Active Reconnaissance

Task 1 Introduction (Görev 1 Giriş)

Ağ Güvenliği Modülünün ilk odasında pasif keşif konusuna odaklanmıştık. Bu ikinci odada ise aktif keşif ve bununla ilgili temel araçlara odaklanıyoruz. Hedefimiz hakkında daha fazla bilgi toplamak için bir web tarayıcısı kullanmayı öğreniyoruz. Ayrıca, ağ, sistem ve hizmetler hakkında bilgi toplamak için ping, traceroute, telnet ve nc gibi basit araçları kullanmayı tartışıyoruz.

Bir önceki odada öğrendiğimiz gibi, pasif keşif, herhangi bir doğrudan angajman veya bağlantı olmadan hedefiniz hakkında bilgi toplamanızı sağlar. Uzaktan izliyorsunuz ya da kamuya açık bilgileri kontrol ediyorsunuz.

Aktif keşif, hedefinizle bir tür temas kurmanızı gerektirir. Bu temas, genellikle sosyal mühendisliğin bir parçası olarak daha fazla bilgi toplamak için bir telefon görüşmesi veya hedef şirkete bir ziyaret olabilir. Alternatif olarak, web sitelerini

ziyaret etmek veya güvenlik duvarlarında bir SSH portunun açık olup olmadığını kontrol etmek gibi hedef sisteme doğrudan bir bağlantı olabilir. Bunu pencereleri ve kapı kilitlerini yakından inceliyormuşsunuz gibi düşünün. Bu nedenle, müşteriden imzalı yasal yetki almadan önce aktif keşif işine girmemeyi unutmamak önemlidir.

Bu odada aktif keşif üzerine odaklanacağız. Aktif keşif, hedef makineye yapılan doğrudan bağlantılarla başlar. Böyle bir bağlantı, diğer şeylerin yanı sıra, günlüklerde istemci IP adresini, bağlantı zamanını ve bağlantı süresini gösteren bilgiler bırakabilir. Ancak, tüm bağlantılar şüpheli değildir. Aktif keşiflerinizin normal istemci etkinliği olarak görünmesini sağlamak mümkündür. Web'de gezinmeyi düşünün; diğer yüzlerce meşru kullanıcı arasında hedef web sunucusuna bağlı bir tarayıcıdan kimse şüphelenmez. Kırmızı takımın (saldırganlar) bir parçası olarak çalışırken ve mavi takımı (savunmacılar) alarma geçirmek istemediğinizde bu tür teknikleri kendi yararınıza kullanabilirsiniz.

Bu odada, çoğu işletim sistemiyle birlikte gelen ya da kolayca elde edilebilen çeşitli araçları inceleyeceğiz. Web tarayıcısı ve yerleşik geliştirici araçları ile başlıyoruz; ayrıca, bir web tarayıcısının etkili bir keşif çerçevesi haline gelmek için nasıl

"silahlandırılabileceğini" gösteriyoruz. Daha sonra, ping, traceroute ve telnet gibi diğer iyi huylu araçları tartışıyoruz. Tüm bu programlar hedefe bağlanmayı gerektirir ve bu nedenle faaliyetlerimiz aktif keşif kapsamına girer.

Bu oda, temel araçlara aşina olmak ve bunları aktif keşifte nasıl kullanabileceklerini görmek isteyen herkes için ilgi çekicidir. Web tarayıcısı geliştirici araçları, grafiksel bir kullanıcı arayüzü sunmasına rağmen, aşinalık kazanmak için biraz çaba gerektirebilir. Kapsanan komut satırı araçlarının kullanımı nispeten basittir.

Önemli Uyarı: Abone değilseniz, AttackBox'ın İnternet erişimi olmayacağını, bu nedenle İnternet erişimi gerektiren soruları tamamlamak için VPN kullanmanız gerekeceğini lütfen unutmayın.

Soru ⇒ Bu araçların neden aktif keşif kapsamına girdiğini anladığınızdan emin olun. AttackBox'ınızı başlatın ve hazır olduğundan emin olun. Özellikle daha sonraki görevlerde soruları yanıtlamak için ona ihtiyacınız olacak.

Cevap ⇒ Cevap Gerekmemektedir.

Task 2 Web Browser (Görev 2 Web Tarayıcısı)

Web tarayıcısı, özellikle tüm sistemlerde kolayca bulunabildiği için kullanışlı bir araç olabilir. Bir hedef hakkında bilgi toplamak için bir web tarayıcısını kullanabileceğiniz birkaç yol vardır.

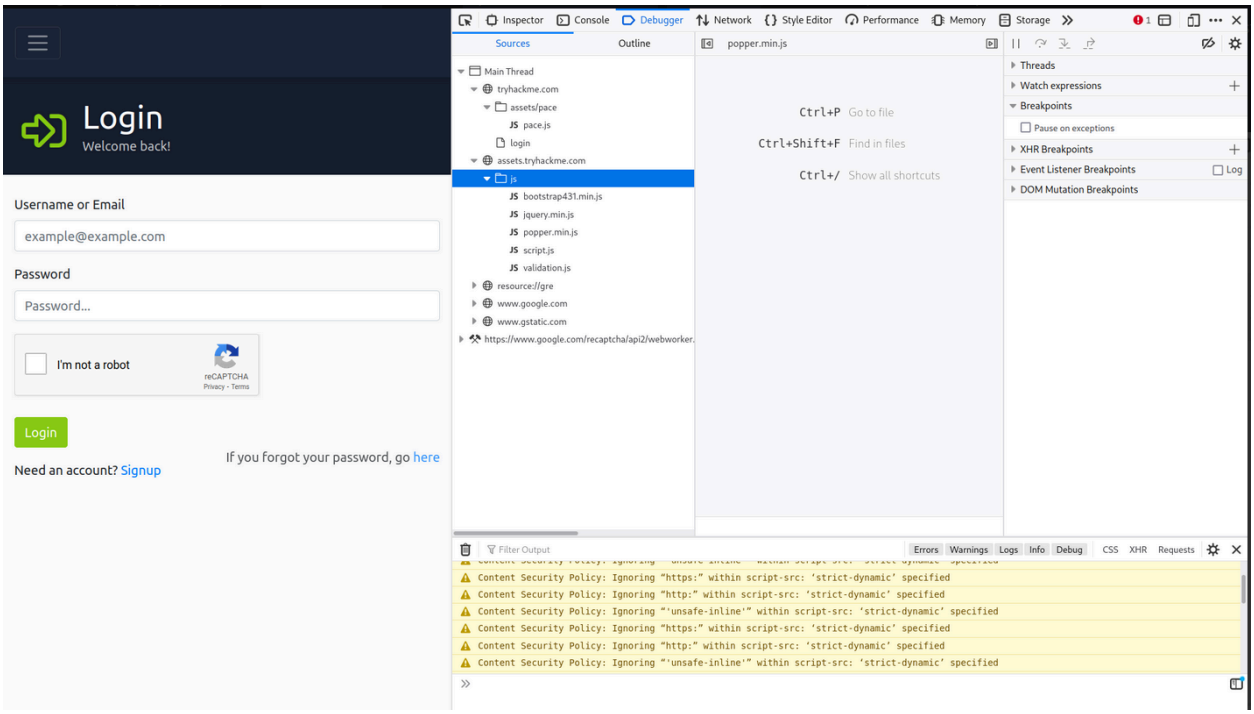
Aktarım düzeyinde, tarayıcı şu adrese bağlanır:

