Sequências de comando para Nmap

Nmap Target Selection

https://hackertarget.com/nmap-cheatsheet-a-quick-reference-guide/

Scan a single IP nmap 192.168.1.1
Scan a host nmap www.testhostname.com
Scan a range of IPs nmap 192.168.1.1-20
Scan a subnet nmap 192.168.1.0/24
Scan targets from a text file nmap -iL list-of-ips.txt
These are all default scans, which will scan 1000 TCP ports. Host discovery will take place.

Nmap Port Selection

Scan a single Port nmap -p 22 192.168.1.1
Scan a range of ports nmap -p 1-100 192.168.1.1
Scan 100 most common ports (Fast) nmap -F 192.168.1.1
Scan all 65535 ports nmap -p- 192.168.1.1
Nmap Port Scan types

Scan using TCP connect nmap -sT 192.168.1.1
Scan using TCP SYN scan (default) nmap -sS 192.168.1.1
Scan UDP ports nmap -sU -p 123,161,162 192.168.1.1
Scan selected ports - ignore discovery nmap -Pn -F 192.168.1.1
Privileged access is required to perform the default SYN scans. If privileges are insufficient a TCP connect scan will be used. A TCP connect requires a full TCP connection to be established and therefore is a slower scan. Ignoring discovery is often required as many firewalls or hosts will not respond to PING, so could be missed unless you select the -Pn parameter. Of course this can make scan times much longer as you could end up sending scan probes to hosts that are not there.

Take a look at the Nmap Tutorial for a detailed look at the scan process.

Service and OS Detection

Detect OS and Services nmap -A 192.168.1.1
Standard service detection nmap -sV 192.168.1.1

More aggressive Service Detection nmap -sV --version-intensity 5 192.168.1.1
Lighter banner grabbing detection nmap -sV --version-intensity 0 192.168.1.1
Service and OS detection rely on different methods to determine the operating system or service running on a particular port. The more aggressive service detection is often helpful if there are services running on unusual ports. On the other hand the lighter version of the service will be much faster as it does not really attempt to detect the service simply grabbing the banner of the open service.

Nmap Output Formats

Save default output to file nmap -oN outputfile.txt 192.168.1.1
Save results as XML nmap -oX outputfile.xml 192.168.1.1
Save results in a format for grep nmap -oG outputfile.txt 192.168.1.1
Save in all formats nmap -oA outputfile 192.168.1.1
The default format could also be saved to a file using a simple file redirect command > file. Using the -oN option allows the results to be saved but also can be monitored in the terminal as the scan is under way.

Digging deeper with NSE Scripts

Scan using default safe scripts nmap -sV -sC 192.168.1.1
Get help for a script nmap --script-help=ssl-heartbleed
Scan using a specific NSE script nmap -sV -p 443 –script=ssl-heartbleed.nse 192.168.1.1
Scan with a set of scripts nmap -sV --script=smb* 192.168.1.1
According to my Nmap install there are currently 471 NSE scripts. The scripts are able to perform a wide range of security related testing and discovery functions. If you are serious about your network scanning you really should take the time to get familiar with some of them.

The option --script-help=$scriptname will display help for the individual scripts. To get an easy list of the installed scripts try locate nse | grep script.

You will notice I have used the -sV service detection parameter. Generally most NSE scripts will be more effective and you will get better coverage by including service detection.

A scan to search for DDOS reflection UDP services

Scan for UDP DDOS reflectors nmap –sU –A –PN –n –pU:19,53,123,161 –script=ntp-monlist,dns-recursion,snmp-sysdescr 192.168.1.0/24
UDP based DDOS reflection attacks are a common problem that network defenders come up against. This is a handy Nmap command that will scan a target list for systems with open UDP services that allow these attacks to take place. Full details of the command and the background can be found on the Sans Institute Blog where it was first posted.

HTTP Service Information

Gather page titles from HTTP services nmap --script=http-title 192.168.1.0/24
Get HTTP headers of web services nmap --script=http-headers 192.168.1.0/24
Find web apps from known paths nmap --script=http-enum 192.168.1.0/24
There are many HTTP information gathering scripts, here are a few that are simple but helpful when examining larger networks. Helps in quickly identifying what the HTTP service is that is running on the open port. Note the http-enum script is particularly noisy. It is similar to Nikto in that it will attempt to enumerate known paths of web applications and scripts. This will inevitably generated hundreds of 404 HTTP responses in the web server error and access logs.

Detect Heartbleed SSL Vulnerability

Heartbleed Testing nmap -sV -p 443 --script=ssl-heartbleed 192.168.1.0/24
Heartbleed detection is one of the available SSL scripts. It will detect the presence of the well known Heartbleed vulnerability in SSL services. Specify alternative ports to test SSL on mail and other protocols (Requires Nmap 6.46).

