Available online at www.sciencedirect.com
ScienceDirect
Computer Communications 30 (2007) 2032–2047
computer communications
www.elsevier.com/locate/comcom

Using the vulnerability information of computer systems to improve the network security

Yeu-Pong Lai *, Po-Lun Hsia

Department of Computer Science and Information Engineering, Chung Cheng Institute of Technology, National Defense University, Tauyuan 33509, Taiwan, ROC

Received 20 September 2006; received in revised form 7 March 2007; accepted 8 March 2007
Available online 13 March 2007

Abstract

In these years, the security problem becomes more important to everyone using computers. However, vulnerabilities on computers are found so frequently that system managers can not patch up all these vulnerabilities on hosts within the network in no time. They need to perform a risk evaluation in order to determine the priority of patching-up vulnerabilities. Besides, they may not have the administrator right on all hosts in the network, but only have the right on these network devices. To keep these vulnerabilities on hosts from exploitation, system managers can set the ACL scripts on network devices. The solution improves security in the network immediately, since some threatened service ports on hosts are blocked from accessed. This paper introduces a method to improve the network security, which consists of the network management, the vulnerability scan, the risk assessment, the access control, and the incident notification. Companioned to the network topology, the risk evaluation indicates the threatened service ports that should be blocked within ACL scripts. These procedures do not cost any extra hardware equipment. With the proposed method, the network security improves almost 40% with only 8% of threatened ports being blocked in the examined Class-B network. The 40% improvement of network security is evaluated with these two indices, the summary of CVSS values and the number of vulnerabilities in the network.

Keywords: Vulnerability; Network topology; Risk evaluation; Access control; Security

1. Introduction

In recent, the network technologies have been applied to many areas, such as tax payments, electronic auctions, electronic commerce, electronic voting. These application services consist of the network devices and the computer hosts. It is very important to protect these application servers and network devices from data tapped or counterfeited by malicious attackers. To guard against these malicious attackers, some commercial hardware and software are designed, such as the firewall systems, the intrusion detection devices, the virus protection software, the vulnerability scanning software, and so on. However, the usage of these hardware and software can not guarantee computer systems against all attacks. According to the report of President's Information Technology Advisory Committee (PITAC) in February 2005, the computer network protocols were designed in the idea for communications between trusted partners [1]. In other words, the security properties were not considered at that time, so that some kinds of attacks might spread out soon via the Internet, such as Distributed Denial of Service (DDOS), the computer virus – Code Red [2], or the computer worms – Slammer [3]. The wider the damage is spread, the more cost it is. As a matter of facts, the damages might result from a single computer on which the software has a vulnerability exploited. It is therefore very important to eliminate the vulnerability exploitation.

The number of exploited vulnerabilities increases dramatically, according to the statistic from the reports of

[*] Corresponding author. Tel.: +886 3 30805249 212; fax: +886 3 6894770.

E-mail addresses: lai@ccit.edu.tw (Y.-P. Lai), shaya.tw@gmail.com (P.-L. Hsia).

0140-3664/$ - see front matter © 2007 Published by Elsevier B.V.
doi:10.1016/j.comcom.2007.03.007

Computer Emergency Response Team/Coordination Center (CERT/CC). The number of found vulnerabilities was from 345 to 5990 in the decade of 1996–2005 [4]. The number of events reported to CERT/CC was 2573 in 1996. In 2003, it was in an astonished number of 137529 [5] (see Fig. 1).

The "victor" in the cyber-war between the system mangers and malicious attackers is the one who can obtain and use the information of vulnerability in computer systems. The "Vulnerability Management System" is therefore introduced in the computer system security management. The information from analyzing the vulnerabilities of computer systems makes the system managers know how to protect the network from known intrusion or infection attacks. The vulnerability management performs system security checks as new vulnerabilities announced or as new hosts installed after a period of time. After these systems with vulnerabilities are patched, the systems are secure from attacks. This patching procedure is surely time-consuming. Besides, some of vulnerabilities may not induce the cost damage. The system managers may like to patch these serious vulnerabilities only rather than others. System mangers should scan intra-network regularly. After the system scanned, system managers may have the following problems.

(1) Which vulnerability is the fist I should patch?
(2) Which host will cause much trouble?
(3) Where are these hosts with vulnerabilities?
(4) Have I the right to patch and manage these hosts?
(5) Is the patched host still good to all application software?

Actually, the above questions are not noticed for most system mangers for a small company or organization. They are only in passive behavior, waiting for users' complaints and then to solve. For solving users' problems, they look up these vulnerability databases announced in some famous Web sites. After the vulnerability information is found, they try to find the patch-up program or setup the threatened hardware or software properly. To some vulnerabilities, the databases only have provided the characteristics but no solution procedure yet. System managers then have no way to solve these problems. This paper introduces a way to evaluate the risk levels of hosts' being intruded and infected. According to the risk level on each host, the system manger orders a list of hosts on which patching processes are required, firstly. Secondly, managers use access control technique on network devices to segregate these hosts with vulnerabilities from others. Managers then notify the owners of these hosts about the vulnerabilities. After that, managers start to patch these hosts on which they have the administrator right. The network security is therefore improved.

The next section introduces the standard vulnerability information announced by some famous organizations. Besides, the background knowledge of network security technologies is also provided in this section. The suggestion of consolidating the vulnerability information is shown in Section 3. Section 4 is then for experimental results and discussions. Finally, the conclusions are given in Section 5.

2. Vulnerability information

There are many network attack events happened in these years by exploiting vulnerabilities. These vulnerabilities might be in the operation system, the application software, the weakness on the computer server hardware, in-appropriate settings of authentication on the system, or misusing of users. The effects of vulnerabilities make the systems be accessed, modified, shut-down out of the security policy in organizations. Since the vulnerabilities result from so many matters, they are very complicated to be defined and classified. This section shows the introduction of vulnerabilities, the classification of vulnerabilities, the way to find vulnerabilities, the evaluation of vulnerabilities, and the analysis of vulnerabilities, in distinct subsections.

2.1. The introduction of vulnerabilities

The computer vulnerabilities result from the flaws in designing the software, hardware and firmware, or even from the wrong configuration or setting on these systems. The vulnerabilities lead to unpredicted results that may reduce the performance of systems or damage the data on systems. In some cases, the "sensitive" data may be

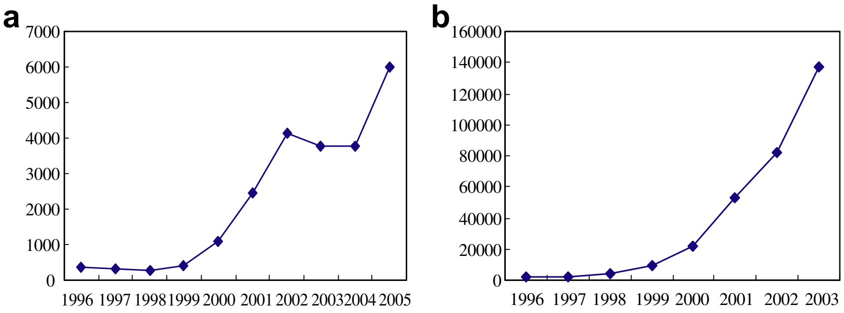


Fig. 1. (a) The number of found vulnerabilities (b) the number of reported events.

Table 1
The vulnerability with reference number of CVE-2006-0010 [8]

Name	CVE-2006-0010 (under review)
Status	Candidate
Description	Heap-based buffer overflow in T2EMBED. DLL in Microsoft Windows 2000 SP4, XP SP1 and SP2, and Server 2003 up to SP1, Windows 98, and Windows ME allows remote attackers to execute arbitrary code via an e-mail message or web page with a crafted Embedded Open Type (EOT) web font that triggers the overflow during decompression.
References	MS:MS06-002 URL: http://www.microsoft.com/technet/security/bulletin/ms06-002.mspx

revealed or duplicated via taking the advantage of the vulnerabilities on these systems. Vulnerabilities affect the systems in a different way with different characteristics. To identify vulnerabilities, each found vulnerability is labeled with a reference number. Some organizations label the same vulnerability with different numbers. To avoid confusion, MITRE established a standard numbering method that is named CVE (Common Vulnerabilities and Exposures) number that indexes to vulnerability information [6]. In the standard format, the information is stated in "Name", "Status", "Description", and "Reference". An example is shown in Table 1. Till now, the vulnerabilities, of 236 products provided by 143 organizations in 21 countries, are labeled with the CVE numbers. The vulnerability database NVD (National Vulnerability Database) of NIST (National Institute of Standards and Technology) is also indexed with CVE numbers [7]. Therefore, the CVE number system is selected in this article.

