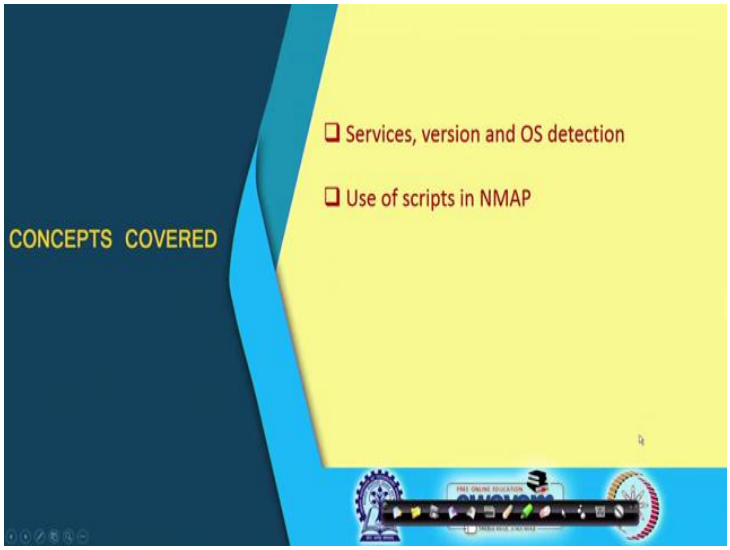
Ethical Hacking
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 58
The NMAP Tool: A Relook (Part – III)

We continue with our discussion on NMAP. In the previous two lectures we have talked about specifically host discovery and the port scanning options.

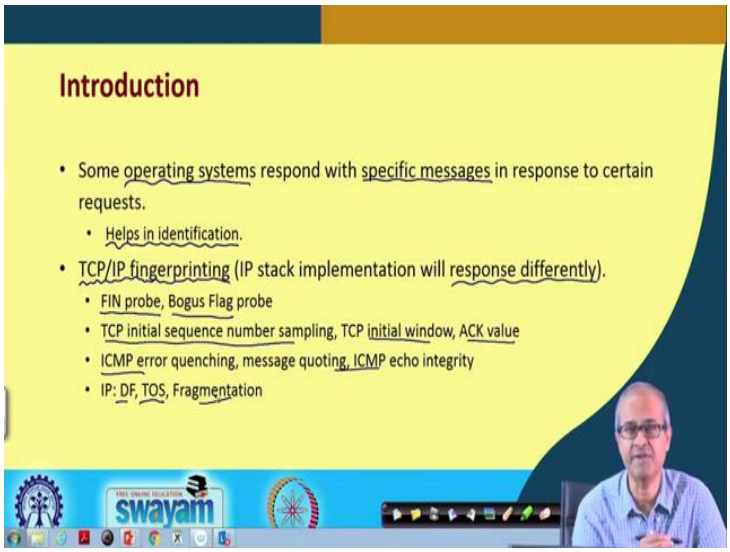
(Refer Slide Time: 00:33)



So, we continue with the discussion in this third part of this lecture, where we shall be mainly talking about how to detect services, version and OS detection options. We shall be looking at how to use the so called in NMAP scripts and lastly we shall be looking at some of the very common example options that we typically use in NMAP.

So, the first thing we talk about is how to detect services, version and the type of operating system that is running on a machine.

(Refer Slide Time: 01:08)



Before that let us try to understand how this kind of scanning is carried out? For instance, how do we detect, what version of operating system is running on a particular host? The idea is like this. Specific operating systems, they respond with specific messages in response to certain kind of requests. Like let us say, I am sending a particular kind of request to a machine; if the machine is running windows, the kind of response I get back will be something; if it is running Linux, the kind of response would be different; if it is running Mac, it can be again different.