- Web sitesine HTTP üzerinden erişildiğinde varsayılan olarak TCP bağlantı noktası 80
- Web sitesine HTTPS üzerinden erişildiğinde varsayılan olarak TCP bağlantı noktası 443

80 ve 443 HTTP ve HTTPS için varsayılan bağlantı noktaları olduğundan, web tarayıcısı bunları adres çubuğunda göstermez. Ancak, bir hizmete erişmek için özel bağlantı noktaları kullanmak mümkündür. Örneğin, https://127.0.0.1:8834/ HTTPS protokolü aracılığıyla 8834 numaralı bağlantı noktasından 127.0.0.1 (localhost) adresine bağlanacaktır. Eğer bu portu dinleyen bir HTTPS sunucusu varsa, bir web sayfası alacağız.

Bir web sayfasında gezinirken, Firefox'ta Geliştirici Araçlarını açmak için PC'de Ctrl+Shift+I veya Mac'te Option + Command + I (⌥ + ⌘ + I) tuşlarına basabilirsiniz. Benzer kısayollar Google Chrome veya Chromium'da da işe başlamanızı sağlayacaktır. Geliştirici Araçları, tarayıcınızın uzak sunucu ile aldığı ve değiş tokuş ettiği birçok şeyi incelemenizi sağlar. Örneğin, JavaScript (JS) dosyalarını görüntüleyebilir ve hatta değiştirebilir, sisteminizde ayarlanan çerezleri inceleyebilir ve site içeriğinin klasör yapısını keşfedebilirsiniz.

Aşağıda Firefox Geliştirici Araçlarının bir ekran görüntüsü bulunmaktadır. Chrome DevTools oldukça benzerdir.

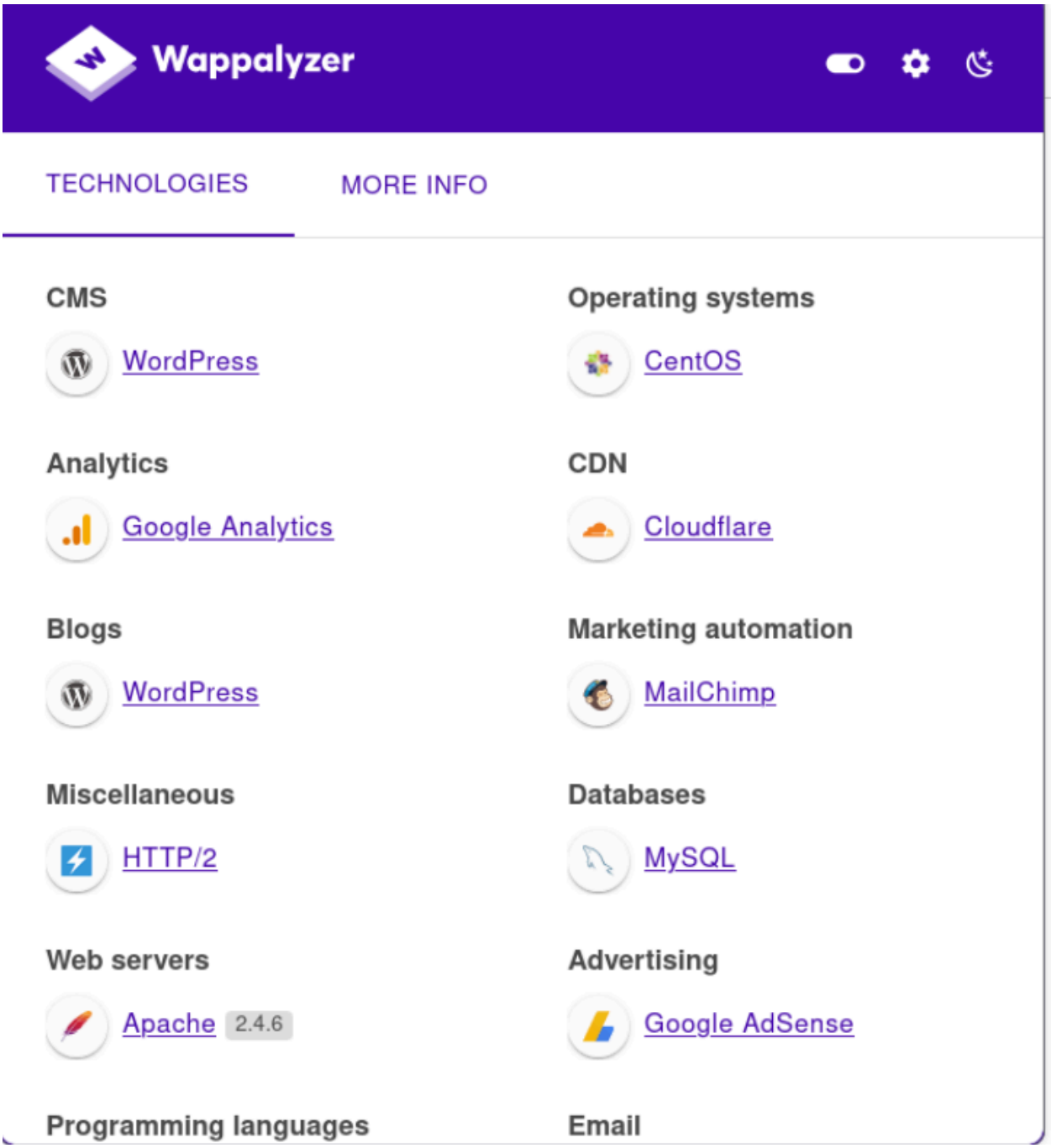


Firefox ve Chrome için sızma testlerine yardımcı olabilecek çok sayıda eklenti de vardır. İşte birkaç örnek:

- FoxyProxy, hedef web sitesine erişmek için kullandığınız proxy sunucusunu hızlı bir şekilde değiştirmenizi sağlar. Bu tarayıcı uzantısı, Burp Suite gibi bir araç kullanıyorsanız veya düzenli olarak proxy sunucularını değiştirmeniz gerekiyorsa kullanışlıdır. Firefox için FoxyProxy'yi buradan edinebilirsiniz.
- User-Agent Switcher and Manager size web sayfasına farklı bir işletim sisteminden veya farklı bir web tarayıcısından erişiyormuş gibi davranma olanağı sağlar. Başka bir deyişle, aslında Mozilla Firefox'tan eriştiğiniz bir

siteye iPhone kullanarak göz atıyormuş gibi davranabilirsiniz. Firefox için User-Agent Switcher ve Manager'ı buradan indirebilirsiniz.

- Wappalyzer, ziyaret edilen web sitelerinde kullanılan teknolojiler hakkında bilgi sağlar. Bu tür bir uzantı, öncelikle diğer kullanıcılar gibi web sitesinde gezinirken tüm bu bilgileri topladığınızda kullanışlıdır. Wappalyzer'ın bir ekran görüntüsü aşağıda gösterilmiştir. Firefox için Wappalyzer'ı burada bulabilirsiniz.



