

# Penetration Test Manual

---

**INSERT DATE**

**David Buyer**  
be281@bfn.org  
Version 3.0

**Prepared for:**

**INSERT CLIENT LOGO**

Test performed by: **INSERT NAME**  
**BUISNESS NAME**  
**ADDRESS**  
**ADDRESS**

Ph **INSERT PHONE**  
Fax **INSERT FAX**

**INSERT EMAIL ADDRESS**

# Table of Contents

Table of Contents .....	2
Introduction .....	3
Section 1. Network Footprinting .....	4
Section 2. Port Scanning .....	8
Section 3. System Fingerprinting .....	11
Section 4. Application probing .....	12
Section 5. Automated Vulnerability Scanning .....	14
Section 6. Exploit Research .....	16
Section 7. Manual System/Application Vulnerability Testing and Verification .	17
Section 8. Firewall & Access Control List Testing .....	19
Section 9. Intrusion Detection System (IDS) Testing .....	22
Section 10. Password Cracking .....	24
Section 11. Denial of Service Testing .....	26
Section 12. Wireless Leak Tests .....	27
Section 13. PBX and Dial-up .....	29
Section 14. Internal Auditing .....	31
Appendix A .....	33
Appendix B .....	34

# Introduction

---

This document was created to aid in the delivery of Network Penetration Tests and Vulnerability Assessments. Use this document for a specific area to be tested (maybe only a firewall audit or internal audit only) or as a full-blown penetration test.

Before engaging in a penetration test, save a copy of this document with the name CLIENT\_NAME\_MONTH\_YEAR\_PENETRATION\_TEST.doc. Use this document to record ALL information gathered during the test.

Commands are typed in Courier font like `this`. Hints and tips are italicized like *this*. Insertion points that should be deleted before pasting information gathered are highlighted like **THIS**. Commands that have parameters that should be changed to correspond with the actual parameters are highlighted like **THIS**. These commands should be changed prior to actually running them.

All the tools listed in this manual are intended to be a starting point for the analyst who is conducting the test. If there are other tools that you are more familiar with or are better than the tools listed in this manual then by all means, use them. Make sure that they provide the same or more information as the tools listed in this manual.

The one important thing to bear in mind before actually beginning any tests is to be sure that the entire scope of the test is agreed upon and in writing. Make sure you have permission (and a written contract) to perform any and all of the tests that you will be using.

If there are any questions or suggestions regarding this document or its usage please contact David Buyer at [be281@bfn.org](mailto:be281@bfn.org)

# 1. Network Footprinting

---

Network footprinting is a way to gather target information and gauge the security posture of the organization to be tested. It is basically an introduction to the systems to be tested. It is best defined as a combination of data collection and information analysis. Although it is often advisable from a legal standpoint to define contractually exactly which systems to test if you are a third-party auditor or even if you are the system administrator, you may not be able to start with concrete system names or IP addresses. In this case you must survey and analyze. The point of this section is to find the number of reachable systems to be tested without exceeding the legal limits of what you may test. Therefore, network footprinting is one way to begin a test; another way is to be given the IP range to test. In this section, no intrusion is being performed directly on the systems except in places considered to be quasi-public domain.

Often times, more hosts are detected during actual testing. Please bear in mind that the hosts discovered later may be inserted in the testing set as a subset of the defined testing and often times only with permission or collaboration with the target organization's internal security team.

---

## 1.1 - Expected Results

- Domain Names
- Server Names
- IP Addresses
- Network Map
- ISP / ASP information
- System and Service Owners
- Possible test limitations

---

## 1.2 - Tasks

### 1.2.1 Name server responses

#### 1.2.1.1 – Use whois to find information on domain

*Try variations of the name, also check for recent mergers, acquisitions, or name changes. If you have trouble getting information from the command line whois then use Sam Spade.*

*Note: Depending on what you find, the nameserver example might change.*

```
whois Acme.net@whois.crsnic.net  
whois Acme.net.@whois.networksolutions.com
```

```
whois Acme Net.@whois.arin.net
```

The whois servers that we used in the previous example are just that. Examples. There are quite a few of these whois server now. Try multiple servers if there is no luck with these.

