

# **NMAP BÖLÜM 1-5**

---

## **Chapter 1. Getting Started with Nmap (Bölüm 1. Nmap ile Başlarken)**

İçindekiler

### **Introduction (Giriş)**

- Nmap Overview and Demonstration (Nmap'e Genel Bakış ve Gösterim)
  - Avatar Online (Avatar Online)
  - Saving the Human Race (İnsan İrkını Kurtarmak)
  - MadHat in Wonderland (MadHat Harikalar Diyarında )
- The Phases of an Nmap Scan ( Bir Nmap Taramasının Aşamaları )
- Legal Issues (Yasal Konular)
  - Is Unauthorized Port Scanning a Crime? (Yetkisiz Port Taraması Suç mudur? )
  - Can Port Scanning Crash the Target Computer/Networks? (Port Taraması Hedef Bilgisayarı/Ağları Çökertebilir mi?)
  - Nmap Copyright (Nmap Telif Hakkı )
- The History and Future of Nmap (Nmap'in Geçmiş ve Geleceği)
  - The History of Nmap (Nmap'in Geçmiş)
  - The Future of Nmap (Nmapin Geleceği)

### **Introduction (Giriş)**

Nmap ("Network Mapper") ağ keşfi ve güvenlik denetimi için ücretsiz ve açık kaynaklı bir yardımcı programdır. Birçok sistem ve ağ yöneticisi, ağ envanteri, hizmet yükseltme programlarını yönetme ve ana bilgisayar veya hizmet çalışma

süresini izleme gibi görevler için de yararlı bulmaktadır. Nmap, ağıda hangi ana bilgisayarların bulunduğu, bu ana bilgisayarların hangi hizmetleri (uygulama adı ve sürümü) sunduğunu, hangi işletim sistemlerini (ve işletim sistemi sürümlerini) çalıştırdıklarını, ne tür paket filtrelerinin / güvenlik duvarlarının kullanıldığını ve dzinelerce başka özelliği belirlemek için ham IP paketlerini yeni yollarla kullanır. Büyük ağları hızla taramak için tasarlanmıştır, ancak tek ana bilgisayarlara karşı iyi çalışır. Nmap tüm büyük bilgisayar işletim sistemlerinde çalışır ve hem konsol hem de grafik sürümleri mevcuttur.

Bu bölüm, Nmap ve tipik olarak nasıl kullanıldığına dair geniş bir genel bakış sağlamak için kurgusal hikayeler kullanır. Önemli bir yasal bölüm, kullanıcıların İSS hesap iptaline ve hatta hukuki ve cezai suçlamalara yol açabilecek tartışmalı kullanımlardan kaçınmalarına (ya da en azından farkında olmalarına) yardımcı olur. Ayrıca uzak makineleri çökertme risklerinin yanı sıra açık kaynak Nmap lisansı (GNU GPL'ye dayalı) ve telif hakkı gibi çeşitli konular da tartışılmaktadır.

## **Nmap Overview and Demonstration (Nmap'e Genel Bakış ve Gösterim)**

Bazen bir şeyi anlamadan en iyi yolu onu çalışırken görmektir. Bu bölüm, (çoğunlukla) kurgusal ancak tipik durumlarda kullanılan Nmap örneklerini içerir. Nmap'e yeni başlayanlar her şeyi bir kerede anlamayı beklememelidir. Bu sadece daha sonraki bölümlerde derinlemesine açıklanan özelliklere genel bir bakıştır. Bu kitap boyunca yer alan "çözümler", güvenlik denetçileri ve ağ yöneticileri için diğer birçok yaygın Nmap görevini göstermektedir.

### **Avatar Online (Avatar Online)**

Felix 15 Aralık'ta iş yerine gelir, ancak çok fazla yapılandırılmış görev beklememektedir. Çalıştığı küçük San Francisco sizma testi firması yaklaşan tatiller nedeniyle son zamanlarda sessizdir. Felix iş saatlerini kablosuz değerlendirme ve savaş sürüsü keşifleri için güçlü Wi-Fi antenleri inşa etme hobisini sürdürerek geçiriyor. Yine de Felix daha fazla iş yapmayı umuyor. Bilgisayar korsanlığı, ağ, güvenlik, Unix ve telefon sistemleri hakkında öğrenebileceği her şeyi öğrenerek geçirdiği çocukluğundan beri hobisi ve hayranlığı olmuştur. Bazen meraklı onu çok ileri götürdü ve Felix neredeyse 1990 Sundevil Operasyonu kovuşturmalarına sürüklendi. Neyse ki Felix ergenlik döneminden sabıka kaydı olmadan çıktı ve

güvenlik zayıflıkları konusundaki uzman bilgisini korudu. Bir profesyonel olarak, eskiden olduğu gibi aynı tür ağ izinsiz girişlerini gerçekleştirebilir, ancak sözleşmeye bağlı olarak kovuşturmadan muafiyet ve hatta bir maaş çeki avantajıyla! Yaratıcı becerilerini gizli tutmak yerine, raporlarını sunarken müşteri yönetimine bunlarla övünebilir. Bu yüzden patronu anten lehimleme işini yanında kesip satış departmanının Avatar Online oyun şirketiyle bir pen-test anlaşması yaptığı duyurduğunda Felix hayal kırıklığına uğramadı.

Avatar Online (AO), yeni nesil devasa çok oyunculu çevrimiçi rol yapma oyunları (MMORPG'ler) yaratmak için çalışan küçük bir şirketdir. Neil Stevenson'in Snow Crash'inde öngörülen Metaverse'den esinlenen ürünleri büyüleyici ancak yine de son derece gizli. Valve Software'in yaklaşan oyun kaynak kodunun yüksek profilli sizintisine tanık olduktan sonra, AO hızlı bir şekilde güvenlik danışmanlarını işe aldı. Felix'in görevi, ortakları fiziksel güvenlik, kaynak kodu denetimi, sosyal mühendislik ve benzeri konularda çalışırken harici (güvenlik duvarının dışından) bir güvenlik açığı değerlendirmesi başlatmaktadır. Felix'in bulunan tüm güvenlik açıklarından faydalananmasına izin verilir.

Güvenlik açığı değerlendirmesinin ilk adımı ağ keşfidir. Bu keşif aşaması, hedefin hangi IP adres aralıklarını kullandığını, hangi ana bilgisayarların mevcut olduğunu, bu ana bilgisayarların hangi hizmetleri sunduğunu, genel ağ topolojisi ayrıntılarını ve hangi güvenlik duvari / filtreleme politikalarının yürürlükte olduğunu belirler.

Taranacak IP aralıklarını belirlemek normalde ARIN (veya başka bir coğrafi kayıt) aramaları, DNS sorguları ve bölge aktarma girişimleri, çeşitli web hafiyelik teknikleri ve daha fazlasını içeren ayrıntılı bir süreç olacaktır. Ancak bu durumda, Avatar Online hangi ağların test edilmesini istediğini açıkça belirtmiştir: 6.209.24.0/24 üzerindeki kurumsal ağ ve 6.207.0.0/22 üzerinde bulunan üretim/DMZ sistemleri. Felix yine de IP whois kayıtlarını kontrol eder ve bu IP aralıklarının AO'ya tahsis edildiğini onaylar[1]. Felix bilinçaltında CIDR gösterimini çözer[2] ve bunu 1.280 IP adresi olarak tanır. Sorun değil.

Dikkatli bir tip olan Felix, ilk olarak Nmap liste taraması (-sL seçeneği) olarak bilinen şeyle başlar. Bu özellik basitçe verilen hedef netblock(lar)daki her IP adresini numaralandırır ve her biri üzerinde bir ters-DNS araması (-n belirtilmediği sürece) yapar. Bunu ilk olarak yapmanın bir nedeni gizliliktir. Ana bilgisayarların isimleri potansiyel güvenlik açıklarına işaret edebilir ve alarm zillerini çaldırmadan hedef ağın daha iyi anlaşılmasını sağlayabilir[3]. Felix'in bunu yapmasının bir başka

nedeni daha var: IP aralıklarının doğru olup olmadığını iki kez kontrol etmek. IP'leri sağlayan sistem yönetici bir hata yapmış olabilir ve yanlış şirketi taramak bir felaket olur. Avatar Online ile imzalanan sözleşme, ağlarına sızmak için hapisten kurtulma kartı olarak işlev görebilir, ancak Felix yanlışlıkla başka bir şirketin sunucusunu tehlikeye atarsa yardımcı olmaz! Kullandığı komut ve sonuçlardan bir alıntı Örnek 1.1'de gösterilmektedir.

Örnek 1.1. Avatar Online IP adreslerine karşı Nmap liste taraması

```
felix> nmap -sL 6.209.24.0/24 6.207.0.0/22

Starting Nmap ( https://nmap.org )
Nmap scan report for 6.209.24.0
Nmap scan report for fw.corp.avataronline.com (6.209.24.1)
Nmap scan report for dev2.corp.avataronline.com (6.209.24.2)
Nmap scan report for 6.209.24.3
Nmap scan report for 6.209.24.4
...
Nmap scan report for dhcp-21.corp.avataronline.com (6.209.24.21)
Nmap scan report for dhcp-22.corp.avataronline.com (6.209.24.22)
Nmap scan report for dhcp-23.corp.avataronline.com (6.209.24.23)
...
Nmap scan report for 6.207.0.0
Nmap scan report for gw.avataronline.com (6.207.0.1)
Nmap scan report for ns1.avataronline.com (6.207.0.2)
Nmap scan report for ns2.avataronline.com (6.207.0.3)
Nmap scan report for ftp.avataronline.com (6.207.0.4)
Nmap scan report for 6.207.0.5
Nmap scan report for 6.207.0.6
Nmap scan report for www.avataronline.com (6.207.0.7)
Nmap scan report for 6.207.0.8
...
Nmap scan report for cluster-c120.avataronline.com (6.207.2.120)
Nmap scan report for cluster-c121.avataronline.com (6.207.2.121)
Nmap scan report for cluster-c122.avataronline.com (6.207.2.122)
...
Nmap scan report for 6.207.3.255
Nmap done: 1280 IP addresses (0 hosts up) scanned in 331.49 seconds
felix>
```

Sonuçları okuyan Felix, ters-DNS girişleri olan tüm makinelerin Avatar Online'a çözümlendiğini görür. Başka hiçbir işletme IP alanını paylaşmıyor gibi görünüyor. Dahası, bu sonuçlar Felix'e kaç makinenin kullanımında olduğu ve çögünün ne için kullanıldığı hakkında kabaca bir fikir veriyor. Artık biraz daha müdahaleci olmaya

ve bir port taraması yapmaya hazırlıdır. Ağda dinlenen her hizmetin uygulama ve sürüm numarasını belirlemeye çalışan Nmap özelliklerini kullanır. Ayrıca Nmap'in işletim sistemi parmak izi olarak bilinen bir dizi düşük seviyeli TCP/IP probu aracılığıyla uzaktaki işletim sistemini tahmin etmeye çalışmasını ister. Bu tür bir tarama hiç de gizli değildir, ancak bu Felix'i ilgilendirmez. O, AO yöneticilerinin bu bariz taramaları fark edip etmediğiyle ilgileniyor. Biraz düşündükten sonra, Felix aşağıdaki komutta karar kılar:

```
nmap -sS -p- -PE -PP -PS80,443 -PA3389 -PU40125 -A -T4 -oA avatartcpscan-%D 6.209.24.0/24 6.207.0.0/22
```

Bu seçenekler daha sonraki bölümlerde açıklanmıştır, ancak burada bunların hızlı bir özeti bulunmaktadır.

**-sS** SYN taraması olarak bilinen verimli TCP port tarama tekniğini etkinleştirir. Felix UDP taraması da yapmak isteseydi sonuna bir U eklerdi, ancak bunu daha sonraya saklıyor. SYN taraması varsayılan tarama türüdür, ancak bunu açıkça belirtmek zarar vermez.

**-p-** Nmap'in 1-65535 arasındaki tüm portları taramasını ister. Bu, yalnızca büyük ölçekli Internet testlerinde en yaygın olarak erişilebilir olduğunu tespit ettiğimiz 1.000 bağlantı noktasını taramak olan varsayılandan daha kapsamlıdır. Bu seçenek biçimini basitçe -p1-65535 için bir kısaltmadır. Felix oldukça gayrimeşru olan sıfır numaralı portu da taramak isteseydi -p0-65535 belirtebilirdi. p seçeneği çok esnek bir sözdizimine sahiptir, hatta farklı bir UDP ve TCP portları kümесinin belirtilmesine izin verir.

**-PE -PP -PS80,443 -PA3389 -PU40125** Bunların hepsi, bir ağdaki hangi hedeflerin gerçekten kullanılabilir olduğunu belirlemek ve kullanılmayan IP adreslerini taramak için çok fazla zaman harcamaktan kaçınmak için birlikte kullanılan ana bilgisayar keşif teknikleridir (ping türleri). Bu özel yöntem ICMP yankı isteği ve zaman damgası isteği paketleri; 80 ve 443 numaralı bağlantı noktalarına TCP SYN paketi; 3389 numaralı bağlantı noktasına TCP ACK paketleri; ve 40,125 numaralı bağlantı noktasına bir UDP paketi gönderir. Nmap, hedef ana bilgisayardan bu problemlerden herhangi birine bir yanıt alırsa, ana bilgisayar çalışır ve tarama için kullanılabilir olarak kabul eder. Bu, Internet üzerinden hedeflere karşı ana bilgisayar keşfi için büyük ölçekli ampirik testlerde bulduğumuz en etkili altı prob kombinasyonudur. Nmap varsayılanı olan -PE -PS443 -PA80 -PP'den daha kapsamlıdır. Bir pen-test durumunda, genellikle çalışıyor görünmeseler bile her ana bilgisayarı taramak

istersiniz. Sonuçta, seçtiğiniz problar göz ardı edilecek şekilde yoğun bir şekilde filtrelenmiş olabilirler, ancak başka bir belirsiz port kullanılabilir olabilir. Kullanılabilir bir konak göstersin ya da göstermesin her IP'yi taramak için, yukarıdakilerin hepsi yerine -Pn seçeneğini belirtin. Felix böyle bir taramayı arka planda başlatır, ancak tamamlanması saatler sürebilir.

**-A** Bu kısayol seçeneği işletim sistemi ve hizmet algılama gibi Gelişmiş ve Agresif Özellikleri açar. Bu yazının yazıldığı sırada -sV -sC -O --traceroute (sürüm algılama, varsayılan komut dosyası kümesiyle Nmap Komut Dosyası Motoru, uzak işletim sistemi algılama ve traceroute) ile eşdeğerdir. Daha sonra -A'ya daha fazla özellik eklenebilir.

**-T4** Zamanlamayı agresif seviyeye ayarlar (#4 of 5). Bu, -T agresif belirtmekle aynıdır, ancak yazması ve hecelemesi daha kolaydır. Genel olarak, hedef ağlarla aranızdaki bağlantı oldukça hızlı ve güvenilirse -T4 seçeneği önerilir.

**-oA avatartcpscan-%D** Sonuçları her formatta (normal, XML, grepable) avatartcpscan-. adlı dosyalara çıktı olarak verir; burada uzantılar sırasıyla .nmap, .xml ve .gnmap'tır. <date> ay, gün ve yılı 073010 gibi bir formatta verir. Tüm çıktı biçimleri başlangıç tarihini ve saatini içerir, ancak Felix dosya adında tarihi açıkça belirtmeyi sever. Normal çıktı ve hatalar da stdout[4]'a gönderilmeye devam eder.

**6.209.24.0/24 6.207.0.0/22** Bunlar yukarıda tartışılan Avatar Online ağ bloklarıdır. Bunlar CIDR gösteriminde verilmiştir, ancak Nmap bunların diğer birçok formatta belirtilmesine izin verir. Örneğin, 6.209.24.0/24 bunun yerine 6.209.24.0-255 olarak belirtilebilir.

Binden fazla IP adresine karşı böylesine kapsamlı bir tarama biraz zaman alabileceğinden, Felix sadece çalıştırılmaya başlar ve Yagi anteni üzerinde çalışmaya devam eder. Birkaç saat sonra taramanın bittiğini fark eder ve sonuçlara bir göz atar. Örnek 1.2 keşfedilen makinelerden birini göstermektedir.

Örnek 1.2. Bir AO güvenlik duvarına karşı Nmap sonuçları

```
Nmap scan report for fw.corp.avataronline.com (6.209.24.1)
(The 65530 ports scanned but not shown below are in state: filtered)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 3.7.1p2 (protocol 1.99)
| ssh-hostkey: 1024 7c:14:2f:92:ca:61:90:a4:11:3c:47:82:d5:8e:a9:6b (DSA)
|_2048 41:cf7d:839d:7f66:0ael:8331:7fd4:5a97:5a (RSA)
_| sshv1: Server supports SSHv1
53/tcp    open  domain       ISC BIND 9.2.1
110/tcp   open  pop3        Courier pop3d
113/tcp   closed auth
143/tcp   open  imap        Courier Imap 1.6.X - 1.7.X
3128/tcp  open  http-proxy  Squid webproxy 2.2.STABLE5
Device type: general purpose
Running: Linux 2.4.X|2.5.X
OS details: Linux Kernel 2.4.0 - 2.5.20
Uptime 3.134 days
```

Eğitimli bir göz için bu, AO'nun güvenlik duruşu hakkında önemli bilgiler aktarır. Felix ilk olarak ters DNS adına dikkat çekiyor-bu makinenin kurumsal ağları için bir güvenlik duvarı olduğu anlaşılıyor. Bir sonraki satır önemlidir, ancak genellikle göz ardı edilir. Bu makinedeki portların büyük çoğunluğunun filtrelenmiş durumda olduğunu belirtir. Bu, güvenlik duvarı kuralları tarafından engellendiği için Nmap'in bağlantı noktasına ulaşamadığı anlamına gelir. Seçilmiş birkaç port dışında tüm portların bu durumda olması güvenlik yeterliliğinin bir işaretidir. Varsayılan olarak reddet, iyi nedenlerden dolayı bir güvenlik mantrasıdır – bu, birisi yanlışlıkla bu makinede SunRPC'yi (111 numaralı bağlantı noktası) açık bıraksa bile, güvenlik duvarı kurallarının saldırganların onunla iletişim kurmasını engelleyeceği anlamına gelir.

Felix daha sonra sırayla her port satırına bakar. İlk bağlantı noktası Güvenli Kabuk'tur (OpenSSH). Sürüm 3.7.1p2 yaygındır, çünkü birçok yönetici önceki sürümleri etkileyen potansiyel olarak istismar edilebilir tampon yönetimi hataları nedeniyle bu sürümü yükseltmiştir. Nmap ayrıca sshv1 NSE betiğinin daha az güvenli SSHv1 protokolünün desteklendiğini bildirdiğini de not etmektedir. Gerçekten paranoid bir sistem yöneticisi yalnızca belirli güvenilir IP adreslerinden SSH bağlantılarına izin verir, ancak yöneticinin evden uzaktayken acil durum erişimine ihtiyaç duyması durumunda açık erişim için tartışılabilir. Güvenlik genellikle ödüne vermemeyi gerektirir ve bu da haklı olabilir. Felix, Ncrack kaba kuvvet ağ kimlik doğrulama kırıcısını ve özellikle sunucuya karşı özel zamanlama tabanlı SSH kullanıcı numaralandırma aracını denemek için bir not alır.

Felix ayrıca 53 numaralı porta da dikkat çekiyor. Uzaktan istismar edilebilir güvenlik açıkları konusunda uzun bir geçmişe sahip olan ISC BIND çalıştırıyor. Daha fazla ayrıntı için BIND güvenlik sayfasını ziyaret edin. BIND 9.2.1, varsayılan yapı savunmasız olmamasına rağmen, potansiyel olarak istismar edilebilir bir arabellek taşmasına sahiptir. Felix, bu sunucunun libbind sorununa karşı savunmasız olmadığını kontrol eder ve bulur, ancak bu konunun dışında. Bu sunucu neredeyse kesinlikle dışarıdan erişilebilen bir ad sunucusu çalıştırılmamalıdır. Bir güvenlik duvarı, feci bir tehlike riskini en aza indirmek için yalnızca temel unsurları çalıştırmalıdır. Ayrıca, bu sunucu herhangi bir alan adı için yetkili değildir - gerçek ad sunucuları üretim ağındadır. Bir yönetici muhtemelen yalnızca güvenlik duvarı içindeki istemcilerin bu ad sunucusıyla iletişim kurmasını istemiş, ancak bunu yalnızca dahili arayüze kilitleme zahmetine girmemiştir. Felix daha sonra bölge aktarım istekleri ve izinsiz sorgular kullanarak bu gereksiz sunucudan önemli bilgiler toplamaya çalışacaktır. Önbellek zehirleme girişiminde de bulunabilir. Felix, [windowsupdate.microsoft.com](http://windowsupdate.microsoft.com) veya başka bir önemli indirme sunucusunun IP'sini taklit ederek, şüphelenmeyen dahili istemci kullanıcılarını, kendisine güvenlik duvarının arkasından tam ağ erişimi sağlayan bir truva atı programını çalıştırması için kandırabilir.

Son açık bağlantı noktası bir Squid proxy'dir. Bu, dahili müşterileri kullanımı için tasarlanmış olabilecek ve dışarıdan (ve özellikle güvenlik duvarından) erişilememesi gereken bir başka hizmettir. Felix'in AO güvenlik yöneticileri hakkındaki başlangıçtaki olumlu görüşü daha da azalır. Felix, internetteki diğer sitelere bağlanmak için bu proxy'yi kötüye kullanıp kullanamayacağını test edecektir. Spam gönderenler ve kötü niyetli bilgisayar korsanları izlerini gizlemek için genellikle bu şekilde proxy kullanırlar. Daha da kritik olanı, Felix dahili ağa proxy yoluyla girmeye çalışacaktır. Bu yaygın saldırısı, Adrian Lamo'nun 2002 yılında New York Times'in iç ağına girme yöntemidir. Lamo, gazetecileri arayarak NY Times ve diğer şirketlere karşı gerçekleştirdiği saldırıları ayrıntılı olarak anlattıktan sonra yakalanmıştır.

Aşağıdaki satırlar bunun bir Linux kutusu olduğunu ortaya koymaktadır ki bu da istismar girişiminde bulunurken değerli bir bilgidir. Üç günlük düşük çalışma süresi, TCP zaman damgası seçeneği değeri için birkaç prob gönderilerek ve satır sıfırda geri çekilerek işletim sistemi parmak izi sırasında tespit edildi.

Felix daha sonra Örnek 1.3'te gösterildiği gibi başka bir makine için Nmap çıktısını inceler.

### Örnek 1.3. Bir başka ilginç AO makinesi

```
Nmap scan report for dhcp-23.corp.avataronline.com (6.209.24.23)
(The 65526 ports scanned but not shown below are in state: closed)
PORT      STATE    SERVICE      VERSION
135/tcp    filtered msrpc
136/tcp    filtered profile
137/tcp    filtered netbios-ns
138/tcp    filtered netbios-dgm
139/tcp    filtered netbios-ssn
445/tcp    open     microsoft-ds Microsoft Windows XP microsoft-ds
1002/tcp   open     windows-icfw?
1025/tcp   open     msrpc      Microsoft Windows msrpc
16552/tcp  open     unknown
Device type: general purpose
Running: Microsoft Windows NT/2K/XP
OS details: Microsoft Windows XP Professional RC1+ through final release

Host script results:
|_nbstat: NetBIOS name: TRACYD, NetBIOS user: <unknown>, NetBIOS MAC: 00:20:35:00:29:a2:7f (IBM)
|_smbv2-enabled: Server doesn't support SMBv2 protocol
| smb-os-discovery:
|   OS: Windows XP (Windows 2000 LAN Manager)
|_ Name: WORKGROUP\JOHND
```

Felix Ağ'da bu Windows XP kutusunu gördüğünde gülmüşüyor. Bir dizi MS RPC güvenlik açığı sayesinde, işletim sistemi yamaları güncel değilse bu makinelerin ele geçirilmesi çok kolaydır. İkinci satır varsayılan durumun kapalı olduğunu gösteriyor, yani güvenlik duvarı bu makine için kendisiyle aynı varsayılan olarak reddet politikasına sahip değil. Bunun yerine özellikle 135-139'da tehlikeli olduğunu düşündükleri Windows portlarını engellemeye çalışmışlar. MS, Windows XP'deki diğer birçok bağlantı noktasında MS RPC işlevsellliğini dışa aktardığı için bu filtre ne yazık ki yetersizdir. TCP portları 445 ve 1025 bu taramadan iki örnektir. Nmap 16552'yi tanıyamamış olsa da, Felix bunun muhtemelen MS Messenger Hizmeti olduğunu bileyek kadar bu modeli gördü. Eğer AO varsayılan filtreleme kullanıyor olsaydı, 16552 numaralı porta ilk etapta erişilemezdi. Sonuçlar sayfasına bakan Felix, bu DHCP ağında birkaç Windows makinesi daha görür. Felix en sevdiği DCOM RPC açığını onlara karşı denemek için sabırsızlanıyor. HD Moore tarafından yazılmıştır ve <http://www.metasploit.com/tools/dcom.c> adresinde mevcuttur. Bu başarısız olursa, deneyeceği birkaç yeni MS RPC güvenlik açığı var.

Felix, ağı ele geçirmek için kullanabileceğini güvenlik açıkları için sonuçları incelemeye devam eder. Üretim ağında, gw.avataronline.com'un aynı zamanda sistemler için ilkel bir güvenlik duvarı görevi gören bir Cisco yönlendiricisi olduğunu görür. Yalnızca ayrıcalıklı bağlantı noktalarını (1024'ün altındakiler) engellemeye tuzağına düşüyorlar, bu da bir grup savunmasız SunRPC ve diğer hizmetleri bu ağda erişilebilir bırakıyor. Clust-\* gibi adlara sahip makinelerin her

birinde Nmap'in tanımadığı düzinelere bağlantı noktası açıktır. Bunlar muhtemelen AO oyun motorunu çalıştırın özel daemonlardır. [www.avataronline.com](http://www.avataronline.com) HTTP ve HTTPS portlarında açık Apache sunucusu olan bir Linux kutusudur. Ne yazık ki, OpenSSL kütüphanesinin istismara açık bir sürümü ile bağlantılıdır. Oops! Güneş batmadan önce, Felix hem kurumsal hem de üretim ağlarındaki ana bilgisayarlara ayrıcalıklı erişim elde etti.

Felix'in de gösterdiği gibi, Nmap güvenlik denetçileri ve ağ yöneticileri tarafından müşteri/kurumsal ağlardaki güvenlik açıklarını bulmaya yardımcı olmak için sık sık kullanılmaktadır. Sonraki bölümlerde Felix tarafından kullanılan teknikler ve diğer birçok Nmap özelliği çok daha ayrıntılı olarak açıklanmaktadır.

### **Saving the Human Race (İnsan İrkını Kurtarmak)**

Şekil 1.1. Trinity saldırıyla başlıyor



Trinity'nin başı oldukça dertte! Kabul ettiğimiz dünyanın aslında makine efendiler tarafından yönetilen sanal bir "Matrix" olduğunu keşfeden Trinity, buna karşı savaşmaya ve insan ırkını bu zihinsel kölelikten kurtarmaya karar verir. Daha da kötüsü, özgür insanlardan oluşan yeraltı kolonisi (Zion) 250.000 güçlü uzayı

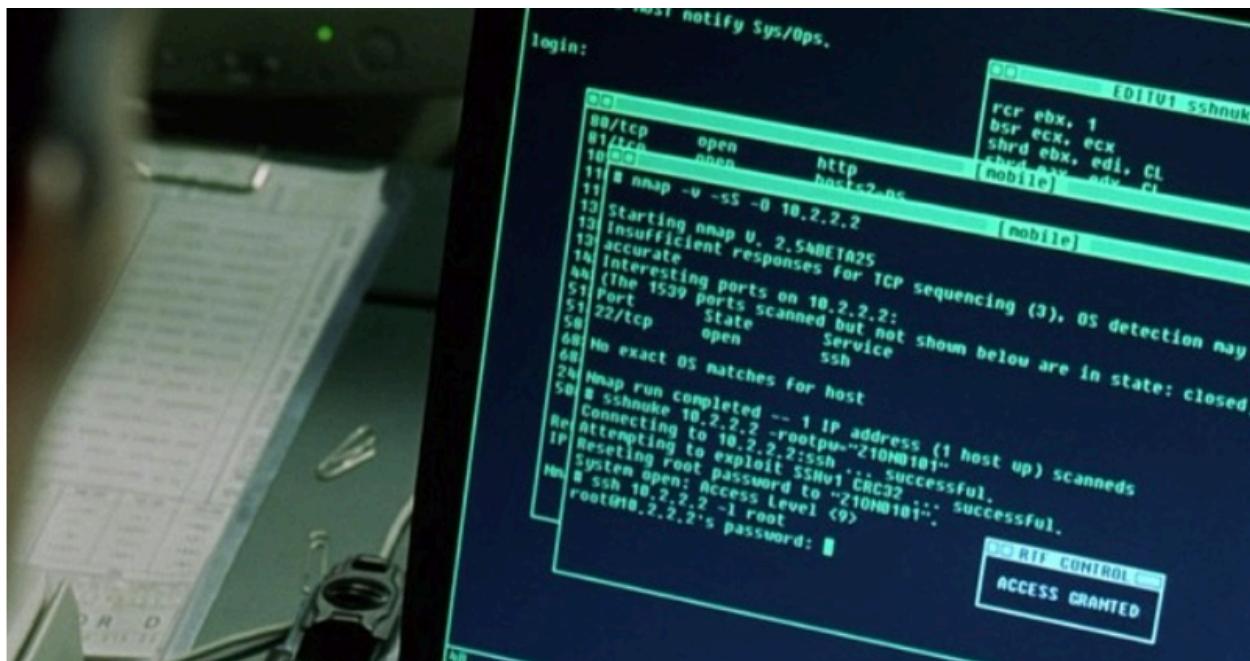
nöbetçinin saldırısı altındadır. Tek umudu, 27 şehir bloğunun acil durum güç sistemini beş dakikadan kısa bir sürede devre dışı bırakmaktır. Önceki ekip bunu denerken öldürdü. Hayatın en kasvetli anlarında, tüm umutlar kaybolmuş gibi göründüğünde, neye başvurmalısınız? Nmap, tabii ki! Ama henüz değil.

Öncelikle, birçok ağda güvenlik duvarları ve saldırısı tespit sistemlerini (IDS) içeren çevre güvenliğini aşması gereklidir. Bu cihazları atlatmak için kullanılan ileri tekniklerin (bu kitabın ilerleyen bölümlerinde ele alınacaktır) farkındadır. Ne yazık ki acil durum güç sistemi yöneticileri böylesine kritik bir sistemi dolaylı yoldan bile olsa internete bağlamamaları gerektiğini çok iyi biliyorlardı. Hiçbir kaynak yönlendirme ya da IP ID sahteciliği taraması Trinity'nin bu "hava boşluğu" güvenliğinin üstesinden gelmesine yardımcı olmayacağından emin. Hızlı düşünerek, motosikletiyle yakındaki bir binanın çatısından atlamayı, elektrik santralinin nöbetçi kulübesine inmeye ve ardından tüm güvenlik görevlilerini dövmeyi içeren akıllıca bir plan yapar. Bu gelişmiş teknik hiçbir fiziksel güvenlik kılavuzunda yer almaz, ancak son derece etkili olduğu kanıtlanır. Bu teknik, zeki bilgisayar korsanlarının her zaman konserve açıkların script-kiddie yaklaşımını kullanmak yerine kendi saldırularını nasıl araştırdıklarını ve tasarladıklarını göstermektedir.

Trinity bilgisayar odasına doğru ilerler ve bir terminalin başına oturur. Ağın özel 10.0.0.0/8 ağ adres alanını kullandığını hemen tespit eder. Ağ adresine yapılan bir ping işlemi düzinece makineden yanıt alınmasını sağlar. Bir Nmap ping taraması mevcut makinelerin daha kapsamlı bir listesini sağlayabilirdi, ancak yayın tekniğini kullanmak değerli saniyeler kazandırdı. Sonra Nmap[5]'i çıkardı. Terminalde 2.54BETA25 sürümü yükülüdür. Bu sürüm eski (2001) ve yeni sürümlere göre daha az verimli, ancak Trinity'nin gelecektен daha iyi bir sürüm yükleyecek zamanı yoktu. Bu iş zaten uzun sürmeyecektir. nmap -v -sS -O 10.2.1.3 komutunu çalıştırır. Bu, 10.2.1.3'e karşı bir TCP SYN taraması ve OS algılaması yürütür ve ayrıntılı çıktı sağlar. Ana bilgisayar bir düzineden fazla bağlantı noktası açık olan bir güvenlik felaketi-AIX 3.2 gibi görünüyor. Ne yazık ki, tehlikeye atması gereken makine bu değil. Bu yüzden aynı komutu 10.2.2.2'ye karşı çalıştırıyor. Bu kez hedef işletim sistemi tanınmıyor (Nmap'i yükseltmesi gerekirdi!) ve yalnızca 22 numaralı bağlantı noktası açık. Bu, Secure Shell şifreli yönetim hizmetidir. Her seksiz lateks giyimli hacker tanrıçasının bildiği gibi, o dönemdeki (2001) birçok SSH sunucusunun CRC32 telafi saldırısı dedektöründe istismar edilebilir bir güvenlik açığı vardır. Trinity, tamamen montaj kodundan oluşan bir açıktan yararlanır ve bunu hedef kutunun kök şifresini Z10N0101 olarak değiştirmek için kullanır. Trinity normal

şartlar altında çok daha güvenli şifreler kullanır. Root olarak oturum açar ve 27 şehir bloğunun acil durum yedek güç sistemini devre dışı bırakmak için bir komut verir ve tam zamanında bitirir! İşte eylemin bir görüntüsü:

Şekil 1.2. Trinity Matrisi tarar



MPAA bunu öğrenip peşimize gözcüler ya da avukatlar göndermezse, <https://nmap.org/movies.html> adresinde tüm hack olayını gösteren bir terminal görüntüleme videosu mevcuttur.

### MadHat in Wonderland (MadHat Harikalar Diyarında)

Bu hikaye öncekilerden farklı olarak gerçekdir. Sık sık Nmap kullanan ve katkıda bulunan Lee "MadHat" Heath tarafından yazılan bu hikaye, Heath'in büyük bir şirkette günlük kullanım için Nmap'i nasıl geliştirdiğini ve özelleştirdiğini anlatmaktadır. Gerçek açık kaynak ruhuyla, bu değerli komut dosyalarını Web sitesinde yayınladı. Kurumsal kimliği korumak için IP adresleri değiştirilmiştir. Bu bölümün geri kalanı kendi sözlerinden oluşmaktadır.

Son birkaç on yılımı bilgisayar öğrenerek geçirdikten ve teknik destekten sistem yöneticiliğine ve büyük bir İnternet şirketinde Bilgi Güvenliği Sorumlusu olarak hayalimdeki işe yükseldikten sonra kendimi bir sorunla karşı karşıya buldum. Tüm IP alanımız için güvenlik izleme sorumluluğu bana verilmiştir. Birkaç yıl önce

başladığında bu sayı dünya çapında neredeyse 50.000 ana bilgisayarı ve o zamandan bu yana iki katına çıktı.

Aylık veya üç aylık değerlendirmelerin bir parçası olarak tüm bu makineleri olası güvenlik açıklarına karşı taramak yeterince zor olurdu, ancak yönetim bunun günlük olarak yapılmasını istediler. Saldırganlar yeni ortaya çıkan bir güvenlik açığından yararlanmak için bir hafta ya da bir ay beklemeyeceklerdir, bu yüzden ben de bu açığı bulmak ve yamamak için o kadar bekleyemem.

Araçları araştırırken port tarayıcıım olarak hemen Nmap'i seçtim. Yaygın olarak en iyi tarayıcı olarak kabul edilir ve ben zaten yıllardır ağlarda sorun gidermek ve güvenliği test etmek için kullanıyorum. Daha sonra Nmap çıktısını toplamak ve çalıştırımlar arasındaki farkları yazdırma için bir yazılıma ihtiyacım vardı. HD Moore'un Nlog'u da dahil olmak üzere mevcut birkaç aracı değerlendirdim. Ne yazık ki bunların hiçbirinin değişiklikleri istediğim şekilde izlemiyordu. Bir yönlendirici veya güvenlik duvarı erişim kontrol listesi yanlış yapılandırıldığında veya bir ana bilgisayar uygunsuz içeriği herkese açık olarak paylaştığında bunu bilmem gerekiyordu. Ayrıca bu diğer çözümlerin ölçeklenebilirliği konusunda da endişeliydim, bu yüzden sorunu kendim çözmeye karar verdim.

Ortaya çıkan ilk sorun hızdı. Ağlarımız dünya çapında yer alıyor, ancak taramayı yapmam için bana yalnızca ABD merkezli tek bir ana bilgisayar sağlandı. Çok durumda, siteler arasındaki güvenlik duvarları taramayı önemli ölçüde yavaşlattı. 100.000 ana bilgisayarın tamamının taraması 30 saatten fazla sürüyordu ki bu da günlük bir tarama için kabul edilemez bir süreydı. Bu yüzden dzünelerce Nmap işlemini paralel olarak çalıştırın nmap-wrapper adlı bir komut dosyası yazdım ve tarama süresini işletim sistemi algılaması da dahil olmak üzere on beş saatte indirdim.

Bir sonraki sorun çok fazla veriyle uğraşmaktı. Ölçeklenebilirlik ve veri madenciliği nedenleriyle bir SQL veritabanı en iyi yaklaşım gibi görünüyordu, ancak zaman baskısı nedeniyle bu fikirden vazgeçmek zorunda kaldım. Gelecekteki bir sürüm bu desteği ekleyebilir. Bunun yerine, her gün için her C sınıfı adres aralığının sonuçlarını saklamak için düz bir dosya kullandım. Bu bilgileri ayırtmanın ve saklamanın en güçlü ve genişletilebilir yolu Nmap XML formatıydı, ancak basit komut dosyalarından ayırtılması çok kolay olduğu için "grepable" (-oG seçeneği) formatını seçtim. Raporlama amacıyla ana bilgisayar başına zaman damgaları da saklanır. Bunlar, yöneticiler makine veya hizmet çökmelerinden tarayıcıyı sorumlu

tutmaya çalıştığında oldukça yardımcı olmuştur. Taramanın sabah 9:45'te çalıştığına dair kanıtım varken, sabah 7:12'de bir hizmet çökmesi olduğunu inandırıcı bir şekilde iddia edemezler.

#### Örnek 1.4. nmap-diff tipik çıktısı

```
> nmap-diff.pl -c3
5 IPs showed changes

10.12.4.8 (ftp-box.foocompany.biz)
  21/tcp  open  ftp
  80/tcp  open  http
  443/tcp open  https
  1027/tcp open  IIS
+ 1029/tcp open  ms-lsa
  38292/tcp open  landesk-cba
OS: Microsoft Windows Millennium Edition (Me)
  Windows 2000 Professional or Advanced Server
  or Windows XP

10.16.234.3 (media.foocompany.biz)
  80/tcp  open  http
+ 554/tcp  open  rtsp
+ 7070/tcp open  realserver

192.168.10.186 (testbox.foocompany.biz)
+ 8082/tcp open  blackice-alerts
OS: Linux Kernel 2.4.0 - 2.5.20

172.24.12.58 (mtafoocompany.biz)
+ 25/tcp  open  smtp
OS: FreeBSD 4.3 - 4.4PRERELEASE

172.23.76.22 (media2.foocorp.biz)
  80/tcp  open  http
  1027/tcp open  IIS
+ 1040/tcp open  netsaint
  1755/tcp open  wms
  3372/tcp open  msdtc
  6666/tcp open  irc-serv
  7007/tcp open  afs3-bos
OS: Microsoft Windows Millennium Edition (Me)
  Windows 2000 Professional or Advanced Server
  or Windows XP
```

Şirket içi bir güvenlik sempozyumunda bu yeni sistemi gösterdiğimde yönetim ve personel çok etkilendi. Ancak rahat durmama izin vermek yerine yeni özellikler istemeye başladılar. Posta ve web sunucularının sayılarını, büyümeye tahminlerini ve daha fazlasını istediler. Bu verilerin hepsi taramalarda mevcuttu, ancak erişilmesi zordu. Bu yüzden verileri sorgulamayı çok daha kolay hale getiren nmap-report adlı başka bir Perl betiği oluştururdum. Açık portlar veya işletim sistemleri gibi özellikleri alıyor ve belirli bir günde eşleşen tüm sistemleri buluyor.

Güvenlik izlemesine yönelik bu yaklaşımla ilgili bir sorun, çalışanların hizmetleri her zaman IANA'ya kayıtlı resmi bağlantı noktalarına yerleştirmemesidir. Örneğin, 22 numaralı bağlantı noktasına (SSH) bir web sunucusu koyabilirler ya da tam tersini yapabilirler. Tam da bu sorunu nasıl çözeceğimi tartışırken, Nmap gelişmiş bir hizmet ve sürüm algılama sistemi ile çıktı (bkz. Bölüm 7, Hizmet ve Uygulama Sürüm Algılama). nmap-report artık port numarasına göre tahmin etmek yerine gerçek hizmetleri bildirmek için sürüm taramasını kullanan bir yeniden tarama özelliğine sahip. Gelecek sürümlerde sürüm algılamayı daha da entegre etmemi umuyorum. Örnek 1.5 nmap-report'un FTP sunucularını listelemesini göstermektedir.

#### Örnek 1.5. nmap-report uygulaması

```
> nmap-report -p21 -rV
[...]
172.21.199.76 (ftp1.foocorp.biz)
  21/tcp  open  ssl|ftp Serv-U ftpd 4.0

192.168.12.56 (ftp2.foocorp.biz)
  21/tcp  open  ftp      NcFTPd

192.168.13.130 (dropbox.foocorp.biz)
  21/tcp  open  ftp      WU-FTPD 6.00LS
```

Mükemmel olmaktan uzak olsalar da, bu betikler güvenliği etkileyen değişiklikler için büyük ağıları izlemede oldukça değerli olduklarını kanıtladılar. Nmap'in kendisi açık kaynak kodlu olduğu için, benim komut dosyalarımı da halka açmak adil görünüyor. Bunları <http://www.unspecific.com/nmap> adresinde ücretsiz olarak kullanıma sundum.

[1] Bu IP adresleri aslında çok çeşitli topçu, füze, tank ve diğer ölümcül silahları test etmek için kullanılan Birleşik Devletler Ordusu Yuma Deneme Alanı'na

kayıtlıdır. Kissadan hisse, yanlışlıkla son derece hassas bir ağı vurmamak için kimi taradığınız konusunda çok dikkatli olmanızdır. Bu hikayedeki tarama sonuçları aslında bu IP aralığından değildir.

[2] Sınıfsız Alanlar Arası Yönlendirme (CIDR) gösterimi, A sınıfı (CIDR /8), B sınıfı (CIDR /16) veya C sınıfı (CIDR /24) gösteriminden daha ayrıntılı ağları tanımlamak için kullanılan bir yöntemdir. Bkz. [https://en.wikipedia.org/wiki/Classless\\_Inter-Domain\\_Routing](https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing).

[3] Hedef ad sunucusunun Felix'in ad sunucusundan gelen şüpheli bir grup ters-DNS sorgusunu günlüğe kaydetmesi mümkün değildir, ancak çoğu kuruluş bu tür günlükleri analiz etmek bir yana tutmaz bile.

[3] Hedef ad sunucusunun Felix'in ad sunucusundan gelen şüpheli bir grup ters-DNS sorgusunu günlüğe kaydetmesi mümkün değildir, ancak çoğu kuruluş bu tür günlükleri analiz etmek bir yana tutmaz bile.

[4] stdout, Nmap'in başlatıldığı Unix xterm veya Windows komut penceresi gibi bir sistem için standart çıktı mekanizmasını temsil eden "C" gösterimidir.

[5] Önceki ekipten deri giyimli seksi bir saldırgan aslında oturumu başlattı. Hangi noktada öldüğü ve kalan görevleri Trinity'ye bıraktığı belli değil.

## **The Phases of an Nmap Scan (Bir Nmap Taramasının Aşamaları)**

Şimdi Nmap'in bazı uygulamalarını gördüğümüze göre, bir Nmap taraması çalıştığında ne olduğuna bakalım. Taramalar aşamalar halinde ilerler ve her aşama bir sonraki aşama başlamadan önce tamamlanır. Aşağıdaki aşama açıklamalarından da görebileceğiniz gibi, Nmap'te port taramasından çok daha fazlası vardır.

**Script pre-scanning** Komut dosyası ön taraması. Nmap Komut Dosyası Motoru (NSE), uzak sistemler hakkında daha fazla bilgi edinmek için özel amaçlı komut dosyalarından oluşan bir koleksiyon kullanır. NSE, --script veya -sC gibi seçeneklerle talep etmediğiniz sürece çalıştırılmaz ve ön tarama aşaması yalnızca buna ihtiyaç duyan komut dosyaları seçildiğinde gerçekleşir. Bu aşama, tek tek hedeflere karşı ayrı ayrı çalıştırılmak yerine Nmap yürütmesi başına yalnızca bir kez çalıştırılması gereken komut dosyaları içindir. Örnekler arasında yaygın ağ hizmetlerinden bilgi almak için yayın sorgularını kullanan dhcp-discover ve broadcast-dns-service-discovery yer alır. NSE Bölüm 9'da tam olarak

belgelenmiştir ve farklı aşamalar "Komut Dosyası Türleri ve Aşamaları" adlı bölümde ele alınmıştır.

**Target enumeration** Hedef numaralandırma. Bu aşamada Nmap, kullanıcı tarafından sağlanan ana bilgisayar belirticilerini araştırır; bunlar ana bilgisayar DNS adları, IP adresleri, CIDR ağ gösterimleri ve daha fazlasının bir kombinasyonu olabilir. Nmap'ten hedeflerinizi sizin için seçmesini istemek için (-iR) bile kullanabilirsiniz! Nmap bu belirteçleri tarama için IPv4 veya IPv6 adreslerinin bir listesine dönüştürür. Bu aşama daha ileri tarama için gerekli olduğundan atlanamaz, ancak Nmap'in ileri çözümleme yapmak zorunda kalmaması için yalnızca IP adreslerini geçerek işlemi basitleştirebilirsiniz. Eğer -sL -n seçeneklerini geçerseniz (ters-DNS çözümlemesi olmadan liste taraması), Nmap hedefleri yazdıracak ve daha fazla tarama yapmayacaktır. Bu aşama "Hedef Ana Bilgisayarları ve Ağları Belirleme" bölümünde ve "Liste Taraması (-sL)" bölümünde ele alınmıştır.

**Host discovery (ping scanning)** Ana bilgisayar keşfi (ping taraması). Ağ taramaları genellikle agdaki hangi hedeflerin çevrimiçi olduğunu ve dolayısıyla daha derin incelemeye değer olduğunu keşfederek başlar. Bu süreçte ana bilgisayar keşfi veya ping taraması denir. Nmap, hızlı ARP isteklerinden TCP, ICMP ve diğer prob türlerinin ayrıntılı kombinasyonlarına kadar birçok ana bilgisayar bulma tekniği sunar. Bu aşama varsayılan olarak çalıştırılır, ancak -Pn (ping yok) seçeneğini kullanarak atlayabilirsiniz (tüm hedef IP'lerin çevrimiçi olduğunu varsayıbilirsiniz). Ana bilgisayar bulma işleminden sonra çıkmak için -sn -n seçeneğini belirtin. Ana bilgisayar keşfi Bölüm 3'ün konusudur.

**Reverse-DNS resolution** Ters-DNS çözümlemesi. Nmap hangi ana bilgisayarların taranacağını belirledikten sonra, ping taramasıyla çevrimiçi bulunan tüm ana bilgisayarların ters-DNS adlarına bakar. Bazen bir ana bilgisayarın adı, işlevi hakkında ipuçları sağlar ve adlar, raporları yalnızca IP numaralarını sağlamak için daha okunaklı hale getirir. Bu adım -n (çözümleme yok) seçeneği ile atlanabilir veya -R (tümünü çözümle) seçeneği ile tüm hedef IP'leri (aşağı olanlar dahil) kapsayacak şekilde genişletilebilir. İsim çözümlemesi "DNS Çözümlemesi" adlı bölümde ele alınmıştır.

**Port scanning** Port tarama. Bu Nmap'in temel işlevidir. Problar gönderilir ve bu problara verilen yanıtlar (ya da yanitsızlıklar) uzak portları açık, kapalı ya da filtrelenmiş gibi durumlara sınıflandırmak için kullanılır. Bu kısa açıklama, Nmap'in

birçok tarama türünü, taramaların yapılandırılabilirliğini ve hız ve doğruluğu artırmaya yönelik algoritmaları kapsamaya başlamaz. Port taramasına genel bir bakış Bölüm 4'te yer almaktadır. Algoritmalar ve komut satırı seçenekleri hakkında ayrıntılı bilgi Bölüm 5'te yer almaktadır. Bağlantı noktası taraması varsayılan olarak gerçekleştirilir, ancak -sn seçeneğiyle bunu atlayabilir ve belirli komut satırı seçeneklerini (--traceroute ve --script gibi) belirterek daha sonraki traceroute ve kısmi Nmap Scripting Engine aşamalarından bazılarını gerçekleştirebilirsiniz.

**Version detection** Sürüm tespiti. Herhangi bir portun açık olduğu tespit edilirse, Nmap uzak sisteme hangi sunucu yazılımının çalıştığını belirleyebilir. Bunu, açık portlara çeşitli probalar göndererek ve yanıtları 6.500'den fazla bilinen hizmet imzasından oluşan binlerce veritabanıyla eşleştirerek yapar. Sürüm tespiti -sV seçeneği ile etkinleştirilir ve Bölüm 7'de tam olarak açıklanmıştır.

**OS detection** İşletim sistemi algılama. Eğer -O seçeneği ile istenirse, Nmap işletim sistemi tespitine geçer. Farklı işletim sistemleri ağ standartlarını çok farklı şekillerde uygular. Bu farklılıklarla öncelikle uzaktaki bir ana bilgisayarda çalışan işletim sistemini belirlemek genellikle mümkün değildir. Nmap, standart bir prob setine verilen yanıtları binden fazla bilinen işletim sistemi yanıtından oluşan bir veritabanıyla eşleştirir. İşletim sistemi tespiti Bölüm 8'de ele alınmaktadır.

**Traceroute** Traceroute. Nmap, --traceroute seçeneği ile etkinleştirilen optimize edilmiş bir traceroute uygulaması içerir. Nmap'in önceki keşif aşamaları tarafından belirlenen mevcut en iyi sonda paketlerini kullanarak paralel olarak birçok ana bilgisayara giden ağ yollarını bulabilir. Traceroute genellikle ara ana bilgisayarlar için başka bir ters-DNS çözümleme turu içerir. Daha fazla bilgi "Ana Bilgisayar Bulma" bölümünde bulunabilir.

**Script scanning** Komut dosyası taraması. Çoğu Nmap Scripting Engine (NSE) komut dosyası, ön tarama ve son tarama aşamaları yerine bu ana komut dosyası tarama aşaması sırasında çalışır. NSE, Lua programlama dili ve ağ bilgisi toplama için tasarlanmış standart bir kütüphane tarafından desteklenmektedir. Bu aşamada çalışan komut dosyaları genellikle etkileşimde bulundukları her hedef ana bilgisayar ve bağlantı noktası numarası için bir kez çalışır. Genellikle hizmet açıklarını tespit etme, kötü amaçlı yazılım bulma, veritabanlarından ve diğer ağ hizmetlerinden daha fazla bilgi toplama ve gelişmiş sürüm tespiti gibi görevleri yerine getirirler. NSE, --script veya -sC gibi seçeneklerle talep etmediğiniz sürece çalıştırılmaz.

**Output** Çıktı. Son olarak, Nmap topladığı tüm bilgileri toplar ve ekrana ya da bir dosyaya yazar. Nmap çeşitli biçimlerde çıktı yazabilir. Varsayılan, insan tarafından okunabilir biçim (etkileşimli biçim) genellikle bu kitapta sunulmaktadır. Nmap diğerlerinin yanı sıra XML tabanlı bir çıktı biçim de sunmaktadır. Çıktıların giriş ve çıkışları Bölüm 13'ün konusudur.

Daha önce de tartışıldığı gibi, Nmap bu aşamalardan hangilerinin çalıştırılacağını kontrol etmek için birçok seçenek sunar. Büyük ağların taranması için, Nmap ana bilgisayarları daha küçük gruplar halinde ele aldığından her aşama birçok kez tekrarlanır. Her grubu tamamen tarar ve bu sonuçları çıktı olarak verir, ardından bir sonraki ana bilgisayar grubuna geçer.

**Script post-scanning** Komut dosyası tarama sonrası. Nmap taramasını ve normal çıktısını tamamladıktan sonra, bu aşamadaki komut dosyaları sonuçları işleyebilir ve son raporları ve istatistikleri sunabilir. "Komut Dosyası Türleri ve Aşamaları" başlıklı bölüme bakın. Nmap henüz bu aşamada herhangi bir komut dosyası içermez, bu nedenle yalnızca kullanıcı kendi tarama sonrası komut dosyalarını ekler ve yürütürse çalışır.

## **Legal Issues (Yasal Sorunlar)**

Doğru kullanıldığında, Nmap ağınıza işgalcilerden korumanıza yardımcı olur. Ancak yanlış kullanıldığında, Nmap (nadır durumlarda) dava edilmenize, kovulmanız, atılmanız, hapse atılmanız veya İSS'niz tarafından yasaklanmanızına neden olabilir. Nmap'i başlatmadan önce bu yasal kılavuzu okuyarak riskinizi azaltın.

### **Is Unauthorized Port Scanning a Crime? (İzinsiz Port Tarama Suç mudur?)**

Nmap ile ağ taramanın yasal sonuçları karmaşıktır ve o kadar tartışmalıdır ki, üçüncü taraf kuruluşlar Şekil 1.3'te gösterildiği gibi konuya ilgili görüşlerini bildiren tişörtler ve tampon çıkartmaları bile basmıştır[6]. Bu konu aynı zamanda birçok tutkulu ancak çoğu zaman verimsiz tartışmalara ve alev savaşlarına da yol açmaktadır. Bu tür tartışmalara katılacak olursanız, birinin evinin kapısını çalmak veya kapısının ve pencerelerinin kilitli olup olmadığını test etmek gibi aşırı kullanılan ve uygun olmayan benzettmelerden kaçınmaya çalışın.

Şekil 1.3. Liman taramasının yasallığı ve ahlaklılığı konusunda güçlü görüşler



Port taramasının yasadışı olmaması gerekiği fikrine katılmakla birlikte, bir tişörtten hukuki tavsiye almak nadiren akıllica olacaktır. Aslında, bir yazılım mühendisi ve yazardan tavsiye almak sadece biraz daha iyidir. Yasaların sizin özel durumunuza nasıl uygulandığını daha iyi anlamak için yetki alanınızdaki yetkili bir avukatla konuşun. Bu önemli feragatnameyi aradan çıkararak, yardımcı olabilecek bazı genel bilgiler vereceğim.

Nmap kullanırken tartışmalardan kaçınmanın en iyi yolu, herhangi bir tarama başlatmadan önce hedef ağ temsilcilerinden her zaman yazılı izin almaktır. İSS'nizin bunu fark etmesi halinde (ya da hedef yöneticilerin yanlışlıkla onlara bir kötüye kullanım raporu göndermesi halinde) size sorun çıkarma ihtimali hala vardır, ancak bunun çözülmesi genellikle kolaydır. Bir sizma testi gerçekleştirirken, bu yetkilendirme İş Bildiriminde yer almalıdır. Kendi şirketinizi test ederken, bu faaliyetin açıkça iş tanımınıza girdiğinden emin olun. Güvenlik danışmanları, bu durumlar için en iyi uygulamaları sunan mükemmel Açık Kaynak Güvenlik Testi Metodolojisi Kılavuzu'na (OSSTMM) aşina olmalıdır.

Hukuk ve (özellikle) ceza mahkemesi davaları Nmap kullanıcıları için kabus senaryosu olsa da, bunlar çok nadirdir. Sonuçta, hiçbir Birleşik Devletler federal yasası port taramasını açıkça suç saymamaktadır. Çok daha sık rastlanan bir

durum ise hedef ağın bir taramayı fark etmesi ve taramanın başlatıldığı ağ hizmet sağlayıcısına (İSS'niz) bir şikayet göndermesidir. Çoğu ağ yöneticisi her gün ağlarından seken çok sayıda taramayı önemsemez ya da fark etmez, ancak birkaçı şikayet eder. Tarama kaynağı İSS, bildirilen IP adresi ve zamana karşılık gelen kullanıcının izini sürebilir, ardından kullanıcıyı azarlayabilir ve hatta hizmetten atabilir. İzinsiz port taraması bazen sağlayıcının kabul edilebilir kullanım politikasına (AUP) aykırıdır. Örneğin, büyük kablolu modem İSS'si Comcast'in AUP'si şöyle der:

Ağ problema veya port tarama araçlarına yalnızca bir ev ağı ile birlikte kullanıldığında veya hedef ana bilgisayar ve/veya ağ tarafından açıkça izin verildiğinde izin verilir. Herhangi bir nedenle yetkisiz port taraması kesinlikle yasaktır.

Bir İSS yetkisiz port taramasını açıkça yasaklamasa bile, bazı "anti-hacking" hükümlerinin geçerli olduğunu iddia edebilir. Elbette bu port taramayı yasadışı yapmaz. Tamamen yasal ve (Amerika Birleşik Devletleri'nde) anayasal olarak korunan birçok faaliyet İSS'ler tarafından yasaklanmıştır. Örneğin, yukarıda alıntılanan AUP, kullanıcıların "makul bir kişinin sakıncalı, saldırgan, uygunsuz, pornografik, ... utanç verici, üzücü, kaba, nefret dolu, irksal veya etnik olarak saldırgan veya başka bir şekilde uygunsuz olduğunu düşünebileceği herhangi bir bilgi veya materyali, bu materyalin veya yayılmasının yasa dışı olup olmadığına bakılmaksızın" iletmesini, depolamasını veya yayılmasını da yasaklamaktadır. Başka bir deyişle, bazı İSS'ler birilerini rahatsız edebilecek ya da kızdırabilecek her türlü davranıştı yasaklamaktadır[7]. Diğer insanların ağlarının gelişigüzel taraması bu potansiyele sahiptir. Yine de bu tür tartışmalı bir tarama yapmaya karar verirseniz, bunu asla işten, okuldan veya refahınız üzerinde önemli kontrolü olan başka bir hizmet sağlayıcısından yapmayın. Bunun yerine ticari bir geniş bant veya kablosuz sağlayıcı kullanın. DSL bağlantınızı kaybetmek ve sağlayıcı değiştirmek zorunda kalmak hafif bir sıkıntıdır, ancak kovulmak veya işten atılmaktan ölçülemeyecek kadar iyidir.

Port taraması (takip eden bilgisayar korsanlığı saldıruları olmadan) içeren yasal davalar nadir olsa da, bunlar gerçekleşmektedir. En kayda değer vakalardan biri, Cherokee County, Georgia acil durum 911 sisteminin bakımı için devam eden bir danışmanlık sözleşmesi olan Scott Moulton adlı bir adamlı ilgiliydi. Aralık 1999'da, Canton, Georgia Polis Departmanını E911 Merkezine bağlayan bir yönlendirici kurmakla görevlendirildi. Bunun E911 Merkezinin güvenliğini tehlkeye atabileceği endişesiyle Scott ilgili ağlarda bazı ön port taramaları başlattı. Bu süreçte, VC3 adlı

rakip bir danışmanlık firmasının sahibi olduğu ve bakımını yaptığı Cherokee County web sunucusunu taradı. Taramayı fark eden firma Scott'a e-posta göndermiş, Scott da 911 Merkezi için çalıştığını ve güvenlik testi yaptığı söylemiştir. VC3 daha sonra bu faaliyeti polise bildirdi. Scott E911 bakım sözleşmesini kaybetti ve Amerika Bilgisayar Dolandırıcılığı ve Kötüye Kullanımı Yasası Bölüm 1030(a)(5)(B)'yi ihlal ettiği iddiasıyla tutuklandı. Bu yasa, "korunan bir bilgisayara yetkisi olmadan kasıtlı olarak erişen ve bu tür bir davranış sonucunda zarara neden olan" (ve diğer gereklilikleri karşılayan) herkes için geçerlidir. VC3 tarafından talep edilen zarar, port taraması ve ilgili faaliyetlerin araştırılması için harcanan zamanı içeriyordu. Scott, VC3'e hakaret davası açtı ve VC3 de Bilgisayar Dolandırıcılığı ve Kötüye Kullanımı Yasası ile Georgia Bilgisayar Sistemleri Koruma Yasasını ihlal ettiği gerekçesiyle karşı dava açtı.

Scott'a karşı açılan hukuk davası duruşmadan önce reddedildi, bu da davanın tamamen degersiz olduğu anlamına geliyordu. Karar birçok Nmap kullanıcısının yüzünü güldürdü:

"Mahkeme, davacının davalının sunucuları üzerinde izinsiz port taraması ve verim testi yapmasının Georgia Bilgisayar Sistemlerini Koruma Yasası veya Bilgisayar Dolandırıcılığı ve Suistimali Yasası'nı ihlal etmediğine karar vermiştir."-Civ. Kanun. No. 1:00-CV-434-TWT (N.D. Ga. 6 Kasım 2000)

Bu hukuk davasında heyecan verici bir zaferdi, ancak Scott hala cezai suçlamaları beklemekteydi. Neyse ki moralini yüksek tuttu ve nmap-hackers mail listesine aşağıdaki notu gönderdi:

Bilgisayar alanındaki uzmanların haklarını savunmak ve korumak konusunda bilgisayar toplumuna bir nebze de olsa faydalı olabildiğim için gurur duyuyorum, ancak böyle bir çabayı desteklemek SON DERECE maliyetli ve bu durumdan memnun değilim. Ancak savaşmaya devam edeceğim ve özellikle de sadece işimi yaparken port taramasının yasadışı bir yanı olmadığını kanıtlayacağım.

Sonunda, ceza mahkemesi de aynı sonuca vardı ve tüm suçlamalar düşürüldü. Scott sonunda haklı çıksa da, altı haneli yasal faturalara maruz kaldı ve mahkeme sistemi boyunca stresli yıllara katlandı. İşin iyi tarafı, avukatlarını ilgili teknik konular hakkında eğitmek için çok fazla zaman harcadıktan sonra Scott başarılı bir adli tıp hizmetleri şirketi kurdu.

Moulton davası (yasal emsal olmasa da) iyi bir örnek teşkil etse de, farklı mahkemeler veya durumlar daha kötü sonuçlara yol açabilir. Pek çok eyaletin

kendi bilgisayar suistimal yasaları olduğunu ve bunlardan bazlarının yetkisiz olarak uzaktaki bir makineye ping atmayı bile yasadışı hale getirebileceğini unutmayın[8].

Diğer ülkelerdeki yasalar da açıkça farklılık göstermektedir. Örneğin, 17 yaşındaki bir genç Finlandiya'da bir bankanın port taramasını yaptığı için bilgisayara izinsiz giriş teşebbüsünden suçlu bulundu. Hedefin soruşturma masraflarını karşılamak üzere para cezasına çarptırıldı. Moulton'un kararı, VC3 makinesi gerçekten çökmüş olsaydı ve yasanın gerektirdiği 5.000 dolarlık hasar rakamını haklı gösterebilse de farklı olabilirdi.

Diğer ucta ise İsraili bir yargıç 2004 yılı başlarında Avi Mizrahi'yi Mossad gizli servisini taramaktan beraat ettirdi. Hatta Yargıcı Abraham Tennenbaum kararında Avi'yi övmüştür:

Bir bakıma, Web sitelerinin güvenlik açıklarını kontrol eden Internet sörfçüleri kamu yararına hareket etmektedir. Eğer niyetleri kötü niyetli değilse ve herhangi bir zarara neden olmuyorlarsa, övülmeleri bile gereklidir.

2007 ve 2008 yıllarında Almanya ve İngiltere'de geniş kapsamlı yeni siber suç yasaları yürürlüğe girmiştir. Bu yasalar "bilgisayar korsanlığı araçlarının" dağıtımını, kullanımını ve hatta bulundurulmasını yasaklamayı amaçlamaktadır. Örneğin, Birleşik Krallık'ta Bilgisayarın Kötüye Kullanılması Yasası'nda yapılan değişiklik "bir programın [Bilgisayarın Kötüye Kullanılması Yasası ihlalinin] işlenmesi ya da işlenmesine yardımcı olmak için kullanılacağına inanarak [bu programı] tedarik etmeyi ya da tedarik etmeyi teklif etmeyi" yasadışı hale getirmektedir. Bu yasalar şimdiden bazı güvenlik aracı yazarlarının dükkânlarını kapatmalarına ya da projelerini başka ülkelere taşımalarına neden oldu. Sorun şu ki, çoğu güvenlik aracı hem etik profesyoneller (beyaz şapkalılar) tarafından ağlarını savunmak için hem de siyah şapkalılar tarafından saldırmak için kullanılabilir. Bu tehlikeli yasalar, araç yazarının ya da kullanıcının niyetine dayanmaktadır ki bu da özneldir ve tahmin edilmesi zordur. Nmap Internet güvenliğine yardımcı olmak için tasarlandı, ancak tutuklanmaktan ve niyetimi bir yargıç ve jüriye savunmak zorunda kalmaktan nefret ederim, özellikle de Almanya gibi dilini bile bilmediğim yabancı bir ülkede. Bu yasaların Nmap kadar yaygın ve popüler araçları etkilemesi pek olası değil, ancak daha küçük araçlar ve bilgisayar suçluları tarafından daha yaygın olarak kötüye kullanılanlar (istismar çerçeveleri gibi) üzerinde caydırıcı bir etkisi oldu.

