Interested in learning more about security?

SANS Institute InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Vulnerability Identification and Remediation Through Best Security Practices

These days, we hear a lot about Pen-Testing (attempting to penetrate a system's security layers in order to demonstrate security risk. While these types of studies are useful and effective, they tend to require specific skills, training, and experience in areas that network professionals are usually not exposed to. This paper looks at Vulnerability Identification Studies which focus on identifying the enticements, common vulnerabilities, and information leakage, the things that account for most of the risk to IT (Infor...

Copyright SANS Institute

Author Retains Full Rights

AD

Vulnerability Identification and Remediation Through Best Security Practices

BJ Bellamy Jr.

December 7, 2001

Table of Contents

1. **Introduction** - The Problem
2. **Methodology** - The framework used to pursue the goal of more effective security
   2.1 Footprinting - Indirect information gathering
   2.2 Scanning - Direct general information gathering
   2.3 Enumeration - Direct and specific information gathering
3. **Best Security Practices** - The solution
4. **Reference**

1. Introduction - [Top of page]

> Only by understanding how attacks work and what an attacker does to compromise a machine can a company position itself so that is can be properly protected. (Cole, 1)

These days, we hear a lot about *Pen-Testing* (attempting to penetrate a system's security layers in order to demonstrate security risk. While these types of studies are useful and effective, they tend to require specific skills, training, and experience in areas that network professionals are usually not exposed to. This paper looks at *Vulnerability Identification Studies* which focus on identifying the enticements, common vulnerabilities, and information leakage, the things that account for most of the risk to IT (Information Technology) that we face today.

The goal of a *Vulnerability Identification Study* is not to actually penetrate a system or demonstrate the degree to which a weakness could be exploited. Rather, it is to allow authorized technicians to see a system from an attacker's perspective so it can be fixed or the risk of intrusion minimized by applying common *Best Security Practices*.

In effect, two tasks are addressed; first is to minimize an intruder's incentive by limiting the enticements that lead an intruder to probe a system; second, to maintain a system so that it is less susceptible to actual intrusion and better prepared to recover from an attack.

This paper provides only a beginning framework from which you can continue to build and refine your security efforts. Many other items must be taken into account and effectively addressed in order to implement a comprehensive security plan. Some of these topics, which are not addressed in this paper, include unix hosts, modem attacks, social engineering, intrusion detection systems, firewalls, encryption, and the detailed hardening of operating systems and applications. But, the basic principal of removing enticements, while both strengthening and monitoring systems through the application of appropriate Best Security Practices, is universal.

All of the tools illustrated in this paper are *freeware* (software that is provided free for authorized and appropriate use, with little or no support provided by the developer). The use of freeware is usually predicated on an agreement or understanding that the technician absolves the author or distributor of any responsibility for damage or disruption caused or incurred by its use or misuse.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Many of these tools are, in fact, *hacker tools*. While many people steer away from *hacker tools*, they provide many significant advantages **if** acquired from reputable sources, tested for virus infection or tampering, and used appropriately by authorized technicians for authorized purposes.

The advantages of using *hacker tools* include:

- The technician can see their systems from the perspective of a potential intruder rather than the more sanitized view that products offer. By using the same tools that most intruders use, you gain the advantage of their perspective.
- Use of these freeware tools can be an instructive exercise for gaining a much deeper understanding of networks and systems. These tools put you "down in the dirt" where you will gain invaluable experience and insight to information technology which you can apply across your range of activities.
- The tools illustrated in this paper are all command-line tools (launched from a command prompt), rather than GUI (Graphical User Interface) tools. From the perspective of a network administrator, GUI tools are difficult to script into larger processes and are less flexible compared to command-line tools. Also, GUI tools tend to focus on scanning a single target host at a time, while scripting command-line tools allow you to more flexibly bulk scan hosts on a network(s) easier. The text output that command-line tools usually produce also lends itself to scripts that use the text as input for exception reports.
- A variety of specialized tools minimizes false readings. Too often, a single tool or product will generate findings which may be misleading due to the context or the single vector they employ. As with a master carpenter, you "measure twice and cut once" - scan with more than one tool and reconcile the results into a much more valid single finding.
- A reduced expense for software. From a budgetary perspective, you will get much more "bang-for-your-buck" by investing in training and the man-hours to perform these tasks than in purchasing "all-you-need-with-one-click" products. The experience gained from beginning with freeware will be invaluable whens you evaluate commercial products for their range of functionality and usefulness.

If we are going to successfully protect our systems, we need to know something of the attacker's motivations and goals.

While most people assume an intruder's goal is unauthorized access to the victim's files, intruders actually seek to gain unauthorized access to a system for a variety of reasons, including the following:

- **Information or data.** The obvious goal is to gain access to information (raw data within context) or data (raw material not with-in context - not meaningful) contained in a system. Often people mistakenly think that this is the only, or even most likely, reason an intruder would be interested in their system. You see this behind the "I have nothing of importance or that I want to hide on my computer" fallacy. They fail to realize that their machine offers much more to an intruder than simply the files it contains.
- **Disk space.** Often, an intruder's goal is the available disk space on a machine. Intruders will create archives of files (pictures, stolen programs, or data) on a system so that their own systems do not contain evidence of their activities. It is not uncommon for hackers to set up *warez* sites (archives of stolen files that are exchanged among hackers) on machines they have broken into. *warez*.
- **Processing power.** Hackers will not waste their own computing power cracking a password file if they can use your computing power. A hacker can quite easily configure a machine to work its heart out cracking a stolen password file while, its user goes about their normal work, unaware that they have been compromised and made into an unknowing accomplice.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

- **Zombies.** One especially effective method of malicious attack is to enlist the help of multiple zombied computers. The intruder will program a machine, perhaps hundreds, to perform a particular type of attack upon demand. In this way, a single intruder, with very limited computing resources, can command a force of hundreds or more to coordinate specific and devastating attacks. The owners of the zombied hosts are not aware they are being compromised in this way, but they may still be responsible due to their lack of reasonable security.

So, every host on your network is an attractive target for many reasons. Many are not listed here and many are not obvious to the system owners. Effective controls must therefore be in place for all systems (hosts, appliances, devices, etc.), regardless of their intended function or perceived value.

2. Methodology [Top of page]

In order to stay focused on the process of identifying those conditions which increase the risk of intrusion, we follow a methodology - a general process that encourages efficiency and effectiveness.

There are several examples of useful and practical methodologies aimed at IT security available on the Internet. These methodologies help keep the practitioner focused on a rational and effective course of inquiry during a vulnerability assessment.