**Results:**

<b>Network Information</b>
<b>Registrar Query</b> INSERT DATA
<b>Domain Query</b> INSERT DATA
<b>Netblock Query</b> INSERT DATA

**1.2.1.2** Is hosting for servers different than network owned?

Sometimes a company will outsource their web, mail, etc to third parties. Put in any outsourced servers along with information below. Remove the example before inserting data.

**Results:**

Server Type	IP Address	Hosting Co.	Part of test?
Web	192.168.1.1	ABC Hosting Co.	Yes
Mail			
DNS			
File			

**1.2.1.3** Use DNS interrogation tools to discover information about known and unknown hosts. Query all DNS servers known to service target domain.

```
nslookup  
dig @dns.host.net acme.net any
```

If the domain is not part of the test, insert the network address. Insert data below

**Results:**

<b>Nslookup query</b> INSERT DATA
<b>Dig Query</b> INSERT DATA

**1.2.1.4** Question the primary, secondary, and ISP name servers for hosts and sub domains. *Again, you can use Sam Spade if you have difficulties with host.*  
host -l -v -t any **Acme.net**

**Results:**

**INSERT DATA**

**1.2.2** Examine the outer wall of the network in an effort to discover network topology.

**1.2.2.1** Use multiple traces to the gateway to define the outer network layer and routers.

*The following commands assume that they are being run from a Unix system. In most flavors of Unix the default packet traceroute sends is UDP as opposed to Windows, which is ICMP.*

*First try UDP packets*

```
traceroute -n mail.acme.net
```

**Results:**

**INSERT DATA**

*Next try ICMP packets*

```
traceroute -I -n mail.acme.net
```

**Results:**

**INSERT DATA**

*Finally try TCP packets*

```
tcptraceroute mail.acme.net
```

**Results:**

**INSERT DATA**

*Also use <http://visualroute.visualware.com> to investigate possible other entry points. Try several different origins and report the findings below.*

Origination: <b>INSERT DATA</b>
Results
<b>INSERT DATA</b>

Origination: INSERT DATA
Results
INSERT DATA

Origination: INSERT DATA
Results
INSERT DATA

### 1.2.3 Information Leakage

**1.2.3.1** Search USENET, newsgroups, job databases, search engines, etc for postings on server and application information, IT positions, security posture information, and anything else that you can find.

Attach as Appendix A

**1.2.3.2** Examine target web server source code and scripts for any information that will help map out the companies network. Also examine e-mail headers, bounced mails, and read receipts.

### 1.3 Notes

INSERT ANY NOTES

### 1.4 Tools

Program Name	Description
Traceroute	Uses ICMP and UDP requests
Tcptraceroute	Uses TCP packets
Whois	Queries a whois database
Host	Looks up host names using domain server
Dig	DNS interrogation tool
Nslookup	DNS interrogation tool
Sam Spade	General-purpose Internet utility

### 1.5 Websites

URL	Description
<a href="http://www.visualroute.com">www.visualroute.com</a>	Visually traceroute website

## 2. Port Scanning

---

Port scanning is the act of systematically scanning system ports on the transport and network level. Included here is also the validation of system reception to tunneled, encapsulated, or routing protocols. This section is used to enumerate live or accessible Internet services as well as penetrating the firewall to find additional live systems.

Each Internet enabled system has 65,535 TCP and UDP possible ports. However, it is not always necessary to test every port for every system. This is left to the discretion of the test team. Port numbers that are important for testing according to the service are listed with the task.

