Scanning in hacking:

***Definition:* Scanning refers to the process of gathering information about a target system, network, or application by probing it with various techniques.**

***Purpose:* To identify potential vulnerabilities, open ports, active hosts, and other useful information that can be used to plan and execute an attack.**

Common scanning activities:

1. Port scanning
2. Network scanning
3. Vulnerability scanning

Would you like more information on scanning techniques or related security topics?

Basic Techniques of Scanning:

1. Port Scanning: Identifying open ports and services on a target system.

2. OS Detection: Identifying the operating system and its version on a target system.

3. Service Version Detection: Identifying the version of services running on a target system.

In this chapter we will discuss various methods for enumerating and scanning a target or goal to gain as much information about the alive targets on a network as possible. This is also part of the information gathering phase, which, as I had mentioned, is key to a successful pentest. This chapter is very essential and is a building block for penetration testers, because later in Chapter 7 you will realize how the information we have gathered in this chapter helps us to compromise targets.

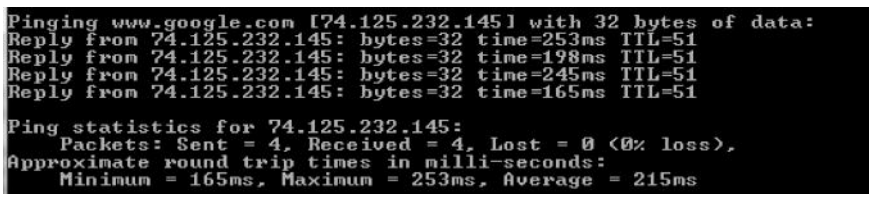
The main goal of this chapter is to learn the following:

- Host discovery
- Scanning for open ports
- Service and version detection
- OS detection
- Bypassing firewalls

We will use a variety of tools in demonstrating these tasks.

Host Discovery

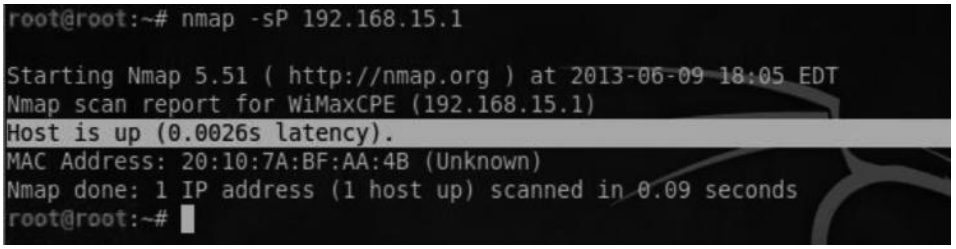
The first step of a network pentest most times would be to know what targets are alive. Since it is not possible to penetrate a target that is not alive without physical access, we always look for alive targets. We can use a variety of methods and tools for discovering alive targets. One of the most common methods is to use icmp requests, that is, ping requests to check if the system is alive or not.



Aswehavegotareply,itmeansthatourtargetisalive.Wecanalsousethe–sPflaginnmap inordertocheckifthetargetisaliveornot.Besides,wecanspecifynetworkrangestoscan;this would make our work simpler.

Command:

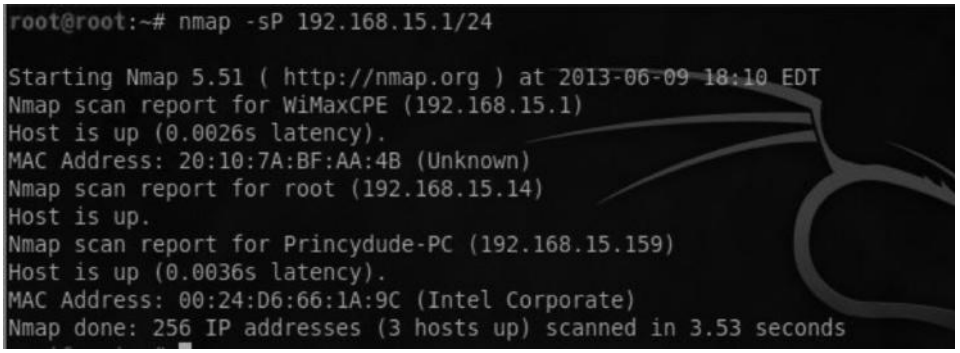
```
nmap-sP<targetHost>
```



Wecanalsoscannetworkrangeswithnmaponthegivennetwork.Hereisthecommandto scan a host range from nmap:

```
nmap-sP192.168.15.1/24
```

/24isaCIDRnotation;itwillscanallthehostsintherange192.168.15.1to192.168.15.255and return those that are up.



Asyoucanseefromthescreenshot,thewholerangewasscannedforalivesystems,andthree live systems were found on the network.

Nowadays,duetotheimplementationofIDS,IPS,Firewalls,andothermoderndefenseson thenetwork,identifyingalivehostscanbeabittrivial.Networkadministratorscommonlyblock icmprequests,whichmeansthatevenifthetargetwerealive,wewouldnotbeabletofigureitout. Thus,wecanuseothertypesofprotocolssuchastcpandudpinordertofigureoutifthetarget isaliveornot,sinceanormaltcporudpconnectmaynotlooksuspicioustofirewallsandother intrusion detection/prevention devices.

In your penetration testing engagments you will find a lot of scenario's where you'd encounter against these modern security defenses. For demonstration purposes, we will use a website named didx.net.Theadministratorhasblockedicmpreqeststoitswebserverbyusing IPtables. Anormal ping request leads us to the following output:

```
root@root:~# nping didx.net

Starting Nping 0.5.51 ( http://nmap.org/nping ) at 2013-06-09 18:19 EDT
SENT (0.0696s) ICMP 192.168.15.14 > 174.121.60.75 Echo request (type=8/code=0) t
tl=64 id=60064 iplen=28
SENT (1.0702s) ICMP 192.168.15.14 > 174.121.60.75 Echo request (type=8/code=0) t
tl=64 id=60064 iplen=28
SENT (2.0729s) ICMP 192.168.15.14 > 174.121.60.75 Echo request (type=8/code=0) t
tl=64 id=60064 iplen=28
SENT (3.0800s) ICMP 192.168.15.14 > 174.121.60.75 Echo request (type=8/code=0) t
tl=64 id=60064 iplen=28
SENT (4.0819s) ICMP 192.168.15.14 > 174.121.60.75 Echo request (type=8/code=0) t
tl=64 id=60064 iplen=28

Max rtt: N/A | Min rtt: N/A | Avg rtt: N/A
Raw packets sent: 5 (140B) | Rcvd: 0 (0B) | Lost: 5 (100.00%)
Tx time: 4.01316s | Tx bytes/s: 34.89 | Tx pkts/s: 1.25
Rx time: 5.01489s | Rx bytes/s: 0.00 | Rx pkts/s: 0.00
Nping done: 1 IP address pinged in 5.09 seconds
```

Isentsomeicmprequestswithnping;youcanclearlyseethatthetargetisnotalive.However, let'strysendingsometcppackets.Bylookingatthedocumentationandusageguideofnping,we canseethatitalsoallowshostdiscoveryviatcpandudp.

```
root@root:~# nping
Nping 0.5.51 ( http://nmap.org/nping )
Usage: nping [Probe mode] [Options] {target specification}

TARGET SPECIFICATION:
  Targets may be specified as hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
PROBE MODES:
  --tcp-connect                    : Unprivileged TCP connect probe mode.
  --tcp                            : TCP probe mode.
  --udp                            : UDP probe mode.
  --icmp                           : ICMP probe mode.
  --arp                            : ARP/RARP probe mode.
  --tr, --traceroute               : Traceroute mode (can only be used with
                                     TCP/UDP/ICMP modes).
```

