on lab exercises into online course. Course designers face some challenges in designing and incorporating computer network security hands-on labs.

One significant challenge faced is selecting hands-on activities to meet educational objectives. Most topics covered in computer science courses are focused on achieving a desired behavior. But in designing computer network security course, we focus on preventing undesired behavior. The ultimate objective is to secure computer network system completely. But security is an engineering trade-off which indicates that it is impossible to make a system secured against every threat. Instead, the goal would be to optimize the security of a system under given constraints, such as cost, end user usability, and information sensitivity. Instead of solely teaching how to defend a system, we design labs to offer our students opportunity to exercise how to attack a system. There is much controversy over whether to teach "how to hack" in a security course. Security is a two edged sword; offensive and defensive. To be effective at defense, students must fully understand the capabilities of a hacker and the tools they employ. So in computer network security hands-on lab design, the techniques to hack a computer host must be involved. If so, we then face another challenge as the following.

If we teach computer network security class students on exercising how to attack a host, then the host cannot be a computer being actually used in the Internet because this will break US Federal Computer Crime Laws, such as the one made in 1986 and amended in 1996, US Compute Fraud and Abuse Act which covers malicious threats, attacks and unauthorized access to computer systems. Some special purpose designed websites are exceptional, for example, the site (hack this site, n.d.) is designed for free hacking. But most sites are not designed for free hacking. So it is necessary to design a system which not only allows hacking a computer host possible, but also confine the attacking within a controlled domain which is a computer security lab environment. However, this type of lab environment is easy to set up for on-campus students, but hard for online students. So the second challenge is to set up an appropriate lab environment to allow online students to exercise hacking technique without connecting to the outside world, and not breaking the law.

The third challenge is affordability. Most universities, especially for mid-size teaching-focused ones, have tight budgets for

course offering. It is also infeasible to require online students to pay additional cost for exercising hands-on activities. To save cost, some universities proposed using Amazon EC2 (Amazon, n.d.; AWS, n.d.; Amazon Machine, n.d.) in their computer network security lab exercises (Yue et al., 2012). We found that using Amazon EC2 is a good idea for conducting labs not only for security course, but also for other computer science courses. However, EC2 still incurs some cost which depends on the hours used for application software and system software. Some other universities, such as Pennsylvania State University, University Park, PA, set up virtual laboratory system VHOL that includes a cluster of 17 Dell PowerEdge R610 machines with varying amounts of resources on each machine (Wu et al., 2014). This system obviously needs a large one time investment that may not be possible for midsize teaching-focus universities, such as Columbus State University (CSU), GA. Another part of the cost is the software needed for students to do hands-on labs. For budget reason, it is a wise choice to design hands-on labs using some software that are free of charge.

The last challenge is the network bandwidth needed if we have lab system set up on-campus. We used to have our students access a virtual lab system located inside CSU through VPN. Unfortunately lots of students complained it was too slow to access the system because most students do the labs at night time. This should be figured out by scheduling different students into different time slots. But it did not work well because most online students have full-time job and their time schedule is not flexible.

We need to design appropriate lab activities to not only reach the educational objectives, but also meet the above challenges. Before discussing our hands-on lab design, we would like to present the contents covered in computer network security offered at CSU.

3. Contents Covered

3.1 Network Recon and Scanning

This section comprehensively covers popular IP network scanning techniques that are adopted by hackers to scan and map networks. The objective is to increase our student's awareness of information security issues, and show them the procedures to break into a computer network system. Network scanning and reconnaissance is a real data gathering technique of an internet-based security assessment. It can help a hacker achieve the following goals: 1) identify accessible TCP and UDP network services; 2) assess filtering system; 3) analyze the operating system running; 4) predict the TCP sequence number of target host for TCP spoofing and sequence prediction attack potential (Beggs, 2014).

