

Another NMAP Tutorial

zebulebu

Introduction

This tutorial is aimed at helping those who want to understand what NMap is, what it can do (and, of course, what it is NOT and what it CANNOT do). It should help not only the curious, but also people studying for their C|EH certification (NMap is the de facto standard tool used for gathering information about remote systems and it is therefore essential that anyone wishing to pass the C|EH exam is intimately familiar with NMap and its usage)

Prerequisites

The tutorial will try to explain things in as simple a way as possible, but there is some basic knowledge that the average user will need to make the most of the information contained here. Basic familiarity with the TCP/IP stack is essential - this does not need to be extensive, but the reader should be familiar with IP addressing, usage of TCP ports, know what UDP is and how TCP sessions are established. I will try to explain some of these concepts in more detail, so if you're not that familiar with them hopefully as I go along you'll pick it up. If you have no knowledge of networking, then before you read this series, grab a quick couple of hours with a good basic networking book, or visit one of the many tutorial sites available on the net. For those who are familiar with networking concepts, please ignore the basic stuff as I go along (remember, everyone has to start from the beginning - don't get too impatient if some of this sounds like I'm trying to teach you to suck eggs!)

If you want to follow along with some of the examples, it would be best to run them from a Linux machine. This isn't an exercise in Windows-bashing - its just that the Linux version of NMap (the original version) just runs better. Also, M\$, in their infinite wisdom, didn't follow RFC standards when implementing the TCP/IP stack in Windows, so some of the advanced types of scans flat out will not work under Windows. A decent sniffer is also a must to enable you to capture scan output in 'real time' so you can see what's happening 'under the bonnet'. My favourite - and the industry standard - is Wireshark, but TCPDump or something similar will work just as well.

Finally - and this is very important - please understand that some people get extremely tetchy about you running scans against their networks. Think about it this way: remember how pissed you get when kids run up and bang on your door then run away again? Imagine that scenario and you'll understand how Network Admins feel when they have to deal with IDS alerts firing off left right and centre when some skr1pt k1dd1e comes knocking on their firewall. Use the information you gain from these tutorials wisely, and with discretion.

What is NMap?

"If you know the enemy and know yourself you need not fear the results of a hundred battles"
Sun-Tzu

Put simply, NMap is the Blue Riband of network scanners. Every Security and Network Admin worth their salt knows what NMap is and how to use it. Even if they have nothing more than a passing familiarity with it, they understand its power and the fact that probably 90% of the world's hackers use it to scan networks for vulnerable systems. That said, NMap should not be perceived as a 'bad guy' tool - far from it. Using NMap you can evaluate your own security as a hacker would see it - which is absolutely invaluable if the best security possible is to be achieved.

NMap is designed to detect any open (and closed! and filtered! and firewalled!) ports on a computer and to determine which services may be running on those ports. It can also be used to 'fingerprint' the Operating System of the target machine - by analysing the manner in which the machine responds to the scan, NMap can make a guess (at varying degrees of accuracy) as to what OS the target may be running. As you can probably imagine, such a tool is absolutely invaluable to a hacker, as it enables them to concentrate their energies on running exploits that may be germane to the services running and Operating System running on the target, rather than attempt thousands of exploits which have no

Another NMAP Tutorial

zebulebu

chance of succeeding.

NMap has a myriad of options for running scans, with hundreds of combinations of scan types possible. It is a command line tool (though a GUI does exist for both the Windows and Linux versions) and should, in this author's opinion, be run from it's native environment. Only by running NMap from the command line will you be able to understand and appreciate exactly what NMap is doing - and become familiar with the required options and switches that will set you on the road to understanding how an attacker would use the tool to attempt to gather information about your network.

