Unit 4 – CFCS

W*ar dialing* — the act of using a computer to scan other computers automatically for accessible modems.

Chapter 9

Network consists of such devices as routers, firewalls, and even generichosts (including servers and workstations) that you must assess as part ofthe ethical hacking process. There are thousands of possible network vulnerabilities, equally as many tools, and even more testing techniques. You can eliminate many well-known network vulnerabilities by simply patching your network hosts with the latest vendor software and firmware patches.

Network infrastructure vulnerabilities :
--are the foundation for all technical security issues in your information systems. These lower-level vulnerabilities affect everything running on your network. That's why you need to test for them and eliminate them whenever possible.

Network infrastructure security involves assessing such areas as:
_ Where such devices as a firewall or IDS (intrusion detection system) are placed on the network and how they are configured
_ What hackers see when they perform port scans and how they can exploit vulnerabilities in your network hosts
_ Network design, such as Internet connections, remote-access capabilities, layered defenses, and placement of hosts on the network
_ Interaction of installed security devices
_ Protocols in use
_ Commonly attacked ports that are unprotected
_ Network host configuration
_ Network monitoring and maintenance

If any of these network security issues is exploited, bad things can happen:
_ A DoS attack can take down your Internet connection — or even your entire network.
_ A hacker using a network analyzer can steal confidential information in e-mails and files being transferred.
_ Backdoors into your network can be set up.
_ Specific hosts can be attacked by exploiting local vulnerabilities across the network.

Choosing Tools :

Sam Spade for Windows (samspade.org/ssw) for network queries from DNS lookups to traceroutes
_ **SuperScan** (www.foundstone.com) for ping sweeps and port scanning
_ **NetScanTools Pro** (www.netscantools.com) for dozens of network security-assessment functions, including ping sweeps, port scanning, and SMTP relay testing
_ **Nmap** (www.insecure.org/nmap) or **NMapWin** (sourceforge.net/ projects/nmapwin) as a happy-clicky-GUI front end for host-port probing and operating-system fingerprinting

_ **Netcat** (www.atstake.com/research/tools/network_utilities) the most versatile security tool for such security checks as port scanning and firewall testing
_ **WildPackets EtherPeek** (www.wildpackets.com) for network analysis.

Vulnerability assessment tools :

These vulnerability-assessment tools will allow you to test your network hosts for various known vulnerabilities as well as potential configuration issues that could lead to security exploits:
_ **GFI LANguard Network Security Scanner** (www.gfi.com) for port scanning and other vulnerability testing
_ **Nessus** (www.nessus.org) as a free all-in-one tool for such tests as ping sweeps, port scanning, and vulnerability testing
_ **Qualys QualysGuard** (www.qualys.com) as a great all-in-one tool for indepth vulnerability testing, if you can justify the cost.

Scanning, Poking, and Prodding

Performing these ethical hacks on your network infrastructure involves following basic hacking steps:
1. Gather information and map your network.
2. Scan your systems to see which are available.
3. Determine what's running on the systems discovered.
4. Attempt to penetrate the systems discovered, if you choose to.

Port scanners

A port scanner shows you what's what on your network. It's a software tool that basically scans the network to see who's there. Port scanners provide basic views of how the network is laid out. They can help identify unauthorized hosts or applications and network host configuration errors that can cause serious security vulnerabilities.
The real trick to assessing your overall network security is interpreting the results you get back. You can get false positives on open ports, and you may have to dig deeper. For example, UDP scans — like the protocol itself — are less reliable than TCP scans and often produce false positives, because many applications don't know how to respond to random incoming UDP scans.
Port-scan tests take time. The length of time depends on the number of hosts you have, the number of ports you scan, the tools you use, and the speed of your network links.Scan more than just the important hosts. Also, perform the same tests with different utilities to see whether you get different results. Not all tools find the same open ports and vulnerabilities.

As an ethical hacker, you should scan all 65,535 UDP and 65,535 TCP ports on each network host that's found by your scanner. If you find questionable ports, look for documentation that the application is known and authorized.