IP Address information

Find Information about IP address nmap --script=asn-query,whois,ip-geolocation-maxmind 192.168.1.0/24
Gather information related to the IP address and netblock owner of the IP address. Uses ASN, whois and geoip location lookups. See the IP Tools for more information and similar IP address and DNS lookups.

Remote Scanning

Depending on network perimeter you are scanning remember scanning Internet resources from an external perspective is key when assessing your exposure. This is the reason we offer a hosted or online version of the Nmap port scanner. To enable remote scanning easily and effectively because anyone who has played with shodan.io knows very well how badly people test their perimeter networks.

Additional Resources

The above commands are just a taste of the power of Nmap. Check out the full set of features by running Nmap with no options. The creator of Nmap Fyodor has a book available that covers the tool in depth. You could also check out our Nmap Tutorial that has more information and tips.

Nmap Cheat Sheet: From Discovery to Exploits – Part 1: Introduction to Nmap

As always during reconnaissance, scanning is the initial stage for information gathering.
What is Reconnaissance?
Reconnaissance is to collect as much as information about a target network as possible. From a hacker's perspective, the information gathered is very helpful to make an attack, so to block that type of malicious attempt, generally a penetration tester tries to find the information and to patch the vulnerabilities, if found. This is also called Footprinting. Usually by information gathering, someone can find the below information:
E-mail Address
Port no/Protocols
OS details
Services Running
Traceroute information/DNS information
Firewall Identification and evasion
And many more…
So for information gathering, scanning is the first part. For scanning, Nmap is a great tool for discovering Open ports, protocol numbers, OS details, firewall details, etc.
Introduction To Nmap

Nmap (Network Mapper) is an open-source tool that specializes in network exploration and security auditing, originally published by Gordon "Fyodor" Lyon. The official website is (http://nmap.org). Nmap is a free and open source (license) utility for network discovery and security auditing. Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime. Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics. It was designed to rapidly scan large networks, but works fine against single hosts. Nmap runs on all major computer operating systems, and official binary packages are available for Linux, Windows, and Mac OS X.

ETHICAL HACKING TRAINING – RESOURCES (INFOSEC)

Installation Of Nmap

Nmap has great support for different environments.

Windows: Install from the official site http://nmap.org For Windows, both GUI and command line options are available. The GUI option for Nmap is Zenmap.

Linux (Ubuntu and Debian): Fire the command in the Linux terminal: apt-get install nmap

In the below image, I have already installed Nmap.

```
root@ubuntu:~# apt-get install nmap
Reading package lists... Done
Building dependency tree
Reading state information... Done
nmap is already the newest version.
The following packages were automatically installed and are no longer required:
  gir1.2-timezonemap-1.0 openjdk-7-jre-lib
Use 'apt-get autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

For Red Hat and Fedora based systems: yum install nmap

For Gentoo Linux based systems: emerge nmap

Here, I will show everything in the Linux terminal.

Nmap Scripting Engine

The Nmap Scripting Engine (NSE) is one of Nmap's most powerful and flexible features. It allows users to write (and share) simple scripts to automate a wide variety of networking tasks. Basically these scripts are written in Lua programming language. Generally Nmap's script engine does lots of things, some of them are below:

Network discovery

This is Nmap's bread and butter. Examples include looking up WhoIs data based on the target domain, querying ARIN, RIPE, or APNIC for the target IP to determine ownership, performing identd lookups on open ports, SNMP queries, and listing available NFS/SMB/RPC shares and services.

Vulnerability detection

When a new vulnerability is discovered, you often want to scan your networks quickly to identify vulnerable systems before the bad guys do. While Nmap isn't a comprehensive vulnerability scanner, NSE is powerful enough to handle even demanding vulnerability checks. Many vulnerability detection scripts are already available, and they plan to distribute more as they are written.

Backdoor detection

Many attackers and some automated worms leave backdoors to enable later reentry. Some of these can be detected by Nmap's regular expression-based version detection.

Vulnerability exploitation

As a general scripting language, NSE can even be used to exploit vulnerabilities rather than just find them. The capability to add custom exploit scripts may be valuable for some people (particularly penetration testers), though they aren't planning to turn Nmap into an exploitation framework such as Metasploit.

As you can see below, I have used (-sc) options (or –script), which is a default script scan for the target network. You can see we got ssh, rpcbind, netbios-sn but the ports are either filtered or closed, so we can say that may be there are some firewall which is blocking our request. Later we will discuss how to identify firewalls and try to evade them.