2.2. The classification of vulnerabilities

Due to the vulnerabilities result from the design flaws or configuration errors in computer hardware, software and firmware, the types of vulnerabilities are divergent. Moreover, the meanings and classifications of vulnerabilities might be different for different research topics. The classification for vulnerabilities are however to simplify the way in understanding the characteristic of certain vulnerabilities. In Taiwan, researchers in the organization, TWCERT/CC, classify the vulnerabilities in two ways [9]. One is based on the effects of the vulnerabilities, illustrated in Table 2. The other is for the service resources on which the vulnerabilities occur, tabulated in Table 3. The service resource view concerns the flaw services provided on the systems cannot perform correctly.

2.3. The software for finding vulnerabilities

This section introduces the software for finding vulnerabilities on hosts. The vulnerability scan is for users understanding how "healthy" their hosts are. To find the known vulnerabilities on these hosts, users can apply scanning tools. Two most well-known tools are introduced in this section, which are "Nmap" and "Nessus". According to the scanned results, system administrators can patch up the systems if the patch programs have been provided. To the vulnerabilities without the patch program provided, system administrators can block inappropriate accesses by the applying personal firewall or monitor related processes for bewaring of abnormal executions. A vulnerability scanning tool performs with its vulnerability database in which these found computer vulnerabilities are specified. While vulnerability is found, the database is expanded by adding the signature of the vulnerability. Users should update the database before they perform the scanning tool to obtain the correct results.

Table 2
Classification according to effects of the vulnerabilities in TWCERT/CC

Types	Description
Effects of the vulnerabilities	
Back doors	The vulnerabilities can be used to open a back door on the system
CGI abuses	The vulnerabilities in Web applications
CGI abuses: XSS	The vulnerabilities cross site script
CISCO	The vulnerabilities on CISCO network devices
Default Unix accounts	The vulnerabilities about Unix default accounts
Denial of service	The vulnerabilities about DOS
Finger abuses	The vulnerabilities about the Finger operation
Firewalls	The vulnerabilities on the firewalls
FTP	The vulnerabilities about FTP services
Gain a shell remotely	The vulnerabilities to gain a shell remotely
Gain root remotely	The vulnerabilities to gain root remotely
General	The vulnerabilities in general problems
Miscellaneous	The vulnerabilities in other aspects
Netware	The vulnerabilities on Novell Netware
NIS	The vulnerabilities in Network Information Services
Peer-to-peer file sharing	The vulnerabilities in P2P file sharing
Port scanners	The vulnerabilities about port scanning
Remote file access	The vulnerabilities about remote file access
RPC	The vulnerabilities in RPC services
Service detection	The vulnerabilities in detecting provided services
Settings	The vulnerabilities in settings
SMTP problems	The vulnerabilities in SMTP services
SNMP	The vulnerabilities in SNMP services
Untested	The vulnerabilities in applications missing test constraints
Useless services	The vulnerabilities not in general services
Windows	The vulnerabilities in Windows operating system
Windows: microsoft bulletins	The vulnerabilities announced by Microsoft
Windows: user management	The vulnerabilities in Windows user management

(1) Nmap [10]:Nmap is a package of functions to perform port-scan via these standard communication protocols, TCP, UDP, and ICMP. According to the responses from these ports on the scanned computer host, the users can have the information about enabled ports, provided services, and software versions. Actually, the software version implies whether this software might have vulnerabilities or not. The database of Nmap contains the information about what vulnerabilities exist on which version of software, where the

Table 3
Classification according to services of the vulnerabilities in TWCERT/CC

Vulnerabilities on service resources and network devices		
Network devices	Remotely login services	Security in web application programs
CISCO	RPC	PHP
3 Com	Telnet	Perl
Nortel	SSH	PHP-nuke
Alcatel	Inetd related	Zope
Cayman DSL router	Finger	Python
Shiva router	Rsh	PhpMyAdmin
HyperARC router	Rlogin	Counter
Zyxel router	Rexec	
ACC router		
Ascend router		
Database	HTTP server	SMTP server
Oracle	IIS	Sendmail
MySQL	Apache	Microsoft Exchange
Microsoft's SQL server	Netscape server	IMail
Msql	Website	MDaemon
Postgre	CERN	
Informix		
FTP server	Proxy server	Others
Wu-ftpd	Proxy related	Password reveal
Proftpd	Squid	Printer related
Serv-U		
Warftp		
Network resource sharing	DNS	Basic test
Samba	Bind	Obtaining the version information of services
NETBIOS	DNS related	
POP server	IMAP server	
Qpopper	Uw-imapd	

software has not been patched up yet. The provided port-scan functions are enumerated as follows:

- Vanilla TCP connect() scanning
- TCP SYN (half open) scanning
- TCP FIN, Xmas, or NULL (stealth) scanning
- TCP ftp proxy (bounce attack) scanning
- SYN/FIN scanning using IP fragments (bypasses some packet filters)
- TCP ACK and Window scanning
- UDP raw ICMP port unreachable scanning
- ICMP scanning (ping-sweep)
- TCP Ping scanning
- Direct (non port-mapper) RPC scanning
- Remote OS Identification by TCP/IP Fingerprinting
- Reverse-ident scanning

For example, the function, Remote OS Identification by TCP/IP Fingerprinting, obtains the version of the operation system on target host. To people know the details in operating systems, this information implies certain vulnerabilities on the host.

(2) Nessus [11]: Nessus is developed with Nessus Attack Scripting Language(NASL), similar to the C language, for system administrators to scan vulnerabilities on these systems they maintain and manage. The NASL also provides the graphic user interface (GUI), so the users and developers can manipulate Nessus easily. With referring to the built-in database and knowledge base in Nessus, users can make network communication connections, fake transmitted network packages, trigger remote network instructions, and so on. The responses from target hosts show whether the vulnerabilities exist or not. Besides, Nessus can scan many hosts concurrently.

Similar to Nmap, Nessus scans these ports on the target host first. After scanned, the information is recorded, such as the potential vulnerabilities, the vulnerability characteristics, and the possible solutions. For efficiency or other reasons, some ports and certain vulnerabilities on the target host might be not needed to be scanned. The list of ports and vulnerabilities can be adjusted by configuring "plug-in" programs in Nessus. Every "plug-in" program is a Nessus script for one vulnerability. By certain tools, C programs can be translated to be the Nessus scripts. As a new vulnerability discovered, users can write the Nessus script to detect the vulnerability. In other words, the system administrators can apply only the newest "plug-in" program if the system has been scanned.

2.4. The evaluation of vulnerabilities

The manufacturers, producing the computer software, hardware, and firmware, always provide maintenance of these systems. To eliminate vulnerabilities from systems, they provide patch-up programs for these found vulnerabilities. If vulnerabilities are not critical or fatal to damage the data and service on the hosts, the system administrators can patch their systems with these vulnerabilities till their regular checks. To these serious vulnerabilities, the system administrators should patch at once. The manufacturers have to evaluate the vulnerabilities and warn users of the danger of vulnerabilities. Since the systems are more and more complicated and sophisticated, it is hard to consider all aspects in designing systems. The vulnerabilities are therefore found and exploited rapidly. There are so many vulnerabilities exists that it is almost impossible to patch up all vulnerabilities. The system managers and system administrators might evaluate vulnerabilities with the ways proposed from manufacturers, coordinators, and researchers to determine the order for patching-up vulnerabilities.

Every organization evaluates vulnerabilities in its own way. For different evaluation ways, these are surely different, the format of describing vulnerabilities and the interval of evaluating scores. Although the intervals of evaluated scores from different organizations are different, these scores indicate the risk of vulnerabilities within the same evaluation system. The manufacturer, Microsoft, classifies vulnerabilities with four ratings, "Critical", "Important", "Moderate", and "Low". The definition of these ratings is shown in Table 4. The security coordinator, US-CERT, rates vulnerabilities within the interval of 0 to 180 with respect to seven aspects, as shown in Table 5. For every aspect, the evaluation value is from 0 to 20. In the research department, Secunia, there are five ratings proposed, which are "Extremely Critical", "Highly Critical", "Moderately Critical", "Less Critical", and "Not Critical", as shown in Table 6. Above all, the system managers and system administrators might be confused in the usage of these evaluation ways from different organizations.

