
Closing Open Holes By Ankit Fadia Ankit@bol.net.in

With the spread of Hackers and Hacking incidents, the time has come, when not only system administrators of servers of big companies, but also people who connect to the Internet by dialing up into their ISP, have to worry about securing their system. It really does not make much difference whether you have a static IP or a dynamic one, if your system is connected to the Internet, then there is every chance of it being attacked.

This manual is aimed at discussing methods of system security analysis and will shed light on as to how to secure your standalone (also a system connected to a LAN) system.

Open Ports: A Threat to Security?

In the Netstat Tutorial we had discussed how the netstat -a command showed the list of open ports on your system. Well, anyhow, before I move on, I would like to quickly recap the important part. So here goes, straight from the netstat tutorial:

Now, the '–a' option is used to display all open connections on the local machine. It also returns the remote system to which we are connected to, the port numbers of the remote system we are connected to (and the local machine) and also the type and state of connection we have with the remote system.

For Example,

C:\windows>netstat -a

Active Connections

Proto	Local Address	Foreign Address	State
TCP	ankit:1031	dwarf.box.sk:ftp	ESTABLISHED
TCP	ankit:1036	dwarf.box.sk:ftp-data	TIME_WAIT
TCP	ankit:1043	banners.egroups.com:80	FIN_WAIT_2
TCP	ankit:1045	mail2.mtnl.net.in:pop3	TIME_WAIT
TCP	ankit:1052	zztop.boxnetwork.net:80	ESTABLISHED
TCP	ankit:1053	mail2.mtnl.net.in:pop3	TIME_WAIT
UDP	ankit:1025	*:*	
UDP	ankit:nbdatagram	*:*	

Now, let us take a single line from the above output and see what it stands for:

Proto	Local Address	Foreign Address	State
TCP	ankit:1031	dwarf.box.sk:ftp	ESTABLISHED

Now, the above can be arranged as below:

Protocol: TCP (This can be Transmission Control Protocol or TCP, User Datagram Protocol or UDP or sometimes even, IP or Internet Protocol.)

Local System Name: ankit (This is the name of the local system that you set during the Windows setup.)

Local Port opened and being used by this connection: 1031

Remote System: dwarf.box.sk (This is the non-numerical form of the system to which we are connected.)

Remote Port: ftp (This is the port number of the remote system dwarf.box.sk to which we are connected.)

State of Connection: ESTABLISHED

'Netstat' with the '–a' argument is normally used, to get a list of open ports on your own system i.e. on the local system. This can be particularly useful to check and see whether your system has a Trojan installed or not. Yes, most good Antiviral software are able to detect the presence of Trojans, but, we are hackers, and need to software to tell us, whether we are infected or not. Besides, it is more fun to do something manually than to simply click on the 'Scan' button and let some software do it.

The following is a list of Trojans and the port numbers which they use, if you Netstat yourself and find any of the following open, then you can be pretty sure, that you are infected.

Port 12345(TCP) Netbus

Port 31337(UDP) Back Orifice

For complete list, refer to the Tutorial on Trojans at: hackingtruths.box.sk/trojans.txt

---

Now, the above tutorial resulted in a number of people raising questions like: If the 'netstat -a' command shows open ports on my system, does this mean that anyone can connect to them? Or, How can I close these open ports? How do I know if an open port is a threat to my system's security of not? Well, the answer to all these question would be clear, once you read the below paragraph:

Now, the thing to understand here is that, Port numbers are divided into three ranges:

The Well Known Ports are those from 0 through 1023. This range or ports is bound to the services running on them. By this what I mean is that each port usually has a specific service running on it. You see there is an internationally accepted Port Numbers to Services rule, (refer RFC 1700 Here) which specifies as to on what port number a particular service runs. For Example, By Default or normally FTP runs on Port 21. So if you find that Port 21 is open on a particular system, then it usually means that that particular system uses the FTP Protocol to transfer files. However, please note that some smart system administrators delibrately i.e. to fool lamers run fake services on popular ports. For Example, a system might be running a fake FTP daemon on Port 21. Although you get the same interface like the FTP daemon banner, response numbers etc, however, it actually might be a software logging your prescence and sometimes even tracing you!!!