Port taramasının yasal durumu ne olursa olsun, çok sayıda şikayet gelmesi halinde İSS hesapları sonlandırılmaya devam edecktir. İSS kötüye kullanım raporlarından

veya hukuki/cezai suçlamalardan kaçınmanın en iyi yolu, ilk etapta hedef ağ yöneticilerini rahatsız etmekten kaçınımaktır. İşte bazı pratik öneriler:

- Tarama izniniz olduğundan emin olun. Muhtemelen ağ taramasının en az %90'ı tartışmaya açık değildir. Kendi makinenizi veya yönettiğiniz ağları taradığınız için nadiren rahatsız edilirsiniz. Tartışmalar diğer ağları tararken ortaya çıkar. Bu tür bir ağ araştırması yapmak için birçok neden (iyi ve kötü) vardır. Belki de ortak paylaşılan dosyaları (FTP, SMB, WWW, vb.) aramak için yatakhaneinizdeki veya bölümünüzdeki diğer sistemleri tariyorsunuzdur. Ya da belki sadece belirli bir yazılıçının IP adresini bulmaya çalışıyorsunuzdur. Favori web sitenizi başka hizmetler sunup sunmadıklarını görmek için veya hangi işletim sistemini çalıştırdıklarını merak ettiğiniz için taramış olabilirsiniz. Belki sadece bağlantıyı test etmeye çalışıyorsunuz ya da kredi kartı bilgilerinizi e-ticaret şirketine vermeden önce hızlı bir güvenlik kontrolü yapmak istiyorsunuz. İnternet araştırması yapıyor olabilirsiniz. Ya da bir sizma girişimine hazırlık olarak ilk keşfi mi yapıyorsunuz? Uzaktaki yöneticiler gerçek niyetinizi nadiren bilirler ve bazen şüphelenirler. En iyi yaklaşım önce izin almaktır. Yönetici olmayan rollere sahip birkaç kişinin, tüm şirket veya kampüste izinsiz bir tarama başlatarak ağ güvensizliğini "kanıtlamaya" karar verdikten sonra başlarının derde girdiğini gördüm. Yöneticiler kendilerine önceden sorulduğunda, sabahın 3:00'ünde büyük bir saldırısı altında olduklarını iddia eden bir IDS alarmıyla uyandırıldıklarında olduğundan daha fazla işbirliği yapma eğilimindedirler. Bu nedenle mümkün olduğunda, bir ağ taramadan önce yazılı izin alın. Adrian Lamo, New York Times'a güvenlik açıklarını daha sonra gazetecilere anlatmak yerine test etmelerini isteseydi muhtemelen hapse girmekten kurtulurdu. Ne yazık ki hayır demiş olabilirler. Bu cevaba hazırlıklı olun.
- Taramanızı mümkün olduğunca sıkı hedefleyin. İnternete bağlı herhangi bir makine, çoğu yöneticinin bu tür Internet arka plan gürültüsünü görmezden gelmesine yetecek kadar düzenli olarak taranır. Ancak yeterli sayıda ağ taramak ya da çok gürültülü/izinsiz taramalar yapmak şikayet oluşturma olasılığını artırır. Bu nedenle, yalnızca web sunucularını ariyorsanız, her makinedeki 65.536 TCP bağlantı noktasının tümünü taramak yerine -p80 belirtin. Yalnızca kullanılabilir ana bilgisayarları bulmaya çalışıyorsanız, tam port taraması yerine bir Nmap ping taraması yapın. Bir /24 netblock yeterli olduğunda bir CIDR /16 (65K ana bilgisayar) taramayın. Rastgele tarama modu

artık sonsuza kadar çalışmak yerine ana bilgisayar sayısını belirten bir argüman alıyor. Bu yüzden -iR 10000 yerine -iR 1000'i düşünün, eğer ilki yeterliyse. T insane yerine varsayılan zamanlamayı (hatta -T kibar) kullanın. Sürüm algılama (-sV) veya NSE (--script) gibi gürültülü ve nispeten müdahaleci taramalardan kaçının. Benzer şekilde, SYN taraması (-sS) bağlantı taramasından (-sT) daha sessiz olmakla birlikte aynı bilgiyi sağlar ve genellikle daha hızlıdır.

- Daha önce de belirtildiği gibi, iş veya okul bağlantılarınızdan tartışmalı bir şey yapmayın. Niyetiniz iyi olsa bile, yetkili biri (örneğin patron, dekan) sizin kötü niyetli bir bilgisayar korsanı olduğunuzu karar verirse kaybedecek çok şeyiniz olur. Port taramanın ne anlama geldiğini bile bilmeyen birine yaptıklarınızı gerçekten açıklamak istiyor musunuz? Bir paravan, hücresel veri veya konut geniş bant hesabı için ayda 40 dolar harcayın. Böyle bir hesaptan birini rahatsız ederseniz bunun sonuçları daha az şiddetli olmakla kalmaz, aynı zamanda hedef ağ yöneticilerinin kitlesel pazar sağlayıcılarını şikayet etme zahmetine bile girme olasılığı daha düşüktür. Ayrıca ilgili AUP'yi okuyun ve buna göre bir sağlayıcı seçin. Sağlayıcınız (yukarıda bahsedilen Comcast gibi) yetkisiz port taramasını ve "saldırgan" materyallerin yayınlanması yasaklıyorsa, bu faaliyet nedeniyle atılırsanız şaşırmayın. Genel olarak, bir servis sağlayıcıya ne kadar çok ödeme yaparsanız o kadar uzlaşmacı olurlar. Bir T1 sağlayıcısının, birisi port taraması yapıldığını bildirdi diye haber vermeden bağlantınızı kesmesi pek olası değildir. Çevirmeli bağlantı ya da ev tipi DSL/kablo sağlayıcısı pekala bunu yapabilir. Tarama başka biri tarafından yapılmış olsa bile bu durum gerçekleşebilir.
- Nmap, gizli taramalar için kaynak-IP sahteciliği, tuzak tarama ve daha yeni boşta tarama tekniği dahil olmak üzere birçok seçenek sunar. Bunlar IDS'den kaçınma bölümünde ele alınmıştır. Ancak her zaman bir değişim tokusu olduğunu unutmayın. Evinizden uzaktaki açık bir WAP'tan 17 tuzakla tarama başlatırsanız ve sonraki taramaları dokuz açık proxy zinciri üzerinden yaparsanız bulunmanız daha zor olur. Ancak biri sizi bulursa, niyetinizden oldukça şüphelenenecektir.
- Tarama yapmak için her zaman meşru bir nedeniniz olsun. Rahatsız olan bir yönetici önce size yazabilir (veya İSS'niz şikayetini size iletebilir) ve faaliyet için bir tür gerekçe bekleyebilir. Yukarıda ele alınan Scott Moulton vakasında, VC3 ilk olarak Scott'a e-posta göndererek neler olup bittiğini sordu. Eğer

Scott'ın verdiği cevapta tatmin olmuş olsalardı, meseleler hukuk ve ceza davasına dönüşmek yerine orada durabilirdi. Araştırma amacıyla internetin büyük bir bölümünü taradığında, projeyi tanımlayan bir ters-DNS adı kullanıyorum ve bu IP adresinde ayrıntılı bilgi ve devre dışı bırakma talimatları içeren bir web sunucusu çalıştırıyorum.

Ayrıca, yardımcı ve müteakip eylemlerin genellikle niyet kanıtı olarak kullanıldığını unutmayın. Tek başına bir port taraması her zaman bir saldırıyla işaret etmez. Ancak bir IIS istismarının yakından takip ettiği bir port taraması, niyeti yüksek sesle ve net bir şekilde yayırlar. Bu önemlidir çünkü kovuşturma (veya kovma, ihraç etme, şikayet etme vb.) kararları genellikle sadece bir bileşene (port taraması gibi) değil olayın bütününe dayanır.

Dramatik vakalardan biri Walter Nowakowski adlı Kanadalı bir adamla ilgiliydi ve görünüşe göre Kanada'da birisinin güvenli olmayan Wi-Fi ağrı üzerinden internete eriştiği için iletişim hırsızlığıyla (Kanada Ceza Kanunu Bölüm S.342.1) suçlanan ilk kişi oldu. Binlerce Kanadalı "savaş şoförü" bunu her gün yapıyor, peki neden o seçildi? Yan eylemler ve niyet yüzünden. İddiaya göre, tek yönlü bir caddede belden aşağısı çıplak, elinde dizüstü bilgisayarıyla yanlış yöne giderken, yukarıda bahsedilen güvenli olmayan kablosuz erişim noktası üzerinden çocuk pornosu indirirken yakalandı. Görünüşe göre polis bu eylemi yeterince korkunç bulmuş ki ilgili suçlamalar için beyin fırtınası yapmış ve çocuk pornografisiyle ilgili birçok suçlamaya iletişim hırsızlığını da eklemiş.

Benzer şekilde, port taramasıyla ilgili suçlamalar genellikle en korkunç vakalar için ayrılmıştır. Paranoyak yöneticiler tarandıklarını polise bildirdiklerinde bile kovuşturma (ya da başka bir işlem) son derece nadirdir. Moulton davasında savcıları motive eden şey muhtemelen 911 acil servisin işin içinde olmasıydı. Yazارınız bu kitabı yazarken milyonlarca internet sunucusunu taramış ve ondan daha az şikayet almıştır.

Tüm bu bölümü özetlemek gerekirse, port taramanın yasal olup olmadığı sorusunun basit bir cevabı yoktur. Her ne kadar istesem de kesin olarak "port taraması asla suç değildir" diyemem. Yasalar yargı bölgeleri arasında önemli ölçüde farklılık gösterir ve davalar kendi özel ayrıntılarına bağlıdır. Olaylar neredeyse aynı olsa bile, farklı hakim ve savcılar bunları her zaman aynı şekilde yorumlamazlar. Sadece dikkatli olunmasını tavsiye edebilir ve yukarıdaki önerileri tekrarlayabilirim.

Test amacıyla, scanme.nmap.org ana bilgisayarını tarama izniniz var. Zaten birkaç örnekte kullanıldığını fark etmiş olabilirsiniz. Bu iznin yalnızca Nmap ile taramayı içerdiğini ve istismarları veya hizmet reddi saldırılardan test etmeyi içermediğini unutmayın. Bant genişliğini korumak için, lütfen bu ana bilgisayara karşı günde bir düzineden fazla tarama başlatmayın. Bu ücretsiz tarama hedefi hizmeti kötüye kullanılrsa, kaldırılacak ve Nmap verilen ana bilgisayar adı/IP'yi çözemedi: scanme.nmap.org.

### **Can Port Scanning Crash the Target Computer/Networks? (Port Tarama Hedef Bilgisayarı/Ağları Çökertebilir mi?)**

Nmap, hedef ağları çökertmek için tasarlanmış herhangi bir özelliğe sahip değildir. Genellikle hafif adımlar atmaya çalışır. Örneğin, Nmap düşen paketleri algılar ve ağın aşırı yüklenmesini önlemek için bunlar oluştukunda yavaşlar. Nmap ayrıca bozuk paketler de göndermez. IP, TCP, UDP ve ICMP başlıklarını her zaman uygundur, ancak hedef ana bilgisayarın paketleri beklemesi gerekmek. Bu nedenlerden dolayı, hiçbir uygulama, ana bilgisayar veya ağ bileşeni bir Nmap taramasına dayalı olarak çökmelidir. Eğer çökerse, bu sistemde satıcı tarafından onarılması gereken bir hatadır.

Nmap tarafından çökertilen sistemlere ilişkin raporlar nadirdir, ancak bunlar gerçekleşmektedir. Bu sistemlerin birçoğu muhtemelen ilk etapta kararsızdı ve Nmap onları en tepeye itti ya da tamamen tesadüf eseri bir Nmap taramasıyla aynı anda çöktüler. Diğer durumlarda, kötü yazılmış uygulamaların, TCP/IP yoğunlarının ve hatta işletim sistemlerinin belirli bir Nmap komutu verildiğinde tekrarlanabilir bir şekilde çöktüğü gösterilmiştir. Yeni ekipmanlar nadiren bu sorunlarla piyasaya sürüldüğünden, bunlar genellikle eski eski cihazlardır. Akıllı şirketler, cihazları sevkiyattan önce test etmek için Nmap ve diğer birçok yaygın ağ aracını kullanır. Bu tür yayın öncesi testleri ihmali edenler, sorunu genellikle bir kutu internete ilk kez dağıtıldığında erken beta testlerinde öğrenirler. Belirli bir IP'nin Internet arka plan gürültüsünün bir parçası olarak taranması nadiren uzun sürer. Sistemleri ve cihazları en son satıcı yamaları ve ürün yazılımı ile güncel tutmak, makinelerinizin bu sorunlara karşı duyarlığını azaltırken, ağınızin güvenliğini ve kullanılabilirliğini de artıracaktır.

Birçok durumda, bir makinenin belirli bir taramadan çöktüğünü bulmak değerli bir bilgidir. Sonuçta, saldırganlar Nmap'in kendisini veya kendi özel komut dosyalarını kullanarak Nmap'in yapabildiği her şeyi yapabilirler. Cihazlar taranırken

çökmemelidir ve eğer çökerse, satıcılar bir yama sağlamaları için baskı yapılmalıdır. Bazı kullanım senaryolarında, kırılgan makinelerin çökertilerek tespit edilmesi istenmez. Bu gibi durumlarda, olumsuz etki riskini azaltmak için çok hafif tarama yapmak isteyebilirsiniz. İşte birkaç öneri:

- Bağlantı taraması (-sT) yerine SYN taraması (-sS) kullanın. Web sunucuları gibi kullanıcı modu uygulamaları ilkini nadiren algılayabilir, çünkü hepsi çekirdek uzayında işlenir ve bu nedenle hizmetlerin çökmek için bir bahanesi yoktur.
- Sürüm taraması (-sV) ve bazı NSE betiklerimiz (-sC veya --script) kötü yazılmış uygulamaları çökertme riski taşıır. Benzer şekilde, bazı hatalı işletim sistemlerinin OS parmak izi alındığında (-O) çöktüğü bildirilmiştir. Özellikle hassas ortamlar için veya sonuçlara ihtiyaç duymadığınız durumlarda bu seçenekleri atlayın.
- T2 veya daha yavaş (-T1, -T0) zamanlama modlarını kullanmak, taramanızı önemli ölçüde yavaşlatşa da, bir port taramasının sisteme zarar verme olasılığını azaltabilir. Eski Linux kutularında, çok sık erişildiklerinde hizmetleri geçici olarak engelleyen bir identd daemon vardı. Bu bir port taramasında olabileceği gibi meşru yüksek yük durumlarında da olabilir. Daha yavaş zamanlama burada yardımcı olabilir. Bu yavaş zamanlama modları sadece son çare olarak kullanılmalıdır çünkü taramaları büyülü sırasına göre veya daha fazla yavaşlatabilirler.
- Taranan port ve makine sayısını gerekli olan en az sayıda makine ile sınırlayın. Taranan her makinenin çökme ihtimali çok düşüktür ve bu nedenle makine sayısını azaltmak şansınızı artırır. Taranan port sayısını azaltmak, ağ cihazlarının yanı sıra son ana bilgisayarlara yönelik riskleri de azaltır. Birçok NAT/güvenlik duvarı cihazı, her port araştırması için bir durum girdisi tutar. Çoğu, tablo dolduğunda eski girdilerin süresini doldurur, ancak bunun yerine ara sıra (acinası) uygulamalar çöker. Taranan bağlantı noktalarını ve ana bilgisayarları azaltmak durum girdilerinin sayısını azaltır ve böylece bu kırılgan ve kusurlu cihazların ayakta kalmasına yardımcı olabilir.

### Nmap Copyright (Nmap Telif Hakkı)

Nmap açık kaynak olsa da, hala saygı duyulması gereken bir telif hakkı lisansına sahiptir. Özgür yazılım olarak, Nmap ayrıca hiçbir garanti taşımaz. Bu konular "Yasal Uyarılar" adlı bölümde çok daha ayrıntılı olarak ele alınmaktadır. Nmap'i özel

yazılımlar ve cihazlar içinde paketlemek ve kullanmak isteyen şirketlerin, Nmap lisansını yanlışlıkla ihlal etmemeleri için bu bölüm okumaları özellikle tavsiye edilir. Neyse ki Nmap Projesi, ihtiyaç duyan şirketler için ticari yeniden dağıtım lisansları satmaktadır.

[6] Bunlar artık feshedilmiş olan AmericanSushi.Com'dan alınmıştır.

[7] Comcast AUP, bu ilk yayınlandıktan sonra geliştirilmiştir. En son sürüm <http://www.comcast.net/terms/use/> adresinde mevcuttur.

[8] Bu konuda avukat Ethan Preston tarafından yazılmış mükemmel bir makaleye <http://grove.ufl.edu/~techlaw/vol6/issue1/preston.html> adresinden ulaşabilirsiniz. Ayrıca <http://www.mcandl.com/computer-security.html> adresinde güvenlik bilgilerinin ve açıklarının yayınlanmasının yasal risklerine ilişkin mükemmel bir makale yazmıştır.

## **The History and Future of Nmap (Nmap'in Geçmişİ ve Geleceği)**

Netcat, tcpdump ve John the Ripper gibi birçok eski ve sevilen güvenlik aracı yıllar içinde çok fazla değişmedi. Wireshark, Metasploit, Cain and Abel ve Snort gibi diğerleri ise piyasaya sürüldükleri günden bu yana sürekli olarak geliştirilmektedir. Nmap bu ikinci kategoridedir. 1997'de sadece Linux'a özel basit bir port tarayıcı olarak piyasaya sürüldü. Sonraki 16+ yıl boyunca işletim sistemi algılama, sürüm algılama, Nmap Scripting Engine, Windows portu, grafik kullanıcı arayüzü, Ncat, Nping, Ndifff ve daha fazlası dahil olmak üzere sayısız değerli özelliği filizlendirdi. Bu hızlı gelişim temposunu gelecekte de sürdürmeyi planlıyoruz!

### **The History of Nmap (Nmap'in Tarihçesi)**

Bu bölüm, 16 yıllık Nmap tarihindeki en önemli kilometre taşlarının bir zaman çizelgesini sunmaktadır. Tüm önemli Nmap değişiklikleri (binlercesi) için Nmap Changelog'u okuyun. Nmap'in eski sürümleri <https://nmap.org/dist/> adresinde ve eski sürümleri <https://nmap.org/dist-old/> adresinde bulunabilir.

- 1 Eylül 1997 - Nmap ilk olarak Phrack dergisi Sayı 51, Madde 11'de yayınlandı. Yeni sürümler planlanmadığı için bir sürüm numarası yoktur. Nmap yaklaşık 2.000 satır uzunluğundadır ve derlenmesi gcc -O6 -o nmap nmap.c -lm kadar basittir.
- 5 Eylül 1997 - Yoğun talep üzerine Phrack kodunun biraz daha geliştirilmiş bir sürümü 1.25 olarak yayınlandı. Gzipped tarball 28KB'dır. Sürüm 1.26 (48KB) 19

gün sonra yayınlandı.

- 11 Ocak 1998 - Insecure.Org tescil edildi ve Nmap, DataHaven Project ISP'deki önceki evinden buraya taşındı.
- 14 Mart 1998 - Renaud Deraison bana bir güvenlik tarayıcısı yazdığını bildirmek için yazıyor ve bazı Nmap kaynak kodlarını kullanıp kullanamayacağını soruyor. Tabii ki evet dedim. Dokuz gün sonra bana Nessus'un ön sürümünü gönderdi ve "3I33t H4ck3rZ için değil, sistem yöneticileri için tasarlandığını" belirtti.
- 12 Aralık 1998 - Nmap sürüm 2.00 halka açık olarak yayınlandı ve birkaç ay süren özel geliştirmeden sonra ilk kez Nmap OS algılaması sunuldu. Teknikleri açıklayan bir makale Phrack 54, Madde 9'da yayınlandı. Bu noktada Nmap birçok dosyaya bölünmüştür, yaklaşık 8.000 satır koddan oluşur, özel bir CVS revizyon kontrol sisteminde tutulur ve tarball boyutu 275KB'dir. nmap-hackers posta listesi başlatılır ve daha sonra 80.000'den fazla üyeye ulaşır.
- 11 Nisan 1999 - Nmap 2.11BETA1 yayınlandı. Bu, geleneksel komut satırı kullanımına alternatif olarak bir grafik kullanıcı arayüzü içeren ilk sürümdür. Birlikte verilen Unix-only NmapFE GUI orijinal olarak Zach Smith tarafından yazılmıştır. Bazı insanlar bunu seviyor, ancak çoğu komut satırı yürütmemeyi tercih ediyor.
- 28 Nisan 2000 - Nmap 2.50 yayınlandı. Bu noktada tarball 461KB'a ulaşmıştır. Bu sürüm -T agresif, doğrudan SunRPC taraması ve Pencere ve ACK tarama yöntemleri gibi zamanlama modlarını içerir.
- 28 Mayıs 2000 - Gerhard Rieger nmap-dev listesine Nmap için geliştirdiği yeni bir "protokol taramasını" anlatan bir mesaj gönderdi ve hatta bir yama da ekledi. Bu o kadar güzel ki, 12 saatten kısa bir süre sonra Nmap 2.54BETA1'i onun yamasıyla birlikte yayınladım.
- 7 Aralık 2000 - Nmap 2.54BETA16, Microsoft Windows üzerinde derlenen ve çalıştırılan ilk resmi sürüm olarak yayınlandı. Windows taşıma işi Ryan Permeh ve Andy Lutomirski tarafından yapıldı.
- 9 Temmuz 2001 - Nmap IP ID boşta tarama Nmap 2.54BETA26 ile tanıtıldı. Tekniği açıklayan bir makale de eşzamanlı olarak yayınlandı. Bu son derece havalı (her zaman pratik olmasa da) tarama tekniği "TCP Idle Scan (-sl)" adlı bölümde açıklanmaktadır.

- 25 Temmuz 2002 - Netscape/AOL'deki işimden ayrıldım ve hayalimdeki işe, Nmap üzerinde tam zamanlı çalışmaya başladım.
- 31 Temmuz 2002 - Nmap 3.00 yayınlandı. Tarball 922K'dır. Bu sürüm Mac OS X desteği, XML çıktısı ve çalışma süresi algılaması içerir.
- 28 Ağustos 2002 - Nmap C'den C++'a dönüştürüldü ve Nmap 3.10ALPHA1 sürümünün bir parçası olarak IPv6 desteği eklendi.
- 15 Mayıs 2003 - Nmap, Trinity'nin bir elektrik santralini hacklemek ve dünyayı kurtarmak için onu kullandığı (ardından gerçek bir SSH istismarı) The Matrix Reloaded filminde yer aldı. Bu, Nmap'in daha önce hiç görümediği kadar tanınmasına yol açar. Hollywood'un dikkatini çeker ve Nmap filmlerdeki bilgisayar korsanlığı sahneleri için standart bir dekor haline gelir. Nmap daha sonra The Bourne Ultimatum, Die Hard 4, The Girl with the Dragon Tattoo, Dredd ve diğer birçok filmde görüldü. Tüm bu filmlerden detaylar ve ekran görüntüleri <https://nmap.org/movies/> adresinde mevcuttur.
- 21 Temmuz 2003 - Nmap hizmet/sürüm tespitinin ilk uygulamasını (Bölüm 7, Hizmet ve Uygulama Sürüm Tespit) bitirdim ve Nmap 3.40PVT1 olarak birkaç düzine üst düzey Nmap geliştiricisi ve kullanıcısına yayıldım. Bunu, sistemi geliştirdikçe ve imzalar ekledikçe önumüzdeki birkaç ay içinde 16 özel sürüm daha takip edecek.
- 16 Eylül 2003 - Nmap hizmet tespiti, Nmap 3.45'in bir parçası olarak herkese açık olarak yayınlandı.
- 20 Şubat 2004 - Nmap 3.50 yayınlandı. Tarball artık 1,571KB'dır. SCO Corporation'ın Nmap'i yeniden dağıtmaması yasaklandı çünkü GPL'ye uymayı reddediyorlar. Nmap'i kaldırmak için Caldera sürüm ISO'larını yeniden oluşturmaları gerekiyor. Bu sürüm, paket izleme ve UDP ping seçeneklerinin yanı sıra bir işletim sistemi sınıflandırma sistemi içerir.
- 31 Ağustos 2004 - Çekirdek Nmap port tarama motoru Nmap 3.70 için yeniden yazıldı. Ultra\_scan olarak adlandırılan yeni motor, hem doğruluğu hem de hızı artırmak için önemli ölçüde geliştirilmiş algoritmalar ve paralelleştirme desteği içeriyor. Farklılıklar özellikle katı güvenlik duvarlarının arkasındaki ana bilgisayarlar için çarpıcıdır.

- 25 Haziran 2005 - Google, 10 üniversite ve yüksek lisans öğrencisine Google'ın Summer of Code girişiminin bir parçası olarak yaz boyunca Nmap üzerinde tam zamanlı çalışmaları için sponsor olur. Projeler arasında Netcat'in Ncat (Chris Gibson) adlı modern bir yeniden uygulaması, ikinci nesil bir işletim sistemi tespit sistemi (Zhao Lei), daha sonra Zenmap (Adriano Monteiro Marques) olacak yeni bir çapraz platform GUI ve <https://seclists.org/nmap-hackers/2005/8> adresinde açıklanan diğer birçok harika proje yer alıyor.
- 8 Eylül 2005 - Nmap, 3.90 sürümüyle birlikte ham ethernet çerçevesi gönderme desteği kazanıyor. Bu, ARP taraması ("ARP Taraması (-PR)" bölümüne bakın) ve MAC adresi sahteciliğinin yanı sıra Microsoft tarafından Windows XP SP2'de getirilen ham IP paketi yasağından kaçınmaya olanak tanır.
- 31 Ocak 2006 - Nmap 4.00 yayınlandı. Tarball artık 2,388KB'dir. Bu sürüm, isteğe bağlı tamamlama tahminleri sağlamak için çalışma zamanı etkileşimi, bir Windows çalıştırılabilir yükleyicisi, GTK2'yi desteklemek için NmapFE güncellemeleri ve çok daha fazlasını içerir.
- 24 Mayıs 2006 - Google, SoC programının bir parçası olarak 10 Nmap yaz geliştiricisine daha sponsor oldu. Zhao ve Adriano projelerini daha da geliştirmek için 2006 SoC'un bir parçası olarak geri döndüler. Diman Todorov, Nmap Scripting Engine'in (Bölüm 9, Nmap Scripting Engine) geliştirilmesine yardımcı olmak üzere desteklenmektedir. Bu ve diğer yedi yetenekli öğrenci ve projeleri <https://seclists.org/nmap-hackers/2006/9> adresinde açıklanmaktadır.
- 24 Haziran 2006 - İki yıllık geliştirme ve test sürecinin ardından 2. nesil işletim sistemi tespit sistemi Nmap 4.20ALPHA1'e entegre edildi. Bu yeni sistem, ilk nesil sistemin sekiz yıl önce piyasaya sürülmüşinden bu yana öğrendiğimiz her şeye ve tasarladığımız yeni fikirlere dayanmaktadır. Bölüm 8, Uzaktan İşletim Sistemi Tespiti'nde açıklanan yeni sistem, eski DB'nin boyutuna ulaşmamız 2,5 yıl sürmesine rağmen, ilk nesilden çok daha doğru ve ayrıntılı olduğunu kanıtlıyor.
- 10 Aralık 2006 - Nmap Scripting Engine, Nmap 4.21ALPHA1'in bir parçası olarak yayınlandı. NSE, kullanıcıların çok çeşitli ağ görevlerini otomatikleştirmek için basit komut dosyaları yazmalarına (ve paylaşmalarına) olanak tanır. Bu ilk sürüm, güvensiz SSHv1 ve SSLv2 protokollerini tespit etmek, identd'den hizmet sahibi bilgilerini almak ve DNS sunucularının

özyinelemeli sorguları destekleyip desteklemediğini test etmek gibi basit görevler için 23 komut dosyası içerir.

- 28 Mayıs 2007 - Google, SoC programının bir parçası olarak altı yaz Nmap geliştiricisine sponsor oldu. Sponsor olan öğrenciler arasında David Fifield de vardı ve yaz sona erdikten sonra da Nmap'in ortak geliştiricisi olmaya devam etti. Hatta bu kitabı hazırlanmasına da büyük ölçüde yardımcı oldu! Nmap öğrencileri ve projeleri <https://seclists.org/nmap-hackers/2007/3> adresinde listelenmiştir.
- 8 Temmuz 2007 - Zenmap grafik ön ucu (o zamanlar Ümit olarak adlandırılıyordu) geliştirildi ve test için Nmap 4.22SOC1 sürümüne entegre edildi. Saygideğer NmapFE GUI kaldırıldı. Zenmap, Bölüm 12, Zenmap GUI Kullanıcı Kılavuzu'nda ele alınmıştır.
- 13 Aralık 2007 - Nmap'in 10. yıldönümünü kutlamak için Nmap 4.50 yayınlandı!
- 1 Haziran 2008 - Nmap 4.65 yayınlandı ve ilk kez çalıştırılabilir bir Mac OS X yükleyicisi içeriyor. Nmap kaynak tarball'u artık dört megabayttır. Bu sürüm 41 NSE komut dosyası, 1.307 işletim sistemi parmak izi ve 4.706 sürüm algılama imzası içerir.
- 18 Ağustos 2008 - Nmap projesi dördüncü Summer of Code'u şimdije kadarki en yüksek başarı yüzdesiyle tamamladı (desteklenen yedi öğrenciden altısı başarılı oldu). Öğrenciler, <https://seclists.org/nmap-dev/2008/q4/193> adresinde açıkladığı gibi Zenmap, Nmap Scripting Engine, OS detection ve Ncat'i büyük ölçüde geliştirdiler.
- 8 Eylül 2008 - Nmap 4.75, 4.68'e göre neredeyse 100 önemli iyileştirme ile piyasaya sürüldü. Bunlar arasında, keşfedilen ağ cihazlarının etkileşimli bir diyagramını çizen Zenmap ağ topolojisi görüntüleyicisi de yer alıyor. Nmap'in ("Ağ Eşleyici") 11 yıllık dağıtımından sonra, artık gerçekten ağ haritaları çizebiliyor. Bu sürüm ayrıca, birkaç taramanın birleşik bir görünümde birleştirilmesine olanak tanıyan tarama toplamayı da içerir. Bu özelliklerin her ikisi de Bölüm 12, Zenmap GUI Kullanıcı Kılavuzu'nda açıklanmıştır). Bu sürüm ayrıca önceki ay Black Hat ve Defcon'da sunduğum Worldscan projemden port sıklığı verilerini de içeriyor. Bu, Nmap'in deneysel olarak en popüler olduğu gösterilen bağlantı noktalarına odaklanmasını sağlar.

- 1 Ocak 2009 - Bu kitabın ilk versiyonu olan Nmap Ağ Taraması yayınlandı. Port tarama üzerine 450+ sayfalık bir kitapla benden başka kimsenin ilgilenip ilgilenmeyecegi sorusunun yaniti yankilan bir evet oldu! İlk 18 ayda 10.000'den fazla kopya satıldı. Diğer yavncılar tarafından Almanca, Korece ve Brezilya Portekizcesi dillerinde de yayımlanmıştır.
- 23 Ocak 2009 - Komut satırından bir ağ üzerinden veri okuyan ve yazan özelliklerle dolu bir ağ yardımcı programı olan Ncat eklendi. Klasik Netcat aracının SSL, bağlantı yeniden yönlendirme ve proxying gibi modern ağ özellikleri için destek ekleyen bir yeniden uygulamasıdır. Bölüm 17, Ncat Başvuru Kılavuzu'na bakın.
- 23 Ocak 2009 - İki Nmap taramasının sonuçlarını karşılaştırın, ağıları düzenli olarak taramayı ve değişiklikleri gösteren bir rapor (XML veya metin) oluşturmayı kolaylaştıran Ndiff yardımcı programı eklendi. Bölüm 16, Ndiff Referans Kılavuzu'na bakın.
- 30 Mart 2009 - İnternetteki milyonlarca makineye bulaşan Conficker solucanını uzaktan tespit etmek için özel bir Nmap sürümü (4.85BETA5) üretildi. Talep o kadar yüksektir ki Nmap çok fazla bant genişliği kullandığı gereklisiyle Dreamhost'un "sınırsız bant genişliği" web barındırma planından çıkarılır. Kaliforniya Üniversitesi, San Diego devreye girer ve hızlı bir ayna sunucu sağlar. Conficker solucanının bir sonraki sürümü, virüs bulaşmış kullanıcıların Insecure.Org ve Nmap.Org'a ulaşmasını engelliyor. Bunu Nmap'in etkinliğine bir övgü olarak kabul ediyoruz.
- 12 Haziran 2009 - Nmap'e SCTP port tarama ve host keşif desteği eklendi. SCTP çoğunlukla telefonla ilgili uygulamalar için kullanılan bir 4. katman protokolüdür.
- 16 Temmuz 2009 - Nmap 5.00 yayınlandı. Tarball 27MB'den fazladır ve 2,003 işletim sistemi parmak izi, 5,512 sürüm algılama imzası ve 59 NSE komut dosyası içerir.
- 24 Ağustos 2009 - Nmap projesi beşinci Summer of Code'u tamamladı. İlk kez tüm öğrenciler başarılı oldu! Nping'in yanı sıra ağ kimlik doğrulama kırıcısı Ncrack'i yarattılar ve Nmap Scripting Engine, Ncat ve Zenmap'te önemli iyileştirmeler yaptılar.

- 29 Mart 2010 - Ağ paketi oluşturma, yanıt analizi ve yanıt süresi ölçümü için kullanışlı bir araç olan Nping eklendi (bkz. Bölüm 18, Nping Referans Kılavuzu).
- 28 Temmuz 2010 - David Fifield ve ben Black Hat ve Defcon konferanslarında canlı demolar da dahil olmak üzere Nmap Scripting Engine'de Uzmanlaşma üzerine sunum yaptık (video: <https://nmap.org/presentations/>).
- 17 Ağustos 2010 - Eğlenmek için en iyi bir milyon web sitesini taramak, popülerliklerine göre ölçeklendirilmiş bir web sitesi simgeleri mozaiği olan Icons of the Web'i yayinallyadığımızda bir yan sanat projesine yol açıyor. New York Times, The Telegraph, Slashdot, Gizmodo, Engadget ve diğer pek çok gazetede haber oldu. Washington D.C.'deki Newseum'da dev bir versiyonu sergileniyor ve Guinness Dünya Rekorları kitabında bile yer aldı.
- 28 Ocak 2011 - Nmap 5.50 yayınlandı. Sadece eğlence için eski Gopher protokolü için destek ekledik, ancak ciddi iyileştirmeler arasında NSE komut dosyalarını 177'ye üç katına çıkarmak ve yeni bir Zenmap komut dosyası seçim arayüzü eklemek vardı. Ayrıca Nping'e ağ analizi ve hata ayıklamayı kolaylaştıran yeni bir yankı modu özelliği ekledik ("Yankı Modu" adlı bölüme bakın). Performans da yüksek bir öncelikti, bazı tarama süreleri saatlerden dakikalara indirildi ve kıyaslama taramamızda bellek tüketimi %90 azaltıldı.
- 4 Kasım 2011 - 3.000'den fazla Nmap kullanıcısına favori güvenlik araçları için anket yaptıktan sonra, SecTools.org web sitemizi yeni verilerle yayındık. Önceki sitelerimiz sadece statik HTML dosyalarıydı, ancak bu yeni sürüm, bilgileri daha iyi paylaşmak için etkileşimli derecelendirme ve inceleme özellikleri içeriyor. Sitede profili çıkarılan en iyi 125 araçtan 49'u listeye yeni eklenmiştir. Elbette Nmap araç ailesi potansiyel çıkar çatışmaları nedeniyle sitenin dışında tutulmuştur.
- 21 Mayıs 2012 - Nmap 6 yayınlandı! Tarball 54MB'den fazladır ve 3,572 işletim sistemi parmak izi, 8,165 sürüm algılama imzası ve 348 NSE komut dosyası içerir.

### **The Future of Nmap (Nmap'in Geleceği)**

Nmap'in geçmişini kataloglamak kolay olsa da geleceği belirsizdir. Nmap büyük bir geliştirme planı ile başlamadı ve önceki zaman çizelgesindeki kilometre taşlarının çoğu bir ya da iki yıldan daha önceden planlanmadı. Gelecekte internetin ve ağın alacağı şekli tahmin etmeye çalışmak yerine, şu anda nerede olduğunu yakından

inceliyor ve yakın vadede Nmap için neyin en yararlı olacağına karar veriyorum. Dolayısıyla Nmap'in bundan 10 yıl sonra nerede olacağına dair hiçbir fikrim yok, ancak her zamanki kadar popüler ve canlı olmasını bekliyorum. Nmap topluluğu, Nmap'i gitmesi gereken her yere yönlendirecek kadar büyütür. Nmap, Windows XP SP2'de ham paket desteğinin aniden kaldırılması, ağ filtreleme uygulamaları ve teknolojisindeki dramatik değişiklikler ve IPv6'nın yavaşça ortaya çıkması gibi daha önce de eğri toplarla karşılaştı. Bunların her biri Nmap'te önemli değişiklikler gerektirdi ve gelecekteki ağ değişikliklerini kucaklamak veya en azından bunlarla başa çıkmak için ayısını yapmamız gerekecek.

10 yıllık plan henüz havada olsa da, önümüzdeki birkaç yıl için planlarımız ve yol gösterici önceliklerimiz var:

- **Nmap Scripting Engine** —Nmap Scripting Engine - NSE'nin yetenekleri ve popülerliği patladı. 2010 ortası itibarıyle, bir önceki yıla göre %68 artışla 131 komut dosyası içermektedir. NSE komut dosyaları da sürekli artan kütüphane ve altyapı özellik desteği sayesinde daha güçlü hale geliyor. Önümüzdeki birkaç yıl içinde NSE'nin 500'den fazla betiğe ulaştığını görmek istiyorum. Nmap yakında, mevcut hostrule ve portrule komut dosyaları gibi tek bir ana bilgisayara veya açık bağlantı noktasına bağlı olmayan prescan ve postscan komut dosyalarını destekleyecektir.
- **Web infrastructure improvements** — Web sitelerinin taraması - Nmap ayrıca web taramasını idare etme becerisinde de büyüyecektir. Nmap ilk geliştirildiğinde, farklı hizmetler genellikle dinledikleri port numarasıyla tanımlanan ayrı daemonlar olarak sağlanır. Artık birçok yeni hizmet HTTP üzerinden çalışmakta ve port numarası yerine URL yolu adıyla tanımlanmaktadır. Bilinen URL yollarının taraması birçok yönden port taramasına (ve Nmap'in uzun yıllardır yaptığı SunRPC taramasına) benzer. Nmap Scripting Engine zaten birçok web tarama özelliğini destekliyor, ancak http brute force şifre kırma, daha iyi spidering desteği, bir HTML/XML ayrıştırıcı, daha iyi web uygulaması parmak izi ve HTTP ve SOCKS proxy'leri aracılığıyla belirli istekleri proxyleme yeteneği görmek istiyorum. Nmap bu proxy'ler üzerinden de port taraması yapabilmelidir.
- **Online scanning web service** — Çevrimiçi tarama web hizmeti - Kullanıcıların kendi makinelerini ve ağlarını "buluttan" tarayarak dışarıdan bir bakış açısıyla nelere maruz kaldıklarını görmelerini sağlayan bir hizmet üzerinde çalışıyoruz.

Tekrarlanan taramalar, e-posta uyarlarında gönderilen değişikliklerle planlanabilir veya çevrimiçi olarak göz atılabilir.

Bu üst düzey önceliklere ek olarak, aklımızda bazı somut görevler de var:

- Nmap'i sadece İngilizce yerine birçok farklı dili destekleyecek şekilde uluslararasılaştırın. Zenmap ön ucu bunu zaten yapıyor ve "Kendi Dilinizde Zenmap" adlı bölümde açıklandığı gibi beş dili destekliyor.
- Regresyonları (hataları) daha kolay yakalamak için bir test koşum sistemi ekleyin. Ncat böyle bir sistemi zaten uygulamıştır ve son derece yararlı olduğu kanıtlanmıştır.
- İşletim Sistemi ve Sürüm tespit sistemleri, diğer araçlarla daha iyi uyumluluk için NIST Ortak Platform Numaralandırma (CPE) standardını destekleyecek şekilde artırılabilir.
- Zenmap yakında, bir makinede yüklü Nmap sürümü tarafından desteklenen komut dosyalarına ve komut dosyası argümanlarına göz atmayı ve seçmeyi kolaylaştıran gelişmiş bir komut dosyası seçim arayüzü sunacak.
- İnsanlara Nmap'i tanıtmak ve etkili bir şekilde kullanmaları için püf noktaları ve teknikleri öğretmek için Nmap ile ilgili daha fazla video üretmeyi, üretilmesini teşvik etmeyi ve dağıtmayı planlıyoruz.

Şu anda üzerinde çalıştığımız ve yakın gelecekte planladıklarımızın daha kaba ve düşük seviyeli bir listesi için <https://nmap.org/svn/todo/nmap.txt> adresindeki Nmap todo listesini her zaman okuyabilirsiniz.

İşletim sistemi ve sürüm algılama gibi geçmişteki en havalı Nmap özelliklerinden bazıları gizlice geliştirildi ve sürpriz bir şekilde yayınlandı. Önümüzdeki yıllarda bunlardan daha fazlasını bekleyebilirsiniz çünkü çok eğlenceliler!

## **Chapter 2. Obtaining, Compiling, Installing, and Removing Nmap (Bölüm 2. Nmap'in Edinilmesi, Derlenmesi, Kurulması ve Kaldırılması)**

### **İçindekiler**

- Introduction (Giriş)

- Testing Whether Nmap is Already Installed (Nmap'in Zaten Yükü Olup Olmadığını Test Etme)
  - Command-line and Graphical Interfaces (Komut Satırı ve Grafik Arayüzler)
  - Downloading Nmap (Nmap İndirme)
  - Verifying the Integrity of Nmap Downloads (Nmap İndirmelerinin Bütünlüğünü Doğrulama)
  - Obtaining Nmap from the Subversion (SVN) Repository (Nmap'i Subversion (SVN) Deposundan Edinme)
- Linux/Unix Compilation and Installation from Source Code (Linux/Unix Kaynak Kodundan Derleme ve Kurulum)
  - Configure Directives (Yönergeleri Yapılandırma)
  - Environment Variables (Ortam Değişkenleri)
  - If You Encounter Compilation Problems (Derleme Sorunlarıyla Karşılaşırsanız)
- Linux Distributions (Linux Dağıtımları)
  - RPM-based Distributions (Red Hat, Mandrake, SUSE, Fedora) (RPM tabanlı dağıtımlar (Red Hat, Mandrake, SUSE, Fedora))
  - Updating Red Hat, Fedora, Mandrake, and Yellow Dog Linux with Yum (Red Hat, Fedora, Mandrake ve Yellow Dog Linux'un Yum ile Güncellenmesi)
  - Debian Linux and Derivatives such as Ubuntu (Debian Linux ve Ubuntu gibi türevleri)
  - Other Linux Distributions (Diğer Linux Dağıtımları)
- Windows
  - Windows Self-installer (Windows Kendi Kendine Yükleyici)
  - Command-line Zip Binaries (Komut Satırı Zip İkilileri)
    - Installing the Nmap zip binaries (Nmap zip ikili dosyalarının yüklenmesi)
  - Compile from Source Code (Kaynak Koddan Derleme)

- Executing Nmap on Windows (Windows'ta Nmap Çalıştırma)
- Apple Mac OS X
  - Executable Installer (Yürüttülebilir Yükleyici)
  - Compile from Source Code (Kaynak Koddan Derleme)
    - Compile Nmap from source code (Nmap'i kaynak koddan derleme)
    - Compile Zenmap from source code (Zenmap'i kaynak koddan derleme)
  - Third-party Packages (Üçüncü Taraf Paketleri)
  - Executing Nmap on Mac OS X (Mac OS X üzerinde Nmap Çalıştırma)
- Other Platforms (BSD, Solaris, AIX, AmigaOS) (Diğer Platformlar (BSD, Solaris, AIX, AmigaOS))
  - FreeBSD / OpenBSD / NetBSD (FreeBSD / OpenBSD / NetBSD)
    - OpenBSD Binary Packages and Source Ports Instructions (OpenBSD İkili Paketleri ve Kaynak Portları Talimatları)
    - FreeBSD Binary Package and Source Ports Instructions (FreeBSD İkili Paket ve Kaynak Portları Talimatları)
    - NetBSD Binary Package Instructions (NetBSD İkili Paket Talimatları)
  - Oracle/Sun Solaris (Oracle/Sun Solaris)
  - IBM AIX
  - AmigaOS
  - Other proprietary UNIX (HP-UX, IRIX, etc.) (Diğer tescilli UNIX (HP-UX, IRIX, vb.))
- Removing Nmap (Nmap'i Kaldırma)

## Introduction (Giriş)

Nmap genellikle tek bir komutla kurulabilir veya yükseltilerilebilir, bu nedenle bu bölümün uzunluğunun sizi korkutmasına izin vermeyin. Çoğu okuyucu, kendilerini

ilgilendiren bölümlere doğrudan atlamak için içindekiler tablosunu kullanacaktır. Bu bölüm, hem kaynak kod derleme hem de ikili kurulum yöntemleri dahil olmak üzere birçok platformda Nmap'in nasıl kurulacağını açıklamaktadır. Nmap'in grafik ve komut satırı sürümleri açıklanmış ve karşılaştırılmıştır. Fikrinizi değiştirmeniz durumunda Nmap kaldırma talimatları da verilmiştir.

### **Testing Whether Nmap is Already Installed (Nmap'in Zaten Yükü Olup Olmadığını Test Etme)**

Nmap'i edinmeye yönelik ilk adım, ona zaten sahip olup olmadığını kontrol etmektir. Birçok ücretsiz işletim sistemi dağıtımını (çoğu Linux ve BSD sistemi dahil) Nmap paketleriyle birlikte gelir, ancak bunlar varsayılan olarak yükü olmayabilir. Unix sistemlerinde, bir terminal penceresi açın ve nmap --version komutunu çalıştırmayı deneyin. Eğer Nmap varsa ve PATH'inizde bulunuyorsa, Örnek 2.1'dekine benzer bir çıktı görmeniz gereklidir.

Örnek 2.1. Nmap'i kontrol etme ve sürüm numarasını belirleme

```
felix~> nmap --version
Nmap version 4.76 ( https://nmap.org )
felix~>
```

Nmap sisteme mevcut değilse (veya PATH'iniz yanlış ayarlanmışsa), nmap gibi bir hata mesajı bildirilir: Komut bulunamadı gibi bir hata mesajı bildirilir. Yukarıdaki örnekte görüldüğü gibi, Nmap komuta sürüm numarasını yazdırarak yanıt verir (burada 4.76).

Sisteminizde zaten bir Nmap kopyası olsa bile, <https://nmap.org/download.html> adresinde bulunan en son sürümü yükseltmeyi düşünmelisiniz. Yeni sürümler genellikle daha hızlı çalışır, önemli hataları düzeltir ve güncellenmiş işletim sistemi ve hizmet sürümü algılama veritabanlarına sahiptir. Halihazırda sisteminizde bulunan sürümden bu yana yapılan değişiklıkların bir listesini <https://nmap.org/changelog.html> adresinde bulabilirsiniz. Bu belgedeki Nmap çıktı örnekleri eski sürümler tarafından üretilen çıktılarla eşleşmeyecek.

### **Command-line and Graphical Interfaces (Komut Satırı ve Grafik Arayüzler)**

Nmap geleneksel olarak bir Unix kabuğundan veya (daha yakın zamanda) Windows komut isteminden çalıştırılan bir komut satırı aracı olmuştur. Bu,

uzmanların bir grup yapılandırma paneli ve dağınık seçenek alanları arasında manevra yapmak zorunda kalmadan tam olarak istedikleri şeyi yapan bir komutu hızlı bir şekilde yürütütmelerine olanak tanır. Bu aynı zamanda Nmap'in kodlanması kolaylaştırır ve kullanıcı topluluğu arasında yararlı komutların kolayca paylaşılmasını sağlar.

Komut satırı yaklaşımının bir dezavantajı, yeni ve seyrek kullanıcılar için korkutucu olabileceğidir. Nmap yüzden fazla komut satırı seçeneği sunar, ancak çoğu kullanıcının görmezden gelebileceği belirsiz özellikler veya hata ayıklama kontrolleridir. GUI arayüzü tercih eden kullanıcılar için birçok grafiksel önyüz oluşturulmuştur. Nmap geleneksel olarak Unix için NmapFE adında basit bir GUI içeriyordu, ancak bu 2007'de 2005'ten beri geliştirmekte olduğumuz Zenmap ile değiştirildi. Zenmap, özellikle sonuç görüntüleme konusunda NmapFE'den çok daha güçlü ve etkilidir. Zenmap'in sekme tabanlı arayüzü, sonuçları aramanıza ve sıralamanıza ve ayrıca çeşitli şekillerde (ana bilgisayar ayrıntıları, ham Nmap çıktısı ve bağlantı noktaları / ana bilgisayarlar) göz atmanızı olanak tanır. Linux, Windows, Mac OS X ve diğer platformlarda çalışır. Zenmap, Bölüm 12, Zenmap GUI Kullanıcı Kılavuzu'nda derinlemesine ele alınmıştır. Bu kitabın geri kalanı komut satırı Nmap çağrılarına odaklanmaktadır. Komut satırı seçeneklerinin nasıl çalıştığını anladıkten ve çıktıyı yorumlayabildikten sonra, Zenmap veya diğer mevcut Nmap GUI'lerini kullanmak kolaydır. Nmap'in seçenekleri, ister radyo düğmelerinden ve menülerden seçin ister komut satırına yazın aynı şekilde çalışır.

### **Downloading Nmap (Nmap İndirme)**

Nmap.Org, Nmap ve Zenmap için Nmap kaynak kodunu ve ikili dosyalarını indirmek için resmi kaynaktır. Kaynak kodu bzip2 ve gzip sıkıştırılmış tar dosyalarında dağıtılr ve ikili dosyalar Linux (RPM formatı), Windows (NSIS çalıştırılabilir yükleyici) ve Mac OS X (.dmg disk görüntüsü) için mevcuttur. Tüm bunları <https://nmap.org/download.html> adresinde bulabilirsiniz.

### **Verifying the Integrity of Nmap Downloads (Nmap İndirmelerinin Bütünlüğünü Doğrulama)**

İnternetten indirilen dosyaların bütünlüğü konusunda paranoyak olmak çoğu zaman işe yarar. Sendmail (örnek), OpenSSH (örnek), tcpdump, Libpcap, BitchX, Fragrouter ve diğerleri gibi popüler paketlere kötü niyetli truva atları bulaşmıştır. Özgür Yazılım Vakfı, Debian ve SourceForge'daki yazılım dağıtım siteleri de başarıyla ele geçirilmiştir. Bu Nmap'in başına hiç gelmemiştir, ancak her zaman

dikkatli olunmalıdır. Bir Nmap sürümünün gerçekliğini doğrulamak için, <https://nmap.org/dist/sigs/?C=M&O=D> adresindeki Nmap imzaları dizininde sürüm için yayınlanan PGP müstakil imzalarına veya kriptografik karmalara (SHA1 ve MD5 dahil) bakın.

En güvenli doğrulama mekanizması müstakil PGP imzalarıdır. İmzalama anahtarı üretim sunucularında asla saklanmadığından, web sunucusunu başarıyla ele geçen biri bile bir truva atı sürümünü taklit edemez ve düzgün bir şekilde imzalayamaz. Çok sayıda uygulama PGP imzalarını doğrulayabilse de ben GNU Privacy Guard'ı (GPG) öneriyorum.

Nmap sürümleri, büyük anahtar sunuculardan veya [https://svn.nmap.org/nmap/docs/nmap\\_gpgkeys.txt](https://svn.nmap.org/nmap/docs/nmap_gpgkeys.txt) adresinden elde edilebilen özel bir Nmap Projesi İmzalama Anahtarı ile imzalanır. Benim anahtarım da bu dosyaya dahil edilmiştir. Anahtarlar gpg --import nmap\_gpgkeys.txt komutu ile içe aktarılabilir. Bunu yalnızca bir kez yapmanız gereklidir, ardından gelecekteki tüm Nmap sürümlerini bu makineden doğrulayabilirsiniz. Anahtarlar güvenmeden önce, parmak izlerinin Örnek 2.2'de gösterilen değerlerle eşleştiğini doğrulayın.

#### Örnek 2.2. Nmap ve Fyodor PGP Anahtar Parmak İzlerinin Doğrulanması

```
flog-> gpg --fingerprint nmap fyodor
pub 1024D/33599B5F 2005-04-24
    Key fingerprint = BB61 D057 C0D7 DCEF E730 996C 1AF6 EC50 3359 9B5F
uid          Fyodor <fyodor@insecure.org>
sub 2048g/D3C2241C 2005-04-24

pub 1024D/6B9355D0 2005-04-24
    Key fingerprint = 436D 66AB 9A79 8425 FDA0 E3F8 01AF 9F03 6B93 55D0
uid          Nmap Project Signing Key (https://insecure.org/)
sub 2048g/A50A6A94 2005-04-24
```

Her Nmap paketi indirme dosyası için (örn. nmap-4.76.tar.bz2 ve nmap-4.76-win32.zip), sigs dizininde adına .asc eklenmiş karşılık gelen bir dosya vardır (örn. nmap-4.76.tar.bz2.asc). Bu müstakil imza dosyasıdır.

Anahtarlığınızdaki uygun PGP anahtarı ve indirilen ayrılmış imza dosyası ile, bir Nmap sürümünü doğrulamak Örnek 2.3'te gösterildiği gibi tek bir GPG komutu gerektirir. Bu örnekte, doğrulanan dosyanın imza dosya adından ".asc" kaldırılarak aynı dizinde bulunabileceği varsayılmaktadır. Durum böyle olmadığında, hedef

dosya adını GPG'ye son argüman olarak aktarmanız yeterlidir. Dosya üzerinde oynanmışsa, sonuçlar Örnek 2.4'teki gibi görünecektir.

#### Örnek 2.3. PGP anahtar parmak izlerini doğrulama (Başarılı)

```
flog> gpg --verify nmap-4.76.tar.bz2.asc
gpg: Signature made Fri 12 Sep 2008 02:03:59 AM PDT using DSA key ID 6B9355D0
gpg: Good signature from "Nmap Project Signing Key (http://www.insecure.org/)"
```

#### Örnek 2.4. Sahte bir dosyayı tespit etme

```
flog> gpg --verify nmap-4.76.tar.bz2.asc nmap-4.76-hacked.tar.bz2
gpg: Signature made Fri 12 Sep 2008 02:03:59 AM PDT using DSA key ID 6B9355D0
gpg: BAD signature from "Nmap Project Signing Key (http://www.insecure.org/)"
```

PGP imzaları önerilen doğrulama tekniği olsa da, SHA2, SHA1 ve MD5 (diğerlerinin yanı sıra) karmaları daha sıradan doğrulama için kullanılabilir hale getirilmiştir. İnternet trafiğinizi gerçek zamanlı olarak manipüle edebilen (ve son derece yetenekli olan) veya Nmap.Org'u ele geçirip hem dağıtım dosyasını hem de özet dosyasını değiştiren bir saldırgan bu testi geçebilir. Bununla birlikte, Nmap'i üçüncü bir taraftan edinirseniz veya yanlışlıkla bozulmuş olabileceğini düşünüyorsanız, yetkili Nmap.Org karmalarını kontrol etmek yararlı olabilir. Her Nmap paketi indirme dosyası için, sigs dizininde adına .digest.txt eklenmiş karşılık gelen bir dosya vardır (örneğin nmap-4.76.tar.bz2.digest.txt). Örnek 2.5'te bir örnek gösterilmiştir. Bu, ayrılmış imza dosyasıdır. Özet dosyasındaki karmalar, Örnek 2.6, "Nmap karmalarını doğrulama" bölümünde gösterildiği gibi gpg, sha1sum veya md5sum gibi yaygın araçlar kullanılarak doğrulanabilir.

#### Örnek 2.5. Tipik bir Nmap sürüm özeti dosyası

```
flog> cat sigs/nmap-4.76.tgz.digest.txt
nmap-4.76.tgz: MD5 = 54 B5 C9 E3 F4 4C 1A DD E1 7D F6 81 70 EB 7C FE
nmap-4.76.tgz: SHA1 = 4374 CF9C A882 2C28 5DE9 D00E 8F67 06D0 BCFA A403
nmap-4.76.tgz: RMD160 = AE7B 80EF 4CE6 DBAA 6E65 76F9 CA38 4A22 3B89 BD3A
nmap-4.76.tgz: SHA224 = 524D479E 717D98D0 2FB0A42B 9A4E6E52 4027C9B6 1D843F95
                           D419F87F
nmap-4.76.tgz: SHA256 = 0E960E05 53EB7647 0C8517A0 038092A3 969DB65C BE23C03F
                           D6DAEF1A CDCC9658
nmap-4.76.tgz: SHA384 = D52917FD 9EE6EE62 F5F456BF E245675D B6EEEBC5 0A287B27
                           3CAA4F50 B171DC23 FE7808A8 C5E3A49A 4A78ACBE A5AEED33
nmap-4.76.tgz: SHA512 = 826CD89F 7930A765 C9FE9B41 1DAFD113 2C883857 2A3A9503
                           E4C1E690 20A37FC8 37564DC3 45FF0C97 EF45ABE6 6CEA49FF
                           E262B403 A52F4ECE C23333A0 48DEDA66
```

### Örnek 2.6. Nmap karmalarını doğrulama

```
flog> gpg --print-md sha256 nmap-4.76.tgz
nmap-4.76.tgz: 0E960E05 53EB7647 0C8517A0 038092A3 969DB65C BE23C03F D6DAEF1A
                           CDCC9658
flog> shasum nmap-4.76.tgz
4374cf9ca8822c285de9d00e8f6706d0bcfaa403  nmap-4.76.tgz
flog> md5sum nmap-4.76.tgz
54b5c9e3f44cladde17df68170eb7cfe  nmap-4.76.tgz
```

Nmap.Org sürümleri bu bölümde açıklandığı gibi imzalanırken, belirli Nmap eklientileri, arayüzleri ve platforma özgü ikili dosyalar diğer taraflarca geliştirilir ve dağıtıılır. İndirmelerinin gerçekliğini belirlemek için farklı mekanizmaları vardır.

### Obtaining Nmap from the Subversion (SVN) Repository (Nmap'i Subversion (SVN) Deposundan Edinme)

Düzenli kararlı ve geliştirme sürümlerine ek olarak, en son Nmap kaynak kodu Subversion (SVN) revizyon kontrol sistemi kullanılarak her zaman kullanılabilir. Bu, yeni özellikler ve sürüm/OS algılama veritabanı güncellemelerini geliştirildikleri anda sunar. Dezavantajı, SVN kafa revizyonlarının her zaman resmi sürümler kadar kararlı olmamasıdır. Bu yüzden SVN en çok Nmap geliştiricileri ve henüz resmi olarak yayınlanmamış bir düzeltmeye ihtiyaç duyan kullanıcılar için kullanışlıdır.

SVN yazma erişimi kesinlikle üst düzey Nmap geliştiricileri ile sınırlıdır, ancak herkesin depoya okuma erişimi vardır. `svn co https://svn.nmap.org/nmap` komutunu kullanarak en son kodu kontrol edin. Daha sonra çalışma dizinizde `svn up` yazarak kaynak kodunuzu güncelleyebilirsiniz.

Çoğu kullanıcı SVN'de sadece /nmap dizinini takip etse de, ilginç bir dizin daha vardır: /nmap-exp. Bu dizin, Nmap geliştiricilerinin Nmap'in dengesini bozmadan yeni şeyler denemek istediklerinde oluşturdukları deneysel Nmap dallarını içerir. Geliştiriciler deneysel bir dalın daha geniş ölçekli testler için hazır olduğunu düşündüklerinde, genellikle konumu nmap-dev posta listesine e-posta ile gönderirler.

Nmap kontrol edildikten sonra, típkı Nmap tarball (bu bölümün ilerleyen kísmalarında açıklanmaktadır) ile yaptığınız gibi kaynak koddan derleyebilirsiniz.

Nmap'te herhangi bir değişiklik yapıldığında e-posta ile gerçek zamanlı (veya özetlenmiş) bildirim ve farklılıklar istiyorsanız,  
<https://nmap.org/mailman/listinfo svn> adresindeki nmap-svn posta listesine kaydolun.

## **Linux/Unix Compilation and Installation from Source Code** **(Linux/Unix Kaynak Kodundan Derleme ve Kurulum)**

Çoğu platform için ikili paketler (daha sonraki bölgümlerde ele alınacaktır) mevcut olsa da, kaynak koddan derleme ve kurulum Nmap'i kurmanın geleneksel ve en güçlü yoludur. Bu, en son sürümün mevcut olmasını sağlar ve Nmap'in sisteminizin kütüphane kullanılabilirliğine ve dizin yapısına uyum sağlamasına olanak tanır.

Örneğin, Nmap mevcut olduğunda sürüm tespiti için OpenSSL kriptografi kütüphanelerini kullanır, ancak çoğu ikili paket bu işlevi içermez. Öte yandan, ikili paketlerin kurulumu genellikle daha hızlı ve kolaydır ve sistemdeki tüm paketlenmiş yazılımların tutarlı bir şekilde yönetilmesine (kurulum, kaldırma, yükselme vb.) olanak tanır.

Kaynak kurulumu genellikle zahmetsiz bir süreçtir; derleme sistemi mümkün olduğunda çok şeyi otomatik olarak algılayacak şekilde tasarlanmıştır. Varsayılan bir kurulum için gereken adımlar aşağıda verilmiştir:

1. Nmap'in en son sürümünü .tar.bz2 (bzip2 sıkıştırma) veya .tgz (gzip sıkıştırma) formatında <https://nmap.org/download.html> adresinden indirin.
2. İndirilen tarball'un sıkışmasını aşağıdaki gibi bir komutla açın:

**bzip2 -cd nmap- <VERSION>.tar.bz2 | tar xvf -**

GNU tar ile, daha basit bir komut olan tar xvzf nmap-<VERSION>.tar.bz2 işinizi görür. Eğer .tgz sürümünü indirdiyseniz, açma komutunda bzip2 yerine gzip yazın.

3. Yeni oluşturulan dizine geçin: cd nmap-<VERSION>
  4. Yapı sistemini yapılandırın: ./configure

Yapilandırma başarılı olursa, Örnek 2.7'de gösterildiği gibi, başarılı yapılandırma için sizi tebrik etmek ve dikkatli olmanız için sizi uyarmak üzere bir ASCII sanat ejderhası görünür.

#### Örnek 2.7. Başarılı yapılandırma ekranı

5. Nmap'i (ve gereksinimleri karşılanıyorsa Zenmap GUI'sini) derleyin: make
  6. Sistem genelinde kurulum için ayrıcalıklı kullanıcı olun: su root
  7. Nmap'i, destek dosyalarını, dokümanları vb. yükleyin: make install

Yukarıda görebileceğiniz gibi, basit bir kaynak derleme ve yükleme işlemi `./configure;make;make install` komutunu root olarak çalıştmaktan biraz daha fazlasını içerir. Bununla birlikte, Nmap'in oluşturulma şeklini etkileyen bir dizi yapılandırma seçeneği mevcuttur.

## Configure Directives (Yönergeleri Yapılandırma)

Unix derleme seçeneklerinin çoğu, yukarıdaki dört numaralı adımda kullanıldığı gibi configue betiği tarafından kontrol edilir. Nmap'in oluşturulma şeklini etkileyen

düzinelerce komut satırı parametresi ve çevresel değişken vardır. Kısa açıklamalar içeren büyük bir liste için ./configure --help komutunu çalıştırın. Bunlar Windows üzerinde Nmap oluşturmak için geçerli değildir. Burada Nmap'e özgü ya da özellikle önemli olan seçenekler yer almaktadır:

`--prefix= <directoryname>` Çoğu yazılımın configure betiklerinde standart olarak bulunan bu seçenek, Nmap ve bileşenlerinin nereye kurulacağını belirler. Varsayılan olarak, önek /usr/local'dır, yani nmap /usr/local/bin içine kurulur, man sayfası (nmap.1) /usr/local/man/man1 içine kurulur ve veri dosyaları (nmap-os-db, nmap-services, nmap-service-probes, vb.) /usr/local/share/nmap altına kurulur. Yalnızca belirli bileşenlerin yolunu değiştirmek istiyorsanız --bindir, --datadir ve/veya --mandir seçeneklerini kullanın. Örnek bir --prefix kullanımı, Nmap'i benim hesabımı ayrıcalıksız bir kullanıcı olarak yüklemek olabilir. ./configure --prefix=</home/fyodor> komutunu çalıştırırmış. Nmap kurulum aşamasında /home/fyodor/man/man1 gibi alt dizinler oluşturur, eğer bunlar zaten mevcut değilse.

`--without-zentool` Bu seçenek Zenmap grafik önyüzünün yüklenmesini engeller. Normalde derleme sistemi Python betik dili gibi gereksinimler için sisteminizi kontrol eder ve ardından hepsi mevcutsa Zenmap'i yükler.

`--with-openssl= <directoryname>` Sürüm tespit sistemi ve Nmap Scripting Engine, ücretsiz OpenSSL kütüphanelerini kullanarak SSL şifreli hizmetleri araştırabilir. Normalde Nmap derleme sistemi sisteminizde bu kütüphaneleri arar ve bulunurlarsa bu özelliği dahil eder. Derleyicinizin varsayılan olarak aramadığı bir konumdaysa, ancak yine de kullanılmasını istiyorsanız, --with-openssl=<directoryname> belirtin. Nmap daha sonra OpenSSL kütüphanelerinin kendileri için <directoryname>/libs ve gerekli başlık dosyaları için <directoryname>/include konumlarına bakar. SSL'i tamamen devre dışı bırakmak için --without-openssl belirtin.

Bazı dağıtımlar, programların çalıştırılmasına izin veren kullanıcı OpenSSL kütüphaneleri ile birlikte gönderilir, ancak bunları derlemek için gereken geliştirici dosyaları yoktur. Bu geliştirici paketleri olmadan, Nmap OpenSSL destegine sahip olmayacağındır. Debian tabanlı sistemlerde libssl-dev paketini yükleyin. Red Hat tabanlı sistemlerde openssl-devel paketini yükleyin.

`--with-libpcap= <directoryname>` Nmap ham IP paketlerini yakalamak için Libpcap kütüphanesini kullanır. Nmap normalde sisteminizde Libpcap'in mevcut bir

kopyasını arar ve sürüm numarası ve platform uygunsanız bunu kullanır. Aksi takdirde Nmap, Libpcap'in kendi yeni kopyasını içerir (Nmap kaynak dizinindeki libpcap/NMAP\_MODIFICATIONS içinde açıklanan bazı yerel değişikliklerle). Nmap'i kendi Libpcap'ınızla bağlantı kurmaya zorlamak isterseniz, yapılandırmaya --with-libpcap=<directoryname> seçeneğini aktarın. Nmap daha sonra Libpcap kütüphanesinin <directoryname>/lib/libpcap.a içinde olmasını ve include dosyalarının <directoryname>/include içinde olmasını bekler. Eğer --with-libpcap=included belirtirseniz Nmap her zaman Libpcap'in tarball'da bulunan sürümünü kullanacaktır.

