

Internet Penetration Testing: A Seasoned Perspective

SystemExperts Corporation

Brad C. Johnson

Abstract

There are many different types of vulnerability assessments. Penetration Analysis focuses on understanding the vulnerabilities of components that you've made available on the network as seen from the perspective of a skillful and determined attacker who has access to that network. We call that set of vulnerabilities, your *exposure profile*. This type of assessment is designed to help protect yourself against potential hacker attacks. A Penetration Analysis is one of the few effective means of *proving* whether a given networking infrastructure actually satisfies the security requirements of an organization.

Since 1994, SystemExperts has been performing penetration tests for some of the largest multinational financial institutions, the premier networking companies, and the most successful dotcoms. The content for this white paper is distilled from our experiences in conducting nearly five hundred tests and in helping clients to respond to some of the most public and thorough hacker attacks. The author of this white paper is the originator of the methodologies and techniques that SystemExperts has been using in this area since the mid 1990s.

Inside

This white paper is for anyone interested in understanding the main issues related to an Internet Penetration Analysis. This type of activity often goes by many names: Internet Readiness Assessment, Tiger Team Attack, and Intrusion Analysis just to name a few. The paper covers topics that are appropriate for a senior manager (e.g., what can you accomplish with an assessment like this or how do you tell the difference between this type of test and others?). It also addresses topics of interest to a technical person (e.g., what are the major technical issues that I need to consider or how do I organize this type of work or how do I assess if somebody is doing a good job of this type of testing for me?).

SystemExperts Corporation

Boston New York Washington D.C Tampa
San Francisco Los Angeles Sacramento

Toll free (USA only): +1 888 749 9800
From outside USA: +1 978 440 9388

www.systemexperts.com
info@systemexperts.com

Introduction

In 1993, Dan Farmer and Wietse Venema wrote a seminal paper in the area of network security assessment named “*Improving the security of your site by breaking into it*” (it was written just before they released their well publicized vulnerability assessment application, called SATAN). This paper helped demonstrate several important concepts:

- Commonly deployed protocols and services have security problems that allow intruders to do bad things to systems
- Neither physical access nor special administrative privileges are necessary to remotely access data or programs on a host
- Common protocols and services can be used in unintended ways like using the mail protocol to establish an interactive session on a host

Since then, hundreds of programs and techniques have been developed to help organizations assess weaknesses in their deployed network infrastructure. These same programs can be used to exploit those weaknesses.

A Wide Spectrum of Related Analyses

There’s an old saying that states “there’s a time and place for everything.” That is an appropriate reply when somebody asks the question “What kind of a security analysis do I need?” In trying to understand the risks to your networked resources, there are a number of different perspectives to be considered. Each perspective reveals its own set of relevant security concerns. Typically, a combination of projects is used to assess important application environments. These projects might include:

- Penetration Analysis:
What exposures, at a system or service level, can a determined intruder exploit?
- War Dialing:
What resources are reachable and potentially exploitable through the common phone network?
- Web Application or Web Content Review:
What vulnerabilities are available solely through the Web components, is it possible for web application users to masquerade as others, and can a web application user escalate his privileges to gain control over the host system itself?
- Security Audit:
Are the documented security policies, practices, and procedures consistent with the business requirements? Are they consistent with best industry practices? Has the environment been implemented as designed?

- Architectural and Design Review:
Do the overall architecture, data flow, controls, connections to business partners, and dependencies on outside services (e.g., code development, managed services, ISPs), create an environment that can reasonably be secured?
- Code Review:
Have the applications been written to eliminate typical security vulnerabilities and to be security-aware?

As you can see, each of these projects provides insight into a distinct part of your computing infrastructure. Each one fills a crucial piece of your overall information security puzzle. It is important to remember that security is a continuous process. It starts with an idea and continues through design, implementation, testing, deployment, and on-going operations. Consequently, the appropriate mix of security projects will vary depending on the environment’s stage in its development life cycle. Changes to your application or changes in the deployment environment will affect your security profile. Every time significant changes occur, you need to reassess your security.

Understanding the Context and Terminology

The Puzzle: Networks, Protocols, Services, Hosts, & Applications

Every network environment is comprised of hundreds, if not thousands, of separate components. To understand the security of such a system, we must focus on the network components that make up the physical network, the application and transport protocols, network services (e.g., mail, file transfer, and authentication mechanisms), the individual hosts (each running its own specific OS, libraries, and applications), and the business applications themselves.