Table 4
Microsoft security alert severity matrix [12]

Rating	Definition
Critical	A vulnerability whose exploitation could allow the propagation of an Internet worm without user action.
Important	A vulnerability whose exploitation could result in compromise of the confidentiality, integrity, or availability of users' data, or of the integrity or availability of processing resources.
Moderate	Exploitability is mitigated to a significant degree by factors such as default configuration, auditing, or difficulty of exploitation.
Low	A vulnerability whose exploitation is extremely difficult, or whose impact is minimal.

Table 5
US-CERT vulnerability matrix [13]

Rating	Definition
0–180	The metric value is a number between 0 and 180 that assigns an approximate severity to the vulnerability. This number considers several factors, including: $3 * (Q1 + Q2 + Q3) * (Q4 * Q5 * Q6 * Q7)/(20\hat{}4)$ Q1: Is information about the vulnerability widely available or known? Q2: Is the vulnerability being exploited in the incidents reported to US-CERT? Q3: Is the Internet Infrastructure at risk because of this vulnerability? Q4: How many systems on the Internet are at risk from this vulnerability? Q5: What is the impact of exploiting the vulnerability? Q6: How easy is it to exploit the vulnerability? Q7: What are the preconditions required to exploit the vulnerability?

Table 6
Secunia advisories [14]

Rating	Definition
Extremely critical	Typically used for remotely exploitable vulnerabilities that can lead to system compromise. Successful exploitation does not normally require any interaction and exploits are in the wild.
Highly critical	Typically used for remotely exploitable vulnerabilities that can lead to system compromise. Successful exploitation does not normally require any interaction but there are no known exploits available at the time of disclosure.
Moderately critical	Typically used for remotely exploitable Denial of Service vulnerabilities against services like FTP, HTTP, and SMTP, and for vulnerabilities that allows system compromises but require user interaction.
Less critical	Typically used for cross-site scripting vulnerabilities and privilege escalation vulnerabilities.
Not critical	Typically used for very limited privilege escalation vulnerabilities and locally exploitable Denial of Service vulnerabilities.

2.5. The analyses of found vulnerability

The vulnerability scanning software can find out the known vulnerabilities on computer systems. After the known vulnerabilities are found, the administrators should analyze the information to evaluate the priority of patching up these vulnerabilities for which the patch-up programs are available. The analysis result may be biased by the administrators' opinions. This section shows three standard ways for analyzing. Section 2.5.1 introduces a risk assessment analysis method. By this method, the system manager can know of which vulnerabilities they should be careful referring to the value of assets. The second way is about plotting the attack graph. If the system managers want to prevent their systems from these outside attackers, they can plot the attack graph to identify what vulnerabilities should be blocked to interdict the path from the external machine to the target host. In Section 2.5.3, the evaluation based on CVSS values is provided. The CVSS values are defined by Forum of Incident Response and Security Teams (FIRST). The Administrators can take the CVSS values to evaluate the priority list for patching-up vulnerabilities.

2.5.1. Risk assessment analysis

The risk assessment method is illustrated in Fig. 2. The risk assessment considers three aspects, ''Asset'', ''Threat'', and ''Vulnerability''. The product of these three values is the risk value of the vulnerability [15]. The priority of patching-up vulnerabilities depends on the damage to these assets of the department. For the hosts storing more important data or providing more critical services, the asset value is higher. It is, therefore, important to assessing the importance of the data and the services on hosts in the department, even though the assessing process is very hard to define. The assessing method depends on the system managers' opinions, so that the method might be biased. Besides, the threat value is referring to the special property of the department. For example, the research and development department might be more important than others, if there are some designed prototype products or high-tech designs protected in the host disk from the competitors. The threat value is determined by decision makers in the department. Anyway, in the risk assessment, the values of assets and threats are taken into account while the managers are indicating the importance of found vulnerabilities.

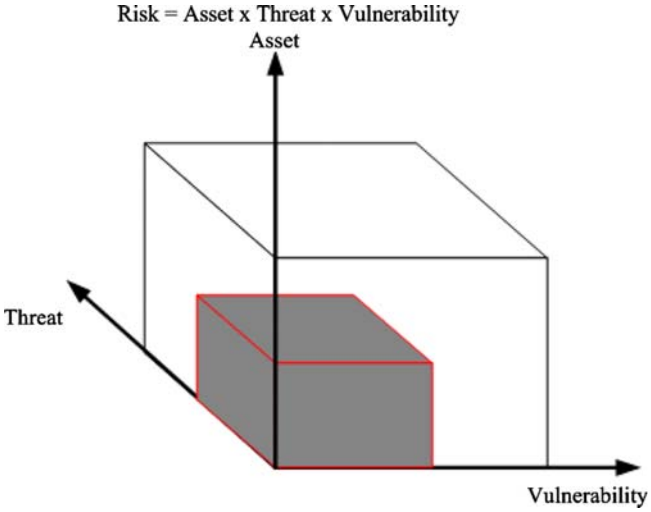


Fig. 2. Risk assessment analysis.

2.5.2. Attack graph analysis

The attack graph shows the sequence of vulnerabilities with the relation to the positions and connections of hosts [16]. Fig. 3 illustrates the attack graph with a path from the begin vertex to the done vertex by exploiting these 4 vulnerabilities, CAN-2002-0364, CVE-2001-1030, CVE-2001-0439, and CVE-2002-0004. Every vertex represents the vulnerability with the notation of a CVE or CAN number. The edges stand for the sequential exploitation of the vulnerabilities. For example, as shown in Fig. 3, the vulnerability of CAN-2002-0364 is a IIS buffer overflow vulnerability. Remote users can exploit the vulnerability to obtain user right on the host. After the exploitation, the remote users can use the port-scan vulnerability, CVE-2001-1030, on the hosts that connects to the exploited host. With several vulnerability exploitations, the target host can then be accessed.

The attack graph indicates the risk topology in a network. If the edges on the graph are weighted with the probability values according to the probability of exploiting vulnerabilities, every path from the ''begin'' vertex to the ''done'' vertex can be evaluated with a risk value by adding all the weight values on the path. The minimum weight value of paths is then the risk value of the target host in this network. With referring the risk values, the system managers can find out the ''weakest'' host in the network, which host should be patched up firstly. Besides, the attack graph might be changed when the network connection is changed. In other words, it is the other solution to change the network topology for reducing the risk values. To evaluate the risk value over the network, every host should be evaluate as a target host and the begin node may be the outer device that remote attackers can access.

2.5.3. CVSS analysis

In RSA Conference held in February 2005, Forum of Incident Response and Security Teams (FIRST) [17] proposed the Common Vulnerability Scoring System (CVSS) [18]. The scoring system is predominated by National Infrastructure Advisory Council (NIAC) and maintained by FIRST. NIAC is in charge for these information security affairs in banking, financing, transporting, energizing, and manufacturing departments. They hope the CVSS becomes a worldwide scoring standard on computer vulnerabilities. CVSS is also supported by these organizations and companies, Department of Homeland Security (DHS)/MITRE, CERT/CC, Cisco, Microsoft, Symantec, ISS, eBay, Qualys. When a computer vulnerability is found,

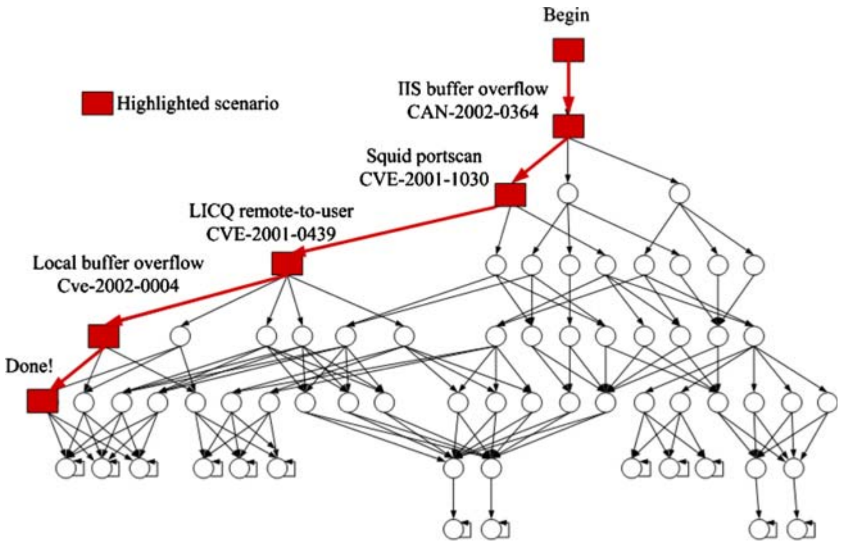


Fig. 3. Example of attack graph.

the CVSS evaluates the vulnerability by whether it can be exploited remotely or whether the attacker should login the system for exploiting the vulnerability. Nevertheless, the CVSS considers the concept of time period also, such as the time required for developing the patch-up program for the vulnerability.