--with-libpcre= <directoryname> PCRE, <http://www.pcre.org> adresinde bulunan Perl uyumlu bir düzenli ifade kütüphanesidir. Nmap normalde sisteminizde bir kopyasını arar ve bu başarısız olursa kendi kopyasına geri döner. Eğer PCRE kütüphaneniz derleyicinizin standart arama yolunda değilse, Nmap muhtemelen onu bulamayacaktır. Bu durumda, yapılandırma için --with-libpcre=<directoryname> seçeneğini belirterek Nmap'e nerede bulunabileceğini söyleyebilirsiniz. Nmap daha sonra kütüphane dosyalarının <directoryname>/lib içinde ve include dosyalarının da <directoryname>/include içinde olmasını bekler. Bazı durumlarda, Nmap ile birlikte gelen PCRE kütüphanelerini sisteminizde bulunanlara tercih etmek isteyebilirsiniz. Bu durumda, --with-libpcre=included seçeneğini belirtin.

--with-localdirs Bu basit seçenek Nmap'e önemli kütüphane ve başlık dosyaları için /usr/local/lib ve /usr/local/include dosyalarına bakmasını söyler. Bu hiçbir zaman gerekliliğinden olmamalıdır, ancak bazı insanlar bu tür kütüphaneleri derleyicilerini onları bulacak şekilde yapılandırmadan /usr/local içine koyarlar. Eğer o kişilerden biriyeniz, bu seçeneği kullanın.

## Environment Variables (Ortam Değişkenleri)

configure komut dosyası çeşitli ortam değişkenlerine duyarlıdır. Bunlar bu değişkenlerden bazıları ve etkileridir.

CFLAGS , CXXFLAGS , LDFLAGS Sırasıyla C derleyicisine, C++ derleyicisine ve bağlayıcıya iletilecek ekstra seçenekler. Nmap'in bazı bölümleri C, bazıları ise C++ dilinde yazıldığından, bunlardan birini kullanacaksanız hem CFLAGS hem de CXXFLAGS kullanmak en iyisidir.

LINGUAS Varsayılan olarak, make install, İngilizce olana ek olarak Nmap man sayfasının mevcut tüm çevirilerini yükleyecektir. LINGUAS ortam değişkeni hangi çevirilerin yükleneceğini kontrol edebilir. Değeri, ISO dil kodlarının boşluk

bırakılarak ayrılmış bir listesi olmalıdır. Örneğin, yalnızca Fransızca ve Almanca çevirileri yüklemek için LINGUAS="fr de" make install komutunu çalıştırabilirsiniz. Tüm çevirilerin yüklenmesini devre dışı bırakmak için configu're'u --disable-nls seçeneği ile çalıştırın veya LINGUAS'ı boş dizeye ayarlayın.

### If You Encounter Compilation Problems (Derleme Sorunlarıyla Karşılaşırsanız)

İdeal bir dünyada, yazılım her sistemde her zaman mükemmel (ve hızlı) bir şekilde derlenirdi. Ne yazık ki, toplum henüz bu nirvana durumuna ulaşamamıştır. Nmap'i taşınabilir hale getirmek için tüm çabalarımıza rağmen, derleme sorunları zaman zaman ortaya çıkmaktadır. İşte kaynak dağıtım derlemesinin başarısız olması durumunda bazı öneriler.

En son Nmap'e yükseltme

Nmap'in en son sürümünü kullandığınızdan emin olmak için <https://nmap.org/download.html> adresini kontrol edin. Sorun çoktan çözülmüş olabilir.

Hata mesajını dikkatlice okuyun

Çıktı ekranında yukarı kaydırın ve komutlar başarısız olduğunda verilen hata mesajlarını inceleyin. Genellikle ilk hata mesajını bulmak en iyisidir, çünkü bu genellikle diğer hataların art arda gelmesine neden olur. Düşük disk alanı veya bozuk bir derleyici gibi bir sistem sorununa işaret edebileceğinden hata mesajını dikkatlice okuyun. Programlama becerisine sahip kullanıcılar daha geniş bir yelpazede sorunları kendileri çözebilirler. Sorunu çözmek için kod değişiklikleri yaparsanız, lütfen bir yama (diff -uw <oldfile> <newfile> ile oluşturulmuş) ve sorunuz ve platformunuzla ilgili tüm ayrıntıları "Hatalar" bölümünde açıklandığı gibi nmap-dev'e gönderin. Değişikliğin temel Nmap dağıtımına entegre edilmesi, diğer birçok kullanıcının yararlanmasını sağlar ve her yeni Nmap sürümünde değişiklik yapmak zorunda kalmanızı önler.

Ask nmap-dev

Araştırmanızın hiçbirini bir çözüme ulaşmazsa, "Hatalar" adlı bölümde açıklandığı gibi Nmap geliştirme (nmap-dev) posta listesine bir rapor göndermeyi deneyin.

İkili paketleri düşünün

Nmap'in ikili paketleri çoğu platformda mevcuttur ve genellikle kurulumu kolaydır. Dezavantajları, güncel olmayabilmeleri ve kendi kendine derleme esnekliğinin bir

kısmini kaybetmenizdir. Bu bölümün ilerleyen kısımlarında birçok platformda ikili paketlerin nasıl bulunacağı anlatılmaktadır ve daha fazlası Internet'te arama yapılarak bulunabilir. Açıkçası, ikili paketleri yalnızca saygın kaynaklardan yüklemelisiniz.

## **Linux Distributions (Linux Dağıtımları)**

Linux, Nmap çalıştırılmak için en popüler platformdur. Bir kullanıcı anketinde, kullanıcıların %86'sı Linux'un Nmap'i çalıştırdıkları platformlardan en az biri olduğunu söylemiştir. Nmap'in 1997'deki ilk sürümü yalnızca Linux üzerinde çalışıyordu.

Linux kullanıcıları kaynak kodu yüklemesi ya da dağıtımları veya Insecure.Org tarafından sağlanan ikili paketleri kullanmak arasında seçim yapabilirler. İkili paketlerin kurulumu genellikle daha hızlı ve kolaydır ve genellikle dağıtımın standart dizin yollarını ve benzerlerini kullanmak için biraz özelleştirilir. Bu paketler ayrıca sistemdeki yazılımın yükseltilmesi, kaldırılması veya incelenmesi açısından tutarlı bir yönetim sağlar. Dezavantaj ise dağıtımlar tarafından oluşturulan paketlerin Nmap.Org kaynak sürümlerinin gerisinde kalmasıdır. Çoğu Linux dağıtımı Nmap paketlerini nispeten güncel tutar, ancak birkaçı çok güncel değildir. Kaynak yüklemeyi seçmek, Nmap'in sisteminiz için nasıl oluşturulduğunu ve optimize edildiğini belirlemeye daha fazla esneklik sağlar. Nmap'i kaynaktan derlemek için "Linux/Unix Kaynak Koddan Derleme ve Kurulum" bölümüne bakın. Burada en yaygın dağıtımlar için basit paket talimatları bulunmaktadır.

### **RPM-based Distributions (Red Hat, Mandrake, SUSE, Fedora) (RPM tabanlı dağıtımlar (Red Hat, Mandrake, SUSE, Fedora))**

Nmap'in her sürümü için RPM paketleri oluşturuyorum ve bunları <https://nmap.org/download.html> adresindeki Nmap indirme sayfasına gönderiyorum. İki paket oluşturuyorum: nmap paketi sadece komut satırı çalıştırılabilir dosyasını ve veri dosyalarını içerirken, zenmap paketi isteğe bağlı Zenmap grafik önyüzünü içerir (bkz. Bölüm 12, Zenmap GUI Kullanıcı Kılavuzu). Zenmap paketi, önce nmap paketinin yüklenmesini gerektirir.

RPM ile kurulum oldukça kolaydır, hatta uygun URL'ler verildiğinde paketi sizin için indirir. Aşağıdaki örnek, ön uç dahil olmak üzere Nmap 4.68'i indirir ve yükler. Elbette bunun yerine yukarıdaki indirme sitesindeki en son sürümü kullanmalısınız.

Mevcut RPM yüklü sürümler yükseltilir. Örnek 2.8 bu yükleme işlemini göstermektedir.

#### Örnek 2.8. Nmap'i ikili RPM'lerden yükleme

```
# rpm -vhU https://nmap.org/dist/nmap-4.68-1.i386.rpm
Retrieving https://nmap.org/dist/nmap-4.68-1.i386.rpm
Preparing... ################################################ [100%]
1:nmap ################################################ [100%]
# rpm -vhU https://nmap.org/dist/zenmap-4.68-1.noarch.rpm
Retrieving https://nmap.org/dist/zenmap-4.68-1.noarch.rpm
Preparing... ################################################ [100%]
1:zenmap ################################################ [100%]
```

Yukarıdaki dosya adlarından da anlaşılacağı gibi, bu ikili RPM'ler normal PC'ler (x86 mimarisi) için oluşturulmuştur. Ayrıca 64-bit Linux kullanıcıları için x86\_64 ikililerini de dağıtıyorum. Bu ikili dosyalar SPARC, Alpha veya PowerPC gibi diğer platformlardaki nispeten az sayıdaki Linux kullanıcısı için çalışmayacaktır. Ayrıca kütüphane sürümleriniz RPM'lerin başlangıçta oluşturulduğu sürümlerden yeterince farklısa yüklemeyi reddedebilirler. Bu durumlarda bir seçenek, Linux satıcınız tarafından özel dağıtımınız için hazırlanmış ikili RPM'leri bulmak olacaktır. Orijinal kurulum CD'leri veya DVD'leri başlamak için iyi bir yerdır. Ne yazık ki, bunlar güncel ya da mevcut olmayı bilir. Diğer bir seçenek de Nmap'i daha önce açıklandığı gibi kaynak koddan kurmaktır, ancak ikili paket bakım tutarlılığı avantajlarını kaybedersiniz. Üçüncü bir seçenek ise yukarıdaki indirme sayfasından dağıtılan kaynak RPM'lerden kendi ikili RPM'lerinizi oluşturmak ve kurmaktadır. Örnek 2.9 bu tekniği Nmap 4.68 ile göstermektedir.

#### Örnek 2.9. Kaynak RPM'lerden Nmap oluşturma ve yükleme

```
> rpmbuild --rebuild https://nmap.org/dist/nmap-4.68-1.src.rpm
[ hundreds of lines cut ]
Wrote: /home/fyodor/rpmdir/RPMS/i386/nmap-4.68-1.i386.rpm
[ cut ]
> su
Password:
# rpm -vhU /home/fyodor/rpmdir/RPMS/i386/nmap-4.68-1.i386.rpm
Preparing... ################################################ [100%]
1:nmap ################################################ [100%]
#
```

Zenmap RPM mimariden bağımsız ("noarch") olduğu için Zenmap'i bu şekilde yeniden oluşturmak gerekli değildir. Bu nedenle Zenmap kaynak RPM'leri yoktur.

RPM paketlerini kaldırmak rpm -e nmap zenmap kadar kolaydır.

### **Updating Red Hat, Fedora, Mandrake, and Yellow Dog Linux with Yum (Red Hat, Fedora, Mandrake ve Yellow Dog Linux'un Yum ile Güncellenmesi)**

Red Hat, Fedora, Mandrake ve Yellow Dog Linux dağıtımları, merkezi RPM depolarından yazılım yükleme ve güncellemelerini yöneten Yum adlı bir uygulamaya sahiptir. Bu, yazılım kurulumunu ve güncellemelerini önemsiz hale getirir. Normalde dağıtıma özel Yum depoları kullanıldığından, yazılımın kendi dağıtımınızla uyumluluğu açısından zaten test edildiğini bilirsiniz. Çoğu dağıtım kendi Yum depolarında Nmap bulundurur, ancak bunu her zaman güncel tutmazlar. Bu özellikle (çoğu insan gibi) dağıtımınızın en son sürümüne her zaman hızlı bir şekilde güncelleme yapmıyorsanız sorun yaratır. Eğer iki yıllık bir Linux sürümü kullanıyorsanız, Yum size genellikle Nmap'in iki yıllık bir sürümünü verecektir. Dağıtımların en son sürümlerinin bile yeni bir Nmap sürümüne güncellenmesi genellikle aylar alır. Bu nedenle, bu sistemlerde Nmap'in en son sürümü için, önceki bölümde açıklandığı gibi dağıtığımız RPM'leri deneyin. Ancak RPM'lerimiz sisteminizle uyumlu değilse veya çok aceleniz varsa, Nmap'i Yum'dan yüklemek genellikle yum install nmap'i çalıştırarak kadar basittir (GUI'yi de istiyorsanız yum install nmap zenmap'i çalıştırın, ancak bazı dağıtımlar henüz Zenmap'i paketlememiştir). Yum, internetteki bir depo ile iletişime geçerek mimariniz için uygun paketi bulur ve ardından gerekli bağımlılıklarla birlikte yükler. Bu, Örnek 2.10'da gösterilmiştir (kısalık için düzenlenmiştir). Daha sonra Nmap ve depodaki diğer paketlere yönelik mevcut güncellemeleri yüklemek için yum update işlemini gerçekleştirebilirsiniz.

Örnek 2.10. Nmap'i bir sistem Yum deposundan yükleme

```
flog~# yum install nmap
Setting up Install Process
Parsing package install arguments
Resolving Dependencies
--> Running transaction check
---> Package nmap.x86_64 2:4.52-1.fc8 set to be updated
--> Finished Dependency Resolution
Dependencies Resolved
=====
Package           Arch    Version        Repository      Size
=====
Installing:
nmap            x86_64  2:4.52-1.fc8   updates       1.0 M

Transaction Summary
=====
Install      1 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total download size: 1.0 M
Is this ok [y/N]: y
Downloading Packages:
(1/1): nmap-4.52-1.fc8.x8 100% |=====| 1.0 MB  00:02
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: nmap                      ##### [1/1]

Installed: nmap.x86_64 2:4.52-1.fc8
Complete!
```

## **Debian Linux and Derivatives such as Ubuntu (Debian Linux ve Ubuntu gibi türevleri)**

LaMont Jones, Nmap deb paketlerini makul ölçüde güncel tutmak da dahil olmak üzere sürdürmektedir. Uygun yükseltme/kurma komutu apt-get install nmap'tır. Bu Ubuntu gibi Debian türevleri için de geçerlidir. En son Debian "kararlı" Nmap paketi hakkında bilgi <http://packages.debian.org/stable/nmap> adresinde mevcuttur ve geliştirme ("kararsız") Nmap ve Zenmap paketleri

<http://packages.debian.org/unstable/nmap> ve

<http://packages.debian.org/unstable/zenmap> adreslerinde mevcuttur.

Bazen Debian'ın Nmap sürümleri güncel Nmap sürümünden bir yıl veya daha fazla geride kalmaktadır. En son sürümü edinmek için bir seçenek, "Linux/Unix Kaynak Kodundan Derleme ve Kurulum" başlıklı bölümde açıklandığı gibi kaynak kodundan derlemektir. Diğer bir seçenek ise RPM formatındaki ikili dosyaları Nmap indirme

sayfasından indirmek, alien komutunu kullanarak deb paketlerine dönüştürmek ve ardından aşağıdaki listede açıklandığı gibi dpkg kullanarak kurmaktadır:

Debian/Ubuntu'ya kurulum için Nmap RPM dosyalarını Debian/Ubuntu deb formatına dönüştürme adımları

1. Eğer alien komutuna sahip değilseniz, sudo apt-get install alien gibi bir komutla yükleyin
2. Platformunuz (x86 veya x86-64) için Nmap RPM'lerini <https://nmap.org/download.html> adresinden indirin. Bu açıklama nmap-5.21-1.x86\_64.rpm kullanacaktır
3. "Nmap İndirmelerinin Bütünlüğünü Doğrulama" bölümünde açıklandığı gibi indirme bütünlüğünü doğrulayın.
4. sudo alien nmap-5.21-1.x86\_64.rpm gibi bir komutla bir Debian paketi oluşturun
5. Debian paketini sudo dpkg --install nmap\_5.21-2\_amd64.deb gibi bir komutla yükleyin
6. Adım 2-5 Zenmap, Ncat ve Nping gibi diğer Nmap RPM'leri için tekrarlanabilir.

### **Other Linux Distributions (Diğer Linux Dağıtımları)**

Burada listelemek için çok fazla Linux dağıtıımı mevcuttur, ancak belirsiz olanların çoğu bile paket ağaçlarında Nmap içerir. Eğer içermiyorlarsa, "Linux/Unix Derleme ve Kaynak Koddan Kurulum" bölümünde anlatıldığı gibi kaynak koddan derleyebilirsiniz.

## **Windows**

Nmap bir zamanlar sadece Unix'e özel bir araçken, 2000 yılında bir Windows sürümü piyasaya sürüldü ve o zamandan beri en popüler ikinci Nmap platformu haline geldi (Linux'un ardından). Bu popülerlik ve birçok Windows kullanıcısının bir derleyiciye sahip olmaması nedeniyle, her büyük Nmap sürümü için ikili yürütülebilir dosyalar dağıtılmaktadır. Nmap'i Windows 7 ve daha yeni sürümlerin yanı sıra Windows Server 2008 ve daha yeni sürümlerde de destekliyoruz. Ayrıca Nmap'i daha önceki Windows sürümlerinde çalıştırmak zorunda olan kullanıcılar için bir kılavuz da bulunduruyoruz. Önemli ölçüde gelişmiş olsa da, Windows portu Unix'teki kadar verimli değildir. İşte bilinen sınırlamalar:

- Nmap, ham paket taramaları için yalnızca ethernet arayüzlerini (çoğu 802.11 kablosuz kart ve birçok VPN istemcisi dahil) destekler. -sT -Pn seçeneklerini kullanmadığınız sürece, RAS bağlantıları (PPP çevirmeli bağlantıları gibi) ve belirli VPN istemcileri desteklenmez. Microsoft, Windows XP SP2'de ham TCP/IP soket desteğini kaldırıldığından bu destek kesildi. Şimdi Nmap bunun yerine daha düşük seviyeli ethernet çerçeveleri göndermelidir.
- Npcap olmadan Nmap kullanırken, genellikle kendi makinenizi kendisinden tarayamazsınız (127.0.0.1 gibi bir geri döngü IP'si veya kayıtlı IP adreslerinden herhangi birini kullanarak). Bu, Windows kendi kendine yükleyicisine dahil olan Npcap'te üzerinde çalıştığımız bir Windows sınırlamasıdır. Npcap kurulumu olmayan kullanıcılar, ham paketler göndermek yerine üst düzey soket API'sini kullandığından, pingleme yapmadan (-sT -Pn) bir TCP bağlantı taraması kullanabilirler.

Windows'taki tarama hızları genellikle Unix'tekilerle karşılaştırılabilir, ancak ikincisi genellikle hafif bir performans avantajına sahiptir. Bunun bir istisnası, Windows ağ API'sindeki eksiklikler nedeniyle Windows'ta genellikle çok daha yavaş olan bağlantı taramasıdır (-sT). Bu utanç verici bir durumdur, çünkü bu tarama tüm ağ türleri üzerinde çalışan tek TCP taramasıdır (ham paket taramaları gibi sadece ethernet değil). Nmap ile birlikte gelen nmap\_performance.reg dosyasındaki Kayıt Defteri değişiklikleri uygulanarak Connect tarama performansı önemli ölçüde artırılabilir. Varsayılan olarak bu değişiklikler Nmap çalıştırılabilir yükleyicisi tarafından sizin için uygulanır. Bu kayıt dosyası Windows ikili zip dosyasının nmap-<version> dizininde ve kaynak tarball'un nmap-<version>/mswin32 dizinindedir (burada <version> belirli sürümün sürüm numarasıdır). Bu değişiklikler, kullanıcı uygulamaları (Nmap gibi) için ayrılan geçici bağlantı noktalarının sayısını artırır ve kapatılan bir bağlantının yeniden kullanılabilmesi için geçen süreyi azaltır. Çoğu kişi bu değişiklikleri çalıştırılabilir Nmap yükleyicisinde uygulamak için kutuyu işaretler, ancak bunları nmap\_performance.reg dosyasına çift tıklayarak veya regedt32 nmap\_performance.reg komutunu çalıştırarak da uygulayabilirsiniz. Değişiklikleri elle yapmak için, bu üç Kayıt Defteri DWORD değerini HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters'a ekleyin:

MaxUserPort 65534 (0x0000ffff) gibi büyük bir değer ayarlayın. MS KB 196271'e bakın.

TCPTimedWaitDelay Minimum değeri (0x0000001e) ayarlayın. MS KB 149532'ye bakın.

StrictTimeWaitSeqCheck TCPTimedWaitDelay'in kontrol edilmesi için 1 olarak ayarlayın.

not: eEye'dan Ryan Permeh, Andy Lutomirski ve Jens Vogt'a Nmap Windows portu üzerindeki sıkı çalışmaları için teşekkür etmek istiyorum. Uzun yıllar boyunca Nmap sadece Unix'e özel bir aracı ve onların çabaları olmasaydı muhtemelen hala öyle olacaktı.

Windows kullanıcılarının Nmap'i yüklemek için üç seçenek vardır ve bunların tümü <https://nmap.org/download.html> adresindeki indirme sayfasından edinilebilir.

### **Windows Self-installer (Windows Kendi Kendine Yükleyici)**

Her Nmap sürümü nmap-<version>-setup.exe (burada <version> belirli sürümün sürüm numarasıdır) adında bir Windows kendi kendine yükleyici içerir. Çoğu Nmap kullanıcısı çok kolay olduğu için bu seçeneği tercih eder. Kendi kendine yükleyicinin bir başka avantajı da Zenmap GUI ve diğer araçları yükleme seçeneği sunmasıdır. Basitçe yükleyici dosyasını çalıştırın ve bir yükleme yolu seçmek ve Npcap'i yüklemek için paneller boyunca size yol göstermesine izin verin. Yükleyici açık kaynak kodlu Nullsoft Scriptable Install System ile oluşturulmuştur.

Tamamlandıktan sonra, Nmap'i komut satırında veya Zenmap aracılığıyla çalışma talimatları için "Windows'ta Nmap'i Çalıştırma" adlı bölümü okuyun.

### **Command-line Zip Binaries (Komut Satırı Zip İkilileri)**

not: Çoğu kullanıcı Nmap'i daha önce tartışılan kendi kendine yükleyici ile kurmayı tercih eder.

Her kararlı Nmap sürümü, Windows komut satırı ikili dosyaları ve bir Zip arşivindeki ilgili dosyalarla birlikte gelir. Grafik arayüz dahil değildir, bu nedenle nmap.exe dosyasını bir DOS/komut penceresinden çalıştırmanız gereklidir. Ya da <https://www.cygwin.com> adresinde bulunan ücretsiz Cygwin sistemine dahil olanlar gibi üstün bir komut kabuğu indirip kurabilirsiniz. Nmap .zip ikili dosyalarını yüklemek ve çalıştmak için adım adım talimatları burada bulabilirsiniz.

### **Installing the Nmap zip binaries (Nmap zip ikili dosyalarının yüklenmesi)**

1. .zip ikili dosyalarını <https://nmap.org/download.html> adresinden indirin.

2. Zip dosyasını Nmap'in bulunmasını istediğiniz dizine çıkarın. Örnek olarak C:\Program Files verilebilir. Nmap çalıştırılabilir ve veri dosyalarını içeren nmap-<version> adında bir dizin oluşturulmalıdır.
3. Daha iyi performans için, daha önce tartışılan Nmap Kayıt Defteri değişikliklerini uygulayın.
4. Nmap ücretsiz Npcap paket yakalama kütüphanesini gerektirir. Zip dosyasında npcap-<version>.exe olarak bulunan yeni bir Npcap yükleyicisi ekledik, burada <version> Nmap sürümü yerine Npcap sürümüdür. Alternatif olarak, en son sürümü <https://npcap.com> adresinden edinebilir ve yükleyebilirsiniz.
5. Nmap'in derlenme şekli nedeniyle, çalışma zamanı bileşenlerinin Microsoft Visual C++ Yeniden Dağıtırlabilir Paketini gerektirir. Birçok sistemde bu paket zaten diğer paketlerden yüklenmiştir, ancak ihtiyacınız olması durumunda zip dosyasından VC\_redist.x86.exe dosyasını çalıştırmanız gereklidir. Bu yükleyicileri sessiz (etkileşimsiz) moda çalıştmak için /q seçeneğini选用.
6. Derlediğiniz Nmap'i çalıştırmak için talimatlar "Windows'ta Nmap Çalıştırma" bölümünde verilmiştir.

### **Compile from Source Code (Kaynak Koddan Derleme)**

Çoğu Windows kullanıcısı Nmap ikili kendine yükleyicisini kullanmayı tercih eder, ancak özellikle Nmap geliştirmeye yardımcı olmayı planlıyorsanız, kaynak koddan derleme bir seçenektedir. Derleme için ticari Visual Studio paketinin bir parçası olan Microsoft Visual C++ 2019 gereklidir. Ücretsiz Visual Studio 2019 Community dahil olmak üzere Visual Studio 2019 sürümlerinden herhangi biri çalışmalıdır.

Nmap'in Windows üzerindeki bazı bağımlılıklarının derlenmesi uygun değildir. Bu nedenle, bağımlılıkların önceden derlenmiş ikili dosyaları Subversion'da /nmap-mswin32-aux dizininde saklanır. İster bir kaynak kod sürümünden ister Subversion'dan olsun, kaynaktan oluştururken aşağıda açıklandığı gibi /nmap-mswin32-aux dizinini kontrol edin.

### **Windows'ta Nmap'i Kaynaktan Derleme**

1. Windows bağımlılıklarını Subversion'dan svn checkout <https://svn.nmap.org/nmap-mswin32-aux> komutuyla indirin. Derleme dosyaları, kontrol edilen bu dizindeki bağımlılıkları arayacak şekilde

yapılmalıdır. Bunun yerine bağımlılıkları kendiniz derlemek istiyorsanız, Visual Studio proje dosyalarını alternatif dizini gösterecek şekilde yeniden yapılandırmanız gereklidir.

2. Nmap kaynak kodunu nmap.org adresinden en son sürümü indirerek mi yoksa bir Subversion istemcisi kullanarak depomuzdan daha yeni (ancak daha az test edilmiş) kodu alarak mı elde edeceğinize karar verin. Bu talimatlar web indirme yaklaşımı içindir, ancak bunun yerine Subversion'ı kullanmak kolaydır ("Nmap'i Subversion (SVN) Deposundan Edinme" adlı bölüme bakın).
3. En son Nmap kaynak dağıtımını <https://nmap.org/download.html> adresinden indirin. Adı nmap-<version>.tar.bz2 veya nmap-<version>.tgz'dir. Bunlar sırasıyla bzip2 veya gzip kullanılarak sıkıştırılmış aynı tar dosyasıdır. Bzip2 ile sıkıştırılmış sürüm daha küçüktür.
4. İndirdiğiniz kaynak kod dosyasının sıkıştırmasını açın. Kaynak kod dizini ve nmap-mswin32-aux aynı üst dizinde olmalıdır. Ücretsiz Cygwin dağıtımının son sürümleri hem .tar.bz2 hem de .tgz formatlarını işleyebilir. Sırasıyla tar xvzf nmap-<version>.tar.bz2 veya tar xvzf nmap-<version>.tgz komutunu kullanın. Alternatif olarak, yaygın WinZip uygulaması bu dosyaların sıkıştırmasını açabilir.
5. Visual Studio'yu ve Nmap çözüm dosyasını (nmap-<version>/mswin32/nmap.sln) açın.
6. Çözüm Gezgini kenar çubukunda 'nmap' çözümüne sağ tıklayın ve "Yapılandırma Yöneticisi"ni seçin. Etkin çözüm yapılandırmasının Release olduğundan emin olun ve ardından Yapılandırma Yöneticisini kapatın.
7. F7 tuşuna basarak ya da GUI'den "Build Solution" seçeneğini seçerek Nmap'i derleyin. Nmap derlenmeye başlamalı ve tüm projelerin başarıyla derlendiğini ve sıfır hata olduğunu belirten "-- Done --" satırıyla sona ermeliidir.
8. Çalıştırılabilir ve veri dosyaları nmap-<version>/mswin32/Release/ dizininde bulunabilir. Hepsi bir arada tutulduğu sürece bunları tercih ettiğiniz bir dizine kopyalayabilirsiniz.
9. Npcap'in kurulu olduğundan emin olun. İkili kendi kendine yükleyicimizi yükleyerek veya zip paketimizden npcap-<version>.exe dosyasını çalıştırarak

edinebilirsiniz. Alternatif olarak, resmi yükleyiciyi <https://nmap.org/> adresinden edinebilirsiniz.

10. Derlediğiniz Nmap'i çalıştırmak için talimatlar bir sonraki bölümde verilmiştir.

Bir Nmap yürütülebilir Windows yükleyicisi veya Zenmap yürütülebilir dosyası oluşturmak istiyorsanız, Nmap SVN deposundaki `docs/win32-installer-zenmap-buildguide.txt` dosyasına bakın.

Birçok kişi Nmap'in Cygwin veya diğer derleyicilerle birlikte gelen `gcc/g++` ile derlenip derlenmeyeceğini sordu. Bazı kullanıcılar bu konuda başarılı olduklarını bildirdiler, ancak Cygwin altında Nmap oluşturmak için talimatlar tutmuyoruz.

### **Executing Nmap on Windows (Windows'ta Nmap Çalıştırma)**

Nmap sürümleri artık Nmap için Zenmap grafik kullanıcı arayüzüünü içeriyor. Nmap yükleyicisini kullandığınız ve Zenmap alanını işaretli bırakırsanız, masaüstünüzde ve Başlat Menüsünde yeni bir Zenmap girişi olmalıdır. Başlamak için buna tıklayın. Zenmap, Bölüm 12, Zenmap GUI Kullanıcı Kılavuzu'nda tam olarak belgelenmiştir. Birçok kullanıcı Zenmap'i severken, diğerleri Nmap'i çalıştırmak için geleneksel komut satırı yaklaşımını tercih eder. Komut satırı arayüzlerine aşina olmayan kullanıcılar için ayrıntılı talimatları burada bulabilirsiniz:

1. Oturum açtiğiniz kullanıcının bilgisayarda yönetici ayrıcalıklarına sahip olduğundan emin olun (kullanıcı `administrators` grubunun bir üyesi olmalıdır).
2. Bir komut/DOS Penceresi açın. Program menü ağacında bulunabilmesine rağmen, en basit yaklaşım "Başlat" → "Çalıştır"ı seçmek ve `cmd<enter>` yazmaktır. Masaüstündeki Cygwin simgesine tıklayarak bir Cygwin penceresi açmak (eğer yüklediyseniz) da işe yarar, ancak gerekli komutlar burada gösterilenlerden biraz farklıdır.
3. Nmap'i yüklediğiniz dizine geçin. Nmap zaten komut yolunuzdaysa bu adımı atlayabilirsiniz (Zenmap isnaller varsayılan olarak onu oraya ekler). Aksi takdirde, aşağıdaki komutları yazın.

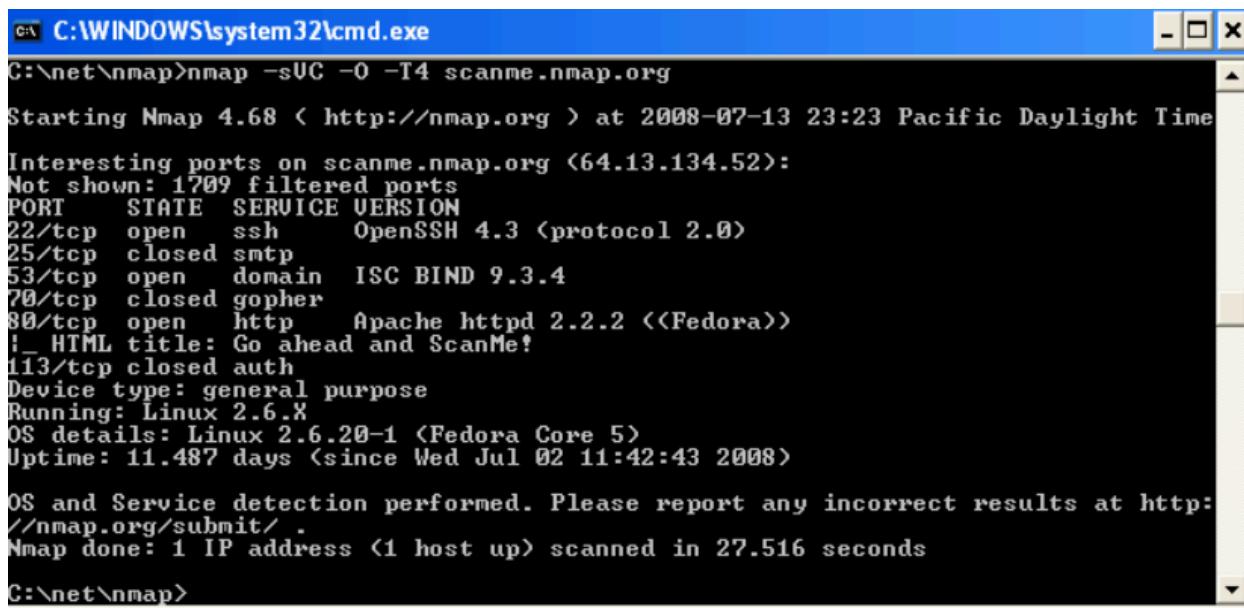
```
c:  
cd "\Program Files (x86)\Nmap"
```

Windows 7'den önceki Windows sürümlerinde, bunun yerine `\Program Files\Nmap` belirtin. Nmap'i varsayılan olmayan bir konuma yüklemeyi seçtiyseniz dizin de

farklı olacaktır.

4. nmap.exe dosyasını çalıştırın. Şekil 2.1 basit bir örneği gösteren bir ekran görüntüsüdür.

Şekil 2.1. Nmap'in Windows komut kabuğundan çalıştırılması



```
C:\WINDOWS\system32\cmd.exe
C:\net\nmap>nmap -sUC -O -T4 scanme.nmap.org
Starting Nmap 4.68 ( http://nmap.org ) at 2008-07-13 23:23 Pacific Daylight Time
Interesting ports on scanme.nmap.org (64.13.134.52):
Not shown: 1709 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 4.3 (protocol 2.0)
25/tcp    closed  smtp
53/tcp    open  domain   ISC BIND 9.3.4
70/tcp    closed  gopher
80/tcp    open  http     Apache httpd 2.2.2 <Fedora>
!_ HTML title: Go ahead and ScanMe!
113/tcp   closed auth
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.20-1 <Fedora Core 5>
Uptime: 11.487 days <since Wed Jul 02 11:42:43 2008>
OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 27.516 seconds
C:\net\nmap>
```

Nmap'i sık sık çalıştırıyorsanız, Nmap dizinini (varsayılan olarak c:\Program Files (x86)\Nmap) komut çalışma yolunuza ekleyebilirsiniz:

1. SystemPropertiesAdvanced.exe dosyasını çalıştırarak Sistem Özellikleri penceresini Gelişmiş sekmesine açın.
2. "Ortam Değişkenleri" düğmesine tıklayın.
3. Sistem değişkenleri bölümünden Yol'u seçin, ardından düzenle'ye basın.
4. Değerin sonuna bir noktalı virgül ve ardından Nmap dizinizi (örneğin c:\Program Files (x86)\Nmap) ekleyin.
5. Yeni bir komut istemi açın ve herhangi bir dizinden nmap scanme.nmap.org gibi bir komut çalıştırabilmeniz gereklidir.

## Apple Mac OS X

Nmap 2001 yılından beri Mac OS X'i desteklemektedir ve destegimiz zaman içinde daha da gelişmiştir. Mac kullanıcıları Nmap'i kendileri derleyebilirken, biz de

çalıştırılabilir bir yükleyici sunuyoruz. Nmap, Mac OS X için OpenSSL, libapr, libsvn gibi diğer projeleri oluşturmak için kullanılan Jhbuild ve gtk-mac-bundler'ı kullanır. Nmap, Mac OS X için Unix yazılımlarını paketleyen MacPorts ve Fink gibi sistemler aracılığıyla da kullanılabilir.

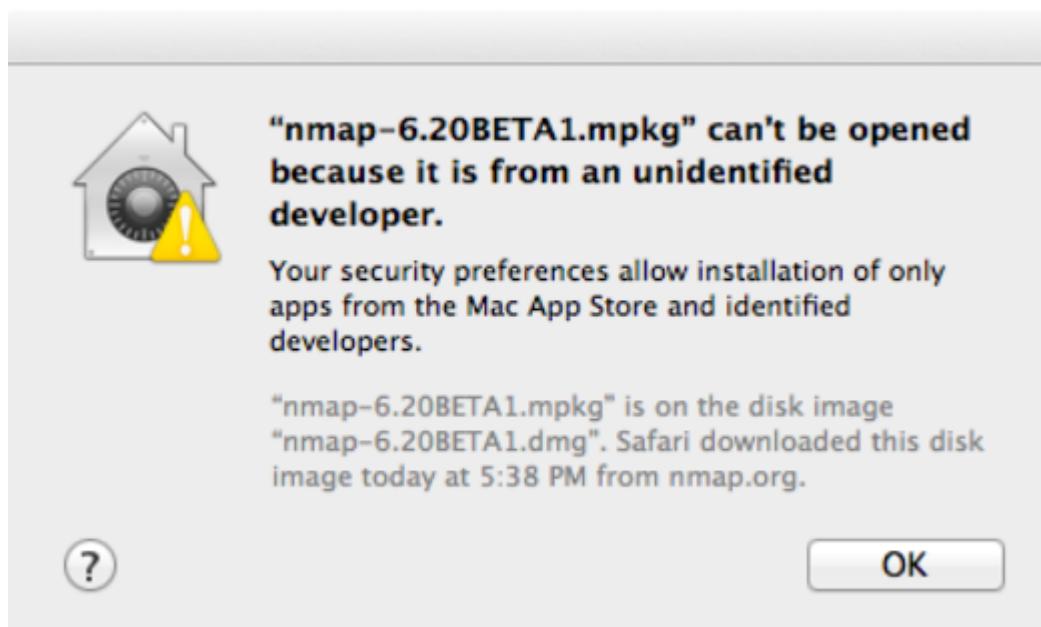
### **Executable Installer (Yürüttülebilir Yükleyici)**

Nmap ve Zenmap'i Mac OS X'e yüklemenin en kolay yolu yükleyicimizi kullanmaktadır. Nmap indirme sayfasının Mac OS X bölümü, nmap-<version>.dmg adlı bir dosya sağlar; burada <version> en son sürümün sürüm numarasıdır. .dmg dosyası "disk görüntüsü" olarak bilinir. Kurulum talimatları aşağıdaki gibidir:

1. nmap-<version>.dmg dosyasını indirin. Açımak için simgeye çift tıklayın. (Dosyayı nasıl indirdiğinize bağlı olarak, otomatik olarak açılabilir).
2. Disk görüntüsünün içeriği görüntülenecektir. Dosyalardan biri nmap-<version>.mpkg adında bir Mac meta-package dosyası olacaktır. Yükleyiciyi başlatmak için açın.

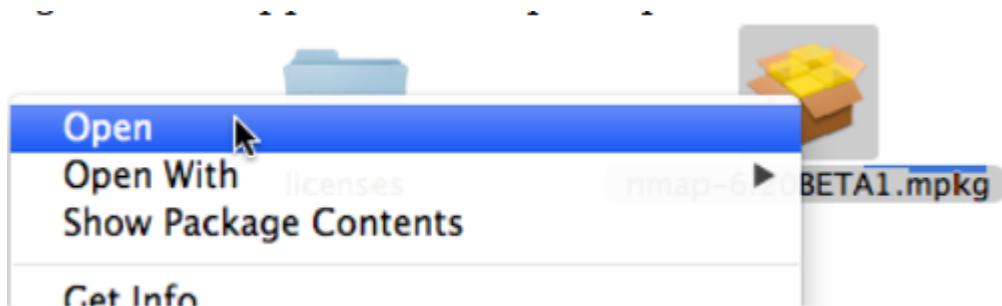
OS X 10.8 ve sonraki sürümlerde Şekil 2.2'deki gibi bir iletişim kutusu görebilirsiniz.

Şekil 2.2. Apple Gatekeeper blok ekranı



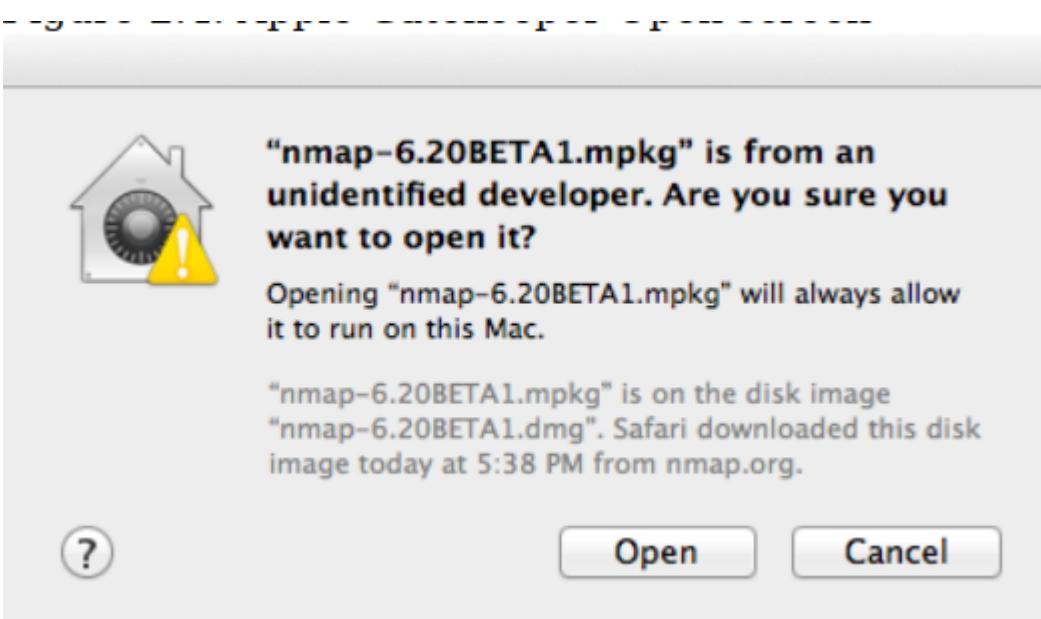
Bu durumda, Şekil 2.3'te gösterildiği gibi .mpkg'ye sağ tıklamak veya kontrol tuşuna basarak tıklamak ve "Aç"ı seçmek gerekir.

Şekil 2.3. Apple Gatekeeper Açı menüsü



İlkine benzer bir iletişim kutusu görünecektir, bu kez bir "Aç" düğmesi vardır (Şekil 2.4'te gösterilmektedir). Devam etmek için düğmeye tıklayın.

Şekil 2.4. Apple Gatekeeper Açık ekranı



3. Yükleyicideki talimatları izleyin. Nmap bir sistem dizinine yüklendiği için sizden parolanız istenecektir.
4. Yükleyici tamamlandığında, disk görüntüsünü simgesine kontrol tuşıyla tıklayıp "Çıkar"ı seçerek çıkarın. Disk görüntüsü artık çöp kutusuna yerleştirilebilir.

Nmap ve Zenmap'in kurulduktan sonra çalıştırılmasıyla ilgili yardım için "Mac OS X'te Nmap'in Çalıştırılması" başlıklı bölümdeki talimatlara bakın.

Yükleyici tarafından yüklenen programlar Intel Mac OS X 10.5 (Leopard) veya sonraki sürümlerde çalışacaktır. Daha önceki sürümleri kullananların kaynaktan derleme yapması veya üçüncü taraf bir paket kullanması gerekecektir. PowerPC (PPC) Mac sistemleri (Apple'ın 2006 yılında satışını durdurduğu) için talimatlar wikimizde mevcuttur.

### **Compile from Source Code (Kaynak Koddan Derleme)**

Mac OS X'te Nmap'i kaynaktan derlemek, uygun bir derleme ortamı oluşturulduktan sonra diğer platformlardan daha zor değildir.

### **Compile Nmap from source code (Nmap'i kaynak koddan derleme)**

Mac OS X'te Nmap'i derlemek için Apple'ın GCC ve diğer normal derleme sistemlerini içeren geliştirici araçları olan Xcode gereklidir. Xcode varsayılan olarak yüklü değildir, ancak Mac App Store'dan ücretsiz olarak indirilebilir. Xcode'u yükledikten sonra "Tercihler" i açın, "İndirilenler" sekmesini seçin ve "Komut Satırı Araçları"nın yanındaki "Yükle" ye tıklayın.

Xcode kurulumları her zaman komut satırı araçlarını içermez. Bunları Uygulamalar klasöründen Xcode'u açarak, Tercihler'i açarak, İndirme başlığı simgesini seçerek ve "Komut Satırı Araçları"nın yanındaki Yükle düğmesine tıklayarak yükleyebilirsiniz.

Xcode'u ve komut satırı araçlarını yükledikten sonra, "Linux/Unix Derleme ve Kaynak Koddan Yükleme" adlı bölümde bulunan derleme talimatlarını izleyin. Mac OS X'in bazı eski sürümlerinde ./configure komutunu ./configure CPP=/usr/bin/cpp ile değiştirmeniz gerekebileceğini unutmayın. Ayrıca, bazı yeni Mac OS X sürümlerinde, Apple tarafından sağlanan kütüphanenin libpcap sürümü çok eski olabilir. Nmap'te bulunan uyumlu sürümü kullanmak için Nmap'i ./configure --with-libpcap=included komutıyla yapılandırmınız gerekebilir veya makinenizde yüklü libpcap'i güncellemeniz gereklidir.

### **Compile Zenmap from source code (Zenmap'i kaynak koddan derleme)**

Zenmap, GTK+ ve PyGTK dahil olmak üzere Mac OS X ile birlikte gelmeyen bazı harici kütüphanelere bağlıdır. Bu kütüphanelerin kendilerine ait birçok bağımlılıkları vardır. Tüm bunları yüklemenin uygun bir yolu, Bölüm 'de açıklandığı gibi bir üçüncü taraf paketleme sistemi kullanmaktır. Bağımlılıklar yüklen dikten sonra,

Zenmap'i her zamanki gibi yüklemek için "Linux/Unix Derleme ve Kaynak Koddan Yükleme" adlı bölümdeki talimatları izleyin.

### **Third-party Packages (Üçüncü Taraf Paketleri)**

Nmap'i kurmak için bir başka seçenek de Mac OS X için Unix yazılımı paketleyen bir sistem kullanmaktadır. Burada tartışılan iki tanesi Fink ve MacPorts'tur. Paket yöneticilerinin nasıl kurulacağını öğrenmek için ilgili projelerin web sitelerine bakın.

Fink kullanarak yüklemek için fink install nmap komutunu çalıştırın. Nmap /sw/bin/nmap olarak yüklenecektir. Kaldırmak için fink remove nmap komutunu kullanın.

MacPorts kullanarak yüklemek için sudo port install nmap komutunu çalıştırın. Nmap /opt/local/bin/nmap olarak yüklenecektir. Kaldırmak için sudo port uninstall nmap komutunu çalıştırın.

Bu sistemler nmap çalıştırılabilir dosyasını global PATH'in dışına yükler. Zenmap'in bunu bulmasını sağlamak için, zenmap.conf dosyasındaki nmap\_command\_path değişkenini "nmap Çalıştırılabilir" bölümünde açıkladığı gibi /sw/bin/nmap veya /opt/local/bin/nmap olarak ayarlayın.

### **Executing Nmap on Mac OS X (Mac OS X üzerinde Nmap Çalıştırma)**

Mac OS X'teki terminal emülatörü Terminal olarak adlandırılır ve /Applications/Utilities dizininde bulunur. Açıığınızda bir terminal penceresi görünür. Burası komutlarınızı yazacağınız yerdir.

Mac OS X'te varsayılan olarak root kullanıcı devre dışıdır. Kök ayrıcalıklarıyla bir tarama çalıştırmak için sudo nmap -sS <hedef> gibi komut adının önüne sudo ekleyin. Sizden normal oturum açma parolanız olan bir parola istenecektir. Bunu yalnızca yönetici ayrıcalıklarına sahip kullanıcılar yapabilir.

Zenmap, X11 uygulamasının yüklenmesini gerektirir. Varsayılan olarak yüklenmemişse, Mac OS X kurulum disklerinde isteğe bağlı bir yükleme olarak mevcut olabilir.

Zenmap başlatıldığında, parolanızı yazmanızı isteyen bir iletişim kutusu görüntülenir. Yönetici ayrıcalıklarına sahip kullanıcılar, Zenmap'in kök kullanıcı olarak çalışmasına ve daha gelişmiş taramalar yapmasına izin vermek için parolalarını girebilirler. Zenmap'i ayrıcalıksız modda çalıştmak için, bu kimlik doğrulama iletişim kutusundaki "İptal" düğmesini seçin.

## **Other Platforms (BSD, Solaris, AIX, AmigaOS) (Diğer Platformlar (BSD, Solaris, AIX, AmigaOS))**

Çoğu Nmap kullanıcısı yazılımı Linux, Windows veya Mac OS X üzerinde çalışmaktadır. Bunları en öncelikli platformlarımız olarak görüyoruz ve her yapının bunları iyi bir şekilde desteklediğinden emin olmak için yapı ve test makineleri bulunduruyoruz.

Nmap ayrıca, kişisel olarak test etmek veya ikili paketleri sık sık oluşturmak için kaynaklara sahip olmadığımız diğer birçok platformda da çalışır. Nmap'in bu sayfadaki platformlar için birinci sınıf desteği sürdürmesine yardımcı olmak için tutkulu bir kullanıcı topluluğuna güveniyoruz ve Nmap'in diğer platformlara genişlediğini görmekten her zaman mutluluk duyuyoruz.

Aşağıdaki bölümlerde Nmap'i belirli platformlarda çalışıtmak için ipuçları verilmektedir.

### **FreeBSD / OpenBSD / NetBSD (FreeBSD / OpenBSD / NetBSD)**

BSD türleri Nmap tarafından iyi bir şekilde desteklenmektedir, bu nedenle "Linux/Unix Derleme ve Kaynak Koddan Kurulum" bölümünde açıklanıldığı gibi basitçe kaynaktan derleyebilirsiniz. Bu, her zaman en son sürüm ve esnek bir derleme sürecine sahip olmanın normal avantajlarını sağlar. İkili paketleri tercih ediyorsanız, bu \*BSD varyantlarının her biri kendi Nmap paketlerini korur. Birçok BSD sistemi ayrıca popüler uygulamaların derlenmesini standartlaştıran bir port ağacına sahiptir. Nmap'i en popüler \*BSD varyantlarına yüklemek için talimatlar aşağıdadır.

İkili paketler kullanarak kurulum

1. <http://www.openbsd.org/ftp.html> adresinden bir yansıtma seçin, ardından FTP'ye girin ve /pub/OpenBSD/<version>/packages/<platform>/nmap-<version>.tgz adresinden Nmap paketini alın. Ya da OpenBSD dağıtım CD-ROM'undan edinin.
2. Kök olarak şunu çalıştırın: pkg\_add -v nmap-<version>.tgz

Kaynak bağlantı noktaları ağacını kullanarak kurulum

1. Port ağacının bir kopyasına sahip değilseniz, <http://openbsd.org/faq/faq15.html> adresindeki talimatları kullanarak CVS aracılığıyla edinin.

2. Kök olarak aşağıdaki komutu çalıştırın (/usr/ports dizinini farklısa yerel port dizinizle değiştirin):

```
cd /usr/ports/net/nmap && make install clean
```

### **FreeBSD Binary Package and Source Ports Instructions (FreeBSD İkili Paket ve Kaynak Portları Talimatları)**

FreeBSD projesinin El Kitabında paket ve port yükleme süreçlerini açıklayan bir bölüm bulunmaktadır. Sürecin kısa bir özeti aşağıdadır.

#### **Installation of the binary package (İkili paketin kurulumu)**

İkili Nmap paketini kurmanın en kolay yolu pkg\_add -r nmap komutunu çalıştırmaktır. Daha sonra X-Window önyüzünü istiyorsanız aynı komutu zenmap argümanıyla çalıştırabilirsiniz. Bunun yerine paketi elle edinmek isterseniz, <http://freshports.org/security/nmap> ve <http://freshports.org/security/zenmap> adreslerinden veya CDROM'dan alın ve pkg\_add <packagename.tgz> komutunu çalıştırın.

#### **Installation using the source ports tree (Kaynak bağlantı noktaları ağacını kullanarak kurulum)**

1. Ports ağacı genellikle sistemin kendisiyle birlikte yüklenir (genellikle /usr/ports içinde). Henüz sahip değilseniz, özel kurulum talimatları yukarıda referans verilen FreeBSD El Kitabı bölümünde verilmiştir.
2. Kök olarak aşağıdaki komutu çalıştırın (/usr/ports dizinini farklısa yerel port dizinizle değiştirin):

```
cd /usr/ports/security/nmap && make install clean
```

### **NetBSD Binary Package Instructions (NetBSD İkili Paket Talimatları)**

NetBSD, Nmap'i normal i386'dan PlayStation 2, PowerPC, VAX, SPARC, MIPS, Amiga, ARM ve adını bile duymadığım birkaç platforma kadar çok sayıda platform için paketlemiştir! NetBSD Nmap paketlerinin bir listesi <ftp://ftp.netbsd.org/pub/NetBSD/packages/pkgsrc/net/nmap/README.html> adresinde ve uygulamaları yüklemek için paket sistemlerini kullanmanın bir açıklaması <http://netbsd.org/Documentation/pkgsrc/using.html> adresinde mevcuttur.

### **Oracle/Sun Solaris**

Solaris uzun zamandır Nmap tarafından iyi bir şekilde desteklenmektedir, ancak bu şekilde kalmasına yardımcı olmak için Nmap topluluğuna büyük ölçüde güveniyoruz. Nmap'i "Linux/Unix Derleme ve Kaynak Koddan Kurulum" bölümünde açıkladığı gibi kaynaktan derlemenizi ve kurmanızı öneririz. Sorun yaşarsanız, "Hatalar" bölümünde açıkladığı gibi nmap-dev posta listesine tüm ayrıntıları içeren bir rapor göndermeyi deneyin. Ayrıca Solaris desteğini iyileştiren bir yama geliştirirseniz bize bildirin, böylece diğer Solaris kullanıcılarının yararına Nmap'e dahil edebiliriz.

## IBM AIX

Nmap, IBM AIX üzerinde "Linux/Unix Kaynak Kodundan Derleme ve Kurulum" adlı bölümdeki yönergeler izlenerek kaynak kodundan kurulabilir. Yalnızca birkaç ayrıntıya dikkat etmeniz gereklidir.

xlc değil, gcc derleyicisini kullanmalısınız. Nmap'in configure betiği, PATH ortam değişkeninde bir yerdeyse gcc'yi otomatik olarak bulacaktır.

Varsayılan as assembler'in bazı sürümleri ya çöküyor ya da bağlanamayan nesne dosyaları üretiyor. Bunun gibi bir derleyici çıktısı görürseniz olan budur:

```
g++: internal compiler error: Segmentation fault (program as)
Please submit a full bug report,
with preprocessed source if appropriate.
See <http://gcc.gnu.org/bugs.html> for instructions.
```

```
ld: 0711-596 SEVERE ERROR: Object ../../nsock/src/libnsock.a[nsock_core.o]
      An RLD for section 2 (.data) refers to symbol 1794,
      but the storage class of the symbol is not C_EXT or C_HIDEXT.
```

Bu sorunu GNU binutils'ten as yükleyerek aşabilirsiniz. (Ancak ld'yi değil; varsayılan ld'yi kullanmaya devam etmek istersiniz.) Bu talimatlar <http://ftp.gnu.org/gnu/binutils> adresinden binutils-2.22 ile AIX 7.1 üzerinde test edilmiştir.

```
$ bzip2 -dc binutils-2.22.tar.bz2 | tar -xvf -
$ cd binutils-2.22
$ ./configure --disable-werror --disable-largefile CFLAGS="-O2 -Wall"
$ gmake
$ cd gas
$ su
# gmake install
```

Bu, /usr/local/bin içinde olduğu gibi kurulacaktır. Özel CFLAG'ler -g'yi atlar, aksi takdirde çalışmaya çalıştığınız hatalardan birine neden olur. Nmap'i oluştururken ve yapılandırırken /usr/local/bin dosyasının PATH'te /usr/bin dosyasından önce geldiğinden emin olmalısınız.

```
$ export PATH="/usr/local/bin:$PATH"
```

Bazı durumlarda GCC, assemblerler için mutlak bir yol kullanacak şekilde yapılandırılır. Bu durumda, varsayılan derleyiciyi geçici olarak yoldan çekmeniz gerekecektir. Bunun böyle olup olmadığını gcc'ye -print-prog-name=as seçeneğini ileterek test edebilirsiniz:

```
$ gcc -print-prog-name=as
/usr/bin/as
```

Eğer /usr/bin/as çıktısını görürseniz, mv /usr/bin/as /usr/bin/as.backup gibi bir komutla as sistemini devre dışı bırakmalısınız. Eğer as çıktısını görüyorsanız, başka bir değişiklik yapmanız gerekmeyez.

Şimdi "Linux/Unix Kaynak Kodundan Derleme ve Kurulum" adlı bölümdeki talimatları izleyin.

## AmigaOS

Açık kaynak geliştirmenin harikalarından biri, kaynakların çoğu şirketin yaptığı gibi yalnızca kâra odaklanmak yerine, genellikle insanların heyecan verici bulduğu şeylere yönlendirilmesidir. Amiga portu da bu doğrultuda ortaya çıkmıştır. Diego Casorran işin çoğunu gerçekleştirdi ve ana Nmap dağıtımına entegre edilen temiz bir yama gönderdi. Genel olarak, AmigaOS kullanıcıları "Linux/Unix Derleme ve Kaynak Koddan Kurulum" bölümündeki kaynak derleme talimatlarını basitçe takip

edebilmelidir. Bazı sistemlerde birkaç engelle karşılaşabilirsiniz, ancak bunun Amiga fanatikleri için eğlencenin bir parçası olduğunu varsayıyorum.

### **Other proprietary UNIX (HP-UX, IRIX, etc.) (Diğer tescilli UNIX (HP-UX, IRIX, vb.))**

Nmap geçmişte HP-UX ve SGI IRIX gibi birçok tescilli Unix çeşidini desteklemiştir. Bu sistemler için yeterli desteği sürdürmek için büyük ölçüde kullanıcı topluluğuna güveniyoruz. Sorun yaşarsanız, "Hatalar" bölümünde açıklandığı gibi, nmap-dev posta listesine tüm ayrıntıları içeren bir rapor göndermeyi deneyin. Ayrıca platformunuzdaki desteği iyileştiren bir yama geliştirirseniz bize bildirin, böylece bunu Nmap'e dahil edebiliriz.

## **Removing Nmap (Nmap'i Kaldırma)**

Nmap'i kaldırma amacınız sadece en son sürümü yükseltmekse, genellikle çoğu ikili paket yöneticisi tarafından sağlanan yükseltme seçeneğini kullanabilirsiniz. Benzer şekilde, en son kaynak kodunu yüklemek ("Linux/Unix Kaynak Kodundan Derleme ve Yükleme" bölümünde açıklandığı gibi) genellikle önceki kaynaktan yüklemelerin üzerine yazar. Yükleme yöntemlerini değiştiryorsanız (kaynaktan RPM'ye veya tersi gibi) veya artık Nmap kullanmıyorsanız ve tükettiği birkaç megabaytlık disk alanını önemsiyorsanız Nmap'i kaldırmak iyi bir fikirdir.

Nmap'in nasıl kaldırılacağı başlangıçta nasıl kurduğunuza bağlıdır (önceki bölümlere bakın). Kaldırma (ve diğer bakım) kolaylığı çoğu ikili paketin önemli bir avantajıdır. Örneğin, Nmap Linux dağıtımlarında yaygın olan RPM sistemi kullanılarak kurulduğunda, root olarak rpm -e nmap zenmap komutu çalıştırılarak kaldırılabilir. Benzer seçenekler diğer paket yöneticileri tarafından da sunulmaktadır; daha fazla bilgi için kendi belgelerine başvurun.

Nmap'i Windows yükleyicisinden yüklediyseniz, Denetim Masası'nı açın, "Program Ekle veya Kaldır"ı seçin ve Nmap için "Kaldır" düğmesini seçin. Wireshark gibi diğer uygulamalar için ihtiyacınız yoksa Npcap'i de kaldırabilirsiniz.

Eğer Nmap'i kaynak koddan kurduysanız, kaldırmak biraz daha zordur. Eğer hala derleme dizinine sahipseniz (make install komutunu çalıştırıldığınız yer), make uninstall komutunu çalıştırarak Nmap'i kaldırabilirsiniz. Artık bu derleme dizinine sahip değilseniz, Nmap sürüm numarasını elde etmek için nmap -V yazın. Daha sonra <https://nmap.org/dist/> veya <https://nmap.org/dist-old/> adresinden Nmap'in o sürümü için kaynak tarball'u indirin. Tarball'un sıkıştırmasını açın ve yeni

oluşturulan dizine (nmap-<version>) geçin. İlk seferde belirttiğiniz kurulum yolu seçeneklerini (--prefix veya --datadir gibi) de dahil ederek ./configure dosyasını çalıştırın. Ardından make uninstall komutunu çalıştırın. Alternatif olarak, Nmap ile ilgili tüm dosyaları silebilirsiniz. Nmap'in 4.50 veya daha yüksek sürümlerinin varsayılan kaynak yüklemesini kullandığınız, aşağıdaki komutlar bunu kaldırır.

```
# cd /usr/local  
# rm -f bin/nmap bin/nmapfe bin/xnmap  
# rm -f man/man1/nmap.1 man/man1/zenmap.1  
# rm -rf share/nmap  
# ./bin/uninstall_zenmap
```

Nmap'i ilk kurarken --prefix veya başka bir install-path seçeneği belirttiyseniz yukarıdaki komutları biraz ayarlamanz gerekebilir. Zenmap ön ucunu yüklememişseniz zenmap, nmapfe ve xnmap ile ilgili dosyalar mevcut değildir.

## Chapter 3. Host Discovery ("Ping Scanning") (Bölüm 3. Ana Bilgisayar Bulma ("Ping Taraması"))

### İçindekiler

- Introduction (Giriş)
- Specifying Target Hosts and Networks (Hedef Ana Bilgisayarları ve Ağları Belirleme)
  - Input From List (`iL`) (Listeden Giriş (-iL))
  - Choose Targets at Random (`iR <numtargets>`) (Hedefleri Rastgele Seçme (-iR <numtargets>))
  - Excluding Targets (`-exclude`, `--excludefile <filename>`) (Hedefleri Hariç Tutma (-- exclude, --excludefile <dosya adı>))
  - Practical Examples (Pratik Örnekler )
- Finding an Organization's IP Addresses (Bir Kuruluşun IP Adreslerini Bulma )
  - DNS Tricks (DNS Hileleri )

- Whois Queries Against IP Registries (IP Kayıtlarına Karşı Whois Sorguları )
- Internet Routing Information (İnternet Yönlendirme Bilgileri )
- DNS Resolution (DNS Çözünürlüğü )
- Host Discovery Controls (Ana Bilgisayar Bulma Kontrolleri)
  - List Scan ( `sL` ) (Liste Taraması (-sL))
  - Disable Port Scan ( `sn` ) (Port Taramasını Devre Dışı Bırak (-sn))
  - Disable Ping ( `Pn` ) (Ping'i Devre Dışı Bırak (-Pn))
- Host Discovery Techniques (Ana Bilgisayar Bulma Teknikleri)
  - TCP SYN Ping ( `PS <port list>` ) (TCP SYN Ping (-PS<port listesi>))
  - TCP ACK Ping ( `PA <port list>` ) (TCP ACK Ping (-PA<port listesi>) )
  - UDP Ping ( `PU <port list>` ) (UDP Ping (-PU<port listesi>))
  - ICMP Ping Types ( `PE` , `PP` , and `PM` ) (ICMP Ping Türleri (-PE, -PP ve -PM))
  - IP Protocol Ping ( `PO <protocol list>` ) (IP Protokolü Ping (-PO<protokol listesi>))
  - ARP Scan ( `PR` ) ( ARP Taraması (-PR) )
  - Default Combination (Varsayılan Kombinasyon )
- Putting It All Together: Host Discovery Strategies (Hepsini Bir Araya Getirme: Ana Bilgisayar Bulma Stratejileri )
  - Related Options (İlgili Seçenekler)
  - Choosing and Combining Ping Options (Ping Seçeneklerini Seçme ve Birleştirme)
    - Most valuable probes ( En değerli probalar )
    - TCP probe and port selection (TCP probu ve port seçimi)
    - UDP port selection ( UDP port seçimi )
    - ICMP probe selection (ICMP prob seçimi )
    - Designing the ideal combinations of probes (İdeal prob kombinasyonlarını tasarlama)

- Host Discovery Code Algorithms (Ana Bilgisayar Bulma Kodu Algoritmaları)

## **Introduction (Giriş)**

Herhangi bir ağ keşif görevindeki ilk adımlardan biri, (bazen çok büyük) bir IP aralığı kümесini aktif veya ilginç ana bilgisayarların bir listesine indirmektedir. Her bir IP adresinin her bir portunu taramak yavaş ve genellikle gereksizdir. Elbette bir ana bilgisayarı neyin ilginç olduğu büyük ölçüde tarama amaçlarına bağlıdır. Ağ yöneticileri yalnızca belirli bir hizmeti çalıştırın ana bilgisayarlarla ilgilenebilirken, güvenlik denetçileri IP adresi olan her bir cihazla ilgilenebilir. Bir yönetici iç ağındaki ana bilgisayarları bulmak için sadece ICMP ping kullanarak rahat edebilirken, harici bir sızma test uzmanı güvenlik duvarı kısıtlamalarından kaçmak için dzinelerce probdan oluşan çeşitli bir set kullanabilir.

Ana bilgisayar keşif ihtiyaçları çok çeşitli olduğundan, Nmap kullanılan teknikleri özelleştirmek için çok çeşitli seçenekler sunar. Ping taraması adına rağmen, bu, her yerde bulunan ping aracıyla ilişkili basit ICMP eko istek paketlerinin çok ötesine geçer. Kullanıcılar bir liste taramasıyla (-sL) veya ping'i devre dışı bırakarak (-Pn) ping adımını tamamen atlayabilir veya çok portlu TCP SYN/ACK, UDP ve ICMP problemlerinin rastgele kombinasyonlarıyla ağı meşgul edebilir. Bu problemlerin amacı, bir IP adresinin gerçekten aktif olduğunu (bir ana bilgisayar veya ağ cihazı tarafından kullanıldığını) gösteren yanıtlar istemektir. Birçok ağda, IP adreslerinin yalnızca küçük bir yüzdesi herhangi bir zamanda aktiftir. Bu durum özellikle 10.0.0.0/8 gibi özel adres alanlarında yaygındır. Bu ağda 16,8 milyon IP var, ancak binden daha az makineye sahip şirketler tarafından kullanıldığını gördüm. Ana bilgisayar keşfi, bu makineleri seyrek olarak tahsis edilmiş IP adresleri denizinde bulabilir.

