

SİBER GÜVENLİK - HACKING - ATÖLYESİ

Onur AKTAŞ

Misafir Yazarlar

Bülent ARSAL

Korhan GÜRLER

Mahmut Esat YILDIRIM



İÇİNDEKİLER

Kimler İçin?	xi
Bu Kitabı Nasıl Okumalıyım?	xiii
Atölye Kullanımı ve Siber Kuvvet (“siberkuvvet.com”)	xvii
Feragatname	xix
Uyarı.....	xix
Teşekkürler	xxi
Temel Yazılım Bilgisi ve Python	1
Ağ ve Güvenlik Bilgisi	23
Giriş	23
İnternet nedir?	27
Ağları Oluşturan Bileşenler	28
Ağların Ağı Nasıl Çalışılır?	32
İnternet Servis Sağlayıcıları Nasıl Çalışır?.....	36
OSI Katmanları	38
Kavramlar, Terimler, Cihazlar.....	44
Temel Linux Bilgisi	57
Hazırlık.....	57
Sanal Makine Kurulumu	59
Temel Linux Komutları	67
Kullanıcı Yönetimi	77
Paket Yönetimi	79
Girdi, Çıktı ve Metin İşleme Yöntemleri	81
Basit Bash Scripting	84
Sonuç.....	87
Okuma Parçası: Ağ’ın Söyledikleri	89

1.Kaba-Kuvvet (Brute-Force) Saldırıları	95
1.1 Genel Bilgi	95
1.2 Saldırı Açıklaması.....	99
1.3 Gerçek Hayat Senaryoları	108
1.4 Atölye	110
1.5 Okuma Listesi	111
1.6 Cevap Verilmesi Gereken Sorular	114
Okuma Parçası: Parola Saldırıları.....	115
2.Sosyal Mühendislik Saldırıları	129
2.1 Genel Bilgi	129
2.2 Saldırı Açıklaması.....	133
2.3 Gerçek Hayat Senaryoları	142
2.4 Atölye	144
2.5 Okuma Listesi	145
2.6 Cevap Verilmesi Gereken Sorular	146
Okuma Parçası: Bilgi Toplama	147
3. Dosya Yerleştirme Saldırıları: Yerel	151
3.1 Genel Bilgi	151
3.2 Saldırı Açıklaması.....	154
3.3 Gerçek Hayat Senaryoları	159
3.4 Atölye.....	160
3.5 Okuma Listesi	161
3.5 Cevap Verilmesi Gereken Sorular.....	162
Okuma Parçası: Çantanızda Bulunması	
Gereken Çakı – Ncat	163

4.Bu Bir Uyarıdır: Tarayıcıdaki Kodlar!	169
4.1 Genel Bilgi	169
4.2 Saldırı Açıklaması.....	177
4.3 Gerçek Hayat Senaryoları	180
4.4 Atölye.....	181
4.5 Okuma Listesi	182
4.6 Cevap Verilmesi Gereken Sorular.....	183
Okuma Parçası: Reverse ve Bind Shell	185
5.Yumuşak Yere Vurun : Yetkisiz Erişimler	189
5.1 Genel Bilgi	189
Senaryo 1	189
Senaryo 2	192
5.2 Saldırı Açıklaması.....	195
Senaryo 1	195
Senaryo 2	200
5.3 Gerçek Hayat Senaryoları	206
5.4 Atölye.....	207
5.5 Okuma Listesi.....	208
5.6 Cevap Verilmesi Gereken Sorular.....	209
Okuma Parçası: Soket Kullanarak Uzaktan Komut Çalıştırma	211
6.Yarış, Yarış, Yarış.....	221
6.1 Genel Bilgi	221
6.2 Saldırı Açıklaması.....	223
6.3 Gerçek Hayat Senaryoları	233
6.4 Atölye.....	234
6.5 Okuma Listesi.....	235
6.6 Cevap Verilmesi Gereken Sorular.....	236
Okuma Parçası: İnternetin Çocukluğu ve Olgunlaşması	237

