Network Scanning

NMap and NetCat

Venkata SreeKrishna K. and Lavanya K. Galla
Revisions added by Hannah Pearson and Dillon Harris

July 8, 2017
Version 4.1

University of Idaho

CS 539: Applied Security Concepts

Executive Summary

NMap is a port scanning tool that can be used to detect hosts and services running on a particular network. Identifying hosts on a network is important because you can determine whether services with known vulnerabilities are already running and decide what attack vectors are most likely to be successful. In this tutorial, basic NMap functions are demonstrated and some more advanced features are mentioned. For the first part of this tutorial, you will be scanning a specified IP range and creating a network map.

NetCat is used to transfer data over a network using TCP or UDP. It was designed to be a reliable back-end tool that can be used by other programs and scripts. Like NMap, NetCat can be used for port scanning. However, it is more commonly used for sending data over a network. More relevant to our purposes, NetCat can be used to set up a reverse shell on a target computer and establish persistence. In this tutorial, we will use NetCat to create a chat program as well as to acquire remote shells.

Prerequisites

Basic knowledge of networking, ports, and the bash command line.

Contents

1 NMap Overview >

- NMap:
  - Port scanning tool;
  - Used for network mapping and host discovery;
  - First released on September 1, 1997;
  - Free and open source;
  - Designed to rapidly scan large networks;
  - Graphical interface Zenmap is available;
  - Supports Unix and variants, Windows, and Mac OS.

NMap is a free software (GPLv2) network and port scanner. It is a command line tool that can identify hosts on a network, classify the status of ports on known hosts, and speculate what the operating system of a scanned host is [1].

The primary legitimate task it is used for is keeping track of network resources and uptime. However, it can be illegitimately used to scan someone else's network to identify vulnerabilities. For this reason, it is never a good idea to scan a network unless you have written permission from the appropriate authority.

Using NMap to scan a network without such permission strays into various levels of illegality, depending on the jurisdiction you reside in. It could get you sued or maybe even imprisoned. It goes without saying that it's good practice to always double-check the address you are about to scan with NMap.

NMap is useful because it has a massive variety of arguments that can be passed to it, can scan entire networks very quickly, is built for every major operating system (and in some cases can come packaged with a GUI), and can even be used to scan the ports of local host.

2 Features <>

1. Port Scanning;
2. Operating System Detection;
3. Version Detection;
4. Host Discovery;
5. Scripting Interaction.

1. Port scanning is used to create a list of all the open ports on target hosts. In general port scan is a process that sends client request to a range of server port addresses on a host with a goal of finding an active port.

2. One of NMap's best-known features is remote OS detection using TCP/IP stack fingerprinting. NMap sends a series of TCP and UDP packets to the remote host and examines practically every bit in the responses. If NMap is unable to guess the OS of a machine, and conditions are good, NMap will provide a URL you can use to submit the fingerprint if you know (for sure) the OS running on the machine. [2]

3. Using its NMap-services database of about 2,200 well-known services, NMap would report that those ports probably correspond to a mail server (SMTP), web server (HTTP), and name server (DNS) respectively. This lookup is usually accurate—the vast majority of daemons listening on TCP port 25 are, in fact, mail servers.

4. Host discovery is a service which is used to identify hosts on a network. For example, listing the hosts that respond to TCP or ICMP request or have a particular port open.

5. NMap scripting Engine allows users to write and share simple scripts using the Lua programming language to automate a wide variety of networking tasks. [2]

3 Legal Issues <>

- Unauthorized port scanning is illegal;
- Do not scan networks without permission;
- Reconnaissance can be interpreted as malicious intent;
- Summary: Check your address ranges before scanning.

There are legal issues associated with using NMap to scan a computer, network, or domain where you do not have explicit authorization to do so. In many jurisdictions, using NMap to scan a random website or computer is illegal or leaves you with the possibility of getting sued or your internet service provider revoking your internet service. [1]

4 NMap: Introduction <>

Several options for port scans:

- DNS lookup;
- Ping scans (check if hosts are up and not filtered);
- Different packet types;
- Firewall evasion.