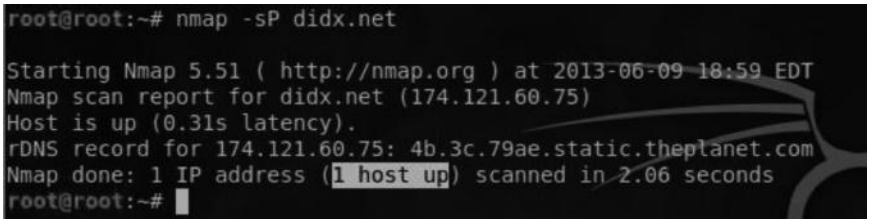
So,Ienteredthefollowingcommandinordertoperformasimpletcp-basedhostdiscovery.

```
nping--tcpdidx.net
```

```
root@root:~# nping --tcp didx.net

Starting Nping 0.5.51 ( http://nmap.org/nping ) at 2013-06-09 18:27 EDT
SENT (0.0156s) TCP 192.168.15.14:33333 > 174.121.60.75:80 S ttl=64 id=27847 iple
n=40  seq=2139527381 win=1480
RCVD (0.3675s) TCP 174.121.60.75:80 > 192.168.15.14:33333 SA ttl=47 id=0 iplen=4
4  seq=1599555088 win=5840 <mss 1360>
SENT (1.0161s) TCP 192.168.15.14:33333 > 174.121.60.75:80 S ttl=64 id=27847 iple
n=40  seq=2139527381 win=1480
RCVD (1.4301s) TCP 174.121.60.75:80 > 192.168.15.14:33333 SA ttl=47 id=0 iplen=4
4  seq=1616119072 win=5840 <mss 1360>
SENT (2.0177s) TCP 192.168.15.14:33333 > 174.121.60.75:80 S ttl=64 id=27847 iple
n=40  seq=2139527381 win=1480
RCVD (2.4269s) TCP 174.121.60.75:80 > 192.168.15.14:33333 SA ttl=47 id=0 iplen=4
4  seq=1631276166 win=5840 <mss 1360>
^C
Max rtt: 413.650ms | Min rtt: 351.629ms | Avg rtt: 391.333ms
Raw packets sent: 3 (120B) | Rcvd: 3 (138B) | Lost: 0 (0.00%)
Tx time: 2.67633s | Tx bytes/s: 44.84 | Tx pkts/s: 1.12
Rx time: 2.67633s | Rx bytes/s: 51.56 | Rx pkts/s: 1.12
Nping done: 1 IP address pinged in 2.69 seconds
```

The output shows 0% packet loss with three packets sent and received, indicating that the targetisindeedalive.Wecanalsouseudptoperformhostdiscovery;whatoptionyouwouldlike to use is up to you.

Alternatively,wecanalsousethe–sPflagquerytoaccomplishthistask,becausewhenyou specifythe–sPflagquerywithnmap,itsendsnotonlyicmpechorequestsbutalsoTCPSYNto port80and443.Therefore,itwillalsoshowthehostasuporinotherwordsalive.



ScanningforOpenPortsandServices

Oncewehavesuccessfullyscannedthenumberoflivehostsonanetwork,weattempttofindopen ports and the services associated with them on a network. Port scanning is the process of discoveringTCPandUDPopenportsonthetargethostornetwork.Openportsrevealtheservicesthat are running upon the network. We perform port scanning in order to look for potential entry points into the systems.

Oneofthemostchallengingtaskswithportscanningistoevadefirewallsandintrusiondetectionandpreventionmechanisms.Ourgoalistomakeourscanlessnoisy.Inthischapter,wewill alsodiscusssomestealthscanningtechniquestomakeyourscanslessnoisy.

Thereexistmanytoolssuchasnetcat,hping2,andUnicornscanforscanningopenports,but nmapisourultimatechoice.However,wewilllookatsomeoftheguiandcommandlinetools too.Butourmainfocuswillbeonnmapasit'soneofthemostcomprehensiveportscanningtools.

TypesofPortScanning

Portscanningisprimarilydividedintotwomaincategories:TCPscanningandUDPscanning. NmapsupportsawidevarietyofscanningmethodssuchastheTCPsynscanandtheTCPconnectscan,andwewilldiscusssomeofthemhereingreatdetail.

Nmapisverysimpletouse;thebasiccommandlineformatfornmapisasfollows:

```
nmap<ScanType><Option><TargetSpecification>
```

Asimpleportcanbelaunchedbythefollowingcommand:

```
nmap<targetIpAddress>
```

Thiswouldreturnustheportsthatareopeneduponthetargethost.

Wecanalsoscanarangebyeitherusingthe CIDR notationthatweusedearlierinthehost discoveryprocessorusingthe*sign.

Command:

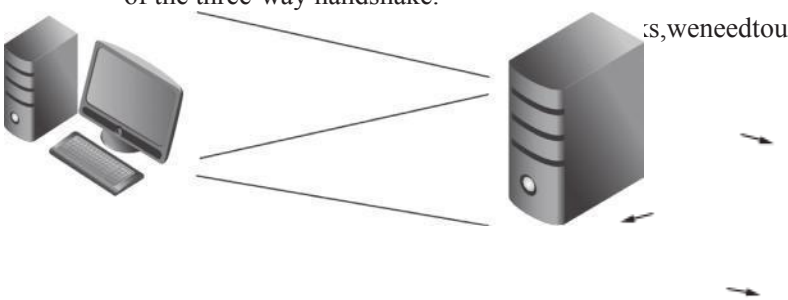
```
nmap192.168.15.*
```

```
root@root:~# nmap 192.168.15.*

Starting Nmap 5.51 ( http://nmap.org ) at 2013-06-09 19:09 EDT
Nmap scan report for WiMaxCPE (192.168.15.1)
Host is up (0.017s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
443/tcp   open  https
49152/tcp open  unknown
50003/tcp open  unknown
MAC Address: 20:10:7A:BF:AA:4B (Unknown)

Nmap scan report for root (192.168.15.14)
Host is up (0.000010s latency).
Not shown: 999 closed ports
PORT    STATE SERVICE
111/tcp open  rpcbind

Nmap done: 256 IP addresses (2 hosts up) scanned in 11.05 seconds
```

Thiswouldscanthewholerange192.168.15.1–255andreturnopenports.Also,youcansee that nmap returns the service associated with each port.

UnderstandingtheTCPThree-WayHandshake

Thetransmissioncontrolprotocol(TCP)wasmadeforreliablecommunication.Itisusedfora widevarietyofprotocolsontheInternetandcontributestowardreliablecommunicationwiththe help of the three-way handshake.



:s,weneedtounderstandhowtheTCPthree- way

- Thefirsthostsendsa SYNpackettothesecondhost.
- Thesecondhostrespondswitha SYN/ACKpacket;itindicatesthatthepacketwasreceived.
- Thefirsthostcompletestheconnectionbysendinganacknowledgmentpacket.

TCPFlags

SYN—Initiatesaconnection.*ACK*—Acknowledge sthatthepacketwasreceived. *RST*—Resets the connections between two hosts. *FIN*—Finishes the connection.

Therearemanyotherflags,andIwouldrecommendyoutospendsometimereading**rfc793**, theTCPprotocolspecification.Icannotemphasizeenoughtheimportanceofunderstandingthe TCP IP; it will help you a lot.