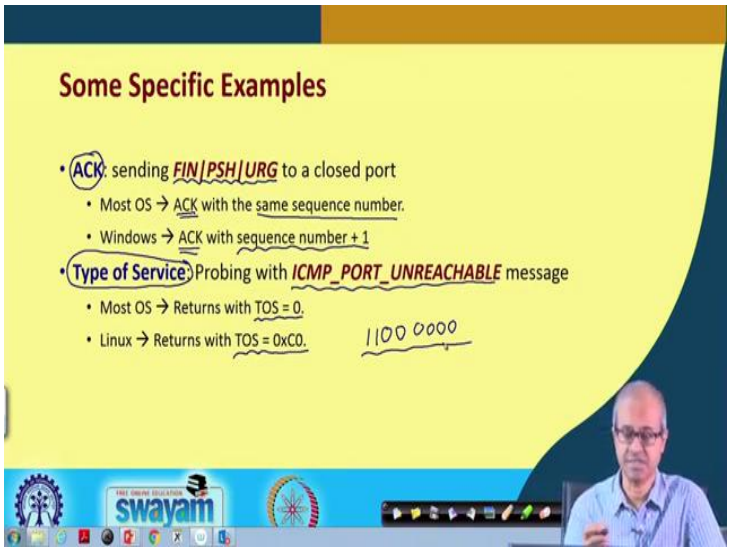
So, by looking at the kind of response I get back, I can guess what kind of operating system is being run, is running on the particular host and this helps in identifying the operating system and also possibly version of the operating system which is running; because responses can be different, ok.

So, we refer to this as TCP/IP fingerprinting; like when you send requests to different operating systems, TCP/IP stack implementation is slightly different across OS versions. So, by looking at the stack implementation, TCP/IP implementation, the response will vary slightly from one version to the other. Depending on that you can identify which version is running; that will give us some clue about identifying the services and the operating systems, ok.

There are various different kinds of requests you can send and the response can be different; like you can try sending a FIN probe; a bogus flag that means, some wrong or

incorrect packet you are trying to send. TCP initial sequence numbers sampling, window, value of the ACK and there are various other ICMP related issues also, which can vary from one system to other, one operating system to another. Related to IP also do not fragment, type of service, fragmentation, there are a number of different things which can vary from one version to another and these are mainly used or tried to be used for identification, ok.

(Refer Slide Time: 04:00)



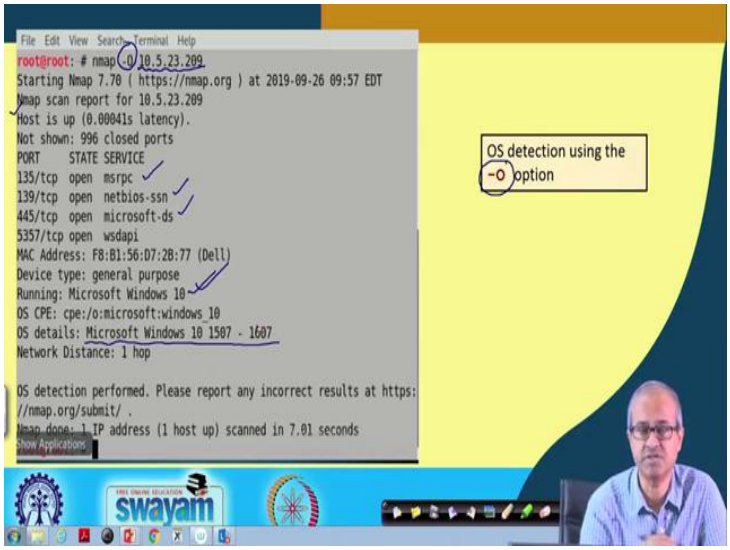
Some specific examples I am showing here; with respect to TCP you think of ACK; suppose we are sending FIN, PSH, URG, urgent data, push, finish all these flag set to a closed port. What will happen is that most of the operating system will send back an acknowledgment packet with the same sequence number. But in windows what happened? Windows handle it slightly differently; it sends back an acknowledgment alright; but this sequence number is also incremented by one.

So, if you look at this sequence number of the packet which is coming back, if it is same, you can conclude that it is a non-windows operating system; if it is incremented by one, you can infer that it is windows; the host is running on some version of windows operating system. Similarly, if you are probing with ICMP PORT UNREACHABLE message and you get back a response, you look at the type of service; for most operating systems the type of service will be returned as 0; but if it is Linux, the type of service is

returned as the hexadecimal code C0. C means, in binary, this is C; this is 0; this will be the 8 bit type of service code that will return by typical Linux systems.