Each of these categories has its own set of security issues that, to a large degree, are managed separately from the others. The configuration and management of your host OS is a separate and distinct function from managing your business application, which is separate from managing the network protocols. You may have done a good job in hardening your host OS, for example, but you may still have security problems caused by inappropriate configuration of the application. Therefore, it is important to ensure that your security assessment activities specifically address each of these important areas.

Intrusions: How?, Why?, and What?

Intrusions are more successful and happen more often than we would like. So what can you do about that? Since it is

impractical to take resources off of the network, it is important to accept that intrusions are likely to occur. If you get past that mental hurdle, you are in the right mindset to prepare for handling a problem.

So what do you need, at a minimum, to be prepared for an intrusion? You need to have some tangible process or mechanism to handle the following:

- **Detection:** Can you tell that an intrusion has happened?
- **Escalation:** When an intrusion has happened, do people know what to do?
- **Recovery:** How do you fix or replace the components that have been damaged in the intrusion?
- **Assessment:** Can you figure out how and why the intrusion occurred?
- **Remedy:** Do you know what to do to prevent this type of intrusion from recurring?

Lower Level Exposures Create Higher Level Problems

Every time there is a brand new “bad” exploit, there are many people who copy it. Fixes for these types of obvious and well-known exploits are usually available quickly, sometimes, measured in hours. Most organizations make protecting themselves against these high profile vulnerabilities a high priority. Unfortunately, most organizations do not effectively deal with the large number of lower level exposures that are discovered each month. It is also true that there are many easy to use exposure and exploit tools that you can run (e.g., ISS Network Scanner, SATAN/SARA, sscan, nessus, nsat, Typhoon, NetRecon, and CyberCop to name a few) to assess your current vulnerabilities. So what do these facts have to do with each other?

- Most assessment tools perform a one-shot analysis. They look for specific exploitable vulnerabilities. None of them are capable of checking for complex combinatorial exposures.
- Many serious exploits are a result of combining several low or medium level vulnerabilities (and tools and techniques) in such a way as to create a high level problem.
- Fixing well-known high level vulnerabilities is necessary but by itself, not sufficient. You also need to address the lower level protocol and software version issues because fixing these will help to eliminate problems that cannot be anticipated.
- Most tools are focused on what can be seen, not on what you don’t see (more on that in the next section).

What You Don’t See Matters

We recommend that our clients run off-the-shelf assessment tools. However, these are not a substitute for the expertise and experience a professional can bring to the interpretation of results from these tools. It is important to understand not only what you’re seeing but *what you’re not seeing*.

When performing a Penetration Analysis, it is essential to use some type of network sniffing package to monitor all of the network traffic between the attack system and the targets. When you send a request to the target environment, you need to see how “it” gets there and what comes back. Some of the things that you need to look for include:

- Are the packets being filtered by an intermediate ISP, by intrusion detection software, or by perimeter services (e.g., a router, gateway, or proxy server)?
- Does the response come back from the target system, or from some other system? Is that because of load balancing, a proxy server, or has it been redirected to some other entity entirely?
- Does the intended target service respond to a request, or is some other service responding in its stead? Are services running on standard ports? Does the target environment “make sense” (e.g., does it look like they are running a Microsoft service on a UNIX box)?
- When you perform a scan of the target host using different scanning methods, do you get the same results?

Paying attention to the details of what you don’t see can tell you just as much about an environment as what you do see. This type of judgement only comes from years of experience with a wide array of operating environments and, for a practical matter, cannot be done with software or junior staff.

Case Study

Let’s look at a case study to see how a Penetration Analysis really works. In this analysis we were asked to identify the security risks associated with a Fortune 500 company’s main web site – a single system. Note the sequence in which the vulnerabilities were discovered and the multiplicative effect of combining them.

In a typical break-in, progress is made through an iterative process. It is not uncommon for these steps to be repeated many times during a Penetration Analysis:

1. network probes
2. traffic analysis

3. research and analysis to refine the exploit opportunities and develop tools (non practitioners often don't realize that this step consumes most of the time)
 4. confirmation of the vulnerability
 5. demonstration and documentation of the exploit
- This particular example had three common problems: loose firewall and filtering characteristics, disclosure of state information, and ineffective authentication. Each of these problems and the resulting vulnerabilities are explained below.

The Initial Toe-Hold

The main web site withstood over two solid weeks of analysis and probing. They had done a good job of protecting it. Like any other determined intruder, we then searched for the path of least resistance by looking for weaker systems in close network proximity to the target system. By finding and combining a number of lower level vulnerabilities on an "adjacent" system, we were able to gain an interactive login session and make changes directly to the main web site itself. Here are the series of problems that we exploited to reveal this significant problem.