Table 9-1 Commonly Hacked Ports

Port Numbers	*Service*	*Protocols*
7	Echo	TCP, UDP
19	Chargen	TCP, UDP
20	FTP data (File Transfer Protocol)	TCP
21	FTP control	TCP
22	SSH	TCP
23	Telnet	TCP
25	SMTP (Simple Mail Transfer Protocol)	TCP
37	Daytime	TCP, UDP
53	DNS (Domain Name System)	UDP
69	TFTP (Trivial File Transfer Protocol)	UDP
79	Finger	TCP, UDP
80	HTTP (Hypertext Transfer Protocol)	TCP
110	POP3 (Post Office Protocol version 3)	TCP
111	SUN RPC (remote procedure calls)	TCP, UDP
135	RPC/DCE end point mapper for Microsoft networks	TCP, UDP
137, 138, 139	NetBIOS over TCP/IP	TCP, UDP
161	SNMP (Simple Network Management Protocol)	TCP, UDP
220	IMAP (Internet Message Access Protocol)	TCP
443	HTTPS (HTTP over SSL)	TCP
512, 513, 514	Berkeley *r* commands (such as rsh, rexec, and rlogin)	TCP

Ping Sweep :

A ping sweep of all your network subnets and hosts is a good way to find out which hosts are alive and kicking on the network. A ping sweep is when you ping a range of addresses using Internet Control Message Protocol (ICMP) packets.
Dozens of Nmap command-line options exist, which can be overwhelmingwhen you just want to do a basic scan. You can just enter nmap on the command line to see all the options available. These command-line options can be used for an Nmap ping sweep:
_ -sP tells Nmap to perform a ping scan.
_ -n tells Nmap to not perform name resolution.

Port scanning
Most port scanners operate in three steps:

1. The port scanner sends TCP SYN requests to the host or range of hostsyou set it to scan. Some port scanners, such as SuperScan, perform ping sweeps to determine which hosts are available before starting the TCP port scans. Most port scanners scan only TCP ports by default. Don't forget about UDP ports. You can scan UDP ports with a UDP port scanner such as Nmap LANguard Network Security Scanner.

2. The port scanner waits for replies from the available hosts.
3. The port scanner probes these available hosts for up to 65,535 possible TCP and UDP ports — based on which ports you tell it to scan — to see which ones have available services on them.

The port scans provide the following information about the live hosts on your network:
_ Hosts that are active and reachable through the network
_ Network addresses of the hosts found
_ Services or applications that the hosts *may be* running.

1- A good tool to perform generic TCP port scans is SuperScan.
2- Nmap

Nmap helps to verify that the ports are actually open
and not being reported as a false positive. Nmap allows you to run the following additional scans:
_ **Connect:** This basic TCP scan looks for any open TCP ports on the host. You can use this scan to see what's running and determine whether IDSs, firewalls, or other logging devices log the connections.
_ **UDP Scan:** This basic UDP scan looks for any open UDP ports on the host. You can use this scan to see what's running and determine whether IDSs, firewalls, or other logging devices log the connections.
_ **SYN Stealth:** This scan creates a half-open TCP connection with the host possibly evading IDS systems and logging. This is a good scan for testing IDSs, firewalls, and other logging devices.
_ **FIN Stealth, Xmas Tree, and Null:** These scans let you mix things up a bit — no pun intended — by sending strangely formed packets to your network hosts so you can see how they respond. These scans basically change around the flags in the TCP headers of each packet, which allows you to test how each host handles them to point out weak TCP/IP implementations and patches that may need to be applied.

Countermeasures :

1- Traffic restriction

Enable only the traffic you need to access internal hosts — preferably as far as possible from the hosts you're trying to protect. You apply these rules in two places:

_ External router for inbound traffic

_ Firewall for outbound traffic

Configure firewalls to look for potentially malicious behavior over time (such as the number of packets received in a certain period of time), and have rules in place to cut off attacks if a certain threshold is reached, such as 100 port scans in one minute.

2- Gathering network information

NetScanTools Pro is a great tool for general network information, such as the number of unique IP addresses, NetBIOS names, and MAC addresses found.