Our goal is to first identify **enticements** (instances that would entice a potential intruder to probe further), and **low-hanging fruit** (actual holes in your systems which provide easy to exploit unauthorized access into or control of a system).

The second goal of this methodology is to **identify and verify vulnerabilities**, rather than to demonstrate our ability to exploit them. Consequently, the issues of actual penetration or intrusion, or the manipulation of a compromised host, are not addressed in this paper.

Third, we continually apply **Best Security Practices** in order to minimize the items we identify in the first two steps, and to better prepare to deal with both unsuccessful and successful attacks.

From a conceptual perspective, once we become familiar with the threats, it becomes clear that following applicable Best Security Practices is most effective way to reduce the risk and minimize potential intrusion or malicious damage.

From a sequential perspective, this methodology becomes a cycle

1. Identify vulnerabilities, enticements, information leakage, and low-hanging fruit,
2. Institute or adjust the appropriate Best Security Practice as remediation and proactive defense,
3. Continually perform appropriate Best Security Practices, and
4. Start this cycle over.

> After you have identified a risk, you can do one of three things with it: You can accept it, you can reduce it, or you can insure yourself against it. Security does not have to be perfect, but the risks have to be manageable. (Schneier, 384)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

2.1 Footprinting - [Top of page]

Footprinting involves indirect probing, which is - interrogating resources on the Internet for information about your systems, without actually *touching* them. You are looking to discover what a potential attacker can also discover without your knowledge.

Tool: whois - The whois tool queries one of the authoritative Domain Name Service (DNS) registration services for information on a specific domain registration. Keep in mind that this is the Internet Domain Naming Service, not Microsoft's LAN based Domains.

- **Whois.exe** is a command-line tools. Running **whois -h** will display the syntax help screen for the whois program
- **VeriSign Whois** - http://www.netsol.com/cgi-bin/whois/whois - The networksolutions DNS database is a prime source for information. This service has recently been acquired by VeriSign.
- **ARIN** - http://whois.arin.net/whois/index.html [As ARIN states on its About ARIN page, "ARIN is a non-profit organization established for the purpose of administration and registration of Internet Protocol (IP) numbers for the following geographical areas: North America, South America, the Caribbean, and sub-Sahara Africa."
- **InterNIC** - http://www.internic.net/whois.html. As InterNIC's home page states, "This website has been established to provide the public information regarding Internet domain name registration services and will be updated frequently."

The *whois* program is illustrated below:

```
D:\>whois TheAuditor.net
[The disclaimer has been removed for brevity, and the details have been altered to
describe a fictional organization]

Registrant:
   Office of the Auditor (THEAUDITOR-DOM)
   1234 South Street
   River City, Kentucky, 12345  US

Domain Name: THEAUDITOR.NET

Administrative
   Contact: Smith, Joe (JS95602) Joe@Smith.com
   The Auditors
   1234 South Street
   River City, Kentucky, 12345
   123-123-1234    (FAX) 987-987-9876

Technical Contact: Morry, Michael (MM22341) Michael@Morry.com
   OnYourSide ISP
   123 Cold Harbor Road
   River City, Kentucky, 12345  US
   123-123-1234    (FAX) 987-987-9876

Billing Contact: Palmore, Barbara (BP2651)
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
   Barbara@Palmore.com
   OnYourSide ISP
   123 Old Harbor Road River City, Kentucky, 12345
   123-123-1234 (FAX)  987-987-9876
   Record last updated on 15-Jun-2001.
   Record expires on 1-Aug-2003.
   Record created on 1-Aug-1999.
   Database last updated on 28-Aug-2001 02:16:00 EDT.
   Domain servers in listed order:
     MyNameServer1 127.0.0.1
     MyNameServer2 127.0.0.2
```

Findings:

Here we see that several names are provided, along with contact information, most of which is not needed here. The names can be used for social engineering and the phone numbers can be used for war-dialing (scanning for modems which might provide unauthorized access). An intruder would also assume that these individuals have above average authority, and that makes them attractive targets.

Not only are the names leaked, but by providing an actual staff email address, you are probably leaking the naming convention for network login accounts. Now when an intruder identifies your CEO as Bill Jones and his email address as BJones@MyNetwork.com, it is safe to assume that the login account is BJones.

The authoritative DNS name servers and their IP addresses (MyNameServer1 127.0.0.1 and MyNameServer2 127.0.0.2) are also clearly disclosed. While this is unavoidable, understand that a potential attacker will assume that these IP addresses either point to your internal (though publicly accessible) networks, or networks close to your internal network.

Many attackers begin by trolling (randomly looking for potential targets) when a bit of our information leakage draws their attention to portions of our systems that would otherwise gone unnoticed. Attacker may not have been focused specifically on *our systems* from the start.

Solution:

- List position titles rather than actual staff names. This also makes management easier since you will not need to update this information each time the position is reassigned.
- Setup an email address specifically to be used here. For example, DNSAdmin@TheAuditor.net
- Use an 800 phone number in order to guard against war-dialing.
- Use post office box addresses or central office locations. Use a location that is intended to be known to the general public rather than the location of your systems.

As you will see repeated throughout this paper, an important principle is to minimize any advantage you give to a potential intruder. This may not stop them, but why make it easy?

Tool: DNS Zone Transfer - Technique used to identify potential targets that an intruder might not have otherwise have noticed.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

DNS servers maintain lists of the hosts for which they can resolve machine name to IP address or the other way around. In many cases, DNS servers exchange their "zones" of information with one another. Requesting a DNS's zone information is called a **zone transfer**.

The problem is that DNS servers are usually configured by default to allow any host to request a full copy of its zone information. This list exposes to potential attackers hosts and IP address segments that they might not otherwise have noticed. The names of these systems often betray their primary function or in some way leak exploitable information.

DNS Zone Transfer is illustrated below:

When nslookup is run, it lists the current DNS host.	**D:\>nslookup** Default Server: nameserver1.com Address: 127.0.0.1
The server command sets the (IP or host name) DNS server targeted for connection and interrogation. This can be found via whois.	**> server 127.0.0.1**
Help will dump the supported arguments and their syntax.	**> help**
[type=any specifies that all records are desired]	**> set type=any**
The ls command requests a list of all DNS records from the current DNS server. The -d switch lists all records. The output is piped a file for later viewing. Note that IP_ means the IP address of the target host.	**> ls -d TargetDomain.com > IP_Zone-Transfer.txt** [nameserver3.com] AuditReview.com. SOA nameserver3.com Fred.nameserver3.com. (2000180200 89900 27920 608300 7300) AuditReview.com. NS pluto.com AuditReview.com. NS dnsf2.com AuditReview.com. A 127.0.0.21 IDSConsole A 127.0.0.4 mail3 MX mail.com database CN 127.0.0.21 firewall A 127.0.0.99 homeoffice A 127.0.0.3 www.police CNAME zina.com

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

	127.0.0.64 kge NS name1.kge.com name1.kge A 127.0.0.90 kge NS name2.kge.com ftp2.kge A 127.0.0.26 main_ids A 127.0.0.98 AuditReview.com. SOA nameserver3.com Fred.nameserver3.com. (2000180200 89900 27920 608300 7300)
Exit the nslookup.exe program and drop back to DOS.	**> exit**

Findings:

The zone transfer above reveals much about the hosts that this DNS server "manages." For example, those listed as NS are other DNS Name Servers, which might include even more potential targets for exploitation. Those listed as MX are post offices.

