Krysta Kaye
Version 1.2e

Vulnerability Assessment of a University Computing Environment

When I began looking at possible solutions to the problems, I found a lot of information on assessing businesses, but very little on assessing a University or portion thereof. This document is designed to be a guide for future administrators who find themselves in the same situation. There are two resources that guided me through my vulnerability assessments: *Threat and Risk Assessment Working Guide*[1]and *Open-Source Security Testing Methodology*[2].

University environments tend to be a mixed bag as far as networking goes. Often the network has grown from several separate networks connecting to each other out of necessity. And in the case of security, it is often left up to the Network Administrators of the various departmental local area networks (LANs) to secure their portion. Some Network Administrators know and understand the need for security while others don't feel it is necessary. Other problems include:

- Faculty, staff and students who are encouraged to learn the new technologies setting up the machines on the live network without any security precautions taken
- Computer labs open to the general University population where access is granted based on current identification cards
- University libraries have public access computers where access is open to both the University population and the community at large
- Mixture of operating systems – Novell, Windows NT, Windows 2000, Windows 98, Unix, Linux, MacOS
- The experience level of the faculty, staff and students who are setting up the computers – often these people are learning the ropes and not necessarily the experts
- No cohesiveness to the institution – departments don't relay information to each other about security issues.

I became involved in security for my University Library after a large crack – the attackers compromised a Linux box, which led them to the NT boxes, which led them to destroying several NT operating system files and wreaking all kinds of havoc on our network. The hack could have been prevented if we had the knowledge (what the holes were, where the holes were, how we could fill the holes or work around them, etc.) at the time. Sure, we had a University-wide Security Policy, but we didn't know what we needed to do to implement the policy, to make it a reality. If only we had an assessment of our security situation before the attack.

To begin the assessment, get a copy of the University Security Policy. If one doesn't exist, bring it up to the appropriate University committee and in the meantime, work on one for your department. There are several articles on writing security policies, see Other Internet Resources: Security Policies at the end of this document. Look the policy over to see what is included. "A security policy establishes what must be done to protect information stored on computers. A well-written policy contains sufficient definition of 'what' to do so that the 'how' can be identified and measured or evaluated."[3] The policy should contain information about:

- The connections to/from the Internet
- Dial-up connections

- Physical security
- Passwords
- User rights and responsibilities
- Administrator rights and responsibilities
- Protection of sensitive information
- Emergency procedures
- Documentation
- Backups
- Logs
- Incident handling
- How people go about reporting a security issue
- Enforcement of the policy
- Who is ultimately responsible[456]

The policy should not be procedural; it should only give a high-level view of the security necessity.[7]

Find out if the University (or State, if you are a public institution) has any procedures in place for security or security incidents. Check with other departments, such as the Computer Science program, University Police Department, centralized computing group, etc. Don't reinvent the wheel if you don't have to. Keep in mind that procedures can be modified to fit the situation. Your procedures should be step-by-step on how to make the Security Policy a reality. Checklists for:

- Assessing and auditing workstations and servers for each operating system
- Dealing with incidents
- User administration (i.e., passwords, account creation, etc.)
- Workstation administration (i.e., imaging, deploying, etc.)
- Inventory (software and hardware) control
- Software licensing
- In depth user responsibilities
- In depth administrator responsibilities
- Remote user access
- Disaster recovery
- Types of connections allowed to/from the Internet
- Dealing with viruses
- Dealing with e-mail (i.e., we keep e-mail for x years, backups, etc.)
- Policy review
- Privacy issues
- Dealing with emergencies (i.e., communications closet flooded, but workstations and servers are still up and running, just no outside communication)
- Documentation for various applications (i.e., installation and maintenance)
- Network map/topology
- Security configuration for each system
- Backups

- Auditing logs
- Handling of identified intruders
- Physical security
- Electrical security
- Theft[8]

Once you have your policies and procedures, you can begin the vulnerability assessment. A good starting guide is *Managing Network Security: How Does a Typical IT Audit Work?* By Fred Cohen.[9] The Cohen guide gave me insight into the process an audit should take. Another good broad assessment can be done using the *Computer Security Self-Assessment Checklist* from MIT.[10] This document provided me with a good general starting point. Always get written permission to perform the vulnerability tests. It should be signed at the very least by your supervisor, if not both your supervisor and their supervisor. This document should explain what you plan to do and what the potential consequences are of the tests. For example:

> I, <supervisor>, give <your name> permission to perform network and host vulnerability testing on the IP range of <ip address range>. This testing will include probing for various vulnerabilities and security holes of machines on the network. The probing will be done with a variety of security assessment tools, such as Kane Security Analyst by Intrusion.com, SAINT from World Wide Digital Security, Inc., etc. Also, <your name> will be performing a penetration test of the various workstations and servers (NT, Novell, Linux, Unix) on the network defined by the above IP range. The penetration testing will be done with a variety of hacking tools, such as Pandora (Novell), Snort (sniffer), L0phtCrack (NT password cracker), etc. <he/she> will also create a network map diagramming the various nodes of the network defined by the above IP range.
>
> <your name> has notified me that the performance of these tests may crash the machines on the network defined by the above IP range. <he/she> will send out an e-mail message to the staff warning of the possibilities of system problems during the testing.
>
> <your name> will be testing the various servers on <date>. <your name> will be testing the workstations between <date range>. The testing of the workstations will be done in sections, based on departments. Each department will receive an e-mail notification when it is their turn. <your name> will attempt to schedule these tests after 5 p.m. to avoid outages during normal business hours.
>
> Supervisor
> Supervisor’s Title
>
> Supervisor’s Supervisor
> Their Title

This document will basically notify the appropriate people of your intentions and cover you, in the event of a server or workstation outage due to penetration testing. In my case, the tests did not crash my systems, except when I ran Denial of Service (DoS) attacks on the servers.

What is an audit? What is an assessment? According to Ira Winkler, "An audit is called for when you want to know how your organization measures up to specific standards, since this kind of testing provides guidance on improving your adherence to these standards…. The simplest definition of security assessment is an overt study to locate security vulnerabilities."[11] So, based on these definitions, an assessment needs to be done first to give an initial baseline, from which you will perform future audits. The assessment of vulnerabilities begins with a network map. There are several port scanning utilities out there, both free and commercial. But, as George Kurtz and Chris Prosise say, "a healthy dose of nmap, the king of port scanning utilities will help identify all those pesky TCP and UDP ports."[12] I chose Nmap to map my network. There are several articles on using Nmap, see the Other Internet Resources list at the end of this document. Nmap is available for both Unix and NT operating systems. One of its best features is that it is free. I found out that Nmap[13] provides more than just a scan of open ports. Nmap will discover hosts on your network if given an IP range, it will resolve the IP to a name address, it will tell you which ports are open, it will tell you what they are normally used for, it will perform TCP sequence prediction on the host, it will give you a "guess" at the host's operating system (even for printers and routers, etc.), and it will even give you the version number of the operating system if it can detect it. Be prepared, though, Nmap gives you a lot of data when used on a large network.

Next, run a vulnerability scanner. Once again, have your written permission signed at least by your supervisor. There are many scanners available. Some are commercial and some are free. I chose Nessus[14] based on a January 8, 2001 Network Computing article titled *Vulnerability Assessment Scanners*.[15] Nessus received the highest marks. "One thing we particularly like is Nessus Security Scanner's "honesty" when it guesses about vulnerabilities and possibly presents inaccurate data. For example, if the product made an assumption about a particular service that might not be entirely accurate, it warned us of this assumption…. Nessus Security Scanner still got the highest overall score simply because it did more things right than the other products."[16] After reading the review and looking at it's features, I knew that Nessus fit my needs – cheap, fast and able to scan and detect vulnerabilities on multiple operating systems. It was nice to learn that Nessus also "has developed quite a following in the security community. Indeed, the U.S. mirror of the Nessus download site is housed by the Treasury Department's Computer Investigative Specialist Program. Nessus was also the number-one tool in a recent survey run by insecure.org, a security Web site."[17] Nessus is made up of two parts – the server and the client. The Nessus server is only available for the Unix operating system. While the client is now available in Unix, NT and Java formats. There are several articles written on the use of Nessus, see the Other Internet Resources list at the end of this document.

The scans I initially ran were "all but dangerous plugins" as put in the Nessus interface. This enabled all of its tests, excluding the DoS attacks. I ran the DoS attacks later. Nessus (like the other available scanners) warns that it could crash your systems. Like I said earlier, I had no problems with any crashes until I performed the DoS attacks.

Nessus can save all of the data collected in HTML format for later viewing. Be aware, though, once again you will get a mountain of data.

Once the network scan was completed, I checked the new machine images for vulnerabilities. I did this on a host-by-host basis. I searched high and low to find some information on auditing an individual workstation and I came across a wonderful resource guide called *Windows NT 4.0 Test Report* by Randy Marchany of Virginia Tech.[18] Marchany created a checklist for auditing Windows NT 4.0 machines. He describes each test by "required action," "expected results," and "comments." His instructions in the "required action" portion are clear and concise. The information in the "expected results" field helps the auditor/assessor understand what the machine's security settings should be. And, his comments field offers other insights into the operating system and its security.

Though, Marchany's auditing only requires c2config and passprop from the *Windows NT Resource Kit*, to be thorough it may help to have: c2config, passfilt, dumpel, netsvc, adduser, ssydiff, regdmp, xcacls, and perms on a CD ready to go.[19] The University of Florida Computer and Network Security has a great checklist called "How to Secure NT" in which they describe the use of many of these tools.

