PENETRATION TESTING: A VITAL COMPONENT OF AN INFORMATION SECURITY STRATEGY

James K. Smith, Texas A&M University, jklatham@gmail.com
Jack D. Shorter, Texas A&M University, jack.shorter@tamuk.edu

==========

ABSTRACT

Wikipedia has defined penetration testing as "a method of evaluating the security of a computer system or network by simulating an attack from a malicious source..." [8]. There are many different levels of penetration tests that can be performed on an organizations network and security infrastructure. These tests can range from simply attempting a brute force password attack on a particular system, all the way up to a simulated attack including, but not limited to social engineering.

Keywords: Penetration Testing, Black Box Penetration Testing, Scanning and Enumeration

INTRODUCTION

There are countless guides, books, and resources available for the modern network administrator or information technology executive to facilitate making the sensitive data on their networks and computer systems secure. He or she can have a highly secure network by having a robust Acceptable Use Policy (AUP), Security Policy (SP), along with well trained staff in both the information technology department and the organization as a whole. The data on the computers within the network can be protected by the most expensive up-to-date firewalls and encryption methods. The equipment can be physically protected by video monitoring, multiple security guards, and locked doors that lead to man-trap hallways. When all of this security is in place, how can a network administrator be confident that the security measures that he has implemented to protect his organization's data are actually working? To make sure that the strategies are effective, a penetration test should be performed at regular intervals or when a change has been made to the organizations information systems or policies.

ADVANTAGES OF PENETRATION TESTING

Wikipedia has defined penetration testing as "a method of evaluating the security of a computer system or network by simulating an attack from a malicious source..." [8]. There are many different levels of penetration tests that can be performed on an organizations network and security infrastructure. These tests can range from simply attempting a brute force password attack on a particular system, all the way up to a simulated attack including, but not limited to social engineering, white and black box methodologies and internet based denial of service (DoS) attacks. The advantages and insight that these different testing methodologies provide an organization can be invaluable.

First and foremost, a well done penetration test will allow an organization's information technology (IT), or information system (IS) departments to identify the vulnerabilities in their current security policies and procedures. For instance, if the team performing the penetration test gained access to sensitive areas through a social engineering technique, the target organization could then coach the users who were responsible for the breach on the proper procedures for granting access to unknown or unauthorized parties. Without the penetration test, the organization may never discover that the vulnerability was present in the security procedures and it could have been exploited by a party with a much more sinister motive.

BLACK BOX PENETRATION TESTING

The black box method of penetration testing involves recruiting a penetration testing team, often referred to as a "red team" or "tiger team" to perform security assessments without any prior knowledge of the existing network. Other than the initial contact to gain authorization to perform the test, there is no contact between the red team and the target organization [7]. This method would be most effective if the organization wanted to measure how vulnerable they were to a completely unknown attacker attempting to gain access. The red team would be forced to go through the entire process of gathering information on the company through various methods and then building upon that information to eventually make their simulated attack.

While the black box method can be effective in establishing a baseline of how secure an organization is to outside attacks, it is highly subjective on the part of the team doing the assessment. This is because there are countless ways of performing reconnaissance on the organization and due to restrictions on time and money a red team is not likely to utilize all of the methods and techniques that are available to them. There are always other methods of gathering information on a target that the red team might not have considered, or even know about.

RECONNAISSANCE

Any well planned attack, whether it is political, military, or personal in nature will start with a reconnaissance of the target. There are multiple avenues in which a red team has the opportunity to gather information about an organization's operations. Passive reconnaissance is the most simple and least dangerous form of gathering this information. It can simply involve sitting outside of a building with a pair of binoculars or a high powered camera waiting for a UPS or FedEx truck to arrive. As the driver unloads the truck the surveillance team can get a view of what type of computer systems the organization uses because hardware vendors usually print their logos and photos of the equipment on the shipping boxes. The red team now knows what type of hardware the organization utilizes to perform its business operations.

The surveillance team can then follow up by returning later that evening or later in the week to rifle through the dumpster. This technique is called "dumpster-diving." In the dumpsters they might find packing lists, installation manuals, and software boxes that were contained in unmarked shipping materials. This information would be considered extremely valuable to any attacker because they would now have a very good picture of both the hardware, and software being used at that location. This information coupled with the potential hardware vulnerabilities discovered earlier provides the red team with nearly everything it would need to start exploiting security vulnerabilities in the organization's computer and network systems.