Bu bölümde ilk olarak Nmap ping taramasının genel olarak nasıl çalıştığı ve üst düzey kontrol seçenekleri ele alınmaktadır. Ardından, nasıl çalışıkları ve her birinin en uygun olduğu zamanlar da dahil olmak üzere belirli teknikler ele alınmaktadır. Nmap birçok ping teknigi sunar çünkü bir hedef ağa giden bir dizi güvenlik duvarı ve yönlendirici filtresini aşmak için genellikle dikkatlice hazırlanmış kombinasyonlar gereklidir. Etkili genel ping tarama stratejileri tartışılmalıdır ve ardından kullanılan algoritmalar düşük seviyeli bir bakış sunulmalıdır.

## Specifying Target Hosts and Networks (Hedef Ana Bilgisayarları ve Ağları Belirleme)

Nmap komut satırında bir seçenek (veya seçenek argümanı) olmayan her şey bir hedef ana bilgisayar belirtimi olarak değerlendirilir. En basit durum, tarama için bir hedef IP adresi veya ana bilgisayar adı belirtmektir.

Bazen bitişik ana bilgisayarlardan oluşan bir ağın tamamını taramak istersiniz. Bunun için Nmap CIDR tarzı adreslemeyi destekler. Bir IPv4 adresine veya ana bilgisayar adına /<numbits> ekleyebilirsiniz ve Nmap, ilk <numbits>'in verilen referans IP veya ana bilgisayar adıyla aynı olduğu her IP adresini tarayacaktır. Örneğin, 192.168.10.0/24, 192.168.10.0 (ikili: 11000000 10101000 00001010 00000000) ile 192.168.10.255 (ikili: 11000000 10101000 00001010 11111111) arasındaki 256 ana bilgisayarı tarayacaktır. 192.168.10.40/24 tam olarak aynı hedefleri tarayacaktır. Scanme.nmap.org ana bilgisayarının 64.13.134.52 IP adresinde olduğu düşünüldüğünde, scanme.nmap.org/16 belirtimi 64.13.0.0 ile 64.13.255.255 arasındaki 65.536 IP adresini tarayacaktır. İzin verilen en küçük değer, tüm Internet'i hedefleyen /0'dır. En büyük değer /32'dir ve tüm adres bitleri sabit olduğu için yalnızca adlandırılmış ana bilgisayıri veya IP adresini tarar.

CIDR notasyonu kısadır ancak her zaman yeterince esnek değildir. Örneğin, 192.168.0.0/16 adresini taramak isteyebilirsiniz ancak .0 veya .255 ile biten IP'leri atlayabilirsiniz çünkü bunlar alt ağ ve yayın adresleri olarak kullanılabilir. Nmap bunu oktet aralığı adresleme yoluyla destekler. Normal bir IP adresi belirtmek yerine, her oktet için virgülle ayrılmış bir sayı veya aralık listesi belirtebilirsiniz. Örneğin, 192.168.0-255.1-254 .0 veya .255 ile biten aralıktaki tüm adresleri atlayacak ve 192.168.3-5,7.1 192.168.3.1, 192.168.4.1, 192.168.5.1 ve 192.168.7.1 adreslerini tarayacaktır. Bir aralığın her iki tarafı da atlanabilir; varsayılan değerler solda 0 ve sağda 255'tir. Tek başına - kullanmak 0-255 ile aynıdır, ancak hedef belirtiminin bir komut satırı seçeneği gibi görünmemesi için ilk sekizlide 0-kullanmayı unutmayın. Aralıkların son oktetlerle sınırlı olması gerekmek: 0-255.0-255.13.37 belirteci, 13.37 ile biten tüm IP adresleri için Internet çapında bir tarama gerçekleştirecektir. Bu tür geniş örneklemeye Internet anketleri ve araştırmaları için yararlı olabilir.

IPv6 adresleri yalnızca tam nitelikli IPv6 adresleri veya ana bilgisayar adları ile belirtilebilir. CIDR ve oktet aralıkları IPv6 için desteklenmez çünkü nadiren kullanışlıdırlar.

Nmap, komut satırında birden fazla ana bilgisayar belirtimini kabul eder ve bunların aynı türde olması gerekmez. nmap scanme.nmap.org 192.168.0.0/8 10.0.0,1,3-7.- komutu beklediğiniz şeyi yapar.

### **Input From List ( -iL ) (Listeden Girdi (-iL))**

Büyük bir ana bilgisayar listesini iletmek komut satırında genellikle gariptir, ancak bu yaygın bir ihtiyaçtır. Örneğin, DHCP sunucunuz taramak istediğiniz 10.000 güncel kiralama listesini dışa aktarabilir. Ya da yetkisiz statik IP adresleri kullanan ana bilgisayarları bulmak için bunlar dışındaki tüm IP adreslerini taramak isteyebilirsiniz. Basitçe taranacak ana bilgisayarların listesini oluşturun ve bu dosya adını -iL seçeneğine bir argüman olarak Nmap'e iletin. Girişler, Nmap tarafından komut satırında kabul edilen biçimlerden herhangi birinde olabilir (IP adresi, ana bilgisayar adı, CIDR, IPv6 veya sekizli aralıklar). Her girdi bir veya daha fazla boşluk, sekme veya satırsonu ile ayrılmalıdır. Nmap'in ana bilgisayarları gerçek bir dosya yerine standart girdiden okumasını istiyorsanız dosya adı olarak bir kısa çizgi (-) belirtebilirsiniz.

### **Choose Targets at Random ( iR <numtargets> ) (Hedefleri Rastgele Seçme (-iR <numtargets>) )**

İnternet çapında anketler ve diğer araştırmalar için hedefleri rastgele seçmek isteyebilirsiniz. Bu, üretilecek IP sayısını argüman olarak alan -iR seçeneği ile yapılır. Nmap, özel, çok noktaya yayın veya ayrılmamış adres aralıkları gibi bazı istenmeyen IP'leri otomatik olarak atlar. Hiç bitmeyen bir tarama için 0 argümanı belirtilebilir. Bazı ağ yöneticilerinin ağlarının izinsiz taramasına karşı çıktığını unutmayın. iR kullanmadan önce "Yasal Sorunlar" başlıklı bölüm dikkatlice okuyun.

Yağmurlu bir öğleden sonra canınız çok sıkılırsa nmap -sS -PS80 -iR 0 -p 80 komutunu deneyerek rastgele web sunucularına göz atabilirsiniz.

### **Excluding Targets ( -exclude , -excludefile <filename> ) (Hedefleri Hariç Tutma (--exclude, --excludefile <dosya adı>))**

Hiçbir koşulda taramak istemediğiniz makinelerin olması yaygın bir durumdur. Makineler o kadar kritik olabilir ki olumsuz bir tepki riskini göze alamazsınız. Nmap taramasının bununla hiçbir ilgisi olmasa bile tesadüfi bir kesinti için suçlanabilirsiniz. Ya da belki de tarandığında çöktüğü bilinen eski bir donanımınız var, ancak henüz tamir edemediniz ya da değiştiremediniz. Ya da belirli IP

aralıkları, tarama yetkiniz olmayan yan şirketleri, müşterileri veya iş ortaklarını temsil ediyor olabilir. Danışmanlar genellikle kendi makinelerinin müşterilerinin ağlarının taranmasına dahil edilmesini istemezler. Sebep ne olursa olsun, --exclude seçeneği ile ana bilgisayarları veya tüm ağları hariç tutabilirsiniz. Bu seçeneğe normal Nmap sözdizimini kullanarak hariç tutulan hedeflerin ve netblokların virgülle ayrılmış bir listesini aktarmanız yeterlidir. Alternatif olarak, hariç tutulan ana bilgisayarların/ağların bir dosyasını oluşturabilir ve bunu --excludefile seçeneği ile Nmap'e iletibilirsiniz. --exclude seçeneği virgül kullanan IP aralıklarıyla (192.168.0.10,20,30) karışmaz çünkü --exclude'un kendisi virgül kullanır. Bu durumlarda --excludefile seçeneğini kullanın.

### **Practical Examples (Pratik Örnekler)**

Bazı araçlar sadece bir ana bilgisayar listesine izin veren veya bir aralık için başlangıç ve bitiş IP adreslerini belirtmenize izin veren basit arayzlere sahipken, Nmap çok daha güçlü ve esnektir. Ancak Nmap'i öğrenmek daha zor olabilir ve yanlış IP adreslerini taramak bazen felaketle sonuçlanabilir. Neyse ki, Nmap liste taramasını (-sL seçeneği) kullanarak bir kuru çalışma sunar. Gerçekte yapmadan önce hangi IP'lerin taranacağını görmek için nmap -sL -n <hedefler> komutunu çalıştırmanız yeterlidir.

Örnekler, Nmap ana bilgisayar belirtimi sözdizimini öğretmenin en etkili yolu olabilir. Bu bölümde en basitinden başlayarak bazı örnekler verilmektedir.

#### **nmap scanme.nmap.org, nmap scanme.nmap.org/32, nmap 64.13.134.52**

Bu üç komutun hepsi, scanme.nmap.org adresinin 64.13.134.52 adresine çözümlendiğini varsayıarak aynı şeyi yapar. Tek bir IP'yi tararlar ve sonra çıkarlar.

#### **nmap scanme.nmap.org/24, nmap 64.13.134.52/24, nmap 64.13.134.-, nmap 64.13.134.0-255**

Bu dört komutun hepsi Nmap'ten 64.13.134.0 ile 64.13.134.255 arasındaki 256 IP adresini taramasını ister. Başka bir deyişle, scanme.nmap.org adresini çevreleyen C sınıfı boyutlu adres alanının taranmasını isterler.

#### **nmap 64.13.134.52/24 --exclude scanme.nmap.org,insecure.org**

Nmap'e 64.13.134.52 çevresindeki C sınıfını taramasını, ancak bu adres aralığında bulunurlarsa scanme.nmap.org ve insecure.org adreslerini atlamasını söyler.

#### **nmap 10.0.0.0/8 --exclude 10.6.0.0/16,ultra-sensitive-host.company.com**

Nmap'e 10.6 ile başlayan her şeyi ve ultra hassas-host.company.com'u atlaması dışında tüm özel 10 aralığını taramasını söyler.

```
egrep '^lease' /var/lib/dhcp/dhcpd.leases | awk '{print $2}' | nmap -iL -
```

Atanmış DHCP IP adreslerinin listesini alın ve bunları tarama için doğrudan Nmap'e besleyin. Standart girdiden okumak için -iL'ye bir kısa çizgi geçildiğine dikkat edin.

**nmap -6 2001:800:40:2a03::3**

2001:800:40:2a03::3 adresindeki IPv6 ana bilgisayarını tarar.

## **Finding an Organization's IP Addresses (Bir Kuruluşun IP Adreslerini Bulma )**

Nmap ağ taramasının birçok yönünü otomatikleştirir, ancak yine de hangi ağları tarayacağını söylemeniz gereklidir. Sanırım -iR belirtebilir ve Nmap'in hedef şirketinizi rastgele vurmasını umabilir ya da tüm Internet'i taramak için 0.0.0.0/0 belirtmek gibi kaba kuvvet yöntemini deneyebilirsiniz. Ancak bu seçeneklerden her ikisi de aylar ya da yıllar alabilir ve muhtemelen başınızı belaya sokabilir. Bu nedenle hedef ağ bloklarını taramadan önce dikkatlice araştırmak önemlidir. Meşru bir sizma testi yapıyor olsanız ve müşteri size netbloklarının bir listesini vermiş olsa bile, bunları iki kez kontrol etmek önemlidir. Müşteriler bazen güncel olmayan kayıtlara sahip olabilir veya bunları yanlış yazabilir. Yanlışlıkla yanlış şirkete girerseniz, müşteriniz tarafından imzalanmış bir yetkilendirme mektubu size yardımcı olmayacağındır.

Çoğu durumda, yalnızca bir şirketin alan adıyla başlarsınız. Bu bölümde, hedef şirketin sahip olduğu, işlettığı veya bağlı olduğu netblokların bir listesine dönüştürmenin en yaygın ve etkili yollarından birkaçı gösterilmektedir. Tipik Linux komut satırı yardımcı programları gösterilmektedir, ancak diğer platformlar için de benzer araçlar mevcuttur.

2006'daki ShmooCon konferansında bir arkadaş yanına geldi ve Nmap dokümantasyonunun target.com'u taramak için birçok örnek yol belirttiğinden şikayet etti. ICANN'in example.com alan adını bu amaç için ayırdığını belirtti ve man sayfasını buna göre revize etmem için bana baskı yaptı. Teknik olarak haklı olsa da, takıntı haline getirmesi garip bir şeydi. Bana kartvizitini uzattığında motivasyonu netlesetti:

Şekil 3.1. Bir kartvizit her şeyi açıklar



Görünüşe göre, birçok Nmap kullanıcısı örnekleri doğrudan man sayfasından kopyaladı ve hedef belirtecini değiştirmeden çalıştırıldı. Böylece target.com taramalar ve ilgili IDS uyarıları ile dolup taştı. Bu olayın şerefine, bu bölümün amacı Target Corporation tarafından atanınan ve kullanılan IP aralıklarını belirlemektir.

### DNS Tricks (DNS Hileleri )

DNS'in birincil amacı alan adlarını IP adreslerine çözümlemektir, bu nedenle başlamak için mantıklı bir yerdır. Örnek 3.1'de, bazı yaygın DNS kayıt türlerini sorgulamak için Linux host komutunu kullanıyorum.

Örnek 3.1. Yaygın DNS kayıt türlerini sorgulamak için host komutunu kullanma

```

> host -t ns target.com
target.com name server ns4.target.com.
target.com name server ns3.target.com.
target.com name server ns1-auth.sprintlink.net.
target.com name server ns2-auth.sprintlink.net.
target.com name server ns3-auth.sprintlink.net.
> host -t a target.com
target.com has address 161.225.130.163
target.com has address 161.225.136.0
> host -t aaaa target.com
target.com has no AAAA record
> host -t mx target.com
target.com mail is handled by 50 smtp02.target.com.
target.com mail is handled by 5 smtp01.target.com.
> host -t soa target.com
target.com has SOA record extdns02.target.com. hostmaster.target.com.

```

Daha sonra yukarıdaki ana bilgisayar adları için IP adreslerini çözümlüyorum (tekrar ana bilgisayar kullanarak) ve www.target.com ve ftp.target.com gibi birkaç yaygın alt alan adını deniyorum. ns3.target.com ve smtp01.target.com gibi isimlerle başlayarak, yeni makineler bulmak için rakamları değiştirmeyi deniyorum. Tüm bunlar beni aşağıdaki target.com adları ve adresleriyle baş başa bırakıyor:

Tablo 3.1. target.com IP'lerini listelemeye ilk geçiş

Hostname	IP Addresses
ns3.target.com	161.225.130.130
ns4.target.com	161.225.136.136
ns5.target.com	161.225.130.150
target.com	161.225.136.0, 161.225.130.163
smtp01.target.com	161.225.140.120
smtp02.target.com	198.70.53.234, 198.70.53.235
extdns02.target.com	172.17.14.69
www.target.com	207.171.166.49

Bu şekilde önemli bir ana bilgisayar adı listesi oluşturulabilse de, ana bilgisayar adlarının ana kaynağı bir bölge aktarımından gelir. Çoğu DNS sunucusu artık bölge aktarma isteklerini reddetmektedir, ancak birçoğu hala izin verdiği için denemeye değer. Alan NS kayıtları ve kurumsal IP aralıklarını tarayarak bulduğunuz her DNS

sunucusunu denedığınızden emin olun. Şimdiye kadar yedi Target ad sunucusu bulduk: ns3.target.com, ns4.target.com, ns5.target.com, ns1-auth.sprintlink.net, ns2-auth.sprintlink.net, ns3-auth.sprintlink.net ve extdns02.target.com. Ne yazık ki, bu sunucuların tümü aktarımı reddetti veya bölge aktarımı için gereken TCP DNS bağlantılarını desteklemiyordu. Örnek 3.2'de yaygın dig (domain information groper) aracı[9] kullanılarak yapılan başarısız bir target.com bölge aktarımı denemesi ve ardından ilgisiz bir kuruluşu (cpsr.org) karşı yapılan başarılı bir deneme gösterilmektedir.

### Örnek 3.2. Bölge aktarımı başarısızlığı ve başarısı

```
> dig @ns2-auth.sprintlink.net -t AXFR target.com
; <>> DiG 9.5.0b3 <>> @ns2-auth.sprintlink.net -t AXFR target.com

; Transfer failed.

> dig @ns2.eppi.com -t AXFR cpsr.org
; <>> DiG 9.5.0b1 <>> @ns2.eppi.com -t AXFR cpsr.org

cpsr.org.          10800  IN      SOA    ns1.findpage.com. root(cpsr.org.
cpsr.org.          10800  IN      NS     ns.stimpy.net.
cpsr.org.          10800  IN      NS     ns1.findpage.com.
cpsr.org.          10800  IN      NS     ns2.eppi.com.
cpsr.org.          10800  IN      A      208.96.55.202
cpsr.org.          10800  IN      MX    0 smtp.electricembers.net.
diac.cpsr.org.    10800  IN      A      64.147.163.10
groups.cpsr.org.  10800  IN      NS     ns1.electricembers.net.
localhost.cpsr.org. 10800  IN      A      127.0.0.1
mail.cpsr.org.    10800  IN      A      209.209.81.73
peru.cpsr.org.    10800  IN      A      208.96.55.202
www.peru.cpsr.org. 10800  IN      A      208.96.55.202
[...]
```

Bu gibi ileri DNS sonuçlarını toplarken yapılan yaygın bir hata, bir alan adı altında bulunan tüm sistemlerin o kuruluşun ağının bir parçası olması ve taramasının güvenli olduğunu varsaymaktadır. Aslında, hiçbir şey bir kuruluşun Internet üzerinde herhangi bir yeri işaret eden kayıtlar eklemesini engellemez. Bu genellikle, markalaşma için kaynak alan adını tutarken üçüncü taraflara dış kaynak hizmetleri sağlamak için yapılır. Örneğin, www.target.com 207.171.166.49 olarak çözümlenir. Bu Target'ın ağının bir parçası mı yoksa taramak istemeyebileceğimiz bir üçüncü taraf tarafından mı yönetiliyor? Üç hızlı ve kolay test DNS ters çözümleme, traceroute ve ilgili IP adresi kayıt defterine karşı whois'tir. İlk iki adım Nmap ile

yapılabilirken, Linux whois komutu üçüncüsü için iyi çalışır. target.com'a karşı yapılan bu testler Örnek 3.3 ve Örnek 3.4'te gösterilmektedir.

Örnek 3.3. www.target.com adresine karşı Nmap reverse-DNS ve traceroute taraması

```
# nmap -Pn -T4 --traceroute www.target.com

Starting Nmap ( https://nmap.org )
Nmap scan report for 166-49.amazon.com (207.171.166.49)
Not shown: 998 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

TRACEROUTE (using port 80/tcp)
HOP RTT      ADDRESS
[cut]
9  84.94  ae-2.ebr4.NewYork1.Level3.net (4.69.135.186)
10 87.91  ae-3.ebr4.Washington1.Level3.net (4.69.132.93)
11 94.80  ae-94-94.csv4.Washington1.Level3.net (4.69.134.190)
12 86.40  ae-21-69.carl.Washington3.Level3.net (4.68.17.7)
13 185.10  AMAZONCOM.carl.Washington3.Level3.net (4.71.204.18)
14 84.70  72.21.209.38
15 85.73  72.21.193.37
16 85.68  166-49.amazon.com (207.171.166.49)

Nmap done: 1 IP address (1 host up) scanned in 20.57 seconds
```

Örnek 3.4. www.target.com IP adresinin sahibini bulmak için whois'i kullanma

```
> whois 207.171.166.49
[Querying whois.arin.net]
[whois.arin.net]

OrgName:    Amazon.com, Inc.
OrgID:      AMAZON-4
Address:    605 5th Ave S
City:       SEATTLE
StateProv:  WA
PostalCode: 98104
Country:    US
[...]
```

Örnek 3.3'te, ters DNS (iki yer) ve ilginç traceroute sonuçları kalın olarak gösterilmiştir. Amazon.com alan adı, web sitesinin Target'ın kendisi tarafından değil de Amazon tarafından işletildiğini kuvvetle muhtemel kılmaktadır. Ardından whois sonuçlarının IP alanı sahibi olarak "Amazon.com, Inc. "i göstermesi tüm şüpheleri ortadan kaldırıyor. Web sitesi Target markalı, ancak en alta "Powered by Amazon.com" ibaresi yer alıyor. Target tarafından güvenliklerini test etmek üzere işe alınmış olsaydık, bu adres alanına dokunmak için Amazon'dan ayrıca izin almamız gereklidir.

Web veritabanları, belirli bir alan adı altındaki ana bilgisayar adlarını bulmak için de kullanılabilir. Örneğin, Netcraft'in <http://searchdns.netcraft.com/?host> adresinde bir web sitesi DNS arama özelliği vardır. Şekil 3.2'de gösterildiği gibi, forma [.target.com](#) yazmak 36 sonuç getirmektedir. Kullanışlı tabloları netblock sahibini de gösteriyor, bu da Amazon'un [www.target.com](#) çalıştırması gibi durumları yakalıyor. Keşfedilen bazı ana bilgisayarları zaten biliyoruz, ancak [sendasmoochie.target.com](#) gibi isimleri tahmin etmemiz pek mümkün olmazdı.

Şekil 3.2. Netcraft 36 Hedef web sunucusu bulur

Site	Site Report	First seen	Netblock	OS
1. <a href="#">www.target.com</a>	<a href="#">Report</a>	October 1995	Amazon.com, Inc.	unknown
2. <a href="#">weeklyad.target.com</a>	<a href="#">Report</a>	January 2005	Akamai Technologies	Linux
3. <a href="#">sites.target.com</a>	<a href="#">Report</a>	August 2005	Target Corporation	F5 Big-IP
4. <a href="#">redcard.target.com</a>	<a href="#">Report</a>	November 2005	Target Corporation	F5 Big-IP
5. <a href="#">www.target.com.au</a>	<a href="#">Report</a>	June 2000	APNIC	Windows 2000
6. <a href="#">targetrewards.target.com</a>	<a href="#">Report</a>	August 2005	Target Corporation	F5 Big-IP
7. <a href="#">cinemared.target.com</a>	<a href="#">Report</a>	August 2005	Target Corporation	F5 Big-IP
8. <a href="#">recipes.target.com</a>	<a href="#">Report</a>	November 2005	Allrecipes.com, Inc.	Windows Server 2003
9. <a href="#">bookmarked.target.com</a>	<a href="#">Report</a>	September	Implex.net	Linux

Google da site:[target.com](#) gibi sorgularla bu amaçla kullanılabilir.

### Whois Queries Against IP Registries (IP Kayıtlarına Karşı Whois Sorguları)

Bir dizi ilk "tohum" IP bulunduktan sonra, beklediğiniz şirkete ait olduklarından emin olmak ve hangi netblokların parçası olduklarını belirlemek için

araştırılmalıdır. Küçük bir şirket 1-16 IP adresinden oluşan küçük bir tahsis sahip olabilirken, daha büyük şirketler genellikle binlerce IP adresine sahiptir. Bu bilgiler Kuzey Amerika için ARIN (American Registry for Internet Numbers) ve Avrupa ve Orta Doğu için RIPE gibi bölgesel veri tabanlarında tutulur. Modern whois araçları bir IP adresini alır ve otomatik olarak uygun kayıt defterini sorgular.

Küçük ve orta ölçekli şirketler normalde ARIN gibi kuruluşlar tarafından tahsis edilen IP alanına sahip değildir. Bunun yerine, İSS'lerinden kendilerine netbloklar tahsis edilir. Bazen bu İSS bilgilerini IP sorgularından alırsınız. Bu genellikle size büyük bir netblock bırakır ve bunun hangi kısmının hedefinize tahsis edildiğini bilemezsiniz. Neyse ki, birçok İSS artık Paylaşımı Whois (SWIP) veya Yönlendirme Whois (RWhois) kullanarak müşteri aralıklarını alt delege ediyor. İSS bunu yaptıysa, müşterinin tam netblock boyutunu öğrenirsiniz.

target.com için daha önce keşfedilen IP adreslerinden biri 161.225.130.163 idi. Örnek 3.5'te bu IP'nin sahibini ve IP tahsis bilgilerini belirlemek için bir whois sorgusu (otomatik olarak ARIN'e yönlendirilir) gösterilmektedir.

Örnek 3.5. 161.225.130.163 içeren netblock'u bulmak için whois kullanma

```
> whois 161.225.130.163
[Querying whois.arin.net]
[whois.arin.net]

OrgName: Target Corporation
OrgID: TARGET-14
Address: 1000 Nicollet TPS 3165
City: Minneapolis
StateProv: MN
PostalCode: 55403
Country: US

NetRange: 161.225.0.0 - 161.225.255.255
CIDR: 161.225.0.0/16
NetName: TARGETNET
NetHandle: NET-161-225-0-0-1
Parent: NET-161-0-0-0-0
NetType: Direct Assignment
NameServer: NS3.TARGET.COM
NameServer: NS4.TARGET.COM
Comment:
RegDate: 1993-03-04
Updated: 2005-11-02

OrgTechHandle: DOMAI45-ARIN
OrgTechName: Domainnames admin
OrgTechPhone: +1-612-696-2525
OrgTechEmail: Domainnames.admin@target.com
```

Şaşırıcı olmayan bir şekilde Target, 161.225.0.0'dan 161.225.255.255'e kadar 65.536 IP'nin tamamını kapsayan devasa bir B Sınıfı ağ bloğuna sahip. Kuruluş Adı Target olduğu için bu, İSS'lerinden gelen sonuçları gördüğümüz bir durum değildir.

Bir sonraki adım, bu aralığa girmeyen daha önce keşfedilmiş tüm IP'leri benzer şekilde aramaktır. Daha sonra daha gelişmiş sorgularla başlayabilirsiniz. whois -h whois.arin.net \? ARIN soru sözdizimini verir. Belirli bir adres, OrgID veya OrgTechEmail ile eşleşen tüm netblokları arayabilmeniz güzel olurdu, ancak IP kayıtları genellikle buna izin vermez. Bununla birlikte, diğer birçok yararlı sorguya izin verilir. Örneğin, whois -h whois.arin.net @target.com target.com adresinde e-posta adresi olan tüm ARIN kişilerini gösterir. whois -h whois.arin.net "n target\*" sorusu target ile başlayan tüm netblock tanıtıcılarını gösterir. Büyük/küçük harfe duyarlı değildir. Benzer şekilde, whois -h whois.arin.net "o target\*" target ile başlayan tüm kuruluş adlarını gösterir. Taramak istediğiniz şirketin bir parçası olup

olmadıklarını belirlemek için her girişle ilişkili adres, telefon numarası ve iletişim e-postasına bakabilirsiniz. Genellikle benzer bir isme sahip olan 3. taraflardır.

### **Internet Routing Information (Internet Yönlendirme Bilgileri)**

İnternetin temel yönlendirme protokolü Sınır Ağ Geçidi Protokolüdür (BGP). Orta ölçekli ve büyük kuruluşları tararken, BGP yönlendirme tabloları tüm dünyadaki IP alt ağlarını bulmanıza yardımcı olabilir. Örneğin, Microsoft Corporation'a ait IP adreslerini taramak istediğiniz varsayılmı. microsoft.com için bir DNS araması 207.46.196.115 IP adresini sağlar. Önceki bölümde tartışıldığı gibi bir whois sorgusu, 207.46.0.0/16 bloğunun tamamının Redmond'daki uygun "One Microsoft Way" adresinde Microsoft'a ait olduğunu gösterir. Bu, taranacak 65.536 IP adresi sağlar, ancak BGP tabloları çok daha fazlasını ortaya çıkarır.

Microsoft gibi kuruluşlara yönlendirme amacıyla otonom sistem (AS) numaraları atanır. Belirli bir IP adresi için bildirilen AS numarasını belirlemek için kullanışlı bir araç <http://asn.cymru.com/> adresinde mevcuttur. Bu forma 207.46.0.0 yazıldığında Microsoft'un AS numarası 8075 bulunur. Daha sonra, bu AS'ye yönlendirilen tüm IP öneklerini bulmak istiyorum. Bunu yapmak için kullanışlı bir araç <http://www.robtex.com/as/> adresinde mevcuttur. Bu sayfada AS8075 yazıp Git düğmesine basmak, bulunan 42 öeki gösteren bir özet ekranına götürür. Bu önekler 339.456 IP adresini temsil eder ve BGP sekmesine tıklanarak numaralandırılabilir.

BGP bilgilerini bu gibi konserve web formlarından elde etmek uygun olsa da, yönlendirme verilerini gerçek yönlendiricilerden elde etmek daha eğlencelidir ve daha güçlü özel sorgulara izin verebilir. Çeşitli kuruluşlar böyle bir hizmet sağlamaktadır. Bir örnek için route-views.routeviews.org adresine telnet yapın veya <http://routeviews.org> adresini ziyaret edin. Elbette bu hizmetler verilere salt okunur erişim sağlar. İnterneti ele geçirmek için şeytani bir planın parçası olarak küresel yönlendirme tablolarını değiştirmeniz gerekiyorsa, bu kitabı kapsamı dışındadır.

### **DNS Resolution (DNS Çözünürlüğü)**

Nmap ana bilgisayar keşfinin ana odağı, aῆda hangi ana bilgisayarların çalışır durumda olduğunu ve yanıt verdiği belirlemektir. Bu, hedef alanını daraltır, çünkü var olmayan bir ana bilgisayarı hollywoodizesiniz. Ancak keşfin burada bitmesine

izin vermeyin. Kızlarla (ya da erkeklerle) sırf nefes alıyorlar diye çıkmazsınız ve ağdaki sizilacak kutuları seçmek de özel bir dikkat gerektirir. Harika bir bilgi kaynağı (ağa bağlı ana bilgisayarlar hakkında, potansiyel randevular hakkında değil) alan adı sistemi olan DNS'dir. Güvenlik bilincine sahip kuruluşlar bile genellikle sistemlerinin işlevini ifşa eden isimler atarlar. Kablosuz erişim noktalarına wap ya da wireless, güvenlik duvarlarına fw, firewall ya da fw-1, henüz yayınlanmamış içeriğe sahip geliştirme web sunucularına ise dev, staging, www-int ya da beta adlarının verilmesi alışılmadık bir durum değildir. Chicago ofisi güvenlik duvarı fw.chi olarak adlandırılan şirkette olduğu gibi, konumlar veya departman adları da sıklıkla ifşa edilir.

Varsayılan olarak, Nmap ana bilgisayar bulma problemini yanıt veren her IP için (yani çevrimiçi olanlar) ters-DNS çözümlemesi gerçekleştirir. Eğer ana bilgisayar keşfi -Pn ile atlanırsa, çözümleme tüm IP'ler için gerçekleştirilir. Yavaş standart DNS çözümleme kütüphanelerini kullanmak yerine, Nmap paralel olarak dzinelerce istek gerçekleştiren özel bir saplama çözümleyici kullanır.

Varsayılanlar genellikle iyi çalışsa da, Nmap DNS çözünürlüğünü kontrol etmek için dört seçenek sunar. Bunlar tarama hızını ve toplanan bilgi miktarını önemli ölçüde etkileyebilir.

-n (DNS çözünürlüğü yok)

Nmap'e bulduğu etkin IP adresleri üzerinde asla ters DNS çözümlemesi yapmamasını söyler. DNS, Nmap'in yerleşik paralel saplama çözümleyicisi ile bile yavaş olabileceğinden, bu seçenek tarama sürelerini azaltır.

-R (tüm hedefler için DNS çözünürlüğü)

Nmap'e hedef IP adreslerinde her zaman ters DNS çözümlemesi yapmasını söyler. Normalde ters DNS yalnızca duyarlı (çevrimiçi) ana bilgisayarlara karşı gerçekleştirilir.

--system-dns (Sistem DNS çözümleyicisini kullan)

Varsayılan olarak, Nmap IP adreslerini doğrudan ana bilgisayarınızda yapılandırılmış ad sunucularına sorgular göndererek ve ardından yanıtları dinleyerek çözümler. Performansı artırmak için birçok istek (genellikle dzinelerce) paralel olarak gerçekleştirilir. Bunun yerine sistem çözümleyicinizi kullanmak için bu seçeneği belirtin (getnameinfo çağrıları yoluyla her seferinde bir IP). Nmap paralel çözümleyicisinde bir hata bulmadığınız sürece bu yavaş ve nadiren

kullanışlıdır (bulursanız lütfen bize bildirin). IPv6 taramaları için her zaman sistem çözümleyicisi kullanılır.

`--dns-servers <server1> [<server2> [...]]` (Ters DNS sorguları için kullanılacak sunucular)

Varsayılan olarak, Nmap DNS sunucularınızı (rDNS çözümlemesi için) resolv.conf dosyanızdan (Unix) veya Kayıt Defterinden (Win32) belirler. Alternatif olarak, alternatif sunucuları belirtmek için bu seçeneği kullanabilirsiniz. Eğer `--system-dns` veya IPv6 taraması kullanıyorsanız bu seçenek geçerli değildir. Birden fazla DNS sunucusu kullanmak, özellikle hedef IP alanınız için yetkili sunucular seçerseniz, genellikle daha hızlıdır. Bu seçenek aynı zamanda gizliliği de artırabilir, çünkü talepleriniz Internet üzerindeki hemen hemen tüm özyinelemeli DNS sunucularından geri dönebilir.

Bu seçenek özel ağları tararken de kullanışlıdır. Bazen sadece birkaç isim sunucusu uygun rDNS bilgisi sağlar ve nerede olduklarını bile bilmiyor olabilirsiniz. Ağ 53 numaralı bağlantı noktası için tarayabilir (belki sürüm algılama ile), ardından çalışan bir tane bulana kadar `--dns-servers` ile her bir ad sunucusunu birer birer belirterek Nmap liste taramalarını (`-sL`) deneyebilirsiniz.

## **Host Discovery Controls (Ana Bilgisayar Bulma Kontrolleri)**

Varsayılan olarak, Nmap port taramaları, işletim sistemi tespiti, Nmap Scripting Engine veya sürüm tespiti gibi daha müdahaleci problardan önce bir ping tarama aşaması içerecektir. Nmap genellikle yalnızca ping tarama aşamasında kullanılabilir olduğu gösterilen makineler üzerinde müdahaleci taramalar gerçekleştirir. Bu, her bir IP adresine karşı tam tarama yapmaya kıyasla önemli ölçüde zaman ve bant genişliği tasarrufu sağlar. Ancak bu yaklaşım her koşul için ideal değildir. Her IP'yi taramak istediğiniz zamanlar (`-Pn`) ve port taraması yapmadan ana bilgisayar keşfi yapmak istediğiniz zamanlar (`-sn`) olabilir. Hatta hedef ana bilgisayarları yazdırma ve ping problemleri göndermeden önce çıkmak istediğiniz zamanlar bile vardır (`-sL`). Nmap bu davranışını kontrol etmek için birkaç üst düzey seçenek sunar.

### **List Scan ( `-sL` ) (Liste Tarama (`-sL`))**

Liste taraması, hedef ana bilgisayarlara herhangi bir paket göndermeden, belirtilen ağ(ilar)daki her ana bilgisayarı listeleyen dejener bir ana bilgisayar bulma

biçimidir. Varsayılan olarak, Nmap hala ana bilgisayarların adlarını öğrenmek için ters-DNS çözümlemesi yapar. Nmap ayrıca sonunda toplam IP adresi sayısını da bildirir. Liste taraması, hedefleriniz için doğru IP adreslerine sahip olduğunuzdan emin olmak için iyi bir sağlık kontrolüdür. Ana bilgisayarlar tanımadığınız alan adlarını gösteriyorsa, yanlış şirketin ağını taramayı önlemek için daha fazla araştırmaya değer.

Hedef IP aralıklarının yanlış olmasının birçok nedeni vardır. Ağ yöneticileri bile kendi ağ bloklarını yanlış yazabilir ve pen-test uzmanlarının endişelenmesi gereken daha da fazla şey vardır. Bazı durumlarda, güvenlik danışmanlarına yanlış adresler verilir. Diğerlerinde ise whois veritabanları ve yönlendirme tabloları gibi kaynaklar aracılığıyla uygun IP aralıklarını bulmaya çalışırlar. Veritabanları güncel olmayabilir ya da şirket IP alanını başka kuruluşlara ödünç veriyor olabilir. Kurumsal ebeveynlerin, kardeşlerin, hizmet sağlayıcıların ve iştiraklerin taranıp taranmayacağı, müşteriyle önceden üzerinde çalışılması gereken önemli bir konudur. Bir ön liste taraması tam olarak hangi hedeflerin taranacağıının teyit edilmesine yardımcı olur.

Önceden liste taraması yapmanın bir başka nedeni de gizliliktir. Bazı durumlarda, IDS uyarılarını tetiklemesi ve istenmeyen dikkatleri üzerine çekmesi muhtemel olan hedef ağa tam ölçekli bir saldırısı ile başlamak istemezsınız. Liste taraması göze batmaz ve hangi makinelerin hedef alınacağını seçmede faydalı olabilecek bilgiler sağlar. Hedefin tüm ters-DNS isteklerini fark etmesi pek olası olmasa da mümkündür. Bu bir sorun olduğunda, "DNS proxying" adlı bölümde açıklandığı gibi --dns-servers seçeneğini kullanarak anonim özyinelemeli DNS sunucuları üzerinden sıçrayabilirsiniz.

Liste taraması -sL komut satırı seçeneği ile belirtilir. Amaç sadece hedef ana bilgisayarların bir listesini yazdırmak olduğundan, bağlantı noktası taraması, işletim sistemi algılama veya ping taraması gibi daha üst düzey işlevler için seçenekler -sL ile birleştirilemez. Eğer ping taramasını devre dışı bırakıp bu tür üst düzey işlevleri yerine getirmek istiyorsanız -Pn seçeneği hakkında bilgi edinin. Örnek 3.6, Stanford Üniversitesi ana web sunucusunu çevreleyen CIDR /28 ağ aralığını (16 IP adresi) numaralandırmak için kullanılan liste taramasını göstermektedir.

Örnek 3.6. Liste taraması ile www.stanford.edu çevresindeki ana bilgisayarları numaralandırma

```
felix-> nmap -sL www.stanford.edu/28

Starting Nmap ( https://nmap.org )
Host www9.Stanford.EDU (171.67.16.80) not scanned
Host www10.Stanford.EDU (171.67.16.81) not scanned
Host scriptorium.Stanford.EDU (171.67.16.82) not scanned
Host coursework-a.Stanford.EDU (171.67.16.83) not scanned
Host coursework-e.Stanford.EDU (171.67.16.84) not scanned
Host www3.Stanford.EDU (171.67.16.85) not scanned
Host leland-dev.Stanford.EDU (171.67.16.86) not scanned
Host coursework-preprod.Stanford.EDU (171.67.16.87) not scanned
Host stanfordwho-dev.Stanford.EDU (171.67.16.88) not scanned
Host workgroup-dev.Stanford.EDU (171.67.16.89) not scanned
Host courseworkbeta.Stanford.EDU (171.67.16.90) not scanned
Host www4.Stanford.EDU (171.67.16.91) not scanned
Host coursework-i.Stanford.EDU (171.67.16.92) not scanned
Host leland2.Stanford.EDU (171.67.16.93) not scanned
Host coursework-j.Stanford.EDU (171.67.16.94) not scanned
Host 171.67.16.95 not scanned
Nmap done: 16 IP addresses (0 hosts up) scanned in 0.38 seconds
```

### Disable Port Scan ( **-sn** ) (Bağlantı Noktası Taramasını Devre Dışı Bırak (-sn))

Bu seçenek Nmap'e konak keşfinden sonra bir port taraması yapmamasını söyler. Tek başına kullanıldığında, Nmap'in ana bilgisayar keşfi yapmasını sağlar, ardından taramaya yanıt veren mevcut ana bilgisayarları yazdırır. Bu genellikle "ping taraması" olarak adlandırılır. Port taraması yapılmasa bile, Nmap Scripting Engine (--script) host scriptleri ve traceroute problema (--traceroute) isteyebilirsiniz. Yalnızca pin taraması, liste taramasından bir adım daha müdahalecidir ve genellikle aynı amaçlar için kullanılabilir. Hedef ağıda hızlı bir şekilde ve fazla dikkat çekmeden hafif bir keşif gerçekleştirir. Saldırganlar için kaç ana bilgisayarın açık olduğunu bilmek, liste taramasıyla sağlanan her bir IP ve ana bilgisayar adının listesinden daha değerlidir.

Sistem yöneticileri de genellikle bu seçeneği değerli bulurlar. Bir ağdaki mevcut makineleri saymak veya sunucu kullanılabilirliğini izlemek için kolayca kullanılabilir. Bu genellikle ping taraması olarak adlandırılır ve yayın adresine ping atmaktan daha güvenilirdir çünkü birçok ana bilgisayar yayın sorgularına yanıt vermez.

Örnek 3.7, favori web sitelerimden biri olan Linux Weekly News'i çevreleyen CIDR /24'e (256 IP) karşı hızlı bir ping taramasını göstermektedir.

Örnek 3.7. Ping taraması ile www.lwn.net çevresindeki ana bilgisayarları keşfetme

```
# nmap -sn -T4 www.lwn.net/24

Starting Nmap ( https://nmap.org )
Host 66.216.68.0 seems to be a subnet broadcast address (returned 1 extra ping)
Host 66.216.68.1 appears to be up.
Host 66.216.68.2 appears to be up.
Host 66.216.68.3 appears to be up.
Host server1.camnetsec.com (66.216.68.10) appears to be up.
Host akqa.com (66.216.68.15) appears to be up.
Host asria.org (66.216.68.18) appears to be up.
Host webcubic.net (66.216.68.19) appears to be up.
Host dizzy.yellowdog.com (66.216.68.22) appears to be up.
Host www.outdoorwire.com (66.216.68.23) appears to be up.
Host www.inspectorhosting.com (66.216.68.24) appears to be up.
Host jwebmedia.com (66.216.68.25) appears to be up.
[...]
Host rs.lwn.net (66.216.68.48) appears to be up.
Host 66.216.68.52 appears to be up.
Host cuttlefish.laughingsquid.net (66.216.68.53) appears to be up.
[...]
Nmap done: 256 IP addresses (105 hosts up) scanned in 12.69 seconds
```

Bu örnek yalnızca 13 saniye sürmüştür, ancak değerli bilgiler sağlamaktadır. Bu C sınıfı büyüklükteki adres aralığında 105 ana bilgisayar çevrimiçi. Bu kadar küçük bir IP alanına siğdirilmiş ilgisiz alan adlarından, LWN'nin bir ortak yerleşim veya özel sunucu sağlayıcısı kullandığı açıktır. LWN makinelerinin son derece güvenli olduğu ortaya çıkarsa, bir saldırgan bu komşu makinelerden birinin peşine düşebilir ve ardından Ettercap veya Dsniff gibi araçlarla yerel bir ethernet saldırısı gerçekleştirebilir. Bu verilerin etik bir kullanımı, makineleri bu sağlayıcıya taşımayı düşünen bir ağ yönetici olabilir. Uzun vadeli bir sözleşme imzalamadan ya da pahalı ve yıkıcı bir veri merkezi taşınması yapmadan önce, listelenen kuruluşlardan birkaçına e-posta göndererek hizmet hakkındaki görüşlerini sorabilir.

sn seçeneği varsayılan olarak bir ICMP yankı isteği, 443 numaralı bağlantı noktasına bir TCP SYN paketi, 80 numaralı bağlantı noktasına bir TCP ACK paketi ve bir ICMP zaman damgası isteği gönderir. Ayrıcalıksız Unix kullanıcıları (veya Npcap yüklü olmayan Windows kullanıcıları) bu ham paketleri gönderemediğinden, bu durumlarda bunun yerine yalnızca SYN paketleri gönderilir. SYN paketi, hedef ana bilgisayarın 80 ve 443 numaralı bağlantı noktalarına bir TCP connect sistem çağrıları kullanılarak gönderilir. Ayrıcalıklı bir kullanıcı yerel bir ethernet ağındaki hedefleri taramaya çalıştığında, --send-ip seçeneği belirtilmediği sürece ARP istekleri (-PR) kullanılır.

Daha fazla esneklik için -sn seçeneği "Ana Bilgisayar Bulma Teknikleri" adlı bölümde tartışılan tekniklerden herhangi biriyle birleştirilebilir. Bu prob tipi ve port numarası seçeneklerinden herhangi biri kullanılırsa, varsayılan problar geçersiz kılınır. Nmap çalıştırın kaynak ana bilgisayar ile hedef ağ arasında sıkı güvenlik duvarları varsa, bu gelişmiş tekniklerin kullanılması önerilir. Aksi takdirde, güvenlik duvari probları veya yanıtlarını düşürdüğünde ana bilgisayarlar gözden kaçabilir.

### **Disable Ping ( -Pn ) (Ping'i Devre Dışı Bırak (-Pn))**

Başka bir seçenek de Nmap keşif aşamasını tamamen atlamaktır. Normalde, Nmap bu aşamayı daha ağır tarama için aktif makineleri belirlemek için kullanır. Varsayılan olarak, Nmap yalnızca açık olduğu tespit edilen ana bilgisayarlara karşı bağlantı noktası taramaları, sürüm algılama veya işletim sistemi algılama gibi ağır problema gerçekleştirir. Pn seçeneği ile ana bilgisayar keşfinin devre dışı bırakılması, Nmap'in belirtilen her hedef IP adresine karşı istenen tarama işlevlerini denemesine neden olur. Dolayısıyla, komut satırında B sınıfı boyutunda bir hedef adres alanı (/16) belirtildiğinde, 65.536 IP adresinin tümü taranır. Bir liste taramasında olduğu gibi uygun ana bilgisayar keşfi atlanır, ancak hedef listesini durdurmak ve yazdırma yerine, Nmap her hedef IP aktifmiş gibi istenen işlevleri yerine getirmeye devam eder.

Nmap ping testlerini devre dışı bırakmak için birçok neden vardır. En yaygın olanlarından biri izinsiz güvenlik açığı değerlendirmeleridir. Mevcut tüm ana bilgisayarlardan yanıt almak amacıyla düzinelere farklı ping sondası belirlenebilir, ancak yine de aktif ancak yoğun güvenlik duvarına sahip bir makinenin bu sondaların hiçbirine yanıt vermemesi mümkündür. Bu nedenle denetçiler herhangi bir şeyi kaçırılmamak için hedef ağdaki her IP'ye karşı 65.536 TCP portunun tamamı gibi yoğun taramalar yaparlar. Muhtemelen hiçbir ana bilgisayarın dinlemediği IP adreslerine yüz binlerce paket göndermek savurganlık gibi görünebilir ve tarama sürelerini büyüklik sırasına göre veya daha fazla yavaşlatabilir. Nmap, orijinal probun aktarım sırasında düşmesi durumunda her bağlantı noktasına yeniden iletişim göndermelidir ve Nmap, yanıt vermeyen bu IP adresleri için gidiş-dönüş süresi (RTT) tahmini olmadığından yanıtları beklemek için önemli bir zaman harcamalıdır. Ancak ciddi sizma testi uzmanları, aktif makineleri kaçırma riskinden bile kaçınmak için bu bedeli ödemeye hazırlıdır. Onlar çalışırken büyük -Pn taramasını arka planda çalışmaya bırakarak her zaman hızlı bir tarama da yapabilirler. Bölüm 6, Nmap Performansını Optimize Etmek daha fazla performans ayarlama tavsiyesi sağlar.

Pn kullanmak için sıkça gösterilen bir diğer neden de test edenin zaten açık olduğu bilinen makinelerin bir listesine sahip olmasıdır. Böylece kullanıcı ana bilgisayar bulma aşamasıyla zaman kaybetmenin bir anlamı olmadığını görür. Kullanıcı kendi aktif ana bilgisayar listesini oluşturur ve daha sonra -iL (listeden girdi al) seçeneğini kullanarak Nmap'e iletir. Bu strateji zaman tasarrufu açısından nadiren faydalıdır. Önceki paragrafta tartışılan yeniden iletişim ve RTT tahmini sorunları nedeniyle, büyük bir listedeki yanıt vermeyen bir IP adresinin bile taraması genellikle tüm ping tarama aşamasından daha fazla zaman alacaktır. Buna ek olarak, ping aşaması, özellikle hedef ana bilgisayarın katı güvenlik duvarı kuralları varsa, Nmap'in bir sonraki port taramasını hızlandıracak RTT örneklerini toplamasına izin verir. Pn belirtmek zaman tasarrufu açısından nadiren faydalı olsa da, listenizdeki bazı makineler aksi takdirde belirtilecek olan tüm keşif tekniklerini engelliyorsa önemlidir. Kullanıcılar tarama hızı ile yoğun şekilde gizlenmiş makineleri kaçırma olasılığı arasında bir denge kurmalıdır.

## **Host Discovery Techniques (Ana Bilgisayar Bulma Teknikleri)**

Bir IP adresinin aktif bir ana bilgisayara kayıtlı olup olmadığını bulmanın kolay olduğu günler vardı. Basitçe bir ICMP yankı isteği (ping) paketi gönderin ve yanıt bekleyin. Güvenlik duvarları bu istekleri nadiren engeller ve ana bilgisayarların büyük çoğunluğu itaatkar bir şekilde yanıt verir. Böyle bir yanıt 1989'dan beri RFC 1122 tarafından istenmekte ve bu açıkça "Her ana bilgisayar Yankı İsteklerini alan ve karşılık gelen Yankı Yanıtlarını gönderen bir ICMP Yankı sunucusu işlevi uygulamalıdır" demektedir.

Ne yazık ki ağ kaçıfları için, birçok yönetici güvenlik kaygılarının RFC gerekliliklerinden üstün olduğuna karar vermiş ve ICMP ping mesajlarını engellemiştir. Örnek 3.8, altı popüler Web sitesine karşı yalnızca ICMP Nmap ping taraması kullanır, ancak yalnızca iki yanıt alır. Bu, ICMP ping probleme yanıt verilmemesine dayanarak ana bilgisayarların artık kullanılamaz olarak varsayılamayacağını göstermektedir. Bu örnekteki "-sn -PE" seçenekleri yalnızca ICMP ping taramasını belirtir. R seçeneği Nmap'e tüm ana bilgisayarlara karşı, hatta kapalı olanlara karşı ters-DNS çözümlemesi yapmasını söyler.

Örnek 3.8. Popüler Internet ana bilgisayarlarına ping atma girişimleri

```
# nmap -sn -PE -R -v microsoft.com ebay.com citibank.com google.com \
      slashdot.org yahoo.com

Starting Nmap ( https://nmap.org )
Host origin2.microsoft.com (207.46.250.252) appears to be down.
Host pages.ebay.com (66.135.192.87) appears to be down.
Host ldl-www.citicorp.com (192.193.195.132) appears to be down.
Host 216.239.57.99 appears to be up.
Host slashdot.org (66.35.250.150) appears to be down.
Host w3.rc.dcn.yahoo.com (216.109.127.30) appears to be up.
Nmap done: 6 IP addresses (2 hosts up) scanned in 3.76 seconds
```

Neyse ki Nmap, standart ICMP yankı isteğinin ötesinde çok çeşitli ana bilgisayar bulma teknikleri sunar. Bunlar aşağıdaki bölümlerde açıklanmıştır. Bu bölümde tartışılan -P seçeneklerinden herhangi birini belirtirseniz, varsayılan keşif sondalarına ekleme yapmak yerine bunların yerini alacaklarını unutmayın.

#### TCP SYN Ping ( **-PS <port list>** )

PS seçeneği SYN bayrağı ayarlanmış boş bir TCP paketi gönderir. Varsayılan hedef port 80'dir (nmap.h dosyasında DEFAULT\_TCP\_PROBE\_PORT\_SPEC değiştirilerek derleme zamanında yapılandırılabilir), ancak alternatif bir port parametre olarak belirtilebilir. Bir port listesi belirtilebilir (örneğin -PS22-25,80,113,1050,35000), bu durumda problemler her porta paralel olarak denenecektir.

SYN bayrağı, uzak sisteme bir bağlantı kurmaya çalıştığını gösterir. Normalde hedef port kapalı olacaktır ve bir RST (reset) paketi geri gönderilecektir. Eğer port açıksa, hedef bir SYN/ACK TCP paketi ile yanıt vererek TCP üç yönlü el sıkışmasının ikinci adımını atacaktır. Nmap çalıştırın makine daha sonra üç yönlü el sıkışmayı tamamlayacak ve tam bir bağlantı kuracak bir ACK paketi göndermek yerine bir RST ile yanıt vererek yeni oluşan bağlantıyı koparır[10].

Nmap portun açık ya da kapalı olmasına ilgilenmez. Daha önce tartışılan RST veya SYN/ACK yanıtı Nmap'e ana bilgisayarın kullanılabilir ve yanıt verebilir olduğunu söyler.

Unix kutularında, yalnızca ayrıcalıklı kullanıcı root genellikle ham TCP paketleri gönderebilir ve alabilir. Ayrıcalıksız kullanıcılar için, her hedef porta karşı connect sistem çağrısının başlatıldığı bir geçici çözüm otomatik olarak kullanılır. Bu, bir bağlantı kurma girişimi olarak hedef ana bilgisayara bir SYN paketi gönderme etkisine sahiptir. Eğer connect hızlı bir başarı veya ECONNREFUSED hatası ile

dönerse, temel TCP yiğini bir SYN/ACK veya RST almış olmalıdır ve ana bilgisayar kullanılabılır olarak işaretlenir. Bağlantı girişimi bir zaman aşımına ulaşılana kadar askıda kalırsa, ana bilgisayar kapalı olarak işaretlenir. Bu geçici çözüm IPv6 bağlantıları için de kullanılır, çünkü ham IPv6 paket oluşturma desteği henüz Nmap'te mevcut değildir.

Örnek 3.8, ICMP yanıt isteklerine yanıt vermedikleri için altı makineden dördünü tespit edememiştir. Örnek 3.9'da gösterildiği gibi, 80 numaralı bağlantı noktasına (HTTP) bir SYN sondası kullanılarak deney tekrarlandığında altı makineden de yanıt alınmıştır.

Örnek 3.9. Port 80 SYN problemini kullanarak ana bilgisayar bulmayı yeniden deneyin

```
# nmap -sn -PS80 -R -v microsoft.com ebay.com citibank.com google.com \
slashdot.org yahoo.com

Starting Nmap ( https://nmap.org )
Host origin2.microsoft.com (207.46.249.252) appears to be up.
Host pages.ebay.com (66.135.192.87) appears to be up.
Host ldl-www.citicorp.com (192.193.195.132) appears to be up.
Host 216.239.57.99 appears to be up.
Host slashdot.org (66.35.250.150) appears to be up.
Host w3.rc.dcn.yahoo.com (216.109.127.30) appears to be up.
Nmap done: 6 IP addresses (6 hosts up) scanned in 0.48 seconds
```

Altı makinenin tamamını tespit etmenin yanı sıra, ikinci çalışma çok daha hızlıdır. Makineler paralel olarak tarandığı ve tarama yanıt beklerken asla zaman aşımına uğramadığı için yarımsanlısanlıya da kısır sürüyor. Bu test tamamen adil değildir çünkü bunların hepsi popüler web sunucularıdır ve bu nedenle TCP port 80'i dinlemeleri beklenebilir. Bununla birlikte, yine de farklı türdeki ana bilgisayarların farklı prob türlerine yanıt verdiği noktasını göstermektedir. Nmap, çeşitli ağların etkili bir şekilde taranmasını sağlamak için birçok tarama türünün paralel olarak kullanılmasını destekler.

### TCP ACK Ping ( **-PA <port list>** )

TCP ACK pingi SYN pingine oldukça benzer. Aradaki fark, tahmin edebileceğiniz gibi, SYN bayrağı yerine TCP ACK bayrağının ayarlanmış olmasıdır. Böyle bir ACK paketi, kurulmuş bir TCP bağlantısı üzerinden veriyi onayladığı iddia eder, ancak

böyle bir bağlantı yoktur. Bu nedenle, uzak ana bilgisayarlar her zaman bir RST paketi ile yanıt vermeli ve bu süreçte varlıklarını ifşa etmelidir.

PA seçeneği SYN probu (80) ile aynı varsayılan portu kullanır ve aynı formatta bir hedef port listesi de alabilir. Ayrıcalıksız bir kullanıcı bunu denerse veya bir IPv6 hedefi belirtilirse, daha önce tartışılan connect geçici çözümü kullanılır. Bu geçici çözüm kusurludur çünkü connect aslında bir ACK yerine bir SYN paketi göndermektedir.

Hem SYN hem de ACK ping problemleri sunmanın nedeni, güvenlik duvarlarını aşma şansını en üst düzeye çıkarmaktır. Birçok yönetici yönlendiricileri ve diğer basit güvenlik duvarlarını, şirket web sitesi veya posta sunucusu gibi genel hizmetlere yönelik olanlar dışında gelen SYN paketlerini engelleyecek şekilde yapılandırır. Bu, kuruluşa gelen diğer bağlantıları engellerken, kullanıcıların Internet'e engelsiz giden bağlantılar yapmasına izin verir. Bu durum bilgisi içermeyen yaklaşım güvenlik duvarı/yönlendirici üzerinde çok az kaynak kullanır ve donanım ve yazılım filtreleri tarafından yaygın olarak desteklenir. Bu yöntemin yaygınlığına sadece bir örnek olarak, Linux Netfilter/iptables güvenlik duvarı yazılımı, man sayfasında aşağıdaki şekilde açıklanan --syn kolaylık seçeneğini sunar.

Yalnızca SYN biti ayarlanmış ve ACK ve RST bitleri temizlenmiş TCP paketleriyle eşleşir. Bu tür paketler TCP bağlantısının başlatılmasını istemek için kullanılır; örneğin, bir arayüze gelen bu tür paketlerin engellenmesi, gelen TCP bağlantılarını önleyecektir, ancak giden TCP bağlantıları etkilenmeyecektir. Bu --tcp-flags SYN,RST,ACK SYN ile eşdeğerdir.

Bunun gibi güvenlik duvari kuralları mevcut olduğunda, SYN ping problemleri (-PS) kapalı hedef portlara gönderildiğinde muhtemelen engellenecektir. Bu gibi durumlarda, ACK probu bu kuralları doğrudan aşarak üstünlük sağlar.

Bir başka yaygın güvenlik duvari türü de beklenmedik paketleri düşüren durum bilgisi kurallarını kullanır. Bu özellik başlangıçta çoğunlukla üst düzey güvenlik duvarlarında bulunmaktaydı, ancak yıllar içinde çok daha yaygın hale geldi. Linux Netfilter/iptables sistemi, aşağıdaki man sayfası alıntısında açıklanıldığı gibi paketleri bağlantı durumuna göre kategorize eden --state seçeneği aracılığıyla bunu destekler:

Olası durumlar, paketin bilinen bir bağlantıyla ilişkili olmadığı anlamına gelen GEÇERSİZ, paketin her iki yönde de paketleri görmüş bir bağlantıyla ilişkili olduğu anlamına gelen KURULMUŞ, paketin yeni bir bağlantı başlattığı veya her iki yönde

de paketleri görmemiş bir bağlantıyla ilişkili olduğu anlamına gelen YENİ ve paketin yeni bir bağlantı başlattığı, ancak bir FTP veri aktarımı veya ICMP hatası gibi mevcut bir bağlantıyla ilişkili olduğu anlamına gelen İLGİLİ'dir.

ACK probunun bu yaklaşımı benimseyen güvenlik duvarlarına karşı çalışması pek olası değildir, çünkü böyle beklenmedik bir paket INVALID durumunda sınıflandırılacak ve muhtemelen düşürülecektir. Örnek 3.10'da Microsoft'a karşı bir ACK ping denemesi gösterilmektedir. Durum bilgisi içeren güvenlik duvarı paketi düşürür ve Nmap'in yanlışlıkla ana bilgisayarın kapalı olduğu sonucuna varmasına neden olur. SYN probunun bu gibi durumlarda çalışma şansı çok daha yüksektir. Bu, hedef ağların güvenlik duvarı kuralları bilinmediğinde veya tutarsız olduğunda hangi tekniğin kullanılacağı sorusunu gündeme getirir. Doğru cevap genellikle her ikisidir. Nmap paralel olarak birçok porta SYN ve ACK problemleri gönderebilir ve aynı zamanda diğer host keşif tekniklerini de uygulayabilir. Bu konu "Hepsini Bir Araya Getirmek" başlıklı bölümde daha ayrıntılı olarak ele alınmaktadır: Ev Sahibi Bulma Stratejileri bölümünde ele alınmaktadır.

Örnek 3.10. Microsoft'a karşı ACK ping denemesi

```
# nmap -sn -PA www.microsoft.com
Starting Nmap ( https://nmap.org )
Warning: Hostname www.microsoft.com resolves to 5 IPs. Using 207.46.192.254.
Note: Host seems down. If it is really up, but blocking ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 2.22 seconds
```

### UDP Ping ( **-PU <port list>** )

Bir başka ana bilgisayar bulma seçeneği de verilen portlara UDP paketi gönderen UDP ping'dir. Bağlantı noktası listesi, daha önce tartışılan -PS ve -PA seçenekleriyle aynı biçimde alır. Herhangi bir port belirtilmemezse, varsayılan 40,125'tir. Bu varsayılan, nmap.h dosyasında DEFAULT\_UDP\_PROBE\_PORT\_SPEC değiştirilerek derleme zamanında yapılandırılabilir. Varsayılan olarak çok yaygın olmayan bir bağlantı noktası kullanılır, çünkü açık bağlantı noktalarına göndermek bu özel tarama türü için genellikle istenmeyen bir durumdur.

Çoğu port için paket boş olacaktır, ancak 53 ve 161 gibi birkaç yaygın port için yanıt alma olasılığı daha yüksek olan protokole özgü bir yük gönderilecektir. --data-length seçeneği tüm portlar için sabit uzunlukta rastgele bir yük gönderir.

Hedef makinede kapalı bir bağlantı noktasına ulaşıldığında, UDP probu karşılığında bir ICMP bağlantı noktası ulaşılamaz paketi ortaya çıkarmalıdır. Bu, Nmap'e makinenin açık ve kullanılabilir olduğunu gösterir. Ana bilgisayar/ağ ulaşılamıyor veya TTL aşındı gibi diğer birçok ICMP hatası türü, kapalı veya ulaşılamayan bir ana bilgisayarın göstergesidir. Yanıt alınamaması da bu şekilde yorumlanır. Açık bir bağlantı noktasına ulaşılırsa, çoğu hizmet boş paketi yok sayar ve herhangi bir yanıt döndürmez. Bu nedenle varsayılan prob portu 40,125'tir ve bu portun kullanımda olma ihtimali oldukça düşüktür. Karakter Oluşturucu (chargen) protokolü gibi birkaç hizmet boş bir UDP paketine yanıt verecek ve böylece Nmap'e makinenin kullanılabilir olduğunu açıklayacaktır. Özel yükler, bunlara sahip portlar için, bir probun yanıt alma olasılığını daha yüksek hale getirir.

Bu tarama türünün birincil avantajı, yalnızca TCP'yi tarayan güvenlik duvarlarını ve filtreleri atlamasıdır. Örneğin, bir zamanlar Linksys BEFW11S4 kablosuz geniş bant yönlendiricim vardı. Bu cihazın harici arayüzü varsayılan olarak tüm TCP portlarını filtreliyordu, ancak UDP problemleri yine de porta ulaşılamıyor mesajlarını ortaya çıkarıyordu ve böylece cihazı ele veriyordu.

### **ICMP Ping Types ( -PE , -PP , and -PM ) (ICMP Ping Türleri (-PE, -PP ve -PM))**

Daha önce tartışılan alışılmadık TCP ve UDP ana bilgisayar keşif türlerine ek olarak, Nmap her yerde bulunan ping programı tarafından gönderilen standart paketleri gönderebilir. Nmap hedef IP adreslerine bir ICMP tip 8 (yankı isteği) paketi gönderir ve karşılığında mevcut ana bilgisayarlardan bir tip 0 (yankı yanıtı) bekler. Bu bölümün başında belirtildiği gibi, birçok ana bilgisayar ve güvenlik duvarı artık RFC 1122'nin gerektirdiği şekilde yanıt vermek yerine bu paketleri engellemektedir. Bu nedenle, yalnızca ICMP taramaları İnternet üzerinden bilinmeyen hedeflere karşı nadiren yeterince güvenilirdir. Ancak dahili bir ağı izleyen sistem yöneticileri için bu pratik ve verimli bir yaklaşım olabilir. Bu yanıt isteği davranışını etkinleştirmek için -PE seçeneğini kullanın.

Yanıt isteği standart ICMP ping sorgusu olsa da, Nmap bununla yetinmez. ICMP standartları (RFC 792 ve RFC 950) ayrıca zaman damgası isteği, bilgi isteği ve adres maskesi isteği paketlerini sırasıyla 13, 15 ve 17 kodları olarak belirtir. Bu sorguların görünürdeki amacı adres maskeleri ve geçerli saatler gibi bilgileri öğrenmek olsa da, ana bilgisayar keşfi için kolayca kullanılabilirler. Yaygın olarak desteklenmedikleri için Nmap şu anda bilgi istek paketlerini uygulamamaktadır (RFC 1122 "bir ana bilgisayarın bu mesajları uygulamaması gereği" konusunda

ısrar etmektedir). Zaman damgası ve adres maskesi sorguları sırasıyla -PP ve -PM seçenekleriyle gönderilebilir. Bir zaman damgası yanıtı (ICMP kodu 14) veya adres maskesi yanıtı (kod 18) ana bilgisayarın kullanılabilir olduğunu açıklar. Bu iki soru, yöneticiler özellikle yanıt isteği paketlerini engellediğinde değerli olabilir, ancak diğer ICMP sorgularının aynı amaçla kullanılabileceğini unutmayın.