7.Dijital Ortamda Fazla Yetki Paylaşmak İyi Değildir: Cross Domain Origin Zafiyeti	241
7.1 Genel Bilgi	241
7.2 Saldırı Açıklaması.....	245
7.3 Gerçek Hayat Senaryoları	254
7.4 Atölye.....	255
7.5 Okuma Listesi	256
7.6 Cevap Verilmesi Gereken Sorular	257
8.Birisi SQL'mi Dedi?.....	259
8.1 Genel Bilgi	259
8.2 Saldırı Açıklaması.....	266
8.3 Gerçek Hayat Senaryoları	275
8.4 Atölye.....	277
8.5 Okuma Listesi	278
8.6 Cevap Verilmesi Gereken Sorular	279
Okuma Parçası: Bilmeden Olmazlar.....	281
9.Kitaptaki Son Uyarı!	283
9.1 Genel Bilgi	283
9.2 Saldırı Açıklaması.....	286
9.3 Gerçek Hayat Senaryoları	289
9.4 Atölye.....	290
9.5 Okuma Listesi	291
9.6 Cevap Verilmesi Gereken Sorular	292
Okuma Parçası: BeEf XSS Framework	293
10.Körlemesine SQL	301
10.1 Genel Bilgi	301
10.2 Saldırı Açıklaması.....	306
10.3 Gerçek Hayat Senaryoları	309
10.4 Atölye.....	310

10.5 Okuma Listesi.....	311
10.6 Cevap Verilmesi Gereken Sorular.....	312
Okuma Parçası: Metasploit	313
11.Adım Adım Bellek Taşıma (Buffer Overflow)	319
11.1 Genel Bilgi	319
11.2 Saldırı Açıklaması.....	334
11.3 Gerçek Hayat Senaryoları	351
11.4 Atölye.....	353
11.5 Okuma Listesi.....	354
11.6 Cevap Verilmesi Gereken Sorular.....	355
Okuma Parçası: Sömürüden Sonra Sömürü Gelir	357
Kali Linux Kullanımı.....	361
Son Söz.....	371

Bu Kitabı Nasıl Okumalıyım ?

Elinizde tuttuđunuz bu kitap altı seneden fazla bir süre boyunca gerekleřtirilen onlarca gvenlik testi sonucu en ok karřılařılan ve teknik olarak farklı senaryolardan esinlenerek, konusunda uzman birok kiřiden grř ve neri alınarak, yetkin bir gvenlik uzmanı olmak isteyen kiřiler iin yazılmıřtır. Kitabın cretli olmasının en byk sebeplerinin bařında atlye masrafları gelmektedir. Kitaptan gelen tm gelirler, siber gvenlik eđitimleri ve Siberkuvvet'in geliřtirilmesi iin harcanacaktır.

Teknik bir kitap ne kadar srkleyici olur bilemiyorum ama elimden geldiđince okuyucuyu sıkmayan, eđlenceli ve temel-orta dzeyde bilgiler ieren bir siber gvenlik kitabı oluřturmaya alıřtım.

Kitabın ilk blmlerinde en basit yntemler ile gerekleřtirilebilecek, anlařılması kolay fakat etkisi olduka byk siber saldırılara yer verdim. İlk blmlerde đrenilen temel bilgiler ilerleyen blmlerdeki siber saldırıların yntemlerini anlamak iin olduka nemli. Bu yzden ilk blmlerde anlatılan konulara yeterince hakim deđilseniz gerekli n arařtırmalardan sonra kitaba bařlamanızı neririm.

Her blm kendi ierisinde 5 ana bařlıktan oluřmaktadır. **Genel Bilgi** blmnde saldırı yntemini anlamak iin bilmeniz gereken teknik bilgileri bulabilirsiniz. Genel olarak saldırı ile alakalı teknolojilerin kullanım yntemlerini anlatan bu blm, saldırı yapacađınız sistemi anlamanız ve saldırının yntemlerini daha iyi analiz etmeniz amacıyla ekledim.

Saldırı aıklaması blmnde atlye ortamında bulunmayan bir sisteme yapılan rnek bir saldırı anlatıyorum. Bu blm rnek bir saldırının ve hedef sistemin neler olabileceđi hakkında detaylı bilgiler ieriyor. Bylece atlye blmnde kendiniz rnek yapmadan nce rnek bir saldırının nasıl yapıldıđı hakkında bilgi edinebilirsiniz.