NMap scan results include a list of ports and the state of each port, which is one of the following: open, closed, filtered, unfiltered, open|filtered, closed|filtered.

Initially when an IP address is selected for scanning, NMap looks up the DNS server and resolves the IP. Then NMap pings each of the ports by sending a zero byte packet. If the packets are not received back then it means that the port is closed and if the packets are received back then the port is said to be open. NMap then sends different packets with different timing to determine whether the port is filtered or unfiltered. If firewalls are present on those systems they can interfere with the process.

Here is a more detailed explanation of port states:

1. An open port actively responds to an incoming connection. It shows the available services on the port. [6]
2. A closed port is a port on a target that actively responds to a probe but does not have any service running on the port. Closed ports are commonly found on systems where no firewall is in place to filter incoming traffic. [6]
3. Filtered ports are those which are typically protected by a firewall of some sort that prevents NMap from determining whether or not the port is open or closed. [6]
4. An unfiltered port is a port that NMap can access but is unable to determine whether it is open or closed. [6]
5. An open|filtered port is a port which NMap believes to be open or filtered but cannot determine which exact state the port is actually in. [6]
6. A closed|filtered port is a port that NMap believes to be closed or filtered but cannot determine which respective state the port is actually in. [6]

5 NMap: Target Specification <>

General syntax: **`nmap [target(s)]`**.
Examples of specific options that can be used:

- One IPv4 address: **`nmap 192.168.0.1`**;
- A range of IPv4 addresses: **`nmap 192.168.0.1-100`**;
- A subnet: **`nmap 192.168.0.0/24`**;
- Excluding specific IPs: **`nmap 192.168.0.0/24 -exclude 192.168.0.1`**.

Executing NMap with no command line options will perform a basic scan on the specified target. A target can be specified by IP address or host name. A default NMap scan will check for the 1000 most commonly used TCP/IP ports [6].

Note that while NMap can be used to scan IPv6 targets, using the **`-6`** option, some features such as scanning multiple targets using ranges are not supported [6].
Example: **`$ nmap -6 fe80::29aa:9db9:4164:d80e`**

6 NMap: Task 1 <>

Task 1: Scan the network you are on now. There are several machines on this network located in a hidden folder. Write down the ip addresses of any hosts you find.

Hint: Look at previous slide for syntax help. If that is inadequate, man nmap is always a good option. Third resort: quietly shout at your neighbor or me or someone else and we'll help.

7 NMap: Commonly Used Options <>

1. Aggressive Scan: **`nmap -A [target]`**;
2. OS detection: **`nmap -O [target]`**;
3. Write output to xml file: **`-oX [filename]`**.

Task 2: Scan the network again and save the output to an XML file.

Try using different options and note the difference in scan time and results.

Note that, as common for command line utilities, it is possible to specify multiple options at once.

The -A specifies an aggressive scan. The aggressive scan selects some of the most commonly used options within NMap and is provided as a simple alternative to typing a long string of command line arguments [6].

The -O option scans common ports to and attempts to detect the operating system running on the target machine.

Other output options for output format include: **`-oG [filename]`** to write to a grep file and **`-oA [filename]`** to write to all three available formats [4].

8 NMap: Task 3 <>

Task 3: Scan the network again, using the same options as previously, and save the output in another xml file with a different name. Compare the scans to see if any changes have occurred.

Task hint: You can compare the results of two different scans using **`ndiff scan1.xml scan2.xml`** to see if anything has changed between them [4]. In a real world situation, this might be useful if you're scanning a network and looking to see if any hosts are down, or (worse) if any new unauthorized machines have mysteriously appeared.

9 NMap: Scan Types <>

1. Ping scan: **`nmap -sP [target]`**;
2. No ping scan: **`nmap -PN [target]`**;
3. Syn scan: **`nmap -sS [target]`**;
4. Connect scan: **`nmap -sT [target]`**;
5. UDP scan: **`nmap -sU [target]`**;
6. Protocol scan: **`nmap -sO [target]`**.

The -sP option is used to perform a simple ping on the specified host(s). This option is useful when you want to perform a quick search of the target network to see which hosts are online without actually scanning the target(s) for open ports [6].