```
Applications Places
root@u
File  Edit  Tabs  Help
root@ubuntu:~# nmap -sC -p22,111,139 -T4 [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-16 23:09 IST
Nmap scan report for [redacted]
Host is up (0.27s latency).
PORT    STATE    SERVICE
22/tcp  filtered ssh
111/tcp closed   rpcbind
139/tcp filtered netbios-ssn

Nmap done: 1 IP address (1 host up) scanned in 5.48 seconds
root@ubuntu:~#
```

Now I m going to run a ping scan with discovery mode on (script) so that it will try all possible methods for scanning, that way I will get more juicy information.

```
Applications Places                                                    11:50 PM
root@ubuntu: ~
File  Edit  Tabs  Help
root@ubuntu:~# nmap -sP --script discovery [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-16 23:36 IST
Pre-scan script results:
| broadcast-igmp-discovery:
|   192.168.58.1
|     Interface: eth0
|     Version: 2
|     Group: 224.0.0.252
|     Description: Link-local Multicast Name Resolution (rfc4795)
|_  Use the newtargets script-arg to add the results as targets
| broadcast-eigrp-discovery:
|_  ERROR: Couldn't get an A.S value.
| http-icloud-findmyiphone:
|_  ERROR: No username or password was supplied
| targets-ipv6-multicast-mld:
|   IP: fe80::49fa:9f73:1e2e:c926  MAC: 00:50:56:c0:00:08  IFACE: eth0
|
|_  Use --script-args=newtargets to add the results as targets
| http-icloud-sendmsg:
|_  ERROR: No username or password was supplied
| targets-asn:
|_  targets-asn.asn is a mandatory parameter
| targets-ipv6-multicast-slaac:
|   IP: fe80::49fa:9f73:1e2e:c926  MAC: 00:50:56:c0:00:08  IFACE: eth0
|   IP: fe80::6920:cb76:3022:fd61  MAC: 00:50:56:c0:00:08  IFACE: eth0
|_  Use --script-args=newtargets to add the results as targets
| broadcast-ping:
|   IP: 192.168.58.2  MAC: 00:50:56:f8:58:dd
|_  Use --script-args=newtargets to add the results as targets
Nmap scan report for [redacted]
Host is up (0.0017s latency).

Host script results:
|_ip-geolocation-geoplugin: ERROR: Script execution failed (use -d to debug)
|_asn-query: No Answers
|_ip-geolocation-maxmind: ERROR: Script execution failed (use -d to debug)
|_hostmap-bfk: Error: could not GET http://www.bfk.de/bfk_dnslogger.html?query=[redacted]

root@ubuntu: ~   Pictures   root@ubuntu: ~
```

As you can see in the image, it is trying all possible methods as per script rules. See the next image for more information.

```
Applications Places                                                    11:51 PM
root@ubuntu: ~
File  Edit  Tabs  Help
|   DNS Brute-force hostnames
|     corp.[redacted]
|     ssl.[redacted]
|     mx1.[redacted]
|     mysql.[redacted]
|     linux.[redacted]
|     blog.w[redacted]
|     sql.wh[redacted]
|     local.[redacted]
|     log.wh[redacted]
|     ns1.wh[redacted]
|     mail3.[redacted]
|     stats.[redacted]
|     databa[redacted]
|     ads.wh[redacted]
|     demo.w[redacted]
|     dns.wh[redacted]
|     ns3.wh[redacted]
|     direct[redacted]
|     oracle[redacted]
|     exchan[redacted]
|     gw.whi[redacted]
|     owa.wh[redacted]
|     mta.wh[redacted]
|     firewa[redacted]
|     forum.[redacted]
|     http.w[redacted]
|     cdn.[redacted]
|     id.w[redacted]
|     mx1.[redacted]
|     ftp0[redacted]
|     images[redacted]
|     cms.wh[redacted]
|     log.wh[redacted]
|     intern[redacted]
|_http-robtex-shared-ns: ERROR: Script execution failed (use -d to debug)
|_hostmap-robtex: ERROR: Script execution failed (use -d to debug)
|_ip-geolocation-geobytes: ERROR: Script execution failed (use -d to debug)

root@ubuntu: ~   Pictures   root@ubuntu: ~
```

Can you see the interesting ports and protocols? You can see dns-bruteforce found that host contains some blog, cms, sql, log, mail, and many more. So here we can perform SQL injection,

the blog may be WordPress, Joomla, etc., so we can attack for a known CMS vulnerability, and obviously the method will be black-box pentesting.
In the upcoming chapter I will describe how to write your own Nmap script engine, and how to exploit them using Nmap.
Basic Scanning Techniques
So here I will show the basic techniques for scanning network/host. But before that, you should know some basic stuff regarding Nmap status after scanning.
Port Status: After scanning, you may see some results with a port status like filtered, open, closed, etc. Let me explain this.
Open: This indicates that an application is listening for connections on this port.
Closed: This indicates that the probes were received but there is no application listening on this port.
Filtered: This indicates that the probes were not received and the state could not be established. It also indicates that the probes are being dropped by some kind of filtering.
Unfiltered: This indicates that the probes were received but a state could not be established.
Open/Filtered: This indicates that the port was filtered or open but Nmap couldn't establish the state.
Closed/Filtered: This indicates that the port was filtered or closed but Nmap couldn't establish the state.
Let's Scan Hosts
Scan A Single Network
Go to your Nmap (either Windows/Linux) and fire the command: nmap 192.168.1.1(or) host name.