Once successful reconnaissance has been performed on the target organization, the red team will take the information gathered during that phase and compile a report on what actions were taken. This report will be included with the final report with recommendations to the management of the organization on what measures might be taken to mitigate the risk of reconnaissance methods used to gather information. In the case of the previous scenario the report might suggest that all discarded manuals, software boxes, and reference material be shredded before disposal.

SCANNING AND ENUMERATION

Once initial surveillance has been performed on the target, a red team will then attempt to list and identify the systems and specific services and resources the target provides [3]. The first step in this process is to scan down all available systems on the network. Once available systems have been found, the red team will attempt to enumerate the available systems. Successful enumeration will include information such as a host name, its associated internet protocol (IP) address, and what services are running on the different network hosts [3]. The data collected during the scanning and enumeration will allow the red team to narrow down the scope of the simulated attack which will follow later.

Normally the red team will perform their initial scan with a scanning tool application. These applications vary in their complexity but most will perform both scanning and enumeration. These tools have many different options from an all out scan of an entire range of IP addresses as fast as it can scan, or they can spend hours, or days scanning in a more stealthy fashion. Scanning such as this is normally done in two different phases. One scan will be performed during the day to get an idea of the layout of the entire network. The next scanning attempt will be done in the middle of the night to find equipment that is kept on for 24 hours at a time. This will give the red team the best chance to identify the IP addresses of routers, switches, firewalls, and most importantly the servers.

SOCIAL ENGINEERING

Social engineering is likely the most complicated and risky aspect of attempting an intrusion of an organization. It requires the red team to gain the confidence of users on the organization's system and then provide them with either access, or information on what the team wants to know. It is essentially a confidence game based on appearance and expectations to gain the trust of the system users of

the organization [10].

There are varying methods used to gain someone's trust. One method is for the red team to do research on individuals who work at companies. With the rise in popularity of social networking web sites this has become a much easier task. It can be as easy as going to the social networking site Facebook.com and doing a search on people employed at the organization. The attacker can then create a fake profile, usually of an attractive woman due to the fact that men are much more receptive and disarmed by women. Once this has been done the attacker only needs to pass himself off as a new employee and start sending out friend requests to all members of the organization claiming how excited they are to be working with everyone. Then the attacker is likely to be able to gather names, phone numbers, and start following people on twitter. This will give the red team plenty of information on which to build a back story and then use that information to gain physical access to the organization [6].

WHITE BOX TESTING

The white box penetration testing methodology is a less time consuming method of penetration testing. When a penetration testing team performs a white box test on their target, the target is usually somewhat, if not fully aware of what is taking place [7]. They are usually given full access to all of the network IP ranges, topologies, operating systems, and anything else that might be requested by the tester performing the audit. It might even involve the tester being brought into a building and shown a network port on which to begin their scan [9].

This particular method of penetration testing provides the organization with a very thorough view of all of its software and network vulnerabilities, but this method has some disadvantages. For instance, unless specifically called for, there is very little need to perform social engineering in white box penetration testing because the auditor is already aware of most of the information that he or she needs. Therefore, the organization likely will not get the benefit of having its security policies and user training tested. This is troublesome because of the fact that in most organizations the information systems are normally secure, but the weakest link in the chain is the people within the organization [1]. In an effort to perform its due diligence an organization should always perform both a white box penetration test, as well as a black box penetration test.

AUTOMATED VULNERABILITY SCANS

Automated vulnerability scans are becoming much more popular to use when performing penetration testing. When the appliance is finished performing its scan the automated scanning appliance then creates a report detailing the vulnerabilities on the system that have been found. The report can either be reviewed by an auditor, or sent directly to the information systems staff at the organization.

The automated vulnerability scanners are popular because they do not have an actual penetration tester performing the assessment, and therefore cost less for the organization paying for the service as well as the company providing the service. This is not to say that using an automated vulnerability scan is a bad idea, the problem with it is that it has started to become the de facto standard for an organization to consider its network infrastructure secure. This is partly due to the effective marketing techniques of the companies pushing their automated hardware and software solutions. However, the fact remains that simply relying on a vulnerability scanner on the network is not an effective security posture [5].