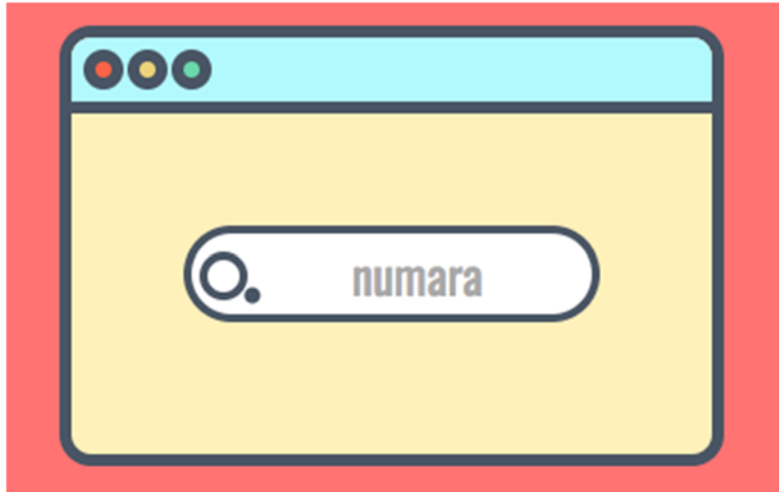
Atlye blm, alıřtırma amalı yapmanız istenen rnekleri anlatan blm. Bu blm bana gre kitaptaki en nemli blm. ođu kaynak iin internet zerinde zellikle İngilizce birok kaynak bulabilirsiniz. Fakat đrendiklerinizi deneyebileceđiniz, alıřtırma yapabileceđiniz, zafiyet

içeren Türkçe sistemlerin olduğu bir web platformu siber güvenlik uzmanı olma yolunda oldukça işinize yarayacaktır. Bu bölümün nasıl kullanılacağını ayrı bir bölümde detaylı olarak bulabilirsiniz.

Gerçek hayat hikayeleri benim ve konusunda uzman birçok profesyonel arkadaşımın konu ile ilgili görüşlerini ve gerçek senaryoları içermektedir. Bu bölümde hem saldırıları gerçekleştirirken dikkat edilmesi gereken noktalar hem de gerçek hayatta karşınıza çıkabilecek saldırı yöntemleri ile ilgili sorunlardan bahsediyorum. Güvenlik testleri gerçekleştirirken başınıza sıkıntılı durumların gelmesini önleyeceğini düşünüyorum.

Okuma Listesi bölümlerinde saldırıda kullanılan altyapıya ve/veya teknolojilere ait daha detaylı bilgiler edinmeniz için sizlere hazırladığım kaynakları ekledim. Bu bölümdekileri okumanız ve anlamamanızın yetkin güvenlik uzmanı olmanız açısından çok önemli olduğunu belirtmek isterim. Koca bir bağlantıyı kitaptan okuyup da internet üzerinden erişmenin zor olduğunu tahmin ediyorum. Bu yüzden okuma listesindeki bağlantıların başında ilgili bağlantının 4 rakamlı bir numarasını yazdım. Bu numarayı “ayna.siberkuvvet.com” adresine girip yazarak ilgili bağlantıya kolayca erişebilirsiniz.

ayna.siberkuvvet.com



İleride bağlantının çalışmaması ihtimalini de göz önüne alarak aynı zamanda sitenin bir kopyasını da sisteme kayıt ettim. Eğer bağlantıda bir sorun varsa kopyası alınan siteye yine “ayna.siberkuvvet.com” üzerinden erişebilirsiniz.

Son olarak “**Cevap Verilmesi Gerekenler**” bölümünde ilgili konuyu derinlemesine anladığınızdan emin olmanız adına cevap vermeniz gereken soruları listeledim. Okuma listesini okuduktan sonra kendinizi bu sorular ile kontrol edebilirsiniz. Bu soruların cevabını Siber Kuvvet (“siberkuvvet.com”) web adresindeki soru/cevap bölümünde sorabilirsiniz.