When NMap scans a system for open ports it will first ping the target to see if it is online. This feature saves time by skipping targets that do not respond. The -PN option instructs NMap to skip the ping discovery check and perform a complete port scan on the target. This is useful when scanning hosts that are protected by a firewall that blocks ping probes [1]. The -PA option performs a TCP ACK scan on the specified target. The -PA option causes NMap to send TCP ACK packets to the specified hosts. This method attempts to discover hosts by responding to TCP connections that are nonexistent in an attempt to solicit a response from the target. [7]

The -sS option performs a TCP SYN scan. The TCP SYN scan is the default option for privileged users. The default TCP SYN scan attempts to identify the 1000 most commonly used TCP ports by sending a SYN packet to the target and listening for a response. This type of scan is said to be stealthy because it does not attempt to open a full-fledged connection to the remote host. This prevents many systems from logging a connection attempt from your scan. [6]

10 NMap: Port Specification Options <>

1. Fast Scan (top 100 ports): **`nmap -F [target]`**;
2. Scan Specific Ports: **`$ NMap -p [port] [target]`**;
3. Scan Ports by Name: **`$ NMap -p [port name(s)] [target]`**;
4. Scan All Ports: **`$ NMap -p "*" [target]`**;
5. Top ports: **`$ NMap -top-ports 10 [target]`**.

1. The -F option instructs NMap to perform a scan of only the 100 most commonly used ports. NMap scans the top 1000 commonly used ports by default. The -F option reduces that number to 100. [6]
2. The -p option is used to instruct NMap to scan the specified port(s). In addition to scanning a single port, you can scan multiple individual ports or a range of ports. [6]

   Ex: **`$ NMap -p 80 10.10.1.44`**, **`$ NMap -p 25,53,80-200 10.10.1.44`**
3. One can search for open SMTP and HTTP ports by name using the -p option. The name(s) specified must match a service in the NMap-services file. [6]

   Ex: **`$ NMap -p smtp,http 10.10.1.44`**
4. The -p "*" option is a wildcard used to scan all 65,535 TCP/IP ports on the specified target. [6]

   Ex: **`$ NMap -p "*" 10.10.1.41`**
5. One can search only the top n ports running by using the top ports command. [6]

   Ex: **`$ NMap -top-ports 10 192.123.1.1`**

11 Commonly Used Ports <>

Certain ports are commonly reserved for specific services:

- 22: ssh;
- 23: telnet;
- 25: smtp (mail);
- 53: dns;
- 80: http;
- 443: https.

Here are some more commonly used ports [4]:

- 21: ftp;
- 88: Kerberos (network authentication protocol);
- 139: SMB (file sharing);
- 389: LDAP (Lightweight Directory Access Protocol, used on windows domains);
- 902: VMWare;
- 3306: MySQL;
- 3389: RDP (remote desktop protocol);
- 9001: Tor.

12 NMap: Timing and IDS Evasion <>

Timing template options:

- -T0 or paranoid: waits 5 minutes;
- -T1 or sneaky: waits 15 seconds;
- -T2 or polite: waits 0.4 seconds;
- -T3 or normal: the default;
- -T4 or aggressive: a good option for this lab;
- -T5 or insane: as subtle as a police siren at a library.

As one might expect, using a slower timing option will mean the scan takes longer to complete. Options -T0 and -T1 are useful if you are concerned about evading an IDS (Intrusion Detection System). However, at that point it is probably more advisable to specify timing specifically rather than using a template [1]. There are ways to do this by using command line options or by creating your own custom template.

13 NMap: Firewall Evasion <>

The best mitigation for thwarting port scans is a well-configured firewall; That said, here are several of NMap's firewall evasion options [4]:

1. Fragment packets: **`-f`**;
2. Spoof source ip: **`-S [ip]`**;
3. Spoof source port: **`-g [port number]`**;
4. Spoof MAC address: **`-spoof-mac [MAC]`**;
5. Append random data: **`-data-length [size]`**.

It isn't hard to imagine that as a defensive measure, systems adminsitrators would configure the network to block anyone who appears to be performing a port scan. In order to evade this defensive measure that may or may not be in place (but really should), it's a good idea to spoof the information in the packets you're sending so they aren't obviously traceable back to you.