Zamanla, iş akışınıza mükemmel şekilde uyan birkaç uzantı bulabilirsiniz.

Soru ⇒ Aşağıdaki web sitesine gidin ve AttackBox Firefox'ta veya bilgisayarınızdaki tarayıcıda Geliştirici Araçlarınızı açtığınızdan emin olun. Geliştirici Araçlarını kullanarak toplam soru sayısını bulun. (İpucu ⇒ JavaScript dosyası script.js'yi bulun. Dosyayı inceleyin ve toplam soru sayısını bulun.)

Cevap ⇒ 8

Task 3 Ping (Görev 3 Ping)

Ping size ping-pong (masa tenisi) oyununu hatırlatmalıdır. Topu atarsınız ve geri almayı beklersiniz. Ping'in birincil amacı uzaktaki sisteme ulaşıp ulaşamadığınızı ve uzaktaki sistemin de size ulaşıp ulaşamadığını kontrol etmektir. Başka bir deyişle, başlangıçta bu, ağ bağlantısını kontrol etmek için kullanılıyordu; ancak biz daha çok farklı kullanımlarıyla ilgileniyoruz: uzaktaki sistemin çevrimiçi olup olmadığını kontrol etmek.

Basit bir ifadeyle, ping komutu uzaktaki bir sisteme bir paket gönderir ve uzaktaki sistem yanıt verir. Bu şekilde, uzaktaki sistemin çevrimiçi olduğu ve ağın iki sistem arasında çalıştığı sonucuna varabilirsiniz.

Daha seçici bir tanım tercih ederseniz, ping uzaktaki bir sisteme ICMP Echo paketi gönderen bir komuttur. Uzak sistem çevrimiçiyse ve ping paketi doğru şekilde yönlendirilmiş ve herhangi bir güvenlik duvarı tarafından engellenmemişse, uzak sistem bir ICMP Echo Yanıtı göndermelidir. Benzer şekilde, ping yanıtı uygun şekilde yönlendirilmişse ve herhangi bir güvenlik duvarı tarafından engellenmemişse ilk sisteme ulaşmalıdır.

Böyle bir komutun amacı, çalışan işletim sistemini ve hizmetleri keşfetmek için daha ayrıntılı taramalar yapmak için zaman harcamadan önce hedef sistemin çevrimiçi olduğundan emin olmaktır.

AttackBox terminalinizde, ping'i ping MACHINE_IP veya ping HOSTNAME olarak kullanmaya başlayabilirsiniz. İkincisinde, sistemin ping paketini göndermeden önce HOSTNAME'i bir IP adresine çözümlemesi gerekir. Bir Linux sisteminde sayıyı belirtmezseniz, durmaya zorlamak için CTRL+c tuşlarına basmanız gerekecektir. Bu nedenle, sadece on paket göndermek istiyorsanız ping -c 10 MACHINE_IP komutunu kullanabilirsiniz. Bu, bir MS Windows sisteminde ping -n 10 MACHINE_IP ile eşdeğerdir.

Teknik olarak ping, ICMP (Internet Control Message Protocol) protokolü altında yer alır. ICMP birçok sorgu türünü destekler, ancak biz özellikle ping (ICMP echo/type 8) ve ping reply (ICMP echo reply/type 0) ile ilgileniyoruz. Ping kullanmak için ICMP detaylarına girmek gerekli değildir.

Aşağıdaki örnekte, toplam paket sayısını 5 olarak belirledik. AttackBox'ın terminalinden MACHINE_IP'ye ping atmaya başladık. MACHINE_IP'nin açık olduğunu ve ICMP echo isteklerini engellemediğini öğrendik. Ayrıca, paket rotası üzerindeki herhangi bir güvenlik duvarı ve yönlendirici de ICMP yankı isteklerini engellemiyor.

```
user@AttackBox$ ping -c 5 MACHINE_IPPING MACHINE_IP (MACHINE_IP) 56
(84) bytes of data.
64 bytes from MACHINE_IP: icmp_seq=1 ttl=64 time=0.636 ms
64 bytes from MACHINE_IP: icmp_seq=2 ttl=64 time=0.483 ms
64 bytes from MACHINE_IP: icmp_seq=3 ttl=64 time=0.396 ms
64 bytes from MACHINE_IP: icmp_seq=4 ttl=64 time=0.416 ms
64 bytes from MACHINE_IP: icmp_seq=5 ttl=64 time=0.445 ms

--- MACHINE_IP ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4097ms
rtt min/avg/max/mdev = 0.396/0.475/0.636/0.086 ms
```

Yukarıdaki örnekte, hedef sistemin yanıt verdiğini açıkça gördük. Ping çıktısı çevrimiçi ve erişilebilir olduğunun bir göstergesidir. Beş paket gönderdik ve beş yanıt aldık. Ortalama olarak, yanıtın sistemimize ulaşmasının 0,475 ms (milisaniye) sürdüğünü ve maksimum 0,636 ms olduğunu fark ettik.

Sızma testi bakış açısından, bu hedef hakkında daha fazla şey keşfetmeye çalışacağız. Örneğin, hangi portların açık olduğu ve hangi hizmetlerin çalıştığı gibi mümkün olduğunca çok şey öğrenmeye çalışacağız.

Aşağıdaki durumu ele alalım: hedef sanal makineyi kapattık ve ardından MACHINE_IP adresine ping atmaya çalıştık. Aşağıdaki örnekte beklediğiniz gibi, herhangi bir yanıt alamıyoruz.

```
user@AttackBox$ ping -c 5 MACHINE_IPPING MACHINE_IP (MACHINE_IP) 56
(84) bytes of data.
From ATTACKBOX_IP icmp_seq=1 Destination Host Unreachable
From ATTACKBOX_IP icmp_seq=2 Destination Host Unreachable
From ATTACKBOX_IP icmp_seq=3 Destination Host Unreachable
From ATTACKBOX_IP icmp_seq=4 Destination Host Unreachable
```

```
From ATTACKBOX_IP icmp_seq=5 Destination Host Unreachable

--- MACHINE_IP ping statistics ---
5 packets transmitted, 0 received, +5 errors, 100% packet loss, time 4098ms
pipe 4
```

Bu durumda, MACHINE_IP'ye sahip hedef bilgisayarı kapattığımızı zaten biliyoruz. Her ping için, kullandığımız sistem, bu durumda AttackBox, "Hedef Ana Bilgisayara Ulaşılamıyor" yanıtını veriyor. Beş paket gönderdiğimizi, ancak hiçbirinin alınmadığını ve %100 paket kaybıyla sonuçlandığını görebiliriz.