```
root@ubuntu:~# nmap [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-20 00:25 IST
Nmap scan report for [redacted]
Host is up (1.1s latency).
Not shown: 978 closed ports
PORT     STATE    SERVICE
3/tcp    filtered compressnet
4/tcp    filtered unknown
9/tcp    filtered discard
13/tcp   filtered daytime
19/tcp   filtered chargen
21/tcp   open     ftp
25/tcp   open     smtp
26/tcp   open     rsftp
53/tcp   open     domain
80/tcp   open     http
110/tcp  open     pop3
139/tcp  filtered netbios-ssn
143/tcp  open     imap
443/tcp  open     https
465/tcp  open     smtps
514/tcp  filtered shell
587/tcp  open     submission
993/tcp  open     imaps
995/tcp  open     pop3s
2222/tcp open     EtherNet/IP-1
3306/tcp open     mysql
5060/tcp filtered sip
```

Scan Multiple Network/Targets

In Nmap you can even scan multiple targets for host discovery/information gathering.

Command: map host1 host2 host3 etc….It will work for the entire subnet as well as different IP addresses.

```
File  Edit  Tabs  Help
root@ubuntu:~# nmap 192.168.58.128  192.168.58.127  192.168.58.129

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-20 00:31 IST
Nmap scan report for 192.168.58.128
Host is up (0.000018s latency).
All 1000 scanned ports on 192.168.58.128 are closed

Nmap done: 3 IP addresses (1 host up) scanned in 13.67 seconds
```

You can also scan multiple website/domain names at a time with the same command. See the below picture. It will convert the domain name to its equivalent IP address and scan the targets.

```
root@ubuntu:~# nmap [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-20 00:30 IST
Nmap scan report for [redacted]
Host is up (0.00052s latency).
Other addresses for [redacted]
rDNS record for 173.[redacted]
All 1000 scanned por[redacted]4) are filtered

Nmap scan report for [redacted]
Host is up (0.00065s latency).
Other addresses for [redacted]
rDNS record for [redacted]
All 1000 scanned [redacted]

Nmap scan report for [redacted]
Host is up (0.00097s latency).
Other addresses for [redacted]
All 1000 scanned por[redacted]

Nmap done: 3 IP addresses (3 hosts up) scanned in 108.79 seconds
Nmap scan report for 192.168.58.254
Host is up (0.00037s latency).
MAC Address: 00:50:56:E2:71:C4 (VMware)
Nmap done: 256 IP addresses (4 hosts up) scanned in 4.82 seconds
```

Scan a Range Of IP address

Command:nmap 192.168.2.1-192.168.2.100

Nmap can also be used to scan an entire subnet using CIDR (Classless Inter-Domain Routing) notation.

Usage syntax: nmap [Network/CIDR]

Ex:nmap 192.168.2.1/24

Scan a list of targets

If you have a large number of systems to scan, you can enter the IP address (or host names) in a text file and use that file as input for Nmap on the command line.

syntax: nmap -iL [list.txt]

Scan Random Targets

The -iR parameter can be used to select random Internet hosts to scan. Nmap will randomly generate the specified number of targets and attempt to scan them.

syntax: nmap -iR [number of host]

It is not a good habit to do a random scan unless you have been given some project.

The –exclude option is used with Nmap to exclude hosts from a scan.

syntax: nmap [targets] –exclude [host(s)]

ex:nmap 192.168.2.1/24 –exclude 192.168.2.10

Aggressive Scan

The aggressive scan selects most commonly used options within Nmap to try to give a simple alternative to writing long strings. It will also work for traceroute, etc.

Command:nmap –A host

Discovery With Nmap

Discovery with Nmap is very interesting and very helpful for penetration testers. During discovery one can learn about services, port numbers, firewall presence, protocol, operating system, etc. We will discuss one by one.

Don’t Ping

The -PN option instructs Nmap to skip the default 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.

Syntax:nmap –PN Target