### 2.1 Expected Results

---

- IP addresses of live systems
- Open, closed or filtered ports
- List of discovered tunneled and encapsulated protocols
- List of discovered routing protocols supported
- Active services
- Network map

### 2.2 Tasks

---

#### 2.2.1 ICMP Sweeps

**2.2.1.1** If for some reason ICMP is allowed through the firewall then use nmap's ping sweep capabilities to determine the existence of all the hosts in a network

```
nmap -n -v -sP IP Address or Range
```

**Results:**

INSERT DATA

#### 2.2.2 Scan Systems and Ports/Services

**2.2.2.1** Perform a syn scan on the network. *If there is multiple networks or time is critical then consider using the -F option or using a higher level timeout option -TI-5*

```
nmap -v -n -P0 -sS -p 1-65535 IP Address or Range
```

**Results:**

INSERT DATA

2.2.2.2 Use UDP scans to enumerate ports as being open or closed on the default UDP testing ports. *UDP scans can take a long time to complete. If there are only a few IP addresses then remove the -F option and add -p 1-65535 to scan all ports.*

```
nmap -v -n -P0 -sU -F IP Address or Range
```

**Results:**

INSERT DATA

2.2.2.3 Consolidate the data gathered above (2.1, 2.2) and enter into the chart below.

**Results:**

Host	Port	State	Service
INSERT DATA	INSERT DATA	INSERT DATA	INSERT DATA

2.2.3 Encapsulated and tunneled protocols.

2.2.3.1 Verify and examine the use of NBT, IPX of IPX/SPX via TCP/IP, RPC and the DCE RPC, PPTP, L2TP, SNMP, GRE, IPSEC, Radius support, and any other protocol that the target organization might be using.

**Notes from 2.2.3.1:**

2.2.4 Routing protocols

2.2.4.1 Verify and examine the use of ARP, RIP, OSPF and Link State Advertisements, BGP, and any other routing protocols that might be used.

**Notes from 2.2.4.1:**

2.3 Notes

INSERT ANY NOTES

## 2.4 Tools

---

<b>Program Name</b>	<b>Description</b>
Nmap	Port Scanner and OS detection

## 3. System Fingerprinting

---

System fingerprinting is the active probing of a system for responses that can distinguish unique systems to operating system and version level. This section should be completed after port scanning a network (or a set of IP addresses) and determining “interesting” systems. Only active fingerprinting methods are used in this section. There are ways to passively fingerprint systems but they will not be discussed in this manual because it is in my opinion that the tools used for this still provide inaccurate results.

### 3.1 Expected Results

---

- OS type
- Patch level
- System type

### 3.2 Tasks

---

**3.2.1** Examine system responses to determine operating system type and patch level. Enter any information gathered in any other section in [this color font](#).

```
nmap -v -O HOST
```

**Results:**

**INSERT DATA**

**3.2.2** Match information gathered in sections 1, 2, and 3 to system responses for more accurate results

### 3.3 Notes

---

**INSERT ANY NOTES**

### 3.4 Tools

---

Program Name	Description
Nmap	Port Scanner and OS detection

## 4. Application probing

This is the active examination of the application listening behind the service. In certain cases more than one application exists behind a service where one application is the listener and the others are considered components of the listening application. A good example of this is PERL installed for use in a Web application. In that case the listening service is the HTTP daemon and the component is PERL.

### 4.1 Expected Results

- Service/Application Types and Patch Level

### 4.2 Tasks

**4.2.1** Identify server uptime, OS and any application information. *You can use the web site [www.netcraft.com](http://www.netcraft.com) to get this information (sometimes).*

**Results:**

Host	Uptime	OS/Info
INSERT DATA	INSERT DATA	INSERT DATA

**4.2.2** Identify the application behind the service and the patch level using banners or fingerprinting

**4.2.2.1** For http try netcat.

```
nc www.acme.net 80
Once connected type HEAD HTTP/1.0
Then hit the enter key a couple of times.
```

**Results:**

INSERT DATA

**4.2.2.2** For other ports

```
nc mail.acme.net Port
```

*For ports using SSL use stunnel to open up a tunnel and then run nc.*  
stunnel -d yourPCName:5000 -r www.acme.net:443 -c  
nc www.acme.net 5000

**Results:**

INSERT DATA

4.2.2.3 Consolidate the data gathered above (4.2.1 – 4.2.2.3) and enter into the chart below. Enter any information gathered in any other section in [this color font](#).