### **IP Protocol Ping ( `-PO <protocol list>` )**

En yeni ana bilgisayar bulma seçeneği, IP başlıklarında belirtilen protokol numarasıyla IP paketleri gönderen IP protokolü ping'dir. Protokol listesi, daha önce tartışılan TCP ve UDP ana bilgisayar bulma seçeneklerindeki bağlantı noktası listeleriyle aynı biçimde alır. Hiçbir protokol belirtilmezse, varsayılan olarak ICMP (protokol 1), IGMP (protokol 2) ve IP-in-IP (protokol 4) için birden fazla IP paketi gönderilir. Varsayılan protokoller, nmap.h dosyasında DEFAULT\_PROTO\_PROBE\_PORT\_SPEC değiştirilerek derleme zamanında yapılandırılabilir. ICMP, IGMP, TCP (protokol 6) ve UDP (protokol 17) için paketlerin uygun protokol başlıklarıyla gönderildiğini, diğer protokollerin ise IP başlığının ötesinde hiçbir ek veri olmadan gönderildiğini unutmayın (--data-length seçeneği belirtilmemişti sürece).

### **ARP Scan ( `-PR` )**

En yaygın Nmap kullanım senaryolarından biri bir ethernet LAN'ını taramaktır. Çoğu LAN'da, özellikle RFC 1918 tarafından verilen özel adres aralıklarını kullananlarda, IP adreslerinin büyük çoğunluğu herhangi bir zamanda kullanılmaz. Nmap, ICMP yanıt isteği gibi ham bir IP paketi göndermeye çalıştığında, işletim sistemi hedef IP'ye karşılık gelen hedef donanım (ARP) adresini belirlemelidir, böylece ethernet çerçevesini düzgün bir şekilde adresleyebilir. Bu da bir dizi ARP isteği göndermesini gerektirir. Bu, yerel bir ethernet ana bilgisayarına karşı bir ping taramasının denendiği Örnek 3.11'de gösterilmektedir. --send-ip seçeneği Nmap'e yerel bir ağ olmasına rağmen IP seviyesinde paketler (ham ethernet yerine) göndermesini söyler. Üç ARP isteğinin Wireshark çıktısı ve zamanlamaları oturuma yapıştırılmıştır.

Örnek 3.11. Çevrimdışı bir hedefin ham IP ping taraması

```
# nmap -n -sn --send-ip 192.168.33.37
Starting Nmap ( https://nmap.org )
0.000000 00:01:29:f5:27:f2 -> ff:ff:ff:ff:ff:ff ARP Who has 192.168.33.37?
0.0999836 00:01:29:f5:27:f2 -> ff:ff:ff:ff:ff:ff ARP Who has 192.168.33.37?
1.999684 00:01:29:f5:27:f2 -> ff:ff:ff:ff:ff:ff ARP Who has 192.168.33.37?
Note: Host seems down. If it is really up, but blocking ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 2.04 seconds
```

Bu örneğin tamamlanması iki saniyeden fazla sürmüşür çünkü (Linux) işletim sistemi ana bilgisayardan vazgeçmeden önce birer saniye arayla üç ARP isteği gönderilmiştir. ARP yanıtlarının genellikle birkaç milisaniye içinde geldiği düşünüldüğünde, birkaç saniyelik beklemeler aşırıdır. Bu zaman aşımı süresini azaltmak işletim sistemi satıcıları için öncelikli değildir çünkü paketlerin büyük çoğunluğu gerçekten var olan ana bilgisayarlara gönderilir. Öte yandan Nmap, 10.0.0.0/8 gibi bir hedef verildiğinde 16 milyon IP'ye paket göndermelidir. Her biri için iki saniye beklemek, birçok hedefe paralel olarak ping atılsa bile büyük bir gecikme haline gelir.

LAN'larda ham IP ping taramalarıyla ilgili başka bir sorun daha vardır. Önceki örnekte olduğu gibi bir hedef ana bilgisayar yanıt vermiyorsa, kaynak ana bilgisayar genellikle çekirdek ARP tablosuna bu hedef IP için eksik bir girdi ekler. ARP tablosu alanı sınırlıdır ve bazı işletim sistemleri dolduğunda kötü tepki verir. Nmap ham IP modunda (--send-ip) kullanıldığında, Nmap bazen ana bilgisayar bulmaya devam etmeden önce ARP önbellek girdilerinin süresinin dolması için birkaç dakika beklemek zorunda kalır.

ARP taraması, kontrolü Nmap'e vererek her iki sorunu da çözer. Nmap ham ARP isteklerini yayınlar ve yeniden iletim ve zaman aşımı sürelerini kendi takdirine göre işler. Sistem ARP önbelleği atlanır. Örnek 3.12 farklı göstermektedir. Bu ARP taraması IP eşdeğerinin onda birinden biraz daha fazla zaman almaktadır.

Örnek 3.12. Çevrimdışı bir hedefin ARP ping taraması

```
# nmap -n -sn -PR --packet-trace --send-eth 192.168.33.37
Starting Nmap ( https://nmap.org )
SENT (0.0060s) ARP who-has 192.168.33.37 tell 192.168.0.100
SENT (0.1180s) ARP who-has 192.168.33.37 tell 192.168.0.100
Note: Host seems down. If it is really up, but blocking ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.23 seconds
```

Örnek 3.12'de -PR veya --send-eth seçeneklerinin hiçbir etkisi yoktur. Bunun nedeni, Nmap'in yerel bir ethernet ağında olduğunu tespit ettiği ethernet ana bilgisayarlarını tararken ARP'nin varsayılan tarama türü olmasıdır. Bu, geleneksel kablolu ethernetin yanı sıra 802.11 kablosuz ağları da içerir. ARP taraması yukarıda bahsedildiği gibi daha verimli olmakla kalmaz, aynı zamanda daha doğrudur. Ana bilgisayarlar sıkılıkla IP tabanlı ping paketlerini engeller, ancak genellikle ARP isteklerini veya yanıtlarını engellemeyecektir ve ağ üzerinde iletişim kurmaya devam ederler. Farklı ping türleri (-PE veya -PS gibi) belirtilese bile, Nmap aynı LAN üzerindeki hedefler için bunun yerine ARP kullanır. Eğer kesinlikle bir ARP taraması yapmak istemiyorsanız, Örnek 3.11, "Çevrimdışı bir hedefin ham IP ping taraması"nda gösterildiği gibi --send-ip belirtin.

Nmap'e ham ethernet çerçeveleri gönderme kontrolü vermek, Nmap'in kaynak MAC adresini kontrol etmesini de sağlar. Bir güvenlik konferansında odadaki tek PowerBook'a sahipseniz ve Apple'a kayıtlı bir MAC adresinden büyük bir ARP taraması başlatılırsa, kafalar size doğru dönebilir. "MAC Adresi Sahteciliği" bölümünde anlatıldığı gibi --spoof-mac seçeneği ile MAC adresinizi taklit edebilirsiniz.

### **Default Combination (Varsayılan Kombinasyon)**

Bu ana bilgisayar bulma tekniklerinden hiçbirini seçilmemezse, Nmap, Windows veya ayrıcalıklı (root) Unix kullanıcıları için -PE -PS443 -PA80 -PP argümanlarına eşdeğer bir varsayılan kullanır. Dikkatli okuyucular bunun her makineye bir ICMP yankı isteği, bir TCP SYN paketi, bir TCP ACK paketi ve bir ICMP zaman damgası isteği gönderdiği anlamına geldiğini bilirler. Bunun bir istisnası, yerel bir ethernet ağındaki herhangi bir hedef için bir ARP taramasının kullanılmasıdır. Ayrıcalıksız Unix kabuk kullanıcıları için varsayılan değer -PS80,443'e eşdeğерdir (hedef ana bilgisayarların 80 ve 443 numaralı portlarına karşı bir TCP bağlantı çağrısı). Güvenlik denetimi için, "İdeal prob kombinasyonlarının tasarlanması" başlıklı bölümde tartışılanlar gibi daha kapsamlı bir ping türü seti kullanmanızı öneririm.

[10] RST paketi, Nmap'in kendisi tarafından değil, beklenmedik SYN/ACK'ye yanıt olarak Nmap'i çalıştırın makinenin çekirdeği tarafından gönderilir.

## **Putting It All Together: Host Discovery Strategies (Hepsini Bir Araya Getirme: Ana Bilgisayar Bulma Stratejileri )**

## Related Options (İlgili Seçenekler)

Önceki bölümlerde Nmap ana bilgisayar keşif aşamasını kontrol etmek ve kullanılan teknikleri özelleştirmek için kullanılan başlıca seçenekler açıklanmaktadır. Bununla birlikte, burada ilgili olan daha birçok genel Nmap seçeneği vardır. Bu bölüm, bu seçenek bayraklarının ping taramasıyla nasıl ilişkili olduğuna dair kısa bir açıklama sunmaktadır. Her seçeneğin tam açıklamaları için Bölüm 15, Nmap Başvuru Kılavuzu'na bakın.

`-v` (same as `--verbose`)

Varsayılan olarak, Nmap genellikle yalnızca etkin, yanıt veren ana bilgisayarları yazdırır. Verbose modu, Nmap'in ana bilgisayarların yanı sıra aktif olanlar hakkında ekstra bilgi yazdırmasına neden olur.

`--source-port <portnum>` (same as `-g`)

Sabit bir kaynak bağlantı noktası ayarlamak, diğer Nmap özelliklerinde olduğu gibi ping taraması (TCP ve UDP) için de çalışır. Bazı saf güvenlik duvarı yöneticileri DNS (port 53) ya da FTP-DATA (port 20)'nın çalışmasını sağlamak için bir kural seti istisnası yaparlar. Elbette bu bir Nmap ping taraması için yeterince büyük bir delik açar. "Kaynak Bağlantı Noktası Manipülasyonu" adlı bölümde bu teknik hakkında daha fazla ayrıntı verilmektedir.

`-n`, `-R`

`n` seçeneği tüm DNS çözümlemesini devre dışı bırakırken, `-R` seçeneği kapalı olanlar da dahil olmak üzere tüm ana bilgisayarlar için DNS sorgularını etkinleştirir. Varsayılan davranış DNS çözümlemesini etkin ana bilgisayarlarla sınırlamaktır. Bu seçenekler özellikle ping taraması için önemlidir çünkü DNS çözünürlüğü tarama sürelerini büyük ölçüde etkileyebilir.

`--dns-servers <server1> [, <server2> [...]]` (Ters DNS sorguları için kullanılacak sunucular)

Nmap varsayılan olarak DNS sunucularınızı (rDNS çözümlemesi için) resolv.conf dosyanızdan (Unix) veya Kayıt Defterinden (Win32) belirlemeye çalışacaktır. Alternatif olarak, alternatif sunucuları belirtmek için bu seçeneği kullanabilirsiniz. Eğer `--system-dns` veya IPv6 taraması kullanıyorsanız bu seçenek geçerli değildir. Birden fazla DNS sunucusu kullanmak genellikle tek bir sunucuya sorulamaktan daha hızlı ve daha gizlidir. En iyi performans genellikle hedef IP alanı için tüm yetkili sunucuların belirtilmesiyle elde edilir.

--data-length <length>

Bu seçenek her pakete <uzunluk> rastgele veri baytları ekler ve TCP, UDP ve ICMP ping tarama türleriyle çalışır (IPv4 taraması yapan ayrıcalıklı kullanıcılar için). Bu, taramanın daha az dikkat çekici olmasına ve her yerde bulunan ping tanılama programı tarafından oluşturulan paketlere daha çok benzemesine yardımcı olur. Snort da dahil olmak üzere birçok saldırısı tespit sistemi (IDS) sıfır baytlık ping paketleri için uyarı verir. Bu seçenek bu uyarılardan kaçınır. 32'lik bir seçenek değeri, bir yanıt isteğinin Windows'tan gelmiş gibi görünmesini sağlarken, 56 varsayılan Linux ping'ini simüle eder.

--ttl <value>

Giden TTL'nin ayarlanması IPv4 ping taraması yapan ayrıcalıklı kullanıcılar için desteklenir. Bu, bir taramanın yerel ağını ötesine yayılmamasını sağlamak için bir güvenlik önlemi olarak yararlı olabilir. Ayrıca yerel bir ping programını daha da inandırıcı bir şekilde simüle etmek için de kullanılabilir. Bazı kurumsal ağlar, kolayca düzeltmeyecekleri bilinen yönlendirme döngülerinden muzdariptir. Giden TTL'yi -ttl ile azaltmak, döngülerle karşılaşıldığında yönlendirici CPU yükünü azaltmaya yardımcı olur.

Konserve zamanlama seçenekleri (-T3, -T4, -T5, vb.)

Daha yüksek -T değerleri, diğer Nmap özelliklerini hızlandırdığı gibi ping taramasını da hızlandırır. Kaynak ve hedef ağlar arasında orta derecede hızlı ve güvenilir bir bağlantı ile (yani çevirmeli modemden daha fazlası), -T4 seçeneği önerilir.

--max-parallelism , --min-parallelism <value>

Bunlar, aynı anda kaç probun olağanüstü olabileceğini etkiler. Varsayılan ping türü (iki prob) ile paralellik değeri kabaca paralel olarak taranan makine sayısıdır. Ping tekniklerini ana bilgisayar başına bir sondaya düşürmek (örneğin -PE), belirli bir paralellik seviyesi için aynı anda taranan ana bilgisayar sayısını iki katına çıkarırken, ana bilgisayar başına dört sondaya çıkarmak (örneğin -PE -PS22,113,50000) bunu yarıya indirir. Çoğu kullanıcı -T4 gibi hazır zamanlama seçeneklerine bağlı kalır.

--min-rtt-timeout , --max-rtt-timeout , --initial-rtt-timeout <time>

Bu seçenekler Nmap'in ping yanıtı için ne kadar bekleyeceğini kontrol eder.

Girdi seçenekleri (-iL <dosya adı>, -iR <sayı>)

Ana bilgisayar giriş seçenekleri Nmap'in geri kalanında olduğu gibi desteklenir. Kullanıcılar genellikle -Pn ile input-from-list (-iL) seçeneğini birleştirerek zaten açık olduğu bilinen ana bilgisayarları ping ile taramaktan kaçınırlar. Zaman kazanmak amacıyla bunu yapmadan önce "Ping'i Devre Dışı Bırak (-Pn)" başlıklı bölüm okuyun. iR seçeneği, tahsis edilen İnternet IP alanından rastgele ana bilgisayarlar seçer. Taramak istediğiniz rastgele ana bilgisayar sayısını argüman olarak alır. Hiç bitmeyen (siz Nmap işlemini iptal edene veya öldürene kadar) bir tarama için sıfır kullanın.

Çıktı seçenekleri (-oA, -oN, -oG, -oX, vb.)

Tüm Nmap çıktı türleri (normal, grepable ve XML) ping taramasını destekler. Bölüm 13, Nmap Çıktı Biçimleri bunların nasıl çalıştığını daha ayrıntılı olarak açıklamaktadır.

--randomize-hosts

Bu seçenekle ana bilgisayar tarama sırasını karıştırmak taramayı daha az dikkat çekici hale getirebilir, ancak aynı zamanda tarama çıktısını takip etmeyi biraz daha zorlaştırabilir.

--reason

Normal Nmap çıktısı bir ana bilgisayarın açık olup olmadığını gösterir, ancak ana bilgisayarın hangi keşif test(ler)ine yanıt verdiği açıklamaz. Bu ayrıntı için --reason seçeneğini ekleyin. Nmap her zaman her probu denemediği için sonuçlar konak keşfi için kafa karıştırıcı olabilir. İlk yanıtı alır almaz durur. Bu nedenle Nmap, çalışma sırasında bir ana bilgisayardan bir ICMP echo yanıtı rapor edebilir, ancak daha sonra ikinci bir çalışma sırasında önce bir RST yanıtı alınabilir ve Nmap'in bunu rapor etmesine neden olabilir.

--packet-trace

Reason'ın sağladığından daha fazla ayrıntı istediginizde --packet-trace'i deneyin. Bu seçenek, sıra numaraları, TTL değerleri ve TCP bayrakları gibi ayrıntılar da dahil olmak üzere Nmap tarafından gönderilen ve alınan her paketi gösterir.

-D <decoy1,decoy2,...>

Tuzaklar, gerçek saldırganı kamufla eden ayrıcalıklı IPv4 ping taramaları için tamamen desteklenir.

-6

TCP bağlantı tabanlı ping taramaları (-PS), çoklu bağlantı noktası modu da dahil olmak üzere IPv6 protokolünü destekler (-PS22,80,113 gibi).

`-S <source IP address>, -e <sending device name>`

Nmap'in diğer fonksiyonlarında olduğu gibi, kaynak adresi ve gönderen cihaz bu seçeneklerle belirtilebilir.

#### Genel seçenekler

Varsayılan olarak, -sn veya -sL belirtilmediği sürece, Nmap ana bilgisayar keşif aşamasından sonra daha müdahaleci taramaya geçer. Böylece dzinelerce genel bağlantı noktası taraması, işletim sistemi algılama ve sürüm algılama seçenekleri kullanılabilir. Daha fazla bilgi için referans kılavuzuna veya ilgili bölmelere bakın.

### **Choosing and Combining Ping Options (Ping Seçeneklerini Seçme ve Birleştirme)**

Etkili tarama, bu ve önceki bölümlerde açıklanan tüm seçenekleri bilmekten daha fazlasını gerektirir. Kullanıcılar, hedef ağ topolojisine ve tarama hedeflerine uyacak şekilde bunları nasıl ve ne zaman kullanacaklarını anlamalıdır.

#### **Most valuable probes (En değerli probalar)**

Mayıs 2009'da en etkili ping probalarını bulmak için bazı araştırmalar yaptık. Binlerce Internet ana bilgisayara karşı dzinelerce farklı sonda ve ana bilgisayar bulma seçeneğini test ettik. Bu araştırmanın sonuçları Tablo 3.2'de gösterilmekte ve farklı probalar etkinliklerine göre sıralanmaktadır. Yüzde, probardan herhangi birine yanıt veren ana bilgisayar sayısı üzerinden verilmiştir. 90 farklı proba test ettik; bu bir alt kümedir.

Tablo 3.2. En iyi ana bilgisayar keşif probaları

Hosts found	Probe
62.47%	<code>-PE</code>
44.17%	<code>-PS443</code>
43.28%	<code>-PA80</code>
43.01%	<code>-PA443</code>
42.47%	<code>-PS80</code>
40.65%	<code>-PA110</code>

40.42%	-PA3389
40.41%	-PS110
39.89%	-PA22
39.62%	-PS21
39.62%	-PA21
38.75%	-PS22
37.50%	-PS3389
36.66%	-PP
31.17%	-PU40125 --source-port 53 --data-length 24
29.96%	-PU31338 --source-port 53 --data-length 24
29.05%	-PU631 --source-port 53 --data-length 24
26.38%	-PU40125
26.09%	-PS25
25.69%	-PA25
25.35%	-PU31338
24.71%	-PU631
24.15%	-PU53 --source-port 53 --data-length 24
22.20%	-PU53
9.09%	-PO2
9.03%	-PO150
7.20%	-PO4
4.21%	-PM

Tablo 3.2'den bazı ilginç özellikler seçebiliriz. Açık ara en iyi tek prob -PE, ICMP echo'dur. Bunu, ortak bağlantı noktalarına çeşitli SYN ve ACK problemleri (-PS ve -PA) takip eder. ICMP zaman damgası, -PP, oldukça iyi sonuç verir. En iyi UDP problemleri (-PU) rastgele yüksek portlara yapılanlardır ve --source-port 53 --data-length 24 ile birleştirildiklerinde daha etkilidirler. Port 25'e (SMTP) SYN ve ACK, diğer TCP problemlerinden belirgin şekilde daha kötüdür, belki de bu port genelliklefiltrelendiği içindir. Çeşitli IP protokol pingleri (-PO) çok etkili değildir ve ICMP adres maskesi isteği olan -PM son sırada yer alır.

En iyi çoklu prob kombinasyonları sadece Tablo 3.2'nin üst kısmındaki girdileri seçerek bulunmaz. Bunun nedeni bazı problemlerin diğeriyile aynı ana bilgisayarların çoğunu bulmasıdır, bu nedenle ikisini birden kullanmak sadece birini kullanmaktan biraz daha iyidir. Bir bilgisayar programına, her bir probun kaç ana bilgisayar bulduğunu ve diğer problemlerle ne kadar örtüşüğünü inceleyerek farklı prob sayıları için optimum kombinasyonu buldurduk. En iyi kombinasyonlar Tablo 3.3'te yer almaktadır. Nmap'in varsayılan dört prob kombinasyonu, kapsamlılık ve hız arasında bir uzlaşma olarak bu listeden seçilmiştir.

Tablo 3.3. En iyi ana bilgisayar keşif probu kombinasyonları

	Hosts found	Probe combination
1 probe	62.47%	-PE
2 probes	77.61%	-PE -PA80
3 probes	83.83%	-PE -PA80 -PS443
4 probes	88.64%	-PE -PA80 -PS443 -PP
5 probes	91.12%	-PE -PA80 -PS443 -PP -PU40125 --source-port 53
6 probes	92.42%	-PE -PS80 -PS443 -PP -PU40125 -PA3389 --source-port 53
7 probes	93.10%	-PE -PS80 -PS443 -PP -PU40125 -PS3389 -PA21 --source-port 53
8 probes	93.69%	-PE -PS80 -PS443 -PP -PU40125 -PS3389 -PA21 -PU161 --source-port 53

### TCP probe and port selection (TCP probu ve port seçimi)

TCP ping seçenekleri Nmap'teki en güçlü keşif tekniklerinden bazlılarıdır. Bir yönetici, çoğu kullanıcıyı etkilemeden ICMP yanıt istek paketlerini engelleyerek kurtulabilir, ancak bir sunucu, sağladığı genel hizmetlere gönderilen SYN paketlerine kesinlikle yanıt vermelidir. Bu arada, ACK paketleri genellikle durum bilgisi olmayan güvenlik duvarlarından geçer. SYN ve ACK problemlerinin her ikisini de kullanmanızı, hedef ağlar hakkında sahip olabileceğiniz bilgilere dayanan port listelerini ve daha genel olarak popüler portları kullanmanızı tavsiye ederim. Tablo 3.4, deneysel testlere dayalı olarak yanıt verme olasılığı en yüksek olan (açık veya kapalı) bağlantı noktalarını listelemektedir ("En Popüler Bağlantı Noktaları Nelerdir?" başlıklı bölümde bakın). Varsayılan bırakma filtresine sahip ana bilgisayarlarda (ulaşılması en zor tür), bunlar yanıt alma olasılığı en yüksek olan bağlantı noktalarıdır.

Tablo 3.4. Azalan erişilebilirlik sırasına göre en değerli TCP prob portları.

Port numarası / Servis	Reasoning (Akıl yürütme)
80/http	Web sunucularının İnternet üzerindeki yaygınlığı, birçok yeni kullanıcının Web'in İnternet olduğuna inanmasına neden olmaktadır.
443/https	SSL, web sitelerinin gizli dizin bilgilerini korumaları için popüler bir yoldur.
113/auth	Kimlik doğrulama (identd) hizmeti sunucuların (genellikle posta veya IRC) kendilerine bağlı istemcilerin kullanıcı adını istemesini sağlar. Yöneticiler genellikle güvenlik duvarı kuralları sunucuların 113 numaralı bağlantı noktasına geri bağlanması engellediğinde oluşabilecek uzun zaman aşımlarından kaçınmak için bu bağlantı noktasınıfiltresiz bırakırlar. Bu bağlantı noktasını ping taraması için kullanmak bazen yanlış pozitif sonuçlara yol açabilir, çünkü bazı yöneticilerin güvenlik duvarlarını, ağlarındaki herhangi bir IP'ye yapılan kimlik doğrulama sorgularına yanıt olarak, o IP'de makine bulunmasa bile RST paketlerini geri gönderecek şekilde yapılandırdıkları bilinmektedir. Yöneticiler bunu sunucuların zaman aşımına uğramasını önlemek ve aynı zamanda portlara erişilmesini engellemek için yaparlar.
21/ftp	Birçok güvenlik duvarı yöneticiyi vefatının yasını tutmayacak olsa da bu dosya aktarım protokolü yaşamaya devam ediyor.
23/telnet	Birçok cihaz hala bu yönetim arayüzü sunuyor, ancak bu bir güvenlik kabusu.
25/smtp	Mail, şirketlerin güvenlik duvarlarından geçmesine izin verdikleri bir başka Internet "katil uygulaması".
53/domain	Alan adı sunucuları son derece yaygındır.
22/ssh	SSH, uzaktan terminal yönetimi için standart olarak nihayet Telnet'i geride bırakmış gibi görünüyor.
110/pop3	POP3, e-postaları uzaktan okumak için yaygın bir yöntemdir.
3389/ms-term-server	Microsoft Terminal Hizmetleri, kullanıcıların (ve bazen bilgisayar korsanlarının) uzaktaki bir bilgisayardaki uygulamalara ve verilere erişmesine olanak tanır.
1723/pptp	Noktadan Noktaya Tünelleme Protokolü genellikle Microsoft Windows üzerinde VPN çözümleri uygulamak için kullanılır.
8080/http-proxy	Port 8080 üzerinde bir web proxy çalıştırılmak yaygındır. Bu bağlantı noktası bazen 80 numaralı bağlantı noktası kullanılmadığında alternatif

bir HTTP bağlantı noktası olarak da kullanılır.

Yukarıdaki listedekiler gibi popüler bağlantı noktalarına ek olarak, en az bir yüksek numaralı bağlantı noktası seçilmesi önerilir. Kötü yapılandırılmış birçok güvenlik duvarı yalnızca ayrıcalıklı bağlantı noktaları, yani 1.024'ün altındakiler için varsayılan reddetme özelliğine sahiptir. Bu tür bir güvenlik duvarının arkasındaki makineleri yakalamak için genellikle 40.000 veya 10.042 gibi yüksek numaralı bir bağlantı noktası seçerim.

Taranacak portları seçerken, platform çeşitliliğini vurgulamayı unutmayın. Ping taramanızı iki portla sınırlıyorsanız, HTTP (80) ve SSH (22) muhtemelen HTTP (80) ve HTTPS'den (443) daha iyidir çünkü son ikisi ilgili web hizmetleridir ve HTTPS'ye sahip birçok makinede genellikle HTTP zaten mevcut olacaktır. Aynı makinede iki erişilebilir bağlantı noktasını bulmak, ping tarama amaçları için bir tane bulmaktan daha iyi değildir. Amaç, geniş bir ana bilgisayar kümесinin bunlardan en az biriyle eşleşeceği şekilde bağlantı noktalarını seçmektir.

Değerli bağlantı noktası tablosunun, her yerde bulunan Windows SMB bağlantı noktası 135 gibi istemci odaklı birçok bağlantı noktasını içermediğini unutmayın. Bunun temel nedeni, bu tablonun yalnızca bağlantı noktalarının büyük çoğunluğununfiltrelendiği varsayılan güvenlik duvarlarının arkasındaki ana bilgisayarlara bakmasıdır. Bu durumlarda, 135-139 ve 445 gibi Windows bağlantı noktaları genellikle engellenir. Bu makineler bir güvenlik duvarının arkasında olmadığından, açık portlar ping taraması için önemsizdir çünkü binlerce kapalı port da aynı şekilde çalışır.

### **UDP port selection (UDP bağlantı noktası seçimi)**

UDP portlarını seçerken, açık bir portun problara yanıt verme olasılığının düşük olduğunu unutmayın. Filtrelenmemiş bağlantı noktaları istenir. Açık portlardan kaçınmak için DNS (port 53) ve SNMP (161) gibi yaygın UDP hizmetlerini hariç tutmayı düşünebilirsiniz. Öte yandan, güvenlik duvari kuralları genellikle o kadar genişir ki, bu problemler (özellikle 53 numaralı bağlantı noktasına) geçebilir ve kapalı bir bağlantı noktasına çarpabilir. Bu yüzden en azından 53 numaralı portu ve 37.452 gibi keyfi olarak seçilmiş yüksek numaralı bir portu seçmenizi tavsiye ederim.

### **ICMP probe selection (ICMP prob seçimi)**

ICMP için, standart ping (yankı isteği) genellikle denemeye değerdir. Birçok yönetici, hata ayıklama için yararlı olduğu veya RFC 1122 gerektirdiği için buna özellikle izin verir. Ayrıca adres maskesi veya zaman damgası isteklerinden en az birini kullanırdım. Bunlar, yöneticilerin yankı isteği paketlerini kasıtlı olarak engellediği, ancak diğer ICMP sorgularını unuttuğu ağlar için değerlidir.

### **Designing the ideal combinations of probes (İdeal prob kombinasyonlarının tasarılanması)**

Tüm bu ping türlerinin bir ping tarama stratejisinde nasıl birleştirileceği hedef ağın özelliklerine ve tarama hedeflerine bağlıdır. Dahili ağlar için varsayılan ping türü genellikle iyi çalışır. Varsayılan, ara sıra bir ana bilgisayarı kaçırmamanın önemli olmadığı çoğu sıradan tarama için de iyidir. Daha fazla sonda eklemek, ping taramasının biraz daha uzun sürmesi pahasına, ara sıra gizli makineleri yakalamaya yardımcı olabilir. Geçen süre kabaca her makineye gönderilen prob sayısı ile orantılıdır. İnternet üzerinden hedef ağların güvenlik taramaları için, daha fazla sonda eklemek genellikle tavsiye edilir. Daha önce tartışılan tekniklerin çeşitli bir kümesini dahil etmeye çalışın. İşte ana bilgisayarların büyük çoğunluğunu yakalaması gereken bir dizi ping seçeneği: -PE -PP -

PS21,22,23,25,80,113,443,31339 -PA80,113,443,10042. Ayrıca --source-port 53'ü eklemek de faydalı olabilir. Sonuçlar ne kadar daha iyi olacak ve ne kadar uzun sürecek? Bu elbette hedef ağa bağlıdır, ancak Nmap rastgele hedef seçimi seçeneği (-iR) hızlı bir test gerçekleştirmeyi kolaylaştırır. Örnek 3.13, Nmap'in 50.000 rastgele IP adresi oluşturduğunu ve ardından varsayılan bir ping taraması gerçekleştirdiğini göstermektedir. Varsayılanın dört prob olduğunu hatırlamalısınız: bir ICMP yankı isteği, 443 numaralı bağlantı noktasına bir TCP SYN, 80 numaralı bağlantı noktasına bir TCP ACK ve bir ICMP zaman damgası isteği.

Örnek 3.13. 50.000 IP adresi oluşturma, ardından varsayılan seçeneklerle ping taraması

```

# nmap -n -sL -iR 50000 | awk '/^Host / {print $2}' | sort -n > 50K_IPs
# head -5 50K_IPs
3.0.47.195
3.100.112.251
3.100.98.39
3.10.89.120
3.101.183.79
# nmap -n -sn -T4 -iL 50K_IPs -oA 50KHosts_DefaultPing

Starting Nmap ( https://nmap.org )
Host 4.178.9.27 is up (0.27s latency).
Host 12.135.202.138 is up (0.20s latency).
Host 12.151.172.161 is up (0.089s latency).
[thousands of lines cut]
Host 222.99.77.130 is up (0.20s latency).
Nmap done: 50000 IP addresses (3927 hosts up) scanned in 2532.05 seconds

```

50.000 adresin taranması 42 dakikadan biraz fazla sürmüş ve 3.927 ana bilgisayar tespit edilmiştir. Daha geniş bir ping teknigi yelpazesı kullanmanın etkilerini belirlemek için, aynı 50 bin ana bilgisayar, varsayılan dört yerine bağlantı noktası başına 14 prob ile yeniden tarandı. Örnek 3.14'te gösterildiği gibi, Nmap 785 (%20) daha fazla ana bilgisayar tespit edebilmiştir. Bu işlem yaklaşık 147 dakika sürdü, yani neredeyse 3,5 kat daha uzun. Tespit edilen tüm yeni ana bilgisayarlar göz önüne alındığında, bu ekstra zaman iyi harcanmış. Yeni ana bilgisayarların hepsinin meşru olmayacağıını unutmayın. Ping problemlerinin sayısını artırmak, Nmap'in var olmayan bir ana bilgisayarın aktif görünmesine neden olan ağ yapıtlarına çarpmayı olasılığını artırır. SYN veya ACK paketleri için 113 numaralı bağlantı noktasına bir RST döndüren güvenlik duvarları buna bir örnektir.

Örnek 3.14. Ekstra problarla ping taramasını tekrarlama

```

# nmap -n -sn -PE -PP -PS21,22,23,25,80,113,443,31339 -PA80,113,443,10042 \
-T4 --source-port 53 -iL 50K_IPs -oA 50KHosts_ExtendedPing
Starting Nmap ( https://nmap.org )
Host 4.238.177.186 is up (0.44s latency).
Host 12.135.202.138 is up (0.13s latency).
Host 12.151.172.161 is up (0.092s latency).
[thousands of hosts cut]
Host 222.94.94.113 is up (0.23s latency).
Nmap done: 50000 IP addresses (4712 hosts up) scanned in 8842.31 seconds

```

Müşteriler için güvenlik denetimleri gerçekleştirirken, normalde TCP analizine Örnek 3.14, "Ekstra problarla ping taramasını tekrarlama" bölümünde gösterilenler gibi kapsamlı ping tarama seçenekleriyle en yaygın 1.000 porta (varsayılan) karşı

bir port taramasıyla başlarım. Böyle bir tarama çok uzun sürmüyor ve hızlı bir şekilde çalışmaya başlamamı sağlıyor. Ayrıca çalışırken arka planda tüm 65K TCP portlarına karşı -Pn (ping devre dışı) taramaları başlatıyorum. Bu taramalar bittiğinde, ki bu günler sonra da olabilir, bunları ilk hızlı taramamla karşılaştırıyorum ve bulunan yeni portları ya da makineleri inceliyorum.

## **Host Discovery Code Algorithms (Ana Bilgisayar Bulma Kodu Algoritmaları)**

Nmap gibi açık kaynak kodlu yazılımların en büyük faydalardan biri, meraklı kullanıcıların yazılımın işleyişi hakkında cevap almak istediklerinde her zaman kaynak kodunu inceleyebilmeleridir. En üst düzey ping tarama işlevi nexthost'tur ([targets.cc](#)'de, bir hedef listesini başlatmak için massping'i çağırır. Massping de listeyi ultra\_scan'e (scan\_engine.cc içinde) aktarır. Ultra\_scan Nmap'in genel amaçlı tarama fonksiyonudur ve paketleri gönderme, alma ve yorumlama gibi tüm zor işleri yapar. Ultra\_scan hakkında daha fazla bilgi için "Tarama Kodu ve Algoritmalar" bölümüne bakın.

Kaynak kodu analizi, Nmap'in işleyişini en küçük ayrıntısına kadar tam olarak anlamanın tek yolu olsa da, Nmap'i anlamak için her zaman en kolay yaklaşım değildir. Çoğu durumda, bir dizi komut satırı seçeneği verildiğinde Nmap'in davranışını keşfetmenin en etkili yolu, Nmap tarafından gönderilen ve alınan tüm paketleri yazdırın --packet-trace seçeneğini eklemektir.

Kaynak kodu ve --packet-trace seçeneği, Nmap işleminin en ince ayrıntılarını öğrenmek için mükemmel kaynaklar olduğundan, burada yalnızca ana bilgisayar keşfinin nasıl çalıştığını yüksek düzeyde tartışacağım. Nmap çalıştırıldığında, yüz binlerce hatta milyonlarca ana bilgisayar içeren ağlardan geçebilir. Bu nedenle Nmap bunları bir seferde başa çıkabilecek kadar küçük bloklara ayırır (düzinelerce ila birkaç bin ana bilgisayar). ultra\_scan daha sonra tıkanıklık kontrollerinin izin verdiği kadar hızlı paketler göndererek blok boyunca ilerler. Kullanıcı tarafından istenen tüm problemleri her bir ana bilgisayara bir kerede göndermek yerine, Nmap ilk probu tüm hedeflere gönderir, ardından ikinci probu gönderir ve bu şekilde devam eder. Bir prob için kesin bir yanıt alındığında, o ana bilgisayar uygun şekilde yukarı veya aşağı olarak işaretlenir ve ona başka prob gönderilmez. Yeniden iletimlerden

sonra bile herhangi bir sondaya yanıt vermeyen bir hedef ana bilgisayar aşağı olarak işaretlenir. Nmap, her ana bilgisayar kesin bir yanıt alana ya da zaman aşımına uğrayana kadar bekler. Sonunda, Nmap bloktaki yeni ana bilgisayarları tüketir ve yeniden iletimler tamamlandıça bekleyen prob sayısı sıfıra düşer. Ping tarama alt sistemi sonuçları döndürür, böylece Nmap port taramasına veya hedef makinelerin istenen diğer problamalarına başlayabilir. Nmap bir ana bilgisayar bloğunu tamamen bitirdiğinde, sonuçları yazdırır ve bir sonraki bloğu ping tarayıcısına aktarır.

Birden fazla ana bilgisayar, genellikle ana bilgisayar başına birden fazla prob ile paralel olarak işlenir. Bekleyen problemlerin sayısı ve zaman aşımı süreleri, ağ gecikmesi ve güvenilirliğine bağlı olarak gerçek zamanlı olarak değiştirilir. Ultra\_scan performans algoritmaları "Tarama Kodu ve Algoritmalar" bölümünde daha ayrıntılı olarak açıklanmaktadır.

## **Chapter 4. Port Scanning Overview (Bölüm 4. Port Taramaya Genel Bakış)**

### **İçindekiler**

- Introduction to Port Scanning (Port Taramaya Giriş )
  - What Exactly is a Port? (Port Tam Olarak Nedir?)
  - What Are the Most Popular Ports? (En Popüler Portlar Nelerdir?)
  - What is Port Scanning? (Port Tarama Nedir?)
  - Why Scan Ports? (Portları Neden Taramalıyız?)
- A Quick Port Scanning Tutorial (Hızlı Bir Port Tarama Eğitimi )
- Command-line Flags (Komut Satırı Bayrakları)
  - Selecting Scan Techniques (Tarama Tekniklerini Seçme )
  - Selecting Ports to Scan (Taranacak Portları Seçme)
  - Timing-related Options (Zamanlamayla İlgili Seçenekler )

- Output Format and Verbosity Options (Çıktı Formатı ve Verbosity Seçenekleri)
- Firewall and IDS Evasion Options ( Güvenlik Duvarı ve IDS Kaçınma)
- Specifying Targets (Seçenekleri Hedefleri Belirleme)
- Miscellaneous Options (Çeşitli Seçenekler )
- IPv6 Scanning ( 6 ) (IPv6 Tarama (-6))
- SOLUTION: Scan a Large Network for a Certain Open TCP Port (ÇÖZÜM: Belirli Bir Açık TCP Portu için Büyük Bir Ağ Tarama)
  - Problem (Sorun)
  - Solution (Çözüm)
  - Discussion (Tartışma)
  - See Also (Ayrıca bkz.)

## Introduction to Port Scanning (Port Taramaya Giriş )

Nmap yıllar içinde işlevsellik açısından büyümüş olsa da, etkili bir port tarayıcı olarak başladı ve bu temel işlevi olmaya devam ediyor. Basit nmap <hedef> komutu, <hedef> ana bilgisayarında en sık kullanılan 1.000 TCP bağlantı noktasını tarar ve her bağlantı noktasını açık, kapalı,filtrelenmiş,filtrelenmemiş, açık|filtrelenmiş veya kapalı|filtrelenmiş olarak sınıflandırır.

### **What Exactly is a Port? (Liman Tam Olarak Nedir?)**

Portlar, iletişim kanalları arasında ayrim yapmak için kullanılan basit bir yazılım soyutlamasıdır. IP adreslerinin ağlardaki makineleri tanımlamak için kullanılmasına benzer şekilde, bağlantı noktaları da tek bir makinede kullanımda olan belirli uygulamaları tanımlar. Örneğin, web tarayıcınız varsayılan olarak HTTP URL'lerindeki makinelerin TCP bağlantı noktası 80'e bağlanacaktır. Bunun yerine güvenli HTTPS protokolünü belirtirseniz, tarayıcı varsayılan olarak 443 numaralı bağlantı noktasını deneyecektir.

Nmap port kullanan iki protokol ile çalışır: TCP ve UDP. Her protokol için bir bağlantı dört unsurla benzersiz bir şekilde tanımlanır: kaynak ve hedef IP adresleri ve bunlara karşılık gelen kaynak ve hedef bağlantı noktaları. Tüm bu unsurlar, ana bilgisayarlar arasında gönderilen her paketin başlığına yerleştirilen basit numaralardır. Protokol, IP veri (yük) bölümünde ne tür bir paketin bulunduğuunu belirten sekiz bitlik bir alandır. Örneğin, TCP altı numaralı protokoldür ve UDP 17'dir. IPv4 adresleri 32 bit uzunluğundadır, portlar ise 16 bit uzunluğundadır. IPv6 adresleri 128 bit uzunluğundadır. Diğer IP, TCP ve UDP başlık düzeni ayrıntıları "TCP/IP Referansı" adlı bölümde bulunabilir.

Çoğu popüler hizmet iyi bilinen bir port numarasına kayıtlı olduğundan, açık portların hangi hizmetleri temsil ettiği genellikle tahmin edilebilir. Nmap, kayıtlı bağlantı noktası ve protokol numaraları için iyi bilinen hizmetin yanı sıra truva atı arka kapıları ve İnternet Atanmış Numaralar Kurumu'na (IANA) kaydolma zahmetine girmeyen diğer uygulamalar için ortak bağlantı noktalarını içeren bir nmap-services dosyası içerir. Nmap bu hizmet adını port numarasıyla birlikte referans olarak yazdırır.

Bağlantı noktası numarası alanı 16 bit genişliğinde olduğundan, değerler 65.535'e ulaşabilir. Mümkün olan en düşük değer olan sıfır geçersizdir. Programların genellikle ağ iletişimini için nasıl yazıldığını tanımlayan Berkeley sockets API, port sıfırın bu şekilde kullanılmasına izin vermez. Bunun yerine, sıfırinci port isteğini joker karakter olarak yorumlar, yani programcı hangisinin kullanıldığını önemsemez. Sistem daha sonra uygun bir bağlantı noktasını numarası seçer. Örneğin, programcılar giden bir bağlantı için hangi kaynak bağlantı noktası numarasının kullanıldığını nadiren önemserler. Bu yüzden sıfır olarak ayarlarlar ve işletim sisteminin bir tane seçmesine izin verirler.

Sıfır numaralı bağlantı noktasının geçersiz olsa da, hiçbir şey birisinin başlık alanında bunu belirtmesini engellemez. Bazı kötü niyetli trojan arka kapıları, çoğu port taramasında görünmeden gayrimeşru erişim sunmanın gizli bir yolu olarak güvenliği ihlal edilmiş sistemlerin sıfır numaralı portunu dinler. Bununla mücadele etmek için Nmap, açıkça belirtildiğinde (örn. -p0-65535) sıfır numaralı bağlantı noktasının taranmasına izin verir.

Geçerli bağlantı noktalarının ilk sınıfı olan 1'den 1.023'e kadar olan numaralar ayrılmış bağlantı noktaları olarak bilinir. Unix sistemleri (Windows'un aksine) uygulamaların bu bağlantı noktalarına bağlanabilmesi ve bu bağlantı noktalarını

dinleyebilmesi için özel (root) ayrıcalıklara sahip olmasını gerektirir. Buradaki fikir, uzaktaki kullanıcıların kötü niyetli, ayrıcalıksız bir kullanıcı tarafından değil, bir yönetici tarafından başlatılan geçerli bir hizmete bağlandıklarına güvenmelerini sağlamaktır. SSH için kayıtlı bağlantı noktası 22 yerine 2,222 olsaydı, kötü niyetli bir kullanıcı bu bağlantı noktasında sahte bir SSH arka plan programı başlatabilir ve bağlanan herkesten parola toplayabilirdi. Çoğu yaygın sunucu uygulaması ayrılmış bağlantı noktalarını dinlediğinden, bunlar genellikle taraması en verimli olanlardır.

Geçici bağlantı noktası aralığı başka bir bağlantı noktası sınıfıdır. Bu bağlantı noktası havuzu, gerektiğinde tahsis edilmek üzere sistem tarafından kullanıma sunulur. Bir uygulama port sıfır ("herhangi bir port" anlamına gelir) belirttiğinde, sistem bu aralıktan bir port seçer. Bu aralık işletim sistemine göre değişir ve genellikle yapılandırılabilir. Birçok eşzamanlı bağlantı açık olduğunda tükenmesini önlemek için en az birkaç bin bağlantı noktası içermelidir. Nmap connect taraması, her hedef makinede belirtilen her portu taradığı için bir seferde yüzlerce port kullanabilir. Linux'ta, /proc/sys/net/ipv4/ip\_local\_port\_range dosyasını kullanarak aralığı görüntüleyebilir veya ayarlayabilirsiniz. Örnek 4.1, Linux sistemimde aralığın 32,768 ila 61,000 olduğunu göstermektedir. Bu kadar geniş bir aralık neredeyse tüm durumlarda yeterli olmalıdır, ancak ben sadece nasıl yapılacağını göstermek için genişlettim.

Örnek 4.1. Linux üzerinde geçici bağlantı noktası aralığını görüntüleme ve artırma

```
felix/# cat /proc/sys/net/ipv4/ip_local_port_range
32768 61000
felix/# echo "10000 65000" > /proc/sys/net/ipv4/ip_local_port_range
felix/# cat /proc/sys/net/ipv4/ip_local_port_range
10000 65000
felix/#
```

SunRPC bağlantı noktaları genellikle geçici aralıkta bulunur. Diğer uygulamalar, bir dosya aktarımı veya başka bir olay için geçici bağlantı noktalarını geçici olarak açar. FTP istemcileri bunu genellikle aktif mod aktarımı talep ederken yapar. Bazı P2P ve anlık mesajlaşma istemcileri de bunu yapar.

IANA'nın bu kitabın dilinden biraz farklı olan kendi liman sınıflandırma şeması vardır. <http://www.iana.org/assignments/port-numbers> adresindeki yetkili bağlantı noktası listeleri, alanı aşağıdaki üç sınıfa ayırır:

İyi bilinen portlar

Bunlar, belirli bir hizmet için IANA'ya kaydedilmiş olan ayrılmış bağlantı noktalarıdır (yukarıda tartışıldığı gibi 1 ila 1.023 aralığında). Bilinen örnekler sırasıyla SSH, SMTP ve HTTP hizmetleri için 22, 25 ve 80 numaralı bağlantı noktalarıdır.

#### kayıtlı limanlar

Bu portlar 1,024 ila 49,151 aralığında yer alır ve bilinen portlarla aynı şekilde IANA'ya kaydedilmiştir. Bunların çoğu iyi bilinen portlar kadar yaygın olarak kullanılmamaktadır. Temel fark, ayrıcalıklı olmayan kullanıcıların bu portlara bağlanabilmesi ve böylece kayıtlı portlarındaki hizmetleri çalıştırabilmesidir. Kullanıcılar, ayrılmış port aralığında bulundukları için çoğu platformda iyi bilinen portlar için bunu yapamazlar.

#### dinamik ve/veya özel bağlantı noktaları

IANA, 49152'den 65535'e kadar olan bağlantı noktası numaralarını, geçici bağlantı noktaları bölümünde tartışılanlar gibi dinamik kullanıcılar için ayırır. Sadece bir şirket içinde kullanılan özel hizmetler de bu portları kullanabilir.

Bu kitap IANA'ya herhangi bir atıfta bulunmadan kayıtlı veya iyi bilinen bağlantı noktalarından bahsettiğinde, genellikle ayrılmış bağlantı noktası aralığına girip girmediklerine bakılmaksızın, nmap-services dosyasında Nmap ile kayıtlı bağlantı noktaları anlamına gelir.

### **What Are the Most Popular Ports? (En Popüler Limanlar Hangileridir?)**

Her bir port numarasının ne sıklıkla açık olduğunu belirlemek için 2008 Yazını on milyonlarca Internet ana bilgisayarını tarayarak ve işletmelerden veri toplayarak geçirdim. En yaygın hizmet portlarına aşina olmak önemlidir ve hangilerinin listeye girdiğini görmek de ilginçtir. Aşağıdaki iki liste, empirik tarama verilerimiz tarafından belirlenen en iyi TCP ve UDP bağlantı noktalarını sunmaktadır.

Listelenen hizmet nmap-services dosyamızda bulunan hizmettir. Orada her port için en yaygın hizmeti listelemeye çalışıyoruz, ancak elbette bir portun farklı şeyler için kullanılması mümkündür.

#### İlk 20 (en yaygın açık) TCP bağlantı noktası

1. Port 80 (HTTP)-Eğer bu hizmeti bilmiyorsanız, yanlış kitabı okuyorsunuz demektir. Bu, keşfettiğimiz açık portların %14'ünden fazlasını oluşturuyordu.
2. Port 23 (Telnet)-Telnet, güvensiz (şifrelenmemiş) olmasına rağmen (özellikle yönlendiriciler ve akıllı anahtarlar gibi cihazlarda bir yönetim portu olarak)

yaşamaya devam etmektedir.

3. Port 443 (HTTPS)-SSL ile şifrelenmiş web sunucuları varsayılan olarak bu portu kullanır.
4. Port 21 (FTP)-FTP, Telnet gibi, ölmesi gereken bir başka güvensiz protokoldür. Anonim FTP'de bile (kimlik doğrulama kılama endişesinden kaçınarak), veri aktarımı hala kurcalanmaya maruz kalmaktadır.
5. Port 22 (SSH)-Secure Shell, Telnet (ve bazı durumlarda FTP) için şifrelenmiş bir yedek.
6. Port 25 (SMTP)-Basit Posta Aktarım Protokolü (aynı zamanda güvensiz).
7. Port 3389 (ms-term-server)-Microsoft Terminal Hizmetleri yönetim portu.
8. Port 110 (POP3)-E-posta alımı için Postane Protokolü sürüm 3 (güvensiz).
9. Port 445 (Microsoft-DS)-MS Windows hizmetleriyle IP üzerinden SMB iletişim için (dosya/yazıcı paylaşımı gibi).
10. Port 139 (NetBIOS-SSN)- MS Windows hizmetleriyle (dosya/yazıcı paylaşımı gibi) iletişim için NetBIOS Oturum Hizmeti. Bu, Windows makinelerde 445'ten daha uzun süredir desteklenmektedir.
11. Port 143 (IMAP)-Internet Mesaj Erişim Protokolü sürüm 2. Güvensiz bir e-posta alma protokolü.
12. Port 53 (Etki Alanı)-Etki Alanı Adı Sistemi (DNS), ana bilgisayar/etki alanı adları ve IP adresleri arasında dönüşüm sağlayan güvensiz bir sistemdir.
13. Port 135 (MSRPC)-MS Windows hizmetleri için bir başka yaygın port.
14. Port 3306 (MySQL)-MySQL veritabanları ile iletişim için.
15. Port 8080 (HTTP-Proxy)-Yaygın olarak HTTP proxy'leri için veya normal web sunucuları için alternatif bir port olarak kullanılır (örneğin, başka bir sunucu zaten 80 numaralı portu dinliyorsa veya yalnızca yüksek portlara bağlanabilen ayrıcalıksız UNIX kullanıcıları tarafından çalıştırıldığında).
16. Port 1723 (PPTP)-Noktadan noktaya tünelleme protokolü (genellikle İSS'lere geniş bant bağlantıları için gerekli olan bir VPN uygulama yöntemi).
17. Port 111 (RPCBind)-SunRPC program numaralarını geçerli TCP veya UDP port numaralarıyla eşler.

18. Port 995 (POP3S)-Güvenlik için SSL eklenmiş POP3.
19. Port 993 (IMAPS)-Güvenlik için SSL eklenmiş IMAPv2.
20. Port 5900 (VNC)-Grafiksel bir masaüstü paylaşım sistemi (güvensiz).  
İlk 20 (en yaygın açık) UDP bağlantı noktası
  1. Port 631 (IPP)-İnternet Yazdırma Protokolü.
  2. Port 161 (SNMP)-Basit Ağ Yönetimi Protokolü.
  3. Port 137 (NETBIOS-NS)-Dosya ve yazıcı paylaşımı gibi Windows hizmetleri için birçok UDP portundan biri.
  4. Port 123 (NTP)-Ağ Zaman Protokolü.
  5. Port 138 (NETBIOS-DGM)-Başka bir Windows hizmeti.
  6. Port 1434 (MS-SQL-DS)-Microsoft SQL Server.
  7. Port 445 (Microsoft-DS)-Başka bir Windows Hizmetleri portu.
  8. Port 135 (MSRPC)-Başka bir Windows Hizmetleri portu.
  9. Port 67 (DHCPs)-Dinamik Ana Bilgisayar Yapılandırma Protokolü Sunucusu (ağa katıldıklarında istemcilere IP adreslerini verir).
10. Port 53 (Etki Alanı)-Domain Name System (DNS) sunucusu.
11. Port 139 (NETBIOS-SSN)-Başka bir Windows Hizmetleri portu.
12. Port 500 (ISAKMP)-İnternet Güvenlik Birliği ve Anahtar Yönetimi Protokolü IPsec VPN'leri kurmak için kullanılır.
13. Port 68 (DHCPc)-DHCP istemci portu.
14. Port 520 (Rota)-Yönlendirme Bilgi Protokolü (RIP).
15. Port 1900 (UPNP)-Microsoft Basit Hizmet Keşif Protokolü, Evrensel tak ve çalıştır cihazlarının keşfedilmesini sağlar.
16. Port 4500 (nat-t-ike)-IPsec bağlantılarını başlatırken (İnternet Anahtar Değişimi sırasında) Ağ Adresi Çevirisi geçişini müzakere etmek için.
17. Port 514 (Syslog)-Standart UNIX günlük arka plan programı.

18. Port 49152 (Değişir)-IANA tarafından belirtilen dinamik/özel portların ilki. Buradan itibaren port aralığının sonuna kadar (65536) hiçbir resmi port kaydedilemez. Bazı sistemler bu aralığı geçici bağlantı noktaları için kullanır, bu nedenle belirli bir numara talep etmeden bir bağlantı noktası bağlayan hizmetler, bunu yapan ilk programsa genellikle 49152 tassis edilir.
19. Port 162 (SNMPTrap)-Basit Ağ Yönetimi Protokolü tuzak portu (Bir SNMP aracı tipik olarak 161'i kullanırken bir SNMP yöneticisi tipik olarak 162'yi kullanır).
20. Port 69 (TFTP)-Önemsiz Dosya Aktarım Protokolü.

### **What is Port Scanning? (Port Tarama Nedir?)**

Port tarama, hangi durumda olduklarını belirlemek için çok sayıda portu uzaktan test etme eylemidir. En ilginç durum genellikle açıktır, yani bir uygulamanın bağlantı noktasını dinlediği ve bağlantıları kabul ettiği anlamına gelir. Böyle bir tarama yapmak için birçok teknik mevcuttur. Bölüm 5, Port Tarama Teknikleri ve Algoritmaları, her birinin en uygun olduğu koşulları açıklamaktadır.

Birçok port tarayıcısı geleneksel olarak tüm portları açık ya da kapalı durumlara ayıırken, Nmap çok daha ayrıntılıdır. Portları altı duruma ayırır. Bu durumlar portun kendine özgü özellikleri değildir, ancak Nmap'in onları nasıl gördüğünü açıklar. Örneğin, hedefle aynı ağdan yapılan bir Nmap taraması 135/tcp bağlantı noktasını açık olarak gösterebilirken, aynı anda İnternet üzerinden aynı seçeneklerle yapılan bir tarama bu bağlantı noktasını filtrelenmiş olarak gösterebilir.

Nmap tarafından tanınan altı bağlantı noktası durumu  
open

Bir uygulama bu bağlantı noktasında aktif olarak TCP bağlantılarını veya UDP paketlerini kabul ediyor. Bunları bulmak genellikle port taramasının birincil hedefidir. Güvenliği düşünen insanlar her açık portun bir saldırısı yolu olduğunu bilirler. Saldırganlar ve pen-testerler açık portlardan faydalananmak isterken, yöneticiler meşru kullanıcıları engellemeden bunları kapatmaya veya güvenlik duvarlarıyla korumaya çalışır. Açık portlar güvenlik dışı taramalar için de ilgi çekicidir çünkü ağ üzerinde kullanılabilecek hizmetleri gösterirler. Açık bir bağlantı noktası sizin çok heyecanlandırmadan önce, uygulamanın bir TCP sarmalayııcı (tcpd) ile korunuyor olabileceğini veya uygulamanın kendisinin yalnızca onaylı istemci IP adreslerine hizmet verecek şekilde yapılandırılmış olabileceğini

unutmayın. Bu tür durumlar yine de kapalı bir porttan daha fazla saldırısı yüzeyi bırakır.

#### closed

Kapalı bir port erişilebilirdir (Nmap prob paketlerini alır ve yanıtlar), ancak onu dinleyen bir uygulama yoktur. Bir ana bilgisayarın çevrimiçi olduğunu ve bir IP adresi kullandığını göstermede (ana bilgisayar bulma veya ping taraması) ve işletim sistemi algılamanın bir parçası olarak yardımcı olabilirler. Kapalı bağlantı noktalarına erişilebildiğinden, bazılarının açılması ihtimaline karşı daha sonra taranmaya değer olabilirler. Yöneticiler bu tür bağlantı noktalarını bir güvenlik duvarı ile engellemeyi düşünebilir, böylece daha sonra tartışılacak olan filtrelenmiş durumda görünürlər.

#### filtered

Nmap portun açık olup olmadığını belirleyemez çünkü paket filtreleme probleminin porta ulaşmasını engeller. Filtreleme özel bir güvenlik duvarı cihazından, yönlendirici kurallarından veya ana bilgisayar tabanlı güvenlik duvarı yazılımından kaynaklanıyor olabilir. Bu portlar çok az bilgi sağladıkları için saldırganları hayal kırıklığına uğratır. Bazen tip 3 kod 13 (hedefe ulaşılamıyor: iletişim idari olarak yasaklandı) gibi ICMP hata mesajlarıyla yanıt verirler, ancak yanıt vermeden problemleri bırakır filtreler çok daha yaygındır. Bu, Nmap'i, probun filtreleme yerine ağ tikanıklığı nedeniyle düşürülmeye ihtimaline karşı birkaç kez yeniden denemeye zorlar. Bu tür bir filtreleme taramaları önemli ölçüde yavaşlatır.

#### unfiltered

Filtrelenmemiş durum, bir bağlantı noktasının erişilebilir olduğu, ancak Nmap'in açık mı yoksa kapalı mı olduğunu belirleyemediği anlamına gelir. Yalnızca güvenlik duvarı kural kümelerini eşlemek için kullanılan ACK taraması bağlantı noktalarını bu duruma sınıflandırır. Filtrelenmemiş bağlantı noktalarını Pencere taraması, SYN taraması veya FIN taraması gibi diğer tarama türleriyle taramak, bağlantı noktasının açık olup olmadığını çözmeye yardımcı olabilir.

#### open|filtered

Nmap, bir portun açık mı yoksafiltrelenmemiş mi olduğunu belirleyemediğinde portları bu duruma yerleştirir. Bu, açık portların yanıt vermediği tarama türleri için gerçekleşir. Yanıtın olmaması, bir paket filtresinin probu veya ortaya çıkardığı herhangi bir yanıtı düşürdüğü anlamına da gelebilir. Bu yüzden Nmap portun açık

mi yoksa filtrelenmiş mi olduğunu kesin olarak bilemez. UDP, IP protokolü, FIN, NULL ve Xmas taramaları portları bu şekilde sınıflandırır.

closed|filtered

Bu durum, Nmap bir portun kapalı mı yoksa filtrelenmiş mi olduğunu belirleyemediğinde kullanılır. Yalnızca "TCP Idle Scan (-sl)" adlı bölümde tartışılan IP ID Idle taraması için kullanılır.

Nmap doğru sonuçlar üretmeye çalışsa da, tüm öngörülerinin hedef makineler (veya bunların önündeki güvenlik duvarları) tarafından döndürülen paketlere dayandığını unutmayın. Bu tür ana bilgisayarlar güvenilmez olabilir ve Nmap'i karıştırmak veya yaniltmak için yanıtlar gönderebilir. Çok daha yaygın olanı, Nmap probleme gerekliği gibi yanıt vermeyen RFC uyumlu olmayan ana bilgisayarlardır. FIN, NULL ve Xmas taramaları bu soruna özellikle duyarlıdır. Bu tür sorunlar belirli tarama türlerine özgüdür ve bu nedenle Bölüm 5, Bağlantı Noktası Tarama Teknikleri ve Algoritmaları'nın ilgili bölümlerinde ele alınmıştır.

### **Why Scan Ports? (Neden Portları Taramalısınız?)**

Port taraması sadece eğlence ve keyif için yapılmaz. Ağlarınızı düzenli olarak taramanın çok sayıda pratik faydası vardır. Bunların başında güvenlik gelir. Ağ güvenliğinin temel ilkelerinden biri, sunulan hizmetlerin sayısını ve karmaşıklığını azaltmanın saldırganların içeri girme fırsatını azaltmasıdır. Çoğu uzak ağ tehlikesi, bir TCP veya UDP bağlantı noktasını dinleyen bir sunucu uygulamasının istismar edilmesinden kaynaklanır. Çoğu durumda, istismar edilen uygulama hedeflenen kuruluş tarafından bile kullanılmaz, ancak makine kurulduğunda varsayılan olarak etkinleştirilmiştir. Bu hizmet devre dışı bırakılmış ya da bir güvenlik duvarı tarafından korunuyor olsaydı, saldırısı engellenmiş olurdu.

Her açık portun tehlikeye atılmak için bir fırsat olduğunu fark eden saldırganlar, hedefleri düzenli olarak tarayarak tüm açık portların bir envanterini çıkarır. Bu dinleme hizmetleri listesini, savunmasız yazılımlar için favori istismar listeleriyle karşılaştırırlar. Bir makineyi tehlikeye atmak için sadece bir eşleşme yeterlidir, bu da genellikle tüm ağı istila etmek için kullanılan bir dayanak noktası oluşturur. Kimi hedefledikleri konusunda daha az ayrımcı olan saldırganlar genellikle istismar edilebilir bir uygulamanın yalnızca varsayılan bağlantı noktasını tararlar. Bu, her portu taramaktan çok daha hızlıdır, ancak hizmet varsayılan olmayan bir portta çalışırken gözden kaçacaktır. Bu tür saldırganlar genellikle "script kiddies" olarak alay edilir, çünkü genellikle güvenlik hakkında daha yetenekli biri tarafından

yazılmış bir istismar betiğinin nasıl çalıştırılacağından daha fazlasını bilmezler. Birçok kuruluşa bu tür saldırganların savunmasız ana bilgisayarlar bulması kaçınılmazdır. Oldukça can sıkıcı olabilirler, ancak sayılarının çokluğu ve internete erişilebilen makinelere durmaksızın saldırımı genellikle insanları sistemleri hızlı bir şekilde yamalamaya iter. Bu da daha ciddi, hedefli saldırıların başarılı olma olasılığını azaltır.

Bu kırıcılarla karşı önemli bir savunma, sistem yöneticilerinin Nmap gibi araçlarla kendi ağlarını düzenli olarak taramasıdır. Açık portların listesini alın ve kullanılmayan tüm hizmetleri kapatın. Kullanılabilir kalması gerekenlerin tamamen yamalı olduğundan ve satıcının güvenlik bildirim listesinde olduğundan emin olun. Mümkün olan yerlerde güvenlik duvarı kuralları eklenmeli ve erişim yalnızca meşru kullanıcılarla sınırlanmalıdır. Çoğu popüler uygulama için Web'de sertleştirme talimatları mevcuttur, bu da kırıcının fırsatını daha da azaltır. Nmap bunların çoğunu sizin için yapamaz, ancak başlangıç için mevcut hizmetlerin listesini oluşturur. Bazı yöneticiler bunun yerine netstat kullanmaya çalışır, ancak bu iyi ölçeklenmez. Her makineye erişim gerektirir ve bazı mobil makineleri gözden kaçırmak kolaydır. Ayrıca, ortalama bir kablosuz erişim noktası, VoIP telefon veya yazıcıda netstat çalıştırılamazsınız. Buna ek olarak, ele geçirilmiş bir makinenin yanlış bilgi veren truva atlı bir netstat'a sahip olma riski her zaman vardır. Saldırganlar tarafından yüklenen modern rootkitlerin çoğu bu işlevi içerir. Yalnızca Nmap'e güvenmek de bir hatadır. Dikkatli bir tasarım, yapılandırma denetimi ve düzenli tarama kombinasyonu tavsiye edilir.

Güvenlik, port taramasının en yaygın nedeni olsa da, yöneticiler genellikle bunun başka amaçlara da uygun olduğunu görürler. Makinelerin ve sundukları hizmetlerin bir envanterini oluşturmak varlık takibi, ağ tasarımları, politika uyumluluk kontrolleri, yazılım lisansı takibi, kullanılabilirlik testi, ağ hata ayıklama ve daha fazlası için yararlı olabilir.

## A Quick Port Scanning Tutorial (Hızlı Bir Port Tarama Eğitimi )

Nmap'i geliştirirken hedeflerimden biri, özel ve gelişmiş taramalar için esnekliği korurken en yaygın kullanımı basit tutmaktadır. Bu, dzinelerce seçenek sunarak, ancak belirtildiklerinde aklı başında varsayılanlar seçerek komut satırı arayüzü

ile gerçekleştirilir. Yeni başlayan biri nmap <hedef> gibi basit bir komutla başlayabilir. Bu arada, ileri düzey kullanıcılar bazen o kadar çok seçenek belirtirler ki terminal satırları dolanır.

Benzer bir denge komut çıktısı için de sağlanmalıdır. En önemli sonuçlar, man sayfasını bile okumamış olan sıradan bir kullanıcının bile dikkatini çekmelidir. Yine de çıktı, Nmap'i her gün binlerce makineye karşı çalıştırın profesyonel sizme testçilerine uyacak kadar kapsamlı ve öz olmalıdır. Bu kitabı veya Nmap kaynak kodunu okuyacak kadar akıllı kullanıcılar, tarayıcı üzerinde daha fazla kontrol ve Nmap çıktısının gerçekte ne anlama geldiğine dair içgörülerden yararlanır.

Bu eğitimde bazı yaygın Nmap port tarama senaryoları gösterilmekte ve çıktılar açıklanmaktadır. Kapsamlı olmaya çalışmak yerine, amaç sadece yeni kullanıcıları bu bölümün geri kalanını anlayabilecek kadar iyi tanımaktır.