1. **HOST DISCOVERY:** One way to "find" hosts is to do some type of protocol probe. We chose an ICMP ECHO probe. While a non-response doesn't imply that a system isn't there (e.g., the request may be blocked by an ISP or firewall), a response normally indicates that indeed the system is there. We chose this method because, at the time, it was likely to be undetected and it is very quick. Since no intermediary system stopped the probes, we were quickly able to find six systems in the same IP address space as the target main web site. In most instances, allowing ICMP ECHO requests is a low-level security concern.
2. **HOST SCANNING:** Knowing that the customer was acutely aware of our activities and had asked us to be as invisible as possible, it was important to try to discover what was running on those systems without causing blatant network activity or triggering Intrusion Detection System alarms. Since SNMP is available on almost all systems and is often not included as a high priority in intrusion detection management, we probed all of the hosts for SNMP agents. History shows that most people either do not change the default SNMP database (MIB) password (community string), or if they do, they change it to something obvious. Even though they had changed the default community string, within an hour, we were able to guess what they had used for the SNMP agents on these systems and start performing probes of the SNMP databases. By looking at this database, which contains hardware, version, and state information, we found that there had been many connections between the main web host and one other system (by looking at the TCP

connection table). Changing the community string from its default value is usually considered adequate protection, however, they changed it to something too obvious.

3. **PRODUCTION QUALITY vs. NON PRODUCTION QUALITY:** Having now discovered that there was significant traffic between this system and the main web site, we carefully probed the second system to determine if the system services were vulnerable to attack. In the same way that SNMP agent read requests are typically not included in intrusion detection management, unfortunately, neither are HTTP GET requests. Most sites do not monitor their web server logs for odd requests. We quickly found that this system was also running the same web server as the main web site and sent a series of HTTP GET requests looking for known server-side script problems. We found one that allowed us to execute arbitrary commands. As it turns out, this second host was a staging server used by the company to beta-test upgrades before loading to the main web site. So the problem we found was that the Internet-reachable *non-production host was not as well hardened as the production host.*
4. **TRUST RELATIONSHIPS:** Because these two systems needed to communicate with each other on occasion, there was a network trust relationship between them. Unfortunately, the trust was excessive. Because of that, it was easy to attempt an interactive login session from the staging server to the main web host. While we did not have a known username and password, the main web site allowed us access as an ordinary *guest.*
5. **NO CONTROLS ON OUTGOING TRAFFIC:** We found that the site had no controls on the outgoing traffic and this allowed us to make a connection back to our site. This lack of control could be easily used by an attacker to launch attacks against other sites. It is only beginning to become standard practice to configure firewalls to control outbound traffic just as closely as inbound. Some of the recent Distributed Denial of Service attacks (DDoS) have demonstrated that the *apparent* attacker may not be the attacker at all. Instead, some other unfortunate site that has been compromised is used as an intermediary. Unfortunately, the intermediate didn't abide by that rule and they allowed obvious "bad" things out.

In our case study, we were able to initiate an interactive session back to our own systems and start editing Web pages from the comfort of our office. This should have been blocked by the firewall.

Ouch. All of this was done undetected by using common protocols and by combining low to medium level vulnerabilities to create a situation where a very bad exploit was possible.

Paying Attention to State

After finding these system-level vulnerabilities, we turned our attention to the web application itself.

HTTP is a stateless protocol. However, most business applications that run on the web need to keep track of what's going on (i.e., keep state). So what do people do? Well, state can only be stored in two places: on the server and on the client. However, it can be maintained in a variety of ways. Some of the common ways are to save transaction state on the server, to save it in cookies on the client, to include it as part of some session ID that is stored in the HTML pages themselves, or to include it in the URL that is sent to the client.

If the application is storing any state on the client, then the big security question is, "If this data is sensitive, how does the server make sure this information is kept private?" Unfortunately, many server applications do not use well-known methods for securing sensitive data. Too many people make the mistake of inventing their own algorithms or designs that often ignore important security issues. Let's take a look at an example of how the client in this case study made this mistake.

The organization in the case study designed cookies that contained sensitive information that could be viewed at the client system (browser) because the designers made two mistakes: 1. They did not consider the authenticated user of the browser to be a threat and 2. They did not think the cookie was accessible on the client system because it was a secure cookie in an SSL session. Many organizations mistakenly believe that using SSL will give them security without having to design secure applications. It is true that SSL does not store SSL cookies to disk but it does store them in memory. We used an in-memory cookie viewer to look at the sensitive information in the cookie. "In memory" doesn't mean un-viewable. The fact is that your browser knows how to read cookies in memory so anyone willing to write some simple code can, as well.