Host	Application	Patch level
INSERT DATA	INSERT DATA	INSERT DATA

4.2.3 Verify the application to the system and the version.

4.2.4 Identify the components of the listening service

4.3 Notes

INSERT ANY NOTES

4.4 Tools

Program Name	Description
Nc	Network Utility
Stunnel	SSL tunnel tool

4.5 Websites

URL	Description
<a href="http://www.netcraft.com">www.netcraft.com</a>	Network inquiry site

## 5. Automated Vulnerability Scanning

---

Testing for vulnerabilities using automated tools is an efficient way to determine existing holes and system patch level. Although many automated scanners are currently on the market, it is important for the tester to identify and incorporate the current “underground” scripts/exploits into this testing. The more experience the tester has the more methods will be known and therefore better results.

### 5.1 Expected Results

---

- List of system vulnerabilities
- Type of application or service by vulnerability
- Patch levels of systems and applications

### 5.2 Tasks

---

**5.2.1** Measure the target organizations security posture against the currently popular scanning tools. Verify the information gathered using alternate vulnerability scanners of choice or any other scanners that are available.

#### 5.2.1.1 Nessus

Insert the output of Nessus

**Results:**

INSERT DATA

#### 5.2.1.2 Nikto

Insert the output from nikto

**Results:**

INSERT DATA

### 5.3 Notes

---

INSERT ANY NOTES

### 5.6 Tools

---

<b>Program Name</b>	<b>Description</b>
Nessus	Automated Vulnerability Scanner
Nikto	Web Scanner

# 6. Exploit Research

---

This section covers the research involved in finding vulnerabilities up until the report delivery. This involves searching online databases and mailing lists specific to the systems being tested. Do not confine yourself to the web – consider using IRC, Newsgroups, and the underground FTP sites.

## 6.1 Expected Results

---

- Patch levels of systems and applications
- List of possible denial of service vulnerabilities and other vulnerabilities

## 6.2 Tasks

---

### 6.2.1 Identify all vulnerabilities according to applications

**6.2.1.1** Check the following sites databases for vulnerabilities that were identified in the automated vulnerability section:

<http://icat.nist.gov/icat.cfm>  
<http://www.securityfocus.com>  
<http://www.cert.org>

Note: These are not the only sites to check, they are only a starting point. Again, the more experience the tester has the more resources will be available.

**6.2.2** Identify all vulnerabilities. List their CVE (if one is available) numbers and any relevant information gathered. Also list any remediation steps acquired. Attach all vulnerabilities and related information on vulnerabilities as Appendix B

## 6.3 Notes

---

**INSERT ANY NOTES**

## 6.4 Websites

---

URL	Description
<a href="http://icat.nist.gov/icat.cfm">http://icat.nist.gov/icat.cfm</a>	Security information site
<a href="http://www.securityfocus.com">http://www.securityfocus.com</a>	Security information site
<a href="http://www.cert.org">http://www.cert.org</a>	Security information site

# 7. Manual System/Application Vulnerability Testing and Verification

---

This section is necessary for eliminating false positives, expanding the hacking scope, and discovering the data flow in and out of the network. Manual testing refers to a person or persons at a computer using creativity, experience, and ingenuity to test the target network. Application testing refers to the testing of non-daemon/daemon applications accessible from the Internet. This section is generally referred to as a Penetration Test.

## 7.1 Expected Results

---

- List of areas secured by obscurity or visible access.
- List of actual vulnerabilities minus false positives.
- List of systems that were actually compromised (*include screen shots*).

## 7.2 Tasks

---

### 7.2.1 Manually checking systems/applications

There are many means to accomplish this task (Too many to list as well as trying to keep up with the latest penetration method). Techniques to penetrate systems need to be understood and developed over time. The only way to become an effective Penetration tester is time and experience. The following is a list of starting points to start out the tests:

- Test exploits found in the Exploit Research section.
- Use your own methods to exploit systems/applications.
- Decompose or deconstruct if necessary to access the source code.
- Examine the processes of the application.
- Test the inputs of the application.
- Examine the outputs of the application.
- Examine the communications, trusts, and relationships of the application.
- Determine the limits of authentication and access control.
- Measure the limitation of the defined variables.
- Examine the use a caching.
- Exhaust all resources that are available in the scope of work.