En basit Nmap komutu tek başına nmap'tir. Bu, yaygın Nmap seçeneklerinin ve sözdiziminin bir kopya sayfasını yazdırır. Daha ilginç bir komut, aşağıdakileri yapan nmap <hedef> komutudur:

1. <hedef>i bir ana bilgisayar adından DNS kullanarak bir IPv4 adresine dönüştürür. Ana bilgisayar adı yerine bir IP adresi belirtilirse bu arama atlanır.
2. Ana bilgisayara, varsayılan olarak bir ICMP yankı istek paketi ve 80 numaralı bağlantı noktasına bir TCP ACK paketi ile ping atarak çalışır durumda olup olmadığını belirler. Eğer değilse, Nmap bu durumu bildirir ve çıkar. Bu testi atlamak için -Pn belirtebilirdim. Bölüm 3, Ana Bilgisayar Bulma ("Ping Taraması") kısmına bakın.
3. Bir ters-DNS sorgusu kullanarak hedef IP adresini tekrar isme dönüştürür. DNS'nin çalışma şekli nedeniyle, ters isim komut satırında belirtilen <hedef> ile aynı olmayabilir. Hızı ve gizliliği artırmak için bu soru -n seçeneği ile atlanabilir.
4. nmap-services'de listelenen en popüler 1.000 portun TCP port taramasını başlatır. Genellikle bir SYN gizli taraması kullanılır, ancak ham paketler göndermek için gerekli ayrıcalıklara sahip olmayan root olmayan Unix kullanıcıları için bunun yerine bağlantı taraması kullanılır.
5. Sonuçları standart çıktıya normal insan tarafından okunabilir biçimde yazdırır ve çıkar. Bölüm 13, Nmap Çıktı Biçimleri'nde açıklanlığı gibi diğer çıktı biçimleri

ve konumları (dosyalar) belirtilebilir. Örnek 4.2, <hedef> olarak scanme.nmap.org kullanıldığında sonuçları gösterir.

Örnek 4.2. Basit tarama: nmap scanme.nmap.org

```
# nmap scanme.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 994 filtered ports
PORT      STATE    SERVICE
22/tcp    open     ssh
25/tcp    closed   smtp
53/tcp    open     domain
70/tcp    closed   gopher
80/tcp    open     http
113/tcp   closed   auth

Nmap done: 1 IP address (1 host up) scanned in 4.99 seconds
```

Örnek 4.2'deki ilk çıktı satırı basitçe Nmap'i indirmek için URL'yi verir. Nmap'in başladığı zaman ve sürüm numarası da normalde verilir, ancak bunlar tutarlılık ve satır kaydını önlemek için genellikle bu kitaptan çıkarılmıştır.

Bir sonraki satır hedef IP adresini (bu durumda IPv4) ve varsa ters DNS adını (PTR kaydı olarak da bilinir) sağlar. Nmap "ilginç portları" göstermeyi vaat eder, ancak taranan tüm portlar hesaba katılır. Açık oldukları veya o ana bilgisayar için nadiren görülen bir durumda oldukları için en ilginç olarak kabul edilen bağlantı noktaları ayrı ayrı maddelendirilir. Birçok bağlantı noktası tek bir açık olmayan durumda olduğunda, bunlar varsayılan durum olarak kabul edilir ve sonuçları binlerce ilginç olmayan girişle seyretilmekten kaçınmak için tek bir satırda toplanır. Bu durumda, Nmap 994 portun filtrelendiğini not eder.

Daha sonra ilginç bağlantı noktaları tablosu gelir ve anahtar tarama sonuçlarını sağlar. Sütunlar kullanılan seçeneklere bağlı olarak değişir, ancak bu durumda her bağlantı noktası için bağlantı noktası numarası ve protokolü, durumu ve hizmet protokolünü sağlar. Buradaki hizmet, nmap-services dosyasında bağlantı noktasına bakılarak yapılan bir tahmidir. Eğer portlardan herhangi birinin ismi bu dosyada kayıtlı değilse, servis bilinmeyen olarak listelenecaktır. Bu bağlantı noktalarından üçü açık ve üçü kapalıdır.

Son olarak, Nmap çıkmadan önce bazı temel zamanlama istatistiklerini rapor eder. Bu istatistikler, belirtilen hedeflerin sayısı, ping taramasının açık olduğunu tespit ettiği hedeflerin sayısı ve geçen toplam süredir.

Bu basit komut genellikle ihtiyaç duyulan tek şey olsa da, ileri düzey kullanıcılar genellikle çok daha ileri giderler. Örnek 4.3'te tarama dört seçenekle değiştirilmiştir. -p0- Nmap'ten mümkün olan her TCP portunu taramasını ister, -v Nmap'ten bu konuda ayrıntılı olmasını ister, -A uzak işletim sistemi tespiti, hizmet/sürüm tespiti ve Nmap Scripting Engine (NSE) gibi agresif testleri etkinleştirir. Son olarak, -T4 taramayı hızlandırmak için daha agresif bir zamanlama politikası sağlar.

Örnek 4.3. Daha karmaşık: nmap -p0- -v -A -T4 scanme.nmap.org

```

# nmap -p0- -v -A -T4 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Completed Ping Scan at 00:03, 0.01s elapsed (1 total hosts)
Scanning scanme.nmap.org (64.13.134.52) [65536 ports]
Discovered open port 22/tcp on 64.13.134.52
Discovered open port 53/tcp on 64.13.134.52
Discovered open port 80/tcp on 64.13.134.52
SYN Stealth Scan Timing: About 6.20% done; ETC: 00:11 (0:07:33 remaining)
Completed SYN Stealth Scan at 00:10, 463.55s elapsed (65536 total ports)
Completed Service scan at 00:10, 6.03s elapsed (3 services on 1 host)
Initiating OS detection (try #1) against scanme.nmap.org (64.13.134.52)
Initiating Traceroute at 00:10
64.13.134.52: guessing hop distance at 9
Completed SCRIPT ENGINE at 00:10, 4.04s elapsed
Host scanme.nmap.org (64.13.134.52) appears to be up ... good.
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 65530 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 4.3 (protocol 2.0)
25/tcp    closed  smtp
53/tcp    open  domain ISC BIND 9.3.4
70/tcp    closed gopher
80/tcp    open  http   Apache httpd 2.2.2 ((Fedora))
|_ HTML title: Go ahead and ScanMe!
113/tcp   closed auth
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.20-1 (Fedora Core 5)
Uptime guess: 2.457 days (since Thu Sep 18 13:13:24 2008)
TCP Sequence Prediction: Difficulty=204 (Good luck!)
IP ID Sequence Generation: All zeros

TRACEROUTE (using port 80/tcp)
HOP RTT      ADDRESS
[First eight hops cut for brevity]
9  10.36 metro0.sv.svcolo.com (208.185.168.173)
10  10.29 scanme.nmap.org (64.13.134.52)

Nmap done: 1 IP address (1 host up) scanned in 477.23 seconds
          Raw packets sent: 131432 (5.783MB) | Rcvd: 359 (14.964KB)

```

Nmap, Örnek 4.3'te istenen ayrıntı düzeyini kesinlikle sağlamıştır! Neyse ki ekstra çıktıının anlaşılması kolaydır. İlk 13 yeni satır çalışma zamanı bilgisidir ve kullanıcı iyi haberler almayı umarak terminale bakarken neler olup bittiğini bilmesini sağlar. İyi haberin ne olduğu, sorunları düzeltmek zorunda olan bir sistem yöneticisi mi, raporlamak için bazı sorunlara ihtiyaç duyan bir pen-tester mi yoksa bunları istismar etmeye çalışan bir black-hat cracker mı olduğuna bağlıdır. Yaklaşık bir düzine benzer satır kısalık nedeniyle çıkarılmıştır. "Açık bağlantı noktası keşfedildi" satırları, açık bağlantı noktalarının olduğu gibi bildirilmesini sağlar, böylece tarama

bitmeden önce onlara vurmaya başlayabilir. "Tarama zamanlaması" satırı bir tamamlanma süresi tahmini sağlar, böylece ekranı bilmeye devam edip etmeyeceğini veya öğle yemeği yiip yemeyeceğini bilir. Ağ koşulları (gecikme, tıkanıklık, bant genişliği, vb.) ve paket filtreleme kuralları çok fazla değişiklik gösterdiğinde, aynı tarama seçeneklerinin bir ana bilgisayara karşı tamamlanması 30 saniye, diğerine karşı ise 45 dakika sürebilir. Tarama sırasında geçerli zaman tahminini istiyorsanız, sadece enter tuşuna basın.

Bağlantı noktası tablosunda yeni bağlantı noktası görünmüyordu. Taranan tüm ekstra bağlantı noktaları filtrelenmiş durumdadır ve filtrelenmiş bağlantı noktası toplamı 994'ten 65.530'a yükselmiştir. Yeni öğelendirilmiş bağlantı noktası olmasa da girişler değişmiştir. Yeni bir VERSION sütunu, dinleme hizmeti için uygulama adını ve sürüm ayrıntılarını sağlar. Bu, -A seçeneği tarafından etkinleştirilen özelliklerden biri olan hizmet algılamadan gelir. Servis tespitinin bir diğer özelliği de SERVICE sütunundaki tüm servis protokollerinin gerçekten doğrulanmış olmasıdır. Önceki taramada, nmap-services port numarası aramasının nispeten zayıf sezgiselliğine dayanıyorlardı. Bu tablo araması bu sefer doğru oldu, ancak her zaman doğru olmayacağından emin olmak gereklidir.

A ile eklenen bir başka özellik de Bölüm 9, Nmap Komut Dosyası Motoru'nda derinlemesine tartışılan Nmap Komut Dosyası Motorudur. Burada gösterilen tek komut dosyası HTML başlığıdır. Düzinelerce başka komut dosyası var, ancak hiçbirini bu makine için yararlı çıktı bulamadı. Traceroute sonuçları da -A ile eklenmiştir. Bu seçenek çoğu traceroute programından daha verimli ve daha güçlündür çünkü problemler paralel olarak gerçekleştirilebilir ve Nmap uygun bir prob türü belirlemek için tarama sonuçlarını kullanır (bu durumda port 80'e TCP paketleri).

Kalan yeni satırların çoğu, Bölüm 8, Uzak İşletim Sistemi Algılama'da derinlemesine tartışılan işletim sistemi algılamasından (-A ile de etkinleştirilir) gelir. Son satır, tüm bu ekstra bilgilerin bir bedeli olduğunu göstermektedir; taramanın tamamlanması Örnek 4.2, "Basit tarama: nmap scanme.nmap.org "dan neredeyse 100 kat daha uzun sürmüştür (5'e kıyasla 477 saniye).

## Command-line Flags (Komut Satırı Bayrakları)

Eğitimde bir Nmap port taramasının ne kadar basit olabileceği gösterilmiş olsa da, sistemi daha güçlü ve esnek hale getirmek için dzinelerce komut satırı bayrağı mevcuttur. Bu bölüm yalnızca port taramalarıyla ilgili seçenekleri kapsar ve genellikle bu seçeneklerin yalnızca port taramasıyla ilgili işlevlerini açıklar. Seçenek bayraklarının ve yaptıkları her şeyin kapsamlı bir listesi için Bölüm 15, Nmap Başvuru Kılavuzu'na bakın.

## Tarama Tekniklerinin Seçilmesi

Bir port taraması yapmayı düşünürken dikkat edilmesi gereken ilk hususlardan biri hangi tekniklerin kullanılacağına karar vermektedir. Nmap yaklaşık bir düzine yöntem sunmaktadır ve bu bölüm bunların kısa bir özetini sunmaktadır. Tam kapsam bir sonraki bölümde yer almaktadır. UDP taramasının (-sU) TCP tarama türlerinden herhangi biriyle birleştirilebilmesi dışında, aynı anda yalnızca bir tarama yöntemi kullanılabilir. Hafıza yardımı olarak, port tarama tipi seçenekleri -s<C> şeklindedir, burada <C> tarama adında öne çıkan bir karakterdir, genellikle ilk karakterdir. Bunun tek istisnası, kullanımdan kaldırılan FTP sığrama taramasıdır (-b). Varsayılan olarak, Nmap bir SYN Taraması gerçekleştirir, ancak kullanıcının ham paketler göndermek için uygun ayrıcalıkları yoksa (Unix'te root erişimi gerektirir) veya IPv6 hedefleri belirtilmişse bir bağlantı taraması yerine geçer.

Nmap tarafından desteklenen port tarama yöntemleri

"TCP SYN (Gizli) Tarama (-sS)" (-sS) adlı bölüm

Bu en popüler tarama türüdür çünkü en popüler protokolün (TCP) portlarını taramanın en hızlı yoludur. Bağlantı taramasından daha gizlidir ve tüm işlevsel TCP yiğinlarına karşı çalışır (FIN taraması gibi bazı özel amaçlı taramaların aksine).

"TCP Bağlantı Taraması (-sT)" (-sT) adlı bölüm

Connect scan, diğer yöntemlerin çoğunu yaptığı gibi ham paketlere güvenmek yerine makineleri taramak için aynı adı taşıyan sistem çağrısını kullanır. Genellikle ayrıcalıksız Unix kullanıcıları tarafından ve IPv6 hedeflerine karşı kullanılır çünkü SYN taraması bu durumlarda çalışmaz.

"UDP Taraması (-sU)" adlı bölüm (-sU)

UDP bağlantı noktalarını da unutmamın; onlar da pek çok güvenlik açığı sunar.

"TCP FIN, NULL ve Xmas Taramaları (-sF, -sN, -sX)" (-sF, -sX, -sN) adlı bölüm

Bu özel amaçlı tarama türleri, arkalarındaki sistemleri keşfetmek için güvenlik duvarlarını gizlice geçmekte ustadır. Ne yazık ki bazı sistemlerin (özellikle Windows varyantlarının) sergilemediği hedef davranışlara dayanırlar.

#### "TCP ACK Taraması (-sA)" (-sA) adlı bölüm

ACK taraması genellikle güvenlik duvarı kural kümelerini haritalamak için kullanılır. Özellikle, güvenlik duvarı kurallarının durum bilgisi içerip içermediğini anlamaya yardımcı olur. Dezavantaj ise açık ve kapalı portları ayırt edememesidir.

#### "TCP Pencere Taraması (-sW)" (-sW) adlı bölüm

Pencere taraması ACK taraması gibidir, ancak belirli makinelere karşı açık ve kapalı portları tespit edebilir.

#### "TCP Maimon Taraması (-sM)" (-sM) adlı bölüm

Güvenlik duvarlarını aşan bu belirsiz tarama türü FIN taramasına benzer, ancak ACK bayrağını da içerir. Bu, daha fazla paket filtreleme güvenlik duvarından geçmesini sağlar, ancak FIN taramasından daha az sisteme karşı çalışmasının dezavantajı vardır.

#### "TCP Idle Scan (-sl)" (-sl <zombi ana bilgisayar>) adlı bölüm

Boşta tarama en gizli tarama türündür ve bazen güvenilir IP adresi ilişkilerinden faydalananabilir. Ne yazık ki, aynı zamanda yavaş ve karmaşıktır.

#### "IP Protokol Taraması (-sO)" (-sO) adlı bölüm

Protokol taraması, hedef makine tarafından hangi IP protokollerinin (TCP, ICMP, IGMP, vb.) desteklendiğini belirler. Bu teknik olarak bir port taraması değildir, çünkü TCP veya UDP port numaraları yerine IP protokol numaraları arasında geçiş yapar. Yine de taranan protokol numaralarını seçmek için -p seçeneğini kullanır, sonuçlarını normal port tablosu formatında raporlar ve hatta gerçek port tarama yöntemleriyle aynı temel tarama motorunu kullanır. Yani buraya ait olması için bir port taramasına yeterince yakındır.

#### "TCP FTP Bounce Scan (-b)" (-b <FTP bounce proxy>) adlı bölüm

Bu kullanımdan kaldırılmış tarama türü, FTP sunucularını proxy ile port taraması yapmaları için kandırır. Çoğu FTP sunucusu artık bunu önlemek için yamalanmıştır, ancak işe yaradığında kısıtlayıcı güvenlik duvarlarından gizlice geçmek için iyi bir yoldur.

## Selecting Ports to Scan (Taranacak Bağlantı Noktalarını Seçme)

Nmap'in port kayıt dosyası (nmap-services), her bir TCP veya UDP portunun ne sıkılıkla açık bulunduğuna dair deneysel veriler içerir. Bu veriler, on milyonlarca Internet adresinin taranması ve ardından bu sonuçların büyük kuruluşların katkıda bulunduğu dahili tarama verileriyle birleştirilmesiyle toplanmıştır. Varsayılan olarak, Nmap taraması istenen her protokolün en popüler 1.000 portunu tarar. Alternatif olarak, her protokolde yalnızca en yaygın 100 bağlantı noktasını taramak için -F (hızlı) seçeneğini veya taranacak rastgele sayıda bağlantı noktası belirtmek için --top-ports seçeneğini belirtebilirsiniz.

Bu hazır port setlerinden hiçbirini ihtiyaçlarınızı karşılamadığında, -p seçeneği ile komut satırında rastgele bir port numarası listesi belirtilebilir. p seçeneğinin sözdizimi karmaşık olabilir ve en iyi örneklerle açıklanabilir.

p seçeneği ile port seçim örnekleri

**-p 22** Tek bir bağlantı noktasını (bu durumda bağlantı noktası 22) -p argümanı olarak yalnızca bu numarayı belirterek tarayın.

**-p ssh** Sayılar yerine bağlantı noktası adları belirtilebilir. Bir ismin birden fazla portla eşleşebileceğini unutmayın.

**-p 22,25,80** Birden fazla port virgülle ayrılabilir. Herhangi bir protokol belirtilmediğine dikkat edin, bu nedenle komut satırında belirtilen tarama yöntemleri için aynı bağlantı noktası numaraları kullanılacaktır. SYN taraması (-sS) gibi bir TCP taraması belirtilirse, TCP bağlantı noktaları 22, 25 ve 80 taranır. Bunlar sırasıyla SSH, SMTP ve HTTP hizmetlerine karşılık gelir. Bir UDP taraması seçilirse (-sU), bu üç UDP bağlantı noktası taranır. Her ikisi de belirtilirse, bu üç bağlantı noktası her protokol için taranır ve toplam altı taranan bağlantı noktası elde edilir. IP protokol taraması (-sO) ile, bu üç IP protokolü (XNS IDP, Leaf-1 ve ISO-IP'ye karşılık gelen) taranır.

**-p80-85,443,8000-8005,8080-8085** Port aralıkları, başlangıç ve bitiş portu bir tire ile ayrılarak belirtilebilir. Birden fazla aralık veya tek tek portlar virgülle belirtilebilir. Bu seçenek 80, 81, 82, 83, 84, 85, 443, 8000 vb. portları tarar. Bağlantı noktası numaralarına göre, bu kullanıcı muhtemelen TCP'yi tarıyor ve web sunucularını arıyor.

**-p-100,60000-** Birinci bağlantı noktasını belirtmek için aralığın başını veya mümkün olan son bağlantı noktasını belirtmek için sonunu atlayabilirisiniz (TCP ve UDP için

65535, protokol taraması için 255). Bu örnek, birden 100'e kadar olan bağlantı noktalarını ve 60.000'e eşit veya daha büyük tüm bağlantı noktalarını tarar.

**-p-** Tüm aralığı taramak için başlangıç ve bitiş numaralarını atlayın (sıfır hariç).

**-pT:21,23,110,U:53,111,137,161** TCP ve UDP portlarının ayrı listeleri, listelerin başına T: (TCP için) veya U: getirilerek verilebilir. Bu örnekte üç TCP bağlantı noktası (FTP, Telnet ve POP3) ve dört UDP hizmeti (DNS, rpcbind, NetBIOS ve SNMP) taranmaktadır. Hem TCP hem de UDP bağlantı noktalarını belirtmek, yalnızca Nmap'e bir UDP taraması (-sU) ve -sS, -sA veya -sF gibi TCP tarama yöntemlerinden birini de söyleerseniz önemlidir.

**-p http\*** Benzer adlara sahip bağlantı noktalarını eşleştirmek için joker karakterler kullanılabilir. Bu ifade, http (80), http-mgmt (280), https (443) ve http-proxy (8080) dahil olmak üzere sekiz bağlantı noktası numarasıyla eşleşir. Komut kabuğunuza bağlı olarak, dosya adı glob'u olarak değerlendirilmemesi için yıldız işaretinden kaçmanız gerekebilir.

**-p 1-1023,[1024-]** Bir aralığın parantez içine alınması, bu bağlantı noktası numaralarının yalnızca nmap-services'te kayıtlısa taramasına neden olur. Bu örnekte, tüm ayrılmış portlar (1-1,023), artı nmap-services'de kayıtlı tüm yüksek portlar. Bu, nmap-services'in daha hassas seçim için açık port frekans verileriyle artırılmasından önce Nmap'in varsayılan davranışydı.

### **Timing-related Options (Zamanlama ile İlgili Seçenekler)**

Port taraması genellikle bir Nmap taramasının en çok zaman alan kısmıdır (işletim sistemi tespiti, sürüm tespiti ve NSE komut dosyalarını da içerebilir). Nmap varsayılan olarak hızlı ve verimli olmaya çalışsa da, manuel optimizasyon genellikle yardımcı olur. Nmap, tarama yoğunluğunu ve hızını ihtiyaçlarınıza tam olarak uyacak şekilde uyarlamak için düzinelere seçenek sunar. Bu bölümde port tarama sürelerini optimize etmek için en önemli seçenekler listelenmektedir. Bir süre alan seçenekler varsayılan olarak saniye cinsindendir veya değere ms (milisaniye), s (saniye), m (dakika) veya h (saat) ekleyebilirsiniz. Bu seçeneklerden herhangi biri hakkında daha fazla ayrıntı için "Zamanlama ve Performans" bölümüne bakın. Nmap performansını iyileştirmek için örnekler ve en iyi uygulamalar içeren çok daha kapsamlı bir inceleme Bölüm 6, Nmap Performansını Optimize Etme'de mevcuttur.

En iyi bağlantı noktası tarama performansı seçenekleri

`-T0` through `-T5` Bu zamanlama şablonları birçok değişkeni etkileyerek genel Nmap hızını çok yavaştan (`-T0`) aşırı agresife (`-T5`) ayarlamak için basit bir yol sunar. Bir zamanlama şablonu aşağıda açıklanan daha ayrıntılı seçeneklerle birleştirilebilir ve en ayrıntılı seçenek önceliklidir.

`--min-rtt-timeout`, `--max-rtt-timeout`, `--initial-rtt-timeout` Nmap'in bir port tarama probu yanıtı için bekleyeceği minimum, maksimum ve başlangıç süresi.

`--host-timeout` Nmap'e, taranması verilen süreden daha uzun süren ana bilgisayarlardan vazgeçmesini söyler.

`--min-rate`, `--max-rate` Nmap'in saniyede gönderdiği yoklama paketi sayısı için sırasıyla taban ve tavan değerlerini ayarlar.

`--max-retries` Tek bir porta maksimum port tarama probu yeniden iletim sayısını belirtir.

`--min-hostgroup`, `--max-hostgroup` Nmap'in paralel olarak tarayacağı minimum ve maksimum ana bilgisayar sayısını ayarlar.

`--min-parallelism`, `--max-parallelism` Nmap'in bekletebileceği minimum veya maksimum port tarama probu sayısını (eşzamanlı olarak taranan tüm ana bilgisayarlar arasında) sınırlar.

`--scan-delay`, `--max-scan-delay` Nmap'in herhangi bir ana bilgisayara prob gönderme arasında en az verilen süre kadar beklemesini ister. Nmap paket kaybı tespit ettikçe tarama gecikmesi artabilir, bu nedenle `--max-scan-delay` ile bir maksimum belirtilebilir.

## Output Format and Verbosity Options (Çıktı Formatı ve Verbosity Seçenekleri)

Nmap, raporlarını standart formatında, basit bir satır odaklı "grepable" formatında veya XML olarak yazma olanağı sunar. Bu raporlar `-oN` (normal), `-oG` (grepable) ve `-oX` (XML) seçenekleri ile etkinleştirilir. Her seçenek bir dosya adı alır ve aynı anda birden fazla formatta çıktı almak için birleştirilebilirler. Çıktı karmaşıklığını artırmak için çeşitli seçenekler de mevcuttur. Bu bölümde çıktı ile ilgili en önemli seçenekler ve bunların port taramasına nasıl uygulandığı listelenmektedir. Bu seçeneklerden herhangi biri hakkında daha fazla ayrıntı için "Çıktı" adlı bölüme bakın. Çıktı seçenekleri ve formatları ile ilgili çok daha kapsamlı bir çalışma, birçok örnekle birlikte, Bölüm 13, Nmap Çıktı Formatları'nda mevcuttur.

Port taramaları için geçerli en iyi Nmap çıktı seçenekleri

**-v** Ayrıntı düzeyini artırarak Nmap'in devam etmekte olan tarama hakkında daha fazla bilgi yazdırmasına neden olur. Açık portlar bulundukça gösterilir ve Nmap bir taramanın birkaç dakikadan fazla süreceğini düşündüğünde tamamlanma süresi tahminleri sağlanır. Daha fazla ayrıntı için iki veya daha fazla kullanın.

**-d** Hata ayıklama seviyesini artırarak Nmap'in çalışmasıyla ilgili hataları takip etmek veya sadece nasıl çalıştığını anlamak için yararlı olabilecek ayrıntıları yazdırmasına neden olur. Daha yüksek seviyeler büyük miktarda veri ile sonuçlanır. Seçeneğin bir kez kullanılması hata ayıklama düzeyini bire ayarlar ve her ek -d için bu düzey artırılır. Veya -d5'te olduğu gibi -d'yi istediğiniz seviye ile takip edebilirsiniz. Yeterli bilgi göremiyorsanız, daha yüksek bir seviye deneyin. Maksimum etkili seviye dokuzdur. Ekranınız çok fazla hata ayıklama verisi ile doluya, seviyeyi azaltın. Taranan port veya hedef sayısı ve kullanılan özellikler gibi tarama yoğunluğunun azaltılması da yalnızca istediğiniz hata ayıklama mesajlarının izole edilmesine yardımcı olabilir.

**--packet-trace** Nmap'in gönderilen veya alınan her paketin bir özeti yazdırmasına neden olur. Bu genellikle hata ayıklama için kullanılır, ancak aynı zamanda yeni kullanıcıların Nmap'in örtülerin altında tam olarak ne yaptığını anlamaları için değerli bir yoldur. Binlerce satır yazdırılmaktan kaçınmak için, -p20-30 gibi taranacak sınırlı sayıda bağlantı noktası belirtmek isteyebilirsiniz.

**-oN <filename>** (normal output) Çıktıyı Nmap'in normal biçiminde <dosya adı>'na yazın. Bu biçim, çalışma zamanında Nmap tarafından yazdırılan standart etkileşimli çıktı ile kabaca aynıdır.

**-oX <filename>** (XML output) Çıktıyı Nmap'in XML biçiminde <dosya adı>'na yazın. XML'in <dosya adı> olarak belirtilerek oraya yönlendirilmesini istediğiniz sürece normal (insan tarafından okunabilir) çıktı stdout'a yazdırılmaya devam edecektir. Bu, Nmap sonuçlarını işleyen komut dosyaları ve programlar tarafından kullanılmak üzere tercih edilen biçimdir.

**-oG <filename>** (grepable format output) Çıktıyı Nmap'in grepable olarak adlandırılan biçimde <dosya adı>'na yazın. Bu tablo biçimini, her ana bilgisayarın çıktısını tek bir satıra sıyrıarak açık bağlantı noktaları, belirli işletim sistemleri, uygulama adları veya diğer veriler için grep yapmayı kolaylaştırır. Greplenebilir çıktıının - <dosya adı> olarak belirtilerek oraya yönlendirilmesini istediğiniz sürece normal çıktı stdout'a yazdırılmaya devam edecektir. Bu format basit grep ve awk komut satırları ile ayırtırma için iyi çalışsa da, önemli komut dosyaları ve programlar bunun

yerine XML çıktısını kullanmalıdır. XML biçimini, greplenebilir biçimden yer vermediği önemli bilgiler içerir ve genişletilebilirlik, XML'in kendisine güvenen araçları bozmadan yeni bilgilerle güncellenmesini kolaylaştırır.

**-oA <basename>** (output to all formats) Kolaylık olması açısından, tarama sonuçlarını normal, XML ve grepable formatlarında bir kerede saklamak için -oA <basename> belirtebilirsiniz. Bunlar sırasıyla <basename>.nmap, <basename>.xml ve <basename>.gnmap dosyalarında saklanır. Çoğu programda olduğu gibi, dosya adlarının önüne Unix'te ~/nmaplogs/foocorp/ veya Windows'ta c:\hacking\sco gibi bir dizin yolu ekleyebilirsiniz.

**--resume <filename>** Kötü sonuçlanan tarama sırasında oluşturulan normal (-oN) veya greplenebilir (-oG) çıktı dosyasını belirterek iptal edilen bir taramayı devam ettirin. Nmap çıktı dosyasında belirtilenleri kullanacağından --resume dışında herhangi bir seçenek kullanmayın. Daha sonra dosyayı ayırtırır ve önceki Nmap yürütmesinin durdurulduğunda üzerinde çalıştığı ana bilgisayarda taramaya (ve dosyaya günlük tutmaya) devam eder.

**--append-output** Nmap'e tarama sonuçlarını üzerine yazmak yerine belirtilen çıktı dosyalarına (-oN veya -oX gibi argümanlarla) eklemesini söyler.

**--open** Yalnızca açık bağlantı noktaları olan ana bilgisayarları gösterin ve yalnızca bunlar için açık bağlantı noktalarını gösterin. Burada, "açık bağlantı noktaları" açık, açık|filtrelenmiş ve filtrelenmemiş olmak üzere açık olma olasılığı olan tüm bağlantı noktalarıdır.

## **Firewall and IDS Evasion Options (Güvenlik Duvarı ve IDS Kaçırma Seçenekleri)**

Nmap, IDS'leri tespit edilmeden geçmek veya güvenlik duvarı kurallarını atlatmak için birçok seçenek sunar. Genel bir bakış için "Güvenlik Duvarı/IDS Kaçırma ve Spoofing" adlı bölüme bakın. Güvenlik duvarı ve IDS atlatma tekniklerine pratik örneklerle birlikte kapsamlı bir bakış için Bölüm 10, Güvenlik Duvarlarını ve Saldırı Tespit Sistemlerini Algılama ve Yıkma'ya bakın.

## **Specifying Targets (Hedeflerin Belirlenmesi)**

Tek bir ana bilgisayarı (veya birkaçını) taramak için, Nmap komut satırınızın sonuna adlarını veya IP adreslerini eklemeniz yeterlidir. Nmap ayrıca büyük ağların taranmasını kolaylaştmak için yapılandırılmış bir sözdizimine sahiptir. Nmap'e hedefleri listeleyen bir dosya verebilir ya da Nmap'ten bunları rastgele

oluşturmasını isteyebilirsiniz. Tüm bunlar "Hedef Ana Bilgisayarları ve Ağları Belirleme" bölümünde açıklanmaktadır.

### Miscellaneous Options (Çeşitli Seçenekler)

Burada, belirli kategorilere uymasalar da oldukça kullanışlı olabilecek bazı seçenekler bulunmaktadır. Açıklamalar her bir seçeneğin port tarama ile nasıl ilişkili olduğuna odaklanmaktadır. Her bir seçeneğin daha kapsamlı kapsamı için Bölüm 15, Nmap Referans Kılavuzu'na bakın.

**-6** Nmap'ten IPv6 protokolünü kullanarak hedefi taramasını ister. Bu işlem "IPv6 Taraması (-6)" adlı bölümde açıklanmıştır.

**-r** Nmap, algılamayı biraz daha zorlaştırmak için varsayılan olarak bağlantı noktası tarama sırasını rastgele hale getirir. r seçeneği bunun yerine sayısal sırayla taranmalarına neden olur.

**-Pn** Nmap'e ping testini atlamasını ve sağlanan her hedef ana bilgisayarı taramasını söyler. Ana bilgisayar bulmayı kontrol etmek için diğer seçenekler Bölüm 3, Ana Bilgisayar Bulma ("Ping Tarama") bölümünde açıklanmıştır.

**--reason** İlginç portlar tablosuna Nmap'in bir portu neden bu şekilde sınıflandırdığını açıklayan bir sütun ekler.

## IPv6 Scanning ( **-6** ) (IPv6 Taraması (-6))

Nmap, 2002 yılından bu yana en popüler özellikleri için IPv6 desteği sunmaktadır. Özellikle, ping tarama (yalnızca TCP), bağlantı tarama ve sürüm algılama özelliklerinin tümü IPv6'yi desteklemektedir. Komut sözdizimi, -6 seçeneğini de eklemeniz dışında her zamanki gibidir. Elbette, bir ana bilgisayar adı yerine bir adres belirtirseniz IPv6 sözdizimini kullanmanız gereklidir. Bir adres 3ffe:7501:4819:2000:210:f3ff:fe03:14d0 gibi görünebilir, bu nedenle ana bilgisayar adları önerilir. Örnek 4.4 tipik bir port tarama oturumunu göstermektedir. Çıktı her zamanki gibi aynı görünüyor, "ilginç bağlantı noktaları" satırındaki IPv6 adresi IPv6'yı ele veren tek şey.

Örnek 4.4. Basit bir IPv6 taraması

```
# nmap -6 -sV www.eurov6.org

Starting Nmap ( https://nmap.org )
Nmap scan report for ns1.euro6ix.com (2001:800:40:2a03::3)
Not shown: 996 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      Pure-FTPd
22/tcp    open  ssh      OpenSSH 3.5p1 (protocol 2.0)
53/tcp    open  domain  ISC BIND 9.2.1
80/tcp    open  http    Apache httpd

Nmap done: 1 IP address (1 host up) scanned in 56.78 seconds
```

IPv6 dünyayı tam olarak kasıp kavurmamış olsa da, bazı ülkelerde önemli ölçüde kullanılmaktadır ve çoğu modern işletim sistemi bunu desteklemektedir. Nmap'i IPv6 ile kullanmak için, taramanızın hem kaynağı hem de hedefi IPv6 için yapılandırılmış olmalıdır. İSS'niz (çoğu gibi) size IPv6 adresleri tahsis etmiyorsa, ücretsiz tünel aracları yaygın olarak mevcuttur ve Nmap ile iyi çalışır. Ben <http://www.tunnelbroker.net> adresindeki ücretsiz IPv6 tünel aracı hizmetini kullanıyorum. Diğer tünel aracları Wikipedia'da listelenmiştir. 6to4 tünelleri bir başka popüler, ücretsiz yaklaşımındır.

IPv6'yi destekleyen sistemlerin IPv4 ve IPv6 güvenlik duvarı kuralları her zaman senkronize değildir. "IPv6 Saldırıları" adlı bölümde IPv4'tefiltrelenen bağlantı noktalarına IPv6 üzerinden ulaşmanın gerçek hayattan bir örneği gösterilmektedir.

## **SOLUTION: Scan a Large Network for a Certain Open TCP Port (ÇÖZÜM: Belirli Bir Açık TCP Bağlantı Noktası İçin Büyük Bir Ağı Tarayın)**

### **Problem (Problem)**

Bir ağ üzerinde belirli bir TCP bağlantı noktası açık olan tüm makineleri hızlı bir şekilde bulmak istiyorsunuz. Örneğin, yeni bir Microsoft IIS güvenlik açığı bulunduktan sonra, TCP bağlantı noktası 80 açık olan tüm makineleri taramak ve bu yazılımın savunmasız bir sürümünü çalıştırmadıklarından emin olmak isteyebilirsiniz. Ya da güvenliği ihlal edilmiş bir kutuyu araştırırsanız ve saldırganın 31337 numaralı bağlantı noktasında çalışan bir arka kapı bıraktığını tespit

ederseniz, tüm ağınızı bu bağlantı noktası için taramak, güvenliği ihlal edilmiş diğer sistemleri hızlı bir şekilde belirleyebilir. Tam (tüm portlar) tarama daha sonra yapılacaktır.

## Solution (Çözüm)

En kolay yol koşmaktadır:

```
nmap -Pn -p <portnumber> -oG <logfilename.gnmap> <target networks>
```

Burada web sunucuları için 4096 IP'nin aranmasına ilişkin somut bir örnek verilmiştir (port 80 açık):

```
nmap -Pn -p80 -oG logs/pb-port80scan-%D.gnmap 216.163.128.0/20
```

Dosya adındaki "%D", taramanın çalıştırıldığı sayısal tarihle değiştirilir (örneğin, 1 Eylül 2007'de "090107"). Bu tarama komutu çalışsa da, taranan ağ için uygun zamanlama değerlerini seçmek için biraz çaba harcamak tarama süresini önemli ölçüde azaltır. Yukarıdaki tarama 1.236 saniye sürerken, aşağıdaki optimize edilmiş sürüm aynı sonuçları 869 saniyede sağlamıştır:

```
nmap -T4 -Pn -p80 --max-rtt-timeout 200ms --initial-rtt-timeout 150ms  
--min-hostgroup 512 -oG logs/pb-port80scan2-%D.gnmap  
216.163.128.0/20
```

Ve bu sürenin çoğu ters-DNS çözümlemesi yapmak için harcanır. Yukarıdaki komut satırına -n ekleyerek bunu hariç tutmak, 4096 ana bilgisayar tarama süresini 193 saniyeye düşürür. Üç dakika boyunca sabırlı olmak, daha önce 21 dakika boyunca sabretmekten çok daha kolay.

Yukarıdaki komutlar grepable-format sonuçlarını belirtilen dosyada saklar. Basit bir egrep komutu 80 numaralı portu açık olan makineleri bulacaktır:

```
egrep '[^0-9]80/open' logs/pb-port80scan2-*.gnmap
```

Egrep kalıbü, 3180 gibi sahte eşleşen portlardan kaçınmak için [^0-9] ile öncelenir. Tabii ki sadece 80 numaralı portu taradığımız için böyle bir şey olamaz, ancak çok portlu taramalar için hatırlanması gereken iyi bir uygulamadır. Eğer sadece IP adreslerini istiyorsanız ve başka bir şey istemiyorsanız, egrep çıktısını awk '{print \$2}' komutuna yönlendirin.

## Discussion (Tartışma)

Bazen bir hikaye, çözüm bölümündeki komut satırlarına nasıl karar verdiğimde olduğu gibi, kararları anlamanın en iyi yoludur. Evde canım sıkılıyordu ve Playboy adlı popüler bir derginin ağını keşfetmeye başladım. Ana sitelerinde devasa bir görsel hazinesi var, ancak çoğu ücretli abonelik kimlik doğrulama sisteminin arkasında kilitli. Ağlarında görüntüleri ücretsiz olarak sunan başka sistemler bulup bulamayacağımı merak ediyordum. Parola doğrulaması yerine belirsizliğe dayanan hazırlama ya da geliştirme sunucuları olabileceğini düşündüm. Bu tür sunucular teorik olarak herhangi bir port numarasını dinleyebilse de, en olası port 80 numaralı TCP portudur. Bu yüzden mümkün olduğunda hızlı bir şekilde tüm ağlarını bu açık port için taramaya karar verdim.

İlk adım hangi IP adreslerinin taranacağını belirlemektir. Playboy adlı kuruluşlar için American Registry for Internet Numbers'da (ARIN) bir whois araması gerçekleştiriyorum. Sonuçlar Örnek 4.5'te gösterilmektedir.

#### Örnek 4.5. Playboy'un IP alanını keşfetme

```
core-> whois -h whois.arin.net n playboy
[Querying whois.arin.net]
[whois.arin.net]

OrgName:    Playboy
OrgID:      PLAYBO
Address:    680 N. Lake Shore Drive
City:       Chicago
StateProv:   IL
PostalCode: 60611
Country:    US

NetRange:   216.163.128.0 - 216.163.143.255
CIDR:       216.163.128.0/20
NetName:    PLAYBOY-BLK-1
NetHandle:  NET-216-163-128-0-1
Parent:     NET-216-0-0-0-0
NetType:    Direct Assignment
NameServer: NS1-CHI.PLAYBOY.COM
NameServer: NS2-CHI.PLAYBOY.COM
[...]
```

Bu, Playboy'a kayıtlı 4096 IP'yi (216.163.128.0/20 net aralığı) göstermektedir. "Bir Kuruluşun IP Adreslerini Bulma" bölümünde tartışılan teknikleri kullanarak kontrol ettikleri çok daha fazla netblok bulabilirdim, ancak 4096 IP bu örnek için yeterlidir.

Daha sonra bu makinelere olan gecikme süresini tahmin etmek istiyorum, böylece Nmap ne bekleyeceğini bilecek. Bu gerekli değildir, ancak Nmap'e uygun zamanlama değerlerini vermek onu hızlandırabilir. Bu özellikle bunun gibi tek portlu -Pn taramaları için geçerlidir. Nmap, gecikme süresini ve paket düşme oranını doğru bir şekilde tahmin etmek için her ana bilgisayardan yeterli yanıt almaz, bu yüzden komut satırında ona yardımcı olacağım. İlk düşüncem, Örnek 4.6'da gösterildiği gibi ana web sunucularına ping atmak.

Örnek 4.6. Gecikme tahmini için Playboy'un web sunucusuna ping atma

```
# ping -c5 www.playboy.com
PING www.phat.playboy.com (209.247.228.201) from 205.217.153.56
64 bytes from free-chi.playboy.com (209.247.228.201): icmp_seq=1 time=57.5 ms
64 bytes from free-chi.playboy.com (209.247.228.201): icmp_seq=2 time=56.7 ms
64 bytes from free-chi.playboy.com (209.247.228.201): icmp_seq=3 time=56.9 ms
64 bytes from free-chi.playboy.com (209.247.228.201): icmp_seq=4 time=57.0 ms
64 bytes from free-chi.playboy.com (209.247.228.201): icmp_seq=5 time=56.6 ms

--- www.phat.playboy.com ping statistics ---
5 packets transmitted, 5 received, 0% loss, time 4047ms
rtt min/avg/max/mdev = 56.652/57.004/57.522/0.333 ms
```

Maksimum gidiş dönüş süresi 58 milisaniyedir. Ne yazık ki, bu IP adresi (209.247.228.201) taramak istediğim 216.163.128.0/20 ağ bloğu içinde değil. Normalde bu yeni netbloku hedef listesine eklerdim, ancak taramamı orijinal 4096 IP ile sınırlamaya karar verdim. Bu süreleri kullanmak muhtemelen gayet iyi, ancak hedef ağdaki IP'lerden gerçek değerleri bulmak daha da iyi olurdu. Önceki whois sorgusunda gösterilen bir ad sunucusundan Playboy'un genel DNS kayıtlarını elde etmek için dig kullanıyorum. Çıktı Örnek 4.7'de gösterilmektedir.

Örnek 4.7. Playboy'un DNS kayıtlarını incelemek

```

core-> dig @ns1-chi.playboy.com playboy.com. any
; <>> Dig 8.3 <>> @ns1-chi.playboy.com playboy.com. any
[...]
;; ANSWER SECTION:
playboy.com.          1D IN A            209.247.228.201
playboy.com.          1D IN MX           10 mx.la.playboy.com.
playboy.com.          1D IN MX           5 mx.chi.playboy.com.
playboy.com.          1D IN NS            ns15.customer.level3.net.
playboy.com.          1D IN NS            ns21.customer.level3.net.
playboy.com.          1D IN NS            ns29.customer.level3.net.
playboy.com.          1D IN NS            ns1-chi.playboy.com.
playboy.com.          1D IN NS            ns2-chi.playboy.com.
playboy.com.          1D IN SOA           ns1-chi.playboy.com. dns.playboy.com. (
2004092010      ; serial
12H              ; refresh
2h30m            ; retry
2w1d             ; expiry
1D )             ; minimum

;; ADDITIONAL SECTION:
mx.chi.playboy.com.   1D IN A            216.163.143.4
mx.la.playboy.com.   1D IN A            216.163.128.15
ns1-chi.playboy.com. 1D IN A            209.247.228.135
ns2-chi.playboy.com. 1D IN A            64.202.105.36

;; Total query time: 107 msec

```

DNS sorusu, hedef 216.163.128.0/20 ağ bloğu içinde iki MX (posta) sunucusu ortaya çıkarır. mx.chi ve mx.la adları farklı bölgelerde (Chicago ve Los Angeles) olduğunu ima ettiğinden, her ikisini de gecikme açısından test etmeye karar verdim. Ping sonuçları Örnek 4.8'de gösterilmektedir.

Örnek 4.8. MX sunucularına ping atma

```

core-> ping -c5 mx.chi.playboy.com
PING mx.chi.playboy.com (216.163.143.4) 56(84) bytes of data.

--- mx.chi.playboy.com ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4000ms

core-> ping -c5 mx.la.playboy.com
PING mx.la.playboy.com (216.163.128.15) 56(84) bytes of data.

--- mx.la.playboy.com ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4011ms

```

Bu girişim tam bir başarısızlığı! Ana bilgisayarlar ICMP ping paketlerini engelliyor gibi görünüyor. Posta sunucusu olduklarından, TCP bağlantı noktası 25 açık olmalıdır, bu nedenle Örnek 4.9'da gösterildiği gibi bağlantı noktası 25'e karşı TCP pingi gerçekleştirmek için hping2'yi kullanarak tekrar deniyorum.

#### Örnek 4.9. MX sunucularına TCP ping atma

```
core# hping2 --syn -p 25 -c 5 mx.chi.playboy.com
eth0 default routing interface selected (according to /proc)
HPING mx.chi.playboy.com (eth0 216.163.143.4): S set, 40 headers + 0 data bytes
46 bytes from 216.163.143.4: flags=SA seq=0 ttl=51 id=14221 rtt=56.8 ms
46 bytes from 216.163.143.4: flags=SA seq=1 ttl=51 id=14244 rtt=56.9 ms
46 bytes from 216.163.143.4: flags=SA seq=2 ttl=51 id=14274 rtt=56.9 ms
46 bytes from 216.163.143.4: flags=SA seq=3 ttl=51 id=14383 rtt=61.8 ms
46 bytes from 216.163.143.4: flags=SA seq=4 ttl=51 id=14387 rtt=57.5 ms

--- mx.chi.playboy.com hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 56.8/58.0/61.8 ms

core# hping2 --syn -p 25 -c 5 mx.la.playboy.com
eth0 default routing interface selected (according to /proc)
HPING mx.la.playboy.com (eth0 216.163.128.15): S set, 40 headers + 0 data bytes
46 bytes from 216.163.128.15: flags=SA seq=0 ttl=52 id=58728 rtt=16.0 ms
46 bytes from 216.163.128.15: flags=SA seq=1 ttl=52 id=58753 rtt=15.4 ms
46 bytes from 216.163.128.15: flags=SA seq=2 ttl=52 id=58790 rtt=15.5 ms
46 bytes from 216.163.128.15: flags=SA seq=3 ttl=52 id=58870 rtt=16.4 ms
46 bytes from 216.163.128.15: flags=SA seq=4 ttl=52 id=58907 rtt=15.5 ms

--- mx.la.playboy.com hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 15.4/15.8/16.4 ms
```

Aradığım sonuçlar bunlardı. LA ana bilgisayarının yanıt vermesi hiçbir zaman 16 milisaniyeden fazla sürmezken, Chicago'daki 62 milisaniyeye kadar sürüyor. Kaliforniya'daki bir makineden tarama yaptığım düşünüldüğünde bu şaşırtıcı değil. Dikkatli olmakta fayda var ve yoğun tarama sırasında gecikme süresi artabilir, bu yüzden Nmap'in yanıtlar için 200 milisaniyeye kadar beklemesine izin vermeye karar verdim. Zaman aşımı 150 ms ile başlamasını sağlayacağım. Bu yüzden ona --max-rtt-timeout 200ms --initial-rtt-timeout 150ms seçeneklerini iletiyorum. Genel olarak agresif bir zamanlama modu ayarlamak için satırın başında -T4 belirtiyorum.

Tüm taramanın tamamlanma süresini en aza indirmeyi, ana bilgisayar sonuçlarının ilk grubu döndürülmeden önce geçen süreyi en aza indirmeye tercih ettiğim için, büyük bir tarama grubu boyutu belirtiyorum. --min-hostgroup 512 seçeneği, en az 512 IP'nin paralel olarak taranması için belirtilir (mükün olduğunda). Hedef ağ

boyutunun (4096) tam bir faktörünü kullanmak, --min-hostgroup 500 belirtmiş olsaydım sonunda oluşacak küçük ve daha az verimli 96 ana bilgisayar bloğunu önler. Tüm bu zamanlama sorunları Bölüm 6, Nmap Performansını Optimize Etme'de çok daha derinlemesine açıklanmıştır.

Bir ping işlemi tek portlu taramanın kendisi kadar uzun süreceğinden önceki bir ping aşamasıyla zaman kaybetmeye gerek yoktur. Bu yüzden bu aşamayı devre dışı bırakmak için -Pn belirtilir. n argümanı ile ters-DNS çözümlemesini atlayarak önemli ölçüde zaman kazanılır. Aksi takdirde, ping taraması devre dışı bırakıldığında, Nmap 4096 IP'nin tamamını aramaya çalışacaktır. Ben web sunucularını arıyorum, bu yüzden -p80 ile 80 portunu talep ediyorum. Elbette 81 veya 8080 gibi standart olmayan portlarda çalışan HTTP sunucularını kaçıracağım. Ayrıca 443 numaralı bağlantı noktasındaki SSL sunucuları da bulunamayacaktır. Bunları -p seçeneğine ekleyebilirsiniz, ancak bir port daha bile tarama süresini iki katına çıkaracaktır ki bu da kabaca taranan port sayısıyla orantılıdır.

Örnek 4.10. Taramayı başlatma

```
# nmap -T4 -p80 -Pn --max-rtt-timeout 200ms --initial-rtt-timeout 150ms \
--min-hostgroup 512 -n -oG pb-port80scan-%D.gnmap 216.163.128.0/20
Warning: You specified a highly aggressive --min-hostgroup.
Starting Nmap ( https://nmap.org )
Nmap scan report for 216.163.128.0
PORT      STATE      SERVICE
80/tcp    filtered  http

Nmap scan report for 216.163.128.1
PORT      STATE      SERVICE
80/tcp    filtered  http

Nmap scan report for 216.163.128.2
PORT      STATE      SERVICE
80/tcp    filtered  http

Nmap scan report for 216.163.128.3
PORT      STATE      SERVICE
80/tcp    filtered  http
[ ... ]
Nmap scan report for 216.163.143.255
PORT      STATE      SERVICE
80/tcp    filtered  http

Nmap done: 4096 IP addresses (4096 hosts up) scanned in 192.97 seconds
```

Nmap 4096 IP'nin tamamını yaklaşık üç dakika içinde tarar. Normal çıktı filtrelenmiş durumda bir sürü port gösterir. Bu IP'lerin çoğu muhtemelen aktif ana

bilgisayarlar değildir - Nmap SYN problemlerine yanıt alamadığı için port filtrelenmiş görünür. Web sunucularının listesini Örnek 4.11'de gösterildiği gibi çıktı dosyasında basit bir egrep ile elde ediyorum.

#### Örnek 4.11. Açık portlar için Egrep

```
# egrep '[^0-9]80/open' pb-port80scan-* .gnmap
Host: 216.163.140.20 () Ports: 80/open/tcp//http///
Host: 216.163.142.135 () Ports: 80/open/tcp//http///
```

Tüm bu çabadan sonra, 4096 IP'den yalnızca iki erişilebilir web sunucusu bulundu! Bazen böyle olur. İlk, 216.163.140.20 (ters DNS adı yok) beni bir Microsoft Outlook Web Access (webmail) sunucusuna getiriyor. Ağlarını ele geçirmeye çalışıyor olsaydım bu beni heyecanlandırabilirdi, ancak şu anda tatmin edici değil. Bir sonraki sunucu (ters isim [mirrors.playboy.com](http://mirrors.playboy.com)) çok daha iyi. Umdüğüm gigabaytlarca ücretsiz imajları sunuyor! Özellikle Linux ISO imajlarının yanı sıra önemli FreeBSD, CPAN ve Apache arşivleri de sunuyor! En son Fedora Core ISO'larını 6 Mbps gibi hatırlı sayılır bir hızda indiriyorum. Playboy'daki bant genişliği bolluğu şaşırtıcı değil. Daha sonra diğer Playboy ağ bloklarını tarıyorum, düzinelerce web sunucusu daha buluyorum, ancak bazı içerikleri bu kitap için uygun değil.

Bu port taraması için alışılmadık bir neden olsa da, tek port taramaları daha önce ifade edilen diğer birçok amaç için yaygındır. Burada açıklanan teknikler herhangi bir tek portlu TCP taramasına kolaylıkla uygulanabilir.

#### See Also (Ayrıca Bakınız)

Sürüm tespiti, bir ağ üzerinde dinleme yapan belirli uygulamaları bulmak için kullanılabilir. Örneğin, 22 numaralı bağlantı noktası açık olan tüm ana bilgisayarları bulmak yerine OpenSSH'nin belirli bir savunmasız sürümünü arayabilirsiniz. Bu çözümdeki teknikler yalnızca TCP için iyi çalıştığından, bu aynı zamanda tek portlu UDP taramaları için de kullanılabilir. Talimatlar "ÇÖZÜM: Güvensiz veya Standart Olmayan Uygulama Sürümü Çalıştıran Tüm Sunucuları Bulun" adlı bölümde verilmiştir.

Bölüm 6, Nmap Performansını Optimize Etme, tarama hızı optimizasyonunu çok daha derinlemesine incelemektedir.

# Chapter 5. Port Scanning Techniques and Algorithms

## (Bölüm 5. Liman Tarama Teknikleri ve Algoritmaları)

İçindekiler

- Introduction (İçindekliler)
- TCP SYN (Stealth) Scan ( `sS` ) (Giriş TCP SYN (Gizli) Taraması (-sS))
- TCP Connect Scan ( `sT` ) ( TCP Bağlantı Taraması (-sT))
- UDP Scan ( `sU` ) (UDP Taraması (-sU))
  - Distinguishing Open from Filtered UDP Ports (Açık ve Filtrelenmiş UDP Portlarını Ayırt Etme)
  - Speeding Up UDP Scans (UDP Taramalarını Hızlandırmaya )
- TCP FIN, NULL, and Xmas Scans ( `sF` , `sN` , `sX` ) (TCP FIN, NULL ve Xmas Taramaları (-sF, -sN, -sX))
- Custom Scan Types with `-scanflags` (--scanflags ile Özel Tarama Türleri)
  - Custom SYN/FIN Scan (Özel SYN/FIN Taraması)
  - PSH Scan (PSH Taraması)
- TCP ACK Scan ( `sA` ) (TCP ACK Taraması (-sA))
- TCP Window Scan ( `sW` ) ( TCP Window Taraması (-sW) )
- TCP Maimon Scan ( `sM` ) (TCP Maimon Taraması (-sM))
- TCP Idle Scan ( `sI` ) (TCP Idle Taraması (-sI))
  - Idle Scan Step by Step ( Idle Taraması Adım Adım)
  - Finding a Working Idle Scan Zombie Host (Çalışan Bir Boşta Tarama Zombi Ana Bilgisayarı Bulma)
  - Executing an Idle Scan (Boşta Tarama Yürütme)
  - Idle Scan Implementation Algorithms ( Idle Taraması Uygulama Algoritmaları)
- IP Protocol Scan ( `sO` ) (IP Protokolü Taraması (-sO))
- TCP FTP Bounce Scan ( `b` ) (TCP FTP Sığrama Taraması (-b))

- Scan Code and Algorithms (Tarama Kodu ve Algoritmaları)
  - Network Condition Monitoring (Ağ Durumu İzleme)
  - Host and Port Parallelization (Ana Bilgisayar ve Bağlantı Noktası Paralelleştirme)
  - Round Trip Time Estimation (Gidiş Dönüş Süresi Tahmini)
  - Congestion Control (Tıkanıklık Kontrolü)
  - Timing probes ( Zamanlama Problemleri)
  - Inferred Neighbor Times (Çıkarılan Komşu Süreleri)
  - Adaptive Retransmission (Uyarlanabilir Yeniden İletim)
  - Scan Delay (Tarama Gecikmesi)

## **Introduction (Giriş)**

Otomotiv tamiri yapan bir acemi olarak, ilkel aletlerimi (çekiç, koli bandı, anahtar vb.) elimdeki işe uydurmak için saatlerce uğraşabilirim. Başarısız olduğumda ve külüstürümü gerçek bir tamirciye götürdüğümde, her zaman işi zahmetsiz hale getiren mükemmel aleti çıkarana kadar büyük bir alet sandığını karıştırır. Port tarama sanatı da benzerdir. Uzmanlar düzinelere tarama tekniğini anlar ve belirli bir görev için uygun olanı (veya kombinasyonu) seçerler. Deneyimsiz kullanıcılar ve script çocukları ise her sorunu varsayılan SYN taraması ile çözmeye çalışırlar. Nmap ücretsiz olduğu için, port tarama ustalığının önündeki tek engel bilgidir. Bu kesinlikle otomotiv dünyasını geride bırakıyor; burada bir dikme yayı kompresörüne ihtiyacınız olduğunu belirlemek için büyük bir beceri gerekebilir, ancak yine de bunun için binlerce dolar ödemek zorunda kalırsınız.

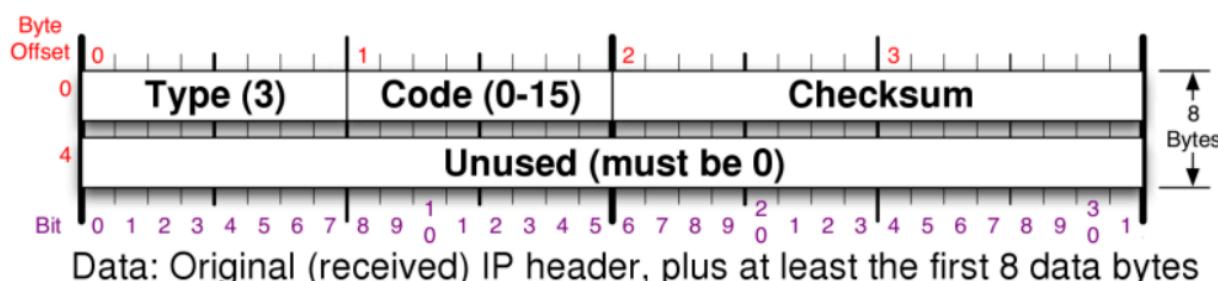
Bir önceki bölümde Nmap ile port taraması genel hatlarıyla anlatılmış ve "Tarama Tekniklerinin Seçilmesi" bölümünde Nmap'in desteklenen tarama türlerinin kısa bir özeti verilmiştir. Bu bölüm, bu tarama türlerinin her birini derinlemesine açıklamaktadır. Her bir tarama türü için tipik kullanım senaryoları ve talimatlar verilmekte, ayrıca nasıl çalışıklarını gösteren kablo üzerinde paket izleri sunulmaktadır. Daha sonra ultra\_scan algoritması (çoğu tarama yönteminin

kullandığı), performansı artırmak için değiştirilebilecek yönlerde vurgu yapılarak tartışılmaktadır.

Tarama türlerinin çoğu yalnızca ayrıcalıklı kullanıcılar tarafından kullanılabilir. Bunun nedeni, Unix sistemlerinde root erişimi gerektiren ham IP paketleri (hatta ethernet çerçeveleri) gönderip almalarıdır. Windows'ta bir yönetici hesabı kullanmak gerekli değildir, çünkü Npcap zaten işletim sistemine yüklenliğinde Nmap bu platformdaki ayrıcalıksız kullanıcılar için çalışır. Nmap 1997'de piyasaya sürüldüğünde kök ayrıcalıkları gerektirmek ciddi bir sınırlamaydı, çünkü birçok kullanıcının yalnızca paylaşılan kabuk hesaplarına erişimi vardı. Şimdi ise dünya farklı. Bilgisayarlar daha ucuz, çok daha fazla insan her zaman doğrudan Internet erişimine sahip ve masaüstü Unix sistemleri (Linux ve Mac OS X dahil) yaygın. Nmap'in Windows sürümü artık mevcut olup daha da fazla masaüstünde çalışmasına olanak sağlamaktadır. Tüm bu nedenlerden dolayı, kullanıcıların Nmap'i nadiren sınırlı paylaşımı kabuk hesaplarından çalıştırmaları gereklidir. Ayrıcalıklı seçenekler Nmap'i çok daha güçlü ve esnek hale getirdiği için bu bir şans.

Nmap'in prob yanıtlarını nasıl ele aldığına tartışırken, birçok bölüm ICMP hata mesajlarını türlerine ve kod numaralarına göre tartışır. Tür ve kod, ICMP başlıklarında bulunan ve mesajın amacını tanımlayan sekiz bitlik alanlardır. Nmap port tarama teknikleri sadece ICMP tip 3 ile ilgilidir, bunlar hedefe ulaşılamaz mesajlardır. Şekil 5.1 böyle bir paketin ICMP başlık düzenini göstermektedir (Şekil 1, "IPv4 başlığı" bölümünde gösterildiği gibi bir IP paketinin veri bölümünden kapsüllemeştir).

Şekil 5.1. ICMPv4 hedefe ulaşılamıyor başlık düzeni



Farklı hedefe ulaşılamıyor mesajlarını temsil eden on altı kod vardır. Hepsi Tablo 5.1'de gösterilmiştir, ancak Nmap yalnızca yıldızla işaretlenmiş olan 0-3, 9, 10 ve 13 kodlarıyla ilgilenir.

Tablo 5.1. ICMP hedefe ulaşılamıyor (tip 3) kod değerleri

<b>Code</b>	<b>Description</b>
0*	Ağa erişilemiyor
1*	Ana bilgisayara ulaşılamıyor
2*	Protokole ulaşılamıyor
3*	Bağlantı noktasına ulaşılamıyor
4	Parçalama gerekli ancak parçalama yapma biti ayarlı
5	Kaynak rota başarısız oldu
6	Hedef ağ bilinmiyor
7	Hedef ana bilgisayar bilinmiyor
8	Kaynak ana bilgisayar yalıtılmış (eski)
9*	Hedef ağ idari olarak yasaklanmıştır
10*	Hedef ana bilgisayar idari olarak yasaklandı
11	Hizmet türü (TOS) için ağa erişilemiyor
12	Ana bilgisayara TOS için erişilemiyor
13*	Filtreleme ile idari olarak yasaklanmış iletişim
14	Ana bilgisayar öncelik ihlali
15	Öncelik kesintisi yürürlükte

## **TCP SYN (Stealth) Scan ( -sS ) (TCP SYN (Gizli) Tarama (-sS))**

SYN taraması iyi bir nedenden dolayı varsayılan ve en popüler tarama seçenekidir. Hızlı bir şekilde gerçekleştirilebilir, müdahaleci güvenlik duvarları tarafından engellenmeyen hızlı bir ağda saniyede binlerce bağlantı noktasını tarayabilir. SYN taraması, TCP bağlantılarını asla tamamlamadığı için nispeten göze batmaz ve gizlidir. Ayrıca Nmap'in FIN(NULL/Xmas, Maimon ve boşta taramaları gibi belirli platformların kendine has özelliklerine bağlı olmak yerine herhangi bir uyumlu TCP yiğinına karşı çalışır. Ayrıca açık, kapalı ve filtrelenmiş durumlar arasında net ve güvenilir bir ayrım yapılmasını sağlar.

SYN taraması Nmap'e -sS seçeneği iletilerek talep edilebilir. Ham paket ayrıcalıkları gerektirir ve mevcut olduklarında varsayılan TCP taramasıdır. Bu yüzden Nmap'i root ya da Administrator olarak çalıştırırken -sS genellikle atlanır. Bu varsayılan SYN tarama davranışları, üç ana durumun her birinde bir bağlantı noktası bulan Örnek 5.1'de gösterilmektedir.

Örnek 5.1. Üç bağlantı noktası durumunu gösteren bir SYN taraması

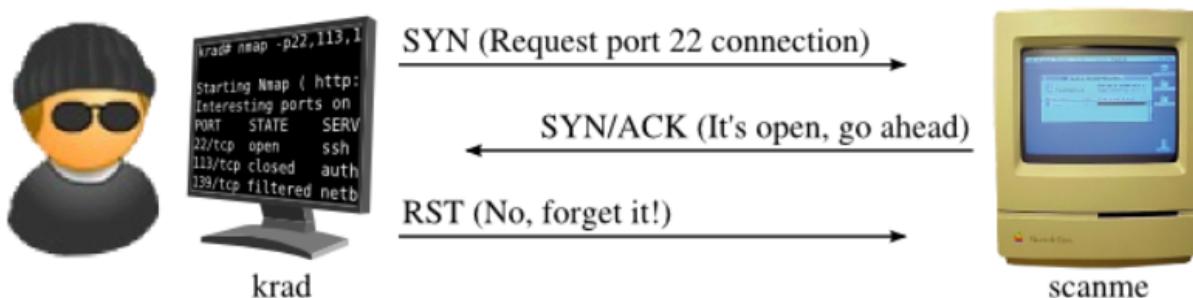
```
krad# nmap -p22,113,139 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
PORT      STATE    SERVICE
22/tcp    open     ssh
113/tcp   closed   auth
139/tcp   filtered netbios-ssn

Nmap done: 1 IP address (1 host up) scanned in 1.35 seconds
```

SYN taramasını herhangi bir alt düzey TCP bilgisi olmadan kullanmak oldukça kolay olsa da, tekniği anlamak olağanışı sonuçları yorumlarken yardımcı olur. Neyse ki bizim için, korkunç siyah şapka kırıcısı Ereet Hagiwara, Japon Windows kullanıcılarını terörize etmeye ara vererek Örnek 5.1 SYN taramasını bizim için paket düzeyinde gösterdi. İlk olarak, açık port 22'ye karşı davranış Şekil 5.2'de gösterilmiştir.