Notice how the names of some hosts are too informative (information leakage). The device named "firewall" can now be targeted, and, if compromised, the entire network is put at risk. The device named "main_ids" is probably the main Intrusion Detection System (IDS), which, if compromised or disabled, will put your entire network at risk.

Solutions

The solution is to configure this DNS server to allow zone transfers only with specific trusted hosts, not just anyone who asks. This will help minimize information leakage, though it is still advised to name systems in ways that do not identify their geographic location, their function(s), or other characteristics that could attack unwanted attention or leak exploitable information.

2.2 Scanning: - [Top of page]

Scanning is interrogating your systems for available services, resource sharing, software version information, user account information, and other conditions which might be exploitable. Here we begin to gather the general characteristics of the study subject, while the detailed characteristics are gathered in the next section, Enumeration.

When dealing with network security, TCP/IP networking specifically, it is necessary to have a clear understanding of the concepts of *Ports* and *Services*.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Ports and Services are fundamental to IP networking, and effective security practices take them into careful consideration. For a detailed explanation of **ports**, please refer to Arthur Hunt's SANS paper, An Explanation of Ports (Hunt).

Think of a **port** as a "door" into a computer. On the outside of the port is the network that the computer is connected to. On the inside of the port there is a program listening for a knock at that door. When someone from the network knocks on a specific port/door, the program listening at that port/door will respond by providing its specific service.

For each computer that uses the TCP/IP protocol, there are 65,535 possible Transmission Control Protocol (TCP) ports, along with another 65,535 User Datagram Protocol (UDP) ports. The main difference between the two protocols is that TCP provides error checking and correcting features which help insure that packets are successfully delivered, or, in the event of a failure, the sender and/or recipient are notified and corrective actions can be put into play. UDP is connection-less and makes no attempt to guarantee the successful delivery of packets.

The first 1,023 ports are considered **well known**, meaning that there is general agreement on which services will be provided through which ports. For example, web servers provide their service through TCP 80, DNS servers through UDP and TCP 53, and Email post offices provide their Simple Mail Transfer Protocol (SMTP) services through TCP 25.

If a port has a program actively listening, then that port is considered **open**, otherwise it is considered **closed**.

For reference, a very comprehensive list of ports and the services expected to be provided through those ports is available at www.iana.org/assignments/port-numbers.

One very useful resource is a web page authored by Robert Graham. In Mr. Graham's words;

> This document explains what you see in firewall logs, especially what port numbers means. You can use this information to help figure out what hackers are up to. (Graham)

Trojan programs are **malware** (malicious software) that infect a system and then perform tasks or provide services without the owner's knowledge, and which are malicious or misleading in nature. Many Trojan programs will begin listening at a port for an intruder to request access to the system, which the Trojan program then grants.

Another helpful resource is http://www.simovits.com/nyheter9 902.html. This page provides useful details about specific Trojan programs. If you find an open port that you cannot account for, this resource can provide useful information in determining if you have been infected. It is important to realize that any program (malicious or otherwise) can listen at any port number. Too often, it is simply assumed that if a particular port is open, the service it provides can be implied by that port number. An intruder can place a malicious program at a commonly used port in order to hide or obscure its presence.

The issues of ports and services apply to both workstations and servers, not to mention printers, switches, and most other devices that use the TCP/IP protocols.

Port scanning is simply the process of knocking at some, or all, of the TCP and UDP ports to determine which are open and what services are being provided through those open ports.

Port scanning can reveal several things about a remote host. For example:

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

- Whether a host is online and reachable. If a system is offline, a port scan will report that the system could not be reached.
- If a firewall exists between the technicians and the target host, port scanning will report which ports, if any, are being filtered by a firewall or similar device.
- Many services will respond to a port scan's "knock at the door" by replying with their "banner", which usually contains the name, version, manufacturer, and other information about the service. This is also referred to as "banner grabbing".
- Obviously, port scanning can help identify the types of services the study subject offers. For example, if TCP 80 is open, it is likely a web server, and if port TCP 1433 is open, it is likely a Microsoft SQL database server.

While a properly configured firewall will help protect hosts from being port scanned, they can be circumvented. Rather than trying to protect a weakness, reduce your risk by repairing the weakness, then provide protection in general. For example, rather than using a firewall to protect an open anonymous ftp service which is not needed, remove the ftp service (while still filtering it at your firewall - remember that defense in many layers is much more effective than with just one layer). It is impossible to protect against every technique that an intruder might dream up.

Tool: Fscan is a port scanning tool that is highly configurable and very fast.

The fscan tool is illustrated below:

If you run fscan with no parameters at all, it will display the syntax help screen describing these and other optional parameters.

```
C:\>Fscan
FScan v1.12 - Command line port scanner.
Copyright 2000 (c) by Foundstone, Inc.
http://www.foundstone.com

FScan [-abefhqnv?] [-cditz ] [-flo ] [-pu [,-]] IP[,IP-IP]

 -?/-h - shows this help text
 -a    - append to output file (used in conjunction with -o option)
 -b    - get port banners
 -c    - timeout for connection attempts (ms)
 -d    - delay between scans (ms)
 -e    - resolve IP addresses to hostnames
 -f    - read IPs from file (compatible with output from -o)
 -i    - bind to given local port
 -l    - port list file - enclose name in quotes if it contains spaces
 -n    - no port scanning - only pinging (unless you use -q)
 -o    - output file - enclose name in quotes if it contains spaces
 -p    - TCP port(s) to scan (a comma separated list of ports/ranges)
 -q    - quiet mode, do not ping host before scan
 -r    - randomize port order
 -t    - timeout for pings (ms)
 -u    - UDP port(s) to scan (a comma separated list of ports/ranges)
 -v    - verbose mode
 -z    - maximum simultaneous threads to use for scanning

Example: fscan -bp 80,100-200,443 10.0.0.1-10.0.1.200
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
This example would scan ports 80, 100, 101 ... through 200 and 443
on all IP addresses between 10.0.0.1 and 10.0.1.200 inclusive
and grab the banners from those ports on those hosts.
```