3- Traffic denial

Deny ICMP traffic to specific hosts you're trying to protect. Most hosts don't need to have ICMP enabled — especially inbound ICMP requests — unless it's needed for a network management system that monitors hosts using this protocol.

SNMP scanning

SNMP scanning Simple Network Management Protocol (SNMP) is a protocol built into virtually every network device. Network management programs (such as HP OpenView and LANDesk) use SNMP for remote network host management.

Vulnerabilities :

Most network hosts run SNMP that isn't hardened or patched to prevent known security vulnerabilities. The majority of network devices have SNMP enabled and don't even need it! If SNMP is compromised, a hacker can gather such network information as ARP tables and TCP connections to attack your systems. If SNMP shows up in port scans, you can bet that a hacker will try to compromise the system.

Here are some other utilities for SNMP enumeration:

The commercial tool SolarWinds (www.solarwinds.net)

Free Windows GUI-based Getif (www.wtcs.org/snmp4tpc/getif.htm)

Text-based SNMPUTIL for Windows

Countermeasures:

Always disable SNMP on hosts if you're not using it — period.

Block the SNMP port (UDP port 161) at the network perimeter.

Change the default SNMP community string from public to another value that's more difficult to guess. This makes SNMP harder to hack.

Banner grabbing

Banners are the welcome screens that divulge software version numbers and other host information to a network host. This banner information may identify the operating system, the version number, and the specific service packs, so hackers know possible vulnerabilities.

You can grab banners by using either Telnet or Netcat.

Telnet : You can telnet to hosts on the default telnet port (TCP port 23) to see whether you're presented with a login prompt or any other information.

Just enter the following line at the command prompt in Windows or UNIX:

telnet ip_address

You can also telnet to other commonly used ports with these commands:

SMTP: telnet ip_address 25

HTTP: telnet ip_address 806

Netcat:

Netcat can grab banner information from routers and other network hosts, such as a wireless access point or managed Ethernet switch.

The following steps bring back information about a host that runs a Web server for remote management purposes:

1. Enter the following line to initiate a connection on port 80: nc –v ip_address 80

2. Wait for the initial connection. Netcat returns the message hostname [ip_address] 80 (http) open.

3. Enter the following line to grab the home page of the Web server: GET / HTTP/1.0

Countermeasures :

The following steps can reduce the chance of banner-grabbing attacks:

If there is no business need for services that offer banner information, disable those unused services on the network host.

If there is no business need for the default banners, or if you can customize the banners displayed, configure the network host's application or operating system to either disable the banners or remove information from the banners that could give an attacker a leg up.

Firewall rules :

As part of your ethical hacking, you can test your firewall rules to make sure they're working like they're supposed to.

Testing :

A few tests can verify that your firewall actually does what it says it's doing.

Some security-assessment tools can not only test for open ports, but also determine whether traffic is actually allowed to pass through the firewall. All-in-one tools have broad testing capabilities that make the network scanning process a lot less painful and can save you tons of time.

Nessus, QualysGuard, and GFI LANguard Network Security Scanner provide similar results. These tools generally identify open ports on the test network and presents information on SNMP, operating-system information, and special alerts to look for.

Netcat :Netcat can test certain firewall rules without having to test a production system directly. For example, you can check whether the firewall allows port 23 (telnet) through.

Follow these steps to see whether a connection can be made through port 23:

1. Load Netcat on a client machine inside the network. This allows you to test from the inside out
2. Load Netcat on a testing computer outside the firewall. This allows you to test from the outside in.
3. Enter the Netcat listener command on the client (internal) machine with the port number you're testing.
   For example, if you're testing port 23, enter this command: nc –l –p 23 cmd.exe
4. Enter the Netcat command to initiate an inbound session on the testing (external) machine. You must include the following information: • The IP address of the internal machine you're testing • The port number you're testing
   For example, if the IP address of the internal (client) machine is 10.11.12.2 and the port is 23, enter this command:
   nc –v 10.11.12.2 23