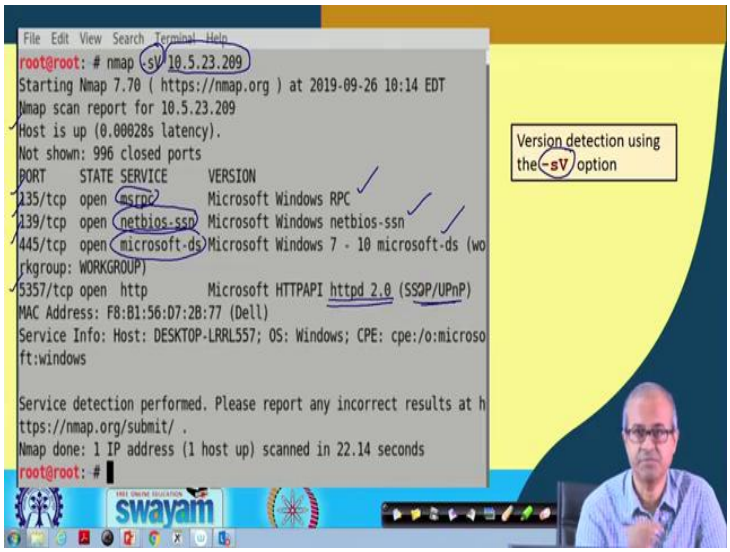
So, if you look at the TOS of the response, you can clearly identify whether it is a Linux or some other operating system. This is usually the way you try and guess the type of operating system and the services which are running, ok.

(Refer Slide Time: 06:05)



Some examples here, here we are giving an option **-o**; **-o** is the OS discovery option. We will here say, we are scanning a particular host 10.5.23.209. If you see the report, the first thing is mentioned; host is up and what are the ports which are of open and that is also shown, and finally, it also shows that it is running Microsoft windows 10. Because, it has obtained some unique fingerprint response which is unique to Microsoft windows also version 10; it has uniquely identified that. So, you can also specify some OS details, Microsoft windows 10 release number also 1507 to 1607; some other details also you can obtain ok, fine. This is one thing.

(Refer Slide Time: 07:21)



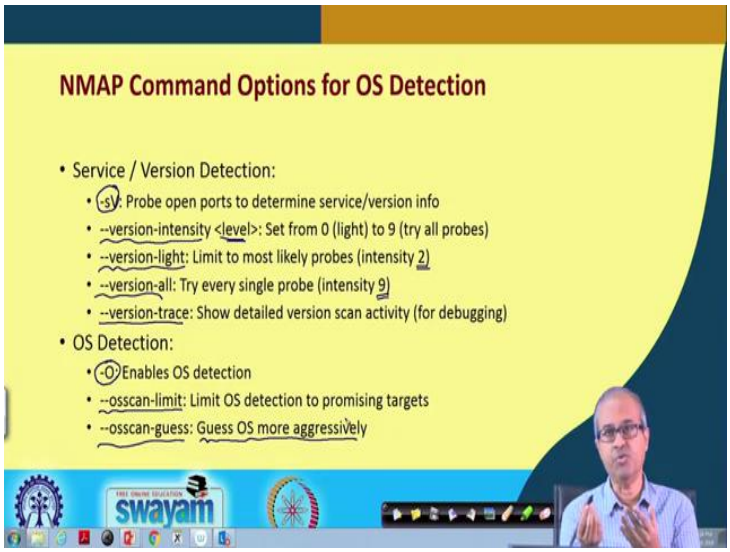
Let us take another example, where we are trying to look at the different versions of the applications that are running on a particular host; we are using the **-sV** option. **sV** means the different services which are running, what are the versions? For example, I am running **telnet**; which **telnet** version is running?

Now, the question is why do you need to know the version? You see, for many services certain versions have known vulnerabilities; the later versions have tried to plug the vulnerabilities. So, you are trying to find out whether this service which is running, is one of the vulnerable versions; if so, then may be a readymade exploit is available for that; you can try to run that exploit and try to break into the system; try to hack into the system, ok. So, here we are running NMAP with the **-sV** option on a particular host.

Here it again says host is up; it talks about the open ports; well, it also, these open ports correspond to these Microsoft remote procedure call, netbios, microsoft-ds, these services running on these port numbers 135, 139, 445 and also it has obtained some information about the version, which version it is running.

Like for example, for RPC, it was not able to obtain a version; netbios-ssn, it is Microsoft windows and for this microsoft-ds, it is windows 7 to 10, Microsoft-ds workgroup and the other one 5357, it is Microsoft HTTPAPI httpd 2.0 with some details. So, you can get some version information also, which can help you to mount further attacks based on this information, ok, service detection performed, ok.

(Refer Slide Time: 09:46)



So, some specific NMAP commands that are used for this operating system detection or service detection are as follows. This **sV** already we have seen with an example; **- - version-intensity** with some level, you can start from 0 which is the lightest to 9 try all probes; that means, how exhaustive will be your discovery process; you can specify by giving a number from 0 to 9; 0 means, least effort; 9 means, maximum effort.