The Registered Ports are those from 1024 through 49151. This range of port numbers is not bound to any specific service. Actually, Networking utlites like your Browser, Email Client, FTP software opens a random port within this range and starts a communication with the remote server. A port number within this range is the reason why you are able to surf the net or check your email etc.

If you find that when you give the netstat -a command, then a number of ports within this range are open, then you should probably not worry. These ports are simply opened so that you can get your software applications to do what you want them to do. These ports are opened temporarily by various applications to perform tasks. They act as a buffer transfering packets (data) received to the application and vis-a-versa. Once you close the application, then you find that these ports are closed automatically. For Example, when you type www.hotmail.com in your browser, then your browser randomly chooses a Registered Port and uses it as a buffer to communicate with the various remote servers involved.

The Dynamic and/or Private Ports are those from 49152 through 65535. This range is rarely used, and is mostly used by trojans, however some application do tend to use such high range port numbers. For Example,Sun starts their RPC ports at 32768.

So this basically brings us to what to do if you find that Netstat gives you a couple of open ports on your system:

1. Check the Trojan Port List and check if the open port matches with any of the popular ones. If it does then get a trojan Removal and remove the trojan.

2. If it doesn't or if the Trojan Remover says: No trojan found, then see if the open port lies in the registered Ports range. If yes, then you have nothing to worry, so forget about it.

****
HACKING TRUTH: A common technique employed by a number of system administrators, is remapping ports. For example, normally the default port for HTTP is 80. However, the system administrator could also remap it to Port 8080. Now, if that is the case, then the homepage hosted at that server would be at:

http://domain.com:8080 instead of
http://domain.com:80

The idea behind Port Remapping is that instead of running a service on a well known port, where it can easily be exploited, it would be better to run it on a not so well known port, as the hacker, would find it more difficult to find that service. He would have to port scan high range of numbers to discover port remapping.

The ports used for remapping are usually pretty easy to remember. They are choosen keeping in mind the default port number at which the service being remapped should be running. For Example, POP by default runs on Port 110. However, if you were to remap it, you would choose any of the following: 1010, 11000, 1111 etc etc

Some sysadmins also like to choose Port numbers in the following manner: 1234,2345,3456,4567 and so on... Yet another reason as to why Port Remapping is done, is that on a Unix System to be able to listen to a port under 1024, you must have root previledges.

****

Firewalls

Use of Firewalls is no longer confined to servers or websites or commerical companies. Even if you simply dial up into your ISP or use PPP (Point to Point Protocol) to surf the net, you simply cannot do without a firewall. So what exactly is a firewall?

Well, in non-geek language, a firewall is basically a shield which protects your system from the untrusted non-reliable systems connected to the Internet. It is a software which listens to all ports on your system for any attempts to open a connection and when it detects such an attempt, then it reacts according to the predefined set of rules. So basically, a firewall is something that protects the network(or systen) from the Internet. It is derived from the concept of firewalls used in vehicles which is a barrier made of fire resistant material protecting the vehicle in case of fire.

Now, for a better 'according to the bible' defination of a firewall: A firewall is best described as a software or hardware or both Hardware and Software packet filter that

allows only selected packets to pass through from the Internet to your private internal network. A firewall is a system or a group of systems which guard a trusted network(The Internal Private Network from the untrusted network (The Internet.)

NOTE: This was a very brief desciption of what a firewall is, I would not be going into the details of their working in this manual.

Anyway,the term 'Firewalls', (which were generally used by companies for commerical purposes) has evolved into a new term called 'Personal Firewalls'. Now this term is basically used to refer to firewalls installed on a standalone system which may or may not be networked i.e. It usually connects to an ISP. Or in other words a personal firewall is a firewall used for personal use.