Another good source for auditing/assessing an NT workstation/server is *NT 40 Auditor's Checksheet*.[20] This checklist offers information on the potential threats and vulnerabilities of different parts of the NT system, how-tos on defending against those threats and vulnerabilities, where to find information on remedies and fixes for those threats and vulnerabilities, and the author's personal experience in using the fixes and remedies. I used this checklist more for making recommendations in my final report.

Speaking of the report, it is now time to sift through all of the data that we have accumulated to this point. "The assessment report must include comprehensive information on how to secure the client's technical and non-technical vulnerabilities. This is the key thing that makes an assessment different from an audit: An assessment report highlights the vulnerabilities and tells how to protect the systems in detail. Perhaps the biggest indicator of a report's quality is whether it simply notes how to fix the problems encountered, or goes on to address their underlying causes."[21] The report should not be written completely in computerese, it needs to be written such that the audience (generally non-IT management) can understand it. If the audience doesn't understand it, they will most likely set it aside and ignore it. This can spell disaster for security.

Once the report is completed and approved by management, it is time to fix/patch the various systems and implement the recommendations. Also, keep in mind that security is a never-ending process. Your assessment only covers the current moment in time. The security of your University will change as systems are added and subtracted from the network, applications are added and subtracted from systems, and faculty, staff and students come and go. Audits will need to be performed regularly in a University environment, this may need to be at the beginning or ending of each semester (roughly 3 times a year). This is especially true in the case of public access workstations, where it is more difficult to control the users' actions. As the cracking world gains newer and more powerful tools, the security on these public access machines will be tried first. Which means that the security and system administrator(s) need to be kept up-to-date on security issues. I recommend joining several of the

Security Focus (http://www.securityfocus.com - click on Mailing Lists in the left frame) and SANS (http://www.sans.org/sansnews) listservs.

Other Internet Resources:

Security Policies –

"Infosyssec: The Security Information Portal for Information System Security Professionals Security Policy Writing Styles & Guides". 2000. URL: http://www.infosyssec.com/infosyssec/secpol1.htm. May 27, 2001.

"CAF "Academic Computing Policy Statements" Archive". June 28, 1999. Electronic Frontier Foundation. URL: http://www.eff.org/pub/CAF/policies/. May 27, 2001.

Bergsma, Kathy. "Security at Other Universities". August 24, 1999. University of Florida. URL: http://logjam.nerdc.ufl.edu/~security/universities.html. May 27, 2001.

Helwig, Steven M. "Security Policy for Higher Educational Institutions". December 15, 2000. SANS. URL: http://www.sans.org/infosecFAQ/policy/higher_edu.htm. May 27, 2001.

"College and University Information Security Professionals Policies and Procedures". October 20, 2000. Massachusetts Institute of Technology. URL: URL: http://web.mit.edu/security/www/cuispnew/policies.htm. May 27, 2001.

"Security Policy Research Project". 1999-2000. SANS Institute. URL: http://www.sans.org/y2k/sec_policy.htm. May 27, 2001.

"Security Policy Issues". May 24, 2001. SANS Institute. URL: http://www.sans.org/infosecFAQ/policy/policy_list.htm. May 27, 2001.

Using Nmap -

Nmap http://www.nmap.org/

"Intrusion Detection FAQ: What is Nmap and What Can It Do?". SANS Institute. URL: http://www.sans.org/newlook/resources/IDFAQ/What_is_nmap.htm. May 27, 2001.

Using Nessus -

Nessus http://www.nessus.org

Brooks, Greg. "Nessus – Get on Board". February 15, 2001. SANS Institute. URL: http://www.sans.org/infosecFAQ/audit/nessus2.htm. May 27, 2001.

Stark, Vernon. "Nessus - A Very Capable Security Auditing Tool". August 6, 2000. SANS Institute. URL: http://www.sans.org/infosecFAQ/audit/nessus.htm. May 27, 2001.

Auditing -

Stephanou, Tony. "Assessing and Exploiting the Internal Security of an Organization". March 13, 2001. SANS Institute. URL: http://www.sans.org/infosecFAQ/audit/internal_sec.htm. May 27, 2001.

Herman, Ben. "Routine External and Internal "Hacking", An Important Part of Information Assurance". April 19, 2001. SANS Institute. URL: http://www.sans.org/infosecFAQ/attack/routine.htm. May 27, 2001.

Policy Compliance -
Naidu, Krishni. "How to Check Compliance with your Security Policy". January 30, 2001. SANS Institute. URL: http://www.sans.org/infosecFAQ/policy/compliance.htm. May 27, 2001.