3. Vulnerability information consolidation

This section introduces a method for the vulnerability information consolidation. It is organized as follows. Section 3.1 is for the network exploration. In Section 3.2, the method is presented to collect vulnerability information. Section 3.3 is for the risk evaluation of hosts in the network. Finally, the access control in network devices is provided for blockages of the vulnerabilities from being exploited. With the proposed consolidation method, the system mangers can know the vulnerabilities in the computer network, order the priority of the vulnerabilities, and block the vulnerabilities from malicious users. The network security can be then improved immediately, even thought the system managers can not patch up all vulnerabilities.

3.1. Network exploration

To use the access control on the network devices for improving the network security, the locations of hosts should be found at first. Although the system mangers have right to control all the network devices, they still need to check the network connection and topology regularly. Section 3.1.1 shows the way check the network topology. The method to locate hosts is then introduced in Section 3.1.2.

3.1.1. Network topology

To plot the data link layer topology is more difficult than to plot the network topology. The topology at the data link layer shows the real connections between network devices, which can be plotted by the link layer discovery protocol (LLDP). By using the routing table on the default router, the network topology is then determined.

There are many network device manufacturers provide the technology for the link layer discovery protocol, such as Cisco Discovery Protocol (CDP) of Cisco corporation, Extreme Discovery Protocol (EDP) of Extreme Networks corporation, Cabletron Discovery Protocol (CDP) of Enterasys Networks corporation, and Nortel Discovery Protocol (NDP) of Nortel Networks corporation. These protocols are not compatible. In May 2005, IEEE proposes the standard of 802.1AB Station and Media Access Control Connectivity Discovery [19]. The data link layer protocol can explore all devices designed with respect to the 802.1AB standard.

3.1.2. Location of hosts

With the ICMP echo request, host locations can be found in the same subnet, except these hosts on which the response for ICMP echo requests are set as the default "disable" setting in the host firewall system. Nevertheless, the network devices record the MAC addresses of the existing hosts. The IP and MAC address of hosts can be determinate via the recorded MAC address table on network switches and the ARP table on network routers.

The positions of these network devices can be determined by applying the SNMP protocol to access the management database MIB-II of every switch to collect the location information of switch interfaces. The procedures on CISCO devices can be done as follows [20].

Step1: Find the record in the MAC address table of devices
Step2: Check the bridge port number
Step3: Find the interface number of the bridge port number
Step4: Check the location of the interface number
Step5: Check the ARP table on the router to match IP and MAC

Step4: shows which network interface of the switch connect to the host with the interface of the certain MAC address. For specifying the IP address, Step 5 determines the relation between IP addresses and MAC addresses by referring the ARP table in the network router.

3.2. Required vulnerability information

To improve the network security, the access controls are used in blocking the vulnerabilities from malicious users. The access control setting consists of the port number and the IP address. After access control is set, all of the vulnerabilities via the port on the host can not be exploited. With referring to the physical connections of hosts, the system mangers can efficiently improve the network security by setting the access controls to block vulnerabilities via these ports. One access control may block several vulnerabilities on a host. In this section, the method to scan vulnerability is shown in Section 3.2.1. The scanned vulnerability information is transformed into meta-data stored in a vulnerability database. The format of the vulnerability data is defined in Section 3.2.2.

3.2.1. Vulnerability scan

The vulnerability scanning tools cost communication bandwidth in a network, so it should be prevented to scan a large scalar of hosts. Generally, the vulnerability scans are for certain ports on which the popular application services are provided. These hosts providing application services always contain more important data. These ports are port 21 over TCP for File Transfer Protocol (FTP), 22 over TCP for Secure Shell (SSH), 23 over TCP for Telnet, 25 over TCP for Simple Mail Transfer Protocol (SMTP), 53 over TCP for Domain Name System (DNS), 80 over TCP for Hypertext Transfer Protocol (HTTP), 110 over TCP for Post Office Protocol version 3 (POP3), 443 over TCP for Hypertext Transport Protocol Secure (HTTPS), 1433 over TCP for MS-sql-s, 3128 over TCP for Proxy, 3306 over TCP for MySQL.

The first of scanning procedures is to download the newest signatures for scanning tools. After server hosts found, the full scan on these hosts will be performed. Examples for the scanned information are shown in Table 7.

Table 7
Examples of the scanned vulnerability information

Hosts	Application services	Vulnerability number
192.168.1.201	microsoft-ds (445/TCP)	CVE-2000-1200
	microsoft-ds (445/TCP)	CVE-2005-1984
	domain (53/TCP)	CVE-1999-0024
192.168.1.205	http (80/TCP)	CVE-2000-1016
	http (80/TCP)	CVE-1999-0678
	http (80/TCP)	CAN-2005-1769
	https (443/TCP)	CVE-2000-1016
192.168.1.206	smtp (25/TCP)	CVE-2002-1278
	ftp (21/TCP)	CVE-2001-0550
	telnet (23/TCP)	CVE-1999-0619

3.2.2. Vulnerability database

This section presents the way to combine the scanned information. The information of each scanned vulnerability is in a pair of the CVE number and the computer port number, where CVE number is for identifying the vulnerability and the computer port is what the vulnerability is accessed via After all hosts in the network scanned, every vulnerability is determined in the three tuples of "host IP address", "vulnerabilities number", and "access port number". Besides, the scanning time is also recorded. The information about scan tasks is tabulated as shown in Table 8. The scanning time implies the version of vulnerability signatures or patterns for the scan tools. The variable, "id", is for identifying the scan task that scans numbers of hosts with IP addresses in certain range. Table 9 is for the scanned results of found vulnerabilities on hosts. The service port numbers for vulnerabilities are also recorded, since the services might not be set to access through the default service ports.

After the information about the vulnerabilities found by scanning the intra-network, a MySQL database is introduced to store the vulnerability information from the NVD database maintained by NIST. The NVD database is described in the eXtensible Markup Language (XML). The vulnerability database consists of these fields, "CVE number", "access rights", "exploitation locations", and so on, as shown in Table 10. There are three different access rights in systems, which are "admin", "user", and "other". If the value in this field of "admin" is recorded with 1, by exploiting the vulnerability, the malicious users can obtain the administrator right on the target system. The fields for the access rights are all in binary format. Some vulnerabilities can be exploited for both the normal user right and the administrator right. The "other" field identifies these vulnerabilities exploited with regarding to the unauthorized access right. For example, with exploiting the vulnerabilities of the "other" right, malicious users can explore the source codes or data in database systems. The exploitation locations are classified to be "remote" and "local", according to the exploitation can be performed from remote sites or local sites.

After the vulnerability database built, the characteristics of vulnerabilities within the network system should be considered. The access right and the exploitation location are two important indices in reducing the intrusion probability by vulnerability exploitation. To malicious users, the administrator right on hosts in the network system is the final goal in intruding the host. If the administrator right has been gained, they can take this host as a hoping site or zombie to attack others, use the storage space or computation power of this host, or collect files and data on the host.