The following example will perform a port scan of the host at IP address 127.0.0.1, specifically ports 1 through 140. The results will be written to a file named Results.txt. The other parameters (-abqv) instruct fscan to (a) append the results to the output file, (b) grab all available banners from open ports, (q) run quietly - without pinging the target, and (v) to report more (verbose) detail than usual.

```
C:\>fscan -abqve -o Results.txt -p 1-1024 -z 254 127.0.0.3-10
FScan v1.12 - Command line port scanner.
Copyright 2000 (c) by Foundstone, Inc.
http://www.foundstone.com

 Scan started at Wed Nov 28 13:38:22 2001

127.0.0.3      25/tcp  MyDomain.com
   220 TargetHost.Network.com ESMTP Server (Microsoft Exchange Internet Ma
   il Service 5.5.2650.21) ready[0D][0A]
127.0.0.3     139/tcp  MyDomain.com
   [83][00][00][01][8F]
127.0.0.3     119/tcp  MyDomain.com
   200 Microsoft Exchange Internet News Service Version 5.5.2650.23
(posting
   allowed)[0D][0A]
127.0.0.3     110/tcp  MyDomain.com
   +OK Microsoft Exchange POP3 server version 5.5.2650.23 ready[0D][0A]
127.0.0.3     143/tcp  MyDomain.com
   * OK Microsoft Exchange IMAP4rev1 server version 5.5.2650.23
(MyDomain.co
   m) ready[0D][0A]
127.0.0.3      27/tcp  MyDomain.com
127.0.0.3     135/tcp  MyDomain.com
127.0.0.3     593/tcp  MyDomain.com
   ncacn_http/1.0
127.0.0.3     389/tcp  MyDomain.com
127.0.0.3     563/tcp  MyDomain.com
127.0.0.3     636/tcp  MyDomain.com
127.0.0.3     993/tcp  MyDomain.com
127.0.0.3     995/tcp  MyDomain.com
127.0.0.4     139/tcp  Host2260
   [83][00][00][01][8F]
127.0.0.4     135/tcp  Host2260
127.0.0.4     475/tcp  Host2260

[F2][FA][F2][F2][F2][F2][F2][F2][F2][F1]8[B1][9A][B9]8[B1][9A][B9]8[B1][9A]

[B9]8[B1][9A][B9]8[B1][9A][B9]8[B1][9A][B9]8[B1][9A][B9]8[B1][9A][B9]8[B1]
   [9A][B9]8[B1][9B][91]Sl-[CD][CD]X3[C9][F5]6[8F][A2]
127.0.0.4     515/tcp  Host2260
   [01]
127.0.0.5     139/tcp  Host2262
   [83][00][00][01][8F]
127.0.0.5     135/tcp  Host2262
127.0.0.5     515/tcp  Host2262
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
   [01]
127.0.0.6     139/tcp  Host2263
   [83][00][00][01][8F]
127.0.0.9      80/tcp  www.MyDomain.com
   HTTP/1.1 400 Bad Request[0D][0A]Server: Microsoft-IIS/4.0[0D][0A]Date:
Wed
   , 28 Nov 2001 18:37:50 GMT[0D][0A]Content-Type:
text/html[0D][0A]Content-L
   ength: 87[0D][0A][0D][0A]
The
  parameter is incorrect.

127.0.0.9     139/tcp  www.MyDomain.com
  [83][00][00][01][8F]
127.0.0.9     135/tcp  www.MyDomain.com
127.0.0.9     443/tcp  www.MyDomain.com
127.0.0.10     80/tcp  [Unknown hostname]
  HTTP/1.1 400 Bad Request[0D][0A]Server: Microsoft-IIS/4.0[0D][0A]Date: Wed
  , 28 Nov 2001 18:39:10 GMT[0D][0A]Content-Type: text/html[0D][0A]Content-L
  ength: 87[0D][0A][0D][0A]
The
  parameter is incorrect.

127.0.0.10    139/tcp  [Unknown hostname]
  [83][00][00][01][8F]
127.0.0.10    135/tcp  [Unknown hostname]
127.0.0.10    443/tcp  [Unknown hostname]

 Scan finished at Wed Nov 28 13:39:46 2001
 Time taken: 8192 ports in 83.840 secs (97.71 ports/sec)

C:\>
```

Finding:

Attackers will note the banners that service respond with, which usually include the software name and its version number, and use that information to check any number of publicly available vulnerability databases for known vulnerabilities in that specific program, along with instructions for performing an attack. Notice the banners returned from ports 25 and 80.

Solution:

Make sure that only necessary ports are open. An unnecessary open port is an invitation and opportunity for intrusion.

Tool: FPort identifies programs listening at ports

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Even when a suspicious port is identified. Further investigation must be performed to verify what service is actually being provided through that port and that it is not a security problem. Of course, if this service is not required, it should be removed after it has been verified.

The FPort program will report which ports, both TCP and UDP, are open and the path to each program listening at each port. The FPort program must be run at the local console and with administrative privileges.

The FPort program is illustrated below:

Here we see an example of launching fport with no parameters so we can view the syntax help screen.

```
D:\>fport /?
FPort v1.33 - TCP/IP Process to Port Mapper
Copyright 2000 by Foundstone, Inc.
http://www.foundstone.com
Usage:
        /p       sort by port
        /a       sort by application
        /i       sort by pid
        /ap      sort by application path

Special thanks to Gary Nebbett for light
```

Below fport is run to determine which ports are open and the executable file providing its service through that port.

```
D:\>fport
FPort v1.33 - TCP/IP Process to Port Mapper
Copyright 2000 by Foundstone, Inc.
http://www.foundstone.com

Pid   Process            Port  Proto Path
776   sshd           ->  22    TCP   C:\WINNT\sshd.exe
396   svchost        ->  135   TCP   C:\WINNT\system32\svchost.exe
8     System         ->  139   TCP
8     System         ->  445   TCP
592   MSTask         ->  1027  TCP   C:\WINNT\system32\MSTask.exe
8     System         ->  1029  TCP
8     System         ->  1035  TCP
8     System         ->  1253  TCP
8     System         ->  1255  TCP
8     System         ->  1257  TCP

396   svchost        ->  135   UDP   C:\WINNT\system32\svchost.exe
8     System         ->  137   UDP
8     System         ->  138   UDP
8     System         ->  445   UDP
228   lsass          ->  500   UDP   C:\WINNT\system32\lsass.exe
208   services       ->  1028  UDP   C:\WINNT\system32\services.exe
1684  iexplore       ->  1258  UDP   C:\Internet Explorer\iexplore.exe
```

Findings:

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Secure SHell, and the d indicates it is a Daemon. Daemons are to *nix what services are to Windows. "ssh" can be thought of as providing encrypted telnet services across a network. Telnet is unencrypted, so a sniffer could capture an entire session in clear-text, while "ssh" encrypts the session data so that a sniffer would only see meaningless (encrypted) material.