   Alternative testing tools: These utilities test firewall rules more robustly than Netcat:
   Firewalk: A UNIX-based tool (www.packetfactory.net/firewalk)
   Firewall Informer: A commercial tool by BLADE Software (www. blade-software.com)

Countermeasures :
The following countermeasures can prevent a hacker from testing your firewall:
Limit traffic to what's needed.
Set rules on your firewall (and router, if needed) to pass only traffic that you absolutely must pass. For example, have rules in place that allow HTTP inbound to an internal Web server and outbound for external Web access. This is the best defense against someone poking at your firewall.
Block ICMP to help prevent abuse from some automated tools, such as Firewalk.
Enable stateful packet inspection on the firewall, if you can. It can block unsolicited requests.8

Network analyzer
A network analyzer is a tool that allows you to look into a network and analyze data going across the wire for network optimization, security, and/or troubleshooting purposes. Network analyzers are often generically referred to as sniffers, though that's actually the name and trademark of a specific product from Network Associates, Sniffer (the original network-analysis tool).
A network analyzer is handy for sniffing packets.
Watch for the following network traffic behavior: What do packet replies look like? Are they coming from the host you're testing or from an intermediary device? Do packets appear to traverse a network host or security device, such as a router, a firewall, IDS, or a proxy server? When assessing security and responding to security incidents, a network analyzer can help you View anomalous network traffic and even track down an intruder. Develop a baseline of network activity and performance before a security incident occurs, such as protocols in use, usage trends, and MAC addresses.
When your network behaves erratically, a network analyzer can help you :
• Track and isolate malicious network usage
• Detect malicious Trojan-horse applications
• Monitor and track down DoS attacks
You can use one of the following programs for network analysis:
EtherPeek by WildPackets (www.wildpackets.com). It delivers a ton of features that the higher-end network analyzers of yesterday have for a fraction of their cost.
EtherPeek is available for the Windows operating systems. I download the open-source Ethereal network analyzer from www. ethereal.org8.

Two other powerful and free utilities can perform such functions as network analysis:
• ettercap (ettercap.sourceforge.net) for Windows and UNIXbased operating systems. I cover ettercap in more detail in "ARP spoofing," later in the chapter.
• dsniff (www.monkey.org/~dugsong/dsniff) for UNIX-based operating systems.

A network analyzer is just software running on a computer with a network card. It works by placing the network card in promiscuous mode, which enables the card to see all the traffic on the network, even traffic not destined to the network-analyzer host. The network analyzer performs the following functions: Captures all network traffic Interprets or decodes what is found into a human-readable format Displays it all in chronological order
Here are a few caveats for using a network analyzer:
To capture all traffic, you must connect the analyzer to either
• A hub on the network
• A monitor/span/mirror port on a switch9

Whether you connect your network analyzer inside or outside your firewall, you see immediate results. It can be an overwhelming amount of information, but you can look for these issues first:
Odd traffic, such as • Unusual amount of ICMP packets
• Excessive amounts of multicast or broadcast traffic
• Packet types that don't belong, such as NetBIOS in a NetWare environment Internet usage habits, which can help point out malicious behavior of a rogue insider or system that has been compromised, such as • Web surfing • E-mail • IM Questionable usage, such as
• Many lost or oversized packets
• High bandwidth consumption that may point to a Web or FTP server that doesn't belong

Reconnaissance probes and system profiling from port scanners and vulnerability-assessment tools, such as a significant amount of inbound traffic from unknown hosts — especially over ports that are not used very much, such as FTP or telnet.
Hacking in progress, such as tons of inbound UDP or ICMP echo requests, SYN floods, or excessive broadcasts. Nonstandard host names on your network. For example, if your systems are named Computer1, Computer2, and so on, a computer named GEEKz4evUR should raise a red flag.
Hidden servers (especially Web, SMTP, FTP, and DHCP) that may be eating network bandwidth or serving illegal software or even access into your network hosts.
Attacks on specific applications that show such commands as /bin/rm,9

Before getting started, configure your network analyzer to capture and store the most relevant data:
If your network analyzer permits it, configure your network analyzer software to use a first-in, first-out buffer.9

If your network analyzer permits it, record all the traffic into a capture file, and save it to the hard drive. This is the ideal scenario — especially if you have a large hard drive, such as 50GB or more. You can easily fill a several-gigabyte hard drive in a short period of time.

