
Getting geographical Information using an IP Address By Ankit Fadia ankit@bol.net.in

---

Getting the Internet Protocol or the IP Address of a remote system is said to the most important step in hacking of a system. Sometimes, however we get an IP in order to get more information on someone or some host. But, how can an IP Address be used to get more information on the location etc of a system? Well, this manual is aimed at answering just this question.

Actually, the IP address (Actually the entire TCP/IP Protocol) is structured or designed such that one cannot tell as to in which country a system having the given IP is situated, by simply looking at it. An IP Address has no fields, which tell you the country in which the computer using it resides in. So, all myths like 'The Second or the third field of an IP stands for the country in which the system using it resides' are definitely false and untrue.

However, yes sometimes one can guess or deduce as to in which country and even in which city the system using an IP resides in, by simply looking at the first three fields of the IP. Let us take an example to understand what I mean to say by this. Now, before I move on the example, let us understand how exactly IP Addresses are awarded to you.

Firstly, your ISP registers at the central authority and gets a particular range of IP addresses between which the various customers (people who dial into their servers) can be awarded IP addresses. Most ISP's are given a Class C network Address. A class C Network address contains a 24-bit Network Prefix (the first three fields) and an 8-bit Host number (the last field). It is referred to as "24's" and is commonly used by most ISP's.

****
HACKING TRUTH: For the benefit of beginners, I have included below a snippet from one of my earlier manuals, which explains IP Addresses better: (Even if you are not a newbie, I do suggest you read the below snippet, as it might just be helpful.)

Like in the real world, everyone has got an individual Home Address or telephone number so that, that particular individual can be contacted on that number or address, similarly all computers connected to the Internet are given a unique Internet Protocol or IP address which can be used to contact that particular computer. In geek language an IP address would be a decimal notation that divides the 32- bit Internet addresses (IP) into four 8-bit fields.

Does the IP address give me some information or do the numbers stand for anything?

Let take the example of the following IP address: 202.144.49.110 Now the first part, the numbers before the first decimal i.e. 209 is the Network number or the Network Prefix.. This means that it identifies the number of the network in which the host is. The second part i.e. 144 is the Host Number that is it identifies the number of the host within the Network. This means that in the same Network, the network number is same. In order to provide flexibility in the size of the Network, here are different classes of IP addresses:

Address Class	Dotted Decimal Notation Ranges
Class A (/8 Prefixes)	1.xxx.xxx.xxx through 126.xxx.xxx.xxx
Class B (/16 Prefixes)	128.0.xxx.xxx through 191.255.xxx.xxx
Class C (/24 Prefixes)	192.0.0.xxx through 223.255.255.xxx

The various classes will be clearer after reading the next few lines.

Each Class A Network Address contains a 8 bit Network Prefix followed by a 24-bit host number. They are considered to be primitive. They are referred to as "/8"s" or just "8's" as they have an 8-bit Network prefix.
In a Class B Network Address there is a 16 bit Network Prefix followed by a 16-bit Host number. It is referred to as "16's".

A class C Network address contains a 24-bit Network Prefix and a 8 bit Host number. It is referred to as "24's" and is commonly used by most ISP's.

Due to the growing size of the Internet the Network Administrators faced many problems. The Internet routing tables were beginning to grow and now the administrators had to request another network number from the Internet before a new network could be installed at their site. This is where sub-netting came in.

Now if your ISP is a big one and if it provides you with dynamic IP addresses then you will most probably see that whenever you log on to the net, your IP address will have the same first 24 bits and only the last 8 bits will keep changing. This is due to the fact that when sub-netting comes in then the IP Addresses structure becomes:

xxx.xxx.zzz.yyy

where the first 2 parts are Network Prefix numbers and the zzz is the Subnet number and the yyy is the host number. So you are always connected to the same Subnet within the same Network. As a result the first 3 parts will remain the same and only the last part i.e. yyy is variable.
****

For Example, if say an ISP xyz is given the IP: 203.98.12.xx Network address then you can be awarded any IP, whose first three fields are 203.98.12. Get it?

So, basically this means that each ISP has a particular range in which to allocate all its subscribers. Or in other words, all subscribers or all people connected to the internet using the same ISP, will have to be in this range. This in effect would mean that all people using the same ISP are likely to have the same first three fields of their IP Addresses.

This means that if you have done a lot of (By this I really mean a lot) of research, then you could figure out which ISP a person is using by simply looking at his IP. The ISP name could then be used to figure out the city and the country of the person. Right? Let me take an example to stress as to how cumbersome but easy (once the research is done) the above method can be.