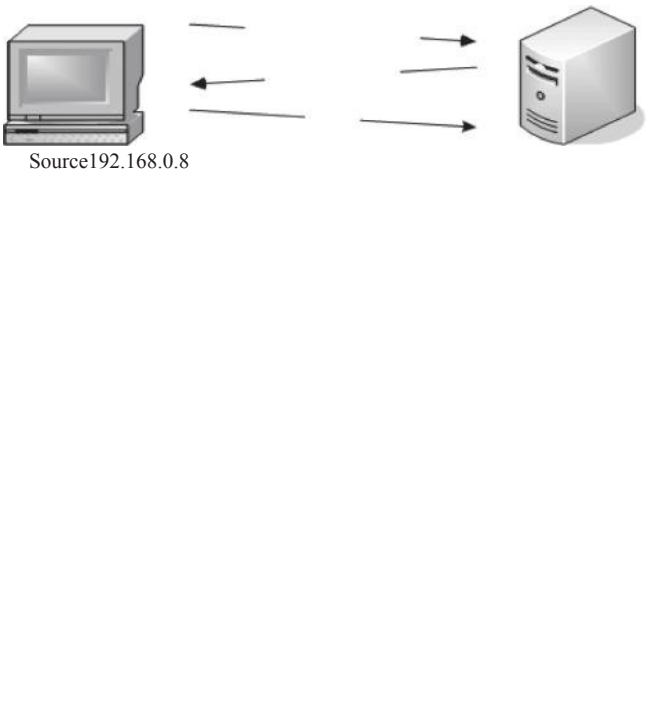
PortStatusTypes

Withnmapyouwouldseeoneoffourportstatustypes:

Open—Itmeansthattheportisaccessibleandanapplicationislisteningon it.
Closed—Itmeansthattheportisinaccessibleandnoapplicationislisteningonit.
Filtered—Itmeansthatnmapisnotabletofigureoutiftheportisopenorclosed,asthepackets arebeingfiltered,whichprobablymeansthatthemachineisbehindafirewall.
Unfiltered—Itmeansthattheportsareaccessiblebynmapbutitisnotpossibletofigureoutif they are open or closed.

TCPSYNScan

TheTCPSYNscanisthedefaultscanthatrunsagainstthetargetmachine.Itisthefastestscan. Youcantweakittomakeitevenfasterbyusingthe–noption,whichwouldtellthenmaptoskip the DNS resolution.



ThisdiagramillustrateshowaTCPSYNscanworks:

- ThesourcemachinesendsaSYNpackettoport80inthedestinationmachine.
- IfthemachinerespondswithSYN/ACKpacket,Nmapwouldknowthattheparticularport is *open* on the target machine.

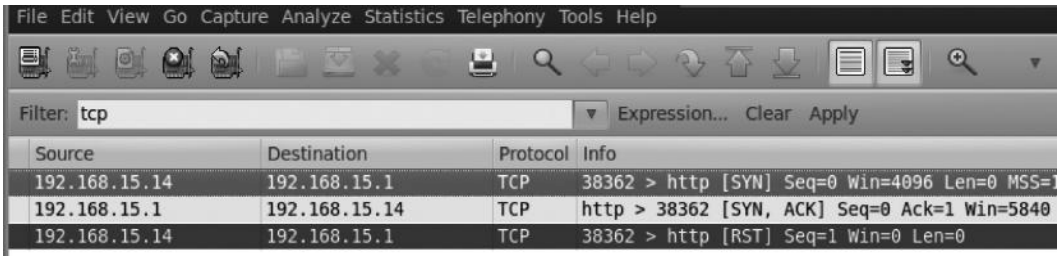
- TheoperatingsystemwouldsendaRST(Reset)packetinordertoclosetheconnection, since we already know that the port is open.
- However, if there is no response from the destination after sending the SYN packet, the nmap would know that the port is *filtered*.
- If you send a SYN packet and the target machine sends a RST packet, then nmap would know that the port is *closed*.

Command:Thecommand/syntaxfortheTCPSYNscanisasfollows:

```
nmap-sS<targetIP>
```

```
root@root:~# nmap -sS -n 192.168.15.1 -p 80

Starting Nmap 5.51 ( http://nmap.org ) at 2013-06-09 20:49 EDT
Nmap scan report for 192.168.15.1
Host is up (0.0024s latency).
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 20:10:7A:BF:AA:4B (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.24 seconds
```

Fromthispicture,youcanseethatIhavespecifiedtwoadditionalparameters(–nand–p). The–nparametertellsthenmapnottoperformthenameresolution;thisiscommonlyusedto increasethespeedofthescan.The–pparameterisusedtospecifytheportstoscan,whichinthis case is port 80.



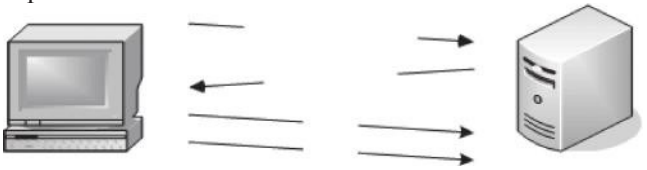
Ialsoran*Wireshark*(anetworkanalysistool)whileperformingthisscantorecordthebehavior of the packets. The output was what we expected.

Asyoucanseefromthefirstlinethesource192.168.15.14sendsaSYNpackettothedestination192.168.15.1.Thedestinationrespondswithas SYN,ACKinthesecondline.Thesource 192.168.15.14thensendsaRSTpackettoclosetheconnection,thusdisplayingthebehaviordiscussedearlier.Ihavealsousedthe“TCP”filtertofilterouttcpprotocol–relatedrequests.

Thepositivesideofthisscanisthatitisprettyfast;itsdownsideisthatitisoftendetectedby IDS,IPS,andfirewalls.Wewilltalkaboutsometechniquestoperformnoiselessscanslaterin this chapter.

TCPConnectScan

The TCP connect scan is similar to the SYN scan, with a slight difference in that it completes thethree-wayhandshake.TheTCPconnectscanbecomesthedefaultscanifthe SYNscanisnot supportedbythemachine.Acommonreasonforthatcouldbethatthemachineisnotprivileged to create its own RAW packet.



Source192.168.0.8

Destinat

ion192.168.0.10

Thisdiagramillustratesthatit'sworking:

- ThesourcemachinesendsaSYNpacketatPort80.
- ThedestinationmachinerespondswithaSYN/ACK.
- ThesourcemachinethensendsanACKpackettocompletethethree-wayhandshake.
- ThesourcemachinefinallysendstheRSTpacketinordertoclosetheconnection.

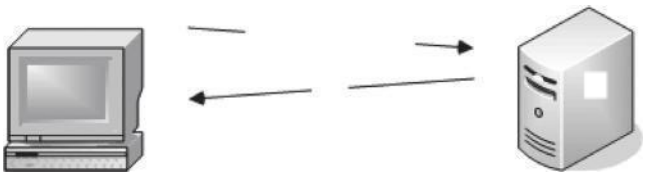
The TCP connect scan can be accomplished by specifying an additional **–sC** parameter with nmap.

Hereisanexample:

```
root@root:~# nmap -sC 192.168.15.1

Starting Nmap 5.51 ( http://nmap.org ) at 2013-06-09 21:04 EDT
Nmap scan report for WiMaxCPE (192.168.15.1)
Host is up (0.0052s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
```

NULL,FIN,andXMASScans

NULL,FIN,andxmasscansaresimilartoeachother.Themajoradvantageofusingthesescans forpentestisthatmanytimestheygetpastfirewallsandIDSandcanbereallybeneficialagainst Unix-basedOSasallthreeofthesescansdonotworkagainstWindows-basedoperatingsystems, becausetheysendaresetpacketregardlessofwhethertheportisopenorclosed.Theseconddis-advantageisthatitcannotbeexactlydeterminediftheportisopenorfiltered.Thisleavesusto manually verify it with other scan types.



Source192.168.0.8

Destination192.168.

0 7
.

A null scan is accomplished by sending no flags/bits inside the TCP header. If no response comes,itmeansthattheportis*open*;ifa*RST*packetisreceived,itmeansthattheportis*closed* or *filtered*.