In addition to covering different techniques to gather information, determining network range, and identifying hosts used, different techniques to identify active TCP network service are also presented in detail. ICMP ping-sweeping, open TCP scanning method, stealth TCP scanning methods including half-open SYN flag scanning, inverse TCP scanning, ACK flag probe scanning, and TCP fragmentation scanning, third party spoofed TCP scanning methods including FTP bounce scanning, proxy bounce scanning, sniffer-based spoofed scanning, and IP ID header scanning (Beggs, 2014) are all covered in this section.

3.2 Vulnerabilities and Exploits

Vulnerability is defined as a software bug or misconfiguration allowing for potential unauthorized access. Identified vulnerabilities of a computer network system can be exploited to establish a persistent access to a target from attackers. Different types of scanners are introduced including ISS Internet Scanner, SAINT, Retina by eEye, and Nessus. Nessus (Open-Vas is a variation of original Nessus) is discussed in detail in this section. Three exploitation tools including Immunity Canvas, Core Impact, and Metasploit, are mentioned. In the computer network security course (course number is CPSC6128 at CSU), we focus on Metasploit and its framework. Metasploit Framework (MSF) is an attack platform of choice for one of the fastest growing careers in IT security: Penetration Testing. It contains integrated suite of tools for discovering, running, and testing exploit code.

This section discusses how to use MSF as an exploitation platform. It begins with a detailed discussion of three MSF interfaces: msfweb, msfconsole, and msfcli. We demonstrate all of the features offered by MSF as an exploitation platform. By working through real-world vulnerabilities against popular closed source applications, our students are able to learn how to use tools and MSF to quickly build reliable attacks as standalone exploits. In this section, we also explain how to integrate an exploit directly into MSF by providing a line-by-line analysis of an integrated exploit module. The Meterpreter payload system is examined as well.

3.3 Post Exploitation

This is the final stage of a hacker's attack. The purpose is to set up a persistent access to a victim. To return to an exploited system and access to services and data, normally, an attacker places a backdoor on a compromised system. In this section, we first, discuss different backdoor programs; second, present how to use and maintain persistence agent with MSF; third, discuss how to create standalone persistent agent; and finally cover network steganography.

3.4 Cryptography

In this section, we examine modern cryptography from a theoretical perspective, with an emphasis on "provable security". Cryptographic primitives are the building blocks of various cryptographic applications. We focus on exploring notions of security for a given cryptographic primitive, its various constructions and respective security analysis based on the security notion. The cryptographic primitives covered in this section include pseudorandom functions, symmetric encryption (block ciphers), hash functions and random oracles, message authentication code, asymmetric encryption, and digital signatures.

3.5 IPsec

IP datagrams have no inherent security. IP source address can be spoofed. The contents of IP datagrams can be sniffed and modified. IP datagrams can be replayed. IPsec is a method for protecting IP datagrams. It provides confidentiality, data integrity, origin authentication, and anti-replay protection services. IPsec security protocols include Authentication Header (AH) and Encapsulating Security Payload (ESP). IPsec was first introduced in Windows 2008 server. It is a set of extensions to TCP/IP used to protect network data.

3.6 SSL and IKE

IKE, the short name of Internet Key Exchange, is used to establish SAs automatically. IKE authentication is done with either PSK or PKI. IKE protocol is over-engineered. It has two phases while phase 1 is to establish bi-directional IKE SA, and phase 2 is to use IKE SA to securely negotiate the IPsec pair of SAs. SSL, the short name of Secure Socket Layer, is an application layer protocol (Alshamsi et al., 2004). It was originally designed by Netscape in 1993. SSL is mostly utilized to protect HTTP transactions, and has been used for other purposes like IMAP and POP3, etc. (Alshamsi et al., 2004). SSL is composed of handshake protocol, change cipher specification protocol, alert protocol, and application data protocol.