14 NMap: Questions <>

1. What is the port status if it does not allow entry or access to a service?
2. On which port is DNS?
3. What service is located on port 22?
4. What does it mean if a ping scan has not discovered a machine?
5. How can port scanning assist with identification of vulnerabilities?

Answers in appendix.

15 NMap: More Questions <>

6. Which of the following does NMap require for OS identification?

   (a) one open and one closed port;

   (b) two open ports and one filtered port;

   (c) one closed port;

   (d) one open port.

Answer in appendix.

16 NMap: Challenge 1 <>

Challenge 1: Use NMap to discover all the services running on the network. Create a network map, complete with IP addresses and all additional information you can provide about the operating systems and additional services running on the network.

Note: I have hidden the virtual network you are scanning in a folder and am not listing any information about it. However, several hints are available in the appendix.

17 Challenge 2 <>

Challenge 2: Find and connect to a bind shell listening on one of the ports of a host on the network.

As a class and individually, figure out which port(s) the service(s) are listening on and on which IP address, and try to connect to them. If you are successful, you should be able to perform operations on the target machine; try something like `whoami` or `ls` or `pwd`.

There should be enough shells open for everyone to find one, but since you're all scanning the same machine it is a free for all and you have competition. May the odds be ever in your favor. Don't kill all the shells before anyone else gets a chance to connect, but if you do, please start more (or I'll have to).

18 NetCat Overview <>

- Networking program used to write and read data across TCP and UDP network connections
- Released in 1996;
- Network debugging and investigation tool.

NetCat has often been referred to as a "Swiss Army Knife". NetCat functions as both a standalone program and a backend tool in a wide range of applications. It provides a basic TCP/UDP networking subsystem that allows users to interact manually or via script with network applications and services on the application layer. It lets us see raw TCP and UDP data before it gets wrapped in the next highest layer like FTP, SMTP or HTTP. [8]

19 Features <>

- Oubound or inbound connections, TCP or UDP, to or from any ports;
- Ability to use any local source port and any source address;
- Built-in port scanning capabilities;
- Hex dump of transmitted and received data.

Some of the other features include:

- Reading command line arguments from standard input.
- Featured tunneling mode which permits user defined tunneling.
- Optional ability to let other program service establish connections. [8]

20 Uses of NetCat <>

1. Port Scanning;
2. Banner grabbing;
3. Port Listening and redirection;
4. File transfers;
5. Backdoor.

1. Though NetCat has port scanning capabilities it is not preferred because it contains only basic functions or options. NMap is much better for port scanning.
2. Banner grabbing is used to determine the version, operating system or other relevant information about a particular service. It is important if one is looking for a vulnerability associated with a particular version of some service.
3. This is used to redirect both ports and traffic. This is particularly useful if you want to obscure the source of an attack. This technique can also be used to hide NetCat traffic on more common ports, or change ports of applications whose normal ports might be blocked by a firewall.
4. It has the ability to both pull and push files. All the NetCat file transfers are unencrypted.
5. NetCat can be used as back-door where different files and scripts can be executed. [9]

21 Banner Grabbing <>

- Allows individual to gather information about running services/versions from a machine;
- Run netcat in client mode;
- Command:
  `$ nc -v [ip address port]`

Once you have established a connection with the machine, by issuing the get command the return information gives us the web version software and version number. Depending upon which port to be used in banner grabbing their respective information will be displayed like ssh version for port 22 and the HTTP version information for port 80. [8]

22 Questions <>

5. If you would like to establish a UDP connection between a client and server using netcat which command would you use?
6. How to know if netcat is running in client or server mode?

Hint: Use "man netcat" command or Google to help find the answer.

23 Challenge <>

Group up with your partner and create a simple chat room using netcat.

Time limit: 20 minutes

Hint: Think about the client server interaction of netcat. Also if you get stuck use the "man netcat" command or Google for help.

24 Challenge <>

- Group up with your partner and create a TCP connection were you bind a bash shell on the server, so the client can interact with the server remotely through the shell.

Time limit: 25 minutes

25 Question <>

- What are some ways you could trigger a server to run the netcat command used in the prior slide, so you could take advantage of their shell?

26 Challenge <>

- In this challenge you need to transfer a file from one VM to the other. Pair up with your partner to do this. The file is a `.cpp` program where you need to transfer it from your partner's VM to your VM and run it in-order to know what it is, and vice versa.

Time Limit: 20-25 minutes

The commands have not been covered for this yet. Use "man netcat" or Google to help find out how you can perform the file transfer process.

27 Conclusion <

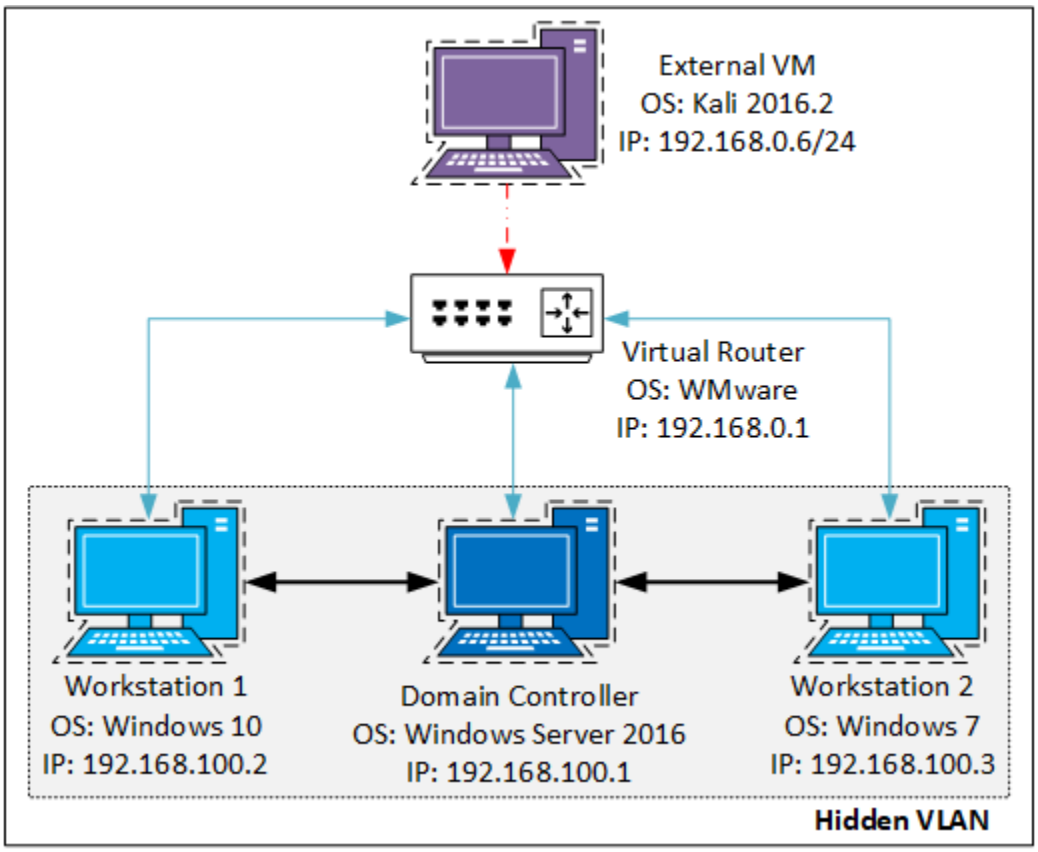
- NMap can be installed on Windows, Linux or Mac OS X;
- Additional parameters give NMap the power to control parallel scanning of a certain number of IP addresses;
- NetCat has two modes of operation : Client and Server;
- The `-e` option which allows netcat to execute programs is what it makes netcat so powerful a.k.a. our reverse shell example.

28 Appendix: VM information, Answers, and Changelog

VM information:

1. Kali comes with NMap and NetCat already installed. So use Kali!
2. If another Linux OS is being used you will most likely have to install NMap and NetCat with the following commands in a bash terminal:
   ```
   $ sudo apt-get update
   $ sudo apt-get install nmap
   $ sudo apt-get install netcat
   ```
3. Set up several other VMs in a hidden folder. This could include an Ubuntu server, a Windows server, an entire windows domain, or, one of the most fun for port scanning, Metasploitable. Recommendation: Use what you have readily available, plus Metasploitable (because it is fun).

Network Diagram:



Answers:

1. What is the port status if it does not allow entry or access to a service? **Closed**
2. On which port is DNS? **53**
3. What service is located on port 22? **SSH**
4. What does it mean if a ping scan has not discovered a machine? **Either the firewall is filtering the ping scan packets out before it can reach the machine, or the machine is not actually running.**
5. How can port scanning assist with identification of vulnerabilities? **May find vulnerable services running or identify promising attack vectors.**
6. The answer is **(a)**: NMap requires one open and one closed port to perform OS identification.
7. Append a -u flag before you initiate the client and server. This will tell netcat to use UDP instead of TCP.
8. The -l flag denotes listening , or server mode. the absence of it indicates a client mode.

   NetCat originates on port 12345, yet the attacker would see the attack coming from port 54321. The piped data is a one way connection, therefore the source cannot receive any response from target. A second relay from target to source must be established to receive a response from the target computer.

Challenge Hints:

1. In order to find all the IP addresses, scan the entire subnet. You can see which subnet you're on using `ifconfig` on a *nix box and `ipconfig` on Windows. For this tutorial, you should be scanning the same subnet that you're on.
2. Hint 1: Use -O to identify what OS is running. That might be useful information.
   Hint 2: For this challenge start searching ports which are not common. Try to search a range of ports
   Hint 3: Try using the -sV option to identify which services are running.
3. Use the commands -l for the server to listen and on the client side use the IP address. Don't forget to add the port on which you want to listen.
4. In order to do this challenge use the concept of the previous challenge where on the source the IP address must be present and destination must be able to listen to the port number which you assign. Remember in this challenge you will have to use the symbols " $<$, $>$ " in order to specify the file name. Figure out which symbol must be present on source and which one on the destination.

Changelog:

Network Scanning: NMap and NetCat			
Ver.	Date	Authors	Changes
v1	Feb. 10th 2016	Venkata SreeKrishna K. and Lavanya K. Galla	First draft of tutorial
v2	Jun. 1st 2016	Adam Odell	Moderate content addition and fixed grammar mistakes
v2.1	Jul. 19th 2016	Adam Odell	Added appendix and other minor enhancements
v4.0	Jan. 30th 2017	Hannah Pearson and Dillon Harris	Revised executive summary. Changed slide content and syntax, removed news slide due to time and lack of practical importance, and rearranged NMap slides to follow a more intuitive sequence. Added Commonly Used Ports, NMap Timing, and NMap Firewall Evasion Options to NMap section; removed, edited, and added questions, changing challenge 2 and tasks 1 and 3 in NMap section; and added one question and three challenges to NetCat section
v4.1	July 8th 2017	Ananth Jillepalli	Standardization (network layout diagram, edits, consistency, TeX markup cleaning, and more).

References

[1] *NMap*. NMap Security Scanner.
https://NMap.org/

[2] Lyon, G. NMap Network Scanning, NMap Project, The Internet, 2009.

[3] *NMap 7 Brings Faster Scanning and Improved IPv6 Support*. 23 November 2015.
http://www.infoworld.com/article/3007301/network-security/nmap-7-brings-faster-scanning-and-improved-ipv6-support.html

[4] Clark, Ben. *RTFM: Red Team Field Manual*. 2013.

[5] *SourceForge Accused of Hijacking NMap Project Account*. 3 June 2015.
http://www.digitaltrends.com/computing/sourceforge-accused-of-hijacking-n

[6] Marsh, Nicholas. NMap Cookbook: The Fat-free Guide to Network Scanning. CreateSpace, 2010.

[7] *Introducing NMap*
http://scitechconnect.elsevier.com/wp-content/uploads/2013/09/Introducing-NMap.pdf

[8] *Introduction to NetCat*
http://scitechconnect.elsevier.com/wp-content/uploads/2013/09/Introduction-to-NetCat.pdf

[9] Gregg, Michael. Certified ethical hacker (CEH) cert guide. Pearson IT Certification, 2013.