Solutions:

Run these reports frequently and compare the findings in order to verify that no new services have been opened without authorization. This creates a second layer of security; not only do you port scan your systems to verify that no new ports have been open, but when a new port is noticed, comparing Fport reports (current to baseline) allow you to verify the legitimacy of the new service.

2.3. Enumeration: - [Top of page]

In enumeration we directly interrogate our systems searching for the detailed data and information that an intruder would use to launch a more effective attack.

As we begin to scan our systems directly, it is important to understand the difference between "local" and "domain" accounts in the Microsoft world. Both the local and domain environments must be secured. Securing only one is like "putting a steel door on a grass hut."

Windows machines have two default user accounts, Administrator and Guest. These accounts are local to that machine and are not logically related to the domain account a person would employ from that machine.

Once the machine has joined a domain, it uses the domain accounts to authenticate the person logged into that machine so that they can make use of the networked resources that are managed by that domain. Think of a domain as a collection of user accounts and the privileges each account can exercise in that domain and with its resources (disks, printers, files, etc.). Just as there is a administrator and guest account on each machine, there is a domain administrator and domain guest account for each domain.

A seldom-noticed issue is that a person sitting at one machine can log into a remote machine as that remote machine's local administrator or local guest account and access its resources accordingly. Joining a domain does not suspend or disable the local accounts. This is true of both servers and workstations.

The risk is that if a local administrator/guest account has a trivial password an intruder can, from a remote host, *connect to* that system using its local administrator account and exercise administrative privileges on that machine and its resources (disks, files, printers, programs, etc.). To make matters worse, since these systems are multi-user systems, that intruder can login as the local administrator **at the same time** the authorized user is logged into a domain.

Findings:

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Tool: enum - enum queries a target about its domain affiliation, trust relationships, and other NetBIOS configurations.

The enum program is illustrated below:

Running enum with no parameters will produce its syntax help screen.

```
D:\>enum
usage:  enum  [switches]  [hostname|ip]
  -U:  get userlist
  -M:  get machine list
  -N:  get namelist dump (different from -U|-M)
  -S:  get sharelist
  -P:  get password policy information
  -G:  get group and member list
  -L:  get LSA policy information
  -D:  dictionary crack, needs -u and -f
  -d:  be detailed, applies to -U and -S
  -c:  don't cancel sessions
  -u:  specify username to use (default "")
  -p:  specify password to use (default "")
  -f:  specify dictfile to use (wants -D)
```

Below we run a basic query of the target host.

```
D:\>enum -L 127.0.0.3
server: 127.0.0.3
setting up session... success.
opening lsa policy... success.
server role: 3 [primary (unknown)]
names:
  netbios: AuditReview01
  domain: AuditReview
quota:
  paged pool limit: 33554432
  non paged pool limit: 1048576
  min work set size: 65536
  max work set size: 251658240
  pagefile limit: 0
  time limit: 0
trusted domains:
  indeterminate
PDC: AuditPDC1
netlogon done by a PDC server
cleaning up... success.
```

Finding:

In the example above, the AuditReview domain has been identified as the domain in which this machine (AuditReview01) is a member. Note that the host AuditPDC1 is the PDC (Primary Domain Controller) for the AuditReview domain. Now you know which machine holds the current list of user accounts and their passwords for this entire domain!

```
D:\>enum -U -d 127.0.0.3
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
server: 127.0.0.3
setting up session... success.
getting user list (pass 1, index 0)... success, got 8.
  Guest (Built-in account for guest access to the computer/domain)
  attributes: disabled locked_out
  User1   attributes:
  User2   attributes:
  adminexch (For Exchange - Do NOT delete)
  attributes:
  adminSQL (For SQL services -- Do NOT delete )
  attributes:
  User3   attributes:
  User4   attributes:
cleaning up... success.
```

Finding:

In the example above, a list of all the user accounts in this domain has been provided. Hackers like to say, "If I have a user account, I am half way in!" Access usually only requires two things: the account name and the password. We discover that the guest account is disabled. Be sure to also look for information leakage in the comments fields.

```
D:\>enum -P 127.0.0.3
server: 127.0.0.3
setting up session... success.
password policy:
  min length: 7 chars
  min age: none
  max age: 30 days
  lockout threshold: 3 attempts
  lockout duration: 71582788 mins
  lockout reset: 30 mins
cleaning up... success.
```

Finding:

We might find, worst case, that there are no requirements for passwords or their characteristics. With this information, we can now tailor a password attack that will not lockout accounts or waste time testing for passwords that would not meet this criteria.

```
D:\>enum -G -d 127.0.0.3
server: 127.0.0.3
setting up session... success.
Group: Account technicians
Group: Administrators
AuditReview\Domain Admins
AuditReview\adminexch
AuditReview\NPATSON
AuditReview\BACKUP
AuditReview\adminSQL
Group: Backup technicians
AuditReview\Repluser
Group: Guests
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
AuditReview\Domain Guests
AuditReview\IUSR_WebSite
Group: Print technicians
Group: Replicator
AuditReview\Repluser
Group: Server technicians
Group: Users
AuditReview\Domain Users
Group: Address (C)
AuditReview\User6
Group: AuditReview Legal Staff (C)
AuditReview\User9
Group: Payroll (C)
AuditReview\User9
Group: Audit Guide Updates (C)
```

Finding:

The detailed group information shown above shows an attacker which user accounts belong to which groups. Now an attacker will know which accounts have administrative privileges. This also tells the attacker what type of resources exist (*legal staff* and *Payroll* for example).

In the next example, we enumerate the shares provided by the target host.

You will see the term "share" used as a noun. In the Microsoft world, shares are the mechanism for providing portions of a host's filing system (files, programs, disk space, etc.) for use by other authorized hosts on a network.

So, it becomes very important to identify all of the **shares** being made available by our systems, especially since shares can be provided by both servers or workstations. Worse, domain users often have adequate permissions to create shares of their own local file systems without the approval or knowledge of the network administrators.