```
root@ubuntu:~# nmap -PN [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 11:49 IST
Nmap scan report for [redacted]
Host is up (1.1s latency).
Not shown: 975 closed ports
PORT     STATE    SERVICE
1/tcp    filtered tcpmux
3/tcp    filtered compressnet
4/tcp    filtered unknown
6/tcp    filtered unknown
7/tcp    filtered echo
9/tcp    filtered discard
13/tcp   filtered daytime
17/tcp   filtered qotd
19/tcp   filtered chargen
21/tcp   open     ftp
22/tcp   filtered ssh
25/tcp   open     smtp
26/tcp   open     rsftp
53/tcp   open     domain
80/tcp   open     http
110/tcp  open     pop3
143/tcp  open     imap
443/tcp  open     https
465/tcp  open     smtps
514/tcp  filtered shell
587/tcp  open     submission
993/tcp  open     imaps
995/tcp  open     pop3s
2222/tcp open     EtherNet/IP-1
3306/tcp open     mysql
```

By specifying these options, Nmap will discover the open ports without ping, which is the unpingable system.

Ping Only Scan

The -Sp option is responsible for a ping only scan. It will be more useful when you have a group of IP addresses and you don't know which one is reachable. By specifying a particular target, you can get even more information, like MAC address.

Syntax:nmap –Sp target

```
root@ubuntu:~# nmap -sP 192.168.58.1/24

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 12:00 IST
Nmap scan report for 192.168.58.1
Host is up (0.0023s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.58.2
Host is up (0.00012s latency).
MAC Address: 00:50:56:F8:58:DD (VMware)
Nmap scan report for 192.168.58.128
Host is up.
Nmap scan report for 192.168.58.254
Host is up (0.00037s latency).
MAC Address: 00:50:56:E2:71:C4 (VMware)
Nmap done: 256 IP addresses (4 hosts up) scanned in 4.82 seconds
```

TCP Syn Scan

Before we start, we must know the syn packet.

Basically a syn packet is used to initiate the connection between the two hosts.

The TCP SYN ping sends a SYN packet to the target system and listens for a response. This alternative discovery method is useful for systems that are configured to block standard ICMP pings.

The -PS option performs a TCP SYN ping.

Syntax:nmap –PS targets

```
root@ubuntu:~# nmap -PS [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 12:13 IST
Nmap scan report for [redacted]
Host is up (1.1s latency).
Not shown: 975 closed ports
PORT     STATE    SERVICE
1/tcp    filtered tcpmux
3/tcp    filtered compressnet
4/tcp    filtered unknown
6/tcp    filtered unknown
7/tcp    filtered echo
9/tcp    filtered discard
13/tcp   filtered daytime
17/tcp   filtered qotd
19/tcp   filtered chargen
21/tcp   open     ftp
22/tcp   filtered ssh
25/tcp   open     smtp
26/tcp   open     rsftp
53/tcp   open     domain
80/tcp   open     http
110/tcp  open     pop3
143/tcp  open     imap
443/tcp  open     https
465/tcp  open     smtps
514/tcp  filtered shell
587/tcp  open     submission
993/tcp  open     imaps
995/tcp  open     pop3s
2222/tcp open     EtherNet/IP-1
3306/tcp open     mysql
```

The default port is port80. You can also specify other ports like –PS22, 23, 25, 443.

TCP Ack Ping Scan

This type of scan will only scan of Acknowledgement(ACK) packet.

The -PA performs a TCP ACK ping on the specified target.

The -PA option causes Nmap to send TCP ACK packets to the specified hosts.

Syntax:nmap –PA target

```
File  Edit  Tabs  Help
root@ubuntu:~# nmap -PA

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 12:23 IST
Nmap scan report for
Host is up (1.1s latency).
Not shown: 978 closed ports
PORT     STATE    SERVICE
1/tcp    filtered tcpmux
3/tcp    filtered compressnet
4/tcp    filtered unknown
7/tcp    filtered echo
13/tcp   filtered daytime
17/tcp   filtered qotd
19/tcp   filtered chargen
21/tcp   open     ftp
25/tcp   open     smtp
26/tcp   open     rsftp
53/tcp   open     domain
80/tcp   open     http
110/tcp  open     pop3
143/tcp  open     imap
443/tcp  open     https
465/tcp  open     smtps
514/tcp  filtered shell
587/tcp  open     submission
993/tcp  open     imaps
995/tcp  open     pop3s
2222/tcp open     EtherNet/IP-1
```

This method attempts to discover hosts by responding to TCP connections that are nonexistent in an attempt to solicit a response from the target. Like other ping options, it is useful in situations where standard ICMP pings are blocked.

UDP Ping scan

The –PU scan only on udp ping scans on the target. This type of scan sends udp packets to get a response.

Syntax:nmap –PU target