Kitabın tamamında Türkçe kelimeler kullanmaya özen gösterdim. Okuma listesinde de Türkçe kaynaklardan yararlanmaya çalıştım. Fakat İngilizce araştırma yapmak isteyen olursa diye parantez içerisinde tanımların İngilizce karşılığına da yer verdim.

Web Application Firewall (WAF)

Web sayfaları ve uygulamaları için özel tasarlanmış, oldukça gelişmiş işlemler yapabilen cihazlardır. Bir web sayfasından dönen cevaplardaki veya web sayfasına giden isteklerdeki paketlerin, her türlü parametresi ile oynayabilme (veri ekleyen, veri çıkaran, değiştiren) gibi bir yeteneği vardır. Bu da şu demektir: kızdığı bir durum varsa onu silebilir, değiştirebilir.

Web application firewall cihazları uygulaması ve işletmesi en zor sistemlerden biridir. Çünkü bu sistemi yöneten kişilerin korudukları web sayfası ve uygulamasının nasıl çalıştığını, hangi servisleri kullandığını çok iyi bilmesi gerekmektedir. Ayrıca sayfada ya da uygulamada yapılacak güncellemelere göre hemen WAF'ta gerekli düzenlemeyi yapması elzemdir. Aksi takdirde normal kullanıcı isteklerini de engelleme ihtimali çok yüksektir. Bu nedenle WAF sistemleri engelleme modunda doğrudan kullanılmaz, monitör modda yani öğrenme modunda bir süre çalıştırılarak bilgi toplaması sağlanır, sonrasında gerekli kurallar düzenlenerek engelleme yapmaya başlar.

Öğrenme modundan sonra, yazılımın kendisinde bir zafiyet varsa bile düzgün konfigürasyonu yapılmış bir WAF sistemi ile saldırganların bu zafiyeti kullanması çok kolay bir şekilde engellenebilir. Bir kere konfigürasyonu yapıldıktan sonra bu sistemi atlatmak oldukça zordur. Bu sistemlerde OWASP'in en güncel 10 saldırısı gibi birçok saldırı için varsayılan olarak engelleme gelmektedir. Yani siz bir ' (tırnak işareti) attığınızda WAF bunu farkedecektir.

Bununla birlikte database firewall, DNS firewall'lar da bulunmaktadır. WAF ile benzer mantıklarda çalıştığından dolayı sadece kavram olarak bilinmesi bu aşamada yeterli olacaktır.

Data Leak/Leakage Prevention (DLP)

Bu sistemler daha çok içeriden dışarıya (kendi ağınızdan dışarıdaki bir ağa) veri sızıntısının tespit edilmesi ve engellenmesi için kullanılır. Host ve network DLP olarak iki çeşidi bulunur. Saldırganlar sızdıkları bir ağdan dışarıya bir veri çıkarmak istediğinde (kredi kartı bilgisi vb.)

5.3 Gerçek Hayat Senaryoları

Yetkisiz erişimler doğru yetkilendirme kontrollerinin yapılmadığı her alanda olabilir. Güvenlik testlerinde bu alanları bulmak sizin hayal gücünüze ve testleri yaptığınız ortamdaki teknik ve iş akışına ait bilginize dayalıdır. Yapacağınız en iyi şey, testlerden önce basitçe iş akışını anlamak ve kullanılan teknolojiye hakim olmaktır. Yalnızca web uygulamalarında dahi karşınıza çok farklı yöntem ve teknolojiler çıkabilir. Örneğin yalnızca JavaScript ile yazılmış web uygulamalarında sunucudan kontrol edilmiş değerlerin, istemciye gönderilip sonrasında tekrar sunucuya gönderilerek işleme alındığına birkaç kez şahit oldum. Sayıları artırılabilir bu örneklerin tespiti için bol bol okumanız, sistemlerin çalışma mantığına hakim olmanız ve denemekten çekinmemeniz büyük önem arz ediyor.

