

Yara ile Tehdit Avı

written by Mert SARICA | 1 September 2017

If you are looking for an English version of this article, please visit [here](#).

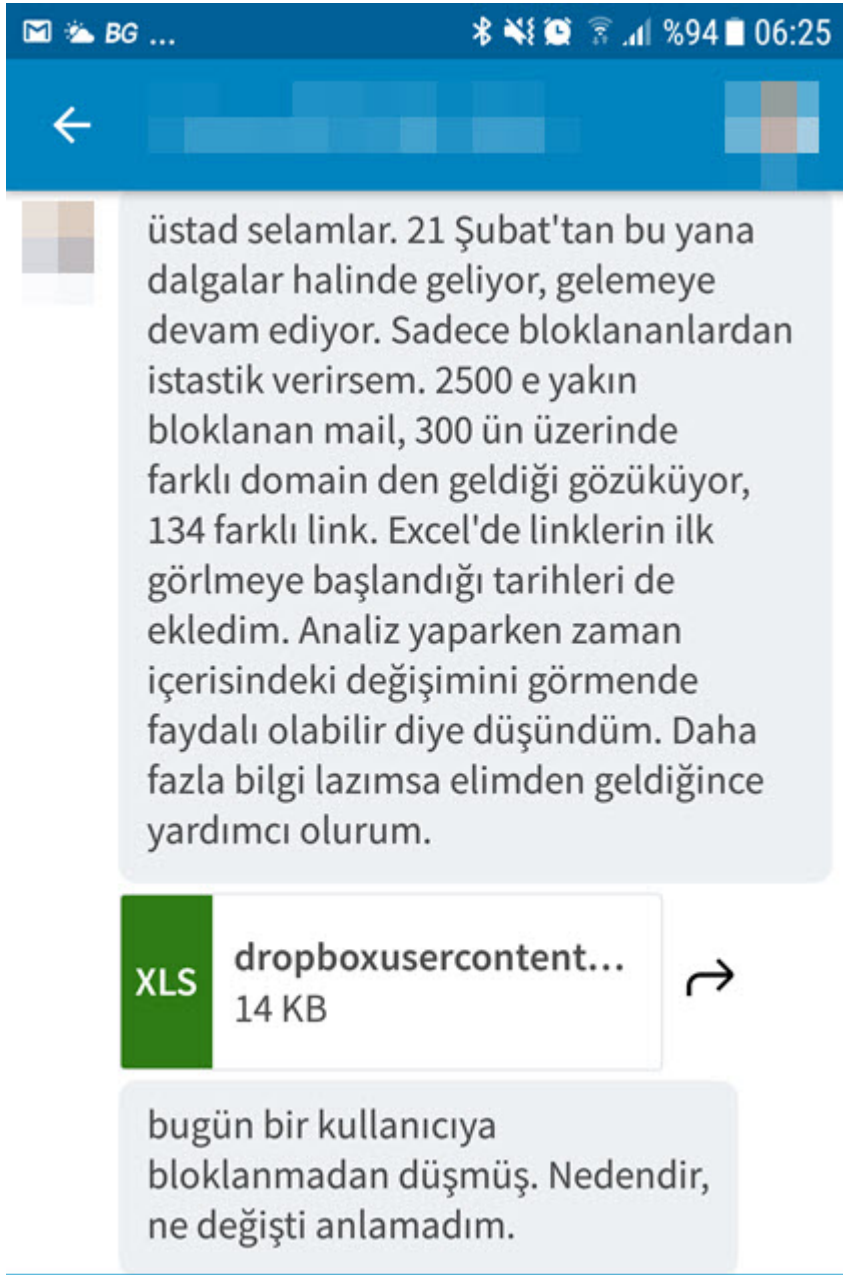
Dünya genelinde, son kullanıcı sistemlerindeki verileri zararlı yazılımlarla (cryptolocker vb.) şifreledikten sonra dosyaların şifresiz halini silip ardından da şifre çözme anahtarını kullanıcılara satmaya çalışarak bundan kazanç sağlama furyası hız kesmeden devam ediyor. Zaman zaman güvenlik araştırmacıları tarafından şifreleme algoritmalarının hatalı kullanımına bağlı olarak zararlı yazılımlar tarafından şifrelenen veri çözülebilse de, çoğu vakada kullanıcılar çoğunlukla art niyetli kişilere talep edilen yüksek miktartlı çözme bedelini ödemek zorunda kalıyorlar. Her vaka sonrasında veri yedeklemenin kıymeti daha net anlaşılrsa da, 1000 nasihat yerine 1 musibet ile hareket eden kullanıcılar olduğu sürece art niyeli kişilerin bu kazanç kapısından yakın gelecekte kolay kolay vazgeçmeyecekler gibi görünüyor.