```
root@ubuntu:~# nmap -PU 192.168.58.128

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 12:36 IST
Nmap scan report for 192.168.58.128
Host is up (0.000016s latency).
All 1000 scanned ports on 192.168.58.128 are closed

Nmap done: 1 IP address (1 host up) scanned in 0.43 seconds
root@ubuntu:~# nmap -PU22,80,25,443 192.168.58.128

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 12:36 IST
Nmap scan report for 192.168.58.128
Host is up (0.000015s latency).
All 1000 scanned ports on 192.168.58.128 are closed

Nmap done: 1 IP address (1 host up) scanned in 0.40 seconds
```

You can also specify the port number for scanning, like –PU 22, 80, 25, etc. In the above picture, the target is my LAN's IP, which doesn't have any UDP services.

Sctp init ping

The -PY parameter instructs Nmap to perform an SCTP INIT ping. This option sends an SCTP packet containing a minimal INIT chunk. This discovery method attempts to locate hosts using the Stream Control Transmission Protocol (SCTP). SCTP is typically used on systems for IP based telephony.

Syntax:nmap –PY target

```
root@ubuntu:~# nmap

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:21 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.61 seconds
root@ubuntu:~# nmap -PY -Pn

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:22 IST
Nmap scan report for
Host is up (0.48s latency).
Other addresses for
rDNS record for
Not shown: 998 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
```

In the picture, though there is no sctp services on the machine, we have to use the –pn option for discovery.

ICMP Echo ping

The -PE option performs an ICMP (Internet Control Message Protocol) echo ping on the specified system.

Syntax:nmap –PE target

```
root@ubuntu:~# nmap -PE [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:47 IST
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
Nmap scan report for [redacted]
Host is up (1.1s latency).
Not shown: 975 closed ports
PORT     STATE    SERVICE
1/tcp    filtered tcpmux
3/tcp    filtered compressnet
4/tcp    filtered unknown
6/tcp    filtered unknown
7/tcp    filtered echo
9/tcp    filtered discard
13/tcp   filtered daytime
17/tcp   filtered qotd
19/tcp   filtered chargen
21/tcp   open     ftp
22/tcp   filtered ssh
25/tcp   open     smtp
26/tcp   open     rsftp
53/tcp   open     domain
80/tcp   open     http
110/tcp  open     pop3
143/tcp  open     imap
443/tcp  open     https
465/tcp  open     smtps
514/tcp  filtered shell
587/tcp  open     submission
993/tcp  open     imaps
995/tcp  open     pop3s
```

This type of discovery works best on local networks where ICMP packets can be transmitted with few restrictions.

ICMP Timestamp ping

The -PP option performs an ICMP timestamp ping.

```
root@ubuntu:~# nmap -PP -Pn [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:51 IST
Nmap scan report for [redacted]
Host is up (1.5s latency).
Not shown: 975 closed ports
PORT      STATE    SERVICE
1/tcp     filtered tcpmux
3/tcp     filtered compressnet
4/tcp     filtered unknown
6/tcp     filtered unknown
7/tcp     filtered echo
9/tcp     filtered discard
13/tcp    filtered daytime
17/tcp    filtered qotd
19/tcp    filtered chargen
21/tcp    open     ftp
22/tcp    filtered ssh
25/tcp    open     smtp
26/tcp    open     rsftp
53/tcp    open     domain
80/tcp    open     http
110/tcp   open     pop3
143/tcp   open     imap
443/tcp   open     https
465/tcp   open     smtps
514/tcp   filtered shell
587/tcp   open     submission
993/tcp   open     imaps
995/tcp   open     pop3s
2222/tcp  open     EtherNet/IP-1
3306/tcp  open     mysql
```

ICMP Address mask ping

The -PM option performs an ICMP address mask ping.

Syntax:nmap –PM target

```
root@ubuntu:~# nmap -PM [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:48 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try
Nmap done: 1 IP address (0 hosts up) scanned in 2.66 seconds
root@ubuntu:~# nmap -PP [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:48 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try
Nmap done: 1 IP address (0 hosts up) scanned in 2.36 seconds
root@ubuntu:~# nmap -PM [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:50 IST
Nmap scan report for 192.168.58.128
Host is up (0.000017s latency).
All 1000 scanned ports on 192.168.58.128 are closed

Nmap done: 1 IP address (1 host up) scanned in 0.45 seconds
```

This unconventional ICMP query (similar to the -PP option) attempts to ping the specified host using alternative ICMP registers. This type of ping can occasionally sneak past a firewall that is configured to block standard echo requests.

IP Protocol Ping

The -PO option performs an IP protocol ping.

Syntax:nmap –PO protocol target