Çoğu otomatik zafiyet tarama yazılımları yetkisiz erişimleri doğrulayamaz. Bazı durumlarda doğrulamak için sistemlere müdahale etmeniz gerekebilir. Yönetici kullanıcısının parolasını değiştirdiğimizi hatırlayın. Eğer aktif çalışan bir sistemde güvenlik testi yapıyorsanız, başınız biraz ağrıyabilir. Öncesinde sizinle birlikte çalışan müşterinin size atadığı güvenlik testi sorumlusu ile iletişime geçip, olası bir problemde sistemlerin hızlı bir şekilde eski ayarlarına geri döndürüleceğine dair garanti almanız önemlidir.

Okuma Parçası: Soket Kullanarak Uzaktan Komut Çalıştırma

Bir önceki okuma parçasında uzaktan komut çalıştırmak için iki yöntemden bahsetmiştim. Her iki yöntemde de saldırgan tarafında veya hedef sistem üzerinde sömürülen servisin dışında bir port açılması gereklidir.

Uzaktaki hedef sisteme halihazırda bir port üzerinden erişebiliyorsak aynı portu kullanarak bağlantı açmaya çalışabiliriz. Bunu yaparken iki amacımız var: birincisi tespit edilmesini zorlaştırıyoruz, ikincisi ise yapması zevkli.

Örneğimizde TCP üzerinden işlemler yapacağız.

İşlemleri yapmadan öğrenmemiz gereken şeylerden birisi soketler. Linux'da soketler bir bağlantının en son noktasıdır. Linux'da açılan soketler aynı zamanda açılan soketi adresleyen bir dosya tanımlayıcısı (file descriptor) oluştururlar.

İkincisi ise dosya tanımlayıcıları. Dosya tanımlayıcıları Linux işletim sisteminde bir dosyaya ya da giriş veya çıkış işlemlerinin kaynağına erişmek için kullanılan işlemlere özgü oluşturulan dosyalardır.

Linux'da dosya tanımlayıcılarının oluşması için yapılması gereken işlemlerin içerisinde dosya işlemleri ve soket açma da var. Fakat yalnızca bir işlem başlatmak da bazı dosya tanımlayıcılarını oluşturur.

Örnek C kodu

```
Fd_test_1.c
#include <stdio.h>

int main (int argc, char *argv[])
{
    char in;
    scanf("%c",&in);
    return 0;
}
```

6.4 Atlye

Yarış durumunu oluşturmak için istek yapıldıkça bir dosya oluşturup hemen silen bir web sayfası hazırladık. Görevin - eğer kabul edersen - web sayfasının oluşturduğu dosya adını ve izin bilgilerini tespit edip silinmeden önce dosyayı okumak. Gerekli bilgiler siberkuvvet.com sayfasında mevcut.

Burada kitaptan duman çıkartıp yazıları silseydik havalı olurdu.

6.5 Okuma Listesi

Bu bölümdeki bağlantılara ayna.siberkuvvet.com üzerinden bağlantı kodlarını (başlarındaki dört haneli rakamlar) kullanarak kolayca ulaşabilirsiniz.

1046 - <https://www.mehmetince.net/web-uygulamalarinda-race-condition-zafiyeti-ve-etkileri/>

Race condition nedir ne değildir öğrenmek ve web uygulamalarında örneklerle açıklamasını okumak için tavsiye ederim.

1047 - http://members.comu.edu.tr/msahin/courses/isletim_sistemi_giris/ders04.pdf

İşletim sistemlerinde yarış durumlarını yazının bir bölümünde bulabilirsiniz.

1048 - <http://oguzkartal.net/blog/index.php/2015/11/18/concurrent-eszamanli-programlama-ve-race-condition-tehlikesi/>

Yazılım geliştirirken ortaya çıkan problemler ile ilgili örneği bu blog yazısından okuyabilirsiniz.

makale.php?makaleId=2 and 2=1

select BASLIK,YAZI from makalelerdiyebirtablo where id=2 and 2=1



Zafiyeti doğruladık. Sistemde SQL Injection zafiyeti var. Bilgi çekmeye başlayalım, öncelikle mysql sürümünü alalım. Bunun için iki sorguyu birleştiren union sorgusunu kullanacağız. İki sorguyu birleştirdiğimizde ilk sorgunun boş olmasını sağlayacağız, böylece ikinci sorgudan gelenleri ekranda gösterebileceğiz.