Getting hold of NMap

NMap is available from Insecure.org at the following URL: <http://insecure.org/nmap/>
Please consider donating should you need to download it - although NMap is 100% free and Open Source (released under the GPL) the guys and gals who develop it need your support in order to continue to develop/provide the tool. Most Linux distros come with NMap as an available package - my personal preference is to run it from the excellent BackTrack Live CD. This CD is a bootable distro based on SLAX and designed for penetration testing/auditing and should be the first thing any serious Security Admin downloads. For the Linux neophyte, although I strongly recommend you run it in it's 'home' environment, Windows binaries are available from the download section of the NMap site here: <http://insecure.org/nmap/download.html>

Once NMap is installed (I won't go into how to do that from here - it's as simple as can be and there are instructions provided in the readme files for all versions) you're probably going to want to jump straight in and start scanning. As is usually the case with well-supported open-source tools, NMap comes with a fantastic ManPage (Manual for Linux n00bs) which will provide you with everything you need to know to use the tool. However, if you were the sort of person who 'RsTFM' then you wouldn't be reading this tutorial now would you? ;)

Understanding the basics of NMap and scanning theory

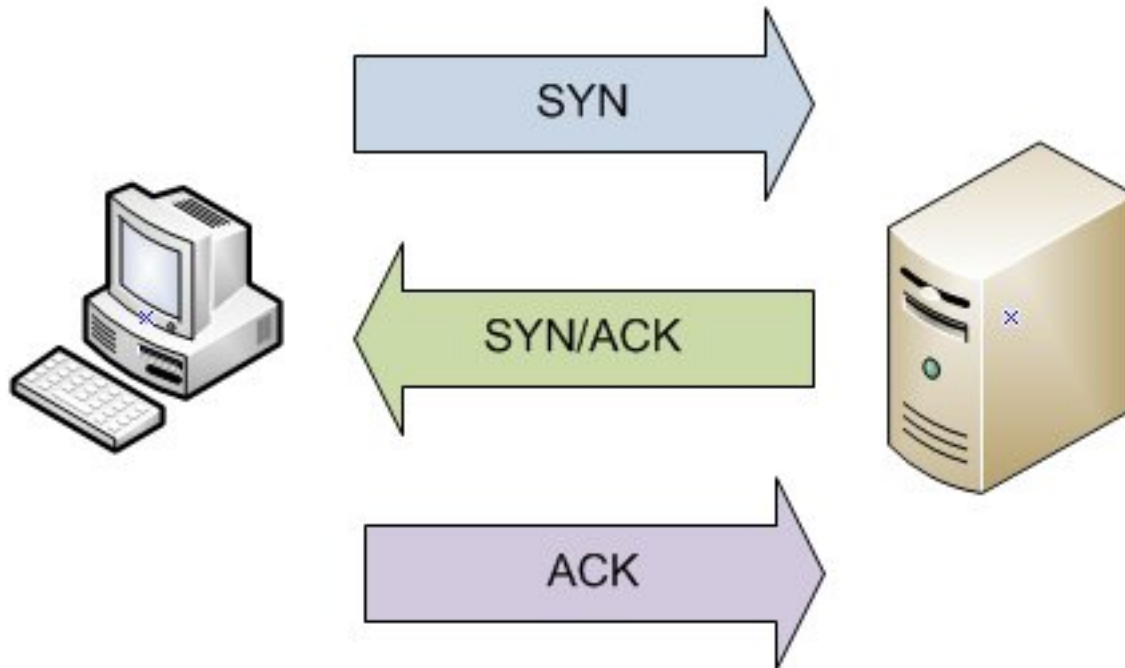
To understand how NMap works, I recommend reading the following basics of the TCP session initiation process. It is absolutely essential that you understand how a TCP session is established, maintained and closed before starting to use NMap in anger. Failure to do this will ensure that not only will you not even begin to understand how powerful NMap can be, but you will also miss out on scanning functionality that could mean your scans miss systems that are 'live'. If you already understand TCP Sessions, then skip this bit to get to the good stuff!