A Case of Mistaken Identity

This client was also using a Web server package that created dynamically generated HTML pages that included, among other things, a Session ID. This Session ID was comprised of many different parts: a time stamp, a host ID, the user ID, and some private server application information. By making repeated queries to the server, we were able to generate a set of Session IDs that allowed us to analyze them as a group. We discovered that part of the Session ID seemed to be our user ID. Unfortunately, many server applications store state on the client (e.g., this Session ID) and they do not check the data when it comes back to ensure it has not changed unexpectedly. So, we wondered if the server checked to make sure that requests were coming in from the same legitimate user.

We logged in through the site's Basic Authentication page and started a transaction. After the first request, once we got the dynamic page back to our browser, we saved the file to disk. We then used a common editor to change the user ID part of the Session ID to a different ID. We then re-loaded that changed page back into the browser and sent it back to the server. Lo and behold, we were now masquerading as this other user ID and could perform any transaction that user was permitted to perform. Unfortunately, since this site was a brokerage account and we were soon able to demonstrate the ability to perform transactions valued at over a billion dollars - that's a lot of zeros. The problem was that while the application was effective at initial authentication, it failed to re-authenticate the user properly.

In summary, the initial Penetration Analysis revealed the ability to arbitrarily change pages on the main web site. Additional activities, focused on the web infrastructure, revealed both a leakage of confidential information and the ability to execute transactions masquerading as any valid user.

Separating the Professionals from the Amateurs

The following sections address some of the more management-oriented issues of performing a Penetration Analysis. Whether you are performing this work yourself, are looking to hire somebody to do it, or are part of a project where a third party group has already been chosen, these are characteristics of the engagement that should be explicitly considered.

A Matter of Trust: Don't Hire Hackers

One of the most important things to find out about any firm you are considering for a Penetration Analysis is whether they use external resources (in particular, hackers) to help with the work. If the answer is yes, you should walk away. No, you should run away because *integrity counts*.

Hackers are interested in making themselves look good, in making other people look bad, and they are interested in making the environment as insecure as possible for potential future break-ins (we've seen this really happen). Even if trustworthiness was not an issue, the vast majority of hackers do not have the discipline or professionalism to switch from the search for the "big-exploit" to the systematic profiling of all problems, even "uninteresting" ones.

Exposures Can Be Proven Without Catastrophe

Another important part of scoping any Penetration Analysis is deciding what constitutes success for demonstrating a particular

vulnerability. While it is overkill to predetermine exact success criteria for each and every test (out of tens or hundreds that will be attempted), it is important to agree upon a philosophy of success.

Our philosophy is one of *non-destructive proof*. We prove that a particular exploit is possible without causing unexpected damage to the target environment. This is probably best explained by using an example. Let's say that you want to prove that you can change the running state of a network device (i.e., turn it on or off). One of the ways to do this is via the SNMP protocol. If one could learn the SNMP community string (password) for the SNMP MIB, then you would be able to change the state of the machine from "on" to "off." However, knowing the community string would give you the ability to perform other management functions, like setting the time, as well. If we could change the time just a little instead of changing the state of the device from "on" to "off", we can prove that the larger exploit is possible without causing disruption to the environment that is being tested.

Special Case Testing

Sometimes, a Penetration Analysis will be focused on a small set of hosts, or services, or special resources (e.g., DNS name server or a router or a firewall). Other times, it will be a broad "find-what-you-can-find" look at the entire environment. No matter where in the spectrum a particular analysis falls, there are several types of testing that need to be dealt with as special cases. These are:

- Denial of Service (DoS) – forcing a component to become unavailable¹
- Distributed Denial of Service (DDOS) – usually associated with making a host (or set of hosts) unavailable by overwhelming the network connected to it with various types of network traffic
- Social Engineering – trying to "trick" organizational staff into giving sensitive information or using malicious or nefarious methods to obtain this information

¹ Keep in mind that there are many different types of components that one might want to try and make unavailable. They can include things such as a host, an application, a port, a particular process, a file, a host device (e.g., a modem), or the hosts' name. The ways in which a component can be made unavailable vary widely and include methods such as overwhelming the component with volumes of requests, changing its configuration, overflowing a buffer to make it "crash", masquerading as another user, or using up available resources that it needs.