The most glaring weakness in an automated vulnerability scanner is that it is in fact automated. If the vulnerability has not been identified and input into the scanner's repository by the vendor, it is missed completely and will return a false negative [5]. Desautels also provides the following scenario.

A hacker decides to perform research against a common technology like your firewall. That hacker might spend minutes, months or even years doing research just for the purpose of identifying an exploitable security vulnerability. Once that vulnerability is identified the hacker has an ethics based decision to make. Does he notify the vendor of his discovery and release a formal advisory or does he use his discovery to hack networks, steal information and profit. If the hacker decides to notify the vendor and release an advisory then there is usually a wait period of 1-3 months before the vendor releases a patch. This lag time means that the vendor's customers will remain vulnerable for at least another 1-3 months, most probably longer. What's even more interesting is that this vulnerability may have been discovered previously by a different researcher that didn't notify the vendor. If that's the case then that probably means that the vulnerability has been in use as a tool to break into networks for a

while. Who knows, it could have been discovered months or even years ago? That type of unpublished vulnerability is known as a 0day [exploit] and is the favorite weapon of the malicious hacker. [5]

Another weakness of relying solely on an automated vulnerability scanner is that it only tests the network itself. It does not test the people within the organization. If the people within the organization can be compromised, no amount of money spent on vulnerability scanning will protect the organization from an intrusion by a hostile party.

The most effective technique for using an automated system to perform a vulnerability scan is in conjunction with a manual scan performed by an authorized penetration tester. This will reduce the frequency of false positives, and it also provides the organization with data on where its other security measures may be lacking.

THE FUTURE OF PENETRATION TESTING

Penetration testing as a security strategy is roughly 35 years old. The original penetration test can be traced back to the US Air Force's Multics Security Evaluation in 1974. [2] Since then penetration and security testing has evolved into what most consider a complicated but revealing and useful process.

Recently the future of penetration as it is known today has become less clear than it was two or three years ago. In December of 2008 Brian Chess, an executive at Fortify Software wrote in CSO Online:

People are now spending more money on getting code right in the first place than they are on proving it's wrong. However, this doesn't signal the end of the road for penetration testing, nor should it, but it does change things. Rather than being a standalone 'product', it's going to be more like a product feature. Penetration testing is going to cease being an end unto itself and re-emerge as part of a more comprehensive security solution. [4]

Mr. Chess also alluded to the fact that both Hewlett Packard and IBM have both purchased companies that specialize in developing penetration testing software for web applications. This indicates that software publishers, programmers and developers are starting to take security of their programs much more seriously and are going to attempt to write their code more securely following the secure development life cycle (SDLC). While there will still be a need for the code to have penetration tests run against it, it will be done as a part of the development cycle rather than after the fact.

Other professionals in the information security industry have taken umbrage to Mr. Chess' predictions and have proceeded to write papers which dispute this claim. One of Mr. Chess' critics, Ivan Arce, has written a twelve point rebuttal to Mr. Chess' column in CSO Online. Mr. Arce states that a practice that is 35 years old does not simply disappear or go through drastic changes in a single year [2]. This point is a valid one and aside from some groundbreaking idea in the world of programming, is likely to hold true. He also argues that even if all developers were to begin adhering strictly to the SDLC when programming new software packages, there is still existing and legacy software being used by organizations which is not likely to be replaced within the next year [2].

Some of Mr. Arce's points include that penetration testing is operational in nature and that simply testing an applications in a lab is not enough [2]. This is a very valid point and a strong argument. This is due to the fact that in a lab, products are set up to vendor specifications and the vendor is usually already aware of their application's vulnerabilities. In an actual live operating environment, this is not often the case. People often take shortcuts when setting up software. The individual performing the install might forget to enable a crucial setting during the setup. Executives within the organization might decide that some security measures do not make the applications features available enough to end users, and order the IT department to bypass security features. There is a whole range of issues that might come up when an application is used in a live operating environment that might not be duplicated in a lab environment.

Penetration testing is also a strategic methodology [2]. It allows an organization to see threats that cannot be duplicated in a lab. A successful social networking attack by an intruder will bypass security measures built into applications nearly every time. People will always be the weakest link within a security strategy. The SDLC does not account for this weakness.