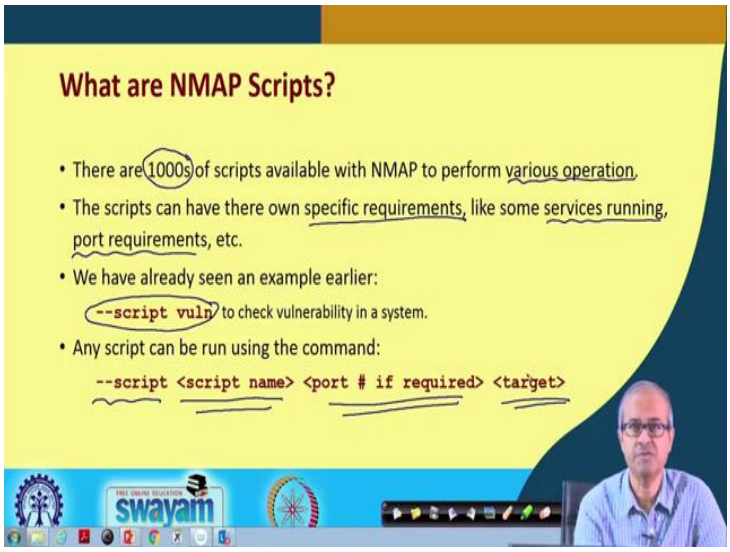
Well, you can directly specify instead of specifying the intensity, you can say **version-light**; that means, you are trying to scan with a low intensity value 2. **version-all**, you scan with an intensity value 9; but in version-intensity you can do a fine tuning specify the exact value. **version-trace**, you show detailed version scan activity; what is happening; all entire details will be logged; so that you can use it for debugging purpose; what is actually happening.

For OS detection we have seen that we can use the **-o** option; you can have **- - osscan-limit**; that means, you only limit your scan to promising targets; do not scan all the machines, all the hosts; only some of the important or notable hosts will be scanned; **osscan-guess**, here you are using more aggressive scan; aggressive mechanisms are there where you can try to guess OS in a more aggressive way which usually will take less time; but the chance of errors will be more obviously, ok.

Now, another very important and useful feature that is available in NMAP, I should not say feature, but utilities that are available is the use of scripts. In NMAP, you have learnt

about the commands; but using this commands you can build application, scripts; there is a scripting language available under NMAP; you can develop scripts on that for specific applications, ok.

(Refer Slide Time: 12:11)



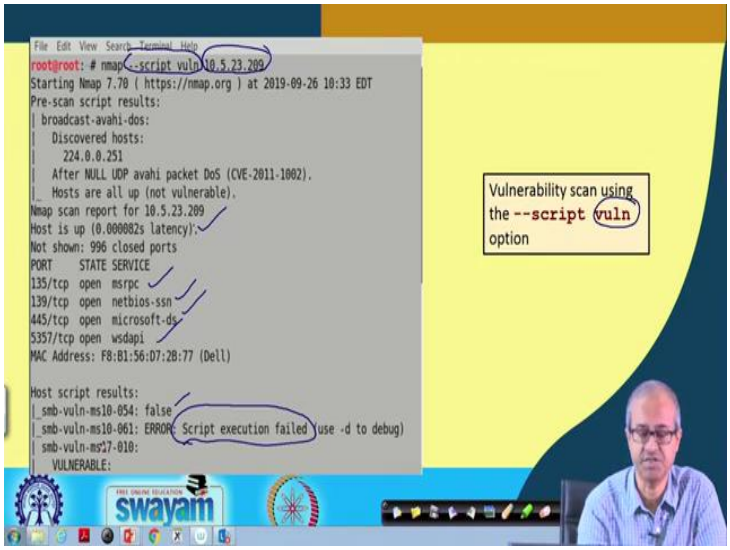
Now, in NMAP there are 1000 of such scripts which are available; someone has written it and there is a repository where all the scripts are available. Some various useful operations, scanning operations, the corresponding scripts are already available and there is a feature in NMAP so that you can run a specific script which is actually a collection of many NMAP commands in some particular sequence.

But, the thing is that when you run a particular script, some specific requirements may be there for the scripts for it to run successfully; like for example, you may need that some specific services must be running; otherwise the scripts will not run; some particular ports must be open; otherwise the script will not run.