Since the methodologies use to perform these tests, the approvals needed, and the level of coordination necessary while the testing is underway, are completely different than a Penetration Analysis, we strongly recommend that these types of assessments be structured as distinct projects.

Conclusion

Penetration Analysis should be a regular part of your security program. Minor problems, the kind of things that are not picked up by the off-the-shelf security tools, can be combined into serious exploits. Experience counts: recognizing what does not come back from the probes can be more important than what does. Finally, when talking about the security of your IT infrastructure and protecting data about your customers and your business as a whole, the testing organization's integrity counts most of all.

About SystemExperts Corporation

Founded in 1994, SystemExperts™ is the premier provider of network security consulting services. Our consultants are world-renowned authorities who bring a unique combination of business experience and technical expertise to every engagement. We have built an unrivaled reputation by providing practical, effective solutions for securing our clients' enterprise computing infrastructures. Through a full range of consulting services, based on our signature methodologies, we develop high level security architectures and strategies, design and implement security solutions, perform hands-on assessments, and provide a wide variety of both on-site and off-site services.

Our consultants are frequent speakers at technical conferences around the world. Our courses on penetration testing, wireless security, secure electronic commerce, intrusion detection, firewalls, VPNs, and NT/Windows 2000 security at Usenix, SANs, NetworkWorld-Interop, CSI, and InternetWorld are among the most popular and highest rated because our consultants bring years of practical experience to bear. In addition, our consultants have been technical advisors and on-air guests for CNN, Dateline NBC, WatchIT, and CBS News Radio and we wrote the authoritative reference work on Windows® 2000, the Windows® 2000 Security Handbook (Osborne McGraw-Hill).

We provide consulting services on both a fixed-price and time-and-materials basis. We are flexible and we can structure any project so that it is just right for you. You will appreciate the difference of working with genuine experts who are committed to earning a long term partnership with you by over-delivering and providing unmatched personal attention.

Our consultants provide a wide range of services. Below is a sampling of areas in which we advise our clients.

Security Consulting

Our experts conduct network and host security analyses and a wide variety of penetration tests. In addition, using our signature workshop-style methodology, our consultants will work with your team to review the security of applications or systems in their full environmental context. During these comprehensive reviews, we will thoroughly explore the business as well as technical issues and we will balance the cost, schedule, and operational constraints of each technical alternative. Many of our clients include these reviews as the jumping off point for planning and prioritizing their security initiatives each year.

Security Blanket & Emergency Response

It is not a question of *if* your organization will be the target of a hacker, it is only a question of *when*. Preparation minimizes the impact of an attack and ensures a rapid recovery. Our security experts will work with you so you'll be well prepared and if you are attacked and web sites or critical business resources are compromised, we have the experience and expertise to respond to the intrusion in a pragmatic, professional manner. Our emergency response teams quickly assess the situation, properly preserve evidence for use by law enforcement, lock out the attacker, and develop and help implement a plan to quickly regain control of the IT environment.

Intrusion Detection and Event Management

In security it is axiomatic that what you can't prevent, you must detect. We have helped dozens of companies (including several of the largest companies in the world) develop comprehensive intrusion detection plans and implement them.

Technical Skills at the "Guru" Level

Sometimes getting the details right is all that counts. We help our clients to resolve the toughest firewall, VPN, wireless, PKI, authentication, authorization, networking, and configuration problems in NT/Windows 2000, Unix, and heterogeneous environments. In addition we frequently perform code reviews of critical applications and web sites.

Security Policy & Best Practices

Security starts with understanding the underlying business and regulatory requirements. Security policy is the means by which these requirements are translated into operations directives and consistent behaviors. We assist organizations in developing and updating policies and identifying where clients' current security practices, policies, or procedures differ from best industry practice.

Security Stolen/Lost Laptop Analysis

Many organizations expend considerable effort and resources to secure their internal networks, key computing resources, and connections to the Internet. Few recognize that a significant amount of their most proprietary information is traveling around the country on the largely unsecured laptop computers of road warriors and senior executives. SystemExperts' laptop analysis will help you to understand the potential risk of a lost or stolen laptop and what measures you can take to mitigate those exposures.

VPN and Wireless

Certain technologies like VPN and Wireless are becoming ubiquitous and yet most organizations don't know how to properly secure them. We do - and we can help you.

To learn more about how SystemExperts can put its expertise to work for you, contact us today at +1 . 888 . 749 . 9800
Boston Los Angeles New York San Francisco Tampa Washington DC Sacramento
www.SystemExperts.com info@SystemExperts.com