Just like ports, file shares are doors into a host and can be exploited to the point of completely taking over a system.

```
D:\>enum -S 127.0.0.3
server: 127.0.0.3
setting up session... success.
enumerating shares (pass 1)... got 16 shares, 0 left:
  NETLOGON  D  ADMIN$  REPL$  IPC$  C$  Samples  E$  F$  connect$
Address
  Add-ins  tracking.log  PWRCHUTE  Passwords Resources  Temp
SecretStuff$
cleaning up... success.
```

Finding:

In the host example above, there are 4 shares. These happen to be the default hidden administrative shares which are hidden from some tools because of the "$" at the end of their names. This is NOT a security feature and should NEVER be used as such.

The IPC$ share is the Inter-Process Communications share, and is used for many

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Windows (NetBIOS or SMB for example) networking functions.

- The ADMIN$ share points to the %systemroot% folder, which is usually C:\winnt\, the root of the system files.
- For each logical fixed disk, there is also an hidden administrative share. So, if there are C:, D:, and E: drives on the system, there will also be C$, D$, and E$ shares.

In the above example, note that **SecretStuff$** is a hidden share, though not hidden to net-fizz or other non-Microsoft programs. The **Passwords** share will also attract attention.

The names of shares should not be enticing - encouraging the intruder to look further. Remember, part of the risk created by enticements is that by encouraging an intruder to poke around our systems, they might discover a vulnerability that would not have been discovered through the general surface scanning most intruders perform.

Basic scanning like port probing, share scanning, and trace routing are intended to point-out items the intruder would consider worth investigating. If you do not provide anything of interest, you may avoid a detailed probe that could uncover vulnerabilities that would otherwise have been unknown.

Even so, it is critical to understand that you cannot rely on "*security by anonymity*" to any degree. It is nice when it happens, every little bit of protection helps. However, sanitizing a system of enticements and information leakage does not minimize the need to enthusiastically harden your systems or following the **security Best Security Practices** appropriate for each of your systems.

Solutions:

Where practical, Windows NT hosts should have the following registry set.

HKLM/System/CurrentControlSet/Control/LSA/RestrictAnonymous=1

See the "W5 - Information Leakage via null session connections" section of the The SANS Institutes, "The Twenty Most Critical Internet Security Vulnerabilities" for more information (SANS).

As with open ports, only the shares actually needed should exist. Each share is a potential opening for an intruder to gain unauthorized access. Only necessary shares should exist, and all others should be removed.

3. Best Security Practices - [Top of page]

Best Security Practices are the things that work, not simply the things we were taught.

Fully and completely securing a system is the goal some set. There is, however, general agreement that this is impractical and usually an impossible goal. A much more realistic goal is to set up a regular routine where you identify and correct as many vulnerabilities as practical.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

> The goal of a company in protecting its computers and networks is to make it so difficult for an attacker to gain access that he gives up before he gets in. Today, because so many sites have minimal or no security, attackers usually gain access relatively quickly and with a low level of expertise. Therefore, if a company's site has some security, the chances of an attacker exploiting its systems are decreased significantly, because if he meets some resistance, he will probably move on to a more vulnerable site. This is only true for an opportunistic attacker who scans the Internet looking for any easy target. (Cole, 27)

The idea is not that you should protect a system to the point it cannot be compromised, but to secure it at least enough that most intruders will not be able to break in, and will choose to direct their efforts elsewhere. It is just like putting bars and locks on our windows and doors. We do it not to "keep the robbers out", but to persuade them to turn their attention to our neighbors.

There is another beneficial Return on Investment (ROI) aspect to implementing effective Best Security Practices. Rather than directing our efforts at protecting against the thousands of specific threats (this exploit, that Trojan virus, these mis-configurations), we focus our energies into those tasks that provide the most comprehensive protection against the majority of threats.

> These few software vulnerabilities account for the majority of successful attacks, simply because attackers are opportunistic - taking the easiest and most convenient route. They exploit the best-known flaws with the most effective and widely available attack tools. They count on organizations not fixing the problems and they often attack indiscriminately, scanning the Internet for any vulnerable systems. (SANS)

So, as we face the "swarm of threats" buzzing around us, rather than swat at each one at a time, we pull out our can of "bug-insecticide" and start spraying so as to minimize our effort while also significantly reducing our risk.

Best Security Practices are very dynamic, constantly changing and evolving. Administrators should include their own Best Security Practices and modify those listed below to best fit their environment. It is vital that you develop a collection of Best Security Practices which take into consideration your needs risks, resources, and then apply to your systems so as to most effectively protect them from intrusion or disruption.

Notice that there is no order of importance or sequence in this list of Best Security Practices. While it can be argued that one may provide more over all protection than another, the difficult truth at the heart of protecting information technology systems is that the battle has to be fought on several fronts, with different tactics, with different weapons, and on different schedules. Information systems are unavoidably complex and fluid, so the most effective way to apply security is in layers. This means you should place security measures at different points in your network, allowing each to do what it does best. From an attacker's perspective, you have constructed a series of obstacles of varying difficulty between the attacker and your systems. You also secure each component in your system (firewalls, routers, servers, hosts, and appliances) so that even if an attacker works their way through your obstacle-course, at the end they will find systems that are resistant to attack.

Backups

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Maintain full and reliable backups of all data, log files, and anything else that is necessary or useful for normal operations. Archive all software (purchased or freeware), upgrades, and patches off-line so that it can be reloaded when necessary. Also backup configurations, such as the Windows registry and text/binary configuration files, used by the operating systems or applications.

The topic of effective backup strategies is well considered and documented. The "devil is in the details" of designing a backup strategy that provides adequate protection. You will need to consider the media, retention requirements, storage, rotation, methods (incremental, differential, etc.) and the scheduling.

One useful example of a reference is Practical Unix & Internet Security, which is viewable at http://www.ebone.at/books/programmers/sonstiges/oreillybookself/tcpip/puis/ch07_02.htm

---

Anti-Virus Systems

Install anti-virus protection systems at key points and, of critical importance, keep them current! Key points would include servers (focusing on files) and post offices (focusing on in and outbound email and attachments).

While computer viruses are publicly stereotyped as disfocused attempts to damage and disrupt systems, of greater concern is the type that quietly, skillfully, and effectively alters the victim system, allowing an intruder privileged and covert access. These types of viruses can expose your network, including systems that have not be infected, to covert attack.

Remove All Unnecessary Accounts

Simply disabling an account is not sufficient to guard against an intruder abusing it. A favorite tactic is to enter a system through one account, re-enable a disabled account (preferably with greater privilege, or the potential of escalating its privilege), and continuing the intrusion using that account. This helps to hide the intruders activity behind an account that "blends" in with the crowd. It is easy for an administrator to assume that a disabled account has been re-enabled (if noticed at all) by another administrator, or that "*it's just one of those things.*"

Particularly dangerous are the privileged accounts (administrators, power users, executive staff) which were disabled until someone takes that person's place or because it is setup "just so," or there are automated processes that rely on it. These "placeholder" accounts are very inviting to an intruder, and their abuse is not particularly noticeable.