The combinations of the access right and the exploitation location are tabulated in Table 11, which are "RA", "LA", "RU", "LU", "RO", "LO", "S", and "O". The "RO" characteristic means the malicious users can exploit the vulnerabilities to obtain the administrator right on the

Table 8
Vulnerability scanning task

Fields	Data type	Length	Explanation
Id	Int	11	An identification number for the scan task, for example: 1124094064
Time_start	Datetime		The time of this scan begins, for example: 2005-08-15 10:13:27
Time_finish	Datetime		The time of this scan completes, for example: 2005-08-15 10:17:43

Table 9
Scanned results for found vulnerabilities

Fields	Data type	Length	Explanation
Id	Int	11	An identification number for the scan task, for example: 1124094064
Host	Varchar	15	The IP address of the host on which the vulnerability found, for example: 192.168.1.214
Service	Varchar	64	The service type and service port number through which the vulnerability can be exploited, for example: http (80/TCP)
Cve_id	Varchar	15	The CVE number of this vulnerability, for example: CVE-2003-0993

Table 10
Data fields of the vulnerability database

Fields	Data type	Length	Explanation
Cve_id	Varchar	15	The CVE number of this vulnerability, for example: CVE-2003-0993
Admin	Int	2	The value is set as 1, if the administrator right can be obtained by exploiting this vulnerability
User	Int	2	The value is set as 1, if the normal user right can be obtained by exploiting this vulnerability
Other	Int	2	The value is set as 1, if this vulnerability the can be exploited without any access right
Local	Int	2	The value is set as 1, if the a vulnerability can be exploited from local sites
Remote	Int	2	The value is set as 1, if the a vulnerability can be exploited from remote sites
User_init	Int	2	The value is set as 1, if the a vulnerability can be exploited only when the malicious users logons the system or have the normal user right
Score	Int	6	The CVSS value of the vulnerability

Table 11
Vulnerability characteristics

Class	Access route	Obtained right	CVE examples	
RA	Remote	Administrator	CVE-2005-1208	In Microsoft HTML Help, the un-recognized data may lead to a fault so that un-authorized remote users can execute any programs with the right of the SYSTEM account.
LA	Local	Administrator	CVE-2003-0188	The file exploration program "lv" does not perform enough check procedures, so it can be used to execute some commands. The malicious local users can execute a trap file via using "v(edit)" command in "lv" to exploit the trap for executing any other programs and commands.
RU	Remote	Users	CVE-2005-1214	Attackers can exploit the vulnerability in the Microsoft Agent to cheat other users into accessing un-trusted web-pages to obtain the normal user right.
LU	Local	Users	CVE-2005-0004	In the mysqlaccess script of MySQL, local users are allowed to modify any temp files.
RO	Remote	Un-authorized Access	CVE-2004-0815	In Samba with the version before 3.0.6, there is an authentication problem. Malicious users can transmit a painstaking request to Samba server for unauthorized access.
LO	Local	Un-authorized Access	CVE-2002-1105	The client software of the Cisco Virtual Private Network (VPN) allows local users use a program to obtain the group password.
S	Local and Remote	User Required	CVE-2003-0370	KDE 2.2.2 has a security flaw in SSL. The man-in-the-middle attacks can be performed while users execute the Konqueror with the SLL settings.
O	Local and Remote	Other	CVE-1999-2000	Windows NT FTP server can be accesses without performing the authentication process.

target host from a remote site. The "LA" characteristic means the malicious users can only exploit the vulnerabilities locally for gaining the administrator right. By exploiting the vulnerabilities with the characteristics "RO" and "LO", the attacker can gain the right to export certain data or to execute certain programs remotely or locally. "S" means the attacker should have the rights to execute the application, for example, the right to start up connections to the malicious web-pages. The "O" characteristic is then for some configuration mistakes on systems Table 12.

3.3. Risk evaluation of ports on hosts

After vulnerabilities are scanned and identified, the "risk" of the network is assessed by distinctly evaluating

Table 12
Weights of threat classes

Threat class	RA	LA	RU	LU	RO	LO	S	O
Weight	T_{RA}	T_{LA}	T_{RU}	T_{LU}	T_{RO}	T_{LO}	T_{S}	T_{O}

each computer port in the three aspects of "Vulnerability", "Threat", and "Asset". The access controls on network devices therefore restrain these high risk ports from being accessed. These three aspects are discussed in the following.

(1) Vulnerability aspect: every scanned vulnerability is identified with a corresponding CVE number. Besides, the "Base Score" is provided for the risk of the vulnerability. The base score relating to the hardware devices or the application programs might be determined by the FIRST or the manufactures, according to the properties and effects of the vulnerabilities. If manufactures do not provide the base scores for their products to FIRST, the FIRST determines base scores for these products righteously.

(2) Threat aspect: to reduce the risk in a network, from insider or outsider attacks, the effects of vulnerabilities should be classified according to their characteristics, such as RA, LA, RU, LU, RO, LO, S, and O. These characteristics can be considered as the threat levels to the network, since they identify two things. One is whether the vulnerabilities can be exploited remotely or locally. The other is what access right can be obtained via exploiting the vulnerabilities. The classification method should be determined by the system mangers in their own way. For general, the security assessment is in assuring the "Confidentiality", "Authentication", and "Integrity" of data in systems. If the administrator right has been obtained by malicious users, they can execute any programs and access any data. In other words, the properties of "Confidentiality", "Authentication", and "Integrity" are no longer assured. Besides, the access route is also important. If the vulnerabilities can be exploited remotely, the malicious users can use them from anywhere. The threat levels are then in a decreasing order of RA > LA > RU > LU > RO > LO > S > O. The system managers can weight the threat classes to assess the risk of the network. If the mangers want to peak the vulnerabilities exploited remotely, the weights of threat classes can be decreased in the order of RA, RU, RO, LA, LU, LO, S, and O.

(3) Asset aspect: the asset of a host is the number of service ports provided via service ports, hosts can provide application services. A host can not provide service if all ports are blocked. In other words, the host is isolated from others. There is no access allowed. The more service ports a host opened, the more ways it can communicate with others. The vulnerabilities can be exploited only through service ports. Some of vulnerabilities are exploited via the same service ports. For example, on a host, there are four vulnerabilities, three via TCP 80 port and one via TCP 21 port. The TCP 80 ports are in the risk of three vulnerability exploitations. To TCP 21 port, there is one exploitation only. The number of vulnerabilities on ports should cause different weights for these ports. The system managers should consider the weights in their own way. However, the number of vulnerabilities on certain service port may be very large. Managers should quantize the weights into several levels first. In Table 13, there are 6 levels for weights.

Above all, the risk evaluation of certain port on a host can be formulated as follows:

$$(\mathrm{Host})i = \left(\sum_{j=1}^{k_i} V_{ij} \times T_{ij} \right) \times Ai$$

i, the service port i via which the vulnerabilities can be exploited; k_i, the number of scanned vulnerabilities on the port i in the host; V_{ij}, the CVSS Base Score of the jth vulnerability on port i; T_{ij}, the weight of threat class for the jth vulnerability on port i; A_i, the asset weight of the port $i$according to k_i.

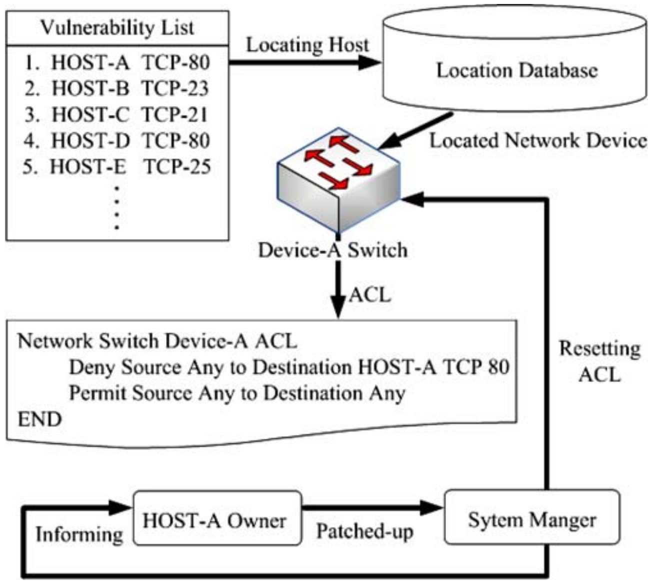
3.4. Access control in network devices

After the risk of the service ports on hosts assessed, the ports via which vulnerabilities can be exploited are listed with their risk values in decreasing. The mangers then decide how many ports should be disabled to isolate the vulnerabilities from being exploited for improving the network security. By looking up these host locations in the location database, the managers locate these network devices that these hosts connect to. The disabling process is to set the Access Control List (ACL) on network devices. The next is to inform the owners of these hosts to patch-up or up-grade their hardware or software. These owners should inform the managers when they complete the patching-up or up-grading procedures for mangers' resetting the network devices. Above steps are shown in Fig. 4.