Command:

```
nmap-sN<targetIpAddress>
```

FINScan

Destination192.168.0.7

AFINflagisusedtocloseacurrentlyopensession.InaFINscanthesendersendsaFINflag to the target machine: if no response comes from the target machine, it means that the port is *open*;ifthetargetmachinerespondswitha*RST*,itmeansthattheportis*closed*.

Command:

```
nmap-sF<targetIpAddress>
```

Source192.168.0.8

Destination192.168.

0 7
.

The XMAS scan sends a combination of FIN, URG, and PUSH flags to the destination. It lightensthepacketjustlikeaChristmastreeandthatiswhyitiscalledanXMASscan.Itworks just like the FIN and null scans. If there is *no* response, the port is *open*; if the target machine respondswitha*RST*packet,theportis*closed*.

Command:

```
nmap-sX<targetIpAddress>
```

Source69.240.103.51

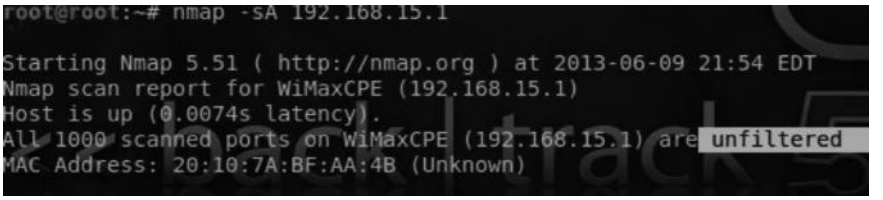
D
e
s
t
i
n
a
t
i
o
n
6
8
.
4
6
.
2
3
4
.
1
6
1

TheTCPACKscanisnotusedforportscanningpurposes.Itiscommonlyusedtodetermine thefirewallandACLrules(accesslist)andwhetherthefirewallisabletokeeptrackofthecon- nections that are being made.

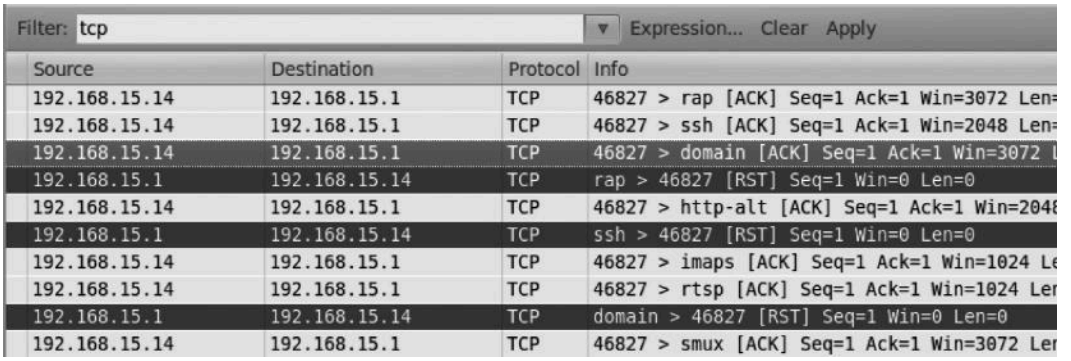
Thewaythisworksisthatthesourcemachinesendsanacknowledge**(ack)**packetinsteadofa synpacket.Ifthefirewallisstateful,itwouldknowthatthetherewasnoSYNpacketbeingsent and will not allow the packet to reach the destination.

Responses

- Ifthereisnoresponse,thismeansthatthefirewallisstatefulandit'sfilteringyourpackets.
- Ifyoureceivearesetpacket,itmeansthatthepacketreachedthedestination.



ThecapturefromwiresharkalsogivesabetterinsightintotheTCPACKscan.



Command:

```
nmap-sA<targetIpAddress>
```

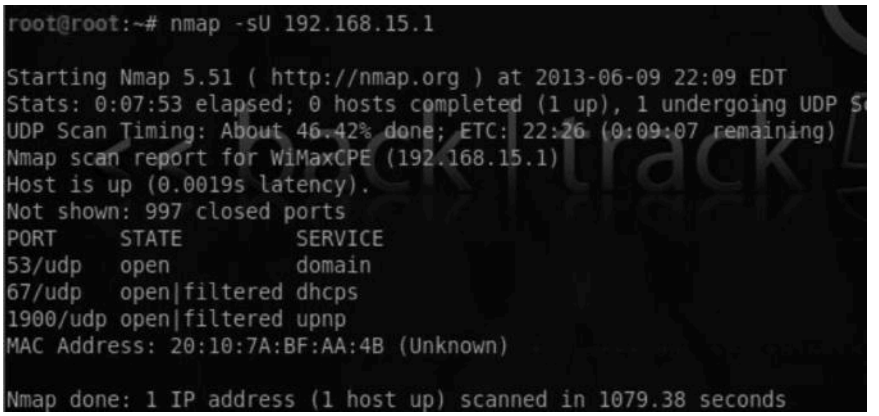
UDPPortScan

UDPstandsfor"userdatagramprotocol";itdoesnotensurethereliabilityofthecommunication andisnotusedforcommunication,wherethedataareveryimportanttous.Therearemanyports thatuseUDP;theUDPportscancanbeusedtodeterminethecommonservicesthatarelistening upon UDP. Some of the popular UDP services are DHCP, SNMAP, and DNS.

TheUDPportscanworksbysendinganemptyUDPheader;anykindofUDPresponsefrom thetargetportwouldrevealthattheportis*open*.Noresponsewouldmeanthateitherthepor tis *open* or it is *filtered*. A closed port is determined on the basis of ICMP error messages; if it responds with"ICMPPortunreachableerror,"thiswouldmeanthattheportisclosed.AnyotherICMP response means that the port is filtered.

Command:

```
nmap-sU<targetIpAddress>
```



AnonymousScanTypes

WediscussedavarietyofscantypesincludingbothTCPandUDP.Wealsodiscussedsomeof the scans that can be used for anonymous scanning; in other words, your host iP would not be revealedatthedestinationwhenyoureperformingportscanning.Thesetypesofscansarevery usefulifyouwishtoremainanonymouswhilescanningyourtarget.Boththescantechniqueswe havediscussedinthischapterrelyspecificallyuponusinganotherhost/servertoperformascan for you.

IDLEScan

The IDLE scan is a very effective and stealthy scanning technique. The idea behind the IDLE scanistointroduceazombietoscananotherhost.Thistechniqueisstealthybecausethevictim hostwouldreceivepacketsfromthezombiehostandnottheattackerhost.Inthisway,thevictim would not be able to figure out where the scan originated.

However,therearesomeprerequisitesforlaunchingtheidlescan,whichareasfollows:

1. FindingagoodcandidatewhoseIPIDsequenceisincrementalandrecordingitsIPID.
2. ThehostshouldbeIDLEonthenetwork.

ScanningforaVulnerableHost

Let'snowtalkaboutscanningforavulnerablehostforthezombiescan.Wecanuseatoolcalled Hping2 for figuring out if a host is a good candidate for an IDLE scan. Hping2 is mainly used forfirewalltestingpurposes;thecreatorofthistoolisalsotheonewhointroducedtheconceptof IDLE scanning.