Rename Default Administrative Accounts

It is trivial to identify the actual Microsoft Administrator account, but then why make it easy for them? Renaming the default Administrator accounts may not slow down a moderately skilled attacker, but it will defeat most of the automated tools and techniques used by less skilled attackers, who make the assumption your system is using default account names.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

This should include not only domain accounts, but also the local Windows Administrator account.

Name Servers and Workstations Securely

Database servers, for example, have typically been named db1, db2, and so on. The host name alone can advertise to a potential attacker a host's primary service or purpose and how important *you* consider the host to be.

A server named "test" tells a potential intruder have found something:

- You consider important enough to spend time testing,
- Something that is not likely used by the same person day after day and who would notice minor changes in its behavior,
- A box that is not as likely to be secured, protected, or monitored as would be production host, or
- A box that administrators undoubtedly log into, which might be a natural stepping stone in to your core network.

The same applies for names like Lab01, Temp, and Dev. This points out boxes that are not only "exceptions," but used in ways that tend to weaken security.

Do not name boxes for the people who primarily use them. This simply provides a "directory" of executives, administrators, and other users likely to have privileged rights on the network.

Publicizing the computers that executives work from is particularly unwise. From an intruders perspective, executives are people who demand excessive privilege, user-friendliness and convenience over security, and who usually are the least likely to practice effective security habits.

Enable and Monitor Logging and Auditing

Focus on logging only those events that either alert you to problems, or, in some way, help you manage that system. Too much logging will generate useless data which will only obscure the important information buried within.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

> One of the maxims of security is, "Prevention is ideal, but detection is a must." As long as you allow traffic to flow between your network and the Internet, the opportunity for an attacker to sneak in and penetrate the network, is there. New vulnerabilities are discovered every week, and there are very few ways to defend yourself against an attacker using a new vulnerability. Once you are attacked, without logs, you have little chance of discovering what the attackers did. Without that knowledge, your organization must choose between completely reloading the operating system from original media, and then hoping the data back-ups were OK, or take the risk that you are running a system that a hacker still controls.
>
> You can not detect an attack if you do not know what is occurring on your network. Logs provide the details of what is occurring, what systems are being attacked, and what systems have been compromised. (SANS)

Capturing event information is useful only if the logs are reviewed in a regular and timely manner. And since it is impractical to wade through screen after screen looking for indications of a problem, the monitoring process should be automated, to produce alerts as near to real-time as practical and to produce meaningful exception reports.

Regularly Scan Systems

This will help you determine if your systems have been compromised, or that a well-meaning client has unknowingly exposed the network to intrusion. This is where you take the pulse of your network.

> Both legitimate users and attackers connect to systems via open ports. The more ports that are open, the more possible ways that someone can connect to your system. Therefore, it is important to keep the least number of ports open on a system necessary for it to function properly. All other ports must be closed. (SANS)

Examples of types of scans would include;

- Scan for NetBIOS shares using a tool like enum

> The Server Message Block (SMB) protocol, also known as the Common Internet Filing System (CIFS), enables file sharing over networks. Improper configuration can expose critical systems files or give full file system access to any hostile party connected to the Internet.
>
> Enabling file sharing on Windows machines makes them vulnerable to bother information theft and certain types of quick-moving viruses. (SANS)

- You want to verify that only the file shares you believe to exist, actually do. This includes both client and servers machines.
- Run password auditing tools to help determine the effective strength of client account passwords and to insure that password policies are being followed and are effective.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

- Perform full port scans using a tool like fscan
  Port scans should cover all ports (1-65,535), both UDP and TCP, on all systems. This includes;
  - both clients and servers
  - devices such as routers, switches, printers
  - and anything else connected (physically through wire or virtually through wireless technology) to your network.

These scans will help determine that only the ports you believe to be open are indeed the only open ports.

Port scans alert when new ports are opened without your knowledge or adequate approval.

Unfortunately, it is not uncommon for one patch to undo a correction a previous patch applied. So, if you should assume that every time a system is "altered," previously patched vulnerabilities could be reopened or new ones created.

By fully scanning a system after each alteration, you give yourself the opportunity to identify inadvertent vulnerabilities before an attacker notices and exploits them.

Keep Current on Software Updates

This includes operating systems and client or servers applications. Check with each of your vendor to see if they have email-based alert services that regularly advise of newly released patches and updates. All such relevant updates should be applied as soon as reasonable.

It is true that yesterday's exploit can be tomorrow's disaster, if you apply today's patch next week. 8-) However, this should be balanced with reasonable testing and preparation. Too many "patches" cause more disruption than the actual vulnerability might. So, design a plan for reaching an adequate level of comfort with each new patch/update that includes testing and research as quickly as practical.

One helpful practice is to closely monitor the discussion boards that deal with software patches to see if others are having problems or have suggestions for successful implementation.

Examples of informative and useful security newsletters, alerting services, and discussion forums include:

- SANS Newsletter Subscription Service. Free subscription sign-up available at: http://server2.sans.org/sansnews
- bugtraq - Free subscription sign-up available at: www.securityfocus.com/cgi-bin/forums.pl
- Microsoft's Product Security Notification - Free subscription sign-up available at: www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/notify.asp

---

Password Policies

While there are promising technologies on the horizon which could replace passwords as a method of authenticating clients, at present we are reliant on passwords.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

A password policy should define the required characteristics of accepted passwords for each system. These characteristics should include;

- minimum length
- composition; alpha, upper or lower case, numeric, special
- effective life
- uniqueness (how often a password can be reused)
- lockout properties; under what conditions, and for how long.

These characteristics differ from system to system because each has different capabilities.