In my country, say there are three main ISP's:

ISP Name	Network Address Allotted
ISP I	203.94.47.xx
ISP II	202.92.12.xx
ISP III	203.91.35.xx

Now, if I get to know the IP of an e-pal of mine, and it reads: 203.91.35.12, then I can pretty easily figure out that he uses ISP III to connect to the internet. Right? You might say that any idiot would be able to do this. Well, yes and no. You see, the above method of finding out the ISP of a person was successful only because we already had the ISP

and Network Address Allotted list with us. So, what my point is, that the above method can be successful only after a lot of research and experimentation. And, I do think such research can be helpful sometimes.

Also, this would not work, if you take it all on in larger scale. What if the IP that you have belongs to someone living in a remote igloo in the North Pole? You could not possibly get the Network Addresses of all the ISP's in the world, could you?

NOTE: In the above case, you also get to know the city of the system using the given IP, as most ISP's use different network addresses in different cities. Also, some ISP's are operational in a single city.

So, is there a better method of getting the location of an IP? Yes, Reverse DNS lookups hold the key.

Just as DNS lookup converts the hostname into IP address, a Reverse DNS Lookup converts the IP address of a host to the hostname. By hostname, what I mean to say is that it given us the name of the remote system in alphabets and numbers and periods. For Example, mail2.bol.net.in would be a hostname, while 203.45.67.98 would not be a hostname.

The popular and wonderful Unix utility 'nslookup' can be used for performing Reverse DNS lookups.
So, if you using a *nix box or if you have access to a shell account, then the first this to do is to locate where the nslookup command is hidden by issuing the following command:

' whereis nslookup '.

Once you locate where the utility is hidden, you could easily use it to perform both normal and reverse DNS lookups. As this is not a manual on using the 'nslookup' command, I will simply giving a basic relevant outline. In order to get a more detailed description of how this works or how to use it, read the *nix man pages or the documentation.

We can use 'nslookup' to perform a reverse DNS lookup by mentioning the IP of the host at the prompt.
For Example,

$>nslookup IP Address

Note: The below IP's and corresponding hostnames have been made up. They may not actually exist.

Let us say, that above, instead of IP Address, we type 203.94.12.01 (which would be the IP I want to trace.).

$>nslookup 203.94.12.01

Then, you would receive a response similar to: mail2.bol.net.in

Now, if you carefully look at the hostname that the Reverse DNS lookup, gave us, then the last part reveals the country in which system resides in. You see, the '.in' part signifies that the system is located in India. All countries have been allotted country codes, which more often than not are the last part of the hostnames of the systems located in that country. This method can also be used to figure out as to which country a person lives in, if you know his email address. For Example, if a person has an email address ending in .ph then he probably lives in Philippines and if it ends in .il then he lives in Israel and so on. Some common country codes are:

Country	Code

Australia	.au
Indonesia	.id
India	.in
Japan	.jp
Israel	.il
Britain	.uk

For a complete list of country codes, visit:

http://www.alldomains.com/
http://www.iana.org/domain-names.html

****
General Extra Tip: To get the complete list of US State Abbreviation codes, visit:
http://www.usps.gov/ncsc/lookups/abbr_state.txt
****
Windows users can perform Reverse DNS queries by downloading an utility called Samspade from: http://www.samspade.com/

Another method of getting the exact geographical location of a system on the globe is by making use of the WHOIS database. The WHOIS database is basically the main database, which contains a variety of information like contact details, name etc on the person who owns a particular domain name. So, basically what one does in a WHOIS query, is supply the WHOIS service with the hostname on which he wants more information. The WHOIS service then replies with the information stored in its database.

This method can be used to get some pretty accurate information on a particular IP or hostname; however, it is probably of no use if you are trying to point out the exact location of a dynamic IP. But, again this can be used to get atleast the city in which the ISP used by the victim is situated.

You can carry out WHOIS queries at: http://www.alldomains.com/

You could also directly enter the following in the location bar of your Browser and perform a WHOIS enquiry.

Enter the following in the location bar of your browser:

http://205.177.25.9/cgi-bin/whois?abc.com

Note: Replace abc.com with the domain name on which you want to perform a WHOIS query.

This method cannot be used to get the contact address of a person, if the IP that you use to trace him, belongs to his ISP. So, either you need to know the domain name (which is registered on his name) or have to remain satisfied knowing only the city (and ISP) used by the person.