Now that you have a basic desciption as to what a firewall is, let us move on to why exactly you need to install a Firewall? Or, how can not installing a firewall pose a threat to the security of your system?

You see, when you are connected to the Internet, then you have millions of other untrusted systems connected to it as well. If somehow someone found out your IP address, then they could do probably anything to your system. They could exploit any vulnerability existing in your system, damage your data, and even use your system to hack into other computers.

Finding out someone'e IP Address is not very difficult. Anybody can find out your IP, through various Chat Services, Instant Messengers (ICQ, MSN, AOL etc), through a common ISP and numerous other ways. Infact finding out the IP Address of a specific person is not always the priority of some hackers.

What I mean to say by that is that there are a number of Scripts and utilities available which scan all IP addresses between a certain range for predefined common vulnerabilities. For Example, Systems with File Sharing Enabled or a system running an OS which is vulnerable to the Ping of Death attack etc etc As soon as a vulnerable system is found, then they use the IP to carry out the attacks.

The most common scanners look for systems with RAT's or Remote Administration Tools installed. They send a packet to common Trojan ports and display whether the victim's system has that Trojan installed or not. The 'Scan Range of IP Addresses' that these programs accept are quite wide and one can easily find a vulnerable system in the matter of minutes or even seconds.

Trojan Horses like Back Orifice provide remote access to your system and can set up a password sniffer. The combination of a back door and a sniffer is a dangerous one: The back door provides future remote access, while the sniffer may reveal important information about you like your other Passwords, Bank Details, Credit Card Numbers, Social Security Number etc If your home system is connected to a local LAN and the attacker manages to install a backdoor on it, then you probably have given the attacker

the same access level to your internal network, as you have. This wouls also mean that you will have created a back door into your network that bypasses any firewall that may be guarding the front door.

You may argue with me that as you are using a dial up link to your ISP via PPP, the attacker would be able to access your machine only when you are online. Well, yes that is true, however, not completely true. Yes, it does make access to your system when you reconnect, difficult, as you have a dynamic Internet Protocol Address. But, although this provides a faint hope of protection, routine scanning of the range of IP's in which your IP lies, will more often than not reveal your current Dynamic IP and the back door will provide access to your system.

****
HACKING TRUTH: Microsoft Says: War Dialer programs automatically scan for modems by trying every phone number within an exchange. If the modem can only be used for dial-out connections, a War Dialer won't discover it. However, PPP changes the equation, as it provides bidirectional transportmaking any connected system visible to scanners—and attackers.

****

So how do I protect myself from such Scans and unsolicited attacks? Well, this is where Personal Firewalls come in. They just like their name suggests, protect you from unsolicited connection probes, scans, attacks.

They listen to all ports for any connection requests received (from both legitimate and fake hosts) and sent (by applications like Browser, Email Client etc.) As soon as such an instance is recorded, it pops up a warning asking you what to do or whether to allow the connection to initiate or not. This warning message also contains the IP which is trying to initiate the connection and also the Port Number to which it is trying to connect i.e. the Port to which the packet was sent. It also protects your system from Port Scans, DOS Attacks, Vulnerability attacks etc. So basically it acts as a shield or a buffer which does not allow your system to communicate with the untrusted systems directly.

Most Personal Firewalls have extensive logging facilities which allows you to track down the attackers. Some popular firewalls are:

1.BlackICE Defender : An IDS for PC's. It's available at http://www.networkice.com.

2. ZoneAlarm: The easiest to setup and manage firewall. Get it for free at: www.zonelabs.com

Once you have installed a firewall on your system, you will often get a number of Warnings which might seem to be as if someone is trying to break into your system, however, they are actually bogus messages, which are caused by either your OS itself or

due to the process called Allocation of Dynamic IP's. For a details description of these two, read on.