Again, the preceding was just a starting point and is by no means and ultimate guide to penetrating systems and applications. Hey, the information is out there and as I said on the website, I wasn't going to reveal all my methods.

### 7.3 Notes

---

**INSERT ANY NOTES**

# 8. Firewall & Access Control List Testing

---

The firewall and screening router are two defenses often found on a network that control the flow of traffic between the enterprise network and the Internet. Both operate via security policies and ACLs. This section is designed to assure that only that which should be expressly permitted be allowed in the network, all else should be denied. However, this is often difficult when no written security policy exists and the analyst is forced to make assumptions as to the acceptable risk. This is not the job of the tester. The security tester must attempt to find the limits of the firewall and/or the screening router both as a system and as a service.

## 8.1 Expected Results

---

- Information on the firewall as a service and a system
- Information on the routers as a service
- Outline of the network security policy by the ACL
- List of the types of packets which may enter the network
- List of the types of protocols with access inside the network
- List of live systems found

## 8.2 Tasks

---

**8.2.1** Verify the firewall type. *Use information collected from intelligence gathering and port scans*

**Results:**

INSERT DATA

**8.2.2** Identify if the firewall is modern and capable of maintaining session state information.

```
hping www.acme.net -S -c 1 -p 139
hping www.acme.net -S -A -c 1 -p 139
hping www.acme.net -S -A -c 1 -p 135
```

**Results:**

INSERT DATA

**8.2.3** Verify the router types and configuration

**8.2.3.1** Use nmap to query the router. *This can also be used on the firewall.*

```
nmap -v -O router.acme.net
```

**Results:**

INSERT DATA

8.2.3.2 If telnet is open or any other port use netcat.

```
nc router.acme.net 23
```

**Results:**

INSERT DATA

8.2.3.3 If telnet is open then run Brutus to brute force the username (if applicable) and password.

**Results:**

INSERT DATA

8.2.3.4 If snmp is open then use the SolarWinds tools to brute force the community names and download the configuration file. *Once we download the configuration files we can decrypt the passwords using SolarWinds Router Password Decryption (that is, if it's a Cisco router. If the router is another type then research the methods to penetrate the system). SolarWinds is a commercial tool but you can use the trial version for 15 days with the same functionality as the licensed version. Brutus also has a Cisco password decrypt tool.*

**Results:**

INSERT DATA

8.2.3.5 If previous methods resulted in no ports open we can still check to see if the router is at least a Cisco.

Open up a sniffer session using either Ethereal or Sniffer then

```
Nmap -nvv -p1999 router.acme.net
```

*Examine the RST/ACK packet. If the router is a Cisco you should see the word "cisco" somewhere in the text.*

**Results:**

INSERT DATA

8.2.3.6 If the firewall is not passing ICMP echo requests then try other ICMP flags. Attempt to contact known sites with ICMP timestamp and address mask requests.

*Check known good hosts first*

```
sing -mask mail.acme.net -c1
```

```
ping -tstamp mail.acme.net -c1
```

**Results:**

**INSERT DATA**

**8.2.3.7** Test the ACL against the written security policy or against the “Deny All” rule. Review the list of open ports in section 2 to verify the written security policy.

**8.2.3.8** Verify that the firewall is egress filtering local network traffic.

**8.2.3.9** Verify that the firewall and/or router is performing address spoof detection.

**8.2.3.10** The best way to review a firewall is to actually have a printout of their firewall ruleset. Most firewalls have a graphical interface that allows you to print out the ruleset in a nice easy to read printout. Cisco PIX and routers have the Cisco PDM, Checkpoint is graphical already, and others will normally have this functionality. Get a printout and review the ruleset and note any inconsistencies.