Command:
Fromyourconsole,justtype

```
hping2-S-r<TargetIP>
```

S—SendingaSYNflag
R—Fortherelativeid

```
root@root:~# hping2 -S -r 192.168.15.211
HPING 192.168.15.211 (eth0 192.168.15.211): S set, 40 headers + 0 data bytes
len=46 ip=192.168.15.211 ttl=128 id=189 sport=0 flags=RA seq=0 win=0 rtt=0.8 ms
len=46 ip=192.168.15.211 ttl=128 id=+1 sport=0 flags=RA seq=1 win=0 rtt=0.9 ms
len=46 ip=192.168.15.211 ttl=128 id=+1 sport=0 flags=RA seq=2 win=0 rtt=0.8 ms
len=46 ip=192.168.15.211 ttl=128 id=+1 sport=0 flags=RA seq=3 win=0 rtt=0.6 ms
len=46 ip=192.168.15.211 ttl=128 id=+1 sport=0 flags=RA seq=4 win=0 rtt=0.6 ms
len=46 ip=192.168.15.211 ttl=128 id=+1 sport=0 flags=RA seq=5 win=0 rtt=0.7 ms
```

Asyoucansee,theidisincrementedby1;thisshowsusthatthehostisapotentialcandidate for becoming our zombie and can be used to perform an IDLE scan.

Alternatively,wecanusethemetasploitauxiliarymoduleforfiguringoutagoodcandidatefor azombie.Inordertousetheauxiliarymodule,wewouldneedtostartupthemetasploitframe- work. We will talk about metasploit in more detail in Chapter 7.

Fromtheshell,type“msfconsole”tofireupmetasploit.Oncemetasploitisstarted,issuethe following command to load the auxiliary module:

```
msf>useauxiliary/scanner/ip/ipidseq
```

Next,youneedtosettheRhostsvalue;youcaneitherspecifyarangeorasingletarget.Hereis an example:

Forasinglehost
SetRHOSTS<TargetIp>

Forarange
SetRHOSTS 192.168.15.1–192.168.15.255

Finally,youneedtoissuethe*run*commandinordertofinishtheprocess.Hereisthescreen- shot of how this would look:

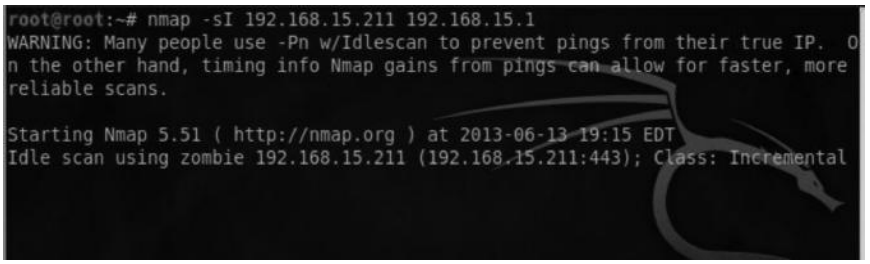
```
88888b.d88b.  .d88b. 888888 8888b. .d8888b 88888b. 888 .d88b. 888888888
888 "888 "88bd8P  Y8b888       "88b88K     888 "88b888d88""88b888888
888  888  88888888888888   .d888888"Y8888b.888  888888888  888888888
888  888  888Y8b.    Y88b. 888  888     X88888 d88P888Y88..88P888Y88b.
888  888  888 "Y8888  "Y888"Y888888 88888P'88888P" 888 "Y88P" 888 "Y888
                                     888
                                     888
                                     888


       =[ metasploit v3.7.0-release [core:3.7 api:1.0]
+ -- --=[ 684 exploits - 355 auxiliary
+ -- --=[ 217 payloads - 27 encoders - 8 nops
       =[ svn r12536 updated 771 days ago (2011.05.04)

Warning: This copy of the Metasploit Framework was last updated 771 days ago.
         We recommend that you update the framework at least every other day.
         For information on updating your copy of Metasploit, please see:
             http://www.metasploit.com/redmine/projects/framework/wiki/Updating

msf > use auxiliary/scanner/ip/ipidseq
msf auxiliary(ipidseq) > set rhosts 192.168.15.211
rhosts => 192.168.15.211
msf auxiliary(ipidseq) > run
```

PerforminganIDLEScanwithNMAP

Nowthatwehaveidentifiedagoodcandidateforourzombie,let'stryperforminganIDLEscan withnmap.Theidlescancanbesimplyperformedbyspecifyingthe–sIparameterwithnmap, followedbytheiPofourzombiehostandthetargetthatwewanttoscanagainst.

Command:

```
nmap-sI<IPAddressOfZombie><IPAddressOfTheTarget>
```



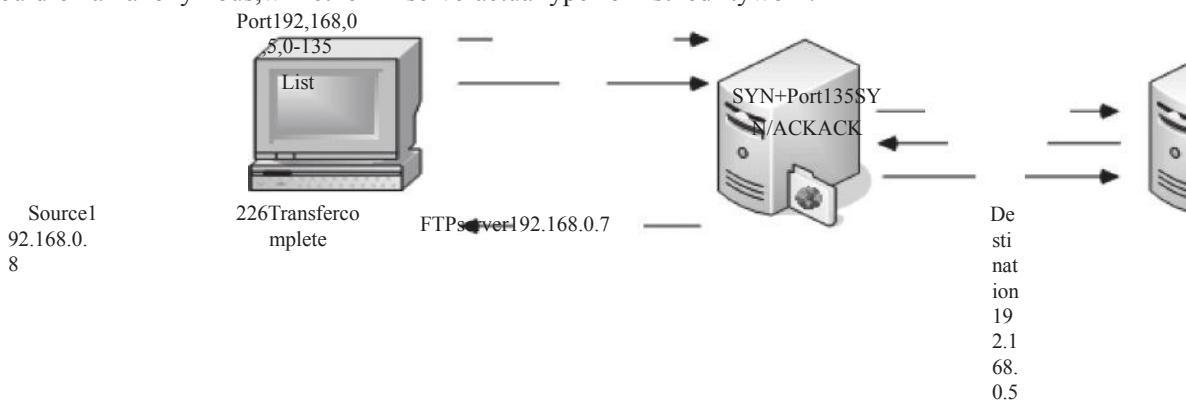
Also,onethingthatwouldbeworthmentioninghereisthatwhileperforminganIDLEscan, youshouldalsousethe–pNoption.Thiswillpreventnmapfromsendinganinitialpacketfrom yourrealIPtothetargethost.Hereisanotherexamplefromthenmapbook,whichshowstheidle scanbeingperformedonriaa.combyusingahostthatbelongstoadobe.com.

```
# nmap -Pn -p- -sI kiosk.adobe.com www.riaa.com

Starting Nmap ( http://nmap.org )
Idlescan using zombie kiosk.adobe.com (192.150.13.111:80); Class: Incremental
Nmap scan report for 208.225.90.120
(The 65522 ports scanned but not shown below are in state: closed)
Port       State       Service
21/tcp     open        ftp
25/tcp     open        smtp
80/tcp     open        http
111/tcp    open        sunrpc
135/tcp    open        loc-srv
443/tcp    open        https
1027/tcp   open        IIS
1030/tcp   open        iad1
2306/tcp   open        unknown
5631/tcp   open        pcanywheredata
7937/tcp   open        unknown
7938/tcp   open        unknown
36890/tcp  open        unknown

Nmap done: 1 IP address (1 host up) scanned in 2594.47 seconds
```

TCPFTPBounceScan

ThistypeofscanexploitsavulnerabilityinsideoldFTPserversthatsupportaproxy-basedFTP connection.Thisvulnerabilitytakesadvantageofafeaturethatexistedinsideoldftpservers,which allowedtheuserstoconnecttotheFTPserverandsendfilestoathird-partyserver.Thiswasdone

byaskingtheservertosendafiletoaspecificportonthetargetmachine.Thiswaytheattacker couldremainanonymous,whiletheFTPserveractuallyperformsthedirtywork.



However,Iwouldliketomentionthatthisbugwaspatchedinsidemostoftheservers duringthe1990swhenitwasfirstfound,andalmostallftpserversarenowadaysconfiguredto blockportcommands,butyoucanstillfindavulnerableFTPserverifyoulooklongenough.