In Fig. 4, the 80 port on Host-A is picked up, because of its highest risk in the ordered vulnerability list. Host-A and Switch-A are then located according to the data in the location database. The ACL setting blocks only connections via one service port on Host-A rather than all connections from-and-to Host-A. In other words, other services on Host-A can perform normally. The blockage eliminates the exploitation of vulnerabilities on this service port. After ACL is set, the mangers inform the owner of Host-A to patch-up these vulnerabilities via the 80 port. The ACL setting will not be changed until the mangers have the acknowledgement from the Host-A owner. The mangers

Table 13
Asset weights for service ports with respect to number of vulnerabilities on them

Number of vulnerabilities	Weight
n_1–$n_2 - 1$	A_1
n_2–$n_3 - 1$	A_2
n_3–$n_4 - 1$	A_3
n_4–$n_5 - 1$	A_4
n_5–$n_6 - 1$	A_5
n_6–Above	A_6



```
Network Switch Device-A ACL
      Deny Source Any to Destination HOST-A TCP 80
      Permit Source Any to Destination Any
END
```

Fig. 4. Flow chart of access controls.

then regularly scan, evaluate, order, and block vulnerabilities to improve the network security.

To sum up, the flow chart for the proposed method is illustrated in Fig. 5. There are two databases introduced, which are the vulnerability database and the location database. The vulnerability database consists of information about the found vulnerabilities and the NVD evaluation of these vulnerabilities. After evaluating and ranking these vulnerabilities, the system mangers decide how many ports should be blocked. The blockage process is executed with the ACL on network devices. The location of network devices and hosts are looked up from the location database storing the network topology and the host location.

4. Experimental results and analyses

The experimental environment is plotted in a Class-B campus network. The evaluation is for reducing the attacks from insiders. The firewall policy and VPN are not discussed. For managing a network, the system managers should make the policy and rule for evaluating the threat and the risk-level of the network. Besides, hosts are so many that the mangers might have no administration right and no time to patch up all hosts. The mangers, however, can know found vulnerabilities and threatened areas in the network after the risk evaluation. By the ACL on network devices, the mangers can isolate these threatened ports from attacks to improve the network security. The isolation on a service port is not relieved until all vulnerabilities on the port having been patched.

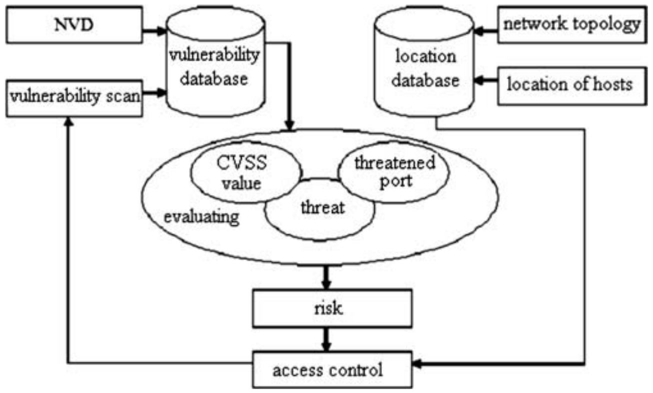


Fig. 5. Flow chart of the proposed method.

4.1. Experimental environment

The campus network for the experiment consists of more than 70 network devices and 1000 hosts. The class of the IP address is in B-class. Besides, there are some preliminaries in carrying out the proposed method.

(1) Network device: the network devices are produced by Cisco only, since the network devices are purchased in one contract.
(2) Scanning performance: the scan tool scans a range of IP addresses. The reduction of communications bandwidth between hosts causes the quality of services down within the range.
(3) Vulnerability information: actually, every country defines its own vulnerability information format. In the experiment, the definition by National Institute of Standards and Technology (NIST) is employed with Common Vulnerabilities and Exposures (CVE) numbers.

The network devices are manufactured by Cisco, so the Cisco Discovery Protocol (CDP) employs to explore the data link layer connections. The CDP can be used to find the port information on the network device, which is about the IP address of the connected hosts, the interface information about the port, and so on. The protocol can be applied to all Cisco network devices, such as Routers, Bridges, and Switches. The usage of Simple Network Management Protocol (SNMP) gains the device types and other useful information.

4.2. Experimental results

In the experiment, the scan tool, Nmap, is employed to find enabled service ports on hosts. These hosts providing service applications are more important to attackers and mangers, since the data in these server hosts are more valuable. After the hosts identified, the other scan tool, Nessus, is employed to scan in detail. The Nessus performs with "Plugins". Every "Pugin" is a small program for a kind of vulnerabilities. If a new vulnerability defined, the Nessus can be upgraded by inserting the new "Plugin" for the new vulnerability. In other words, except the first scan should be a full scan, the scan process can be with only these new "Plugins". The results of the full scan procedure are

635 vulnerabilities found via 375 ports. An example for the scanned result on a host is tabulated in Table 14.

After the risk evaluation of vulnerabilities is completed, vulnerabilities are ranked with the location information. An example is tabulated in Table 15.

These found 635 vulnerabilities are in different subnets. The percentage of vulnerabilities in every subnet is presented in Table 16. With the information, the system managers can know the "health" of subnets easily, so that, they can scan and focus on hosts in these "frail" subnets while new vulnerabilities are defined and new "Pulgins" are announced. That is because the hosts in "frail" subnets are in a "potential" high risk.

In network security issues, that is very hard to evaluate the improvement of the "security" of networks. Intuitively, it can be an index that the number of vulnerabilities is reduced or not. The fewer vulnerabilities exist in a network might mean the more secure than it was. Besides, the CVSS values are worthful to be an index, since the CVSS is defined by the worldwide authority organization, FIRST. The evaluation of security improvement consists of these two indices. One is the number of vulnerabilities in the network. The other is the sum of CVSS values of vulnerabilities in the network. The results are in the followings.

The percentages for "CVSS values" and "number of vulnerabilities" in these tables are computed from the formulas of $(\mathrm{CVSS_O} - \mathrm{CVSS_P})/\mathrm{CVSS_O}$ and [(original no. of vulnerabilities) – (present no. of vulnerabilities)]/(original no. of vulnerabilities), respectively, where $\mathrm{CVSS_O}$ means original CVSS value and $\mathrm{CVSS_P}$ means present CVSS value.

Table 16
Vulnerabilities within subnets

Subnets	Number of vulnerabilities	Percentage (%)
140.132.xb.000	203	31.97
140.132.xt.000	164	25.83
140.132.xd.000	126	19.84
140.132.xw.000	52	8.19
140.132.xl.000	37	5.83
140.132.xo.000	24	3.78
140.132.xx.000	17	2.68
140.132.yd.000	12	1.89
Sum	635	100

4.3. Analyses and discussions

These weights in the risk evaluation of vulnerabilities should be determined by systems managers to meet their requirements. The analyses for setting the weights are from two different aspects: threat-driven and asset-driven.

4.3.1. Threat-driven settings

From the results in Section 4.2, the system mangers can set the threat weight T_{RA} larger for these RA vulnerabilities if they think these are the critical threat vulnerabilities to get the administer right remotely. As shown in Table 19, the number of vulnerabilities reduces about 18.75% with only 5 service ports of 375 threatened service ports closed, even though the total CVSS value and the number of vulnerabilities reduces only 7.27% and 5.98%, respectively. If the weight T_{RA} is not many times to others, like the results in Table 18, the reductions in the total value of CVSS values and the number of vulnerabilities are significant with fewer

Table 14
Example of vulnerabilities information on the Host 140.132.xt.58

Host	Cve_id	Service port	CVSS base score	Threat class
140.132.xt.58	CAN-1999-0497	ftp (21/tcp)	4.9	O
140.132.xt.58	CVE-2001-0500	http (80/tcp)	10	RA
140.132.xt.58	CAN-2005-0048	microsoft-ds (445/tcp)	8	RU
140.132.xt.58	CAN-2005-1984	microsoft-ds (445/tcp)	7	RU
140.132.xt.58	CAN-2005-1208	microsoft-ds (445/tcp)	10	RA
140.132.xt.58	CAN-2005-1983	microsoft-ds (445/tcp)	10	RA
140.132.xt.58	CAN-1999-0504	microsoft-ds (445/tcp)	4.9	O
140.132.xt.58	CVE-2000-1200	microsoft-ds (445/tcp)	3.3	O
140.132.xt.58	CAN-2004-0574	nntp (119/tcp)	10	RA