So, you need to understand that and if the requirements are satisfied, only then this script can run successfully, ok. Like for example, if you run a script called vulnerability, the **vuln**, the command is like that **- - script** name of the script. So, this is already a script written by someone which will check vulnerability in a system. It will do a lot of different kinds of scan; in general the command is like this **- - script**, you specify the name of the script, then you may specify some port numbers if required, then you may specify the target IP address which target to scan for the, this particular script

vulnerability to scanned. So, you can specify one or more targets, IP addresses or host names, ok.

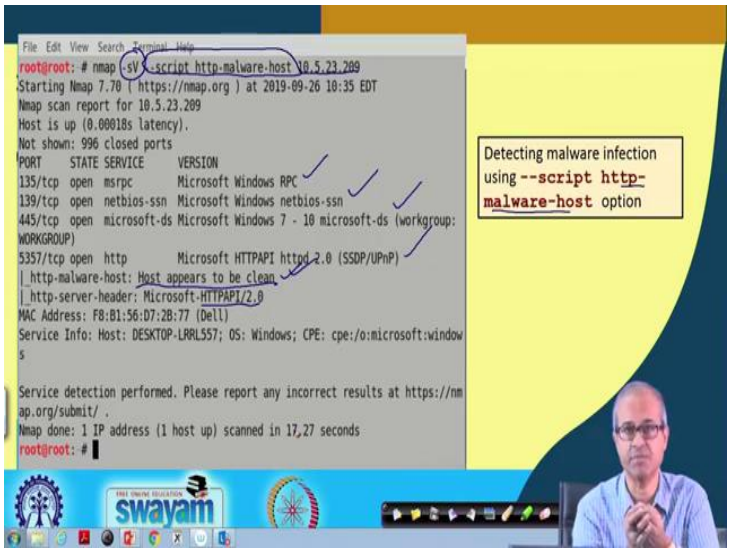
(Refer Slide Time: 14:08)



So, let us look at some specific examples; here I am showing an example where we are running the script vulnerability, **vuln**; you see **- - script vuln** and you are running it on this particular IP address. So, we are running a vulnerability scan; the script is already available, ok. So, discovered hosts here, some hosts; NMAPs scans report for this host is up; it says host is up; these are the 4 services which are up, MAC result so, and script execution field, this is one option; it has been; because one of the ports which are required to run the script was not open.

So, it did this scan to some extent; but finally, it was not complete; of course, here the entire report I am not showing; it generates a long report; a part of the report I am showing here in the screen; this is just a screenshot, ok. So, here you will get a detailed exhaustive scan report, vulnerability scan report; what are the vulnerabilities that the scan was able to detect, ok.

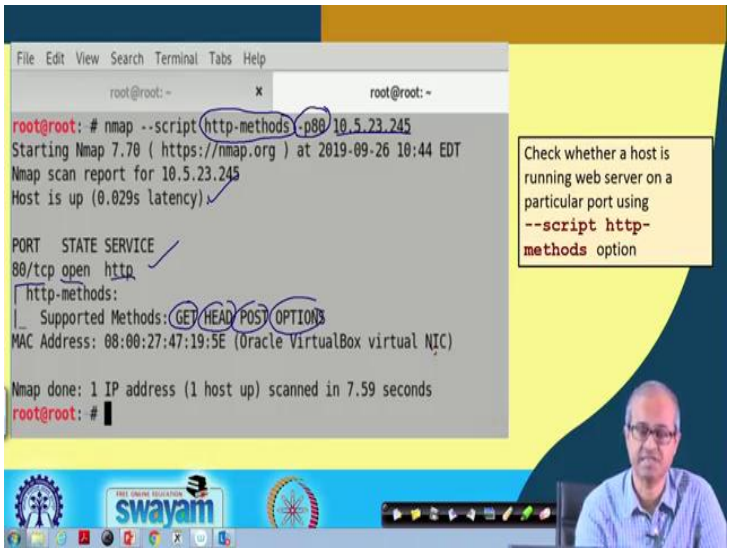
(Refer Slide Time: 15:35)



Similarly, there is another script called **http-malware-host**. So, this is a script which is already written for detecting malwares in the system. Like here I am just running this script and you have to give the flag **-sV** set and this is the particular host which you want to scan. Here again you are seeing the ports which are open; then malware host, **http-malware-host**, host appears to be clean. So, malware was not detected on this host, ok.