**8.3** Notes

**INSERT ANY NOTES**

**8.4** Tools

<b>Program Name</b>	<b>Description</b>
nmap	Port Scanner and OS detection
hping2	Packet crafter
Nc	Network Utility
Brutus	Multiple application brute force tool
SolarWinds	Multiple commercial network tools
Ethereal	Network packet capture
Sniffer	Network packet capture
Sing	ICMP tool

# 9. Intrusion Detection System (IDS) Testing

---

This section is focused on the performance and sensitivity of the IDS, and also, the overall Intrusion Detection posture of the company. Much of this testing cannot be properly achieved without access to the IDS logs and information given by the company itself. Some of these tests are also subject to attacker bandwidth, hop distance, and latency and might affect the outcome of these tests.

## 9.1 Expected Results

---

- Type of IDS
- Note of IDS performance under heavy load
- Type of packets dropped or not scanned by the IDS
- Type of protocols dropped or not scanned by the IDS
- Note of reaction time and type of the IDS
- Note of IDS sensitivity
- Rule map of IDS

## 9.2 Tasks

---

**9.2.1** Verify all IDS types, placements, and policies. This information will need to be gathered by interviewing key people within the organization. The following is a list of starting points to bring up in the interview process. Once the interviews are underway more question will undoubtedly be brought up as well.

Type of IDS

The placement of the IDS's.

How the alarms are handled?

How information is logged, tracked, and monitored?

How are attacks blocked?

How are signatures updated?

Again, the preceding questions are just so you can get started in the interview process.

**9.2.2** Verify performance and sensitivity of IDS. The following is a list of starting points to consider while testing:

Test the IDS for configured reactions to multiple, varied attacks.

Test the IDS for configured reactions to obfuscated URLs.

Test the IDS for configured reactions to speed adjustments in packet sending.

Test the IDS for configured reactions to source port adjustments.

Test the IDS for the ability to handle fragmented packets.

Test the IDS for configured reactions to the network traffic listening configuration in the designated network segments(s).

Test the IDS for alarm states.

Test the signature sensitivity settings over 1 minute, 5 minutes, 60 minutes, and 24 hours.

Test the effect and reactions of the IDS against a single IP address versus various addresses.

### 9.3 Notes

---

**INSERT ANY NOTES**

# 10. Password Cracking

---

Password cracking is the process of validating password strength through the use of automated password recovery tools that expose either the application of weak cryptographic algorithms, incorrect implementation of cryptographic algorithms, or weak passwords due to human factors.

Once gaining administrator or root privileges on a computer system, password cracking may assist in obtaining access to additional systems or applications and is a valid technique that can be used for system leverage throughout a security test. Corporate wide password cracking can also be performed as a simple “after action” exercise and may highlight the need for stronger encryption algorithms for key systems storing passwords, as well as highlight a need for enforcing the use of stronger user passwords through stricter policy, automatic password generation, or pluggable authentication modules.

## 10.1 Expected Results

---

- Password file cracked
- List of login IDs with user or system passwords
- List of systems vulnerable to crack attacks
- List of documents or files vulnerable to crack attacks
- List of systems with user or system login IDs using weak passwords

## 10.2 Tasks

---

**10.2.1** Obtain the password file from the system that stores usernames and passwords

1. *For Unix systems, this will be either /etc/passwd or /etc/shadow*
2. *For Unix systems that happen to perform SMB authentication, you can find NT passwords in /etc/smbpasswd*
3. *For NT systems, use pwdump to attain the hashed passwords*

**10.2.2** Run a password cracking attack on the password file using LOPHT crack. Set it to use an automated dictionary attack and a brute force attack/hybrid attack. *If the passwords are from a Unix system, then use John the Ripper.*

```
John -w:Wordlist PasswordFile
```

### Results:

Number of Passwords to be cracked

Passwords	Percentage	Number
Passwords that were cracked	INSERT DATA	INSERT DATA
Company default/modified passwords		
Less that 8 characters		
Blank password		
Password same as username		

Number of passwords cracked instantly		
Number of passwords cracked in 15 minutes		
Number of passwords cracked in 30 minutes		
Number of passwords cracked in 1 hour		
Number of passwords cracked in 2 hours		
Number of passwords cracked in 10 hours		

**10.2.3** Run automated password crackers on encrypted files that are encountered in an attempt to gather more intelligence and highlight the need for stronger document or file system encryption.