Table 15
Example of ranked vulnerabilities

Item	Risk evaluation	Host	Service port	Network device	Device Interface
1	84.48	140.132.xt.245	microsoft-ds (445/tcp)	140.132.xr.87	Fa0/17
2	78.59	140.132.xb.9	http (80/tcp)	140.132.xr.44	Gi1/0/5
3	78.59	140.132.xb.9	https (443/tcp)	140.132.xr.44	Gi1/0/5
4	71.82	140.132.xd.155	https (443/tcp)	140.132.xr.42	Fa0/34
5	71.82	140.132.xd.156	https (443/tcp)	140.132.xr.42	Fa0/33
6	66.00	140.132.xd.155	http (80/tcp)	140.132.xr.42	Fa0/34
7	66.00	140.132.xd.156	http (80/tcp)	140.132.xr.42	Fa0/33
8	39.44	140.132.xd.155	ftp (21/tcp)	140.132.xr.42	Fa0/34
9	39.44	140.132.xd.156	ftp (21/tcp)	140.132.xr.42	Fa0/33
10	38.79	140.132.xl.114	ftp (21/tcp)	140.132.xr.64	Fa0/2

ports blocked. For example, with the blockages of 5 ports, the total CVSS value is reduced for 15.27% in Table 18 but 7.27% in Table 19. In the scenarios of 10 ports and 15 ports blocked, the reductions in Table 17 have an approximate 8% lead over the reductions in Table 19. The differences are smaller in the results shown in Table 18 and 19 for more than 15 ports blocked. For 30 ports closed, the reductions in Tables 17 and 19 are in the same values. In Tables 17 and Table 20, the reduction rates, between closing every 5 more ports, are decreasing moderately. However, in Table 19, the reduction difference between closing 15 ports and 20 ports is 11.21% in CVSS value and 11.18% in vulnerability number. The threat weight T_{RA} is set two times larger than T_{LA} in Table 19. The final reduction is similar to other results, although it is not significant in the beginning. The effect of the large T_{RA} scalar is cut for more than 20 ports blocked. These blocked port are then eliminated the vulnerabilities of LA, LU, RO, and LO. The reduction in LU vulnerability achieves 50%.

Table 17
Number of hosts within subnets

Subnets	No.	Percentage (%)
140.132.xx.000	170	14.36
140.132.xo.000	153	12.92
140.132.yf.000	135	11.40
140.132.xp.000	133	11.23
140.132.xw.000	120	10.14
140.132.xt.000	91	7.69
140.132.ye.000	74	6.25
140.132.yc.000	54	4.56
140.132.xq.000	52	4.39
140.132.xl.000	45	3.80
140.132.xz.000	38	3.21
140.132.xd.000	30	2.53
140.132.yd.000	28	2.36
140.132.yl.000	26	2.20
140.132.xb.000	17	1.44
140.132.xy.000	11	0.93
140.132.yk.000	3	0.25
140.132.yg.000	2	0.17
140.132.ym.000	2	0.17
Sum	1184	100

4.3.2. Asset-driven settings

Table 17 differs from Table 21 in the settings of the asset weights. Every service port on hosts is considered as an asset, since application services can be only provided through service ports. The blockage of a service port makes all the application services via the port be denied. From the vulnerability information, some of vulnerabilities are always exploited via certain ports. After hosts in the network scanned, these ports that might be used for exploiting any vulnerability are threatened ports. On the other hand, the ports, through which there are more vulnerabilities exploited, are in higher risk. The asset weights indicate the risks of these threatened ports. Table 22 shows the number of vulnerabilities on these threatened service ports. For the results in Table 21, the risk is quantized into 6 levels. The weight of the highest level is 1.65 for these service ports by which there are more than 15 vulnerabilities possible to be exploited on the host. In Table 17, the quantization is different. The ports with 9 or more vulnerabilities become the most threatened port. The asset weight for these ports is 1.65 also. By comparing the results in Table 17 with these in Table 21, the total value of CVSS and the number of vulnerabilities decrease a little slower before 15 ports closed down, for example 30.38% and 28.66% reduction in Table 17 versus 30.55% and 29.13% reduction in Table 21. However, the improvement gets better after the blockage of more than 20 ports. That is because these ports, with more vulnerabilities crossing, have been closed down. Some quantization levels are no more referred in the quantization way shown in Table 21. For the proposed experiments, the quantization levels are changed rather than weight values. These weight values can, of cause, be changed also. To managers, these two can be determined in their own way, the number of quantization levels and the value of asset weights.

4.3.3. Security improvement

As mentioned, it is very hard to evaluate the security for a network system. These two values can be considered as two indices for security evaluation, which are the summary of CVSS values and the number of vulnerabilities.

Table 18
The results for (T_{RA}, T_{LA}, T_{RU}, T_{LU}, T_{RO}, T_{LO}, T_{S}, T_{O}) = (0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1), (n_1, n_2, n_3, n_4, n_5, n_6) = (1, 2, 3, 4, 6, 9), and (A_1, A_2, A_3, A_4, A_5, A_6) = (1.1, 1.21, 1.32, 1.43, 1.54, 1.65)

Improved items	Number of blocked ports (%)					
	5 ports	10 ports	15 ports	20 ports	25 ports	30 ports
CVSS values	15.27	24.69	30.38	36.09	40.42	42.79
Num of vul.	15.59	24.25	28.66	33.70	37.48	39.53
Num of O vul.	10.20	15.58	17.85	20.96	23.51	24.93
Num of S vul.	100.00	100.00	100.00	100.00	100.00	100.00
Num of LO vul.	50.00	75.00	75.00	87.50	100.00	100.00
Num of RO vul.	25.55	37.23	40.15	47.45	50.36	51.09
Num of LU vul.	66.67	100.00	100.00	100.00	100.00	100.00
Num of RU vul.	22.73	22.73	40.91	45.45	45.45	45.45
Num of LA vul.	29.63	55.56	59.26	66.67	74.07	81.48
Num of RA vul.	6.25	17.50	31.25	40.00	50.00	56.25

Table 19
The results for (T_{RA}, T_{LA}, T_{RU}, T_{LU}, T_{RO}, T_{LO}, T_{S}, T_{O}) = (1, 0.5, 0.25, 0.2, 0.15, 0.1, 0.05, 0.01), (n_1, n_2, n_3, n_4, n_5, n_6) = (1, 2, 3, 4, 6, 9), and (A_1, A_2, A_3, A_4, A_5, A_6) = (1.1, 1.21, 1.32, 1.43, 1.54, 1.65)

Improved items	Number of blocked ports (%)					
	5ports	10 ports	15 ports	20 ports	25 ports	30 ports
CVSS values	7.27	16.93	22.14	33.35	40.21	42.79
Num of vul.	5.98	15.12	19.37	30.55	37.17	39.53
Num of O vul.	3.12	8.78	11.61	19.26	22.95	24.93
Num of S vul.	100.00	100.00	100.00	100.00	100.00	100.00
Num of LO vul.	0.00	25.00	25.00	62.50	100.00	100.00
Num of RO vul.	0.73	13.87	18.25	38.69	49.64	51.09
Num of LU vul.	0.00	33.33	33.33	83.33	100.00	100.00
Num of RU vul.	22.73	40.91	40.91	40.91	45.45	45.45
Num of LA vul.	14.81	29.63	33.33	55.56	77.78	81.48
Num of RA vul.	18.75	28.75	41.25	46.25	50.00	56.25

Table 20
The results for (T_{RA}, T_{LA}, T_{RU}, T_{LU}, T_{RO}, T_{LO}, T_{S}, T_{O}) = (1, 0.7, 0.55, 0.4, 0.25, 0.1, 0.05, 0.01), (n_1, n_2, n_3, n_4, n_5, n_6) = (1, 2, 3, 4, 6, 9), and (A_1, A_2, A_3, A_4, A_5, A_6) = (1.1, 1.21, 1.32, 1.43, 1.54, 1.65)

Improved items	Number of blocked ports (%)					
	5 ports	10 ports	15 ports	20 ports	25 ports	30 ports
CVSS values	11.33	22.76	30.38	35.32	40.21	42.79
Num of vul.	10.87	21.89	28.66	32.76	37.17	39.53
Num of O vul.	6.80	13.60	17.85	20.68	22.95	24.93
Num of S vul.	100.00	100.00	100.00	100.00	100.00	100.00
Num of LO vul.	25.00	62.50	75.00	75.00	100.00	100.00
Num of RO vul.	12.41	31.39	40.15	43.80	49.64	51.09
Num of LU vul.	33.33	83.33	100.00	100.00	100.00	100.00
Num of RU vul.	22.73	22.73	40.91	40.91	45.45	45.45
Num of LA vul.	22.22	51.85	59.26	62.96	77.78	81.48
Num of RA vul.	13.75	21.25	31.25	43.75	50.00	56.25