Siber saldırıların hızla arttığı günümüzde, tehditleri tespit edip en kısa sürede müdahale edebilme kurumlar için büyük önem kazanmaya başladı. Öyle ki vizyoner kurumlar artık mevcut güvenlik teknolojilerini atlatan tehditleri kurum ağ ve sistemlerinde arayabilme adına siber tehdit avcılığına başladılar. Tehdit avcılığına imkan tanıyan teknolojilere baktığınızda, çoğunun kendi imzanızı yazmaya imkan tanıyan Yara aracını ve imzalarını desteklediğini görebiliyorsunuz.

Sevgili Halil ÖZTÜRKÇİ'nin 2014 yılında Yara ile ilgili olarak yayınlamış olduğu blog yazısına baktığınızda, Yara'nın o yıllarda yoğunluklu olarak adli bilişim analizinde ve bellek analizinde kullanılan Volatility aracı özelinde kullanıldığını görebiliyorsunuz. Bugün ise Yara'nın tehdit avcılığından zararlı yazılım analizine, FireEye NX gibi ticari ürünlerden, x64dbg gibi açık kaynak kodlu ve ücretsiz araçlara, paket kaydı yapan (full packet capture) teknolojilere kadar geniş bir alanda kullanılabildiğini görebiliyorsunuz. Bu da güvenlik uzmanlarına, güvenlik üreticilerinden bağımsız olarak kurum içinde kullanılan ve Yara desteği olan güvenlik sistemlerine, cihazlara tespit ve müdahaleye imkan tanıyan kendi yazdıkları imzalarını tanımlama imkanı tanıyor. İmza yazma, geçmiş yıllarda farklı güvenlik teknolojileri ile zor tecrübeler yaşamış olan güvenlik uzmanlarının kulağına nahoş gelse de, mevzu bahis Yara olduğunda işin rengi değişiyor çünkü Yara ile kural yazmanın oldukça basit, katma değerinin ise oldukça

yüksek olduğunu tecrübe ile sabit olarak söyleyebilirim.

Cryptolocker salgınının tekrar zirve yaptığı geçtiğimiz aylarda, sosyal ağlarda ve NetSec e-posta listesinde, kimi güvenlik sistemlerinin, teknolojilerinin bu salgınları tespit etmede ve engellemede yetersiz olduğunu gördüm. Ben de böyle bir durum ile karşı karşıya kalındığında, özellikle defansif güvenlik uzmanlarının Yara ile yazabilecekleri basit bir imza ile içeriği değişen benzer tehditleri nasıl tespit edebileceklerine dikkat çekmek istedim.