```
root@ubuntu:~# nmap -PO [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:52 IST
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
Nmap scan report for [redacted]
Host is up (1.3s latency).
Not shown: 975 closed ports
PORT      STATE    SERVICE
1/tcp     filtered tcpmux
3/tcp     filtered compressnet
4/tcp     filtered unknown
6/tcp     filtered unknown
7/tcp     filtered echo
9/tcp     filtered discard
13/tcp    filtered daytime
17/tcp    filtered qotd
19/tcp    filtered chargen
21/tcp    open     ftp
22/tcp    filtered ssh
25/tcp    open     smtp
26/tcp    open     rsftp
53/tcp    open     domain
80/tcp    open     http
110/tcp   open     pop3
143/tcp   open     imap
443/tcp   open     https
465/tcp   open     smtps
514/tcp   filtered shell
587/tcp   open     submission
993/tcp   open     imaps
995/tcp   open     pop3s
2222/tcp  open     EtherNet/IP-1
```

An IP protocol ping sends packets with the specified protocol to the target. If no protocols are specified, the default protocols 1 (ICMP), 2 (IGMP), and 4 (IP-in-IP) are used.

ARP ping

The –PR option is used to perform an arp ping scan. The -PR option instructs Nmap to perform an ARP (Address Resolution Protocol) ping on the specified target.

SYTAX: nmap –PR target

```
root@ubuntu:~# nmap -PR [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:53 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.65 seconds
root@ubuntu:~# nmap -PR [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:55 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.09 seconds
root@ubuntu:~# nmap -PR [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:55 IST
Nmap scan report for 192.168.58.128
Host is up (0.000015s latency).
All 1000 scanned ports on [redacted]e closed

Nmap done: 1 IP address (1 host up) scanned in 0.41 seconds
```

The -PR option is automatically implied when scanning the local network. This type of discovery is much faster than the other ping methods.

Traceroute

The –traceroute parameter can be use to trace the network path to the specified host.

Syntax: nmap –traceroute target

```
Not shown: 975 closed ports
PORT      STATE     SERVICE
1/tcp     filtered  tcpmux
3/tcp     filtered  compressnet
4/tcp     filtered  unknown
6/tcp     filtered  unknown
7/tcp     filtered  echo
9/tcp     filtered  discard
13/tcp    filtered  daytime
17/tcp    filtered  qotd
19/tcp    filtered  chargen
21/tcp    open      ftp
22/tcp    filtered  ssh
25/tcp    open      smtp
26/tcp    open      rsftp
53/tcp    open      domain
80/tcp    open      http
110/tcp   open      pop3
143/tcp   open      imap
443/tcp   open      https
465/tcp   open      smtps
514/tcp   filtered  shell
587/tcp   open      submission
993/tcp   open      imaps
995/tcp   open      pop3s
2222/tcp  open      EtherNet/IP-1
3306/tcp  open      mysql

TRACEROUTE (using port 80/tcp)
HOP RTT      ADDRESS
1   0.09 ms  192.168.58.2
2   0.09 ms  [redacted]
```

Force Reverse DNS Resolution

The -R parameter instructs Nmap to always perform a reverse DNS resolution on the target IP address.

Syntax: nmap –R target

```
root@ubuntu:~# nmap -r [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:59 IST
Nmap scan report for [redacted]
Host is up (1.3s latency).
Not shown: 975 closed ports
PORT      STATE     SERVICE
1/tcp     filtered  tcpmux
3/tcp     filtered  compressnet
4/tcp     filtered  unknown
5/tcp     filtered  unknown
7/tcp     filtered  echo
9/tcp     filtered  discard
13/tcp    filtered  daytime
17/tcp    filtered  qotd
19/tcp    filtered  chargen
21/tcp    open      ftp
22/tcp    filtered  ssh
25/tcp    open      smtp
26/tcp    open      rsftp
53/tcp    open      domain
80/tcp    open      http
110/tcp   open      pop3
143/tcp   open      imap
443/tcp   open      https
465/tcp   open      smtps
514/tcp   filtered  shell
587/tcp   open      submission
993/tcp   open      imaps
995/tcp   open      pop3s
2222/tcp  open      EtherNet/IP-1
3306/tcp  open      mysql
```

The -R option is useful when performing reconnaissance on a block of IP addresses, as Nmap will try to resolve the reverse DNS information of every IP address.

Disable Reverse DNS Resolution

The -n parameter is used to disable reverse DNS lookups.

Syntax:nmap –n target