Listservs –
Security Focus http://www.securityfocus.com - click on Mailing Lists in the left frame
SANS http://www.sans.org/sansnews

Cited Resources:

[1] Government of Canada, Communications Security Establishment. *Threat and Risk Assessment Working Guide*. Ottawa, 1999. URL: http://www.cse-cst.gc.ca/cse/english/Manuals/ITSG-04e.pdf. May 27, 2001.

[2] Herzog, Pete. *Open-Source Security Testing Methodology.* Eds. Drew Simonis, Emily K. Hawthorn, Jordi Martinez. osstmm.en.1.5. May 5, 2001. URL: http://www.ideahamster.org/osstmm/htm. May 27, 2001.

[3] Northcutt, Stephen. "Basic Policy." Paper presented as part of LoneStar SANS Institute Security Essentials Curriculum 1.1 Track 1. March 22-25, 2001. SANS Institute.

[4] Moss, Brion. "Security Policy Checklist". October 6, 1997. URL: http://queeg.com/~brion/security/secpolicy.html. May 27, 2001.

[5] Guttman, Barbara and Robert Bagwell. *Internet Security Policy*: *A Technical Guide.* NIST Special Publication 800-XX INTERNET SECURITY POLICY: A TECHNICAL GUIDE. July 31, 1997. National Institute of Standards and Technology Computer Security Resource Center. URL: http://csrc.nist.gov/isptg/html/ISPTG-Contents.html. May 27, 2001.

[6] Knight, Eric. *Computer Vulnerabilities.* Release 4. March 8, 2000. URL: http://www.securityfocus.com/data/library/compvuln_draft.pdf. May 27, 2001. p.22-24.

[7] Guttman, Barbara and Robert Bagwell. *Internet Security Policy*: *A Technical Guide.* NIST Special Publication 800-XX INTERNET SECURITY POLICY: A TECHNICAL GUIDE. July 31, 1997. National Institute of Standards and Technology Computer Security Resource Center. URL: http://csrc.nist.gov/isptg/html/ISPTG-Contents.html. May 27, 2001.

[8] "Security Review Checklist". 1997. Rainbow Technologies, InfoSec Services, Spectria Division. URL: http://www.infosec.spectria.com/articles/check-rvw.htm. May 27, 2001.

[9] Cohen, Fred. "Managing Network Security: How Does a Typical IT Audit Work?" URL: http://all.net/journal/netsec/9807.html. May 27, 2001.

[10] "Computer Security Self-Assessment Checklist". June 30, 1998. Massachusetts Institute of Technology. URL: http://web.mit.edu/security/www/isosec-assess.htm. May 27, 2001.

[11] Winkler, Ira. "Audits, Assessments and Tests (Oh, My): Systems security tests come in three basic flavors. Here's how to make sure you're performing only the test(s) you really need". Information Security. July 2000. URL: http://www.infosecuritymag.com/articles/july00/features4.shtml. May 27, 2001.

[12] Kurtz, George and Chris Prosise. "Secure Strategies: Penetration Testing Exposed". Information Security. September 2000. URL: http://www.infosecuritymag.com/articles/september00/features3.shtml. May 27, 2001.

[13] http://www.nmap.org

[14] http://www.nessus.org

[15] Forristal, Jeff and Greg Shipley. "Vulnerability Assessment Scanners". Network Computing. January 8, 2001. URL: http://www.networkcomputing.com/1201/1201f1b2.html. May 27, 2001.

[16] Forristal, Jeff and Greg Shipley. "Vulnerability Assessment Scanners". Network Computing. January 8, 2001. URL: http://www.networkcomputing.com/1201/1201f1b2.html. May 27, 2001.

[17] Berg, Al. "Secure Strategies: A Year Long Series on the Fundamentals of Information Systems Security Part 2 Audits, Assessments & Test (Oh, My)". Information Security. August 2000. URL: http://www.infosecuritymag.com/articles/august00/features4.shtml. May 27, 2001.

[18] Marchany, Randy. "Appendix J: Windows NT 4.0 Test Report". Virginia Tech. URL: http://courseware.vt.edu/marchany/nt40Audit/audit.html. May 27, 2001.

[19] Brenton,Chris. "Poor Man's NT Auditing." Paper presented as part of LoneStar SANS Institute Security Essentials Curriculum 1.1 Track 1. March 22-25, 2001. SANS Institute. P. 5-4 through 5-39.

[20] "NT 4 Auditor's Checksheet". Talisker's Network Security Tools: Securing Windows NT4. URL: http://www.networkintrusion.co.uk/NTTheFull Monty.htm May 27, 2001.

[21] Winkler, Ira. "Audits, Assessments and Tests (Oh, My): Systems security tests come in three basic flavors. Here's how to make sure you're performing only the test(s) you really need". Information Security. July 2000. URL: http://www.infosecuritymag.com/articles/july00/features4.shtml. May 27, 2001.