So, there are some other messages you can see, the MAC address of the target, an http server; what kind of http server version is running, ok. Service detection performed. So, you can scan the host for malware; well what kind of malware; how it is detected; already those scripts someone is written for you; you are just running it blindly; but when if you are informed hacker, if you want to do it by understanding what you are doing, you need to look at this script and make modifications as and when necessary, ok. Because, you really do not know what the script is; you were running it as a black box; you need to see what is there inside, fine.

(Refer Slide Time: 17:13)

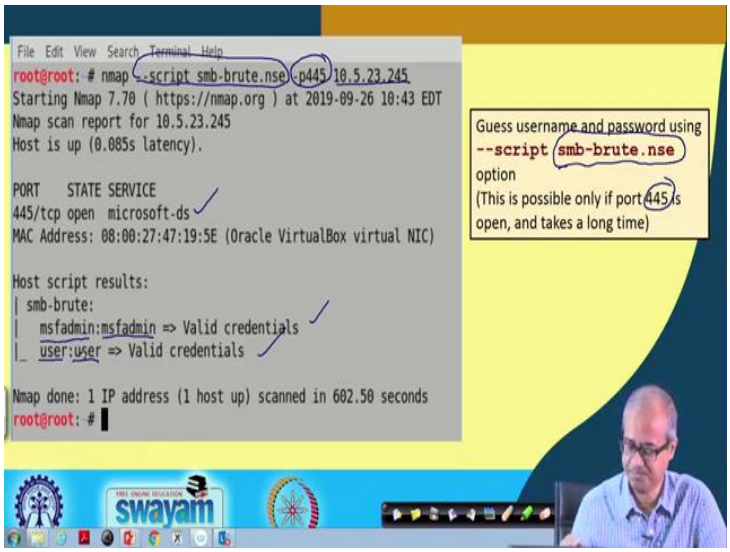


There is another example here, where you are running a script called **http-methods**, where you also specify the port number on which your http server is running; let us say you are saying it is running on port number 80, **-p80**; you are scanning this particular host which is the web server. So, host is up; on 80/tcp, the http port is open and here what you are trying to see this script; it is, you see, in http, http is a protocol right, hypertext transfer protocol; there are many http basic commands which are available, GET, POST, etc.

Now, here you are trying to find out what are the http methods that the web server is supporting? It says supported methods GET, HEAD, POST, OPTIONS, all these commands are available. So, you can send an http request GET, HEAD, POST, OPTIONS and the server responds back. So, these are also some information you need with respect to the web server so that you can later on mount some attacks on the web servers. Because, some of those attacks are based on some http commands; you need to know that what are the commands that are presently acceptable by the web server; the web server supports these commands, ok.

So, here you can also see what kind of web server, Oracle VirtualBox virtual NIC that is running, ok. So, you get some additional information also.

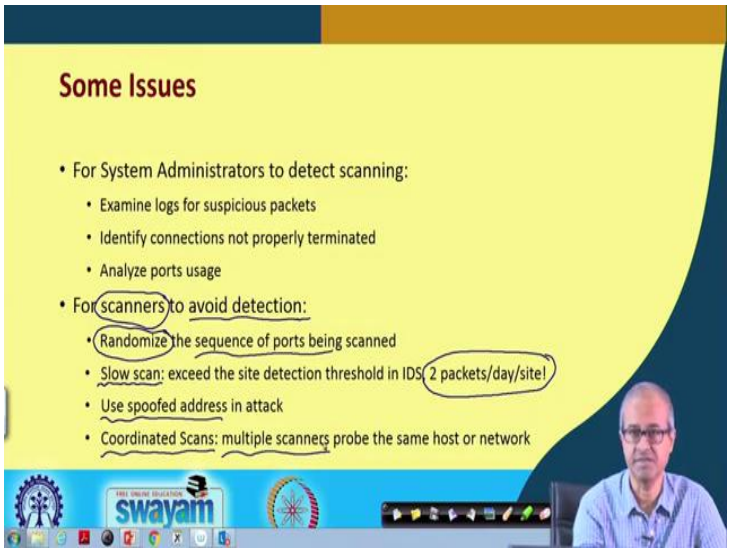
(Refer Slide Time: 18:58)



Here there is another one, where you are trying to mount a brute force attack on username and password breaking; there is a script which is already written. But, one constraint here is again here; this will work only when port number 445 is open and it usually takes a long time; because it checks dictionaries and other things; it has to do a lot of checking, a lot of guessing, ok. So, you are running this one as I said, on port number 445 on a particular host.