Genel olarak, ping yanıtı alamadığımızda, örneğin neden ping yanıtı alamadığımızı açıklayacak birkaç açıklama vardır:

- Hedef bilgisayar yanıt vermiyor; muhtemelen hala açılıyor veya kapalı ya da işletim sistemi çökmüş.
- Ağa bağlı değildir veya yolun karşısında arızalı bir ağ cihazı vardır.
- Bir güvenlik duvarı bu tür paketleri engellemek üzere yapılandırılmıştır. Güvenlik duvarı, sistemin kendisinde çalışan bir yazılım parçası veya ayrı bir ağ cihazı olabilir. MS Windows güvenlik duvarının varsayılan olarak ping'i engellediğini unutmayın.
- Sisteminizin ağ bağlantısı kesilmiş.

Sorular

Soru ⇒ ICMP yankı isteği tarafından taşınan verilerin boyutunu ayarlamak için hangi seçeneği kullanırsınız (İpucu ⇒ Ping'in kılavuz sayfalarına başvurmak için "man ping" komutunu kullanın.)?

Cevap ⇒ -s

Soru ⇒ ICMP başlığının bayt cinsinden boyutu nedir(Ping'in kılavuz sayfalarına başvurmak için "man ping" komutunu kullanın.)?

Cevap ⇒ 8

Soru ⇒ MS Windows Güvenlik Duvarı varsayılan olarak ping'i engelliyor mu? (Y/N)

Cevap ⇒ Y

Soru ⇒ Bu görev için sanal makineyi dağıtın ve AttackBox terminalini kullanarak ping -c 10 MACHINE_IP komutunu verin. Kaç tane ping yanıtı aldınız?

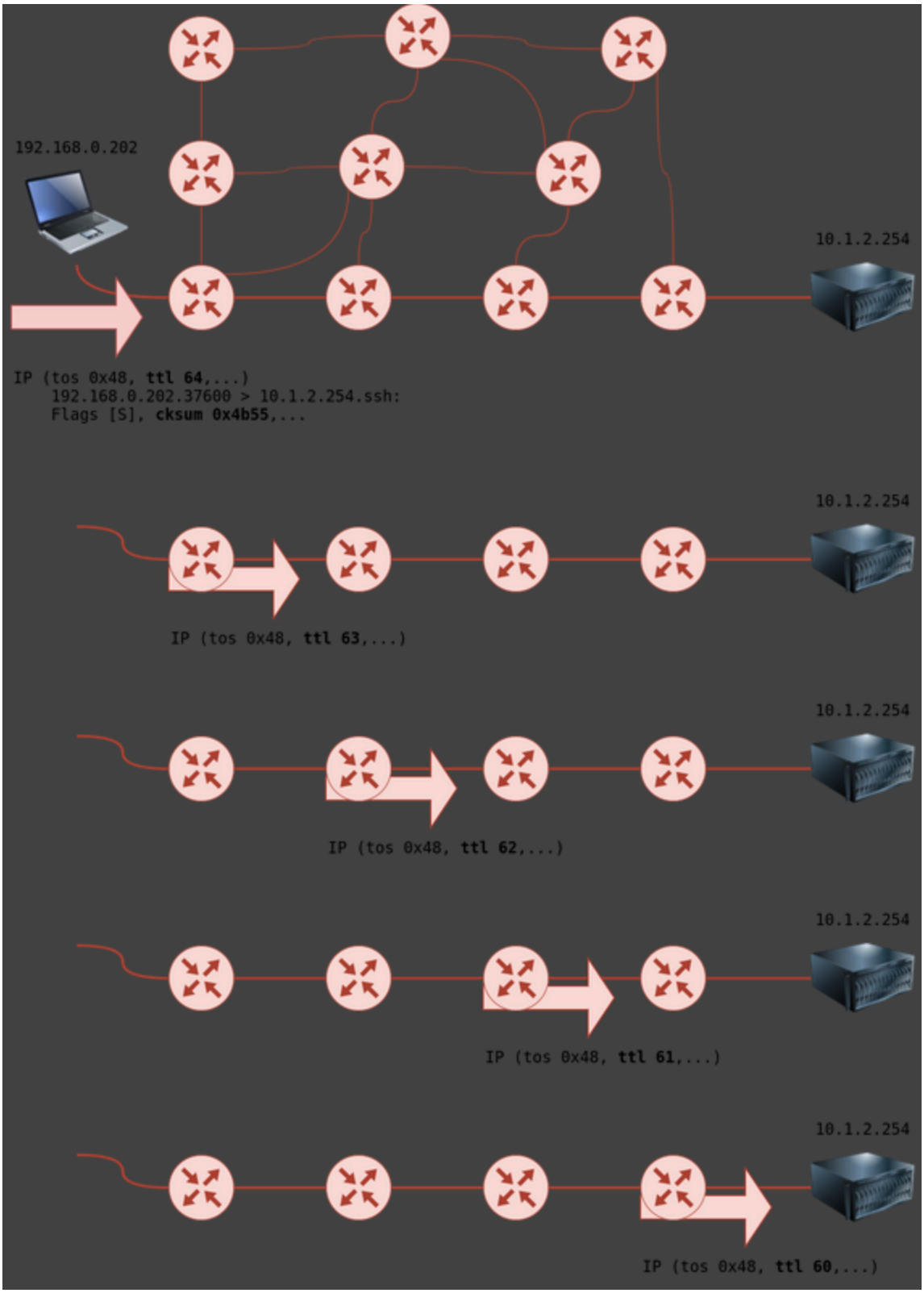
Cevap ⇒ 10

Task 4 Traceroute (Görev 4 Traceroute)