**10.2.4** Run brute force attacks against systems/applications to demonstrate need for stronger passwords or different authentication schemes.

**10.2.4** Verify password aging, account lockout, password length, and password uniqueness.

**Results:**

Password Aging	
Account Lockout	
Password Length	
Password Uniqueness	

**10.3** Notes

INSERT ANY NOTES

**10.4** Tools

Program Name	Description
Pwdump	Dumps hashed passwords from the SAM
L0pht Crack	Password cracking program
John the Ripper	Password cracking program

# 11. Denial of Service Testing

---

**Warning:** Only perform DoS procedures if client has signed off and authorized these tests. They must be closely monitored at all times.

Denial of Service (DoS) is a situation where a circumstance, either intentionally or accidentally, prevents the system from functioning as intended. In certain cases, the system may be functioning exactly as designed, however, it was never intended to handle the load, scope, or parameters being imposed upon it.

---

## 11.1 Expected Results

- List weak points in the Internet presence including single points of failure
- Establish a baseline for normal use
- List system behaviors to heavy use
- List DoS vulnerable systems

---

## 11.2 Tasks

**11.2.1** Verify that administrative accounts and system files and resources are secured properly and all access is granted with “Least Privilege”.

**11.2.2** Check the exposure restrictions of systems to non-trusted networks

**11.2.3** Verify that baselines are established for normal system activity and that procedures are in place to respond to irregular activity.

**11.2.4** Test heavy server and network loads. *There are a tons of tools to accomplish this out there. Use multiple traffic generators and different protocols in this test.*

**11.2.5** Run authorized Nessus DoS scans against targets.

---

## 11.3 Notes

**INSERT ANY NOTES**

---

## 11.4 Tools

Program Name	Description
Nessus	Automated Vulnerability Scanner

# 12. Wireless Leak Tests

---

Wireless tests are designed to map out the distance of the wireless network and to ascertain what vulnerabilities exist as a result of these tests. Many times wireless networks are implemented without security taken into account. Some wireless networks will allow anyone with a wireless network card to receive a DHCP address and link up with the network. This would mean that anyone in a car driving down the street would be able to access the wireless network.

## 12.1 Expected Results

---

- Find the out-most wireless edge of the network
- Find access points into the network

## 12.2 Tasks

---

**12.2.1** Verify the distance in which the wireless communication extends beyond the physical boundaries of the organization.

**12.2.2** Discover all Access Points (AP) on network. Verify that the communication is secure and cannot be challenged or tampered

**12.2.2.1** Use Netstumbler to find the Access Points, default SSID's, and encryption used. Netstumbler uses the Broadcast Probe Request to find AP's, so if this is disabled on the AP you are out of luck. Try Kismet or AirSnort if there is no luck with Netstumbler. Actually, AirSnort is also capable of sniffing traffic and cracking WEP keys.

*Some default SSID's are:*

*Cisco – tsunami*

*Netgear – wireless*

*Linksys – linksys*

*D-Link – default*

### Results:

**INSERT DATA**

**12.2.3** Once AP's are located proceed to sniff wireless network traffic. AirSnort or Ethereal do good jobs at this.

**12.2.4** Crack WEP keys. Use AirSnort for this.

### 12.3 Notes

---

**INSERT ANY NOTES**

### 12.4 Tools

---

<b>Program Name</b>	<b>Description</b>
NetStumbler	Wireless AP seeking tool
AirSnort	Wireless packet capture
Kismet	Wireless utility

# 13. PBX and Dial-up

---

Securing the organization's Private Branch Exchange (PBX) systems will help prevent toll-fraud and theft of information. In addition to PBX testing, finding all dial-up access points into the network (Wardialing) will help establish a starting point to implement an inventory of all dial-up access points.