So, you are checking that port number 445, the microsoft directory service is running; because it needs to access that; MAC address and after this brute force attack, it was able to find only two; this msfadmin, username and password is also a msfadmin; there is one user:user, password was also user; only these two were detected, ok. But this is just an example where not too many users were there on this machine. So, only two obvious vulnerabilities were found; but if there are many users in the system, most likely many other passwords would be cracked through these tools, ok.

(Refer Slide Time: 20:27)



So, there are some issues here; you need to understand and remember; let us talk from system administrator point of view; suppose you are a system administrator. So, what you would like to know? You would like to know, whether your network is subject to some attack, whether someone from outside is running NMAP and creating a map of your network, is scanning your network. How to detect those things; obviously, logs are the best place to look at.

Logs usually record lot of information from outside who is logging in, what kind of packet is being, what kind of connection requests are coming; lot of information are there in the log. But, it is a huge data; you need to spend a lot of time in analyzing the log; carry out data mining from there and get some interesting information which might lead you to suspect or something wrong is going on, ok. You should try to identify connections that are not properly terminated; a connection was made, but no request for connection termination was carried out.

So, maybe this was an attempt to detect a particular port is open or not; that is why while doing the connection you got that information and you did not care to close the connection later. Analyze port usage, some ports well, you are some request is coming on that port, but after that there are no packets on that port. So, if you analyze port use, you can suspect something; whenever something is, some connection is established on a particular port number, most likely there will be a number of packets which will be

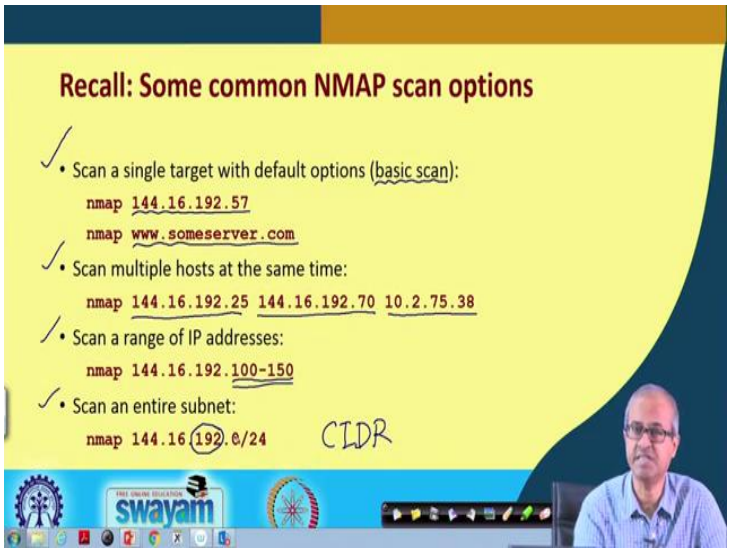
exchanged after that; but if you see that is missing, that may be another suspect, another reason to suspect, ok.

Now, from the other side, this is from the system administrator; now, you think from the point of view of an ethical hacker or not so ethical hacker, but the person who is scanning the network. So, to avoid detection what are the things that you need to do, need to look at? First is that you never scan sequence of ports, 1, 2, 3, 4 sequentially; because in the log it will be recorded and someone can easily see that someone is scanning your port in sequential order. So, you will try to randomize the order of port numbers which you scan; it will be more difficult to guess that a scan is going on.

Another very important thing is slow scan; normally, scans are carried out very fast; but let us say, two packets per day per site; if it is so slow, in whole day only two packets you are sending to get some information. Maybe you will be collecting information over 1 month, 2 months, 6 months and then you will be mounting the attack; but this kind of slow packet requests will normally not get detected; even the ideas, intrusion detection systems will not get triggered with so slow packet rates, ok.

And, always use spoofed address in attack; obviously, so, that your identity will not get disclosed and coordinated scan, instead of the scans coming from one source, if multiple hosts can mount this scanning simultaneously, then just identifying a single target will become difficult; many people are trying to do it at the same time. So, these are some of the, you can say guidelines.