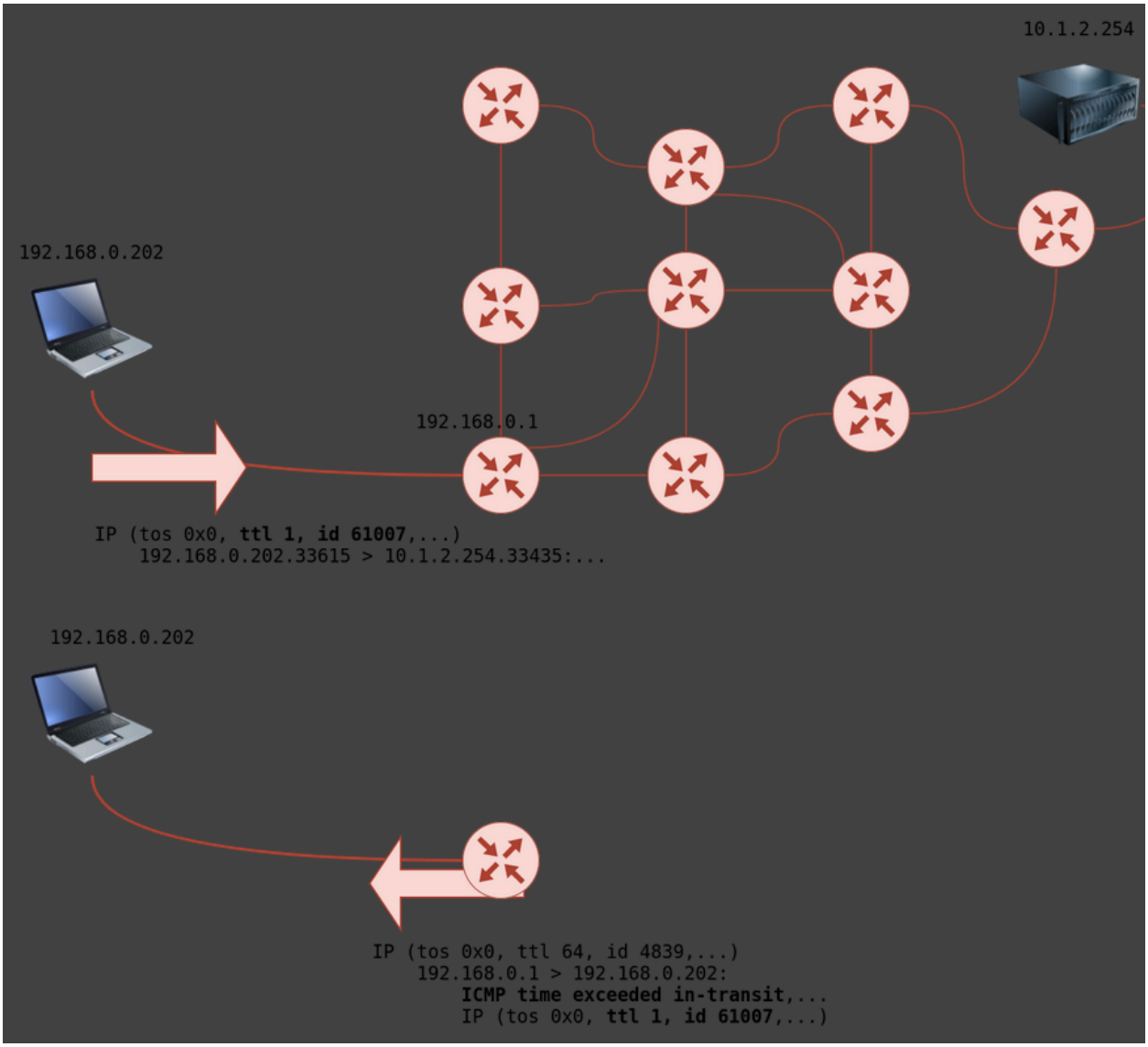
Adından da anlaşılacağı gibi traceroute komutu, sisteminizden başka bir ana bilgisayara giden paketlerin izlediği rotayı takip eder. Traceroute'un amacı, bir paketin sisteminizden hedef ana bilgisayara giderken geçtiği yönlendiricilerin veya atlamaların IP adreslerini bulmaktır. Bu komut aynı zamanda iki sistem arasındaki yönlendirici sayısını da ortaya çıkarır. Sisteminiz ile hedef ana bilgisayar arasındaki atlama sayısını (yönlendirici) gösterdiğinden faydalıdır. Ancak, birçok yönlendirici ağ değişikliklerine uyum sağlayan dinamik yönlendirme protokolleri kullandığından, paketlerin izlediği rotanın değişebileceğini unutmayın.

Linux ve macOS'ta kullanılacak komut traceroute MACHINE_IP, MS Windows'ta ise tracert MACHINE_IP'dir. traceroute, sisteminizden hedef sisteme giden yol üzerindeki yönlendiricileri bulmaya çalışır.

Sisteminizden hedef sisteme giden yolu keşfetmenin doğrudan bir yolu yoktur. Yönlendiricileri IP adreslerini göstermeleri için "kandırmak" için ICMP'ye güveniyoruz. Bunu IP başlık alanında küçük bir Time To Live (TTL) kullanarak başarabiliriz. TTL'deki T zaman anlamına gelse de, TTL bir paketin düşürülmeden önce geçebileceği maksimum yönlendirici/atlama sayısını gösterir; TTL maksimum zaman birimi sayısı değildir. Bir yönlendirici bir paket aldığında, paketi bir sonraki yönlendiriciye aktarmadan önce TTL'yi bir azaltır. Aşağıdaki şekilde IP paketinin bir yönlendiriciden her geçişinde TTL değerinin 1 azaltıldığı görülmektedir. Başlangıçta 64 TTL değeri ile sistemden ayrılır; 4 yönlendiriciden geçtikten sonra 60 TTL değeri ile hedef sisteme ulaşır.


192.168.0.202
10.1.2.254
IP (tos 0x48, ttl 64,...)
192.168.0.202.37600 > 10.1.2.254.ssh:
Flags [S], cksum 0x4b55,...
10.1.2.254
IP (tos 0x48, ttl 63,...)
10.1.2.254
IP (tos 0x48, ttl 62,...)
10.1.2.254
IP (tos 0x48, ttl 61,...)
10.1.2.254
IP (tos 0x48, ttl 60,...)

Ancak, TTL 0'a ulaşırsa düşürülür ve orijinal göndericiye bir ICMP Time-to-Live exceeded mesajı gönderilir. Aşağıdaki şekilde, sistem yönlendiriciye göndermeden önce TTL'yi 1 olarak ayarlamıştır. Yol üzerindeki ilk yönlendirici TTL'yi 1 azaltarak TTL'nin 0 olmasına neden olur. Sonuç olarak, bu yönlendirici paketi atar ve bir ICMP time exceeded in-transit hata mesajı gönderir. Bazı yönlendiricilerin bir paketi atarken bu tür ICMP mesajları göndermeyecek şekilde yapılandırıldığını unutmayın.



Linux'ta traceroute, TTL'si 1 olan IP paketleri içinde UDP datagramları göndererek başlayacaktır. Böylece, ilk yönlendiricinin bir TTL=0 ile karşılaşmasına ve bir ICMP Time-to-Live aşılmış geri göndermesine neden olur. Dolayısıyla, 1 TTL ilk

yönlendiricinin IP adresini size gösterecektir. Ardından TTL=2 ile başka bir paket gönderecek; bu paket ikinci yönlendiricide düşürülecektir. Ve böyle devam eder. Bunu canlı sistemlerde deneyelim.

Aşağıdaki örneklerde, TryHackMe'nin AttackBox'ından aynı komutu, traceroute tryhackme.com'u çalıştırıyoruz. Farklı çalıştırmaların paketler tarafından alınan farklı rotalara yol açabileceğini fark ediyoruz.