Şekil 5.2. Açık port 22'nin SYN taraması

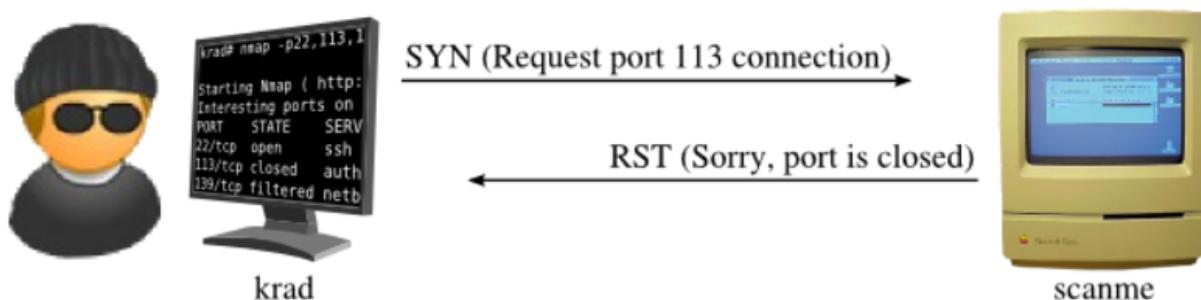


Bu örnekte gösterildiği gibi, Nmap SYN bayrağı ayarlanmış bir TCP paketini (paket başlıklarının neye benzediğini unuttuysanız Şekil 2, "TCP başlığı" bölümüne bakın) 22 numaralı bağlantı noktasına göndererek başlar. Bu, herhangi bir meşru bağlantı girişiminin gerçekleştiği TCP üç yönlü el sıkışmasının ilk adımıdır. Hedef port

açık olduğundan, Scanme SYN ve ACK bayraklarını içeren bir yanıt göndererek ikinci adımı atar. Normal bir bağlantıda, Ereet'in makinesi (krad olarak adlandırılır) SYN/ACK'yi onaylayan bir ACK paketi göndererek üç yönlü el sıkışmayı tamamlar. Nmap'in bunu yapmasına gerek yoktur, çünkü SYN/ACK yanıtı ona zaten portun açık olduğunu söylemiştir. Eğer Nmap bağlantıyı tamamlarsa, o zaman bağlantıyı kapatma konusunda endişelenmesi gerekecektir. Bu genellikle SYN yerine FIN paketlerinin kullanıldığı başka bir el sıkışmayı içerir. Bu yüzden ACK kötü bir fikirdir, ancak yine de bir şeyler yapılmalıdır. SYN/ACK tamamen göz ardı edilirse, Scanme bunun düşüğünü varsayıacak ve yeniden göndermeye devam edecektir. Tam bir bağlantı kurmak istemediğimiz için uygun yanıt, şemada gösterildiği gibi bir RST paketidir. Bu Scanme'ye bağlantı girişimini unutmasını (sıfırlamasını) söyler. Nmap bu RST paketini kolayca gönderebilir, ancak aslında buna gerek yoktur. Krad üzerinde çalışan işletim sistemi de SYN/ACK alır ve Nmap SYN probunu kendisi hazırladığı için bunu beklemez. Böylece işletim sistemi beklenmeyen SYN/ACK'ye bir RST paketi ile yanıt verir. Bu bölümde açıklanan tüm RST paketlerinde ACK biti de ayarlıdır, çünkü bunlar her zaman alınan bir pakete yanıt olarak gönderilir (ve onaylanır). Bu yüzden bu bit RST paketleri için açıkça gösterilmemiştir. Üç yönlü el sıkışma hiçbir zaman tamamlanmadığından, SYN taraması bazen yarı açık tarama olarak adlandırılır.

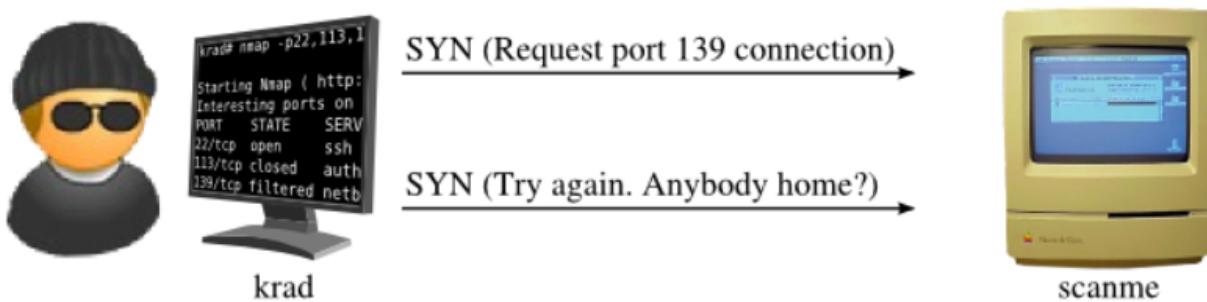
Şekil 5.3 Nmap'in 113 numaralı portun kapalı olduğunu nasıl belirlediğini göstermektedir. Bu, açık durumdan bile daha basittir. İlk adım her zaman aynıdır - Nmap Scanme'ye SYN probu gönderir. Ancak geri SYN/ACK almak yerine bir RST döndürülür. Bu her şeyi çözer - port kapanmıştır. Bu imanla ilgili daha fazla iletişime gerek yoktur.

Şekil 5.3. Kapalı bağlantı noktası 113'ün SYN taraması



Son olarak, Ereet bize Şekil 5.4'te filtrelenmiş bir portun Nmap'e nasıl göründüğünü göstermektedir. İlk SYN her zamanki gibi ilk olarak gönderilir, ancak Nmap hiçbir yanıt görmez. Yanıt basitçe yavaş olabilir. Önceki yanıtlardan (veya zamanlama varsayılanlarından), Nmap ne kadar bekleyeceğini bilir ve sonunda bir yanıt almaktan vazgeçer. Yanıt vermeyen bir bağlantı noktası genellikle filtrelenir (bir güvenlik duvarı cihazı tarafından engellenir veya belki de ana bilgisayar kapalıdır), ancak bu test kesin değildir. Belki de bağlantı noktası açıktır ancak sonda veya yanıt basitçe düşürülmüştür. Ağlar titrek olabilir. Bu yüzden Nmap SYN probunu yeniden göndererek tekrar dener. Başka bir zaman aşımı süresinden sonra, Nmap pes eder ve bağlantı noktasını filtrelenmiş olarak işaretler. Bu durumda, yalnızca bir yeniden iletim denenmiştir. "Tarama Kodu ve Algoritmalar" adlı bölümde açıklandığı gibi, Nmap dikkatli paket kaybı istatistikleri tutar ve daha az güvenilir ağları tararken daha fazla yeniden iletim dener.

Şekil 5.4. Filtrelenmiş port 139'un SYN taraması



Nmap ayrıca belirli ICMP hata mesajlarını geri alırsa bir bağlantı noktasını filtrelenmiş olarak değerlendirecektir. Tablo 5.2, Nmap'in bir SYN probuna verilen yanıtlarına göre port durumlarını nasıl atadığını göstermektedir.

Tablo 5.2. Nmap bir SYN probuna verilen yanıtları nasıl yorumlar?

Probe Response (Prob Yanıtı)	Assigned State (Atanmış Durum)
TCP SYN/ACK response	open
TCP RST response	closed
No response received (even after retransmissions)	filtered
ICMP unreachable error (type 3, code 1, 2, 3, 9, 10, or 13)	filtered

Bu bölümdeki güzel resimler elinizde olduğunda faydalı olsa da, Nmap, istediğiniz diğer komut satırı bayraklarına ek olarak --packet-trace seçeneğini belirttiğinizde paket düzeyinde tam olarak ne yaptığıni bildirir. Bu, yeni başlayanlar için Ereet yardım etmek için etrafta olmadığında Nmap'in davranışını anlamak için harika bir yoldur. İleri düzey kullanıcılar bile Nmap beklemeyenleri sonuçlar ürettiğinde bunu kullanışlı bulurlar. Hata ayıklama seviyesini -d (hatta -d5) ile de artırmak isteyebilirsiniz. Ardından, amacınız için gerekli olan minimum sayıda bağlantı noktası ve ana bilgisayarı tarayın, aksi takdirde kelimenin tam anlamıyla milyonlarca çıktı satırı ile karşılaşabilirsiniz. Örnek 5.2, Ereet'in üç portlu SYN taramasını paket izleme etkinleştirilmiş olarak tekrarlar (çıktı kısalık için düzenlenmiştir). Komut satırını okuyun, ardından okumaya devam etmeden önce hangi paketlerin gönderileceğini anlayarak kendinizi test edin. Ardından, "SYN Gizli Taraması 1,25 saniye sürdü" ifadesine kadar olan izi okuduğunuzda, okumaya devam etmeden önce RCVD satırlarından port durum tablosunun neye benzeyeceğini bilmelisiniz.

Örnek 5.2. Bir SYN taramasını anlamak için --packet-trace kullanımı

```
krad# nmap -d --packet-trace -p22,113,139 scanme.nmap.org

Starting Nmap ( https://nmap.org )
SENT (0.0130s) ICMP krad > scanme echo request (type=8/code=0) ttl=52 id=1829
SENT (0.0160s) TCP krad:63541 > scanme:80 A iplen=40 seq=91911070 ack=99850910
RCVD (0.0280s) ICMP scanme > krad echo reply (type=0/code=0) iplen=28
We got a ping packet back from scanme: id = 48821 seq = 714 checksum = 16000
massping done: num hosts: 1 num_responses: 1
Initiating SYN Stealth Scan against scanme.nmap.org (scanme) [3 ports] at 00:53
SENT (0.1340s) TCP krad:63517 > scanme:113 S iplen=40 seq=10438635
SENT (0.1370s) TCP krad:63517 > scanme:22 S iplen=40 seq=10438635
SENT (0.1400s) TCP krad:63517 > scanme:139 S iplen=40 seq=10438635
RCVD (0.1460s) TCP scanme:113 > krad:63517 RA iplen=40 seq=0 ack=10438636
RCVD (0.1510s) TCP scanme:22 > krad:63517 SA iplen=44 seq=75897108 ack=10438636
SENT (1.2550s) TCP krad:63518 > scanme:139 S iplen=40 seq=10373098 win=3072
The SYN Stealth Scan took 1.25s to scan 3 total ports.
Nmap scan report for scanme.nmap.org (64.13.134.52)
PORT      STATE     SERVICE
22/tcp    open      ssh
113/tcp   closed   auth
139/tcp   filtered netbios-ssn

Nmap done: 1 IP address (1 host up) scanned in 1.40 seconds
```

SYN taraması, Nmap yayınlanmadan önce en yaygın tarama türü olan TCP bağlantı taramasından (daha sonra ele alınacaktır) daha ince olduğu için uzun zamandır gizli tarama olarak adlandırılmaktadır. Bu lakaba rağmen, varsayılan bir SYN

taramasının hassas ağlardan fark edilmeden geçeceğine güvenmeyin. Yaygın olarak kullanılan saldırısı tespit sistemleri ve hatta kişisel güvenlik duvarları varsayılan SYN taramalarını tespit etme konusunda oldukça yeteneklidir. Gizli tarama için daha etkili teknikler Bölüm 10, Güvenlik Duvarlarını ve Saldırı Tespit Sistemlerini Algılama ve Yıkma'da gösterilmektedir.

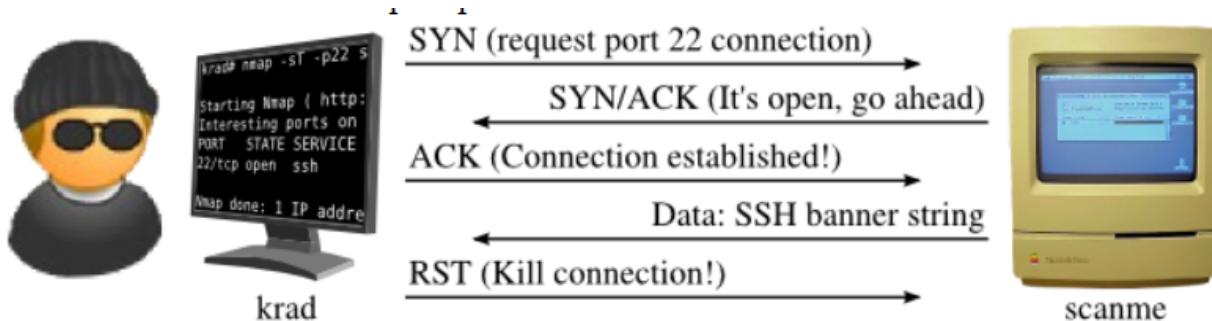
## **TCP Connect Scan ( [-sT](#) ) (TCP Bağlantı Taraması (-sT))**

TCP bağlantı taraması, SYN taraması bir seçenek olmadığındaysa varsayılan TCP tarama türündür. Bu, bir kullanıcının ham paket ayrıcalıklarına sahip olmadığı veya IPv6 ağlarını taradığı durumdur. Diğer tarama türlerinin çoğunun yaptığı gibi ham paketler yazmak yerine, Nmap altta yatan işletim sisteminden connect sistem çağrısını yayinallyarak hedef makine ve bağlantı noktası ile bir bağlantı kurmasını ister. Bu, web tarayıcılarının, P2P istemcilerinin ve diğer ağ özellikli uygulamaların çoğunun bağlantı kurmak için kullandığı aynı üst düzey sistem çağrısidir. Berkeley Sockets API olarak bilinen bir programlama arayüzünün parçasıdır. Nmap, kablodan ham paket yanıtlarını okumak yerine, her bağlantı denemesinde durum bilgisi almak için bu API'yi kullanır. Bu ve FTP sığrama taraması ("TCP FTP Sığrama Taraması (-b)" adlı bölüm) ayrıcalıksız kullanıcılar tarafından kullanılabilen tek tarama türündür.

SYN taraması mevcut olduğunda, genellikle daha iyi bir seçimdir. Nmap'in yüksek seviye bağlantı çağrıları üzerinde ham paketlere göre daha az kontrolü vardır, bu da onu daha az verimli hale getirir. Sistem çağrıları, SYN taramasının yaptığı yarı açık sıfırlamayı gerçekleştirmek yerine açık hedef portlara bağlantıları tamamlar. Bu sadece daha uzun sürmez ve aynı bilgiyi elde etmek için daha fazla paket gerektir, aynı zamanda hedef makinelerin bağlantıyı günlüğe kaydetme olasılığı daha yüksektir. İyi bir IDS her ikisini de yakalayacaktır, ancak çoğu makinede böyle bir alarm sistemi yoktur. Ortalama Unix sisteminizdeki birçok hizmet, Nmap bağlandığında ve ardından veri göndermeden bağlantıyı kapattığında syslog'a bir not ve bazen şifreli bir hata mesajı ekleyecektir. Bu gerçekleştiğinde gerçekten acınası hizmetler çöker, ancak bu nadirdir. Günlüklerinde tek bir sistemden bir sürü bağlantı girişimi gören bir yönetici, bağlantı taraması yapıldığını bilmelidir.

Şekil 5.5, scanme.nmap.org'un 22 numaralı açık portuna karşı yapılan bir bağlantı taramasını göstermektedir. Bunun Şekil 5.2'deki "22 numaralı açık portun SYN taraması" için yalnızca üç paket gerektirdiğini hatırlayın. Açık bir bağlantı noktasına karşı tam davranış Nmap'in üzerinde çalıştığı platforma ve diğer ucta dinleyen hizmete bağlıdır, ancak bu beş paket örneği tipiktir.

Şekil 5.5. Açık port 22'nin bağlantı taraması



İlk iki adım (SYN ve SYN/ACK) SYN taramasıyla tamamen aynıdır. Ardından, krad yarı açık bağlantıyı bir RST paketiyle iptal etmek yerine, SYN/ACK'yi kendi ACK paketiyle onaylayarak bağlantıyı tamamlar. Bu durumda, Scanme'nin SSH banner dizesini (SSH-1.99-OpenSSH\_3.1p1n) artık açık olan bağlantı üzerinden gönderecek zamanı bile olmuştur. Nmap ana işletim sisteminden bağlantının başarılı olduğunu duyar duymaz bağlantıyı sonlandırır. TCP bağlantıları genellikle FIN bayrağını içeren başka bir el sıkışma ile sona erer, ancak Nmap ana bilgisayar işletim sisteminden bağlantıyı hemen bir RST paketi ile sonlandırmamasını ister.

Bu bağlantı taraması örneği SYN taramasından neredeyse iki kat daha fazla paket almış olsa da, bant genişliği farkları nadiren bu kadar önemli olur. Büyük bir taramada portların büyük çoğunluğu kapalı ya da filtrelenmiş olacaktır. Bunlar için paket izleri Şekil 5.3, "113 numaralı kapalı portun SYN taraması" ve Şekil 5.4, "139 numaralı filtrelenmiş portun SYN taraması"nda SYN taraması için açıklananlarla aynıdır. Sadece açık portlar daha fazla ağ trafiği oluşturur.

Bir bağlantı taramasının çıktısı bir SYN taramasından önemli ölçüde farklı değildir. Örnek 5.3 Scanme'nin bir bağlantı taramasını göstermektedir. Nmap ayrıcalıklı olmayan bir hesaptan çalıştırıldığı için -sT seçeneği atlanmış olabilir, bu nedenle bağlantı taraması varsayılan türdür.

Örnek 5.3. Tarama örneğini bağlayın

```
krad~> nmap -T4 -sT scanme.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 994 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    closed smtp
53/tcp    open  domain
70/tcp    closed gopher
80/tcp    open  http
113/tcp   closed auth

Nmap done: 1 IP address (1 host up) scanned in 4.74 seconds
```

## UDP Scan ( -sU ) (UDP Scan (-sU))

İnternet üzerindeki en popüler hizmetler TCP protokolü üzerinden çalışsa da, UDP hizmetleri yaygın olarak kullanılmaktadır. DNS, SNMP ve DHCP (kayıtlı 53, 161/162 ve 67/68 numaralı portlar) en yaygın olanlardan üçüdür. UDP taraması genellikle TCP'den daha yavaş ve daha zor olduğundan, bazı güvenlik denetçileri bu portları görmezden gelir. Bu bir hatadır, çünkü istismar edilebilir UDP hizmetleri oldukça yaygındır ve saldırganlar kesinlikle tüm protokolü görmezden gelmezler. Neyse ki Nmap UDP portlarının envanterini çıkarmaya yardımcı olabilir.

UDP taraması -sU seçeneği ile etkinleştirilir. Aynı çalışma sırasında her iki protokolü de kontrol etmek için SYN taraması (-sS) gibi bir TCP tarama türü ile birleştirilebilir.

UDP taraması, hedeflenen her bağlantı noktasına bir UDP paketi göndererek çalışır. Çoğu port için bu paket boş olacaktır (yüksek), ancak daha yaygın olan birkaç port için protokole özgü bir yük gönderilecektir. Yanıt veya yanitsızlığa bağlı olarak, bağlantı noktası Tablo 5.3'te gösterildiği gibi dört durumdan birine atanır.

Tablo 5.3. Nmap bir UDP probuna verilen yanıtları nasıl yorumlar?

Probe Response (Prob Yanıtı)	Assigned State (Atanmış Durum)
Hedef porttan gelen herhangi bir UDP yanıtı (olağandışı)	open
Yanıt alınmadı (yeniden iletimlerden sonra bile)	open filtered
ICMP bağlantı noktasına ulaşılamıyor hatası (tip 3, kod 3)	closed
Diğer ICMP ulaşılamıyor hataları (tip 3, kod 1, 2, 9, 10 veya 13)	filtered

Bu tablonun en ilginç unsuru açık|filtreli durumu olabilir. Bu, UDP taramasındaki en büyük zorlukların bir belirtisidir: açık portlar boş probalara nadiren yanıt verir.

Nmap'in protokole özgü bir yükle sahip olduğu portların yanıt alma ve açık olarak işaretlenme olasılığı daha yüksektir, ancak geri kalanı için hedef TCP/IP yiğini boş paketi dinleyen bir uygulamaya iletir ve bu uygulama genellikle paketi geçersiz olarak hemen atar. Diğer tüm durumlardaki portlar yanıt verseydi, açık portların hepsi eleme yoluyla çıkarılabilirdi. Ne yazık ki, güvenlik duvarları ve filtreleme cihazlarının da yanıt vermeden paketleri düşürdüğü bilinmektedir. Dolayısıyla, Nmap birkaç denemeden sonra yanıt almadığında, bağlantı noktasının açık mı yoksa filtrelenmiş mi olduğunu belirleyemez. Nmap piyasaya sürüldüğünde, filtreleme cihazları Nmap'in portun açık olduğunu varsayılabileceği (ve yaptığı) kadar nadirdi. İnternet artık daha iyi korunuyor, bu nedenle Nmap 2004 yılında (sürüm 3.70) yanıt vermeyen UDP bağlantı noktalarını açık|filtrelenmiş olarak bildirecek şekilde değişti. Bunu, Ereet'in Felix adlı bir Linux kutusunu taradığını gösteren Örnek 5.4'te görebiliriz.

Örnek 5.4. UDP tarama örneği

```
krad# nmap -sU -v felix

Starting Nmap ( https://nmap.org )
Nmap scan report for felix.nmap.org (192.168.0.42)
(The 997 ports scanned but not shown below are in state: closed)
PORT      STATE          SERVICE
53/udp    open|filtered domain
67/udp    open|filtered dhcpserver
111/udp   open|filtered rpcbind
MAC Address: 00:02:E3:14:11:02 (Lite-on Communications)

Nmap done: 1 IP address (1 host up) scanned in 999.25 seconds
```

Felix'in bu taraması açık|filtrelenmiş belirsizlik sorununun yanı sıra başka bir sorunu da göstermektedir: UDP taraması yavaş olabilir. Felix ve diğer birçok Linux sistemi tarafından gerçekleştirilen ICMP yanıt hızı sınırlaması nedeniyle bu durumda bin portun taraması neredeyse 17 dakika sürmüştür. Nmap, aşağıdaki iki bölümde açıklandığı gibi her iki sorunun da üstesinden gelmek için yollar sunar.

### Distinguishing Open from Filtered UDP Ports (Açık ve Filtrelenmiş UDP Portlarını Ayırt Etme)

Felix taraması durumunda, üç açık|filtrelenmiş bağlantı noktası hariç hepsi kapalıydı. Yani tarama, potansiyel olarak açık portları bir avuçla sınırlandırmada hala başarılıydı. Bu durum her zaman geçerli değildir. Örnek 5.5, yoğun şekilde filtrelenmiş Scanme sitesine karşı bir UDP taramasını göstermektedir.

Örnek 5.5. UDP tarama örneği

```
krad# nmap -sU -T4 scanme.nmap.org
Starting Nmap ( https://nmap.org )
All 1000 scanned ports on scanme.nmap.org (64.13.134.52) are open|filtered
Nmap done: 1 IP address (1 host up) scanned in 5.50 seconds
```

Bu durumda, tarama açık bağlantı noktalarını hiç daraltmadı. 1000 tanesinin hepsi açık|filtreli. Yeni bir strateji gerekiyor.

Tablo 5.3, "Nmap bir UDP probuna verilen yanıtları nasıl yorumlar", Nmap'in UDP problemlerinden belirli bir porta herhangi bir yanıt alamadığında açık|filtreli durumunun olduğunu göstermektedir. Bununla birlikte, nadir durumlarda, bir bağlantı noktasını dinleyen UDP hizmetinin aynı şekilde yanıt vereceğini ve bağlantı noktasının açık olduğunu kanıtlayacağını da göstermektedir. Bu hizmetlerin sık sık yanıt vermemesinin nedeni, Nmap'in gönderdiği boş paketlerin geçersiz sayılmasıdır. Ne yazık ki, UDP hizmetleri genellikle Nmap'in her zaman gönderebileceği ortak bir genel biçimde bağlı kalmak yerine kendi paket yapılarını tanımlar. Bir SNMP paketi SunRPC, DHCP veya DNS istek paketinden tamamen farklı görünür.

Her popüler UDP hizmeti için uygun paketi göndermek için, Nmap'in prob formatlarını tanımlayan büyük bir veritabanına ihtiyacı olacaktır. Neyse ki Nmap, Bölüm 7, Hizmet ve Uygulama Sürüm Algılama'da açıklanan hizmet ve sürüm

algılama alt sisteminin bir parçası olan nmap-service-probes biçiminde buna sahiptir.

Sürüm taraması -sV (veya -A) ile etkinleştirildiğinde, her açık|filtrelenmiş bağlantı noktasına (bilinen açık olanların yanı sıra) UDP problemleri gönderilir. Eğer problemlerden herhangi biri açık|filtrelenmiş bir porttan yanıt alırsa, durum açık olarak değiştirilir. Felix taramasına -sV eklenmesinin sonuçları Örnek 5.6'da gösterilmektedir.

Örnek 5.6. Felix'in UDP tarama sonuçlarını sürüm tespiti ile iyileştirme

```
krad# nmap -sUV -F felix.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for felix.nmap.org (192.168.0.42)
Not shown: 997 closed ports
PORT      STATE          SERVICE      VERSION
53/udp    open           domain      ISC BIND 9.2.1
67/udp    open|filtered  dhcpserver
111/udp   open           rpcbind     2 (rpc #100000)
MAC Address: 00:02:E3:14:11:02 (Lite-on Communications)

Nmap done: 1 IP address (1 host up) scanned in 1037.57 seconds
```

Bu yeni tarama 111 ve 53 numaralı portların kesinlikle açık olduğunu gösteriyor. Yine de sistem mükemmel değil - 67 numaralı bağlantı noktası hala açık|filtreli. Bu özel durumda, bağlantı noktası açıktır ancak Nmap'in DHCP için çalışan bir sürüm probu yoktur. Bir başka zor hizmet de SNMP'dir ve genellikle yalnızca doğru topluluk dizesi verildiğinde yanıt verir. Birçok cihaz topluluk dizesi herkese açık olacak şekilde yapılandırılmıştır, ancak hepsi öyle değildir. Bu sonuçlar mükemmel olmasa da, test edilen üç bağlantı noktasından ikisinin gerçek durumunu öğrenmek yine de yararlıdır.

Ereet, Felix sonuçlarını ayırt etmedeki başarısından sonra, dikkatini geçen sefer tüm portları açık|filtreli olarak listeleyen Scanme'ye geri çevirir. Örnek 5.7'de gösterildiği gibi sürüm tespiti ile tekrar dener.

Örnek 5.7. Scanme'nin UDP tarama sonuçlarını sürüm algılama ile iyileştirme

```
krad# nmap -sUV -T4 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 999 open|filtered ports
PORT      STATE SERVICE VERSION
53/udp    open  domain  ISC BIND 9.3.4

Nmap done: 1 IP address (1 host up) scanned in 3691.89 seconds
```

Ereet sonunda açık portu bulsa da, önce Nmap sürümünü güncellemeyerek bir hata yaptı. Nmap sürüm 5.10BETA1 ve daha yeni sürümler, port taraması ya da host keşfi için seçildiklerinde üç düzineden fazla iyi bilinen UDP portuna uygun servis protokolü istekleri gönderen bir yük sistemine sahiptir. Sürüm tespiti kadar kapsamlı olmasa da, Örnek 5.5'teki 53 numaralı açık bağlantı noktasını hızlı bir şekilde tespit edebilirdi.

Bu sonuç bir önceki Scanme taramasının beş saniyesine karşılık bir saat sürmüştür, ancak bu sonuçlar gerçekten yararlıdır. Ereet'in gülümsemesi genişler ve tehlikeye atmak istediği bir makinede açık bir ISC BIND ad sunucusu olduğuna dair bu kanıt karşısında gözleri parlar. Bu yazılımın uzun bir güvenlik açığı geçmişti vardır, bu yüzden belki de bu son sürümde bir açık bulabilir.

Ereet, UDP saldırılарını açık olduğu doğrulandığı için 53 numaralı porta odaklayacaktır, ancak açık|filtreli olarak listelenen diğer 999 portu da unutmaz. Felix'teki dhcpserver portunda tanık olduğumuz gibi, bazı açık UDP hizmetleri Nmap sürüm tespitinden bile gizlenebilir. Ayrıca şu ana kadar sadece varsayılan portları taradı, açık olması muhtemel 64529 port daha var. Kaytlara geçmesi için, 53 Scanme'deki tek açık UDP bağlantı noktasıdır.

Bu sürüm tespit tekniği Nmap'in açık ve filtrelenmiş portları otomatik olarak ayırt etmesinin tek yolu olsa da, manuel olarak denenebilecek birkaç numara vardır. Bazen özel bir traceroute yardımcı olabilir. Nmap veya Nping gibi bir araçla bilinen açık bir TCP veya UDP portuna karşı bir traceroute yapabilirsiniz. Sonra aynı işlemi şüpheli UDP bağlantı noktasına karşı deneyin. Atlama sayılarındaki farklılıklar açık ve filtrelenmiş bağlantı noktalarını ayırt edebilir. Ereet bunu Örnek 5.8'de Scanme'ye karşı dener. İlk komut bilinen-açık bağlantı noktasını 53'e karşı bir UDP traceroute yapar. İkinci komut aynı şeyi kapalı olduğu varsayılan 54 numaralı bağlantı noktasına karşı yapar. Yer kazanmak için ilk birkaç atlama atlanmıştır.

Örnek 5.8. TTL tutarsızlıklarını olan UDP bağlantı noktalarının belirsizliğini giderme girişimi

```
krad# nping --udp --traceroute -c 13 -p 53 scanme.nmap.org
Starting Nping ( https://nmap.org/nping )
SENT (7.0370s) UDP 192.168.0.21:53 > 64.13.134.52:53 ttl=8 id=4826 iplen=28
RCVD (7.1010s) ICMP 4.69.134.222 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=248 id=38454 iplen=56
SENT (8.0400s) UDP 192.168.0.21:53 > 64.13.134.52:53 ttl=9 id=38166 iplen=28
RCVD (8.1050s) ICMP 4.68.18.204 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=247 id=39583 iplen=56
SENT (9.0420s) UDP 192.168.0.21:53 > 64.13.134.52:53 ttl=10 id=6788 iplen=28
RCVD (9.1080s) ICMP 4.59.4.78 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=246 id=59897 iplen=56
SENT (10.0440s) UDP 192.168.0.21:53 > 64.13.134.52:53 ttl=11 id=366 iplen=28
RCVD (10.1100s) ICMP 69.36.239.221 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=243 id=42710 iplen=56
SENT (11.0470s) UDP 192.168.0.21:53 > 64.13.134.52:53 ttl=12 id=63478 iplen=28
SENT (12.0490s) UDP 192.168.0.21:53 > 64.13.134.52:53 ttl=13 id=56653 iplen=28

Max rtt: 73.003ms | Min rtt: 0.540ms | Avg rtt: 48.731ms
Raw packets sent: 13 (364B) | Rcvd: 10 (560B) | Lost: 3 (23.08%)
Tx time: 12.02836s | Tx bytes/s: 30.26 | Tx pkts/s: 1.08
Rx time: 13.02994s | Rx bytes/s: 42.98 | Rx pkts/s: 0.77
Nping done: 1 IP address pinged in 13.05 seconds

krad# nping --udp --traceroute -c 13 -p 54 scanme.nmap.org
Starting Nping ( https://nmap.org/nping )
SENT (7.0370s) UDP 192.168.0.21:53 > 64.13.134.52:54 ttl=8 id=56481 iplen=28
RCVD (7.1130s) ICMP 4.69.134.214 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=248 id=22437 iplen=56
SENT (8.0400s) UDP 192.168.0.21:53 > 64.13.134.52:54 ttl=9 id=23264 iplen=28
RCVD (8.1060s) ICMP 4.68.18.76 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=247 id=50214 iplen=56
SENT (9.0430s) UDP 192.168.0.21:53 > 64.13.134.52:54 ttl=10 id=9101 iplen=28
RCVD (9.1070s) ICMP 4.59.4.78 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=246 id=880 iplen=56
SENT (10.0450s) UDP 192.168.0.21:53 > 64.13.134.52:54 ttl=11 id=35344 iplen=28
RCVD (10.1110s) ICMP 69.36.239.221 > 192.168.0.21 TTL=0 during transit (type=11/code=0) ttl=243 id=44617 iplen=56
SENT (11.0470s) UDP 192.168.0.21:53 > 64.13.134.52:54 ttl=12 id=53857 iplen=28
SENT (12.0490s) UDP 192.168.0.21:53 > 64.13.134.52:54 ttl=13 id=986 iplen=28

Max rtt: 76.488ms | Min rtt: 0.546ms | Avg rtt: 48.480ms
Raw packets sent: 13 (364B) | Rcvd: 11 (616B) | Lost: 2 (15.38%)
Tx time: 12.02988s | Tx bytes/s: 30.26 | Tx pkts/s: 1.08
Rx time: 13.03165s | Rx bytes/s: 47.27 | Rx pkts/s: 0.84
Nping done: 1 IP address pinged in 13.05 seconds
```

Bu örnekte, Ereet hem açık hem de kapalı portların yalnızca on birinci hop'una ulaşabilmisti. Dolayisyla bu sonclar, bu ana bilgisayara karasi bağlantı noktası durumlarını ayirt etmek icin kullanilamaz. Denemeye degerdi ve önemli sayida durumda işe yaradi. Tarama güvenlik duvarının hedef ana bilgisayardan en az bir ya da iki atlama önce olduğu durumlarda çalışması daha olasıdır. Öte yandan Scanme, kendi Linux iptables ana bilgisayar tabanlı güvenlik duvarını çalıştmaktadır. Dolayisyla filtrelenmiş ve açık portlar arasında hop sayısı açısından bir fark yoktur.

Başka bir teknik de uygulamaya özel araçları ortak bağlantı noktalarına karasi denemektir. Örneğin, bir kaba kuvvet SNMP topluluk dizesi kırcısı 161 numaralı bağlantı noktasına karasi denenebilir. Nmap'in sürüm tespit probu veritabanı büyündükçe, sonuçlarını harici özel araçlarla artırma ihtiyacı azalır. Özel bir topluluk dizesine sahip SNMP cihazları gibi özel durumlar için hala yararlı olacaklardır.

### Speeding Up UDP Scans (UDP Taramalarını Hızlandırma)

UDP taramasıyla ilgili diğer büyük zorluk bunu hızlı bir şekilde yapmaktadır. Açık ve filtrelenmiş portlar nadiren yanıt gönderir, bu da Nmap'i zaman aşımına uğratır ve ardından probun veya yanıtın kaybolması durumunda yeniden iletimler gerçekleştirir. Kapalı portlar genellikle daha da büyük bir sorundur. Genellikle bir ICMP portuna ulaşılamıyor hatası gönderirler. Ancak kapalı TCP portları tarafından bir SYN veya bağlantı taramasına yanıt olarak gönderilen RST paketlerinin aksine, birçok ana bilgisayar ICMP port ulaşılamaz mesajlarını varsayılan olarak sınırlar. Linux ve Solaris bu konuda özellikle katıdır. Örneğin, Felix üzerindeki Linux 2.4.20 çekirdeği hedefe ulaşılamayan mesajları saniyede bir ile sınırlar (`net/ipv4/icmp.c` içinde). Bu, Örnek 5.4, "UDP tarama örneği" ndeki taramanın neden bu kadar yavaş olduğunu açıklar.

Nmap hız sınırlamasını algılar ve hedef makinenin düşürecegi gereksiz paketlerle ağı doldurmaktan kaçınmak için buna göre yavaşlar. Ne yazık ki, Linux tarzı saniyede bir paket sınırı, 65.536 portlu bir taramanın 18 saatten fazla sürmesine neden olur. Burada UDP tarama performansını iyileştirmek için bazı öneriler bulunmaktadır. Ayrıca daha ayrıntılı tartışma ve genel tavsiyeler için Bölüm 6, Nmap Performansını Optimize Etme'yi okuyun.

Ana bilgisayar paralellliğini artırın ⇒ Nmap tek bir hedef ana bilgisayardan saniyede sadece bir port ulaşılamıyor hatası alıyorsa, bu tür 100 ana bilgisayarı aynı anda tarayarak 100/saniye hata alabilir. Bunu `--min-hostgroup`'a büyük bir değer (100 gibi) geçerek uygulayın.

Önce popüler bağlantı noktalarını tarayın ⇒ Çok az sayıda UDP port numarası yaygın olarak kullanılır. En yaygın 100 UDP portunun taraması (-F seçeneği kullanılarak) hızlı bir şekilde tamamlanacaktır. Daha sonra, arka planda ağını 65K portluk çok günlük bir taramasını başlatırken bu sonuçları inceleyebilirsiniz.

Sürüm algılama taramalarına `--version-intensity 0` ekleyin ⇒ Önceki bölümde belirtildiği gibi, açık ve filtrelenmiş UDP portlarını ayırt etmek için genellikle versiyon tespitini (`-sV`) gereklidir. Sürüm tespitini, hedef makinelerde bulunan her açık veya açık|filtrelenmiş bağlantı noktasına çok sayıda uygulama protokolüne özgü sonda göndermeyi içerdiginden nispeten yavaştır. Sürüm yoğunluğunun 0 olarak belirtilmesi, Nmap'i yalnızca belirli bir bağlantı noktasının numarasına karşı etkili olma olasılığı en yüksek olan problemleri denemeye yönlendirir. Bunu `nmap-service-probes` dosyasındaki verileri kullanarak yapar. Bu bölümün ilerleyen kısımlarında gösterileceği gibi, bu seçeneğin performansa etkisi büyiktür.

Güvenlik duvarının arkasından tarama ⇒ TCP'de olduğu gibi, paket filtreleri taramaları önemli ölçüde yavaşlatabilir. Birçok modern güvenlik duvarı paket hızı sınırlarını ayarlamayı kolaylaştırır. Taramayı güvenlik duvarı üzerinden değil de arkasından başlatarak bu sorunu aşabilirsınız, bunu yapın.

Yavaş ana bilgisayarları atlamak için --host-timeout kullanın ⇒ ICMP hızı sınırlı ana bilgisayarların taranması, her proba hızlı bir hedefe ulaşılamıyor paketiyle yanıt verenlere göre çok daha fazla zaman alabilir. Maksimum tarama süresi belirtmek (15 dakika için 15m gibi), Nmap'in bu kadar süre içinde taramayı tamamlamamışsa tek tek ana bilgisayarlardan vazgeçmesine neden olur. Bu, tüm duyarlı ana bilgisayarları hızlı bir şekilde taramanıza olanak tanır. Daha sonra arka planda yavaş ana bilgisayarlar üzerinde çalışabilirsınız.

v kullanın ve rahatlayın ⇒ Verbosity (-v) etkinleştirildiğinde, Nmap her ana bilgisayarın taramasının tamamlanması için tahmini süre sağlar. Yakından izlemenize gerek yoktur. Nmap sizin adınıza yorulmadan tarama yaparken biraz uyuyun, en sevginiz bara gidin, kitap okuyun, diğer işlerinizi bitirin ya da başka bir şekilde eğlenin.

UDP taramalarını optimize etme ihtiyacına mükemmel bir örnek Örnek 5.7, "Scanme'nin UDP tarama sonuçlarını sürüm tespiti ile iyileştirme". Tarama istenen verileri elde etti, ancak bu ana bilgisayarı taramak bir saatten fazla sürdü! Örnek 5.9'da Ereet bu taramayı tekrar çalıştırıyor. Bu kez -F --version-intensity 0 seçeneklerini ekliyor ve bir saat süren tarama 13 saniyeye düşüyor! Yine de aynı anahtar bilgi (53 numaralı portta çalışan bir ISC Bind daemon) tespit edilir.

#### Örnek 5.9. UDP Tarama Süresini Optimize Etme

```
krad# nmap -sUV -T4 -F --version-intensity 0 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 99 open|filtered ports
PORT      STATE SERVICE VERSION
53/udp    open  domain  ISC BIND 9.3.4

Nmap done: 1 IP address (1 host up) scanned in 12.92 seconds
```

## **TCP FIN, NULL, and Xmas Scans ( -sF , -sN , -sX ) (TCP FIN, NULL ve Xmas Taramaları (-sF, -sN, -sX))**

Bu üç tarama türü (bir sonraki bölümde açıklanan --scanflags seçeneği ile daha da fazlası mümkün) açık ve kapalı portlar arasında ayırmak için TCP RFC'deki ince bir boşluktan yararlanır. RFC 793'ün 65. sayfasında "[hedef] port durumu KAPALI ise .... bir RST içermeyen gelen bir segment yanıt olarak bir RST gönderilmesine neden olur" denmektedir. Bir sonraki sayfada, SYN, RST veya ACK bitleri ayarlanmadan açık portlara gönderilen paketler tartışılmakta ve şöyle denmektedir: "buraya ulaşmanız pek olası değil, ancak ulaşırsanız, segmenti bırakın ve geri dönün."

Bu RFC metniyle uyumlu sistemleri tararken, SYN, RST veya ACK bitlerini içermeyen herhangi bir paket, bağlantı noktası kapalıysa RST döndürülmesine ve bağlantı noktası açıksa hiçbir yanıt alınmamasına neden olur. Bu üç bitten hiçbirini dahil edilmediği sürece, diğer üçünün (FIN, PSH ve URG) herhangi bir kombinasyonu tamamdır. Nmap bunu üç tarama türü ile kullanır:

Null scan (-sN) ⇒ Hiçbir biti ayarlamaz (TCP bayrak başlığı 0'dır)

FIN scan (-sF) ⇒ Sadece TCP FIN bitini ayarlar.

Xmas scan ( -sX ) ⇒ FIN, PSH ve URG bayraklarını ayarlayarak paketi bir Noel ağacı gibi aydınlatır.

Bu üç tarama türü, prob paketlerinde ayarlanan TCP bayrakları dışında davranış açısından tamamen aynıdır. Yanıtlar Tablo 5.4'te gösterildiği gibi ele alınır.

Tablo 5.4. Nmap bir NULL, FIN veya Xmas tarama probuna verilen yanıtları nasıl yorumlar?

Probe Response (Prob Yanımı)	Assigned State (Atanmış Durum)
Yanıt alınmadı (yeniden iletimlerden sonra bile)	<code>open filtered</code>
TCP RST packet	<code>closed</code>
ICMP ulaşılamıyor hatası (tip 3, kod 1, 2, 3, 9, 10 veya 13)	<code>filtered</code>

Bu tarama türlerinin en önemli avantajı, bazı durumsal olmayan güvenlik duvarlarından ve paket filtreleme yönlendiricilerinden gizlice geçebilmeleridir. Bu

tür güvenlik duvarları, SYN biti ayarlanmış ve ACK temizlenmiş tüm TCP paketlerini engelleyerek gelen TCP bağlantılarını önlemeye çalışır (gidenlere izin verirken). Bu yapılandırma o kadar yaygındır ki Linux iptables güvenlik duvarı komutu bunu uygulamak için özel bir --syn seçeneği sunar. NULL, FIN ve Xmas taramaları SYN bitini temizler ve böylece bu kuralların içinden geçer.

Diğer bir avantajı ise bu tarama türlerinin SYN taramasından bile biraz daha gizli olmasıdır. Yine de buna güvenmeyin - çoğu modern IDS ürünü bunları tespit edecek şekilde yapılandırılabilir.

En büyük dezavantajı, tüm sistemlerin RFC 793'ü harfiyen takip etmemesidir. Bazı sistemler, portun açık olup olmadığına bakmaksızın problara RST yanıtları gönderir. Bu da tüm portların kapalı olarak etiketlenmesine neden olur. Bunu yapan başlıca işletim sistemleri Microsoft Windows, birçok Cisco cihazı ve IBM OS/400'dür. Ancak bu tarama Unix tabanlı sistemlerin çoğuna karşı çalışır. Nmap OS tespiti bu tuhaflığı test ettiğinden, nmap-os-db dosyasını inceleyerek taramanın belirli bir sistem türüne karşı çalışıp çalışmadığını öğrenebilirsiniz. Test T2 açık bir porta NULL paket gönderir. Dolayısıyla, T2(R=N) gibi bir satır görürseniz, bu sistem RFC'yi destekliyor gibi görünür ve bu taramalardan biri ona karşı çalışmalıdır. T2 satırı daha uzunsa, sistem bir yanıt göndererek RFC'yi ihlal etmiştir ve bu taramalar çalışmamayacaktır. Bölüm 8, Uzak İşletim Sistemi Algılama, işletim sistemi parmak izini daha ayrıntılı olarak açıklamaktadır.

Bu taramaların bir başka dezavantajı da açık portları filtrelenmiş olanlardan ayırt edemeleridir. Paket filtresi bir ICMP hedef yasaklandı hatası gönderirse, Nmap bir portunfiltrelendiğini bilir. Ancak çoğu filtre herhangi bir yanıt vermeden yasaklı problemleri bırakır ve portların açık görünmesine neden olur. Nmap durumun hangisi olduğundan emin olamadığından, yanıt vermeyen portları açık|filtreli olarak işaretler. Sürüm tespiti (-sV) eklemek, UDP taramalarında olduğu gibi belirsizliği ortadan kaldırabilir, ancak bu, bu taramanın gizli doğasının çoğunu yener. Yine de bağlantı noktalarına bağlanmaya istekliySENİZ ve bağlanabiliyorsanız, bir SYN taraması da kullanabilirisiniz.

Bu tarama yöntemlerini kullanmak basittir. Tarama türünü belirtmek için -sN, -sF veya -sX seçeneklerini eklemeniz yeterlidir. Örnek 5.10 iki örnek göstermektedir. İlkı, Para'ya karşı bir FIN taraması, beş açık portun hepsini tanımlıyor (açık|filtreli olarak). Bir sonraki uygulama, scanme.nmap.org'a karşı bir Xmas taraması o kadar iyi çalışmıyor. Kapalı portu tespit ediyor, ancak 995 filtrelenmiş portu dört açık

porttan ayırt edemiyor, 999'unun tümü açık|filtrelenmiş olarak listeleniyor. Bu, Nmap'in neden bu kadar çok tarama yöntemi sunduğunu göstermektedir. Her durumda tek bir teknik tercih edilmez. Ereet'in Scanme hakkında daha fazla bilgi edinmek için başka bir yöntem denemesi gerekecektir.

#### Örnek 5.10. Örnek FIN ve Xmas taramaları

```
krad# nmap -sF -T4 para

Starting Nmap ( https://nmap.org )
Nmap scan report for para (192.168.10.191)
Not shown: 995 closed ports
PORT      STATE            SERVICE
22/tcp    open|filtered  ssh
53/tcp    open|filtered  domain
111/tcp   open|filtered  rpcbind
515/tcp   open|filtered  printer
6000/tcp  open|filtered  X11
MAC Address: 00:60:1D:38:32:90 (Lucent Technologies)

Nmap done: 1 IP address (1 host up) scanned in 4.64 seconds

krad# nmap -sX -T4 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 999 open|filtered ports
PORT      STATE  SERVICE
113/tcp  closed auth

Nmap done: 1 IP address (1 host up) scanned in 23.11 seconds
```

Bu taramaların güvenlik duvarını aşan tam gücünü göstermek için oldukça zayıf bir hedef güvenlik duvarı yapılandırması gereklidir. Ne yazık ki, bunları bulmak kolaydır. Örnek 5.11, Docsvr adlı bir SCO/Caldera makinesinin SYN taramasını göstermektedir.

#### Örnek 5.11. Docsvr'nin SYN taraması

```
# nmap -sS -T4 docsrv.caldera.com

Starting Nmap ( https://nmap.org )
Nmap scan report for docsrv.caldera.com (216.250.128.247)
(The 997 ports scanned but not shown below are in state: filtered)
PORT      STATE    SERVICE
80/tcp    open     http
113/tcp   closed   auth
507/tcp   open     crs

Nmap done: 1 IP address (1 host up) scanned in 28.62 seconds
```

Bu örnek iyi görünüyor. Sadece iki port açık ve geri kalanı (113 hariç) filtrelenmiş. Modern bir durum bilgisine sahip güvenlik duvarı ile FIN taraması herhangi bir ekstra bilgi üretmemelidir. Yine de Ereet bunu dener ve Örnek 5.12'deki çıktıyı elde eder.

Örnek 5.12. Docsvr'nin FIN taraması

```
# nmap -sF -T4 docsrv.caldera.com

Starting Nmap ( https://nmap.org )
Nmap scan report for docsrv.caldera.com (216.250.128.247)
Not shown: 961 closed ports
PORT      STATE     SERVICE
7/tcp      open|filtered echo
9/tcp      open|filtered discard
11/tcp     open|filtered systat
13/tcp     open|filtered daytime
15/tcp     open|filtered netstat
19/tcp     open|filtered chargen
21/tcp     open|filtered ftp
22/tcp     open|filtered ssh
23/tcp     open|filtered telnet
25/tcp     open|filtered smtp
37/tcp     open|filtered time
79/tcp     open|filtered finger
80/tcp     open|filtered http
110/tcp    open|filtered pop3
111/tcp    open|filtered rpcbind
135/tcp    open|filtered msrpc
143/tcp    open|filtered imap
360/tcp    open|filtered sco12odialog
389/tcp    open|filtered ldap
465/tcp    open|filtered smtps
507/tcp    open|filtered crs
512/tcp    open|filtered exec
513/tcp    open|filtered login
514/tcp    open|filtered shell
515/tcp    open|filtered printer
636/tcp    open|filtered ldapssl
712/tcp    open|filtered unknown
955/tcp    open|filtered unknown
993/tcp    open|filtered imaps
995/tcp    open|filtered pop3s
1434/tcp   open|filtered ms-sql-m
2000/tcp   open|filtered callbook
2766/tcp   open|filtered listen
3000/tcp   open|filtered ppp
3306/tcp   open|filtered mysql
6112/tcp   open|filtered dtspc
32770/tcp  open|filtered sometimes-rpc3
32771/tcp  open|filtered sometimes-rpc5
32772/tcp  open|filtered sometimes-rpc7

Nmap done: 1 IP address (1 host up) scanned in 7.64 seconds
```

Vay canına! Görünüşe göre çok fazla açık port var. Bunların çoğu muhtemelen açıktır, çünkü sadece bu 39'unun filtrelenmesi ve diğer 961'inin kapalı olması (bir RST paketi gönderilmesi) olağanışı olacaktır. Yine de bazlarının veya hepsinin açık yerine filtrelenmiş olması mümkündür. FIN taraması kesin olarak belirleyemez. Bu durumu tekrar gözden geçireceğiz ve bu bölümün ilerleyen kısımlarında Docsvr hakkında daha fazla bilgi edineceğiz.

## **Custom Scan Types with --scanflags (--scanflags ile Özel Tarama Türleri)**

Gerçekten gelişmiş Nmap kullanıcılarının kendilerini konserve taranan türlerle sınırlamalarına gerek yoktur. --scanflags seçeneği, keyfi TCP bayrakları belirterek kendi taramanızı tasarlamanıza olanak tanır. Saticıları sadece Nmap man sayfasını tarayarak belirli kurallar ekleyen saldırısı tespit sistemlerinden kaçarken, yaratıcılığınızı konuşturun!

--scanflags argümanı 9 (PSH ve FIN) gibi sayısal bir bayrak değeri olabilir, ancak sembolik isimler kullanmak daha kolaydır. URG, ACK, PSH, RST, SYN ve FIN'in herhangi bir kombinasyonunu bir araya getirin. Örneğin, --scanflags URGACKPSHRSTSYNFIN her şeyi ayarlar, ancak tarama için pek kullanışlı değildir. Bunların hangi sırada belirtildiği önemsizdir.

İstediğiniz bayrakları belirtmenin yanı sıra, bir TCP tarama türü de belirtebilirsiniz (-sA veya -sF gibi). Bu temel tür Nmap'e yanıtları nasıl yorumlayacağını söyler. Örneğin, bir SYN taraması yanıt vermemeyifiltrelenmiş bir portun göstergesi olarak kabul ederken, bir FIN taraması aynı şeyi açık|filtrelenmiş olarak değerlendirir. Nmap, temel tarama türü için yaptığı gibi davranışacaktır, ancak bunun yerine belirttiğiniz TCP bayraklarını kullanacaktır. Bir temel tür belirtmezseniz, SYN taraması kullanılır.

### **Custom SYN/FIN Scan (Özel SYN/FIN Taraması)**

İlginc bir özel tarama türü SYN/FIN'dir. Bazen bir güvenlik duvarı yöneticisi veya cihaz üreticisi gelen bağlantıları "sadece SYN bayrağı ayarlanmış gelen paketleri bırak" gibi bir kuralla engellemeye çalışır. Bunu sadece SYN bayrağıyla sınırlarlar çünkü giden bir bağlantının ikinci adımı olarak dönen SYN/ACK paketlerini engellemek istemezler.

Bu yaklaşımla ilgili sorun, çoğu uç sistemin diğer (ACK olmayan) bayrakları da içeren ilk SYN paketlerini kabul edecek olmasıdır. Örneğin, Nmap OS parmak izi sistemi açık bir porta SYN/FIN/URG/PSH paketi gönderir. Veritabanındaki parmak izlerinin yarısından fazlası SYN/ACK ile yanıt verir. Böylece bu paketle port taramasına izin verirler ve genellikle tam bir TCP bağlantısı kurmaya da izin verirler. Bazı sistemlerin SYN/RST paketine SYN/ACK ile yanıt verdiği bile

bilinmekteidir! TCP RFC, ilk SYN paketinde hangi bayrakların kabul edilebilir olduğu konusunda belirsizdir, ancak SYN/RST kesinlikle sahte görünmektedir.

Örnek 5.13, Ereet'in Google'da başarılı bir SYN/FIN taraması gerçekleştirdiğini göstermektedir. Görünüşe göre scanme.nmap.org'dan sıkılmış.

Örnek 5.13. Google'in SYN/FIN taraması

```
krad# nmap -sS --scanflags SYNFIN -T4 www.google.com

Starting Nmap ( https://nmap.org )
Warning: Hostname www.google.com resolves to 4 IPs. Using 74.125.19.99.
Nmap scan report for cf-in-f99.google.com (74.125.19.99)
Not shown: 996 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
113/tcp   closed auth
179/tcp   closed bgp
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 7.58 seconds
```

SYN/URG veya SYN/PSH/URG/FIN gibi benzer tarama türleri de genellikle işe yarayacaktır. Eğer ulaşamıyorsanız, daha önce bahsedilen SYN/RST seçeneğini unutmayın.

### PSH Scan

"TCP FIN, NULL ve Xmas Taramaları (-sF, -sN, -sX)" adlı bölümde RFC uyumlu sistemlerin FIN, PSH ve URG bayraklarının herhangi bir kombinasyonunu kullanarak portları taramaya izin verdiği belirtilmiştir. Sekiz olası permütasyon varken, Nmap yalnızca üç konserve modu (NULL, FIN ve Xmas) sunar. Bunun yerine PSH/URG veya FIN/PSH taramasını deneyerek biraz kişisel yetenek gösterin. Sonuçlar üç hazır moddan nadiren farklıdır, ancak tarama algılama sistemlerinden kaçmak için küçük bir şans vardır.

Böyle bir tarama gerçekleştirmek için, --scanflags ile istediğiniz bayrakları belirtin ve temel tür olarak FIN taramasını (-sF) belirtin (NULL veya Xmas seçmek fark yaratmaz). Örnek 5.14 yerel ağdaki bir Linux makineye karşı bir PSH taramasını göstermektedir.

Örnek 5.14. Özel bir PSH taraması

```
krad# nmap -sF --scanflags PSH para

Starting Nmap ( https://nmap.org )
Nmap scan report for para (192.168.10.191)
(The 995 ports scanned but not shown below are in state: closed)
PORT      STATE     SERVICE
22/tcp    open|filtered ssh
53/tcp    open|filtered domain
111/tcp   open|filtered rpcbind
515/tcp   open|filtered printer
6000/tcp  open|filtered X11
MAC Address: 00:60:1D:38:32:90 (Lucent Technologies)

Nmap done: 1 IP address (1 host up) scanned in 5.95 seconds
```

Bu taramaların hepsi aynı şekilde çalıştığından, -sF, -sN ve -sX seçeneklerinden sadece birini tutabilir ve kullanıcıların --scanflags ile diğerlerini taklit etmesine izin verebilirim. Kısayol seçeneklerini hatırlamak ve kullanmak daha kolay olduğu için bunu yapmayı planlamıyorum. Yine de Nmap becerilerinizi göstermek için taklit yaklaşımı deneyebilirsiniz. Daha sıradan olan nmap -sX hedefi yerine nmap -sF --scanflags FINPSHURG hedefini çalıştırın.

**uyarı :** Tecrübelerime göre, gereksiz yere karmaşık Nmap komut satırları kızları etkilemiyor. Genellikle küfürmeyici bir alay ile karşılık verirler, muhtemelen komutun gereksiz olduğunu kabul ederler.

## **TCP ACK Scan ( -sA ) (TCP ACK Taraması (-sA))**

Bu tarama, açık (hatta açık|filtrelenmiş) portları asla belirlemediği için şimdije kadar tartışılan diğerlerinden farklıdır. Güvenlik duvarı kural kümelerinin haritasını çıkarmak, durum bilgisi olup olmadıklarını ve hangi bağlantı noktalarının filtreldiğini belirlemek için kullanılır.

ACK taraması -sA seçeneği belirtilerek etkinleştirilir. Prob paketinde yalnızca ACK bayrağı ayarlanmıştır (--scanflags kullanmadığınız sürece). Filtrelenmemiş sistemleri tararken, açık ve kapalı portların her ikisi de bir RST paketi döndürecektil. Nmap daha sonra bunları filtrelenmemiş olarak etiketler, yani ACK paketi tarafından erişilebilirler, ancak açık mı yoksa kapalı mı oldukları

belirlenemez. Yanıt vermeyen veya belirli ICMP hata mesajlarını geri gönderen bağlantı noktaları filtrelenmiş olarak etiketlenir. Tablo 5.5 tüm ayrıntıları sağlar.

Tablo 5.5. Nmap bir ACK tarama probuna verilen yanıtları nasıl yorumlar?

Probe Response (Prob Yanımı)	Assigned State (Atanmış Durum)
TCP RST yanıtı	unfiltered
Yanıt alınmadı (yeniden iletişimlerden sonra bile)	filtered
ICMP ulaşılamıyor hatası (tip 3, kod 1, 2, 3, 9, 10 veya 13)	filtered

ACK tarama kullanımı diğer tarama türlerinin çoğu benzer, tek bir seçenek bayrağı (bu durumda -sA) eklemeniz yeterlidir. Örnek 5.15 Scanme'ye karşı bir ACK taramasını göstermektedir.

#### Örnek 5.15. Tipik bir ACK Taraması

```
krad# nmap -sA -T4 scanme.nmap.org
Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
Not shown: 994 filtered ports
PORT      STATE      SERVICE
22/tcp    unfiltered ssh
25/tcp    unfiltered smtp
53/tcp    unfiltered domain
70/tcp    unfiltered gopher
80/tcp    unfiltered http
113/tcp   unfiltered auth

Nmap done: 1 IP address (1 host up) scanned in 4.01 seconds
```

ACK taramasının en ilginç kullanımlarından biri durumlu ve durumsuz güvenlik duvarları arasında ayrim yapmaktadır. Bunu nasıl yapacağınızı ve neden yapmak isteyeceğinizi öğrenmek için "ACK Taraması" bölümüne bakın.

Bazen bir sistemden ekstra bilgi toplamak için tarama türlerinin bir kombinasyonu kullanılabilir. Örnek olarak, Örnek 5.12, "Docsvr'nin FIN taraması "nda Docsvr'nin FIN taramasını inceleyerek başlayın. Nmap bu durumda kapalı portları bulur, ancak 39 tanesi açık|filtreli olarak listelenir çünkü Nmap FIN taraması ile bu iki durum arasında karar veremez. Şimdi Örnek 5.16, "Docsvr'nin ACK taraması "nda aynı ana

bilgisayarın ACK taramasına bakın. Daha önce tanımlanmamış 39 bağlantı noktasından ikisinin filtrelenmiş olduğu gösterilmektedir. Diğer 37'si (tablonun üstündeki varsayılan bağlantı noktası satırına göre) filtrelenmemiş durumdadır. Bu açık ya da kapalı anlamına gelir. Bir tarama türü bir bağlantı noktasını açık ya da filtrelenmiş olarak tanımlarken diğeri açık ya da kapalı olarak tanımlarsa, mantık o bağlantı noktasının açık olması gerektiğini belirtir. Her iki tarama türünü birleştirerek Docsrv üzerindeki 37 portun açık, ikisinin filtrelenmiş ve 961'inin kapalı olduğunu öğrendik. Mantıksal çıkarım burada port durumlarını belirlemek için iyi çalışmış olsa da, bu teknigue her zaman güvenilemez. Farklı tarama türlerinin aynı port için her zaman tutarlı bir durum döndürdüğünü varsayar ki bu doğru değildir. Güvenlik duvarları ve TCP yiğini özellikleri, aynı makineye karşı farklı taramaların belirgin şekilde farklı olmasına neden olabilir. Docsrv'ye karşı, bir SYN taramasının SSH portunu (tcp/22) filtrelenmiş olarak değerlendirdiğini, bir ACK taramasının ise filtrelenmemiş olarak değerlendirdiğini gördük. Sınır koşullarını ve garip yapılandırılmış ağları keşfederken, Nmap sonuçlarını yorumlamak deneyim ve sezgiden yararlanan bir sanattır.

Örnek 5.16. Docsrv'nin ACK taraması

```
# nmap -sA -T4 docsrv.caldera.com

Starting Nmap ( https://nmap.org )
Nmap scan report for docsrv.caldera.com (216.250.128.247)
Not shown: 998 unfiltered ports
PORT      STATE    SERVICE
135/tcp   filtered msrpc
1434/tcp  filtered ms-sql-m

Nmap done: 1 IP address (1 host up) scanned in 7.20 seconds
```

## TCP Window Scan ( -sW ) (TCP Pencere Taraması (-sW))

Pencere taraması ACK taramasıyla tamamen aynıdır, ancak bir RST döndürüldüğünde her zaman filtrelenmemiş olarak yazdırmak yerine, açık bağlantı noktalarını kapalı olanlardan ayırmak için belirli sistemlerin bir uygulama ayrıntısından yararlanır. Bunu, döndürülen RST paketlerinin TCP Pencere değerini

inceleyerek yapar. Bazı sistemlerde, açık portlar pozitif bir pencere boyutu kullanırken (RST paketleri için bile), kapalı olanlar sıfır pencereye sahiptir. Pencere taraması, ACK taraması ile aynı çiplak ACK probunu gönderir ve sonuçları Tablo 5.6'da gösterildiği gibi yorumlar.

Tablo 5.6. Nmap, Pencere taraması ACK probuna verilen yanıtları nasıl yorumlar?

Probe Response (Prob Yanıtı)	Assigned State (Atanmış Durum)
TCP RST response with non-zero window field	open
TCP RST response with zero window field	closed
No response received (even after retransmissions)	filtered
ICMP unreachable error (type 3, code 1, 2, 3, 9, 10, or 13)	filtered

Bu tarama internetteki az sayıda sistemin uygulama detayına dayanır, bu nedenle her zaman güvenemezsınız. Bunu desteklemeyen sistemler genellikle tüm portları kapalı olarak döndürecektil. Elbette, makinenin gerçekten hiç açık portu olmaması mümkün değildir. Taranan portların çoğu kapalıysa ancak birkaç yaygın port numarası (22, 25 ve 53 gibi) açıksa, sistem büyük olasılıkla hassastır. Bazen sistemler tam tersi bir davranış bile gösterebilir. Taramanız 997 açık bağlantı noktası ve üç kapalı veya filtrelenmiş bağlantı noktası gösteriyorsa, bu üçü gerçekten açık olanlar olabilir.

Bu tarama her durum için uygun olmasa da, zaman zaman oldukça faydalı olabilir. Örnek 5.12'yi hatırlayın, "Docsvr'nin FIN taraması", temel SYN taramasında bulunmayan birçok açık|filtrelenmiş portu gösterir. Sorun, bu FIN taraması ile açık ve filtrelenmiş portları ayırt edememizdir. Önceki bölümde FIN ve ACK tarama sonuçlarını birleştirerek bunları ayırt edebileceğimizi göstermiştık. Bu durumda, Örnek 5.17'de gösterildiği gibi, bir Pencere taraması FIN tarama sonuçlarını gerektirmeyerek işi daha da kolaylaştırır.

Örnek 5.17. docsvr.caldera.com'un pencere taraması

```
# nmap -sW -T4 docsrv.caldera.com

Starting Nmap ( https://nmap.org )
Nmap scan report for docsrv.caldera.com (216.250.128.247)
Not shown: 961 closed ports
PORT      STATE    SERVICE
7/tcp      open     echo
9/tcp      open     discard
11/tcp     open     systat
13/tcp     open     daytime
15/tcp     open     netstat
19/tcp     open     chargen
21/tcp     open     ftp
22/tcp     open     ssh
23/tcp     open     telnet
25/tcp     open     smtp
37/tcp     open     time
79/tcp     open     finger
80/tcp     open     http
110/tcp    open     pop3
111/tcp    open     rpcbind
135/tcp    filtered msrpc
[14 open ports omitted for brevity]
1434/tcp   filtered ms-sql-m
2000/tcp   open     callbook
2766/tcp   open     listen
3000/tcp   open     ppp
3306/tcp   open     mysql
6112/tcp   open     dtspc
32770/tcp  open     sometimes-rpc3
32771/tcp  open     sometimes-rpc5
32772/tcp  open     sometimes-rpc7

Nmap done: 1 IP address (1 host up) scanned in 7.30 seconds
```

Bu sonuçlar tam olarak Ereet'in istediği şey! FIN taramasında olduğu gibi aynı 39 ilginç bağlantı noktası gösterilir, ancak bu sefer filtrelenmiş iki bağlantı noktası (MS-SQL ve MSRPC) ile gerçekten açık olan 37 bağlantı noktası arasında ayrim yapar. Bunlar Ereet'in önceki bölümde FIN ve ACK tarama sonuçlarını bir araya getirerek elde ettiği sonuçlarla aynıdır. Sonuçların tutarlılık açısından doğrulanması, bir hedef ağa karşı birden fazla tarama türünü denemek için bir başka iyi nedendir.

## **TCP Maimon Scan ( -sM ) (TCP Maimon Taraması (-sM))**

Maimon taraması, adını keşfeden Uriel Maimon'dan almıştır. Bu teknigi Phrack Magazine'in 49. sayısında (Kasım 1996) tanımlamıştır. Bu teknigi içeren Nmap iki sayı sonra yayınlandı. Bu teknik NULL, FIN ve Xmas taraması ile tamamen aynıdır, tek farkı probun FIN/ACK olmasıdır. RFC 793'e (TCP) göre, port açık ya da kapalı olsun, böyle bir proba yanıt olarak bir RST paketi oluşturulmalıdır. Ancak Uriel, birçok BSD türevi sistemin bağlantı noktası açıksa paketi bıraktığını fark etmiştir. Nmap, Tablo 5.7'de gösterildiği gibi açık portları belirlemek için bundan yararlanır.

Tablo 5.7. Nmap, Maimon tarama probuna verilen yanitları nasıl yorumlar?

Probe Response (Prob Yanıtı)	Assigned State (Atanmış Durum)
Yanıt alınmadı (yeniden iletimlerden sonra bile)	open filtered
TCP RST packet	closed
ICMP ulaşılamıyor hatası (tip 3, kod 1, 2, 3, 9, 10 veya 13)	filtered

Maimon taraması için Nmap bayrağı -sM'dir. Bu seçenek 1996 yılında oldukça kullanışlı olsa da, modern sistemler bu hatayı nadiren sergiler. Tüm portlar için bir RST geri göndererek her portun kapalı görünmesini sağlarlar. Bu sonuç Örnek 5.18'de gösterilmektedir.