3.7 Layer 2 Security

OSI model was built to allow different layers to work without the knowledge of each other, but it renders low level layers affect higher level layers. This indicates if one layer is hacked, communications are compromised without other layer being aware of. Security is only as strong as the weakest layer. Layer 2 can be very weak in terms of security. In this section, different type of attacks are introduced including MAC attack, VLAN hopping attack, DHCP attack, ARP attack, and spoofing attack.

3.8 Wireless Security

With the dominance of mobile devices and the need to provide instant network connectivity, wireless networks have become the ubiquitous access point to the Internet. Unfortunately, the convenience of wireless access is accompanied with an increase in effective attacks that result in theft of access and data as well as the denial of service of network resources. In this section, we discuss and present several wireless attacks including wireless reconnaissance, bypassing MAC address authentication, bypassing a hidden service set identifier, compromising a WEP encryption, attacking WPA and WPA2, cloning an access point, and denial of service attack.

3.9 IPv6 and Security

This section introduces IPv6 and its security design. IPv6 supports many new features including increased address space, auto configuration, QoS capabilities, and network-layer security. The IPv6 Authentication Header (AH) provides data integrity and data authentication for an entire IPv6 packet. The IPv6 Encapsulating Security Payload header provides confidentiality and/or authentication and data integrity to encapsulated payload. Anti-replay protection is provided by both AH and ESP Header. These security Extension Headers may be used separately or in combination to support different security needs. The security features in IPv6 can be used to prevent various network attack methods including IP spoofing, some Denial of Service attacks, data modification and sniffing activity (Hagen, 2002; Davies, 2002).

4. Hands-on Lab Design

In order to connect security concepts covered in class to practice, meet the challenges mentioned in the above, lower the cost of

laboratory setup, and in turn enable more accessible laboratory resources, we propose establishing a virtual lab environment for each student on their own computer, and design eight hands-on labs which can be conducted on the virtual system. Virtual labs can also mitigate security concerns in using practical security attack and defense tools for educational purpose. The eight hands-on labs are 1) setting up virtualization environment; 2) refreshing Linux basics; 3) network mapping and vulnerabilities scanning; 4) host exploitation; 5) maintaining access with rootkit tool; 6) cryptography; 7) cracking windows password; 8) denial of service.

4.1 Lab Environment

The purpose of Lab 1 for students is to set up a virtual lab environment for their individual usage. We call the environment iVLab (individual virtual lab system). Each student needs to set up iVLab on their own system as shown in Figure 1 by selecting either using VirtualBox or using VMware. Each iVLab includes three virtual machines with Windows XP, BackTrack 5.0, and Fedora/Ubuntu Linux installed respectively. One more virtual machine installed as an internal router (using vSphere 5.1 for VMware or Virtual Network for VirtualBox) is used to connect the three machines together to form an internal network.

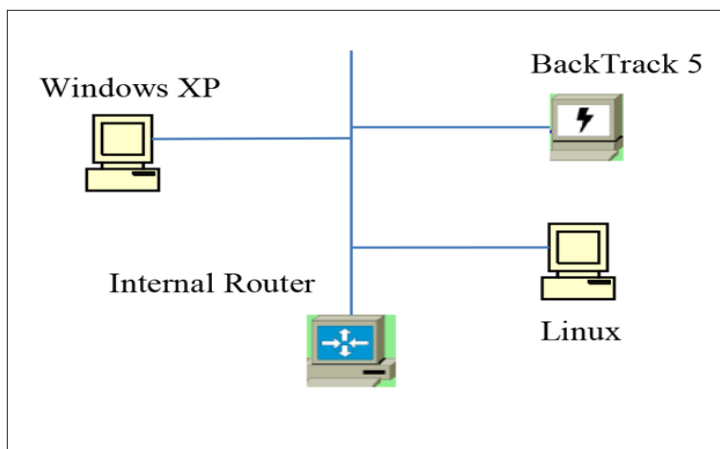


Figure 1. Hosts connected within each iVLab

Each virtual machine has an internal IP address which is assigned by DHCP. BackTrack 5 is a Linux-based penetration testing platform that aids security professionals in the ability to perform assessments in a purely native environment dedicated to hacking. Windows XP has various application installed for different hands-on labs, and is mainly used as a victim host. Linux machine has basic tools installed, such as gcc compiler. Every student has root access privilege to each machine on their own iVLab.