TCP is a STATEFUL protocol. That means that it is 'reliable' in a networking sense - i.e. packets are delivered according to a proper, managed process, allowing for much more accurate delivery than is possible with a 'stateless' or 'connectionless' protocol, such as UDP. Consider the following analogy. If a paperboy checks the address on his route, walks up your garden path and pushes the paper through the correct letterbox, that is akin to a 'stateful' protocol, like TCP. If the paperboy instead rides down your road and chucks papers at random towards any old doorstep or - as used to be the case when I was a kid - just dumps all the papers in the canal, that is akin to UDP.

In order to ensure this reliability, TCP requires that 'sessions' be established and maintained so that an accurate record can be kept of what is occurring with the packets in that transmission. A TCP session is initially established by means of what is commonly known as the 'Three Way Handshake'.

Another NMAP Tutorial

zebulebu



The 'three way handshake' begins with the initiating machine sending a single 'SYN' packet to the target machine (SYN is short for 'Synchronise'). Once the target machine receives the SYN packet, providing it is willing to open a session with the initiating machine, it sends a 'SYN/ACK' packet back, indicating that it is willing to open a session with it (ACK is short for 'Acknowledge'). When the listening machine receives that packet, it knows that the session can be established, and sends a final ACK packet telling the target machine that the session is now established and data transmission can begin. This is, admittedly, a pretty rudimentary explanation of the theory of 'state' and session management, but once you understand these basic principles, you will be able to grasp a lot of the more complex options NMap provides.

Your first scan!

So, with that out of the way, I'm sure you're dying to try your first actual scan of a machine. Go ahead and fire up a command prompt/shell. If you ignored all my earlier advice and plumped for the Windows version, you'll need to navigate to NMap's install directory. You'll also need a target IP address to run your scan against. At this stage, I STRONGLY suggest that you identify a host on your network that you are familiar with, and that you know will not result in the local Security Admin coming and issuing some severe slappage should your scan be detected. In fact, since we will be using the noisiest scan known to man (a TCP Connect scan - more on the differences between scan types in the next instalment of this series) it is imperative that you alert whomever is responsible for security in your environment - lest ye fall victim to the dreaded P45 If you want to go ahead and scan an IP without knowing the full consequences of your actions, then so be it - just don't say I didn't warn you beforehand!

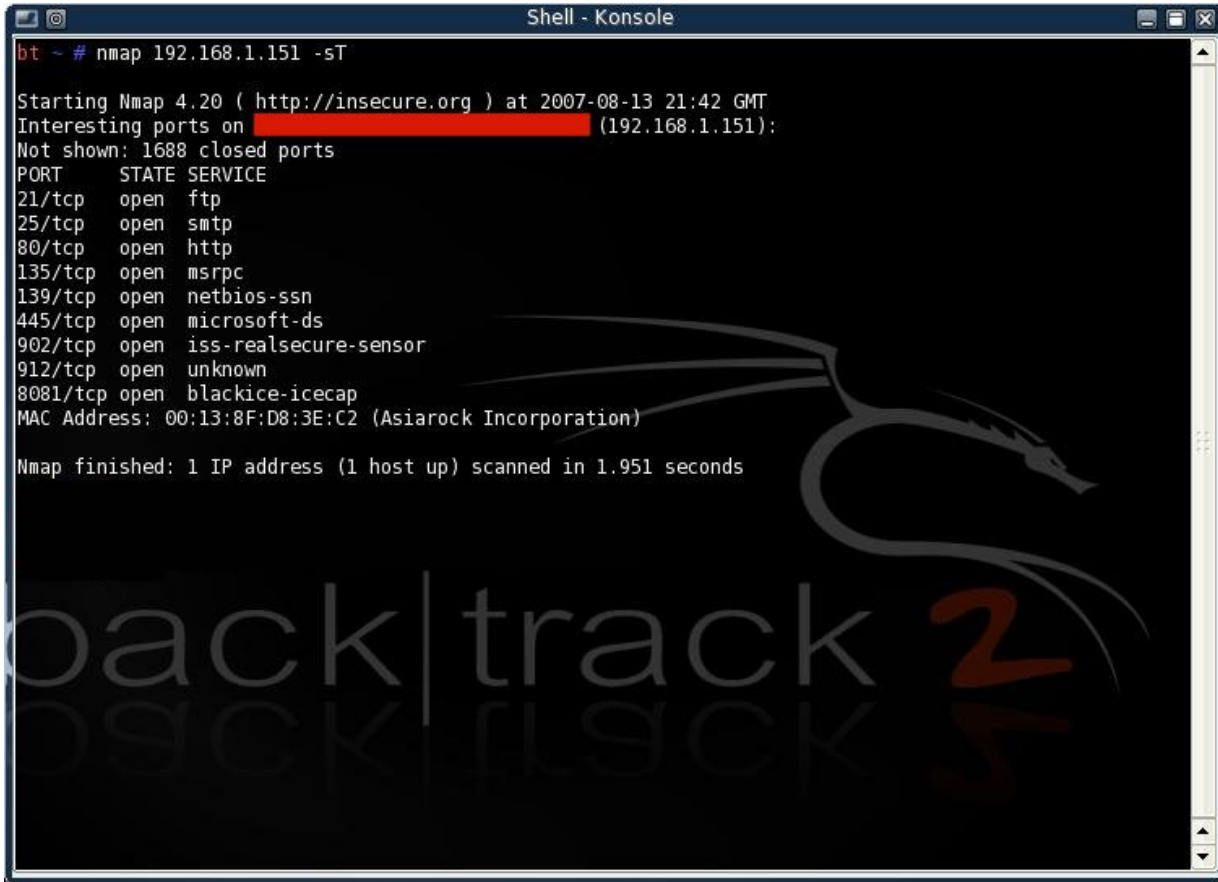
Once you're ready to go, type the following:

```
nmap <ip address> -sT - e.g. nmap 192.168.0.200 -sT
```