Say, the victim has registered a domain name and you want to use it to find out the city in which he resides. Now, one thing to remember in this case is that, if the victim has registered the domain name using any of the various free .com registration services like Namezero.com etc, then the domain name would probably be registered on the

company's name and not the victim's name. So, a WHOIS query will give information on the ISP and not the victim.

****
NEWBIE NOTE: The WHOIS service by default runs on Port 43 of a system. Try performing a WHOIS query by telnetting to Port 43 and manually typing out the query. I have never tried it, however, it might be fun.
****

Yet another and probably the second most efficient method (after Reverse DNS queries) of tracing an IP to its exact geographical location, is to carry out a 'traceroute' on it. The 'tracert' or 'traceroute' commands give you the names or IP's of the routers through which it passes, before reaching the destination. Windows users can perform a trace of an IP, by typing the following at the command line prompt:

C:\windows>tracert IP or Hostname

For more information about the usage and syntax of this command, type: 'tracert' at the command prompt. Anyway, now let us see what is the result, when I do a tracert on my IP. Remember I live in New Delhi which is a city in India. Watch the names of the hostnames closely, as you will find that they reveal the cities through which the packet passes.

C:\windows>tracert 203.94.12.54

Tracing route to 203.94.12.54 over a maximum of 30 hops

1 abc.netzero.com (232.61.41.251) 2 ms 1 ms 1 ms
2 xyz.Netzero.com (232.61.41.0) 5 ms 5 ms 5 ms
3 232.61.41.10 (232.61.41.251) 9 ms 11 ms 13 ms
4 we21.spectranet.com (196.01.83.12) 535 ms 549 ms 513 ms
5 isp.net.ny (196.23.0.0) 562 ms 596 ms 600 ms
6 196.23.0.25 (196.23.0.25) 1195 ms1204 ms
7 backbone.isp.ny (198.87.12.11) 1208 ms1216 ms1233 ms
8 asianet.com (202.12.32.10) 1210 ms1239 ms1211 ms
9 south.asinet.com (202.10.10.10) 1069 ms1087 ms1122 ms
10 backbone.vsnl.net.in (203.98.46.01) 1064 ms1109 ms1061 ms
11 newdelhi-01.backbone.vsnl.net.in (203.102.46.01) 1185 ms1146 ms1203 ms
12 newdelhi-00.backbone.vsnl.net.in (203.102.46.02) ms1159 ms1073 ms
13 mtnl.net.in (203.194.56.00) 1052 ms 642 ms 658 ms

So, the above shows us that the route taken by a data to reach the supplied IP is somewhat like this:

Netzero (ISP from which the data is sent) --- à Spectranet (A Backbone Provider) ----- à New York ISP --- à New York Backbone - à Asia -- à South Asia - à India Backbone -- à New Delhi Backbone -- à Another router in New Delhi Backbone --- à New Delhi ISP.

So, basically this tracert does reveal my real location, which is: New Delhi, India, South Asia. Get it?

Sometimes, doing a 'tracert' on an IP, does not give useful information. You see in the above example, the hostnames returned revealed the city or country in which the system is located. Although, more often than not, you

will get such helpful hostnames, sometimes the hostnames returned are very vague and unhelpful.
So what do you do then? Well, fret not. Simply do the below procedure.

Let us say that the trace ends at the hostname abc.com. This is very vague and gives absolutely no clue as to where the system is located. However, what you could do is, launch your browser and visit: http://www.abc.com/ Now, abc.com is probably an ISP and an ISP, will definitely give its location and the cities in which it operates. So, you could still have a good chance of learning the definite city of the victim.

A very interesting utility is the VisualRoute utility, (http://www.visualroute.com/) which traces a hostname or IP and shows the path taken by the packet to reach the destination on a world map. It is very useful and reveals some excellent information. However, it sometimes does tend to be inaccurate.

****
HACKING TRUTH: Say you have found out the ISP of a person and simply want to learn as to in which country the person resides in. However, visiting the ISP website doesn't help. Nor does the hostname help. So, what do you do? Well, one thing that you could do is, try connecting to Port 13 of the ISP. This is the port, which simply displays the system time. It will tell you how many hours ahead or behind the system is from GMT time.
****

Well, this basically brings us to the end of this manual. Before I sign off, I would like to make it clear that it extremely difficult and surprising if someone is able to get the exact contact address of a person by simply knowing his IP. (Without taking help or breaking into the person's ISP) Anyway, hope you liked this manual. Goodbye.

Ankit Fadia
ankit@bol.net.in