iVLab system can meet all the challenges posted in Section 2. It is appropriate for online students conducting hands-on exercise and allows each student to scan and attack the Windows XP machine without considering affecting the outside world. There is no network traffic issue arisen, and all the software needed is free of charge. iVLab is easy to set up and maintain. If a system is crashed during a scanning or attacking process, it is easy to restart to make the machine restored back. iVLab can be established in student's hands-on Lab1. Let's discuss other hands-on labs designed in our computer network security course CPSC6128 based on iVLab.

4.2 Hands-on Activities

4.2.1 Refreshing Linux Basics

This lab refreshes students Linux basics about file and directory management, user management, process management, communication, editing, and other miscellaneous commands. Students can use the virtual Linux machine in iVLab to complete this lab.

4.2.2 Network Mapping and Vulnerabilities Scanning

The purpose of this lab is to gain an understanding of some basic reconnaissance tools that are available to an outside attacker. Imagine that you are interested in launching an attack against some organization, and assume you have already used Whois and

found the IP address range of the target organization. Now you want to “case the joint” and gather as much information as you can about the target network including the identification of any vulnerabilities which reside on hosts within this network.

In this lab, students use Nmap at BackTrack 5 to perform reconnaissance over the whole internal network to gather needed information to launch attack over a certain target. Students also perform vulnerability scan on Windows XP host using Nessus vulnerability scanner. In order to launch Nessus, students need to create a user account in Nessus by following **Backtrack 5 → Vulnerability Assessment → Network Assessment → Vulnerability Scanners → Nessus Add User**.

Nessus is architected in a client-server model. All communications between client and server is secured using SSL. So we next need to create a digital certificate on server side. In a terminal session please type *nessus -mkcert*. Students start Nessus server by selecting: **Backtrack → Vulnerability Assessment → Network Assessment → Vulnerability Scanners → Nessus Server**.

The Nessus server loads a number of plugins. Once this process is completed, students can connect to the server using Nessus client by selecting **Backtrack → Vulnerability Identification → Nessus → Nessus Client**. Choose “connect” icon at the top of screen and leave the localhost and port as it is. Fill in the username and password you created. The client will connect to the server and your Nessus environment will be ready for use. Students can perform a Nessus scan on the Windows XP host identified by Nmap scan.

4.2.3 Host Exploitation

In the real world, networks are often compromised using vulnerabilities in software and services that they host. This also happens to be one of the easiest ways to gain access given the easy availability of exploits and frameworks that are used to quickly build one. The aim of this lab is to understand the basic exploitation techniques used by hackers & pen-testers alike and understand the post-exploitation possibilities. Using the single IP address on which students performed reconnaissance in the lab discussed in Section 4.2.2, students can exploit the host using the information gathered. Again the target should be within the IP address range of your own network.

To do this lab, students need to have a thorough understanding of Metasploit Exploit Framework. Some tools in Metasploit Framework can be used to perform authorized penetration testing, IDS signature detection, and exploit research. Students use Metasploit to compromise the single Windows XP machine, gain shell access, and transfer a file of students’ choice from the target machine to the Backtrack 5 virtual machine.

4.2.4 Maintaining Access with Rootkit Tool

The purpose of this lab is to demonstrate the final step in the network attack methodology-maintaining access. Once you have compromised a host machine as you did in the previous lab, you may want to install some form of software which will allow you to maintain persistent access to the machine. This should allow for continued access to the compromised host. Further, the software should be transparent and hidden to the end user. To achieve this goal, students need to use HackerDefender Windows Rootkit, or similar tools.