Any individual knows that information technology is a constantly evolving industry. [2] The idea that this aspect will change the

slightest bit, if at all, is absurd. As new developments are made in the field, new opportunities for malicious attackers will continue to grow. New software and technology is very seldom released in a perfect state. This is due to a host of reasons, mostly because of limits on time due to release dates and pressure on programmers to provide their employers with a finished product. Simply because developers are starting to use the SDLC in their programming methodologies does not mean that hackers will just give up and stop searching for vulnerabilities to exploit.

There are also those companies that must comply with government regulations [2]. For instance any company working in the banking industry must perform a security audit at least once a year. This is a mandated aspect of doing business. As slow as the government is to get rid of any regulation, or laws regarding any aspect of business, it is not likely that penetration testing will be removed from the security landscape any time soon.

Finally with the recent financial crisis cybercrime is likely to be on the rise [2]. As programmers and developers begin to lose their employment the need to feed themselves and their family will begin to grow and some might succumb to the pressure to break the law, or at the very least do something unethical for money. Experienced programmers who have direct knowledge of designing and implementing secure software applications are a huge danger simply because they are even more aware of where certain vulnerabilities might be, and how to easily exploit those vulnerabilities.

CONCLUSION

Penetration testing is a vital part of any information security strategy. It provides an organization with information that will allow them to better understand how effective their security strategies are within a real world scenario. Advances in the technology are bringing more and better automation to penetration testing. These advances should not be considered as a complete solution though due to the fact that as the technology changes, so do the methods of attack. A professional penetration tester will always be able to perform a specific attack before an automated system will be able to do so.

The world of information security is changing and evolving, and along with it the standards in penetration testing. An organization should not rely solely on one method of testing their strategies that are in place to protect their confidential data. Black box, white box, and automated strategies all have their place within the realm of penetration testing. They should be used in conjunction with each other, and not as a standalone methodology of testing an organization's security strategy.

RECOMMENDATIONS

Organizations should:

- Perform black box penetration tests to assess their exposure to attacks which involve social engineering, reconnaissance, and where the weaknesses lie within the public domain

- White Box testing to discover where all of their known vulnerabilities are. This includes but is not limited to, automated vulnerability assessments backed up with manual scans performed by a professional penetration tester.

- These tests should be performed at regular scheduled intervals to make sure that as new technologies emerge and are placed within the business environment, that the systems of the organization remain secure.

- Be sure that they are in regulatory compliance with the organization's respective government regulators regarding the use and frequency of penetration testing and vulnerability assessments.

REFERENCES

1. Abarca, David (2005-2007), Personal Communication/Lectures.

2. Arce, I. (2008, December 18). 12 Reasons Penetration Testing Won't Die, in CSO: The Resource for Security Executives. Retrieved

8:00, February 1, 2010, from http://www.cso.com.au/article/270839/12_reasons_penetration_testing_won_t_die?rid302

3. Chess, B. (December, 2008). Penetration testing is dead, long live penetration testing, Retrieved 9:18, February 1, 2010, from http://www.ncc.co.uk/article/?articleref=310511&hilight=brian+chess

4. Bayles, A., Butler, K., Collins, A., Meer, H., et al. (2007). Penetration Tester's Open Source Toolkit. Burlington, MA: Syngress Publishing

5. Desautels, A. (2009, January 7). Network Vulnerability Scanning Doesn't Protect You. Retrieved 8:05 February 1, 2010, from http://snosoft.blogspot.com/2009/01/vulnerability-scanning-doesnt-work.html

6. Goodchild, J. (2009, February 4). Social Engineering: Anatomy of a Hack. Retrieved 7:45 February 1, 2010, from http://www.infoworld.com/d/security-central/social-engineering-anatomy-hack-693

7. Herzog, P. (2006, December 13) Open-Source Security Testing Methodology Manual v2.2. Retrieved 7:50, February 1, 2010, from http://isecom.securenetltd.com/osstmm.en.2.2.pdf

8. Penetration test. (2010, February 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 8:45, February 1, 2010, from http://en.wikipedia.org/w/index.php?title=Penetration_test&oldid=341322433

9. Srp, Shannon (2008), Personal Communication.

10. Sullivan, D. (2009, February) Social Engineering in the Workplace. Retrieved 8:10, February 1, 2010, from http://nexus.realtimepublishers.com/ESMWSv3.php