When network traffic doesn't look right in a network analyzer, it probably isn't. It's better to be safe than sorry10.

Countermeasures :

A network analyzer can be used for good or evil. A few countermeasures can help prevent someone from using an unauthorized network analyzer, but there's no way to completely prevent it. If hackers can connect to your network (physical or wireless), they can capture packets on the network, even if you're using a switch.

1-Physical security :Ensure that adequate physical security is in place to prevent a hacker from plugging into your network: Keep the bad guys out of your server room and wiring closet. A special monitor port on a switch where a hacker can plug in a network analyzer is especially sensitive. Make sure it's extra secure. Make sure that such unsupervised areas as unoccupied desks don't have live network connections.

2- Network-analyzer detection

You can use a network- or host-based utility to determine if someone is running an unauthorized network analyzer on your network:

sniffdet (sniffdet.sourceforge.net) for UNIX-based systems

PromiscDetect (ntsecurity.nu/toolbox/promiscdetect) for Windows.

These tools enable you to monitor the network for Ethernet cards that are running in promiscuous mode. You simply load the programs on your computer, and the programs alert you if they see promiscuous behaviors on the network (sniffdet) or local system (PromiscDetect).

The MAC-daddy attack :

Attackers can use ARP (Address Resolution Protocol) running on your network to make their systems appear to be either your system or another authorized host on your network.

ARP spoofing : An excessive amount of ARP requests can be a sign of an ARP poisoning attack (or ARP spoofing) on your network. What happens is that a client running a program such as the UNIX-based dsniff or the UNIX- and DOS/Windows-based ettercap can change the ARP tables — the tables that store IP addresses to media access control (MAC) mappings — on network hosts. This causes the victim computers to think they need to send traffic to the attacker's computer, rather than the true destination computer, when communicating on the network. This is often referred to as a Man-in-the-Middle (MITM) attack.

Here's a typical ARP spoofing attack with a hacker's computer (Hacky) and two legitimate network users' computers (Joe and Bob):

1. Hacky poisons the ARP caches of victims Joe and Bob by using dsniff, ettercap, or a utility he wrote.
2. Joe associates Hacky's MAC address with Bob's IP address.
3. Bob associates Hacky's MAC address with Joe's IP address.
4. Joe's traffic and Bob's traffic are sent to Hacky's IP address first.
5. Hacky's network analyzer captures Joe's traffic and Bob's traffic.

If Hacky is configured to act like a router and forward packets, it forwards the traffic to its original destination. The original sender and receiver never know the difference!

MAC-address spoofing

MAC-address spoofing tricks the *switch* into thinking you (actually, your computer) are someone else. You simply change your MAC address and masquerade as another user.

You can use this trick to test such access control systems as your IDS, firewall, and even operating-system login controls that check for specific MAC addresses.

UNIX-based systems

In UNIX and Linux, you can spoof MAC addresses with the ifconfig utility.

Follow these steps:

1. While logged in as root, use ifconfig to enter a command that disables the network interface. Insert the network interface number that you want to disable (usually, eth0) into the command, like this:

[root@localhost root]# ifconfig eth0 down

2. Enter a command for the MAC address you want to use.

Insert the fake MAC address and the network interface number (eth0) into the command again, like this:

[root@localhost root]# ifconfig eth0 hw ether *new_mac_address*

You can use a more feature-rich utility called MAC Changer.

Windows

You can use regedit to edit the Windows Registry, or a windows utility called SMAC (www.klcconsulting.net/smac), which makes MAC spoofing a simple process.

Follow these steps to use SMAC:

1. Load the program.

2. Select the adapter for which you want to change the MAC address.

3. Enter the new MAC address in the New Spoofed MAC Address fields, and click Update MAC.

4. Stop and restart the network card with these steps:
i. Right-click the network card in Network and Dialup Connections.
ii. Select Disable, and then right-click again and click Enable for the change to take effect. You may have to reboot for this to work properly.

Countermeasures:
A few countermeasures on your network can minimize the effects of a hacker attack against ARP and MAC addresses on your network.
Prevention
You can prevent MAC-address spoofing if your switches can enable port security to prevent automatic changes to the switch MAC address tables.
No realistic countermeasures for ARP poisoning exist. The only way to prevent ARP poisoning is to create and maintain static ARP entries in your switches for every host on the network. This is definitely something that no network administrator has time to do!
Detection
You can detect these two types of hacks through either an IDS or a stand-alone MAC address monitoring utility.

Denial of service

Denial-of-service (DoS) attacks are among the most common hacker attacks. A hacker initiates so many invalid requests to a network host that it uses all its resources responding to them and ignores legitimate requests.
DoS attacks
The following types of DoS attacks are possible against your network and hosts, and can cause systems to crash, data to be lost, and every user to jump on your case, wondering when Internet access will be restored.
Here are some common DoS attacks:
_ **SYN floods:** The attacker literally floods a host with TCP SYN packets.
_ **Ping of Death:** The attacker sends IP packets that exceed the maximum length of 65,535 bytes, which can ultimately crash the TCP/IP stack on many operating systems.
_ **WinNuke:** This attack can disable networking on older Windows 95 and NT computers.

Distributed attacks
Distributed DoS (DDoS) attacks have an exponentially greater impact on their victims. The most famous was the DDoS attack against eBay, Yahoo. These are some common distributed attacks:

_ **Smurf attack:** An attacker spoofs the victim's address and sends ICMP echo request (ping packets) to the broadcast address. The victim computer gets deluged with tons of packets in response to those echo requests.
_ **Trinoo and Tribe Flood Network (TFN) attacks:** Sets of client- and server-based programs launch packet floods against a victim machine, effectively overloading it and causing it to crash.
DoS attacks can be carried out with tools that the hacker either writes or downloads off the Internet. These are good tools to test your network's IDS/IDP and firewalls.

Countermeasures
Most DoS attacks are difficult to predict, but they can be easy to prevent:
_ Test and apply security patches as soon as possible for such network hosts as routers and firewalls, as well as for server and workstation operating systems.
_ Use IDS and IDP systems to monitor regularly for DoS attacks. You can run a network analyzer in *continuous capture* mode if you can't justify the cost of an all-out IDS or IDP solution.
_ Configure firewalls and routers to block malformed traffic. You can do this only if your systems support it, so refer to your administrator's guide for details.
_ Minimize IP spoofing by either
• Using authentication and encryption, such as a Public Key Infrastructure (PKI)
• Filtering out external packets that appear to come from an internal address, the local host (127.0.0.1), or any other private and nonroutable address such as 10.x.x.x, 172.16.x.x–172.31.x.x, or
192.168.x.x
_ Block all ICMP traffic inbound to your network unless you specifically need it. Even then, you should allow it only in to specific hosts.
_ Disable all unneeded TCP/UDP small services (such as echo and chargen).

General network defenses
_ Stateful inspection on firewalls. This can help ensure that all traffic traversing it is legitimate and can prevent DoS attacks and other spoofing attacks.
_ Rules to perform packet filtering based on traffic type, TCP/UDP ports, IP addresses, and even specific interfaces on your routers before the traffic is ever allowed to enter your network.
_ Proxy filtering and Network Address Translation (NAT).
_ Finding and eliminating fragmented packets entering your network (from Fraggle or other type of attack) via an IDS or IDP system.

Chapter 10

Wireless local area networks (WLANs) — specifically, the ones based on the IEEE 802.11 standard — are increasingly being deployed into both business and home networks.

WLANs are very susceptible to hacker attacks — even more so than wired Networks.