HackerDefender was written by “HolyFather” and is one of the most popular user mode rootkits for windows. HackerDefender and netcat need to be transferred to the WindowsXP Machine. Students are required to install the software using the remote exploit which has been demonstrated in the lab discussed in Section 4.2.3. In other words students must install and configure HackerDefender from a remote session. First run netcat on the windows machine and transfer a file of your choice, using netcat, from Windows/System32 directory to Backtrack5 machine. Students can choose any TCP ports that they wish for this transfer.

Using windows command “netstat”, we can show that netcat can be seen listening on the port selected. Go to Windows TaskManger and demonstrate that nc.exe (netcat executable file) can be seen running. Now configure, install and execute HackerDefender to hide the presence of nc.exe and itself from both Task Manager and netstat command. Be sure to install HackerDefense as a system service. In addition, install HackerDefender in a directory and hide that directory from normal browsing. Finally, install HackerDefender so that it starts up when the system reboots. You may do this with a Windows Registry key. Then hide the HackerDefender registry key entry so that any regular user looking at the windows registry will not see the entry which you created. Again transfer another file using netcat to verify that it is still functional

4.2.5 Cryptography

Cryptography provides protected communication over an unsecure connection. It is used to ensure that information integrity,

confidentiality, and authenticity are maintained. Cryptographic encryption can ensure the confidentiality of information, hashes can ensure the integrity of information; and cryptographic signing can guarantee the authentication of information. In this lab, first, students will be introduced how to use a tool Kryptos 2.0 to encrypt and decrypt information, as well as hashing; second, students can exercise RSA and Diffie-Helman algorithms through some computing examples. Students use Windows XP virtual machine as both sender and receiver.

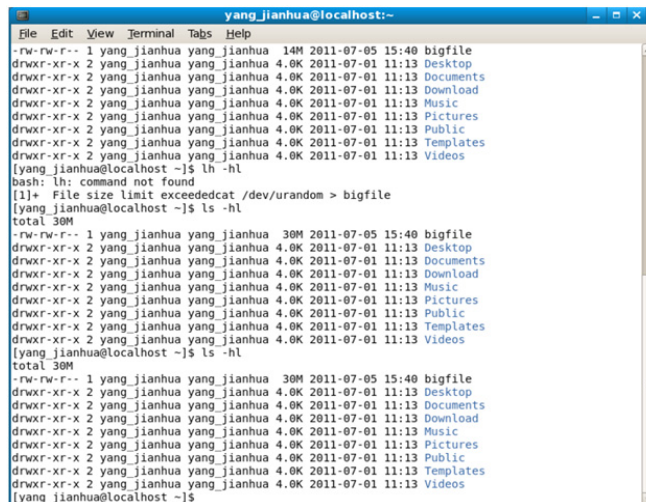
4.2.6 Cracking Windows Password

In the pursuit of gaining access into a computer or a network system, attackers will attempt to steal passwords. A password is normally encrypted and saved in a computer. Once an encrypted password is in the possession of an attacker, it is only a matter of time before it is cracked. One of the most popular Windows based programs for cracking passwords is Cain and Abel. This program comes with multiple methods for capturing and cracking passwords.

In this lab, students are asked to use Cain and Abel program to crack Windows password using Windows XP virtual machine.

