
Gathering Info on Remote Host: Essential Ingredient of Hacking into it. By Ankit Fadia ankit@bol.net.in

---

I get a lot of emails from people asking me how they can break into their ISP or how they can break into a system etc etc. Infact, such questions are almost the most common ones, from all the questions I get. Well, after this popular demand, I thought that an entire manual on breaking into systems was needed. So here goes..

You see, breaking into systems or getting root on a system is not as difficult as it seems. And it by no means requires you to be an Uberhacker. Getting into a system is quite easy and it requires you to know at least one programming language (preferably C), and have a more than an average IQ. However, breaking into systems does require a bit of luck and also a bit of carelessness or stupidity on the part of the system administrator of the target system.

What I mean to say by all this is that, breaking into systems is no big deal, anyone could do that, even a script kiddie, however, the part of the entire Hacking process where more than most people falter is the remaining undetected part. Anonymity or remaining anonymous to the Server logs and preventing detection of a break-in is the most difficult part of Hacking into a system.

What separates a good Hacker from a Script Kiddie or a Lamer is that the former has more than several ways of making sure that no one even suspects that there has been a break in, while on the other hand, the later has no clue what so ever as to what he is doing or what he needs to do to prevent such detection. There are so many ready to Use canned C programs or Hacking utilities available on the net, that a huge number of wannabe hackers, download them and use them to Hack into systems. Well, not only do they do not work properly and flawlessly, they also provide no mechanism of remaining anonymous. What is more, say if you are not using a canned Hacking tool, and are also not trying to remain anonymous, then you stand a greater chance of remaining undetected than if you were using such a tool. So think before you use such tools, you might be able

to get the Password file and become very kewl, however, you will certainly be caught later if not sooner.

The first step that you need to take once you have decided the target computer is to find out as much information as you can about it. You see, to break into a system you need to exploit a vulnerability existing in the services offered by it. Almost all systems have certain open ports, which have certain daemons or services running on them.

****

HACKING TRUTH: There are two types of ports. There are hardware ports, which are the slots existing behind the CPU cabinet of your system, into which you plug-in or connect your hardware to. For Example, COM1, COM2, Parallel Port etc. However, we are not interested in such ports. We are concerned with the other type of ports, which are the virtual or the software ports. Such a virtual port is basically a virtual pipe through which information goes in and out. And all open ports have a service or daemon running on it. A service or a daemon is nothing but the software running on these ports, which provide a certain service to the users who connect to it. For Example, Port 25 is always open on a server handling mails, as it is port where the Sendmail service is running by default.

****

So basically the first step in your quest to breaking into a system is to get as much information on it, as you can. Try to get, the list of open ports, the list of services running on the respective open ports and whole lots of other kind of information to which I will come later.

Anyway, so firstly, get a good Port Scanner, preferably stealth and then do a port scan on the target host. Now one thing that you must remember while doing a port scan is the fact that there are various so called 'stealth' port scanners around which claim to be undetectable, however most of them are detectable. So instead of using such' false claims' port scanners, I suggest you code one on your own.

But why do I need to use a stealth Port Scanner and how can I code my own Port Scanner? Well, the reason as to why you need a stealth port Scanner is that many system

administrators log all port scans and records the IP and other information on such attempts, this makes you susceptible to getting caught. In my opinion the best Port Scanners around are those, which send SYN/FIN packets from a spoofed host, making logging useless. Such a port Scanner would be coded in C, but will not run in Windows. This was just an idea, now it is up to you to code it yourself.

Anyway, let me assume that you have got hold of a good 'impossible to detect' Port Scanner, now scan the target system for all open ports and record the open lists:

Note: In this manual, I have taken up my ISP as an example target system. It would be foo-barred throughout as xxx.bol.net.in

In my case, I found that the following ports were open:

Port Number	Service
21	FTP
23	Telnet
25	SMTP
53	DNS
79	Finger
80	HTTP
110	POP
111	Not Useful

System running and also the FTP daemon running. Well, actually it is the login prompt of the daemon banner which gives us the Operating System running on it. Normally, a typical daemon banner, would have the following Login prompt:

220 xxx2.bol.net.in FTP server (Digital UNIX Version 5.60) ready.