Traceroute A

```
user@AttackBox$ traceroute tryhackme.comtraceroute to tryhackme.com (17
2.67.69.208), 30 hops max, 60 byte packets
 1  ec2-3-248-240-5.eu-west-1.compute.amazonaws.com (3.248.240.5)  2.66
3 ms * ec2-3-248-240-13.eu-west-1.compute.amazonaws.com (3.248.240.13)
7.468 ms
 2  100.66.8.86 (100.66.8.86)  43.231 ms 100.65.21.64 (100.65.21.64)  18.886 m
s 100.65.22.160 (100.65.22.160)  14.556 ms
 3  * 100.66.16.176 (100.66.16.176)  8.006 ms *
 4  100.66.11.34 (100.66.11.34)  17.401 ms 100.66.10.14 (100.66.10.14)  23.614 ms
100.66.19.236 (100.66.19.236)  17.524 ms
 5  100.66.7.35 (100.66.7.35)  12.808 ms 100.66.6.109 (100.66.6.109)  14.791 ms
*
 6  100.65.14.131 (100.65.14.131)  1.026 ms 100.66.5.189 (100.66.5.189)  19.246 m
s 100.66.5.243 (100.66.5.243)  19.805 ms
 7  100.65.13.143 (100.65.13.143)  14.254 ms 100.95.18.131 (100.95.18.131)  0.944
ms 100.95.18.129 (100.95.18.129)  0.778 ms
 8  100.95.2.143 (100.95.2.143)  0.680 ms 100.100.4.46 (100.100.4.46)  1.392 m
s 100.95.18.143 (100.95.18.143)  0.878 ms
 9  100.100.20.76 (100.100.20.76)  7.819 ms 100.92.11.36 (100.92.11.36)  18.669
ms 100.100.20.26 (100.100.20.26)  0.842 ms
10  100.92.11.112 (100.92.11.112)  17.852 ms * 100.92.11.158 (100.92.11.158)  16.68
7 ms
11  100.92.211.82 (100.92.211.82)  19.713 ms 100.92.0.126 (100.92.0.126)  18.603
ms 52.93.112.182 (52.93.112.182)  17.738 ms
12  99.83.69.207 (99.83.69.207)  17.603 ms  15.827 ms  17.351 ms
13  100.92.9.83 (100.92.9.83)  17.894 ms 100.92.79.136 (100.92.79.136)  21.250
ms 100.92.9.118 (100.92.9.118)  18.166 ms
```

```
14  172.67.69.208 (172.67.69.208)  17.976 ms  16.945 ms 100.92.9.3 (100.92.9.
3)  17.709 ms
```

Yukarıdaki traceroute çıktısında 14 numaralı satır var; her satır bir yönlendiriciyi/atlamayı temsil ediyor. Sistemimiz TTL'si 1 olarak ayarlanmış üç paket gönderir, ardından TTL'si 2 olarak ayarlanmış üç paket gönderir ve bu böyle devam eder. Ağ topolojisine bağlı olarak, paketin izlediği rotaya bağlı olarak en fazla 3 farklı yönlendiriciden yanıt alabiliriz. 12 numaralı satırı ele alalım, listelenen IP adresine sahip on ikinci yönlendirici paketi üç kez düşürdü ve bir ICMP time exceeded in-transit mesajı gönderdi. 12 99.83.69.207 (99.83.69.207) 17.603 ms 15.827 ms 17.351 ms satırı, her bir yanıtın sistemimize ulaşması için geçen süreyi milisaniye cinsinden göstermektedir.

Öte yandan, üçüncü satırda yalnızca tek bir yanıt aldığımızı görebiliriz. Çıktıdaki iki yıldız 3 * 100.66.16.176 (100.66.16.176) 8.006 ms * sistemimizin beklenen iki ICMP time exceeded in-transit mesajını almadığını gösterir.

Son olarak, çıktının ilk satırında, AttackBox'tan ayrılan paketlerin farklı rotalar izlediğini görebiliriz. TTL'nin bir olmasına yanıt veren iki yönlendirici görebiliyoruz. Sistemimiz beklenen üçüncü ICMP mesajını hiç almadı.

Traceroute B

```
user@AttackBox$ traceroute tryhackme.comtraceroute to tryhackme.com (10
4.26.11.229), 30 hops max, 60 byte packets
 1  ec2-79-125-1-9.eu-west-1.compute.amazonaws.com (79.125.1.9)  1.475 ms
* ec2-3-248-240-31.eu-west-1.compute.amazonaws.com (3.248.240.31)  9.4
56 ms
 2  100.65.20.160 (100.65.20.160)  16.575 ms 100.66.8.226 (100.66.8.226)  23.2
41 ms 100.65.23.192 (100.65.23.192)  22.267 ms
 3  100.66.16.50 (100.66.16.50)  2.777 ms 100.66.11.34 (100.66.11.34)  22.288 m
s 100.66.16.28 (100.66.16.28)  4.421 ms
 4  100.66.6.47 (100.66.6.47)  17.264 ms 100.66.7.161 (100.66.7.161)  39.562 ms
100.66.10.198 (100.66.10.198)  15.958 ms
 5  100.66.5.123 (100.66.5.123)  20.099 ms 100.66.7.239 (100.66.7.239)  19.253
ms 100.66.5.59 (100.66.5.59)  15.397 ms
 6  * 100.66.5.223 (100.66.5.223)  16.172 ms 100.65.15.135 (100.65.15.135)  0.42
4 ms
```

```
 7  100.65.12.135 (100.65.12.135)  0.390 ms 100.65.12.15 (100.65.12.15)  1.045 ms
100.65.14.15 (100.65.14.15)  1.036 ms
 8  100.100.4.16 (100.100.4.16)  0.482 ms 100.100.20.122 (100.100.20.122)  0.795
ms 100.95.2.143 (100.95.2.143)  0.827 ms
 9  100.100.20.86 (100.100.20.86)  0.442 ms 100.100.4.78 (100.100.4.78)  0.347
ms 100.100.20.20 (100.100.20.20)  1.388 ms
10  100.92.212.20 (100.92.212.20)  11.611 ms 100.92.11.54 (100.92.11.54)  12.675
ms 100.92.11.56 (100.92.11.56)  10.835 ms
11  100.92.6.52 (100.92.6.52)  11.427 ms 100.92.6.50 (100.92.6.50)  11.033 ms 1
00.92.210.50 (100.92.210.50)  10.551 ms
12  100.92.210.139 (100.92.210.139)  10.026 ms 100.92.6.13 (100.92.6.13)  14.58
6 ms 100.92.210.69 (100.92.210.69)  12.032 ms
13  100.92.79.12 (100.92.79.12)  12.011 ms 100.92.79.68 (100.92.79.68)  11.318
ms 100.92.80.84 (100.92.80.84)  10.496 ms
14  100.92.9.27 (100.92.9.27)  11.354 ms 100.92.80.31 (100.92.80.31)  13.000 m
s 52.93.135.125 (52.93.135.125)  11.412 ms
15  150.222.241.85 (150.222.241.85)  9.660 ms 52.93.135.81 (52.93.135.81)  10.
941 ms 150.222.241.87 (150.222.241.87)  16.543 ms
16  100.92.228.102 (100.92.228.102)  15.168 ms 100.92.227.41 (100.92.227.41)
10.134 ms 100.92.227.52 (100.92.227.52)  11.756 ms
17  100.92.232.111 (100.92.232.111)  10.589 ms 100.92.231.69 (100.92.231.69)  1
6.664 ms 100.92.232.37 (100.92.232.37)  13.089 ms
18  100.91.205.140 (100.91.205.140)  11.551 ms 100.91.201.62 (100.91.201.62)  1
0.246 ms 100.91.201.36 (100.91.201.36)  11.368 ms
19  100.91.205.79 (100.91.205.79)  11.112 ms 100.91.205.83 (100.91.205.83)  11.0
40 ms 100.91.205.33 (100.91.205.33)  10.114 ms
20  100.91.211.45 (100.91.211.45)  9.486 ms 100.91.211.79 (100.91.211.79)  13.69
3 ms 100.91.211.47 (100.91.211.47)  13.619 ms
21  100.100.6.81 (100.100.6.81)  11.522 ms 100.100.68.70 (100.100.68.70)  10.181
ms 100.100.6.21 (100.100.6.21)  11.687 ms
22  100.100.65.131 (100.100.65.131)  10.371 ms 100.100.92.6 (100.100.92.6)  10.9
39 ms 100.100.65.70 (100.100.65.70)  23.703 ms
23  100.100.2.74 (100.100.2.74)  15.317 ms 100.100.66.17 (100.100.66.17)  11.492
ms 100.100.88.67 (100.100.88.67)  35.312 ms
24  100.100.16.16 (100.100.16.16)  19.155 ms 100.100.16.28 (100.100.16.28)  19.147
ms 100.100.2.68 (100.100.2.68)  13.718 ms
```

```
25  99.83.89.19 (99.83.89.19)  28.929 ms *  21.790 ms
26  104.26.11.229 (104.26.11.229)  11.070 ms  11.058 ms  11.982 ms
```