5. Reference - [Top of page]

1. **Hunt, Arthur**. "An Explanation of Ports". February 3, 2001. URL: http://www.sans.org/infosecFAQ/securitybasics/ports.htm (7, Dec. 2001).
2. **Cole, Eric**. Hackers Beware - Defending your network from the wiley hacker. Indianapolis: New Riders Publishing, 2001.
3. **Schneier, Bruce**. Secrets & Lies - Digital Security in a Networked World. New York: John Wiley & Sons, Inc. 2000.
4. **The SANS Institute**. "The Twenty Most Critical Internet Security Vulnerabilities (Updated) - The Experts' Consensus". Version 2.501 November 15, 2001. URL: http://www.sans.org/top20.htm (7 Dec. 2001).
5. **Herzog, Pete**. "THE OPEN SOURCE SECURITY TESTING METHODOLOGY MANUAL v. 1.5". Saturday, May 5, 2001. URL: http://uk.osstmm.org/osstmm.htm (7 Dec. 2001).
6. **Kurtz, George and Prosise, Chris**. "SECURE STRATEGIES - PENETRATION TESTING EXPOSED". September 2000. URL: http://www.infosecuritymag.com/articles/september00/features3.shtml (7 Dec. 2001).
7. **Graham, Robert**. "FAQ: Firewall Forensics (What am I seeing?)". Version 0.4.1, June 20, 2000. http://www.robertgraham.com/pubs/firewall-seen.html (7 Dec. 2001).
8. **Federal Best Security Practices (BSPs)**. URL: http://www.bsp.gsa.gov/ (7 Dec. 2001).
9. **Mudge** "L0phtcrack 1.5 Lanman/NT password hash cracker" URL: www.insecure.org/sploits/l0phtcrack.lanman.problems.html (7 Dec. 2001).
10. **IANA**, the Internet Assigned Numbers Authority. "Port Numbers". URL: www.iana.org/assignments/port-numbers (7 Dec. 2001).
11. **Simovits Consulting**. "Ports used by trojans (2001-06-30)". URL: http://www.simovits.com/nyheter9902.html (7 Dec. 2001).
12. **O'Reilly**. "Practical Unix & Internet Security". URL: www.ebone.at/books/programmers/sonstiges/oreillybookself/tcpip/puis/ch07_02.htm (7 Dec. 2001).
13. **NIST** - National Institute of Standards and Technology. "Virus Information" URL: http://csrc.nist.gov/virus/ (27 Nov. 2001).
14. **whois.exe** - http://www.kiraly.com/software/utilities/whois/ (7 Dec. 2001).
15. **FScan.exe** from www.Foundstone.com (28 Nov. 2001).
16. **FPort.exe** from www.Foundstone.com (7 Dec. 2001).
17. **nslookup.exe** should be included with MS Windows.
18. **net.exe** should be included with MS Windows.
19. **nmapNT.exe** - is available at http://www.eeye.com/html/Research/Tools/nmapNT.html and is the Windows version of nmap. Note that SP1 for nmapnt is located at http://www.eeye.com/html/Databases/software/nmapNT/nmapntsp1.zip (7 Dec. 2001).
20. **enum.exe** - a very powerful tool for enumerating NetBIOS information on a specific host. Available at razor.bindview.com/tools/desc/enum_readme.html (7 Dec. 2001).

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

21. **VeriSign Whois** - http://www.netsol.com/cgi-bin/whois/whois - The networksolutions DNS database is a prime source for information. This service has recently been acquired by VeriSign. (7 Dec. 2001).
22. **ARIN** - http://whois.arin.net/whois/index.html [As ARIN states on its About ARIN page, "ARIN is a non-profit organization established for the purpose of administration and registration of Internet Protocol (IP) numbers for the following geographical areas: North America, South America, the Caribbean and sub-Sahara Africa." (7 Dec. 2001).
23. **InterNIC** - http://www.internic.net/whois.html. As InterNIC's home page states, "This website has been established to provide the public information regarding Internet domain name registration services and will be updated frequently." (7 Dec. 2001).
24. **SANS Newsletter Subscription Service**. URL: http://server2.sans.org/sansnews (7 Dec. 2001).
25. **bugtraq** URL: www.securityfocus.com/cgi-bin/forums.pl (30 Nov. 2001).
26. **Microsoft's Product Security Notification.** URL: www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/notify.asp (7 Dec. 2001).

End of Paper - Vulnerability Identification and Remediation Through Best Security Practices

© SANS Institute 2002, Author retains full rights.

Last Updated: September 3rd, 2015

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Crystal City 2015	Crystal City, VAUS	Sep 08, 2015 - Sep 13, 2015	Live Event
SANS Network Security 2015	Las Vegas, NVUS	Sep 12, 2015 - Sep 21, 2015	Live Event
SANS Seoul 2015	Seoul, KR	Sep 14, 2015 - Sep 19, 2015	Live Event
Data Breach Investigation Summit & Training	Dallas, TXUS	Sep 21, 2015 - Sep 26, 2015	Live Event
SANS Baltimore 2015	Baltimore, MDUS	Sep 21, 2015 - Sep 26, 2015	Live Event
SANS Perth 2015	Perth, AU	Sep 21, 2015 - Sep 26, 2015	Live Event
SANS Tallinn 2015	Tallinn, EE	Sep 21, 2015 - Sep 26, 2015	Live Event
SANS ICS Amsterdam 2015	Amsterdam, NL	Sep 22, 2015 - Sep 28, 2015	Live Event
SANS Bangalore 2015	Bangalore, IN	Sep 28, 2015 - Oct 17, 2015	Live Event
SANS DFIR Prague 2015	Prague, CZ	Oct 05, 2015 - Oct 17, 2015	Live Event
SANS Seattle 2015	Seattle, WAUS	Oct 05, 2015 - Oct 10, 2015	Live Event
SANS Tysons Corner 2015	Tysons Corner, VAUS	Oct 12, 2015 - Oct 17, 2015	Live Event
SOS: SANS October Singapore 2015	Singapore, SG	Oct 12, 2015 - Oct 24, 2015	Live Event
SANS Gulf Region 2015	Dubai, AE	Oct 17, 2015 - Oct 29, 2015	Live Event
SANS Cyber Defense San Diego 2015	San Diego, CAUS	Oct 19, 2015 - Oct 24, 2015	Live Event
SANS Tokyo Autumn 2015	Tokyo, JP	Oct 19, 2015 - Oct 31, 2015	Live Event
SANS South Florida 2015	Fort Lauderdale, FLUS	Nov 09, 2015 - Nov 14, 2015	Live Event
SANS Sydney 2015	Sydney, AU	Nov 09, 2015 - Nov 21, 2015	Live Event
SANS London 2015	London, GB	Nov 14, 2015 - Nov 23, 2015	Live Event
Pen Test Hackfest Summit & Training	Alexandria, VAUS	Nov 16, 2015 - Nov 23, 2015	Live Event
SANS Hyderabad 2015	Hyderabad, IN	Nov 24, 2015 - Dec 04, 2015	Live Event
SANS Cape Town 2015	Cape Town, ZA	Nov 30, 2015 - Dec 05, 2015	Live Event
SANS San Francisco 2015	San Francisco, CAUS	Nov 30, 2015 - Dec 05, 2015	Live Event
SANS Milan 2015	OnlineIT	Sep 07, 2015 - Sep 12, 2015	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced