Complete Basic Hacking Explained

Hacking Explained

//////////*1000+ HACKING TRICKS & TUTORIALS - ebook By Mukesh Bhardwaj Blogger - Paid Version - only @ TekGyd | itechhacks | Mukeshtricks4u*///////

This part will discuise some hacking techniques what is used in the field for some while. Hacking attacks progress in a series of stages, using various tools and techniques. A hacking attack consists of the following stages:

* Target Selection: A hacker identifies a specific computer to attack. To pass this stage, some vector of attack must be available.
* Target Identification: The hacker determines the characteristics of the target before actually engaging it.
* Attack Method Selection: The hacker selects one or more specific attacks to use against the target based on the information gathered in the previous stage.
* Attack Progression: The hacker proceeds with the actual attack or series of attacks

The hacker will attempt to find out more about your network through each successive attack, so the stages above actually feed back into the process as more information is gathered from failed attacks. The major techniques used to accomplish the phases of hacking include:

1. Eaves dropping and snooping
2. Denial-of-service
3. Protocol exploitation
4. Impersonation
5. Man-in-the-middle
6. Hijacking

Once you evaluate your network infrastructure and find weaknesses that a hacker can exploit, you can take measures to shore up your network's defenses.

Eavesdropping and Snooping
The first and easiest things a hacker can do to gain information about your network is simply to listen, and then to ask your network computers information about themselves. The hacker may not even contact your computers directly but instead communicate with other computers that provide services your computers rely on (Domain Name Service computers on the Internet, for example).Networked computers will volunteer a remarkable amount of information about themselves and how they are configured, especially if they are left in their default configurations as supplied by operating system vendors.

Hackers will attempt to exploit any data or network service that is exposed to them. Common hacking practices include (but are by no means limited to) the following activities:

* Password capture
* Traffic analysis
* Network address scanning
* Port scanning
* Finger, Whois, NSLookup, and DNS range grabbing
* SNMP data gathering

Password Capture

Most hacking activities place the hacker at some risk of being detected. One activity that does not pose this threat is eavesdropping on the local networking medium for logon information

Many networking protocols do not encrypt passwords, allowing any computer on the path between the client and the server to "overhear" the username and password. Not all encrypted logon procedures are safe from eavesdropping either, because (if the logon procedure is naive) a hacker can record the username and encrypted password to send to the server later in a "replay attack" or decrypt the password if the encryption algorithm is flawed or weak.

Eavesdropping requires software that will listen to all of the communications that flow over a network medium, such as Ethernet, rather than just listening to communications that are sent specifically to the hacker's computer. An eavesdropping hacker must also have access to a computer that is situated on a network link with network traffic flowing over it (such as a campus Ethernet or a computer in the server room of an Internet service provider). The more data that flows over the link, the more likely the hacker will capture passwords sent in the clear, i.e. in unencrypted form.

While the 802.11b wireless networking protocol broadcasts data in an easily compromised form, the protocol eliminates unnecessary broadcasts by transmitting only those packets directed to the MAC address of the specific wireless adapter involved in the communication. This means that the wireless access points that connect wireless devices to the wired network act like switches rather than hubs. For that reason, 802.11b adapters do not work in the full "promiscuous mode" required for true Ethernet "sniffing" or eavesdropping.

Physical location will not restrict the eavesdropping ability of a hacker who has penetrated other computers on the network. The hacker can install software on those computers that will allow them to snoop as well. The hacker may be typing at a computer in New York while a compromised computer in San Francisco records everything that goes over that remote network for the hacker's later perusal. A determined network intruder may even physically intrude on an otherwise secure LAN and connect a snooping device to the network cable. Casual hackers who are more interested in network joyriding or in finding a place to store their pirated software will seldom exhibit this level of effort (or brave this degree of risk), but other network intruders who might target your network for financial gain could easily do so if you don't take precautions.

Network eavesdropping is a technique hackers can use regardless of the technology used to implement the network. An IPX wide area network is just as vulnerable to someone eaves dropping on network connections as the Internet or an intranet that uses TCP/IP is.

In case you think it might be difficult to remotely install a network sniffer on someone else's network, consider that some versions of Windows Server operating systems include the Network Monitor, a very capable network monitor that can be remotely controlled and is rather easily exploited from afar. While it only records data flowing through the local server, data flowing through servers is typically what a hacker would be looking for.

Snooping Windows passwords over the Internet is surprisingly easy. Microsoft has built in a password Challenge/Response authentication mechanism into Internet Explorer to make secure Intranets easy to build. This mechanism allows a web server to challenge a client for that client's password. The client will respond with the account name of the logged-on user and that user'sone-way encrypted password. The password can be decrypted by comparing it to a list of pre-computed decrypted English words or through a brute-force key space comparison. At this point, a hacker has your account name and password, but you would have had to go to the hacker's website to compromise it.

That, too, is surprisingly easy to force. Imagine your boss sent you an e-mail with a link to a website embedded in it and a note saying, "Check this out they may be competition, what do you think?" Would you click the link? Voila!! you're compromised. Forging e-mail is so easy, it's child's play. A hacker can make his e-mail look like it's coming from anyone. All it takes to discover the correct names and e-mail addresses are

a few business cards.

E-mail from nowhere
Telnet to a mail server by opening a command prompt and typing telnet mailserver 25. Use the mail server configured in your e-mail program if you don't know of another one. Best: Telnet directly to the mail server of the recipient, if you can discover it. To discover it, use the NSLookup tool at the command prompt to look up the MX record associated with the recipient e-mail address's e-mail address.

Type the following at the Telnet prompts, pressing return after each listed line. You won't see text until you press return, and when you type the body text you won't see anything until you press a period by itself and hit enter. Replace the text "YOU@YOURSERVER.COM" with your own e-mailaddress.

MAIL FROM:
RCPT TO:
From:
Subject: Concerning your recent activities
Date: Mon, 1 Jan 2003 00:00:01 .
In case you were wondering, forging e-mail is unethical.

QUIT.
Now that you know how easy it is, you should take forged e-mail warnings very seriously.

Network Traffic Analysis
Passwords aren't the only things a determined hacker will listen for while eavesdropping on network traffic. Quite a bit of information about your network can be determined just from the nature of the traffic in and out of your network (or within your network if the hacker has compromised a computer within your security). Some things a hacker will look for include:

* The IP addresses of the source and destination computers of network traffic.
* The locations of gateways and routers.
* The amount of traffic originating from, being sent to, or flowing through computers identifiedby the hacker.
* Particular kinds of network traffic going to or from a computer that might identify thecomputer's function (DNS requests to one computer, or FTP responses from another, forexample).
* Network service availability broadcasts (such as NetBIOS browse list updates) that (from anexternal to a private network) indicate a network security hole or that (within a network) indicate targets for further attack.

The application proxy or Network Address Translation features of a firewall are the best tools for keeping traffic analysis from revealing too much about your network. The firewall will make all of the Internet (or other public network) traffic appears to come from one computer. A hacker from outside will not be able to determine the true extent of your network behind the firewall. You must also configure your firewall not to pass service availability broadcasts beyond your network boundary.

Network Address Scanning
Nearly all hacking attacks these days start with network address and port scanning. The hacker will specify a beginning and ending address to scan, and then the hacker's computer program will attempt to establish a connection to a computer on each of those network addresses in turn. If a computer answers from any one of those addresses the hacker has found another target.

All network technologies that specify an address of one kind or another for each computer on the network are vulnerable to this kind of attack. TCP/IP is the network technology most often scanned by hackers, and tools to scan TCP/IP are widely available. Other technologies such as NWLink,X.25, and FDDI are equally susceptible, if the hacker is willing to find or create the tools necessary to perform the scan.

The best way to foil this kind of attack is to watch for it. A network administrator who determines that this kind of attack is in progress can take steps to halt it, including configuring gateways or routers to discard network traffic from the offending host(s).

You need to configure gateways, packet filters, and routers to log connection requests to hosts that do not exist on your network. Periodically examine log data for network address scanning, and (if the logging software supports it) configure a network alert that will signal if a scan is in progress.

Port Scanning

Once a hacker has identified a target computer, the hacker will attempt to determine which operating system it is running and what services it is providing to network clients. On a TCP/IP-based network (such as the Internet), services are provided on numbered connections called sockets. The set of sockets to which a computer responds often identifies the operating system and supported services of the target computer

There are a number of tools available on the Internet that a hacker can use to determine which sockets are responding to network connection requests. These tools try each port in turn and report to the hacker which ports refuse connections and which do not. The hacker can then concentrate on ports corresponding to services that are often left unsecured or that have security problems.

Port scanning can reveal which operating system your computer is running because each OS has a different set of default services. For example, by scanning the TCP ports between 0 and 150, a hacker can discern Windows hosts by the presence of port 139 in the scan list, NT/2000/XP hosts by the presence of port 135 in the list, and various Unix hosts simply by the presence of TCP/IP services like port 23 (Telnet), which Windows computers do not install by default. This information tells the hacker which tools to use to further compromise your network.

The defense for port scanning is the same as for network address scanning watch for connection attempts to unsupported ports and then deny access to the computers that are doing the scanning. Periodically examine log data for port scanning, and (if the logging software supports it) configure a network alert that will signal if a scan is in progress. Consider setting up a server whose only purpose is to be exploited (a "honey pot") and then simply log every connection attempt to it. If you don't set a DNS name for it and don't advertise its existence, every connection attempt to it is the result of a hacking attempt.

Finger, Whois, NSLookup, and DNS Zone Transfer

There are a number of network services that hackers will use to gather information, if the ports used by those services are enabled on your Internet host. The Finger and Whois services are hacker favorites because they supply the account name and personal contact information for users of network computers. These are useful services for people who need to contact members of your organization or who need to find an e-mail address for a network user, but hackers will take usernames returned by these services and then attempt to break into those accounts by trying commonly used passwords.

By default, Windows does not support Finger or Whois. If you support Unix computers in your network, however, you should either disable these services or curtail the information they return. You can install software for Windows that provides the services, but you probably shouldn't.

Few network users will miss the Finger and Whois services, but the same cannot be said for the DNS service. The DNS service is required by Internet client software to convert human-friendly Internet names such as http://www.microsoft.com/ into computer-friendly IP addresses such as 10.1.1.2. Without the DNS service, the Internet would be useless for public services.

Windows servers do support the DNS service. Most networks that support the use of Internet tools within the network (instead of just the use of Internet tools to connect to services on the Internet) will include support for DNS. A smaller network can rely on an external DNS server to provide Internet name service translation for its clients, but a large IP network or an IP network behind a firewall is difficult to manage

without a DNS server of its own. Active directory requires DNS as well.

Hackers can use a DNS service to discover the structure of your network. Since DNS records the IP addresses and Internet names of all of the servers on your network, a hacker can attain a list of the most important computers in your network. The NSLookup tool is a standard Internet program for interrogating DNS servers, and a hacker can craft a program based on the NSLookup that would even make the hacker's computer appear to be a peer DNS server that needs information. Your task is to configure security in a way that allows clients from within to access the DNS server and get the information they need, but also prevents computers from outside your network from getting that information.

You can foil hackers attempting to gather information about the interior of your network by using different public and private DNS servers in your network. Establish internal DNS services for inside clients, and then set your internal DNS servers to forward to an ISP's DNS server for resolution of names not known to your interior DNS machines. Then block inbound DNS requests from the Internet to your inside domain name servers. To publish the names of public servers, either use your ISP's DNS servers or set up an additional pair of outside DNS servers in your DMZ that is used exclusively for public addresses.

The security problem is compounded by the fact that DNS is a hierarchical service. If one DNS server does not have the answer to a query, it will ask the next server up or down the DNS tree. This means that in a traditionally configured network, a DNS service within your firewall will need to be able to communicate with DNS servers outside the firewall. DNS servers are also configured to transfer blocks of Internet name and address data using a feature called Zone Transfer. In addition, many websites will not respond to Internet requests from client computers that don't have DNS reverse mappings, so the Internet servers that run those sites must be able to connect to your DNS server (via their DNS server or the DNS server up the tree from yours) to verify that the DNS reverse mapping exists.

A firewall can solve these problems by handling name translation inside your network. If your network requirements mandate that computers external to your network must be able to resolve IP addresses for computers inside your firewall or vice versa (if you use a software package that does not support use of a proxy server, for example), you should configure your firewall to disallow connections to your DNS server for all external computers except that of the DNS server up the tree from yours. You should also disable zone transfers for all DNS servers except those within your security domain.

SNMP Data Gathering

The Simple Network Management Protocol (SNMP) is an essential tool for managing large TCP/IP networks. SNMP allows the administrator to remotely query the status of and control the operation of network devices that support SNMP. Unfortunately, hackers can also use SNMP to gather data about a network or interfere with the operation of the network

Again, a firewall solves the problem. There's little reason why any computer outside your network should need to query SNMP, so simply block SNMP messages through your firewalls.

Denial of Service

The next easiest attack on your network is to disable some aspect of it or even bring the entire network down. The hacker may be merely interested in inconveniencing your organization, or they may have a more sinister purpose. In any case, you should remember that it is much easier for one computer to impersonate another computer if that other computer is disabled.

There are a number of methods a hacker can use to disable a computer or a service provided by a computer. Most of these methods affect computers using TCP/IP, because TCP/IP is the most widely used inter network protocol and because the most pressing hacker threat is from the Internet.Methods hackers can use to disable computers or computer services include these:

* Ping of Death (malformed ICMP packets)

* SYN (Synchronize Connection Establishments) Attacks and ICMP (Internet ControlMessage Protocol) flooding
* Service Specific Attacks
* DNS Redirection
* Route redirection: RIP (Router Information Protocol), BGP (Border Gateway Protocol), and ICMP
* SNMP reconfiguration

Ping of Death

Perhaps the most ominous sounding of Network layer attacks is the aptly named Ping of Death. A specially constructed ICMP packet that violates the construction rules can cause the recipient computer to crash if that computer's networking software does not check for invalid ICMP packets.

The only solution for computers outside your gateway (or the gateway computer itself) to resist the Ping of Death is to use a version of the operating system that is not susceptible to the Ping of Death. You can shield computers inside your network by not passing ICMP echo packets through your firewall, many new operation systems is protected now for this kind of attacks.

SYN Attacks and ICMP Flooding

Another way hackers disable the networking capability of computers is by overloading the network protocol software of the target computer with connection attempts or information requests. The initial IP packet of a TCP connection attempt is simple and easy to generate (a distinguishing characteristic of these packets is that they have the SYN bit set). Responding to a connection attempt takes more compute time and memory space than generating the packet does, because the receiving computer must record information about the new connection and allocate memory for connection data. An attacker can send one SYN packet after another to a target computer, and that target computer will then be unable to process other connection attempts from legitimate users because all of its available time and memory will be spent processing SYN requests.

A similar network protocol attack is ICMP flooding, in which the hacker sends a constant stream of ICMP echo requests to the target computer. The target computer then spends most of its time responding to the echo requests instead of processing legitimate network traffic.

Keep your firewall and operating system software updated to prevent against these attacks. You should configure your firewalls or servers to log instances of extremely frequent SYN connection attempts or an abnormally high volume of ICMP traffic in order to protect operating systems outside your firewall that may be vulnerable to these attacks.

Service Specific Denial of Service Attacks

Hackers are usually not interested in crashing your computer. The hacker may instead be more interested in shutting down one of the services supported by your network-connected computer.

Although any service provided by your computer may be the target of a service-specific attack ,there are four services that hackers are particularly attracted to, because they are either fundamental components of a TCP/IP network or fundamental components of Windows networking.The four services are RPC, NetBIOS, DNS, and WINS. Other services, such as Chargen or Time, do not provide a sufficiently rich environment for a hacker to have any real chance of using the service to break into or take down your computer.

Network clients connect to specific ports for each network service, and each service expects the network client to send the data to the service in a specific format. The DNS service, for example, expects that data sent to the DNS port from the client is formatted in a different manner than it is forWINS requests, and DNS will not be able to respond properly to WINS requests sent to it.

This is much like real-world services such as those provided by the Department of Motor Vehicles and the Social Security Administration, each of which needs different information from you in order o perform their services, and each of which has different forms for you to fill out. You could send a form requesting a duplicate Social Security card to the DMV, but you would neither get a Social Security card nor a driver's

license in return. You must send the right form to the right service.

While the repercussions of sending misleading or incorrect information to government institutions can be severe for the perpetrator, it will have negligible effects on the operation of the government service. However, sending incorrect or nonsense messages to a network service can crash the service, and it is difficult to track back to the hacker.

Many implementations of DNS, RPC, and WINS are particularly vulnerable to receiving random information at their ports. Some implementations of DNS also crash if they receive a DNS response without having first sent a DNS request. You can protect against unsolicited DNS responses by only allowing authorized external hosts to communicate with your DNS server.

The NetBIOS service of Windows is vulnerable to an out-of-band attack sent to the Net-BIOS ports. NetBIOS ports should not be accessible to computers outside your network at all, so the best solution to this problem (after installing the latest version of the operating system software) is not to bind NetBIOS to network adapters that can be reached from outside your network.

DNS Cache Pollution

An additional DNS service attack that deserves special mention is DNS cache pollution. A hacker can observe a computer that provides DNS services and determine the sequence used by the computer to provide query IDs for recursive DNS queries. The hacker can then forge a response to the next DNS query that contains invalid information or information that will redirect Internet traffic to a computer the hacker has already suborned. (The hacker may have to perform a denial-of-service attack on the DNS server being queried in order for the substitution to be accepted by the querying, targeted DNS server.)

This sort of attack can cause client computers that rely on the DNS server to not be able to resolve Internet names into valid IP addresses. That alone can cause problems on a TCP/IP network. More dangerous, however, is when a hacker populates the DNS server with valid IP addresses that are different from the correct IP addresses, especially if the hacker controls the computers at those addresses. A DNS cache pollution attack can therefore be the beginning of an impersonation attack on computers in your network.

Route Redirection (RIP, BGP, ICMP)

A hacker can cause a great deal of havoc in your network if the hacker can get control of your network's routers. Routers direct the flow of information within your network (as well as in and out of it), from information stored in their routing tables. By making changes to those routing tables, a hacker can isolate parts of your network and direct network traffic out of your network.

Routers must adapt to network conditions in order to maintain network functionality in the face of slowdowns or failures in network links. The routers in your network will exchange information about routing conditions, accept routing updates from network administrative programs, and communicate with routers outside your network if you allow them to. These routing updates are transmitted using a routing protocol, usually RIP, OSPF, or BGP.

RIP has no authentication capability. If a hacker can communicate with a router that uses RIP to update its network information, then the hacker can easily reconfigure the router to deny service to computers in your network or redirect the network traffic from computers in your network. OSPF provides more security than RIP does, and BGP is fairly secure about who it will communicate within order to update routing tables.

Another way a hacker can get your computers to send data to the wrong address is to send ICMP redirect packets to the computer. An ICMP redirect packet instructs the computer that an IP packet is being sent to the wrong router and that there is another route to the destination address that is either more efficient, faster, or capable of avoiding a network problem. It is difficult to forge ICMP packets, however, because they must appear to come from the router closest to the originating computer.

SNMP Reconfiguration

Many network devices, including Windows Server computers (if you install the SNMP service for them) can be managed remotely using SNMP. In addition to data snooping, a hacker can use SNMP to reconfigure your network to deny service to network computers or even to route data out of your network-depending on the SNMP features of the device the hacker gains control of

Protocol Exploitation

Protocol Exploitation is currently the most popular form of hacking on the Internet. Protocol exploitation is an attack based on exploiting a bug in a public service in order to gain more access than would normally be allowed.

Buffer overruns

The most common form of protocol exploitation is the venerable buffer overrun, which is an artifact of the way that modern compilers of certain programming languages create programs

In a C or C++ program, when a function allocates a local variable say, to copy data into that variable is placed on the program's stack (e.g., its temporary data region). Then, when the function calls another function, the subsequent function's return value is placed on the stack behind the local variable of the calling function.

By targeting services that run under "root" (Unix) or "Local System" (Windows) security contexts, the code is then free to perform nearly any task. Typically, the code would simply execute a call to open a security hole for further exploitation, such as patching the registry to allow the command console to receive commands from the web service, or passing a system call to download a fully exploitable Trojan horse.

Most programmers never consider the fact that code other than what they write might be executing inside their program. Why would they? Aside from someone actually maliciously changing their code while it executed, this sort of thing would never occur and therefore usually doesn't need to be checked for, so most programmers who learn to program in the safe context of a University computer lab don't get in the wasteful habit of checking everything. Many programmers also presume that the language or the compiler performs these checks for them, which is true for many languages, but it's not so in the case of C and C++, the languages used to write the vast majority of Internet services.

Impersonation

Impersonation is the next step for a hacker to take if the hacker still doesn't have access to your network computers. The goal of a hacker is to penetrate your network security and get at the information or resources on the computers in your network. These attacks are a lot harder and a lot rarer than protocol exploitation attacks. These attacks are used when a specific target is the goal, rather than when the desired outcome is the random mayhem caused by protocol exploits.

Merely snooping on your network traffic may give the hacker enough information to log on to your network. If that does not work, the hacker may reduce the functionality of your network via a denial-of-service attack, causing computers on your network to reveal enough information to allow the hacker to break in. The hacker might also pursue a denial-of-service attack just to inconvenience users of your network. Ultimately, you may not be susceptible to any well-known protocol exploitation attacks. By impersonating another computer that the computers on your network trust, the hacker's computer may be able to trick your computers into revealing enough information for the hacker to get through your network security. Alternatively, by impersonating another computer, the hacker's computer may be able to trick one of your computers into executing a command that weakens your security enough to let the hacker in. The tactics a hacker may use depend on the computer or service that the hacker is attempting to impersonate, which include the following:

* Source routed attacks
* DHCP, WINS, and DNS service impersonation
* Password playback, server impersonation, and password capture

Source Routed Attacks

The TCP/IP protocol suite includes a little-used option for specifying the exact route a packet should take as it crosses a TCP/IP-based network (such as the Internet). This option is called source routing, and it allows a hacker to send data from one computer and make it look like it came from another (usually more trusted) computer. Source routing is a useful tool for diagnosing network failures and circumventing network problems, but hackers too easily exploit it and so you should not use it in your TCP/IP network. Configure your firewalls to drop all source-routed TCP/IP packets from the Internet.

The hacker can use source routing to impersonate an already connected user and inject additional information into an otherwise benign communication between a server and the authorized client computer. For example, a hacker might detect that an administrator has telnetted onto a server from a client computer. If that administrator is at a command prompt, the hacker could inject a packet into the communications stream would appear to come from the administrator and would tell the server to execute the change password command thereby locking the administrator account and letting the hacker in.

The hacker also might use source routing to impersonate a trusted external DNS server and send DNS updates to your DNS server. This redirects all of the network clients that rely on the DNS server to translate Internet names into IP addresses, so that the client computers go instead to a hostile server under the control of the hacker. The hacker could then use the hostile server to capture passwords

DHCP, WINS, and DNS Service Impersonation

Another tactic a hacker can use to penetrate your network is to impersonate a service that your client computers get configuration information from at boot time. Network clients can be set up toget their configuration (including the location of the default gateway, DNS, and WINS servers) from a DHCP server, so a hacker who can impersonate a DHCP server can redirect your network clients to talk to almost any hostile host. By impersonating a WINS server, the hacker can return invalid or hostile IP addresses for NetBIOS computer names. By impersonating a DNS server, the hacker can return invalid or hostile IP addresses for Internet names as well.

In order for a hacker to impersonate a DHCP, WINS, or DNS server, the hacker must get control of one computer within your network and then initiate a denial-of-service attack against the legitimate DHCP, WINS, or DNS target computer. Once the target computer goes down, the computer controlled by the hacker can begin satisfying DHCP, WINS, or DNS requests in its place. This is just one way that a hacker can use one compromised computer in your network to penetrate your network security further and gain control of other computers in your network.

A DHCP, WINS, or DNS impersonation attack on your network relies on other attack methods to succeed. The hacker must first gather information about your network in order to identify targets, and then cause a denial of service on the service being impersonated. After that succeeds, the hacker must either gain control of at least one computer in your network that will be used to take the place of the server being impersonated, or redirect network traffic to an external computer that can take the place of the server being impersonated. The defensive measures you put in place to stop denial-of-service attacks and to restrict information about your network will help prevent an impersonation attack as well. You should also watch your network traffic for DHCP, WINS, or DNS services being hosted by unauthorized computers in your network, and you should take swift action to shut down any unauthorized servers.