User (bol.net.in:(none)):

Notice the System name in the brackets on the first line. However, normally almost all FTP daemons are better configured (that is the case in the example target system: xxx.bol.net.in)and their login prompt is somewhat like the below:

220 ftp2.xxx.bol.net.in FTP server ready.

User (mail2.bol.net.in:(none)):

See, no Operating System name. However, with the help of some kewl commands, such systems too can be reveal the OS running on them. However, before we go on, there is one thing that you have to be clear about. Now, we had FTP'ed to xxx.bol.net.in, so you normally expect to connect to Port 21 of xxx.bol.net.in, however that is not true. (Atleast in this case.) If you look at the daemon banner again, then you would notice that the last line says:

220 ftp2.xxx.bol.net.in FTP server ready.

Now how did that happen? Well, is Port 21 not open on xxx.bol.net.in ? Well, no and yes. What actually happens is that, Port 21 of xxx.bol.net.in is open and a daemon there is listening for connections. As soon as a connection is established, it transfers the control or connected the visitor to ftp2.xxx.bol.net.in, which is on the same network as xxx.bol.net.in. Now this, ftp.xxx.bol.net.in system is solely a FTP machine. It has no other services running. So whatever information, we gather from such a FTP port is not of xxx.bol.net.in but of ftp2.bol.net.in. Get it?

Anyway, when you get the login prompt, then login anonymously with the anonymous as the Username and a false email address as the password.

220 ftp2.xxx.bol.net.in FTP server ready.

User (ftp2.xxx.bol.net.in:(none)): anonymous

331 Guest login ok, send your complete e-mail address as password.

Password: xxx@linux.net

230 User anonymous logged in. Access restrictions apply.

Even if you have an account at the FTP server into which you plan to break in, it is always better not to use that pair of Username and Password. Logging in anonymously has many advantages. Say if you did cause some harm to the target system and if you use your (Nonanonymous) Username and Password pair, then if you were not able to edit the server logs you could get into some serious trouble. [Well actually not much, only say your account might be disabled. However, it could be worse.]

Ok, you are in, now let us get the FTP client to tell us which commands are available by typing the help command.

ftp> help

Commands may be abbreviated. Commands are:

! delete literal prompt

? debug ls put

append dir mdelete pwd

asc
O CWD STAT XRMD SIZE

REIN* MODE MSND* REST XCWD HELP PWD MDTM

QUIT RETR MSOM* RNFR LIST NOOP XPWD I mean by that is that all remote FTP commands have to be preceded by the word 'literal'. For example, say you want to execute the remote FTP command: 'stat', then you would type:

ftp> literal stat

****
HACKING TRUTH: According to FTP help, the literal command is described as:

ftp> help literal

literal send arbitrary ftp command

****

Anyway, amongst the remote FTP commands, the commands of interest to us are-: 'stat' and 'syst'. Let us see what they return when executed-:

ftp>literal stat

211- ftp2.xxx.bol.net.in FTP server status:

Version 5.60

Connected to 203.xx.251.198 (203.xx.251.198)

Logged in anonymously

TYPE: ASCII, FORM: Nonprint; STRUcture: File; transfer MODE: Stream

211- No data connection

211 End of status

Note: The IP address is of xxx.bol.net.in and not your machine.

ftp> literal syst

215 UNIX Type: L8 Version: BSD-198911

Voila, we get the Operating System name running on ftp2.xxx.bol.net.in. At last some useful information.

Finger and HTTP both failed, what do we do now? Let us turn to the den of the Buggiest daemon on Earth i.e. Sendmail: Port 25, the SMTP port.

Sendmail is certainly the buggiest daemon on earth; it has the highest number of known exploits amongst all the daemons. So this probably should get us through. Let us telnet to Port 25 and find out whether an exploitable version of Sendmail is running.