(Refer Slide Time: 24:34)

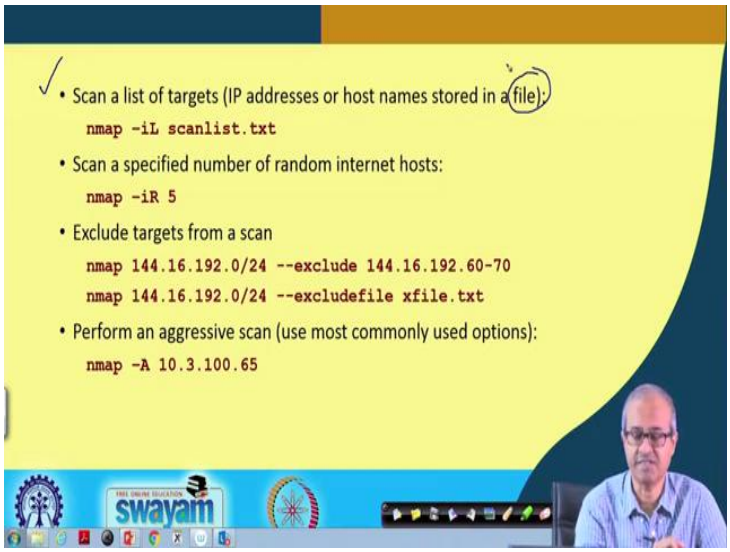


Now, before we end our discussion, let us have a quick recall of some of the common NMAP scan options which mostly people use; let us look at one by one. Scanning a single target with default options; this is what we call as the basic scan. In the basic scan, we use the default options; we do not specify this, the different flags; we do not specify which particular flag to use; we leave it to the default; whatever NMAP takes as default, let NMAP scan according to that. So, you can give a command as simple as that, NMAP followed by just the IP address.

So, NMAP will scan the host as par it default flags, default options; it will create a report for you or you can give an IP address or you can also give a host name, either way. Multiple hosts you can specify; you can scan at the same time; you can specify multiple IP addresses separated by spaces or sometimes separated by commas also. You can specify a range of IP addresses, as we have specified, shown in some examples; 144.16.192; the last byte can be anything from 100 to 150; using dash you can specify a range.

Or you can specify some kind of a subnet, all the hosts in a subnet; say like this, this 144.16.192.0, you see 144 is a class B network and by specifying /24, I am specifying that there is a subnet; so, we are trying to scan this subnet corresponding to 192 in the third byte. So, all 254 hosts inside that subnet let us scan that, ok. So, by this, using this CIDR notation you can specify the entire subnet that you want to scan, ok.

(Refer Slide Time: 27:04)



Let us take this example; scan a list of target; but I am not specifying it; rather I have stored it in a file. So, whatever IP address or hostname you want to scan, suppose I store it in a file first and then using the **-iL** option, I specify the name of the file. So, whatever IP address or hosts names are there in that file, that will get scanned one by one. Then you can specify in random, a number of random internet hosts; you can specify how many; let us say 5 and **-iR** indicates that random. So, randomly the hosts will be selected and that many 5 number of hosts will scanned one by one; this is one.

Sometimes you may want that you do not want to scan some specific hosts; scan others, but do not scan some. So, you can exclude some particular targets from a scan and that you can use using the **- - exclude** command. Like here what I am saying, we are scanning a subnet; let us say, that same example 144.16.192.0/24, but you are excluding all these IP addresses. 144.16.192.60-70, exclude them; you can either specify them by IP addresses or you can use a special version of exclude, **excludefile**; here you can specify a file name. So, the IP addresses or the host to be excluded, they will be stored in this file. So, from that file it will read; it will not scan those, right.

And, lastly perform an aggressive scan; aggressive scan means, the most commonly used action, commonly used options that are used by my typical hackers; use only those options, do not use all; because all will take more time naturally; if you give this **-A** option; this is aggressive; only use the aggressive options for scanning, which most of

the time we will give you the intended information, ok. If you need more information about this NMAP, the command, as I said in that **nmap.org** website there is a nice documentation kind of a book which is available.

In addition there are text books also available on NMAP; there will get all the details; so, it is not possible to cover all commands and all details in the short period of time; but we have tried to give you some kind of a comprehensive overview of the different features available. And, NMAP is a very powerful tool; many people most of the ethical hackers, they use NMAP in the backend to develop their complete penetration testing tool or package.

So, with this we come to the end of this lecture; over the last 3 lectures, we have talked about the NMAP tool, some of the commands; we also showed you some examples. So, I believe so whatever doubts you may be having, at least some of them might have got cleared through these discussions.

Now, in the next lecture we shall be talking about another very important tool which you have also seen in some of the demonstrations, the Wireshark tool; Wireshark, we shall be showing you how to use it, what are the main purposes and main commands and we shall also be seeing some examples there.

Thank you.