Many people complain that as soon as they dial into their ISP, their firewall says that such and such IP is probing Port X. What causes them?
Well, this is quite common. The cause is that somebody hung up just before you dialed in and your ISP assigned you the same IP address. You are now seeing the remains of communication with the previous person. This is most common when the person to which the IP was assigned earlier was using ICQ or chat programs, was connected to a Game Server or simply turned off his modem before his communication with remote servers was complete.

You might even get a message like: Such and Such IP is trying to initiate a Netbios Session on Port X. This again is extremely common. The following is an explanation as to why it happens, which I picked up a couple of days ago: NetBIOS requests to UDP port 137 are the most common item you will see in your firewall reject logs. This comes about from a feature in Microsoft's Windows: when a program resolves an IP address into a name, it may send a NetBIOS query to IP address. This is part of the background radiation of the Internet, and is nothing to be concerned about.

What Causes them? On virtually all systems (UNIX, Macintosh, Windows), programs call the function 'gethostbyaddr()' with the desired address. This function will then do the appropriate lookup, and return the name. This function is part of the sockets API. The key thing to remember about gethostbyaddr() is that it is virtual. It doesn't specify how it resolves an address into a name. In practice, it will use all available mechanisms. If we look at UNIX, Windows, and Macintosh systems, we see the following techniques:

DNS in-addr.arpa PTR queries sent to the DNS server
NetBIOS NodeStatus queries sent to the IP address
lookups in the /etc/hosts file
AppleTalk over IP name query sent to the IP address
RPC query sent to the UNIX NIS server
NetBIOS lookup sent to the WINS server

Windows systems do the /etc/hosts, DNS, WINS, and NodeStatus techniques. In more excruciating detail, Microsoft has a generic system component called a naming service. All the protocol stacks in the system (NetBIOS, TCP/IP, Novel IPX, AppleTalk, Banyan, etc.) register the kinds of name resolutions they can perform. Some RPC products will likewise register an NIS naming service. When a program requests to resolve an address, this address gets passed onto the generic naming service. Windows will try each registered name resolution subsystem sequentially until it gets an answer.

(Side note: User's sometimes complained that accessing Windows servers is slow. This is caused by installing unneeded protocol stacks that must timeout first before the real protocol stack is queried for the server name.).

The order in which it performs these resolution steps for IP addresses can be configured under the Windows registry key

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\ServiceProvider.

Breaking Through Firewalls

Although Firewalls are meant to provide your complete protection from Port Scan probes etc there are several holes existing in popular firewalls, waiting to be exploited. In this issue, I will discuss a hole in ZoneAlarm Version 2.1.10 to 2.0.26, which allows the attacker to port scan the target system (Although normally it should stop such scans.)

If one uses port 67 as the source port of a TCP or UDP scan, ZoneAlarm will let the packet through and will not notify the user. This means, that one can TCP or UDP port scan a ZoneAlarm protected computer as if there were no firewall there IF one uses port 67 as the source port on the packets.

Exploit:
UDP Scan:
You can use NMap to port scan the host with the following command line:

nmap -g67 -P0 -p130-140 -sU 192.168.128.88

(Notice the -g67 which specifies source port).

TCP Scan:
You can use NMap to port scan the host with the following command line:

nmap -g67 -P0 -p130-140 -sS 192.168.128.88

(Notice the -g67 which specifies source port).

Well, that is all for this manual, which is by no means finished. I would be updating it at regular intervals, so kindly hang on. Bye...

Ankit Fadia

Ankit@bol.net.in

http://www.ankitfadia.com

To receive tutorials written by Ankit Fadia on everything you ever dreamt of in your Inbox, join his mailing list by sending a blank email to: programmingforhackers-subscribe@egroups.com

Wanna ask a question? Got a comment to make? Criticize, Comment and more…..by sending me an Instant Message on MSN Messenger. The ID that I use is: ankit_fadia@hotmail.com

Wanna learn Hacking? Wanna attend monthly lectures and discussions on various Networking/Hacking topics? Lectures, Debates and Discussions, get it all by simply joining The Hacking Truths club by clicking Here