```
root@ubuntu:~# nmap -n [redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 14:59 IST
Nmap scan report for [redacted]
Host is up (1.1s latency).
Not shown: 975 closed ports
PORT      STATE     SERVICE
1/tcp     filtered  tcpmux
3/tcp     filtered  compressnet
4/tcp     filtered  unknown
5/tcp     filtered  unknown
7/tcp     filtered  echo
9/tcp     filtered  discard
13/tcp    filtered  daytime
17/tcp    filtered  qotd
19/tcp    filtered  chargen
21/tcp    open      ftp
22/tcp    filtered  ssh
25/tcp    open      smtp
26/tcp    open      rsftp
53/tcp    open      domain
80/tcp    open      http
110/tcp   open      pop3
143/tcp   open      imap
443/tcp   open      https
465/tcp   open      smtps
514/tcp   filtered  shell
587/tcp   open      submission
993/tcp   open      imaps
995/tcp   open      pop3s
2222/tcp  open      EtherNet/IP-1
3306/tcp  open      mysql
```

Reverse DNS can significantly slow an Nmap scan. Using the -n option greatly reduces scanning times – especially when scanning a large number of hosts. This option is useful if you don't care about the DNS information for the target system and prefer to perform a scan which produces faster results.

Alternative DNS lookup method

The –system-dns option instructs Nmap to use the host system's DNS resolver instead of its own internal method.

Syntax:nmap –system-dns target

```
root@ubuntu:~# nmap --system-dns [redacted]
Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 15:06 IST
Nmap scan report for [redacted]
Host is up (1.3s latency).
Not shown: 975 closed ports
PORT      STATE     SERVICE
1/tcp     filtered  tcpmux
3/tcp     filtered  compressnet
4/tcp     filtered  unknown
5/tcp     filtered  unknown
7/tcp     filtered  echo
9/tcp     filtered  discard
13/tcp    filtered  daytime
17/tcp    filtered  qotd
19/tcp    filtered  chargen
21/tcp    open      ftp
22/tcp    filtered  ssh
25/tcp    open      smtp
26/tcp    open      rsftp
53/tcp    open      domain
80/tcp    open      http
110/tcp   open      pop3
143/tcp   open      imap
443/tcp   open      https
465/tcp   open      smtps
514/tcp   filtered  shell
587/tcp   open      submission
993/tcp   open      imaps
995/tcp   open      pop3s
1222/tcp  open      EtherNet/IP-1
3306/tcp  open      mysql
```

Manually Specify DNS server

The –dns-servers option is used to manually specify DNS servers to be queried when scanning.

Syntax: nmap –dns-servers server1 server2 target

```
root@ubuntu:~# nmap --dns-servers 208.[redacted] 208.[redacted] www.[redacted]

Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 15:08 IST
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
Nmap scan report for [redacted]
Host is up (1.3s latency).
Not shown: 975 closed ports
PORT      STATE     SERVICE
1/tcp     filtered  tcpmux
3/tcp     filtered  compressnet
4/tcp     filtered  unknown
5/tcp     filtered  unknown
7/tcp     filtered  echo
9/tcp     filtered  discard
13/tcp    filtered  daytime
17/tcp    filtered  qotd
19/tcp    filtered  chargen
21/tcp    open      ftp
22/tcp    filtered  ssh
25/tcp    open      smtp
26/tcp    open      rsftp
53/tcp    open      domain
80/tcp    open      http
110/tcp   open      pop3
143/tcp   open      imap
443/tcp   open      https
465/tcp   open      smtps
514/tcp   filtered  shell
587/tcp   open      submission
993/tcp   open      imaps
995/tcp   open      pop3s
```

The –dns-servers option allows you to specify one or more alternative servers for Nmap to query. This can be useful for systems that do not have DNS configured or if you want to prevent your scan lookups from appearing in your locally configured DNS server's log file.

List Scan

The -sL option will display a list and performs a reverse DNS lookup of the specified IP addresses.

Syntax:nmap –sL target

```
root@ubuntu:~# nmap -sL
Starting Nmap 6.26SVN ( http://nmap.org ) at 2014-03-22 15:14 I
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
Nmap scan report for
```

In the next installment, I will discuss how to discover services, host, and banners using different methods, and will also discuss how to find firewalls and how to evade them using NSE by Nmap, and how to write your own Nmap script engine. The most important part of Nmap is knowing how to find vulnerability and try to exploit them. Stay tuned.

Reference

http://nmap.org/

NEXT: NMAP CHEAT SHEET: FROM DISCOVERY TO EXPLOITS, PART 2: ADVANCE PORT SCANNING WITH NMAP AND CUSTOM IDLE SCAN

AUTHOR

Revers3r

Revers3r is a Information Security Researcher with considerable experience in Web Application Security, Vulnerability Assessment, Penetration Testing. He is also well-versed in Reverse Engineering, Malware Analysis. He's been a contributor to international magazines like Hakin9, Pentest, and E-Forensics. In his free time, he's contributed to the Response Disclosure Program. website: www.vulnerableghost.com