Server Impersonation, Password Capture, and Password Playback

If the hacker has observed an encrypted logon session to one of your computers, they may not know the username and password being used to log on but might be able to fool your system anyway. The hacker might simply record the encrypted logon credentials and send those same credentials to your computer later. It won't matter that the hacker can't discern what the password is, because the receiving computer expects it in encrypted form anyway.

Older networking protocols are vulnerable to this sort of attack. This attack can be defeated by using challenge and response authentication for passwords or by "salting" the password with the current system time so that it can't be decrypted beyond a minute (or whatever the allotted time resolution is) from the original encryption time. Of course, that requires perfectly synchronized clocks, which requires the Network Time Protocol, which is subject to exploitation.

With challenge and response authentication, the password is never transmitted. Rather, the client indicates that she would like to log on. The server transmits a unique number to the client. Both computers encrypt that number using the client's password as a key. The client transmits the encrypted number back to the server. If the encrypted results match, then the same key was usedto perform the encryption and the server knows that the client knows the correct password. By encrypting a random number, the results will be different each time, making it impossible to derive the password used to encrypt it through mechanisms like snooping.

Windows and most modern versions of Unix use this sort of password encryption and authentication by default for NetBIOS connections from network client computers. Unfortunately, Windows also supports an older LAN Manager Authentication protocol. Networking clients may elect to use an older protocol if they inform a Windows server that they do not support the Windows NT Challenge/Response protocol. Hackers can exploit Windows support for this weaker protocol against newer computers by forging a response packet that appears to come from the server (using source routing, rerouting, or a man-in-the-middle position) and that instructs the modern client to use the weaker LAN Manager protocol. This way, the hacker can make the client use an easily cracked password encryption method or even instruct the client not to use password encryption at all. The hacker can then eavesdrop on the resulting logon traffic and capture the password used by the client to log on to the server. The best solution to this security problem is to configure your Windows computers not to accept LAN Manager Authentication.

Many older UNIX protocols, like Telnet, also don't make use of challenge and response authentication. Hackers can simply sniff these passwords off the network. E-mail clients like POP3 and IMAP, as well as Basic AUTH, transmit the user's password in plain text, so users who check e-mail accounts over the Internet are susceptible to having their passwords sniffed as well. If those passwords are the same as their network accounts, their account information is compromised.

Man-in-the-Middle

A special case of the impersonation attack is the man-in-the-middle attack, where the hacker operates between two computers on your network, or between a client computer on the Internet or other WAN network and your server computer in your secure LAN. When the client computer opens a connection to the server computer, the hacker's computer intercepts it (perhaps via a DNS or DHCP impersonation attack or by rerouting the IP traffic from the client to a compromised computer). The hacker computer opens a connection on behalf of the client computer to the server computer. Ideally (from the hacker's point of view), the client will think he is communicating with the server, the server will think it is communicating with the client, and the hacker computer in the middle will be able to observe and alter all of the communications between them.

Depending on the nature of the communications, the hacker computer may be able to use a man-in-the-middle attack to gain greater access to your network. For example, if the connection is an administrator-level Telnet session into a server computer from a client computer, the hacker computer in the middle could (after passing through the logon credentials to gain entry to the server) download the password file from the server to the hacker computer, instead of uploading HTML pages that the administrator may wish to place on the server computer.

On an insecure network such as the Internet, it is difficult to defend against a man-in-the-middle attack. Fortunately, a successful man-in-the-middle attack is also difficult to construct. The measures you take to protect your network against data gathering, denial-of-service, andimpersonation will help protect you from a man-in-the-middle attack. Nevertheless, you should never connect to your network using an administrative account over an insecure network.

Hijacking
One last hacker trick is the hijacking of an already established and authenticated networking connection. This can occur at two layers of the networking protocol at the TCP connection layer and at the SMB or NFS Session layer. In order for a hacker on the Internet to hijack a network share connection, the hacker will have to do both, because SMB uses TCP ports to make the connection.