If a hacker comprises your WLAN, you can experience the following problems:

_Loss of network access, including e-mail, Web, and other services that can cause business downtime

_ Loss of confidential information, including passwords, customer data, intellectual property, and more

_ Legal liabilities associated with unauthorized users

Most of the wireless vulnerabilities are in the 802.11 protocol and within wireless *access points* (APs) — the central hublike devices that allow wireless clients to connect to the network.

WLAN Security tools :

_NetStumbler (www.netstumbler.com) for AP discovery and enumeration _ Wireless client management software — such as Orinoco's Client Manager software — for AP discovery and enumeration.

_WildPackets' AiroPeek (www.wildpackets.com) or your favorite WLAN analyzer for detailed information on wireless hosts, decryption of encrypted traffic, and more

_ LANguard Network Security Scanner (www.gfi.com) for WLAN enumeration and vulnerability scanning

An external antenna is also something to consider as part of your arsenal.

You can choose among three main types of wireless antennas:

_ **Omnidirectional:** Transmits and receives wireless signals 360 degrees over shorter distances, such as in boardrooms or reception areas. These antennas, also known as dipoles, typically come installed on APs from the factory.

_ **Semidirectional:** Transmits and receives directionally focused wireless signals over medium distances, such as down corridors and across one side of an office.

_ **Directional:** Transmits and receives highly focused wireless signals over long distances, such as between buildings. This antenna, also known as a high-gain antenna, is the antenna of choice for wireless hackers driving around cities looking for vulnerable APs — an act also known as *wardriving.*

Wireless LAN discovery :

1- Checking for worldwide recognition

The first test requires only the MAC address of your AP and access to the Internet.

You're testing to see if someone has discovered your WLAN and posted information

about it for the world to see. If you're not sure what your AP's MAC address is, you should be able to view it by using the arp -a command in DOS.

2- Scanning your local airwaves

Monitor the airwaves around your building to see what authorized and unauthorized APs you can find. You're looking for the SSID (service set identifier), which is your WLAN's name. If you have multiple WLANs, each one has a network SSID associated with it.

Here's where NetStumbler comes into play. NetStumbler can discover SSIDs and other detailed information about wireless APs, including the following:

_ MAC address
_ Name
_ Radio channel in use
_ Vendor name
_ Whether encryption is on or off
_ RF signal strength (signal-to-noise ratio)

Kismet — the popular wireless sniffer (network analyzer) for Linux and BSD UNIX — looks not only for probe responses from APs like NetStumbler does, but also for other 802.11 management packets, such as association responses and beacons. This allows Kismet to detect the presence of a WLAN even when probe-response packets are disabled in the AP — something that NetStumbler can't do.

Wireless Network attacks :

Various malicious hacks — including various DoS attacks — can be carried out against your WLAN. This includes APs that are forced to reveal their SSIDs during the process of being disassociated from the network and rejoining. In addition, hackers can literally jam the RF signal of an AP — especially in 802.11b and 802.11g systems — and force the wireless clients to reassociate to a rogue AP masquerading as the victim AP. Hackers can create man-in-themiddle attacks by maliciously using tools such as ESSID-jack and monkey-jack and can flood your network with thousands of packets per second by maliciously using packet-generation tools such as Gspoof or LANforge — enough to bring the network to its knees. Even more so than with wired networks, this type of DoS attack is practically impossible to prevent on WLANs.

When testing your WLAN security, look out for the following weaknesses:

_ Unencrypted wireless traffic
_ Unauthorized APs
_RF signals that are too strong
_ Wireless equipment that's easy to access physically
_ Default configuration settings

Encrypted Traffic :Wireless traffic can be captured directly out of the airwaves, making this communications medium susceptible to malicious eavesdropping. Unless the traffic is encrypted, it's sent and received in cleartext just like on a standard wired network. On top of that, the 802.11 encryption protocol, Wired Equivalent Privacy (WEP), has its own weakness

that allows hackers to crack the encryption keys and decrypt the captured traffic. WEP uses a fairly strong symmetric (sharedkey) encryption algorithm called RC4.