C:\windows> telnet xxx.bol.net.in 25

220 xxx.bol.net.in ESMTP Sendmail 8.9.1 (1.1.20.3/27Jun00-0346PM) Thu, 29 Jun 2000 14:18:12 0530 (IST)

When you telnet to Port 25, then the first thing that you come across would be a something like the above welcome daemon banner. A daemon banner is a Hacker's best friend. It reveals important information about the host, which proves to be invaluable in breaking into it. It basically tells you which daemon or service is running on that port and also the version of that particular service. Like for example, in this case, the Sendmail daemon banner tells us that ESMTP Sendmail 8.9.1 is running and it also gives us other information about the host at which this service is running.

Anyway, getting back to the topic, this banner reveals a big vulnerability existing in the host computer. It tells us that xxx.bol.net.in is running an old, vulnerable version of

Sendmail. The latest version is Sendmail 8.9.4 (correct me if I am wrong.), so this particular version of Sendmail wouldn't be without any bugs.

So then what you do is visit PacketStorm or search at your favorite Hacking stuff related search engine for a C program which demonstrates how to exploit version 8.9.4 of Sendmail. Now, all this might sound a bit too simple, well it certainly isn't, read on for more info.

Now, there are a couple of things that you need to keep in mind while getting this done. Say, you have found out that the victim runs Sendmail 8.9.4, now you cannot simply break in by running any exploit for this version. By that what, I mean to say is that, an exploit, which is coded to be executed on a Linux platform, will not work if you try to compile and run it on a Windows platform. So basically before you execute the 'kewl' exploit program that you downloaded, you should find out which platform it is meant for and if you are not running that platform, then you will need to get your gray cells working.

This is the stage where real hackers are differentiated between script kiddies, this is when those people who really know something prevail. Normally say if a exploit is designed to work on Linux, then if you edit its code and change its header files (if necessary), then that particular exploit can be made to run on Windows too. However, there are certain exploits, which simply would not run on a different OS than it is designed too.

Anyway, let us get back to point. You have edited the exploit code and made it compatible with your platform. Now what else? Another thing that you want to keep in mind is the Operating System, which the exploit can exploit. You see, there are certain exploits, which work only if the victim system is running a specific Operating System. For Example,

There was once a Sendmail hole, which worked only if the target System was running Sun OS without which, it simply refused to even work.

So in some cases it becomes necessary, to find out the Operating System running at the target system. Although not all exploits require the target system to be running a specific system, but why take a chance. Right?

So basically you should be aware of the following things while getting a ready to use exploit-:

1.) The Daemon name and version you are trying to exploit For Example, Sendmail 8.9.4

2.) The Operating System at which it is designed to run. (If necessary)

3.) The operating System it requires the target system to be running. (If necessary)

That brings us to as to how to find out the Operating System running at the target system? Well, the HTTP port holds the key. Simply, telnet to Port 80 of the target system.

C:\windows>telnet xxx.bol.net.in 80

Now, once you get the input prompt, then, type an invalid HTTP command. For Example, X or Iamgreat or abc etc. Just type anything as long as it is not a valid HTTP command. Then press enter twice.

****

Hacking Truth: After each HTTP command one has to press Enter Twice to send the command to the server or to bring about a response from a server. It is just how the HTTP protocol works.

****

On Port 80 of my example target system, I type simply 'ankit' and press enter twice. This is the kind of response I get:

HTTP/1.1 400 Bad Request

Server: Netscape-Enterprise/3.5.1

The server replies with the version of HTTP it is running (not so important), it gives us an error message and the error code associated with it(again not so important), but it also gives us the OS name and OS version, it is running. Wow!!! It gives hackers who want to break into their server the ultimate piece of information, which they require.

Well, these were the common ways of finding out more information about a host in your quest to break into it. I will soon be updating this manual, hope you enjoyed the first edition. Till the next update, goodbye.

COMING SOON: Finding out more Information about the remote host.

Exploiting the R Services (rlogin etc) or Exploiting Trust Relationships

Exploiting Routers

More Fun with Remote Hosts

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