Nmapgivesyoutheflexibilitytotestifatargetserverisvulnerabletothebounce attack or not.

Command:

```
nmap-b<targetFTPServer>
```

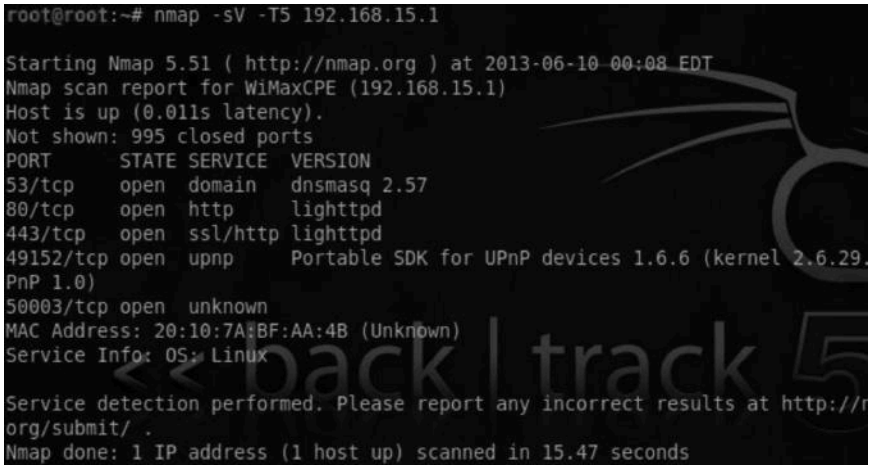
ServiceVersionDetection

So,untilnowwediscussedhowtofigureouttheservicesthatarerunningonacertainport.Inthis section,wewilllearntousenmaptofindtheexactversionoftheservicerunningonaport;this couldhelpuslookforthepotentialexploitsforthatparticularversionoftheservice.

Nmaphasadatabasenamednmap-servicesthatcontainmorethan2200well-knownservices. Theserviceversiondetectioncanbeperformedbyspecifyingthe–svparametertothenmap.

Command:

```
nmap-sV<targetIP>
```



OSFingerprinting

NmaphasahugeOSfingerprintingdatabasewithmorethan2600OSfingerprints.ItsendsTCP andUDPpacketstothetargetmachine,andtheresponsethatisreceivediscomparedwiththe database.Ifthefingerprintmatches,itdisplaystheresults.

Command:

```
nmap-O<TargetAddress>
```

The sample output looks asfollows:

Nmap also has other options for guessing OS, such as –osscan-limit, which would limitthe detection to a few, more promising targets. This would save a lot of time. The second one is–osscan-guess,whichdetectsinabetterandmoreaggressivemanner.Youcanalsousethe–A command to perform both OS and service version detection:

```
nmap-n-A-T5<target IP>
```

The–n–T5parameterwouldspeedupourscan,butyoushouldkeepinmindthatOSdetection andservicedetectionmethodsareveryloudattheotherendandareofteneasilydetectedbyIDS and IPS.

POF

POF stands for *passive OS fingerprinting*. As the name suggests, it does not directly engage with the targetwhileperformingOSfingerprinting;itmonitorsandtriestoidentifytheTCPstack,and based on the TCP stack type, it figures out the type of OS.

ThefollowingparagraphfromofficialdocumentationdescribethecapabilitiesofPOF:

> Common uses for pof include reconnaissance during penetration tests; routine network monitoring; detection of unauthorized network interconnects in corporateenvironments;providingsignalsforabuse-preventiontools;andmiscellaneous forensics.

Output

Nmap has various options for interpreting the output in a user-friendly and readable format. It supportsdifferenttypesofoutputformats.Theoutputformatsmayallowustofilteroutresults from nmap such as open ports, closed ports, and hosts.

Thethreepopularformatsusedarediscussedinbriefnext.

Normal Format
GreppableFormat
XML Format

NormalFormat

Thenormalformatisusedtooutputtheresultsofnmaptoanytextfile.Hereisanexampleofa simpleSYNscan.Theresultswouldbeoutputtedtoafilenamedrafay.txt.

Nmap–sS–PN<targetIP>–oNrafay.txt

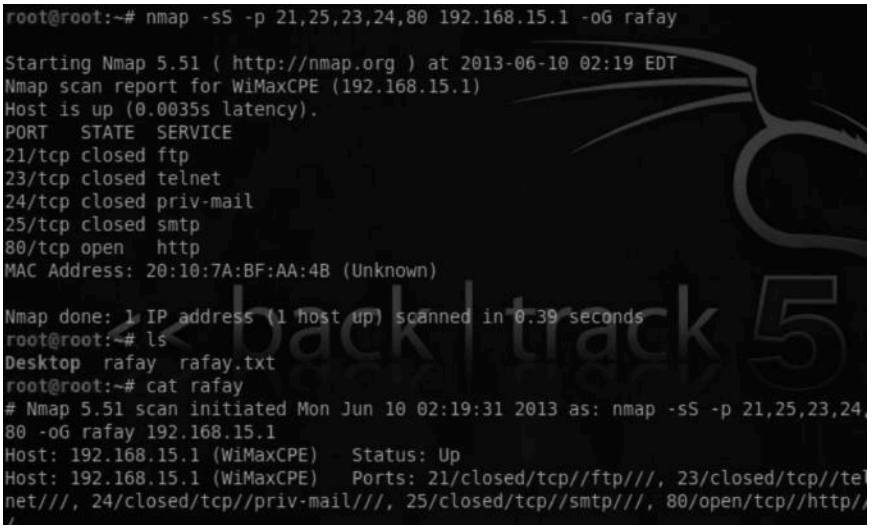
GrepableFormat

InUnix-basedoperatingsystems,wehaveaveryusefulcommand“grep”,whichcansearchfor specificresultssuchasportsandhosts.Withthegrepableformat,theresultsarepresentedwith one host per line.

Example

```
nmap-sS192.168.15.1-oGrafay
```

Thiscommandwouldsavetheoutputintoagrepableformat,whichisonehostperline.



Thefollowingcommandwillhighlightalltheportsthatareopen,whichinthiscaseisonly port 80.

XMLFormat

TheXMLformatisbyfarthemostusefuloutputformatinnmap.ThereasonisthattheXML outputgeneratedfromnmapcanbeeasilyportedovertodradisframeworkandarmitage.

Example

```
nmap-sS192.168.15.1-oX<filename>
```

AdvancedFirewall/IDSEvadingTechniques

ThetechniquesthatwehavediscussedhereareveryloudinnatureandareoftendetectedbyfirewallsandIDS.EvenscantechniquessuchasXMAS,FIN,andNULLarenotthataccurate;also, theydon'tworkontheWindowsoperatingsystem,sotheyhavealimitedadvantageoverfirewalls and IDS.

Inthissection,wewilldiscusssomeofthetechniquesthatcanbeusedtoevadefirewalldetection.Thereisnouniversalmethodtodothis;it'sallbasedontrialanderror.Thus,methodscould workonsomefirewalls/IDSbutfailwithothers.Italldependsuponhowstrongtherulesetsare.

TheNmapbookdiscussesawidevarietyoftechniquesthatcouldbeusedtogetpastfirewalls. Wewillnowbrieflylookatsomeofthem:

- Timingtechnique
- Fragmentedpackets

- Sourceport scan
- SpecifyinganMTU
- Sendingbadchecksums

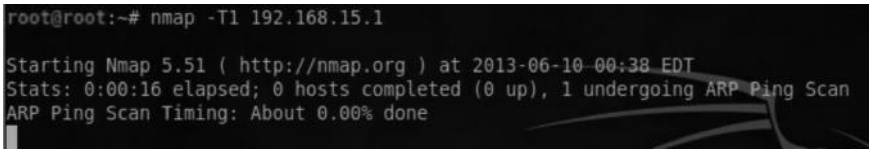
TimingTechnique

Thetimingtechniqueisoneofthebesttechniquestoevadefirewalls/IDS.Theideabehindthis techniqueistosendthepacketsgradually,sotheydonotendupbeingdetectedbyfirewalls/IDS. InnmapwecanlaunchatimingscanbyspecifyingtheTcommandfollowedbyanumberranging from0to5.IncreasingthevaluesfromT0toT5wouldincreasethespeedofthescan.