It is very important that Wardialing receives additional support from the organization and is closely monitored. In some locals dialing large quantities of numbers in sequence is illegal. Consult with the organization on this matter before proceeding with Wardialing efforts.

## 13.1 Expected Results

---

- Find voice mailboxes that are world accessible
- Find PBX systems that are allowing remote administration
- List systems allowing world access to the maintenance terminal
- List all listening and interactive telephony systems
- Find all dial-in servers and modems attached to users systems

## 13.2 Tasks

---

13.2.1 Verify that voicemail PINS are changed often

13.2.2 Review call detail logs for signs of abuse

13.2.3 Ensure administrative accounts don't have default, or easily guessed passwords

13.2.4 Make sure OS is up to date and patched

13.2.5 Check for remote maintenance access to system.

13.2.6 Check the physical security of the maintenance terminal

13.2.7 Identify modems, faxes, and automated operators

13.2.8 Test dial-in authentication

13.2.9 Acquire number blocks.

**13.2.9.1** Check the Internet, phone directories, the InterNIC, and any other method to acquire numbers.

**Results:**

**INSERT DATA**

**13.2.10.2** Once the number blocks are discovered proceed with Wardialing. *There are a few tools to use for wardialing. It all depends on what you want to spend on them. Tools like THC-Scan and ToneLoc are free but lack in functionality. Tools like Phone Sweep are full of nice features but can be costly.*

**13.3** Notes

---

**INSERT ANY NOTES**

# 14. Internal Auditing

---

Securing the organization's external network is only half the battle. An attacker on the internal network can cause much havoc. Additionally, disgruntled employees (or just regular employees for that matter) account for over 70% of security breaches. Securing the internal network should be performed along with any security audit. If it is a large company then you must first agree upon a scope of work for the internal audit. This is usually confined to critical servers or a percentage (sample) of the network.

Any findings from the external audit will be valid for the internal audit but we will not be re-printing any of the findings from the external audit in the internal audit. Use all the sections that are relevant for the internal audit and use **this color font** for any internal findings.

## 14.1 Expected Results

---

- Open, closed or filtered ports
- Active services
- OS type, Patch level
- List of system vulnerabilities

## 14.2 Tasks

---

**14.2.1** Check the internal network for a switched or non-switched architecture. *This can be accomplished using any sniffer and looking at the output. A switched network will only have entries from the system that your currently on and broadcast traffic.*

**Results:**

**SWITCHED OR NON-SWITCHED**

**14.2.1.1** If the network is switched then use arpspoof and dsniff to look for usernames and passwords that are sent over the network in plain text.

Enable ip-forwarding then use:

```
arpspoof TARGET ADDRESS 2>/dev/null 1>/dev/null &  
dsniff -n -w file
```

*Might also want to try the rest of the tools that come along with dsniff too.*

**Results:**

**INSERT DATA**

**14.2.1.2** If the network is non-switched then use dsniff to look for usernames and passwords that are sent over the network in plain text.  
dsniff -n -w file

**Results:**

INSERT DATA

**14.2.2** Use LanGuard to gather information on the network. *If this is a large network then take samples for each network.*

**Results:**

INSERT DATA

**14.2.3** Use MBSA to check security of Microsoft systems. *If this is a large network then take samples for each network. You must also have administrative rights to run on these systems.*

**Results:**

INSERT DATA

**14.2.4** Complete relevant parts of sections 2 - 7

**14.2.5** If any systems are using SNMP then use the tools in SolarWinds to brute force entry into these systems.

**14.3** Notes

INSERT ANY NOTES

**14.4** Tools

<b>Program Name</b>	<b>Description</b>
Arpspoof	Arp spoofing tool
dsniff	Network sniffer
MBSA	Microsoft security analyzer
SolarWinds	Multiple commercial network tools
LanGuard	Network tool

# Appendix A

## Information Leakage

# Appendix B

## Vulnerabilities and Related Information