Traceroute programının ikinci çalıştırılmasında, paketlerin bu kez 26 yönlendiriciden geçerek daha uzun bir rota izlediğini fark ettik. Eğer ağınızdaki bir sisteme traceroute çalıştırıyorsanız, rotanın değişmesi pek mümkün olmayacaktır. Ancak, paketlerin ağımız dışındaki diğer yönlendiriciler üzerinden gitmesi gerektiğinde rotanın sabit kalmasını bekleyemeyiz.

Özetlemek gerekirse, aşağıdakileri fark edebiliriz:

- Sisteminiz ile hedef sistem arasındaki atlama/yönlendirici sayısı traceroute'u çalıştırdığınız zamana bağlıdır. Aynı ağ üzerinde olsanız veya traceroute komutunu kısa bir süre içinde tekrarlasanız bile, paketlerinizin her zaman aynı rotayı izleyeceğinin garantisi yoktur.
- Bazı yönlendiriciler genel bir IP adresi döndürür. Amaçlanan sızma testinin kapsamına bağlı olarak bu yönlendiricilerden birkaçını inceleyebilirsiniz.
- Bazı yönlendiriciler yanıt vermez.

Sorular

Soru ⇒ Traceroute A'da, tryhackme.com adresine ulaşmadan önceki son yönlendiricinin/atlamanın IP adresi nedir (İpucu ⇒ Birden fazla IP adresi varsa, yanıt veren ilk IP adresini belirtin.)?

Cevap ⇒ 172.67.69.208

Soru ⇒ Traceroute B'de, tryhackme.com adresine ulaşmadan önceki son yönlendiricinin/atlamanın IP adresi nedir (İpucu ⇒ Birden fazla IP adresi varsa, yanıt veren ilk IP adresini belirtin.)?

Cevap ⇒ 104.26.11.229

Soru ⇒ Traceroute B'de, iki sistem arasında kaç yönlendirici vardır?

Cevap ⇒ 26

Soru ⇒ Henüz başlatılmadıysa ekli sanal makineyi Görev 3'ten başlatın. AttackBox üzerinde traceroute MACHINE_IP komutunu çalıştırın. AttackBox ile hedef VM arasında kaç tane yönlendirici/atlama olduğunu kontrol edin.(İpucu ⇒ Eğer traceroute yüklü değilse, apt install traceroute ile yükleyebilirsiniz.)

Cevap ⇒ Cevap Gerekmemektedir.

Task 5 Telnet (Görev 5 Telnet)

TELNET (Teletype Network) protokolü 1969 yılında bir komut satırı arayüzü (CLI) aracılığıyla uzak bir sistemle iletişim kurmak için geliştirilmiştir. Bu nedenle, telnet komutu uzaktan yönetim için TELNET protokolünü kullanır. Telnet tarafından kullanılan varsayılan bağlantı noktası 23'tür. Güvenlik açısından bakıldığında, telnet kullanıcı adları ve parolalar da dahil olmak üzere tüm verileri açık metin olarak gönderir. Açık metin olarak göndermek, iletişim kanalına erişimi olan herhangi birinin oturum açma kimlik bilgilerini çalmasını kolaylaştırır. Güvenli alternatif SSH (Secure SHell) protokolüdür.

Ancak, telnet istemcisi basitliği ile başka amaçlar için de kullanılabilir. Telnet istemcisinin TCP protokolüne dayandığını bilerek, Telnet'i herhangi bir servise bağlanmak ve banner'ını almak için kullanabilirsiniz. telnet MACHINE_IP PORT kullanarak, TCP üzerinde çalışan herhangi bir servise bağlanabilir ve hatta şifreleme kullanmadığı sürece birkaç mesaj alışverişinde bulunabilirsiniz.

Diyelim ki 80 numaralı bağlantı noktasını dinleyen bir web sunucusu hakkında daha fazla bilgi edinmek istiyoruz. Sunucuya 80 numaralı bağlantı noktasından bağlanırız ve ardından HTTP protokolünü kullanarak iletişim kurarız. HTTP protokolüne dalmanıza gerek yoktur; sadece GET / HTTP/1.1 komutunu vermeniz yeterlidir. Varsayılan dizin sayfasından başka bir şey belirtmek için GET /page.html HTTP/1.1 komutunu verebilirsiniz; bu komut page.html dosyasını isteyecektir. Ayrıca uzak web sunucusuna iletişim için HTTP sürüm 1.1'i kullanmak istediğimizi belirttik. Hata yerine geçerli bir yanıt almak için host: example için bir değer girmeniz ve iki kez enter tuşuna basmanız gerekir. Bu adımları uygulamak istenen dizin sayfasını sağlayacaktır.

```
pentester@TryHackMe$ telnet MACHINE_IP 80Trying MACHINE_IP...
Connected to MACHINE_IP.
Escape character is '^]'.
GET / HTTP/1.1
host: telnet
```

```
HTTP/1.1 200 OK
Server: nginx/1.6.2
Date: Tue, 17 Aug 2021 11:13:25 GMT
Content-Type: text/html
Content-Length: 867
Last-Modified: Tue, 17 Aug 2021 11:12:16 GMT
Connection: keep-alive
ETag: "611b9990-363"
Accept-Ranges: bytes



...
```

Bizim için özellikle ilgi çekici olan, kurulu web sunucusunun türünü ve sürümünü keşfetmektir, Sunucu: nginx/1.6.2. Bu örnekte, bir web sunucusuyla iletişim kurduk, bu nedenle temel HTTP komutlarını kullandık. Bir posta sunucusuna bağlanırsak, SMTP ve POP3 gibi protokole göre uygun komutları kullanmamız gerekir.

Sorular

Soru ⇒ Ekli sanal makine henüz başlatılmadıysa Görev 3'ten başlatın. AttackBox'ta terminali açın ve 80 numaralı bağlantı noktasından VM'ye bağlanmak için telnet istemcisini kullanın. Çalışan sunucunun adı nedir?

Cevap ⇒ Apache

Soru ⇒ Çalışan sunucunun sürümü nedir (VM'nin 80 numaralı portunda)?