- *T0*—Paranoid
- *T1*—Sneaky
- *T2*—Polite
- *T3*—Normal
- *T4*—Aggressive
- *T5*—Insane

Example
Wewillperformasneakyscan(T1)andanalyzeitsbehaviorinwireshark:

```
nmap-T1<TargetiP>
```



WiresharkOutput

65	120.685689	192.168.15.1	192.168.15.14	TCP	sunrpc > 55648 [RST,
66	120.946563	fe80::44e7:d760:e29d:	ff02::1:2	DHCPv6	Solicit XID: 0x77ce5
67	125.697354	20:10:7a:bf:aa:4b	Vmware_18:20:15	ARP	Who has 192.168.15.1
68	125.697591	Vmware_18:20:15	20:10:7a:bf:aa:4b	ARP	192.168.15.14 is at
69	135.698079	192.168.15.14	192.168.15.1	TCP	55648 > ftp [SYN] Se
70	135.702102	192.168.15.1	192.168.15.14	TCP	ftp > 55648 [RST, AC
71	140.706922	Vmware_18:20:15	20:10:7a:bf:aa:4b	ARP	Who has 192.168.15.1
72	140.712247	20:10:7a:bf:aa:4b	Vmware_18:20:15	ARP	192.168.15.1 is at 2
73	150.705384	192.168.15.14	192.168.15.1	TCP	55648 > pptp [SYN] S
74	150.709004	192.168.15.1	192.168.15.14	TCP	pptp > 55648 [RST, A

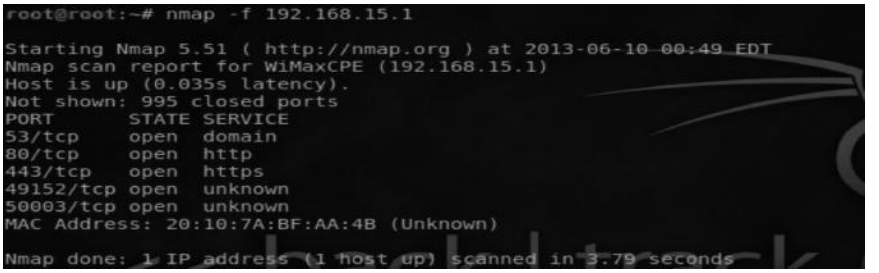
Fromthewiresharkoutput,youcanclearlyseethe“TCP”packetsbeingsentafteracertain time interval.

FragmentedPackets

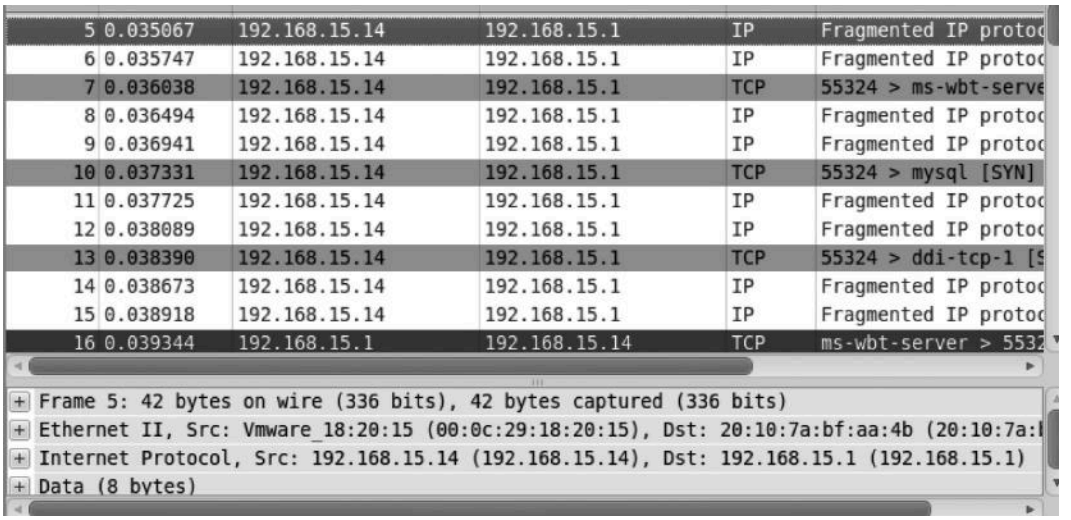
During fragmentation we split the packets into small chunks making it harder for the IDS to detect.TheycangetpastsomeIDSbecausetheIDSwouldanalyzeasinglefragmentbutnotall thepackets.Thereforetheywillnotfindanythingsuspicious.However,manymodernIDScan rebuild the fragments into a single packet, making them detectable.

Example

```
nmap-f192.168.15.1
```



WiresharkOutput



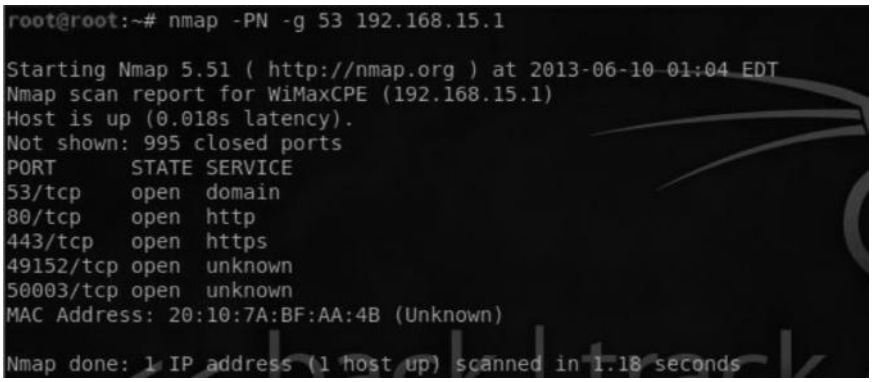
Thisoutputshowsusthatthepacketsaredividedinto8bytesofdata.

SourcePortScan

Itisverycommonforanetworkadministratortoallowtrafficfromacertainsourceport.Wecan usethistoouradvantagetobypassbadlyconfiguredfirewalls.Commonportsthatwecanspecify as source are 53, 80, and 21.

Example
The–gparameterhelpsusspecifyasourceport,whichinthiscaseis53(DNS).

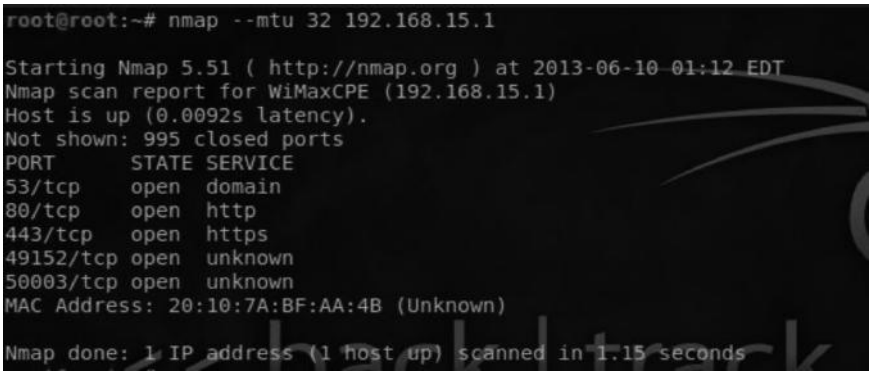
```
nmap-PN-g53192.168.15.1
```



SpecifyinganMTU

MTUstandsformaximumtransmissionunit.ThevaluesthatcanbedefinedasMTUaremultiplesof8(e.g.,8,16,24,32).NmapallowsustospecifyourownMTU.Basedonyourinput, nmapwillgeneratepackets.Forexample,ifyouspecify32,nmapwillgeneratea32bytepacket. ThechangeofthisMTUcanhelpusevadesomeofthefirewalls.