Örnek 5.18. Başarısız bir Maimon taraması

```
# nmap -sM -T4 para
Starting Nmap ( https://nmap.org )
All 1000 scanned ports on para (192.168.10.191) are: closed
MAC Address: 00:60:1D:38:32:90 (Lucent Technologies)

Nmap done: 1 IP address (1 host up) scanned in 4.19 seconds
```

## TCP Idle Scan ( -sl ) (TCP Boşta Tarama (-sl))

1998 yılında güvenlik araştırmacısı Antirez (bu kitabın bazı bölümlerinde kullanılan hping2 aracını da yazmıştır) Bugtraq posta listesine dahiyane bir yeni port tarama teknigi göndermiştir. Idle scan, bilindiği gibi, tamamen kör port taramasına izin

verir. Saldırganlar aslında kendi IP adreslerinden hedefe tek bir paket göndermeden bir hedefi tarayabilirler! Bunun yerine, akıllıca bir yan kanal saldırısı, taramanın aptal bir "zombi ana bilgisayardan" sekmesini sağlar. Saldırı tespit sistemi (IDS) raporları masum zombiyi saldırın olarak işaretleyecektir. Olağanüstü derecede gizli olmasının yanı sıra, bu tarama türü makineler arasındaki IP tabanlı güven ilişkilerinin keşfedilmesine izin verir.

Boşta tarama şu ana kadar tartışılan tekniklerden daha karmaşık olsa da, bunu anlamak için TCP/IP uzmanı olmanız gereklidir. Bu temel gerçeklerden bir araya getirilebilir:

- Bir TCP portunun açık olup olmadığını belirlemenin bir yolu, porta bir SYN (oturum kurma) paketi göndermektir. Hedef makine, bağlantı noktası açıksa bir SYN/ACK (oturum isteği onayı) paketiyle, bağlantı noktası kapalıysa RST (sıfırlama) ile yanıt verecektir. Bu, daha önce tartışılan SYN taramasının temelidir.
- İstenmeyen bir SYN/ACK paketi alan bir makine RST ile yanıt verecektir. İstenmeyen bir RST yok sayılacaktır.
- Internet üzerindeki her IP paketinin bir parça tanımlama numarası (IP ID) vardır. Birçok işletim sistemi gönderdikleri her paket için bu numarayı artırdığından, IPID'yi araştırmak bir saldırana son araştırmadan bu yana kaç paket gönderildiğini söyleyebilir.

### **Idle Scan Step by Step (Adım Adım Rölatif Taraması)**

Temel olarak, boşta tarama her bir port için tekrarlanan üç adımdan oluşur:

1. Zombinin IP kimliğini araştırın ve kaydedin.
2. Zombiden bir SYN paketi oluşturun ve hedefte istenen bağlantı noktasına gönderin. Port durumuna bağlı olarak, hedefin tepkisi zombinin IP ID'sinin artmasına neden olabilir veya olmayabilir.
3. Zombinin IP kimliğini tekrar araştırın. Hedef bağlantı noktasının durumu, bu yeni IP kimliği ile 1. adımda kaydedilen IP kimliği karşılaştırılarak belirlenir.

Bu işlemden sonra, zombinin IP kimliği bir ya da iki artmış olmalıdır. Bir artış, zombinin saldırının sondasına verdiği yanıt dışında herhangi bir paket göndermediğini gösterir. Gönderilen paketlerin olmaması, bağlantı noktasının açık olmadığı anlamına gelir (hedef, zombiye ya yok sayılan bir RST paketi göndermiş

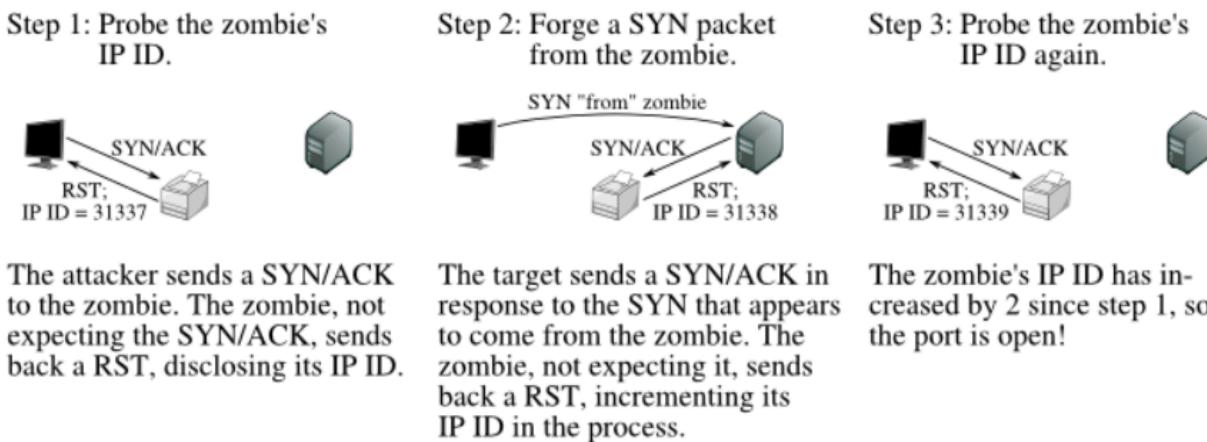
ya da hiçbir şey göndermemiş olmalıdır). İki artış, zombinin iki sonda arasında bir paket gönderdiğini gösterir. Bu ekstra paket genellikle bağlantı noktasının açık olduğu anlamına gelir (hedef muhtemelen sahte SYN'e yanıt olarak zombiye bir SYN/ACK paketi göndermiş ve bu da zombiden bir RST paketi gelmesine neden olmuştur). İkiden büyük artışlar genellikle kötü bir zombi ana bilgisayara işaret eder. Tahmin edilebilir IP kimlik numaralarına sahip olmayabilir veya boşta tarama ile ilgisi olmayan bir iletişimle meşgul olabilir.

Kapalı bir portta olan şey filtrelenmiş bir portta olandan biraz farklı olsa da, saldırgan her iki durumda da aynı sonucu, yani 1 IP ID artışını ölçer. Bu nedenle, boşta taramanın kapalı ve filtrelenmiş portlar arasında ayrılmaması mümkün değildir. Nmap 1'lük bir IP ID artışı kaydettiğinde portu kapalı|filtrelenmiş olarak işaretler.

Daha fazla ayrıntı isteyenler için, aşağıdaki üç diyagram açık, kapalı ve filtrelenmiş liman durumlarında tam olarak ne olduğunu göstermektedir. Her birindeki aktörler şunlardır:

 the attacker,  the zombie, and  the target.

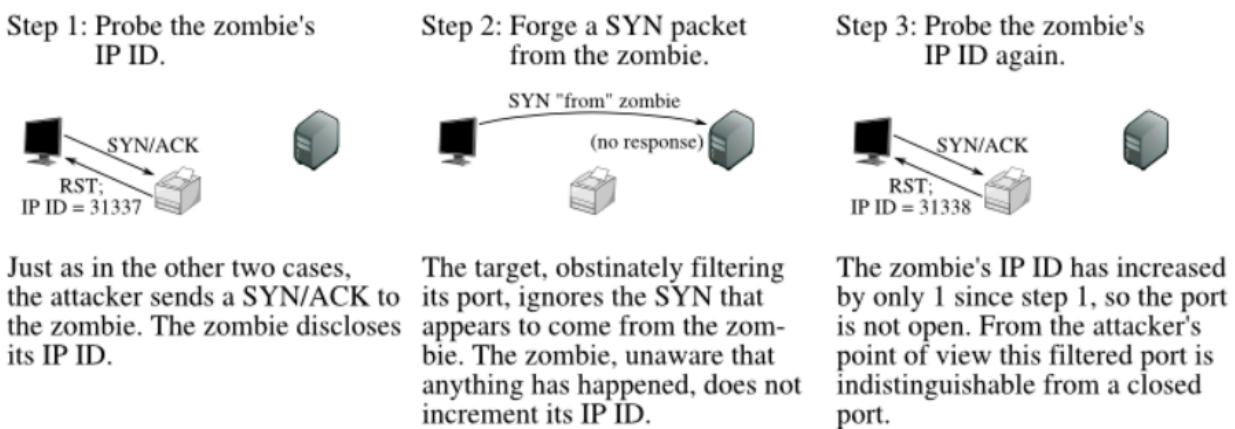
Şekil 5.6. Açık bir portun boşta taranması



Şekil 5.7. Kapalı bir portun boşta taranması



Şekil 5.8. Filtrelenmiş bir bağlantı noktasının boşta taraması



Boşta tarama en üst düzey gizli taramadır. Nmap, kullanıcıların kimliklerini korumalarına yardımcı olmak için tuzak tarama (-D) sunar, ancak bu (boşta taramanın aksine) yine de bir saldırganın tarama sonuçlarını geri almak için gerçek IP adresinden hedefe bazı paketler göndermesini gerektirir. Boşta taramanın bir sonucu da saldırısı tespit sistemlerinin genellikle zombi makinenin kendilerine karşı bir tarama başlattığını iddia eden uyarılar göndermesidir. Böylece başka bir tarafı tarama için tuzağa düşürmek için kullanılabilir. IDS'nizden gelen uyarıları okurken bu olasılığı aklınızda bulundurun.

Boşta taramanın benzersiz bir avantajı, belirli paket filtreleme güvenlik duvarlarını ve yönlendiricileri yenmek için kullanılabilmesidir. IP kaynak adresi filtreleme, hassas bir ana bilgisayara veya ağa bağlanabilecek makineleri sınırlamak için yaygın (zayıf olsa da) bir güvenlik mekanizmasıdır. Örneğin, bir şirket veritabanı sunucusu yalnızca kendisine erişen genel web sunucusundan gelen bağlantılarla

izin verebilir. Ya da bir ev kullanıcısı sadece kendi iş makinelerinden SSH (etkileşimli oturum açma) bağlantılarına izin verebilir.

Daha rahatsız edici bir senaryo, bazı şirket kodamanları ağ yöneticilerinden bir güvenlik duvarı deliği açmalarını talep ettiğinde ortaya çıkar, böylece ev IP adresinden dahili ağ kaynaklarına erişebilir. Bu durum, yöneticiler güvenli VPN alternatiflerini kullanmak istemediğinde veya kullanamadığında meydana gelebilir.

Boşta tarama bazen bu güven ilişkilerinin haritasını çıkarmak için kullanılabilir. Kilit faktör, boşta tarama sonuçlarının zombi ana bilgisayarın bakış açısından açık bağlantı noktalarını listelemesidir. Yukarıda bahsedilen veritabanı sunucusuna karşı yapılan normal bir taramada açık port bulunmayabilir, ancak web sunucusunun IP'sini zombi olarak kullanırken boşta bir tarama yapmak, veritabanıyla ilgili hizmet portlarını açık olarak göstererek güven ilişkisini ortaya çıkarabilir.

Bu güven ilişkilerinin haritasını çıkarmak saldırganlar için hedefleri önceliklendirmede çok faydalı olabilir. Yukarıda ele alınan web sunucusu, özel veritabanı erişimini fark edene kadar bir saldırgan için sıradan görünebilir.

Boşta taramanın bir dezavantajı, diğer tarama türlerinin çoğudan çok daha uzun sürmesidir. "Boşta Tarama Uygulama Algoritmaları" adlı bölümde açıklanan optimize algoritmala rağmen, 15 saniyelik bir SYN taraması boşta tarama olarak 15 dakika veya daha fazla sürebilir. Bir başka sorun da paketleri zombiden geliyormuş gibi gösterip hedef makineye ulaşmasını sağlayabilmeniz gerektidir. Birçok İSS (özellikle çevirmeli bağlantı ve ev tipi geniş bant sağlayıcıları) artık bu tür paket sahteciliğini önlemek için çıkış filtrelemesi uygulamaktadır. Daha üst düzey sağlayıcıların (ortak yerleşim ve T1 hizmetleri gibi) bunu yapma olasılığı çok daha düşüktür. Bu filtreleme yürürlükteyse, Nmap denediğiniz her zombi için hızlı bir hata mesajı yazdıracaktır. İSS'leri değiştirmek bir seçenek değilse, aynı İSS ağında başka bir IP kullanmayı deneyebilirsiniz. Bazen filtreleme yalnızca müşteriler tarafından kullanılan aralığın dışındaki IP adreslerinin sahteciliğini engeller. Boşta tarama ile ilgili bir diğer zorluk, bir sonraki bölümde açıklandığı gibi çalışan bir zombi ana bilgisayar bulmanız gerektidir.

### **Finding a Working Idle Scan Zombie Host (Çalışan Bir Boşta Tarama Zombi Ana Bilgisayarı Bulma)**

Bir IP ID boşta taraması yürütmenin ilk adımı uygun bir zombi bulmaktır. IP ID paketlerini global olarak (iletisim kurduğu ana bilgisayar başına değil) kademeli olarak ataması gereklidir. Dışarıdan gelen trafik IP ID sırasını yükseltip tarama

mantığını karıştıracağından boşta olmalıdır (tarama adı da buradan gelmektedir). Saldırgan ile zombi arasındaki ve zombi ile hedef arasındaki gecikme ne kadar düşük olursa, tarama o kadar hızlı ilerleyecektir.

Boşta bir tarama denendiğinde, Nmap önerilen zombiyi test eder ve onunla ilgili herhangi bir sorunu bildirir. Eğer biri çalışmazsa, diğerini deneyin. Zombi adaylarını bulmak zor olmayacak kadar çok sayıda İnternet ana bilgisayarı savunmasızdır. Ana bilgisayarların boşta olması gerekiğinden, www.yahoo.com veya google.com gibi iyi bilinen bir ana bilgisayar seçmek neredeyse hiçbir zaman işe yaramayacaktır.

Yaygın bir yaklaşım, bir ajan Nmap ping taramasını yürütmektir. Nmap'in rastgele IP seçim modunu (-iR) kullanabilirsiniz, ancak bu muhtemelen önemli gecikme süresine sahip uzak zombilerle sonuçlanacaktır. Kaynak adresinize veya hedefe yakın bir ağ seçmek daha iyi sonuçlar verir. Çalışan bir tane bulana kadar ping tarama sonuçlarındaki her kullanılabilir ana bilgisayarı kullanarak boşta tarama yapmayı deneyebilirsiniz. Her zaman olduğu gibi, boşta tarama gibi beklenmedik amaçlar için birinin makinelerini kullanmadan önce izin istemek en iyisidir.

Çizimlerimizde zombiyi temsil etmesi için bir yazıcı simgesini sadece komik olsun diye seçmedik; basit ağ cihazları genellikle hem az kullanıldıkları (atılı durumda oldukları) hem de IP ID trafik tespiti konusunda savunmasız olan basit ağ yığınlarıyla inşa edildikleri için harika zombiler oluştururlar.

Sadece bir ping taraması yerine zombi adayı ağıda bir port taraması ve işletim sistemi tanımlaması (-O) yapmak iyi bir zombi seçmeye yardımcı olur. Verbose modu (-v) etkin olduğu sürece, OS tespiti genellikle IP ID sırası oluşturma yöntemini belirleyecek ve "IP ID Sırası Oluşturma: Artımlı" gibi bir satır yazdırır. Eğer tip Incremental veya Broken little-endian incremental olarak verilmişse, makine iyi bir zombi adayıdır. Solaris ve diğer bazı sistemler iletişim kurdukları her ana bilgisayar için yeni bir IP ID dizisi oluşturduğundan, bu yine de çalışacağının garantisini değildir. Ana bilgisayar çok meşgul de olabilir. İşletim sistemi tespiti ve açık port listesi de boşta olması muhtemel sistemlerin belirlenmesine yardımcı olabilir.

Zombi adaylarını belirlemek için bir başka yaklaşım da ipidseq NSE betiğini bir ana bilgisayara karşı çalıştırmaktadır. Bu betik, IP kimliği oluşturma yöntemini sınıflandırmak için bir ana bilgisayarı araştırır, ardından işletim sistemi algılamasının yaptığı gibi IP kimliği sınıflandırmasını yazdırır. Çoğu NSE betiği gibi ipidseq.nse de

birçok ana bilgisayara karşı paralel olarak çalıştırılabilir, bu da onu uygun ana bilgisayarları aramak için tüm ağları tararken başka bir iyi seçim haline getirir.

Uygun bir zombi belirlemek başlangıçta biraz çalışma gerektirse de, iyi olanları yeniden kullanmaya devam edebilirsiniz.

### Executing an Idle Scan (Boşta Tarama Yürütme)

Uygun bir zombi bulunduğuanda, tarama yapmak kolaydır. Basitçe -sl seçeneğine zombi ana bilgisayar adını belirtin ve gerisini Nmap hallede. Örnek 5.19, Ereet'in Kiosk adlı bir Adobe makinesinden boşta bir tarama sıçratarak Recording Industry Association of America'yı taramasının bir örneğini göstermektedir.

Örnek 5.19. RIAA'ya karşı boşta bir tarama

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

Starting Nmap ( https://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
```

Yukarıdaki taramadan, RIAA'nın güvenlik konusunda çok bilinçli olmadığını öğreniyoruz (açık PC Anywhere, portmapper ve Legato nsrexec portlarına dikkat edin). Görünüşe göre güvenlik duvarları olmadığından, bir IDS'ye sahip olmaları pek olası değil. Ancak varsa, tarama suçlusu olarak kiosk.adobe.com'u gösterecektir. Pn seçeneği Nmap'in RIAA makinesine ilk ping paketini göndermesini engeller. Bu Ereet'in gerçek adresini ortaya çıkarabilirdi. Tarama uzun sürdü çünkü -p- tüm 65K portlarını taramak için belirtilmişti. Taramalarınız için kiosk kullanmaya çalışmayın, çünkü zaten kaldırılmıştır.

Varsayılan olarak, Nmap zombinin 80 numaralı kaynak portundan hedefe problemler oluşturur. Zombi adına iki nokta üst üste ve port numarası ekleyerek farklı bir port seçebilirsiniz (örneğin -sl kiosk.adobe.com:113). Seçilen bağlantı noktası saldırıcı veya hedef tarafından filtrelenmemelidir. Zombinin SYN taraması portu açık veya kapalı durumda göstirmelidir.

### **Idle Scan Implementation Algorithms (Boşta Tarama Uygulama Algoritmaları)**

"Adım Adım Boşta Tarama" adlı bölüm temel düzeyde boşta taramayı açıklarken, Nmap uygulaması çok daha karmaşıktır. Temel farklar, hızlı yürütme için paralellik ve yanlış pozitifleri azaltmak için fazlalıktır.

Boşta taramayı paralelleştirmek, port durumlarını çıkarmanın dolaylı yöntemi nedeniyle diğer tarama tekniklerinden daha zordur. Nmap hedef üzerindeki birçok porta prob gönderir ve ardından zombinin yeni IP ID değerini kontrol ederse, IP ID artışlarının sayısı kaç hedef portun açık olduğunu ortaya çıkarır, ancak hangilerinin açık olduğunu göstermez. Bu aslında büyük bir sorun değildir, çünkü büyük bir taramada portların büyük çoğunluğu kapalı/filtrelenmiş olacaktır. Sadece açık portlar IP ID değerinin artmasına neden olduğundan, Nmap araya giren artışları görmez ve tüm port grubunu kapalı/filtreli olarak işaretleyebilir. Nmap paralel olarak 100 porta kadar olan grupları tarayabilir. Nmap bir grubu taradıktan sonra zombi IP ID'sinin  $<N>$  kez arttığını tespit ederse, bu grupta  $<N>$  açık port olmalıdır. Nmap daha sonra ikili bir arama ile açık portları bulur. Grubu ikiye böler ve her birine ayrı ayrı prob gönderir. Bir alt grup sıfır açık bağlantı noktası gösteriyorsa, o grubun tüm bağlantı noktaları kapalı/filtreli olarak işaretlenir. Bir alt grup bir veya daha fazla açık bağlantı noktası gösteriyorsa, tekrar bölünür ve işlem bu bağlantı noktaları tanımlanana kadar devam eder. Bu teknik karmaşıklığı artırırsa da, tarama sürelerini bir seferde sadece bir bağlantı noktasını taramaya göre büyülüük sırasına göre azaltabilir.

Güvenilirlik bir diğer önemli boşta tarama sorunudur. Eğer zombi konak tarama sırasında ilgisiz makinelere paket gönderirse, IP ID'si artar. Bu da Nmap'in açık bir port bulduğunu düşünmesine neden olur. Neyse ki, paralel tarama burada da yardımcı olur. Eğer Nmap bir gruptaki 100 portu tararsa ve IP ID artışı iki açık porta işaret ederse, Nmap grubu iki elli portluk alt gruba böler. Nmap her iki alt grupta da IP ID taraması yaptığından, toplam zombi IP ID artışı yine iki olsa iyi olur! Aksi takdirde, Nmap tutarsızlığı tespit edecek ve grupları yeniden tarayacaktır. Ayrıca, zombinin tespit edilen güvenilirlik oranına göre grup boyutunu ve tarama

zamanlamasını değiştirir. Nmap çok fazla tutarsız sonuç tespit ederse, bırakır ve kullanıcıdan daha iyi bir zombi sağlamasını ister.

Bazen bir paket izi, bu gibi karmaşık algoritmaları ve teknikleri anlamanın en iyi yoludur. Bir kez daha, Nmap --packet-trace, istendiğinde bunları üretmeyi önemsiz hale getirir. Bu bölümün geri kalanında gerçek bir yedi portlu boşta taramanın açıklamalı paket izi verilmektedir. IP adresleri Saldırgan, Zombi ve Hedef olarak değiştirilmiş ve iz satırlarının bazı alakasız yönleri (TCP pencere boyutu gibi) netlik için kaldırılmıştır.

```
Attacker# nmap -sI Zombie -Pn -p20-25,110 -r --packet-trace -v Target
Starting Nmap ( https://nmap.org )
```

-Pn gizlilik için gereklidir, aksi takdirde Saldırganın gerçek adresinden hedefe ping paketleri gönderilir. Sürüm taraması da gerçek adresi açığa çıkaracaktır ve bu nedenle -sV belirtilmemiştir. r seçeneği (bağlantı noktası rastgeleleştirmesini kapatır) yalnızca bu örneği takip etmeyi kolaylaştırmak için kullanılır.

Nmap ilk olarak Zombi'ye altı SYN/ACK paketi göndererek ve yanıtları analiz ederek Zombi'nin IP ID dizisi oluşturmasını test eder. Bu, Nmap'in kötü zombileri hemen ayıklamasına yardımcı olur. Ayrıca bazı sistemler (genellikle Microsoft Windows makineleri, ancak tüm Windows kutuları bunu yapmaz) gönderilen her paket için IP kimliğini bir yerine 256 artırdığı için de gereklidir. Bu, IP kimliğini ağ bayt sırasına (big-endian) dönüştürmedikleri zaman little-endian makinelerde olur. Nmap bu sorunu tespit etmek ve çözmek için bu ilk problemleri kullanır.

```
SENT (0.0060s) TCP Attacker:51824 > Zombie:80 SA id=35996
SENT (0.0900s) TCP Attacker:51825 > Zombie:80 SA id=25914
SENT (0.1800s) TCP Attacker:51826 > Zombie:80 SA id=39591
RCVD (0.1550s) TCP Zombie:80 > Attacker:51824 R id=15669
SENT (0.2700s) TCP Attacker:51827 > Zombie:80 SA id=43604
RCVD (0.2380s) TCP Zombie:80 > Attacker:51825 R id=15670
SENT (0.3600s) TCP Attacker:51828 > Zombie:80 SA id=34186
RCVD (0.3280s) TCP Zombie:80 > Attacker:51826 R id=15671
SENT (0.4510s) TCP Attacker:51829 > Zombie:80 SA id=27949
RCVD (0.4190s) TCP Zombie:80 > Attacker:51827 R id=15672
RCVD (0.5090s) TCP Zombie:80 > Attacker:51828 R id=15673
RCVD (0.5990s) TCP Zombie:80 > Attacker:51829 R id=15674
Idlescan using zombie Zombie (Zombie:80); Class: Incremental
```

Bu test zombinin iyi çalıştığını gösteriyor. Her IP kimliği bir öncekine göre bir artış göstermiştir. Yani sistem boşta ve IP ID trafiği tespitine karşı savunmasız görünüyor. Bu umut verici sonuçlar yine de Nmap'in Zombi'ye Hedef'ten gelmiş gibi dört paket gönderdiği bir sonraki teste tabidir. Ardından IP kimliğinin arttığından emin olmak için zombiyi araştırır. Artmadıysa, saldırganın İSS'si sahte paketleri engelliyor ya da zombi iletişim kurduğu her ana bilgisayar için ayrı bir IP Kimliği sıra sayacı kullanıyor olabilir. Her ikisi de sık karşılaşılan durumlardır, bu nedenle Nmap her zaman bu testi gerçekleştirir. En son bilinen Zombi IP kimliği yukarıda gösterildiği gibi 15674 idi.

```
SENT (0.5990s) TCP Target:51823 > Zombie:80 SA id=1390
SENT (0.6510s) TCP Target:51823 > Zombie:80 SA id=24025
SENT (0.7110s) TCP Target:51823 > Zombie:80 SA id=15046
SENT (0.7710s) TCP Target:51823 > Zombie:80 SA id=48658
SENT (1.0800s) TCP Attacker:51987 > Zombie:80 SA id=27659
RCVD (1.2290s) TCP Zombie:80 > Attacker:51987 R id=15679
```

Dört sahte paket ve Saldırgan'ın sondası Zombi'nin IP kimliğini 15674'ten 15679'a yükseltmesine neden oldu. Mükemmel! Şimdi gerçek tarama başlıyor. 15679'un en son Zombi IP kimliği olduğunu unutmayın.

```
Initiating Idlescan against Target
SENT (1.2290s) TCP Zombie:80 > Target:20 S id=13200
SENT (1.2290s) TCP Zombie:80 > Target:21 S id=3737
SENT (1.2290s) TCP Zombie:80 > Target:22 S id=65290
SENT (1.2290s) TCP Zombie:80 > Target:23 S id=10516
SENT (1.4610s) TCP Attacker:52050 > Zombie:80 SA id=33202
RCVD (1.6090s) TCP Zombie:80 > Attacker:52050 R id=15680
```

Nmap 20-23 numaralı bağlantı noktalarını denetler. Ardından Zombie'yi araştırır ve yeni IP kimliğinin 15680 olduğunu, önceki değer olan 15679'dan yalnızca bir yüksek olduğunu bulur. Bu iki bilinen paket arasında IP ID artışı olmamıştır, yani 20-23 numaralı portlar muhtemelen kapalıdır/filtrelenmemiştir. Bir Hedef bağlantı noktasından bir SYN/ACK'nin henüz gelmemiş olması da mümkündür. Bu durumda, Zombie bir RST ile yanıt vermemiştir ve bu nedenle IP Kimliği artmamıştır. Doğruluğu sağlamak için, Nmap bu portları daha sonra tekrar deneyecektir.

```
SENT (1.8510s) TCP Attacker:51986 > Zombie:80 SA id=49278
RCVD (1.9990s) TCP Zombie:80 > Attacker:51986 R id=15681
```

Nmap tekrar prob gönderir çünkü son gönderdiği probdan bu yana saniyenin onda dördü geçmiştir. Zombi (gerçekten boşta değilse) bu süre zarfında diğer ana bilgisayarlarla iletişim kurmuş olabilir, bu da burada tespit edilmezse daha sonra yanlışlıklara neden olabilir. Neyse ki bu gerçekleşmedi: bir sonraki IP kimliği bekleniği gibi 15681'dir.

```
SENT (2.0000s) TCP Zombie:80 > Target:24 S id=23928
SENT (2.0000s) TCP Zombie:80 > Target:25 S id=50425
SENT (2.0000s) TCP Zombie:80 > Target:110 S id=14207
SENT (2.2300s) TCP Attacker:52026 > Zombie:80 SA id=26941
RCVD (2.3800s) TCP Zombie:80 > Attacker:52026 R id=15684
```

Nmap 24, 25 ve 110 numaralı bağlantı noktalarını araştırır ve ardından Zombi IP Kimliğini sorgular. 15681'den 15684'e atladı. 15682 ve 15683'ü atladı, bu da bu üç bağlantı noktasından ikisinin muhtemelen açık olduğu anlamına geliyor. Nmap hangi ikisinin açık olduğunu söyleyemez ve bu bir yanlış pozitif de olabilir. Bu yüzden Nmap taramayı alt gruplara bölerek daha derine iner.

```
SENT (2.6210s) TCP Attacker:51867 > Zombie:80 SA id=18869
RCVD (2.7690s) TCP Zombie:80 > Attacker:51867 R id=15685
SENT (2.7690s) TCP Zombie:80 > Target:24 S id=30023
SENT (2.7690s) TCP Zombie:80 > Target:25 S id=47253
SENT (3.0000s) TCP Attacker:51979 > Zombie:80 SA id=12077
RCVD (3.1480s) TCP Zombie:80 > Attacker:51979 R id=15687
```

İlk alt grup 24 ve 25 numaralı bağlantı noktalarıdır. IP ID 15685'ten 15687'ye atlıyor, yani bu iki porttan biri büyük olasılıkla açık. Nmap böl ve yönet yaklaşımını tekrar deneyerek her bir portu ayrı ayrı araştırır.

```
SENT (3.3910s) TCP Attacker:51826 > Zombie:80 SA id=32515
RCVD (3.5390s) TCP Zombie:80 > Attacker:51826 R id=15688
SENT (3.5390s) TCP Zombie:80 > Target:24 S id=47868
SENT (3.7710s) TCP Attacker:52012 > Zombie:80 SA id=14042
RCVD (3.9190s) TCP Zombie:80 > Attacker:52012 R id=15689
```

24 numaralı bağlantı noktası araştırması IP kimliğinde bir sızrama göstermiyor. Yani bu port açık değil. Şimdiye kadarki sonuçlardan, Nmap geçici olarak belirledi:

- 20-23 numaralı bağlantı noktaları kapalı|filtreli
- 24, 25 ve 110 numaralı bağlantı noktalarından ikisi açıktır
- 24 ve 25 numaralı bağlantı noktalarından biri açık
- Bağlantı noktası 24 kapalı|filtrelenmiş

Bu bulmacaya yeterince uzun süre baktığınızda tek bir çözüm bulacaksınız: 25 ve 110 numaralı portlar açık, diğer beşi ise kapalı/filtreli. Bu mantığı kullanarak, Nmap artık taramayı durdurabilir ve sonuçları yazdırabilir. Eskiden bunu yapıyordu, ancak bu, Zombi gerçekten boşta olmadığında çok fazla yanlış pozitif açık bağlantı noktası üretti. Bu yüzden Nmap sonuçlarını doğrulamak için taramaya devam ediyor:

```
SENT (4.1600s) TCP Attacker:51858 > Zombie:80 SA id=6225
RCVD (4.3080s) TCP Zombie:80 > Attacker:51858 R id=15690
SENT (4.3080s) TCP Zombie:80 > Target:25 S id=35713
SENT (4.5410s) TCP Attacker:51856 > Zombie:80 SA id=28118
RCVD (4.6890s) TCP Zombie:80 > Attacker:51856 R id=15692
Discovered open port 25/tcp on Target
SENT (4.6900s) TCP Zombie:80 > Target:110 S id=9943
SENT (4.9210s) TCP Attacker:51836 > Zombie:80 SA id=62254
RCVD (5.0690s) TCP Zombie:80 > Attacker:51836 R id=15694
Discovered open port 110/tcp on Target
```

Daha önce de belirttiğimiz gibi 25 ve 110 numaralı bağlantı noktalarının problemleri bunların açık olduğunu göstermektedir.

```
SENT (5.0690s) TCP Zombie:80 > Target:20 S id=8168
SENT (5.0690s) TCP Zombie:80 > Target:21 S id=36717
SENT (5.0690s) TCP Zombie:80 > Target:22 S id=4063
SENT (5.0690s) TCP Zombie:80 > Target:23 S id=54771
SENT (5.3200s) TCP Attacker:51962 > Zombie:80 SA id=38763
RCVD (5.4690s) TCP Zombie:80 > Attacker:51962 R id=15695
SENT (5.7910s) TCP Attacker:51887 > Zombie:80 SA id=61034
RCVD (5.9390s) TCP Zombie:80 > Attacker:51887 R id=15696
```

Emin olmak için Nmap 20-23 numaralı portları tekrar dener. Bir Zombi IP Kimliği sorgusu sıra atlaması göstermez. Hedef'ten Zombi'ye bir SYN/ACK'in geç gelmesi ihtimaline karşı, Nmap başka bir IP ID sorgusu dener. Bu da yine açık port olmadığını gösterir. Nmap artık sonuçları yazdırınmak için yeterince emin.

```
The Idlescan took 5 seconds to scan 7 ports.
Nmap scan report for Target
PORT      STATE            SERVICE
20/tcp    closed|filtered  ftp-data
21/tcp    closed|filtered  ftp
22/tcp    closed|filtered  ssh
23/tcp    closed|filtered  telnet
24/tcp    closed|filtered  priv-mail
25/tcp    open             smtp
110/tcp   open              pop3

Nmap finished: 1 IP address (1 host up) scanned in 5.949 seconds
```

Nmap boşta tarama uygulaması hakkında tüm ayrıntılar için, Nmap kaynak kodu dağıtımından idle\_scan.cc dosyasını okuyun.

Port taraması öngörlülebilir IP ID dizilerinin zekice kötüye kullanımı olsa da, başka birçok amaç için de kullanılabilir. Örnekler bu kitap boyunca, özellikle de Bölüm 10, Güvenlik Duvarlarını ve Saldırı Tespit Sistemlerini Tespit Etme ve Yıkma'da yer almaktadır.

## **IP Protocol Scan ( -sO ) (IP Protokol Taraması (-sO))**

IP protokol taraması, hedef makineler tarafından hangi IP protokollerinin (TCP, ICMP, IGMP, vb.) desteklendiğini belirlemenizi sağlar. Bu teknik olarak bir port taraması değildir, çünkü TCP veya UDP port numaraları yerine IP protokol numaraları arasında geçiş yapar. Yine de taranan protokol numaralarını seçmek için -p seçeneğini kullanır, sonuçlarını normal port tablosu biçiminde raporlar ve hatta gerçek port tarama yöntemleriyle aynı temel tarama motorunu kullanır. Yani buraya ait olması için bir port taramasına yeterince yakındır.

Protokol taraması kendi başına faydalı olmasının yanı sıra açık kaynaklı yazılımın gücünü de gösteriyor. Temel fikir oldukça basit olsa da, bunu eklemeyi düşünmemiştim ya da böyle bir işlevsellik için herhangi bir talep almamıştım. Daha sonra 2000 yazında Gerhard Rieger bu fikri tasarladı, bunu uygulayan mükemmel bir yama yazdı ve nmap-hackers posta listesine gönderdi. Bu yamayı Nmap ağacına dahil ettim ve ertesi gün yeni bir sürüm yayındım. Çok az ticari yazılım parçası kendi geliştirmelerini tasarlayacak ve katkıda bulunacak kadar hevesli kullanıcılara sahiptir!

Protokol taraması UDP taramasına benzer bir şekilde çalışır. Bir UDP paketinin bağlantı noktası numarası alanını yinelemek yerine, IP paket başlıklarını gönderir ve sekiz bitlik IP protokolü alanını yineler. Başlıklar genellikle boştur, hiçbir veri içermez ve talep edilen protokol için uygun başlık bile değildir. Bazı popüler protokoller (TCP, UDP ve ICMP dahil) için bir istisna yapılır. Bazı sistemler bunları başka türlü göndermeyeceğinden ve Nmap bunları oluşturmak için zaten işlevlere sahip olduğundan, bunlar için uygun protokol başlıkları dahil edilmiştir. ICMP port ulaşılamaz mesajlarını izlemek yerine, protokol taraması ICMP protokol ulaşılamaz mesajlarını arar. Tablo 5.8, IP problarına verilen yanıtların port durumlarıyla nasıl eşleştirildiğini göstermektedir.

Tablo 5.8. Nmap bir IP protokolü probuna verilen yanıtları nasıl yorumlar?

Probe Response (Prob Yanıtı)	Assigned State (Atanmış Durum)
Hedef ana bilgisayardan herhangi bir protokolde herhangi bir yanıt	<b>open</b> (yanıt tarafından kullanılan protokol için, prob protokolü olması gerekmekz)
ICMP protokolüne ulaşılamıyor hatası (tip 3, kod 2)	<b>closed</b>
Diğer ICMP ulaşılamıyor hataları (tip 3, kod 1, 3, 9, 10 veya 13)	<b>filtered</b> (hedef makineden gönderilirse ICMP'nin açık olduğunu kanıtlarlar)
Yanıt alınmadı (yeniden iletişimlerden sonra bile)	<b>open filtered</b>

TCP veya UDP protokollerindeki açık portlar gibi, her açık protokol de potansiyel bir istismar vektörüdür. Buna ek olarak, protokol tarama sonuçları bir makinenin amacını ve ne tür bir paket filtrelemesinin mevcut olduğunu belirlemeye yardımcı olur. Uç ana bilgisayarlar genellikle TCP, UDP, ICMP ve (bazen) IGMP'den biraz daha fazlasına sahipken, yönlendiriciler genellikle GRE ve EGP gibi yönlendirme ile

ilgili protokoller de dahil olmak üzere çok daha fazlasını sunar. Güvenlik duvarları ve VPN ağ geçitleri IPsec ve SWIPE gibi şifrelemeyle ilgili protokollerini gösterebilir.

Bir UDP taraması sırasında alınan ICMP bağlantı noktası ulaşılamaz iletileri gibi, ICMP protokolü ulaşılamaz iletileri de genellikle hız sınırlıdır. Örneğin, varsayılan bir Linux 2.4.20 kutusundan saniyede birden fazla ICMP hedefine ulaşılamıyor yanıtı gönderilmez. Yalnızca 256 olası protokol numarası olduğundan, bu 65.536 portlu bir UDP taramasına göre daha az sorun teşkil eder. "UDP Taramalarını Hızlandırma" bölümündeki öneriler IP protokol taramalarını hızlandırmak için de geçerlidir.

Protokol taraması, komut satırındaki diğer tarama teknikleriyle aynı şekilde kullanılır. Genel Nmap seçeneklerine ek olarak -sO seçeneğini belirtmeniz yeterlidir. Protokol numaralarını seçmek için normal port (-p) seçeneği kullanılır. Ya da nmap-protocols veritabanında listelenen tüm protokoller taramak için -F seçeneğini kullanabilirsiniz. Varsayılan olarak, Nmap tüm 256 olası değeri tarar. Örnek 5.20, Ereet'in Polonya'daki bir yönlendiriciyi ve ardından yerel ağımdaki tipik bir Linux kutusunu taramasını göstermektedir.

Örnek 5.20. Bir yönlendiricinin ve tipik bir Linux 2.4 kutusunun IP protokol taraması

```
# nmap -sO 62.233.173.90 para

Starting Nmap ( https://nmap.org )
Nmap scan report for ntwklan-62-233-173-90.devs.futuro.pl (62.233.173.90)
Not shown: 240 closed ports
PROTOCOL STATE SERVICE
1      open      icmp
4      open|filtered ip
6      open      tcp
8      open|filtered egp
9      open|filtered igrp
17     filtered   udp
47     open|filtered gre
53     filtered   swipe
54     open|filtered narp
55     filtered   mobile
77     filtered   sun-nd
80     open|filtered iso-ip
88     open|filtered eigrp
89     open|filtered ospfigp
94     open|filtered ipip
103    filtered   pim

Nmap scan report for para (192.168.10.191)
Not shown: 252 closed ports
PROTOCOL STATE SERVICE
1      open      icmp
2      open|filtered igmp
6      open      tcp
17     filtered   udp
MAC Address: 00:60:1D:38:32:90 (Lucent Technologies)

Nmap done: 2 IP addresses (2 hosts up) scanned in 458.04 seconds
```

## **TCP FTP Bounce Scan ( -b ) (TCP FTP Sıçrama Taraması (-b))**

FTP protokolünün (RFC 959) ilginç bir özelliği de proxy FTP bağlantılarını desteklemesidir. Bu, bir kullanıcının bir FTP sunucusuna bağlanmasına ve ardından dosyaların üçüncü taraf bir sunucuya gönderilmesini istemesine olanak tanır. Böyle bir özellik birçok düzeye kötüye kullanıma açıktır, bu nedenle çoğu sunucu bunu desteklemeyi bırakmıştır. Bu özelliğin izin verdiği kötüye kullanımlardan biri, FTP sunucusunun diğer ana bilgisayarları port taramasına neden olmasıdır. FTP sunucusundan sırayla hedef ana bilgisayarın her bir ilginç portuna bir dosya göndermesini isteyin. Hata mesajı portun açık olup olmadığını açıklayacaktır. Bu,

güvenlik duvarlarını atlamak için iyi bir yoldur çünkü kurumsal FTP sunucuları genellikle diğer dahili ana bilgisayarlara herhangi bir eski Internet ana bilgisayarından daha fazla erişime sahip oldukları yerlere yerleştirilir. Nmap, -b seçeneği ile FTP sıçrama taramasını destekler. <kullanıcı adı>: <parola>@<sunucu>:<port> şeklinde bir argüman alır. <Server> savunmasız bir FTP sunucusunun adı veya IP adresidir. Normal bir URL'de olduğu gibi <kullanıcı adı>:<şifre> atlanabilir, bu durumda anonim oturum açma bilgileri (kullanıcı:anonim şifre:-wwwuser@) kullanılır. Bağlantı noktası numarası (ve önceki iki nokta üst üste) da atlanabilir, bu durumda <sunucu> üzerindeki varsayılan FTP bağlantı noktası (21) kullanılır.

Örnek 5.21'de ana Microsoft FTP sunucusundan sıçrayarak google'ı tarama girişimi gösterilmektedir.

Örnek 5.21. Bir FTP sıçrama taraması denemesi

```
# nmap -Pn -b ftp.microsoft.com google.com
Starting Nmap ( https://nmap.org )
Your FTP bounce server doesn't allow privileged ports, skipping them.
Your FTP bounce server sucks, it won't let us feed bogus ports!
```

FTP bounce taramasını sık kullananlar bu hata mesajına alıssa iyi olur. Bu güvenlik açığı 1997 yılında Nmap yayınlandığında yaygındı, ancak büyük ölçüde düzeltildi. Savunmasız sunucular hala etrafta, bu yüzden her şey başarısız olduğunda denemeye değer. Amacınız bir güvenlik duvarını aşmaksa, hedef ağı açık port 21 için tarayın (hatta sürüm algılamalı tüm portları tarıyorsanız herhangi bir FTP hizmeti için), ardından her birini kullanarak bir sıçrama taraması deneyin. Nmap size ana bilgisayarın savunmasız olup olmadığını söyleyecektir. Eğer sadece izinizi kaybettirmeye çalışıyorsanız, kendinizi hedef agdaki ana bilgisayarlarla sınırlamanıza gerek yoktur (ve aslında olmamalıdır). Savunmasız FTP sunucuları için rastgele Internet adreslerini taramaya başlamadan önce, sistem yöneticilerinin sunucularını bu şekilde kötüye kullanmanızdan hoşlanmayabileceğini göz önünde bulundurun.

Örnek 5.22 Scanme üzerinde birkaç ilginç bağlantı noktasına karşı başarılı bir sıçrama taramasını göstermektedir. Verbose seçeneği (-v) ekstra ayrıntı sağlamak

için verilmiştir. Verilen sunucu türü "JD FTP Sunucusu", bunun bir HP JetDirect baskı sunucusu olduğu anlamına gelir.

Örnek 5.22. Başarılı FTP sızrama taraması

```
krad~> nmap -p 22,25,135 -Pn -v -b XXX.YY.111.2 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Attempting connection to ftp://anonymous:-wwwuser@XXX.YY.111.2:21
Connected:220 JD FTP Server Ready
Login credentials accepted by ftp server!
Initiating TCP ftp bounce scan against scanme.nmap.org (64.13.134.52)
Adding open port 22/tcp
Adding open port 25/tcp
Scanned 3 ports in 12 seconds via the Bounce scan.
Nmap scan report for scanme.nmap.org (64.13.134.52)
PORT      STATE    SERVICE
22/tcp    open     ssh
25/tcp    open     smtp
135/tcp   filtered msrpc

Nmap done: 1 IP address (1 host up) scanned in 21.79 seconds
```

## Scan Code and Algorithms (Tarama Kodu ve Algoritmalar)

2004 yılında, Nmap'in birincil port tarama motoru daha yüksek performans ve doğruluk için yeniden yazıldı. İşlev adından sonra ultra\_scan olarak bilinen yeni motor SYN, connect, UDP, NULL, FIN, Xmas, ACK, window, Maimon ve IP protokol taramalarının yanı sıra çeşitli ana bilgisayar keşif taramalarını da gerçekleştirmektedir. Geriye sadece kendi motorlarını kullanan boşta tarama ve FTP sızrama taraması kalıyor.

Bu bölümdeki diyagramlar her bir tarama türünün nasıl çalıştığını gösterirken, Nmap uygulaması çok daha karmaşıktır çünkü bağlantı noktası ve ana bilgisayar paralelleştirme, gecikme tahmini, paket kaybı tespiti, zamanlama profilleri, anormal ağ koşulları, paket filtreleri, yanıt oranı sınırları ve çok daha fazlası hakkında endişelenmesi gereklidir.

Bu bölüm ultra\_scan motorunun her düşük seviye detayını sağlamaz. Eğer bunu isteyecek kadar meraklı iseniz, kaynaktan almanız daha iyi olacaktır. ultra\_scan ve

scan\_engine.cc'de tanımlanan üst düzey yardımcı fonksiyonlarını Nmap tarball'undan bulabilirsiniz. Burada en önemli algoritmik özellikleri ele alıyorum. Bunları anlamak, Bölüm 6, Nmap Performansını Optimize Etme'de açıkladığı gibi taramalarınızı daha iyi performans için optimize etmenize yardımcı olur.

### **Network Condition Monitoring (Ağ Durum İzleme)**

Bazı yazarlar, durum bilgisiz çalışma nedeniyle tarayıcılarının Nmap'ten daha hızlı olmasına övünürler. Basitçe bir paket selini patlatıyorlar, sonra yanıtları dinliyorlar ve en iyisini umuyorlar. Bu, hızlı anketler ve hızın kapsamlılık ve doğruluktan daha önemli olduğu diğer durumlar için değerli olsa da, güvenlik taraması için uygun bulmuyorum. Durumsuz bir tarayıcı, yeniden iletmek ve gönderme hızını azaltmak için düşen paketleri tespit edemez. Ağ yolunun yarısındaki meşgul bir yönlendirici tarayıcının paket selinin %80'ini düşürürse, tarayıcı yine de çalışmayı başarılı olarak değerlendirecek ve ne yazık ki yanlış sonuçlar yazdıracaktır. Öte yandan Nmap, çalışırken RAM'de kapsamlı bir durum kaydeder. Bir PDA'da bile genellikle bol miktarda bellek mevcuttur. Nmap her probu sıra numaraları, kaynak veya hedef portları, ID alanları veya yanıtları (ve dolayısıyla düşmeleri) tanımmasını sağlayan diğer özelliklerle (prob türüne bağlı olarak) işaretler. Daha sonra, çizgiyi aşmadan ve yanlışlığa maruz kalmadan veya paylaşılan bir ağı haksız yere tıkamadan ağını (ve verilen komut satırı seçeneklerinin) izin verdiği kadar hızlı kalmak için hızını uygun şekilde ayarlar. Bir IDS kurmamış olan bazı yöneticiler, tüm ağlarının Nmap SYN taramasını fark etmeyebilir. Ancak, Quake ping süresini etkileyen kaba bir paket taşıma tarayıcısı kullanırsanız, yöneticinin araştıracağına inansanız iyi olur!

Nmap'in tıkanıklık kontrol algoritmaları çoğu tarama için önerilse de, geçersiz kılınabilirler. --min-rate seçeneği, Nmap'in normal tıkanıklık kontrol sınırlarını aşsa bile paketleri belirttiğiniz hızda (veya daha yüksek) gönderir. Benzer şekilde, --max-retries seçeneği Nmap'in bir paketi kaç kez yeniden iletebileceğini kontrol eder. min-rate 100 --max-retries 0 gibi seçenekler basit durumsuz tarayıcıların davranışını taklit edecektir. Saniyede 100 yerine 200 paketlik bir hız belirleyerek bu hızı iki katına çıkarabilirsiniz, ancak çok açgözlü olmayın; sonuçlar yanlış veya eksikse aşırı hızlı bir taramanın pek bir değeri yoktur. --min-rate değerini kullanmanın riski size aittir.

### **Host and Port Parallelization (Ana Bilgisayar ve Bağlantı Noktası Paralelleştirme)**

Bu bölümdeki diyagramların çoğu, tek bir bağlantı noktasının durumunu belirlemek için bir teknigin kullanımını göstermektedir. Bir sonda göndermek ve yanıt almak,

kaynak ve hedef makineler arasında en az bir gidiş dönüş süresi (RTT) alır. RTT'niz 200 ms ise ve bir makinede 65.536 bağlantı noktasını tarıyorsanız, bunları seri olarak işlemek en az 3,6 saat sürer. Bu şekilde 20.000 makineden oluşan bir ağın taradığınızda beklenme süresi sekiz yıldan fazla olacaktır. Bu açıkça kabul edilemez bir durumdur, bu nedenle Nmap taramalarını paralelleştirir ve düzinelere makinenin her birinde aynı anda yüzlerce bağlantı noktasını tarayabilir. Bu, hızları birkaç büyülü sırasına göre artırır. Bir seferde taranan ana bilgisayar ve bağlantı noktası sayısı --min-hostgroup, --min-parallelism, -T4, --max-rtt-timeout ve diğerleri dahil olmak üzere Bölüm 6, Nmap Performansını Optimize Etme'de açıklanan argümanlara bağlıdır. Ayrıca Nmap tarafından algılanan ağ koşullarına da bağlıdır.

Birden fazla makineyi tararken, Nmap yükü bunlar arasında verimli bir şekilde yaymaya çalışır. Bir makine bunalmış görünüyorrsa (paketleri düşürür veya gecikmesi artarsa), Nmap bu ana bilgisayar için yavaşlarken diğerlerine karşı tam hızda devam eder.

### **Round Trip Time Estimation (Gidiş Dönüş Süresi Tahmini)**

Bir prob yanıtı her alındığında, Nmap probun gönderilmesinden bu yana geçen mikrosaniyeleri hesaplar. Buna instanceRTT diyeceğiz ve Nmap bunu zamanlamayla ilgili üç önemli değerin çetelesini tutmak için kullanır: srtt, rttvar ve timeout. Nmap her ana bilgisayar için ayrı değerler ve paralel olarak taranan tüm ana bilgisayar grubu için birleştirilmiş değerler tutar. Bunlar aşağıdaki gibi hesaplanır:

**srtt** Düzeltilmiş ortalama gidiş dönüş süresi. Bu, Nmap'in en doğru RTT tahmini olarak kullandığı şeydir. Gerçek bir aritmetik ortalama kullanmak yerine, formül daha yeni sonuçları tercih eder çünkü ağ koşulları sık sık değişir. Formül şöyledir:

```
newsrtt = oldsrtt + (instanceRTT - oldsrtt) / 8
```

**rttvar** Bu, gidiş dönüş süresindeki gözlemlenen varyans veya sapmadır. Buradaki fikir, RTT değerleri oldukça tutarlıysa, Nmap'in srtt'yi bekledikten kısa bir süre sonra pes edebileceğidir. Varyans oldukça yüksekse, Nmap bir probdan vazgeçmeden önce srtt'den çok daha uzun süre beklemelidir, çünkü nispeten yavaş yanıtlar yaygındır. Formül aşağıdaki gibidir (ABS mutlak değer işlemini temsil eder):

```
newrttvar = oldrttvar + (ABS(instanceRTT - oldsrtt) - oldrttvar) / 4
```

`timeout` Bu, Nmap'in bir probdan vazgeçmeden önce beklemeye istekli olduğu süredir. Şu şekilde hesaplanır:

```
timeout = newsrtt + newrttvar * 4
```

Bir prob zaman aşımına uğradığında, Nmap probu yeniden iletebilir veya filtrelenmiş gibi bir port durumu atayabilir (tarama türüne bağlı olarak). Nmap, genel tarama devam ederken geç bir yanıt gelmesi ihtimaline karşı zaman aşımından sonra bile bazı durum bilgilerini saklar.

Bu basit zaman tahmin formülleri oldukça iyi çalışıyor gibi görünmektedir. TCP tarafından kullanılan ve RFC 2988, Computing TCP's Retransmission Timer'da tartışılan benzer tekniklere gevşek bir şekilde dayanmaktadır. Bu algoritmaları yıllar içinde port taramasına daha iyi uyacak şekilde optimize ettik.

### Congestion Control (Tıkanıklık Kontrolü)

Yeniden iletim zamanlayıcıları Nmap'in TCP'den aldığı tek teknik değildir. Nmap en yaygın olarak TCP ile kullanıldığından, aynı kuralların çoğunu takip etmek sadece adıldır. Özellikle de bu kurallar, herkesin bencilce ağı kullandığı bir ortak trajediye dönüşmeden verimi en üst düzeye çıkarmak için yapılan önemli araştırmaların sonucu olduğu için. Varsayılan seçenekleriyle Nmap oldukça naziktir. Nmap, taramanın ne kadar agresif olduğunu kontrol etmek için TCP'den sonra modellenen üç algoritma kullanır: bir tıkanıklık penceresi, üstel geri alma ve yavaş başlatma. Tıkanıklık penceresi Nmap'in aynı anda kaç tane prob gönderebileceğini kontrol eder. Pencere doluysa, Nmap bir yanıt alınana veya bir prob zaman aşımına uğrayana kadar daha fazla göndermez. Üstel geri alma, Nmap'in düşen paketleri algıladığından ölçüde yavaşlamasına neden olur. Düşen paketler tespit edildiğinde tıkanıklık penceresi genellikle bire düşürülür. Yavaş adında olmasına rağmen, yavaş başlangıç, ağır performans sınırlarını belirlemek için tarama hızını kademeli olarak artırmak için oldukça hızlı bir algoritmadır.

Tüm bu teknikler RFC 2581, TCP Congestion Control'de açıklanmıştır. Bu belge ağ uzmanları Richard Stevens, Vern Paxson ve Mark Allman tarafından yazılmıştır. Sadece 10 sayfa uzunluğundadır ve verimli TCP yiğinları (veya diğer ağ protokollerİ veya port tarayıcıları) uygulamakla ilgilenen herkes bunu büyüleyici bulmalıdır.

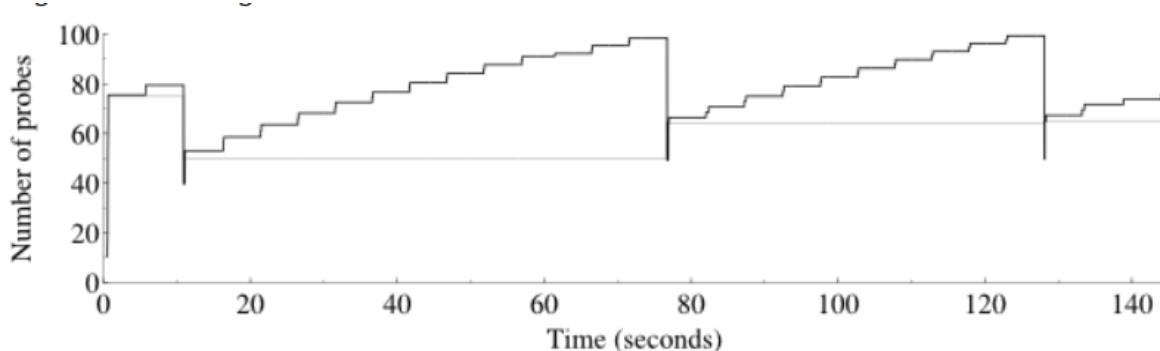
Nmap bir grup hedefi taradığında, bellekte her hedef için bir tıkanıklık penceresi ve eşik değerinin yanı sıra bir bütün olarak grup için bir pencere ve eşik değeri tutar. Tıkanıklık penceresi bir seferde gönderilebilecek prob sayısıdır. Tıkanıklık eşiği, yavaş başlatma ve tıkanıklıktan kaçınma modları arasındaki sınırı tanımlar. Yavaş

başlatma sırasında, tıkanıklık penceresi yanıtlarına yanıt olarak hızla büyür. Tıkanıklık penceresi tıkanıklık eşğini aştığında, tıkanıklıktan kaçınma modu başlar ve bu sırada tıkanıklık penceresi daha yavaş artar. Bir düşüşten sonra, hem tıkanıklık penceresi hem de eşik önceki değerlerinin bir kısmına düşürülür.

Ancak TCP akışları ile Nmap port taramaları arasında önemli bir fark vardır. TCP akışlarında, gönderilen her pakete (ya da en azından büyük bir kısmına) yanıt olarak ACK beklemek normaldir. Aslında, tıkanıklık penceresinin uygun şekilde büyümesi bu varsayıma bağlıdır. Nmap genellikle kendini farklı bir durumda bulur: varsayılan güvenlik duvarına sahip bir hedefle karşılaşlığında, gönderilen paketlerin çok azına yanıt verilecektir. Aynı şey, yalnızca birkaç canlı ana bilgisayar içeren bir ağ adresi bloğunu ping tararken de olur. Bunu telafi etmek için Nmap, gönderilen paketlerin alınan yanıtlarına oranını takip eder. Grup tıkanıklık penceresi her değiştiğinde, değişiklik miktarı bu oranla çarpılır. Başka bir deyişle, az sayıda paket yanıt aldığından, her yanıt daha fazla ağırlık taşıır.

Tipik bir port taraması sırasında grup tıkanıklık penceresi ve eşinin nasıl değiştiğinin grafiksel bir açıklaması Şekil 5.9'da gösterilmektedir. Tıkanıklık penceresi siyah renkte ve tıkanıklık eşigi gri renkte gösterilmiştir.

Şekil 5.9. Tıkanıklık penceresi ve eşik



Tıkanıklık penceresi düşük başlar ve tıkanıklık eşigi yüksek başlar. Yavaş başlatma modu başlar ve pencere boyutu hızla artar. Büyük "stairstep" atlamları zamanlama pinglerinin sonucudur. Yaklaşık 10 saniyede, bir düşme tespit edildiğinde tıkanıklık penceresi 80 sondaya ulaşmıştır. Hem tıkanıklık penceresi hem de eşik azaltılır. Tıkanıklık penceresi, başka bir düşüş tespit edildiğinde yaklaşık 80 saniyeye kadar büyümeye devam eder. Ardından döngü tekrarlanır, bu da ağ koşulları sabit olduğunda tipiktir.

Tarama sırasında düşüşler korkulacak bir şey değildir. Tıkanıklık kontrol algoritmalarının amacı, kapasitesini keşfetmek için ağın dinamik olarak araştırmaktır. Bu şekilde bakıldığından, düşüşler Nmap'in tıkanıklık penceresi için doğru boyutu belirlemesine yardımcı olan değerli geri bildirimlerdir.

### **Timing probes (Zamanlama problemleri)**

Bu algoritmalar bölümünde tartışılan her teknik, ağ paket kaybı ve gecikmesini tespit etmek ve tahmin etmek için (bir düzeyde) ağ izlemeyi içerir. Bu gerçekten de hızlı tarama süreleri elde etmek için kritik öneme sahiptir. Ne yazık ki, yoğun güvenlik duvarlı sistemleri tararken iyi veri elde etmek genellikle zordur. Bu filtreler genellikle paketlerin ezici çoğunluğunu herhangi bir yanıt vermeden düşürür. Nmap, yanıt veren bir bağlantı noktası bulmak için 20.000 veya daha fazla sonda göndermek zorunda kalabilir ve bu da ağ koşullarını izlemeyi zorlaştırır.

Bu sorunla mücadele etmek için Nmap, port tarama pingleri olarak da bilinen zamanlama problemini kullanır. Nmap, yoğun şekilde filtrelenmiş bir ana bilgisayarda yanıt veren en az bir bağlantı noktası bulduysa, diğer bağlantı noktalarından yanıt almadan geçen her 1,25 saniyede bir bu bağlantı noktasına bir sonda gönderir. Bu, Nmap'in ağ koşulları izin verdikçe taramalarını hızlandırmak veya yavaşlatmak için yeterli düzeyde izleme yapmasına olanak tanır.

### **Inferred Neighbor Times (Çıkarılmış Komşu Zamanları)**

Bazen port tarama pingleri bile yardımcı olmaz çünkü hiçbir yanıt veren port bulunamamıştır. Makine kapalı olabilir (ve -Pn ile taranmış olabilir) ya da her bir port filtrelenmiş olabilir. Ya da belki de hedefin birkaç duyarlı portu vardır, ancak Nmap henüz onları bulacak kadar şanslı değildir. Bu durumlarda Nmap, aynı anda taradığı tüm makine grubu için koruduğu zamanlama değerlerini kullanır. Gruptaki herhangi bir makineden en az bir yanıt alındığı sürece, Nmap'in çalışabileceği bir şey vardır. Elbette Nmap bir gruptaki ana bilgisayarların her zaman benzer zamanlama özelliklerini paylaştığını varsayılamaz. Bu nedenle Nmap, bir gruptaki duyarlı ana bilgisayarlar arasındaki zamanlama farklılıklarını izler. Eğer büyük farklılıklar gösterirlerse, Nmap güvenli tarafta olmak için komşu ana bilgisayarlar için uzun zaman aşımıları çıkarır.

### **Adaptive Retransmission (Uyarlanabilir Yeniden İletim)**

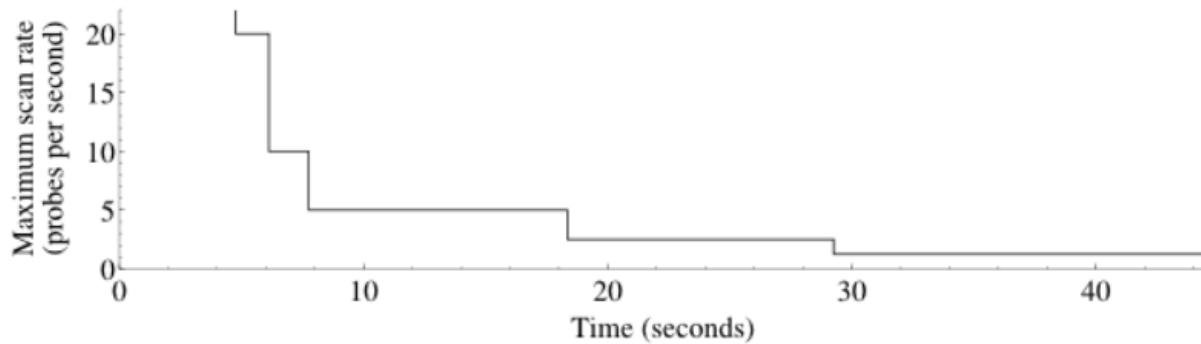
En basit tarayıcılar (ve durum bilgisi olmayanlar) genellikle problemleri yeniden iletmezler. Sadece her bağlantı noktasına bir sonda gönderir ve yanıtta ya da

yanıtsızlığa göre rapor verirler. Biraz daha karmaşık tarayıcılar belirli sayıda yeniden iletim yapacaktır. Nmap, bir hedefe karşı yapılan her tarama için dikkatli paket kaybı istatistikleri tutarak daha akıllı olmaya çalışır. Herhangi bir paket kaybı tespit edilmezse, Nmap bir prob yanıtı alamadığında yalnızca bir kez yeniden iletебilir. Büyük paket kayıpları görüldüğünde, Nmap on ya da daha fazla kez yeniden iletim yapabilir. Bu, Nmap'in hızlı, güvenilir ağlardaki ana bilgisayarları hızlı bir şekilde taramasını sağlarken, sorunlu ağları veya makineleri tararken doğruluğu (biraz hız pahasına) korur. Yine de Nmap'in sabrı sınırsız değildir. Belirli bir noktada (on yeniden iletim), Nmap bir uyarı yazdıracak ve daha fazla yeniden iletimden vazgeçecektir. Bu, kötü niyetli ana bilgisayarların kasıtlı paket düşüşleri, yavaş yanıtlar ve benzer saçmalıklarla Nmap'i çok fazla yavaşlatmasını öner. Bu teknik tarpitting olarak bilinir ve genellikle spam göndericilere karşı kullanılır.

### **Scan Delay (Tarama Gecikmesi)**

Paket yanıt hızı sınırlaması belki de Nmap gibi port tarayıcılarının karşılaşacağı en tehlikeli sorundur. Örneğin, Linux 2.4 çekirdekleri UDP (-sU) veya IP protokolü (-sO) taraması sırasında dönen ICMP hata mesajlarını saniyede bir ile sınırlar. Eğer Nmap bunları normal düşüşler olarak saydı, sürekli olarak yavaşlayacak (üstel geri çekilmeyi hatırlayın) ancak yine de problemlerin büyük çoğunluğunun düşmesine neden olacaktır. Bunun yerine, Nmap bu durumu tespit etmeye çalışır. Paketlerin büyük bir kısmı düşüğünde, tek bir hedefe gönderilen her prob arasında kısa bir gecikme (5 milisaniye kadar az) uygular. Düşmeler önemli bir sorun olmaya devam ederse, Nmap düşmeler sona erene veya Nmap izin verilen maksimum tarama gecikmesine ulaşana kadar gecikmeyi iki katına çıkarmaya devam edecektir. Yanıt hızı sınırlı bir Linux ana bilgisayarının UDP 1-50 portlarını tararken tarama gecikmesinin etkileri Şekil 5.10'da gösterilmiştir. Başlangıçta, tarama hızı tarama gecikmesi ile sınırsızdır, ancak elbette tıkanıklık kontrolü gibi diğer mekanizmalar kendi sınırlarını uygular. Düşüşler tespit edildiğinde, tarama gecikmesi iki katına çıkar, yani maksimum tarama hızı etkin bir şekilde yarıya indirilir. Grafikte, örneğin, saniyede beş paketlik bir maksimum tarama hızı 200 milisaniyelik bir tarama gecikmesine karşılık gelir.

Şekil 5.10. Tarama gecikmesinden etkilenen tarama hızı



Maksimum tarama gecikmesi varsayılan olarak problemler arasında bir saniyedir. Tarama gecikmesi bazen, yavaş bir ana bilgisayarın hız sınırlama kuralları olmasa bile, bu ana bilgisayara yetişemediği durumlarda etkinleştirilir. Bu, boş harcanan (düşen) tarama paketlerini azaltarak toplam ağ trafiğini önemli ölçüde azaltabilir. Ne yazık ki küçük tarama gecikme değerleri bile bir taramanın birkaç kat daha uzun sürmesine neden olabilir. Nmap varsayılan olarak muhafazakardır, TCP ve UDP problemleri için saniye uzunluğunda tarama gecikmelerine izin verir. Öncelikleriniz farklıysa, Bölüm 5, Port Tarama Teknikleri ve Algoritmaları'nda tartışıldığı gibi --max-scan-delay ile maksimum tarama gecikmelerini yapılandırabilirsiniz.

**next next next**