Cevap ⇒ 2.4.61

Task 6 Netcat (Görev 6 Netcat)

Netcat ya da kısaca nc, bir pentester için çok değerli olabilecek farklı uygulamalara sahiptir. Netcat hem TCP hem de UDP protokollerini destekler. Bir

dinleme portuna bağlanan bir istemci olarak işlev görebilir; alternatif olarak, seçtiğiniz bir portu dinleyen bir sunucu olarak hareket edebilir. Bu nedenle, TCP veya UDP üzerinden basit bir istemci veya sunucu olarak kullanabileceğiniz kullanışlı bir araçtır.

İlk olarak, Telnet ile yaptığınız gibi bir sunucuya bağlanabilir ve önceki telnet MACHINE_IP PORT'umuza oldukça benzeyen nc MACHINE_IP PORT'u kullanarak banner'ını toplayabilirsiniz. GET satırından sonra SHIFT+ENTER tuşlarına basmanız gerekebileceğini unutmayın.

```
pentester@TryHackMe$ nc MACHINE_IP 80GET / HTTP/1.1
host: netcat

HTTP/1.1 200 OK
Server: nginx/1.6.2
Date: Tue, 17 Aug 2021 11:39:49 GMT
Content-Type: text/html
Content-Length: 867
Last-Modified: Tue, 17 Aug 2021 11:12:16 GMT
Connection: keep-alive
ETag: "611b9990-363"
Accept-Ranges: bytes
...
```

Yukarıda gösterilen terminalde, nc MACHINE_IP 80 kullanarak MACHINE_IP port 80'e bağlanmak için netcat kullandık. Ardından, GET / HTTP/1.1 kullanarak varsayılan sayfa için bir get yayınladık; hedef sunucuya istemcimizin HTTP sürüm 1.1'i desteklediğini belirtiyoruz. Son olarak, ana bilgisayarımıza bir isim vermemiz gerekiyor, bu yüzden yeni bir satır ekledik, ana bilgisayar: netcat; ana bilgisayarınıza herhangi bir isim verebilirsiniz, çünkü bunun bu alıştırma üzerinde bir etkisi yoktur.

Aldığımız Server: nginx/1.6.2 çıktısına dayanarak, 80 numaralı bağlantı noktasında, gelen bağlantıları dinleyen Nginx sürüm 1.6.2'ye sahip olduğumuzu söyleyebiliriz.

Bir TCP portunu dinlemek ve başka bir sistemdeki dinleme portuna bağlanmak için netcat kullanabilirsiniz.

Bir port açmak ve onu dinlemek istediğiniz sunucu sisteminde, nc -lp 1234 veya daha iyisi, Linux odasından hatırlayacağınız gibi nc -v -l -n -p 1234'e eşdeğer olan nc -vnlp 1234 komutunu verebilirsiniz. Port numarasından önce doğrudan -p geldiği sürece harflerin tam sırası önemli değildir.

option	meaning
-l	Listen mode (Dinleme modu)
-p	Specify the Port number (Bağlantı noktası numarasını belirtin)
-n	Numeric only; no resolution of hostnames via DNS (Yalnızca sayısal; ana bilgisayar adlarının DNS aracılığıyla çözümlenmesi yok)
-v	Verbose output (optional, yet useful to discover any bugs) (Ayrıntılı çıktı (isteğe bağlı, ancak herhangi bir hatayı keşfetmek için yararlıdır))
-vv	Very Verbose (optional) (Çok Ayrıntılı (isteğe bağlı)
-k	Keep listening after client disconnects (İstemci bağlantısı kesildikten sonra dinlemeye devam et)

Notlar

- dinlemek istediğiniz bağlantı noktası numarasından hemen önce -p seçeneği görünmelidir.
- n seçeneği DNS aramalarını ve uyarılarını önleyecektir.
- 1024'ten küçük bağlantı noktası numaralarını dinlemek için root ayrıcalıkları gerekir.

İstemci tarafında, nc MACHINE_IP PORT_NUMBER komutunu verirsiniz. İşte yankı için nc kullanımına bir örnek. Sunucuyla başarılı bir şekilde bağlantı kurduktan sonra, istemci tarafında yazdığınız her şey sunucu tarafında yankılanacaktır ve bunun tersi de geçerlidir.

Aşağıdaki örneği ele alalım. Sunucu tarafında, 1234 numaralı portu dinleyeceğiz. Bunu nc -vnlp 1234 komutuyla başarabiliriz (nc -lvnp 1234 ile aynı). Bizim durumumuzda, dinleme sunucusu MACHINE_IP IP adresine sahiptir, bu nedenle nc MACHINE_IP 1234 komutunu çalıştırarak istemci tarafından ona bağlanabiliriz. Bu kurulum, bir tarafta yazdığınız her şeyi TCP tünelinin diğer tarafına yankılayacaktır. İşlemin bir kaydını aşağıda bulabilirsiniz. Dinleme sunucusunun ekranın sol tarafında olduğuna dikkat edin.

Soru ⇒ Sanal makineyi başlatın ve AttackBox'ı açın. AttackBox yüklendikten sonra, VM port 21'e bağlanmak için Netcat kullanın. Çalışan sunucunun sürümü nedir?

Cevap ⇒ 0.17

Task 7 Putting It All Together (Görev 7 Her Şeyi Bir Araya Getirmek)

Bu odada birçok farklı aracı ele aldık. İlkel bir ağ ve sistem tarayıcısı oluşturmak için bunlardan birkaçını bir kabuk betiği aracılığıyla bir araya getirmek kolaydır. Hedefe giden yolu haritalamak için traceroute, hedef sistemin ICMP Echo'ya yanıt verip vermediğini kontrol etmek için ping ve hangi bağlantı noktalarının açık ve erişilebilir olduğunu kontrol etmek için telnet kullanabilirsiniz. Mevcut tarayıcılar, nmap ile sonraki dört odada göreceğimiz gibi, bunu çok daha gelişmiş ve sofistike seviyelerde yaparlar.

Command	Example
ping	`ping -c 10 MACHINE_IP` on Linux or macOS
ping	`ping -n 10 MACHINE_IP` on MS Windows
traceroute	`traceroute MACHINE_IP` on Linux or macOS
tracert	`tracert MACHINE_IP` on MS Windows
telnet	`telnet MACHINE_IP PORT_NUMBER`
netcat as client	`nc MACHINE_IP PORT_NUMBER`
netcat as server	`nc -lvnp PORT_NUMBER`

Bunlar temel araçlar olmakla birlikte, çoğu sistemde hazır olarak bulunurlar. Özellikle, bir web tarayıcısı hemen hemen her bilgisayarda ve akıllı telefonda yüklüdür ve alarm vermeden keşif yapmak için cephaneliğinizde önemli bir araç olabilir. Geliştirici Araçları hakkında daha derin bilgi edinmek istiyorsanız, Walking An Application'a katılmanızı öneririz.

Operating System	Developer Tools Shortcut
Linux or MS Windows	`Ctrl+Shift+I`

macOS	`Option + Command + I`

Soru ⇒ Daha sofistike araçlara geçmeden önce bu odada sunduğumuz farklı temel ancak gerekli araçlar üzerinde ustalık kazandığınızdan emin olun.

Cevap ⇒ Cevap Gerekmemektedir.