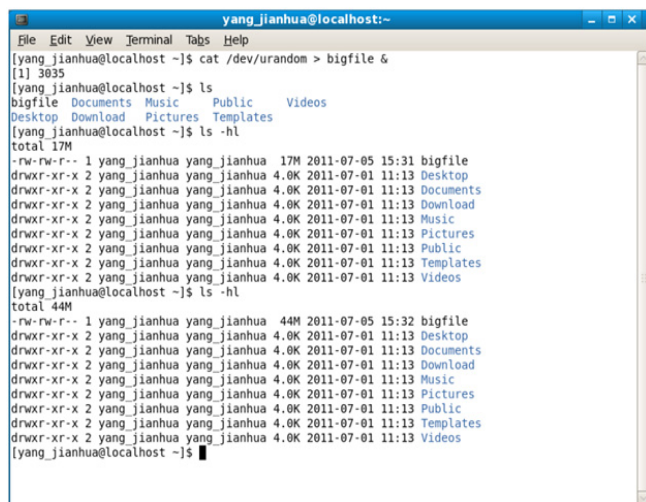
4.2.7 Denial of Service

A Denial of Service occurs whenever any legitimate user of some service is prevented from using that service. This could range from malicious attacks by some third party to exhaust your system resources through an unlimited running program or much upstream network traffic. In this lab, it has demonstrated to use a small program to exhaust their hard drive space by increasing the size of a file unlimitedly. We also show a simple approach to prevent the local resource exhaustion.



```
yang_jianhua@localhost:~$ ls -lh
-rw-rw-r-- 1 yang_jianhua yang_jianhua 14M 2011-07-05 15:40 bigfile
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Desktop
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Documents
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Download
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Music
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Pictures
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Public
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Templates
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Videos
[yang_jianhua@localhost ~]$ lh -hl
bash: lh: command not found
[!]+ File size limit exceededcat /dev/urandom > bigfile
[yang_jianhua@localhost ~]$ ls -lh
total 30M
-rw-rw-r-- 1 yang_jianhua yang_jianhua 30M 2011-07-05 15:40 bigfile
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Desktop
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Documents
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Download
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Music
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Pictures
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Public
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Templates
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Videos
[yang_jianhua@localhost ~]$ ls -lh
total 30M
-rw-rw-r-- 1 yang_jianhua yang_jianhua 30M 2011-07-05 15:40 bigfile
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Desktop
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Documents
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Download
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Music
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Pictures
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Public
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Templates
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Videos
[yang_jianhua@localhost ~]$
```

Figure 2. Display the size of bigfile with time



```
yang_jianhua@localhost:~$ cat /dev/urandom > bigfile &
[1] 3035
[yang_jianhua@localhost ~]$ ls
bigfile Documents Music Public Videos
Desktop Download Pictures Templates
[yang_jianhua@localhost ~]$ ls -lh
total 17M
-rw-rw-r-- 1 yang_jianhua yang_jianhua 17M 2011-07-05 15:31 bigfile
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Desktop
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Documents
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Download
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Music
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Pictures
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Public
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Templates
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Videos
[yang_jianhua@localhost ~]$ ls -lh
total 44M
-rw-rw-r-- 1 yang_jianhua yang_jianhua 44M 2011-07-05 15:32 bigfile
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Desktop
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Documents
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Download
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Music
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Pictures
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Public
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Templates
drwxr-xr-x 2 yang_jianhua yang_jianhua 4.0K 2011-07-01 11:13 Videos
[yang_jianhua@localhost ~]$
```

Figure 3. Verify The Size Of Big File

Students use Linux virtual machines to conduct this lab. Start your Linux and log in as an unprivileged user. This user will be referred to as user_id (in my Linux host, I use user “yang_jianhua” throughout this lab). Open a terminal at your Linux, and execute “cat /dev/urandom > bigfile &” which means random numbers are generated and write to “bigfile” continuously. The size of this file will be increased continuously. This point can be verified by execute “ls -hl” as shown in Figure 2.

The hard drive can be exhausted with the increasing of the size of bigfile. One way to limit the size of a file created by a user is to add some rules to “limits.conf”. For example, if we add “yang_jianhua hard fsize 30000” to limits.conf which means limit the size of a file created by user “yang_jianhua” as 30M.

We modify “limits.conf” and run the same command to see if the size of “bigfile” is limited. It is clearly shown in Figure 3 that the size of “bigfile” is limited by 30M.

Students are required to create a program to open multiple files and create multiple processes to exhaust the computer system.