Example
```
nmap-mtu32<targetip>
```

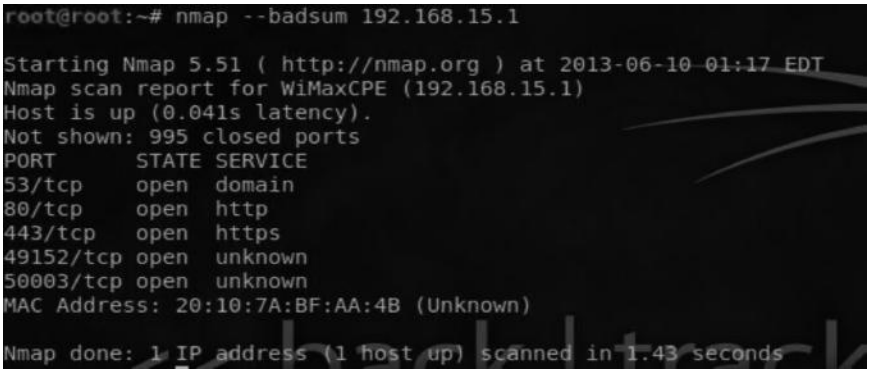


SendingBadChecksums

Checksums are used in the TCP header for error detection. However, we can use incorrect checksumstoouradvantage.Bysendingbad/incorrectchecksums,wecanbypasssomefirewalls depending upon the rule sets and how they are configured.

Example

```
nmap-badsum<TargetIP>
```



Decoys

Thisisthelastmethodthatwewilldiscussinthissection.Itisveryeffectivewhenyouwantto usestealth.Theideabehindthisscanistosendspoofedpacketsfromotherhosts,whichwould makeitverydifficultfornetworkadministratorstodetectfromwhichhostthescanoriginated. Since the decoy has the potential to generate a very large number of packets, it could cause a possible DOS (denial of service).

Example

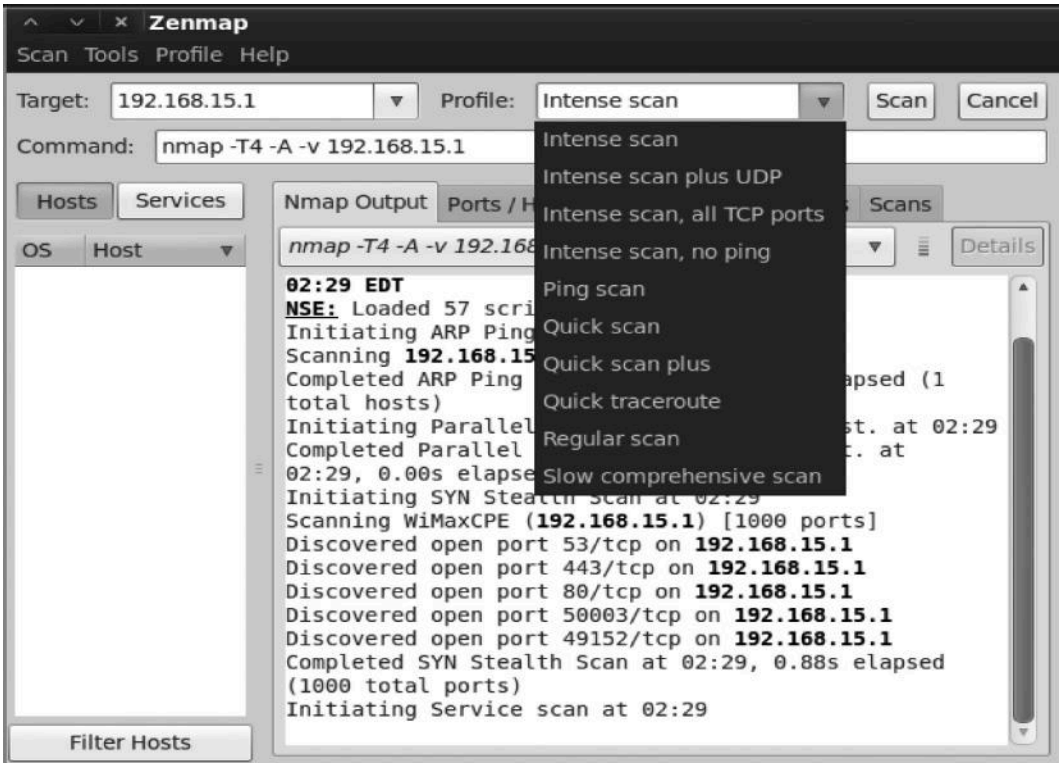
```
nmap-DRND:10<targetiP>
```

ThiscommandwouldgeneratearandomnumberofdecoysforthetargetiP.

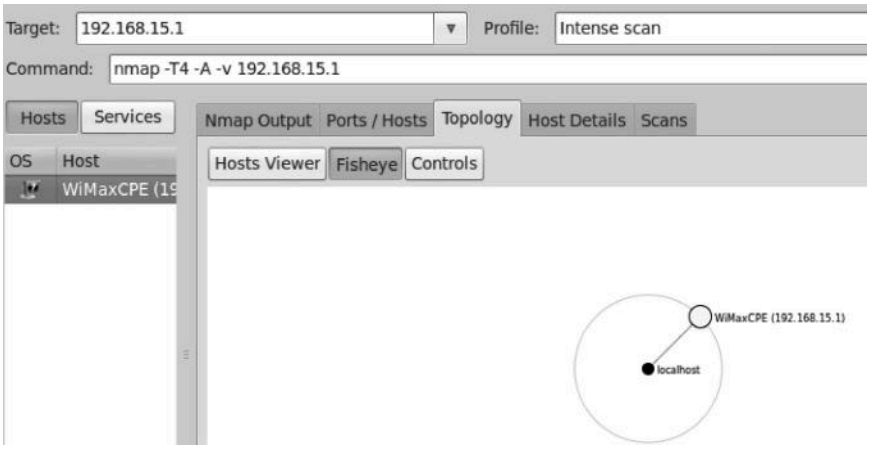
ZENMAP

ZenmapisaGUIversionofnmap.PersonallyIamnotabigfanofthistool,butIthoughtitwould be worth mentioning for all the GUI lovers. It does include some built-in profiles for scanning and

IguessIhavetalkedabouteveryparameterthattheyhaveusedintheirscanningprofiles.Sojust take some time to understand the scanning profiles, their function, and most importantly what they are doing in background by inspecting the packets through wireshark.



Thetopologyoptioninsidezenmapwilldrawapictureofthenetworktopology.Inthisway youcanvisualizewhereexactlythehostislocated.



Further Reading

We have discussed pretty much everything that you need that can help you get started with nmap, but if you are interested in learning more about the different types of scanning and evasion techniques, I highly recommend you go ahead and read the book *NMAP Network Scanning* by Gordon "Fyodor" Lyon, the creator of nmap. This book describes every method inside nmap in great detail. However, I suggest you read the "PORT-SCAN Types" chapter to understand the pros and cons of every type of scan. The knowledge of what type of scan to use in a certain situation would make you a better pentester. The book is freely available for download at nmap.org/book. You can also buy the print version from amazon.com.