Since this is a host on your local network - you DID follow my advice and scan a host on your LAN, right? - the results should be returned pretty swiftly. They will look something like the following:

Another NMAP Tutorial

zebulebu



```
bt ~ # nmap 192.168.1.151 -sT
Starting Nmap 4.20 ( http://insecure.org ) at 2007-08-13 21:42 GMT
Interesting ports on [redacted] (192.168.1.151):
Not shown: 1688 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
25/tcp    open  smtp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
902/tcp   open  iss-realsecure-sensor
912/tcp   open  unknown
8081/tcp   open  blackice-icecap
MAC Address: 00:13:8F:D8:3E:C2 (Asiarock Incorporation)

Nmap finished: 1 IP address (1 host up) scanned in 1.951 seconds
```

Your output will probably be quite different to mine, but as you can see from the output of this test scan I just ran against a box in my lab, a number of open ports have been discovered on the scanned machine, including ports 21, 25, 135, 139 and 445. Already you should be able to see the value of NMap - even at this basic level - as you can see I have now discovered that the box I ran the scan against is running with FTP and SMTP ports open, and is also running services normally associated with a Windows machine (ports 135, 139 and 445 are all associated with SMB running on TCP/IP - Windows filesharing). Looking further down the scans you can also see that the scanned machine is running some services that may look a little odd. For instance, port 8081 appears to be up - and NMap has reported that this is the port that something called 'BlackIce-IceCap' runs on. A quick spot of Googling provides the information that this is a security monitoring program that is often used by hackers! However, as you will see in the next tutorial, results of such clumsily cobbled together 'default' scans are often inaccurate - as it happens, in this case, I know for a fact that the service running on Port 8081 is McAfee's AV Framework service. NMap can be fine-tuned to enumerate exactly what software is running on a particular port - rather than hazarding an uneducated guess, as in this case.

Going Further

As you should already be able to see from this brief introduction, NMap is an excellent tool. In the next section I'll take you a little deeper into the functionality of the tool, explain some different types of scans that can be performed and introduce concepts such as OS fingerprinting and Service Enumeration. This is where NMap really becomes invaluable to the Security/Network Admin!