5. Conclusion

In this paper, we analyze the challenges to design hands-on labs for online students to study computer network security. The contents covered in computer network security offered at CSU are presented. Eight hands-on labs based on the contents covered are designed to meet the challenges. The revised computer network security course has been offered at Spring 2014 for graduate students at CSU, and the hands-on labs have been adopted for the class. From our survey, we obtained that more than 90% of the students were satisfied with the new contents and hands-on labs. Most of the students mentioned that they could learn some real-world hacking techniques, and this is helpful for them to design more secured computer program/system, as well as defending their systems. One issue that students complained is that they need to have a computer with more memory installed to support four virtual machines in iVLab system. Based on the survey, most students suggested the memory size of the host computer system is at least 8G which is not standard configuration for some old computer system. The students who could not set up their own iVLab system on their own computer were suggested to use VLab supported by Poly Engineering School of New York University to complete their lab assignments.

6. Citations and References

- [1] Information Security Job Market Overview, http://www.odinjobs.com/Information-Security_job_market_overview.html
- [2] Hill, J. M. D., Curtis, J., Carver, A., Humphries, J. W., Pooch, U. W. (2001). Using an isolated network laboratory to teach advanced networks and security. In *SIGCSE'01: In: Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education*, p 36-40, New York, NY, USA, ACM Press, 2001.
- [3] Amazon Elastic Compute Cloud (Amazon EC2), <http://aws.amazon.com/ec2/>
- [4] AWS in Education, <http://aws.amazon.com/education/>
- [5] Amazon Machine Images (AMIs), <http://aws.amazon.com/amis>
- [6] Hack this site, <https://www.hackthissite.org/>
- [7] Yue, C., Zhu, W., Williams, G. L., Chow, E. (2012). Using Amazon EC2 in Computer Network Security Lab Exercises: design, results, and analysis. In: *Proceedings of the ASEE Annual Conference and Exposition*. American Society for Engineering Education, 2012.
- [8] Wu, D., Fulmer, J., Johnson, S. (2014). Teaching Information Security with virtual Laboratories. *Innovative Practices in Teaching Information Sciences and Technology*, p 179-192, Springer International Publishing Switzerland, 2014.
- [9] Beggs, R. W. (2014). Mastering Kali Linux for Advanced Penetration Testing, Packet Publishing, 2014. <http://www.safaribooksonline.com/library/view/mastering-kali-linux/9781782163121/ch09s05.html>.
- [10] Alshamsi, A., Saito, T. (2004). A Technical Comparison of IPsec and SSLb, <https://eprint.iacr.org/2004/314.pdf> Hagen, S. (2002). IPv6 Essentials. Sebastopol: O'Reilly & Associates, Inc.
- [11] Davies, J. (2002). Introduction to IP Version 6. Microsoft Word Version. February 2002, <http://www.microsoft.com/windows2000/techinfo/howitworks/communications/nameadmgmt/introipv6.asp>.



Jianhua Yang earned his Ph.D. degree in computer science from University of Houston, Houston, TX USA at 2006. He is currently working at TSYS School of Computer Science, Columbus State University (CSU), Columbus, GA USA as an Associate Professor. Before joining CSU, he was an Assistant Professor at Bennett College for Women from 2006 to 2008, University of Maryland Eastern Shore from 2008 to 2009, and Associate Professor at Beijing Institute of Petro-Chemical Technology, Beijing, China from 1990 to 2000. His current research interests are computer network and information security.



Thomas Reddington earned his MS degree in Physics for the University of Pittsburgh, Pittsburgh PA in 1974. He is currently working at New York University (NYU) Polytechnic School of Engineering as an Industry Professor of Computer Science. Before joining NYU he was the Director of Security Research at Bell Labs, Alcatel-Lucent Technologies. His current duties include teaching of Security and Programming Language classes, as well as being the Director of the Cybersecurity Program at NYU.