In order to hijack an existing TCP connection, a hacker must be able to predict TCP sequence numbers, which the two communicating computers use to keep IP packets in order and to ensure that they all arrive at the destination. The hacker must also be able to redirect the TCP/IP connection to the hacker computer, and also launch a denial-of-service attack against the client computer so that the client computer does not indicate to the server that something is wrong. In order to hijack an SMB session (such as a drive mapping to a file share), the hacker must also be able to predict the correct NetBIOS Frame ID, Tree ID, and the correct user ID at the server level of an existing NetBIOS communications link

Wireless Scanning Wardriving / Warchalking
Wireless scanning is a method to find an available wireless network access point. It allows you to identify wireless networks through the use of WNIC (wireless network interface card) running in promiscuous mode and a software that will probe for access points. Once an open wireless access point is found, the wardriver usually maps it, so at the end he would have a map of access points with their properties (SSID, WEP, MAC etc.). Whenever the attacker wants to return into the network, he/she usually logs packets for later analysis, or to run them though a WEP key cracker when a weak key is being used. There are many different types of wireless scanning. The most known and used scanning method is Wardriving, next comes Warchalking. There are many other methods such as Warstrolling, Warflying etc., however this articles deals with Wardriving and Warchalking only.

Why War?
The term war, which is used in Wardriving, Warchalking etc., was taken from the old days of WarDialing. WarDialing, the hacking practice of phoning up every extension of a phone network until the number associated with a modem is hit upon, has been replaced by WarDriving with the introduction of wireless LANS.

WarDriving Lets take a drive
Wardriving is the first and well known method used to find available wireless networks (means unsecured). It is usually done with a mobile device such as a laptop or iPaq. Wardriving scanning is accomplished in an easy way: the attacker takes the device with him/her into a car, and detects networks (NetStumbler for Windows, BSD-AriTools for BSD, and airsnort for Linux). Once an open access point is detected, the attacker maps it, explores, or stumbles into a pipe to the internet.

The equipment necessary to WarDrive is: A wireless network interface card (PCMCIA), a device capable of locating itself on a map (GPS, not always necessary), a laptop or any other mobile device, Linux Red Hat or Debian (Windows is not recommended), Wireless tools (WEPCrack, AirSnort etc.)

The equipment is all off the shelf and pretty inexpensive.

WarChalking The hobo language

Now a new "language" is developing, WarChalking. The idea is based on the "hobo symbols" and is there to tell persons on the street where there is an open wireless network node, and what the settings are. It may look like incomprehensible squiggles, and most people would walk past thinking it is odd graffiti, but it conveys a lot of info that is understood by the hackers. Furthermore, it is now being adopted by those that are sharing networks voluntarily as a way to give the info out to the community." Zig

WarChalking was conceived by a group of friends in June 2002, and published by Matt Jones.

WarChalking is simply drawing a chalk symbol on a wall or pavement to indicate the presence of a wireless

network, so that other can easily notice it and the details about it. WarChalking is a the modern version of the hobo sign language, which was used by low-tech kings of the road to alert each other to shelter, food and potential trouble. The chalks symbols are nothing more than giving a visual cue to of a wireless network.

The following are the WarChalking symbols:

Symbol Key
SSID Open Node)(
Bandwidth
SSID Closed Node ()
WEP Node SSID Access Contact (W)
Bandwidth
Example for a WarChalking symbol:
Retina)(1.5

This symbol indicates a open node with SSID Retina and bandwidth equal to 1.5MBps.
With the use of these symbols, wardrivers can a lot about the node, and whether this is a worth network. Anyone initiated in the ways of WarChalking will recognize what it means, and get online.

Securing WLANs
Securing a wireless network is much simpler than securing a wired network. Building a secure wireless network can be done within few steps. So, you ask yourself why then its easy to break into a wireless network? the answer is very simple. Whenever a company wants to connect their employees wirelessly into the company network, the administrators often forget to change the default settings of a router, firewall, access point, enabling WEP and more.

Further more, far too many systems administrators forget that the wireless network extends beyond the walls of a building. There may be security guards at the door, and firewalls on the fixed cable network, but the wireless back door is wide open.

The Wireless network security issues are not discussed in this article. WLANs security issues were discussed in my previous article Wireless Security & Hacking.