Table 21
The results for (T_{RA}, T_{LA}, T_{RU}, T_{LU}, T_{RO}, T_{LO}, T_{S}, T_{O}) = (0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1), (n_1, n_2, n_3, n_4, n_5, n_6) = (1, 2, 4, 7, 11, 16), and (A_1, A_2, A_3, A_4, A_5, A_6) = (1.1, 1.21, 1.32, 1.43, 1.54, 1.65)

Improved items	Number of blocked ports (%)					
	5ports	10 ports	15 ports	20 ports	25 ports	30 ports
CVSS values	15.27	24.69	30.55	35.70	39.99	42.58
Num of vul.	15.59	24.25	29.13	33.23	37.01	39.37
Num of O vul.	10.20	15.58	18.13	20.40	22.95	24.93
Num of S vul.	100.00	100.00	100.00	100.00	100.00	100.00
Num of LO vul.	50.00	75.00	87.50	87.50	100.00	100.00
Num of RO vul.	25.55	37.23	40.88	45.99	48.91	50.36
Num of LU vul.	66.67	100.00	100.00	100.00	100.00	100.00
Num of RU vul.	22.73	22.73	40.91	45.45	45.45	45.45
Num of LA vul.	29.63	55.56	66.67	70.37	77.78	81.48
Num of RA vul.	6.25	17.50	28.75	40.00	50.00	56.25

From the experimental results, the scales of weights affect the reduction rate in the summary of CVSS values and the number of vulnerabilities. For improving network security, the managers should tune the weights for their network system quickly to locate the high threatened subnets and to determine the high risk service ports. In the proposed experimental results, there are 375 threatened service ports. With closing 8% of threatened service ports, the summary of CVSS values and the number of vulnerabilities, are reduced about 42.79% and 39.53% in Tables 17. That percentage, 8%, is from 30 ports divided by 375 ports. In other words, the system managers can reduce almost 40% of vulnerabilities from being exploited by setting only 30 ACL scripts. The effort causes a great security improvement. Besides, there is no side-effect on other service ports of hosts. The blockages are only on these threatened service ports.

5. Conclusions

There are many researchers dedicated to improve the network security. Some focus on outsiders' attacks. To prevent the hosts accessed from the outsiders, the configuration settings are important in these network devices, such

Table 22
Number of vulnerabilities exploited via service ports

Port	No.	Percentage (%)
80/tcp	112	17.64
443/tcp	75	11.81
general/icmp	51	8.03
21/tcp	46	7.24
445/tcp	37	5.83
111/tcp	34	5.35
139/tcp	30	4.72
22/tcp	24	3.78
sometimes-rpc	23	3.62
unknown	23	3.62
23/tcp	16	2.52
general/tcp	13	2.05
79/tcp	12	1.89
513/tcp	12	1.89
161/udp	12	1.89
514/tcp	11	1.73
113/tcp	8	1.26
69/udp	8	1.26
2049/udp	7	1.10
6112/tcp	6	0.94
512/tcp	6	0.94
7100/tcp	6	0.94
2049/tcp	6	0.94
13/tcp	5	0.79
13/udp	5	0.79
9/tcp	5	0.79
7/tcp	5	0.79
7/udp	4	0.63
25/tcp	4	0.63
3389/tcp	3	0.47
3128/tcp	3	0.47
53/udp	2	0.31
general/udp	2	0.31
389/tcp	2	0.31
995/tcp	2	0.31
465/tcp	2	0.31
587/tcp	2	0.31
6000/tcp	2	0.31
2301/tcp	1	0.16
53/tcp	1	0.16
751/udp	1	0.16
997/udp	1	0.16
1433/tcp	1	0.16
42/tcp	1	0.16
137/udp	1	0.16
119/tcp	1	0.16
898/tcp	1	0.16
Sum	635	100

as the firewall, the gateway, the router, and the switch. Some are in management views. They develop methods to evaluate risks. To keep these hosts in a higher risk away from attackers, they set authentication schemes strictly. These methods of network security are introduced in Section 2. This paper proposes a method in improving network security without extra hardware cost. This proposed method consists of the network management, the vulnerability scan, the risk assessment, the access control, and the incident notification, as illustrated in Fig. 5. Every procedure is modular. The vulnerability scan tool can be upgraded by ‘‘Plugins’’ programs. If a new scan tool is developed, the Nessus can also be replaced. The weights in risk evaluation formula can be adjusted by the system managers. Besides, system managers can set scripts to execute these procedures automatically. To examine the performance of the proposed method, an experiment of a real Class-B network environment is taken and shown in Section 4. There are more than 1000 hosts within the network. The experimental results are evaluated with the summary of CVSS values and the number of vulnerabilities. In the future work, the specific service ports may be considered. Since the security improvement is based on the blockage of these threatened service ports with ACL scripts, the manager may need to define specific weights for some service ports if these service ports should be enabled anyway, for example the ports on Honey-Pot hosts or the ports for important applications.

Acknowledgements

This work was supported in part by International Collaboration for Advancing Security Technology (iCAST) project, National Science Council under the Grants NSC 95-3114-P-001-002-Y02, and NSC95-3114-P-014-001-Y.

References

[1] PITAC, Cyber security: a crisis of prioritization, <http://www.nitrd.gov/pitac/reports/20050301_cybersecurity/cybersecurity. pdf, 2005.>

[2] Code Red Worm, <http://www.cert.org/advisories/CA-2001-19.html/>.

[3] Nimda Worm, <http://www.cert.org/advisories/CA-2001-26.html/>.

[4] CERT/CC, CERT/CC Statistics 1988-2005, <http://www.cert.org/stats/cert_stats.html#vulnerabilities/>.

[5] CERT/CC, CERT/CC Statistics 1988-2005, <http://www.cert.org/stats/cert_stats.htm#incidents/>.

[6] CVE, <http://cve.mitre.org/>.

[7] NVD, <http://nvd.nist.gov/>.

[8] CVE-2006-0010, <http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2006-0010/>, 2005.

[9] TWCERT/CC, <http://www.cert.org.tw/service/VulDB/>.

[10] Nmap, <http://www.nmap.com/>.

[11] The Nessus Project, <http://www.nessus.org/>.

[12] Microsoft security response center security bulletin severity rating system, <http://www.microsoft.com/technet/security/bulletin/rating.mspx/>.

[13] US-CERT Vulnerability Matrix, <http://www.kb.cert.org/vuls/html/fieldhelp#metric/>.

[14] Secunia Advisories, <http://secunia.com/about_secunia_advisories/>.

[15] Zeki Yazar, A qualitative risk analysis and management tool, SANS Institute, 2002.

[16] O. Sheyner, J. Wing, Tools for generating and analyzing attack graphs, LNCS 3188 (2004) 344–371.

[17] FIRST, <http://www.first.org/>.

[18] CVSS, <http://www.first.org/cvss/>.

[19] IEEE Std 802.1AB, Station and media access control connectivity discovery, IEEE Standard for Local and metropolitan area networks, 2005.

[20] CISCO Document ID:40367, CISCO-SNMP Community String Indexing, <http://www.cisco.com/warp/public/477/SNMP/camsnmp40367.pdf/>, 2005.

Yeu-Pong Lai received his BS degree in Information Science from Chung Cheng Institute of Technology, Taiwan, Republic of China, in 1993; the MS degree in Mechatronics in 1996 and the MS degree in Electronics Research in 1997 both from King's College London, University of London, UK; the PhD degree in Computer Science and Information Engineering from the Department of Computer Science and Information Engineering at National Chung Cheng University in 2004. He is on the faculty of Chung Cheng Institute of Technology, National Defense University, as an associate professor. His major interests are information security, network security, cryptography, fast computation, and computer arithmetic.

Po-Lun Hsia, born in 1972, received his BS degree in Electronic Engineering from Chung Cheng Institute of Technology, Taiwan, ROC, in 1993; the MS degree in Computer Science and Information Engineering in 2006 from Chung Cheng Institute of Technology. He is currently a major instructor in Army Communication Electronic Information School. Besides, he is in charge of designing information management systems and planning computer network topology in the school.