makale?makaleId=10 union all select null,@@version

*select BASLIK,YAZI from makalelerdiyebirtablo where id=10 union all
select null,@@version*



Bu sorgunu çalışması için ilk sorgudaki sütun isimlerinin sayısını doğru tahmin etmemiz lazım, görüldüğü gibi 2 değil. 3 adet sorguyu deneyelim.

makale.php?makaleId=10 union all select null,null,@@version

*select BASLIK,YAZI,ID from makalelerdiyebirtablo where id=10 union all
select null,null,@@version*



Substring fonksiyonu ise bir string değerinin yalnızca belirli bölümlerini almaya yarar. Bu fonksiyon üç parametre alır.

- Bir string
- Kaç karakter sonra string değerini bölecek
- Son olarak kaç karakter ilerleyecek

Örneğin ilk karakterden sonra beş karakter ileri git :

```
select substring('siberkuvvet',1,5)
```

```
substring('siberkuvvet',1,5)
```

```
siber
```

Ya da 6. karakterden sonra 6 karakter ileri git:

```
select substring('siberkuvvet',6,6)
```

```
substring('siberkuvvet',6,6)
```

```
kuvvet
```

Şimdi bir SQL komutu üzerinden bildiklerimizi hatırlayalım;

Basit bir SQL sorgusu, ID değeri 1 olan veritabanı kayıtlarını uyeler tablosundan getirdi.

```
select * from uyeler where id='1'
```

id	username	pass	email
1	admin	0	0

And koşulunda her iki tarafın da doğru olması gerekiyordu. Fakat aşağıdaki örnekte 1 değeri 2 değerine eşit olmadığından id değeri 1 olan kayıt olsa dahi hiçbir veri gelmedi.

10.4 Atlye

Atlyede bir srpriz sizi bekliyor. Veritabanı MySQL yerine ok daha **kk** bir **SQL** sunucusu. Hedef ise yine tablo adını ekmek. Tablo adında 2 rakam ve 2 harf var. Arařtırma, kod yazma, farklı dřnme iin gzel bir rnek olduėunu dřnyorum.

Genel resim şu şekilde:

The screenshot shows the Immunity Debugger interface for the process 'calc.exe'. The main window displays assembly code for the 'ntdll' module, with the instruction pointer (EIP) at address 77590279. The registers window shows the EIP register containing the address 77590279. The memory dump window shows the contents of the stack, including the return address 77590279 and the return value 0.

```

Immunity Debugger - calc.exe - [CPU - thread 00000E68, module ntdll]
File View Debug Plugins ImmLib Options Window Help Jobs
le m t w h c p k b z r ... s ? Code audit

77590213 C2 0600 RETI 8
77590217 90 NOP
77590218 90 NOP
77590219 90 NOP
7759021A 90 NOP
7759021D 8BFF MOV EDI,EDI
7759021E 58FF PUSH EBP
7759021F 8BEC MOV EBP,ESP
77590220 64101 10000000 MOV ERX,DWORD PTR FS:[10]
77590225 F3B0 240F0000 PUSH DWORD PTR DS:[ERX+24]
7759022C FF75 08 PUSH DWORD PTR SS:[EBP+8]
7759022F E3 CC85FAFF CML ntdll.ZuRemoveProcessDebu
77590234 E0 POP EBP
77590235 C2 0400 RETI 4
77590238 90 NOP
77590239 90 NOP
7759023A 90 NOP
7759023B 90 NOP
7759023C 90 NOP
7759023D 6A 08 PUSH 8
7759023F 03F5E477 PUSH ntdll.7754FB98
77590244 E8 6F9FAFF CALL ntdll.77546B88
7759024F 64101 10000000 MOV ERX,DWORD PTR FS:[10]
77590254 64A0 30 MOV ERX,DWORD PTR DS:[ERX+30]
77590252 8078 02 00 CMP BYTE PTR DS:[ERX+2],0
77590255 75 09 JNE SHORT ntdll.77590254
77590258 F405 0402FE7F 0 TEST BYTE PTR DS:[77FE0204],2
7759025F 74 28 JNE SHORT ntdll.77590259
77590261 64101 10000000 MOV ERX,DWORD PTR FS:[10]
77590267 F680 C80F0000 20 TEST BYTE PTR DS:[ERX+FC],20
7759026E 75 19 JNE SHORT ntdll.77590269
77590270 8B65 FC 00 MOV DWORD PTR SS:[EBP+4],0
77590274 E8 C762F9FF CALL ntdll.Db2BreakPoint

Registers (FPU)
EAX: 77FD9000
ECX: 00000000
EDX: 77590230 ntdll.DbUIRemoteBreak in
EBX: 00000000
ESP: 02B6FEA4
EBP: 02B6FECC
ESI: 00000000
EDI: 00000000
EIP: 77590279 ntdll.77590279
C 0 ES 0023 32bit 0(FFFFFFFF)
P 1 CS 0018 32bit 0(FFFFFFFF)
A 0 SS 0023 32bit 0(FFFFFFFF)
Z 1 DS 0023 32bit 0(FFFFFFFF)
S 0 FS 0038 32bit 77FD9000(FFF)
T 0 GS 0000 NULL
D 0
O 0 LastErr ERROR_SUCCESS (00000000)
EPL 00000246 (NO,IR,E,EE,NS,FE,GE,LE)
ST0 empty 0
ST1 empty 0
ST2 empty 0
ST3 empty 0
ST4 empty 0
ST5 empty 0
ST6 empty 0
ST7 empty 0
FST 0000 Cond 0 0 0 0 ESPU0Z01 (GT)
FCM 027F Prec NEAR,S3 Hask 1 1 1 1 1 1

Address Hex Dump ASCII
002D4658 03 00 00 00 20 00 00 00 *...hiz
002D4659 00 00 00 00 68 DE E8 7R ...hiz
002D465A 97 21 17 95 00 00 00 00 0?i...
002D465B F0 00 00 00 EC 90 00 00 ...i...
002D465C 00 00 00 00 B4 E3 1F 00 ...i...
002D465D 08 22 11 00 60 23 11 00 =24...
002D465E CD 56 74 94 82 84 00 =w...
002D465F C7 06 10 1C 01 00 00 00 ...0...
002D4660 15 03 11 00 70 20 11 00 2...+...
002D4661 20 3F 11 00 12 00 00 00 2...+...
002D4662 A9 31 11 00 A9 31 11 00 2...+...
002D4663 03 00 00 00 00 00 00 00 0...+...
002D4664 00 00 00 00 00 00 00 00 0...+...
002D4665 01 00 00 00 90 34 11 00 0...+...
002D4666 03 00 00 00 00 00 00 00 0...+...
002D4667 00 00 00 00 00 00 00 00 0...+...
002D4668 00 00 00 00 00 00 00 00 0...+...
002D4669 53 5C 49 00 78 5C 49 00 W...i...
002D466A 76 8C 49 00 80 5C 49 00 y...i...
002D466B 05 C9 42 00 10 21 4E 00 ...i...

Step into (F7) Paused

```

Henüz temel sorulara cevap vermedik. Sıradaki sorumuz şu: işlemci bir sonraki işlemin ne olacağını nereden biliyor?

İşlemci içerisinde *EIP* adı verilen register işlemci tarafından çalıştırılacak bir sonraki instruction'ın adresini tutar. Böylece işlemci her seferinde *EIP* register'ı içerisindeki adrese (RAM'e) gidip ilgili veriyi çalıştırır. Yazılımlar çalışırken sürekli olarak başka başka fonksiyonlar çalıştığından *EIP* değeri gerektiğinde stack'a kayıt edilir (burası biraz karışık detaylı olarak araştırmak isterseniz calling convention, function call, call, ret komutlarını ve kavramlarını araştırabilirsiniz). Resimlere biraz detaylı bakarak bunu görebiliriz. *EIP* içerisindeki adres CPU tarafından çalıştırılacak komutların olduğu adrestir.