Countermeasures
The simplest solution to the WEP problem is to use a VPN for all wireless communications. The wireless industry has come up with a solution to the WEP problem called Wi-Fi Protected Access (WPA). WPA uses the Temporal Key Integrity Protocol (TKIP) encryption system, which fixes all the known WEP issues. WPA requires an 802.1x authentication server, such as a RADIUS server, to manage user accounts for the WLAN.

Rogue networks

Watch out for unauthorized APs and wireless clients attached to your network that are running in ad-hoc mode.
Using NetStumbler or your client manager software, you can test for APs that don't belong on your network. You can also use the network monitoring features in a WLAN analyzer such as AiroPeek.

Look for the following rogue AP characteristics:
_ Odd SSIDs, including the popular default ones *linksys*, *tsunami*, *comcomcom*, and *wireless.*
_ Odd AP system names — that is, the name of the AP if your hardware supports this feature — not to be confused with the SSID.
_ MAC addresses that don't belong on your network. Look at the first three bytes of the MAC address (the first six numbers), which specify the vendor name. You can perform a MAC-address vendor lookup at coffer.com/mac_find to find information on APs you're unsure of.
_ Weak radio signals, which can indicate that an AP has been hidden away or is on the outside of your building.
_ Communications across a different radio channel than what your network communicates on.

Countermeasures
The only way to detect rogue APs and hosts on your network is to monitor your WLAN proactively, looking for indicators that wireless clients or rogue APs might exist. But if rogue APs or clients don't show up in NetStumbler or in your client manager software, that doesn't mean you're off the hook. You may also need to break out the WLAN analyzer, wireless IDS, or other network management application.
You can enable MAC-address filtering controls on your AP so that wireless clients must have an authorized MAC address before being allowed to connect. The problem with this ountermeasure is that hackers can easily spoof MAC addresses in UNIX by using the ifconfig command and in Windows with the SMAC utility.

Physical-security problems
Various physical-security vulnerabilities can result in physical theft, the reconfiguration of wireless devices, and the capturing of confidential information. You should look for the following security vulnerabilities when testing your systems:
_ APs mounted on the outside of a building and accessible to the public.
_ Poorly mounted antennas — or the wrong types of antennas — that broadcast too strong a signal and that are accessible to the public.

You can view the signal strength in NetStumbler or your wireless client manager.

Countermeasures :
Secure APs, antennas, and other equipment in secure closets, ceilings, or other places that are difficult for a would-be intruder to access physically.
Terminate your APs outside any firewall or other network perimeter security devices — or at least in a DMZ — whenever possible.

If wireless signals are propagating outside your building where they don't belong, either
_ Turn down the transmit power setting of your AP.
_ Use a smaller or different antenna (semidirectional or directional) to decrease the signal.

1- Vulnerable wireless workstations

Wireless workstations have tons of security vulnerabilities — from weak passwords to unpatched security holes to the storage of WEP keys locally.
One serious vulnerability is for wireless clients using the Orinoco wireless card. The Orinoco Client Manager software stores encrypted WEP keys in the Windows Registry — even for multiple networks.
Countermeasures :
Regularly perform vulnerability assessments on your wireless workstations, as well as your other network hosts.
_ Apply the latest vendor security patches and enforce strong user passwords.
_ Use personal firewalls on these systems to keep malicious intruders off of those systems and out of your network.
_ Install antivirus software.
_ Consider installing an antispyware application such as PestPatrol

2 -Default Configuration Settings : Similar to wireless workstations, wireless APs have many known vulnerabilities.
The most common ones are default SSIDs and admin passwords. The more specific ones occur only on certain hardware and software versions that are posted in vulnerability databases and vendor Web sites.

Countermeasures
You can implement some of the simplest and effective security countermeasures for WLANs :
_ Make sure that you change default admin passwords, AP names, and SSIDs.
_ Disable SSID broadcasting if you don't need this feature.
_ Disable SNMP if you're not using it.
_ Apply the latest firmware patches for your APs and WLAN cards. This countermeasure helps to prevent various vulnerabilities, including the UDP broadcast exploit. If you find that it doesn't, consider using another vendor's wireless products.