Cryptolocker salgınına baktığımızda, 24 saatte çok sayıda farklı e-posta

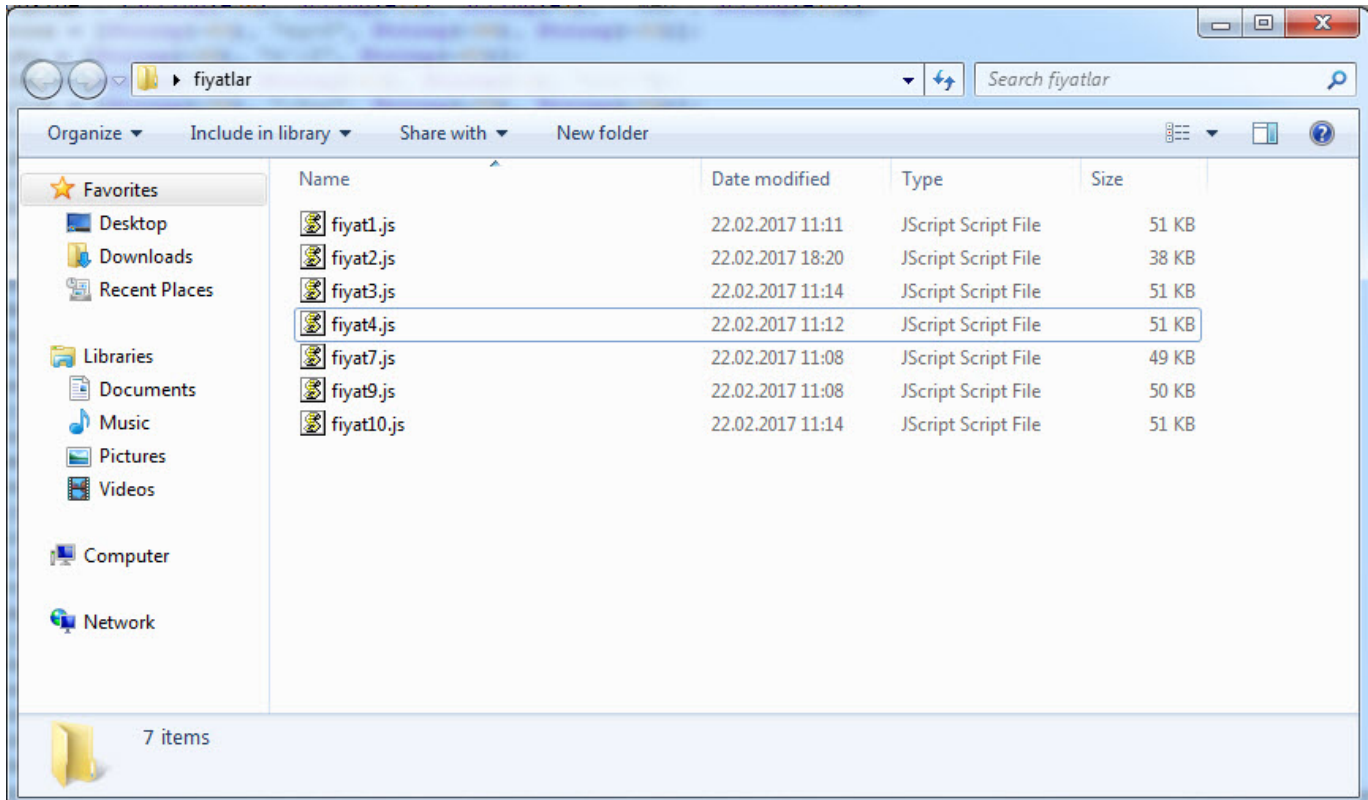
adresinden fiyatX.zip adı altında Cryptolocker varyantı gönderiliyordu. Her bir zip dosyası içinde karmaşılaştırılmış (obfuscated) JavaScript koduna sahip bir indirici (downloader) bulunuyor ve çalıştırıldıktan hemen sonra şifreleme zararlı yazılımını indirip sistemde çalıştırıyordu.

Doruk Tekin rammer@tele2.at

Ekte gönderilen mallar için birim fiyat ve teslim süresi rica ederim.

<https://dl.dropboxusercontent.com/s/be9kvoy3m3hwjk69/fiyat3.zip>

İyi günler.



İş, boyutları ve içeriği birbirinden tamamen farklı olan varyantları tespit etmeye geldiğinde Yara ile bunu oldukça kolay bir şekilde yapabilirsiniz. İlk olarak boyutları listelediğimizde bir varyant hariç tamamının 55 KB'den ufak

olduğunu görüyoruz. Dosyaların içeriğine baktığımızda ise her ne kadar içerik tamamen farklı olsa da String fonksiyonu kullanılarak karmaşık kod çözümlenerek alan adı ve indirilecek dosya ortaya çıktığı için String fonksiyonu üzerinden ilerleyebiliriz.

The screenshot displays a Windows desktop environment with several open applications:

- Notepad++ Windows:** Three windows are open, each containing JavaScript code. The code appears to be a series of variable assignments and function definitions, likely generated by a script. The code is written in a mix of English and Turkish, with many variables and functions named in Turkish.
- Command Prompt Window:** A window titled "C:\Windows\system32\cmd.exe" is open, showing the results of several `grep` commands. The commands are searching for the string "String" in files named `fiyat1.js` through `fiyat10.js`. The output shows the line number and the count of matches for each file.
- File Explorer Window:** A window titled "C:\Users\Mert\Desktop\fiyatlar\fiyat" is open, showing a list of files. The files are `fiyat1.js` through `fiyat10.js`, `grep.exe`, and `wc.exe`. The files are listed with their dates, times, sizes, and file types.

The Command Prompt window shows the following output:

```
C:\Users\Mert\Desktop\fiyatlar>grep "String" fiyat1.js ! wc -l
442

C:\Users\Mert\Desktop\fiyatlar>grep "String" fiyat2.js ! wc -l
199

C:\Users\Mert\Desktop\fiyatlar>grep "String" fiyat3.js ! wc -l
444

C:\Users\Mert\Desktop\fiyatlar>grep "String" fiyat4.js ! wc -l
443

C:\Users\Mert\Desktop\fiyatlar>grep "String" fiyat7.js ! wc -l
430

C:\Users\Mert\Desktop\fiyatlar>grep "String" fiyat9.js ! wc -l
436

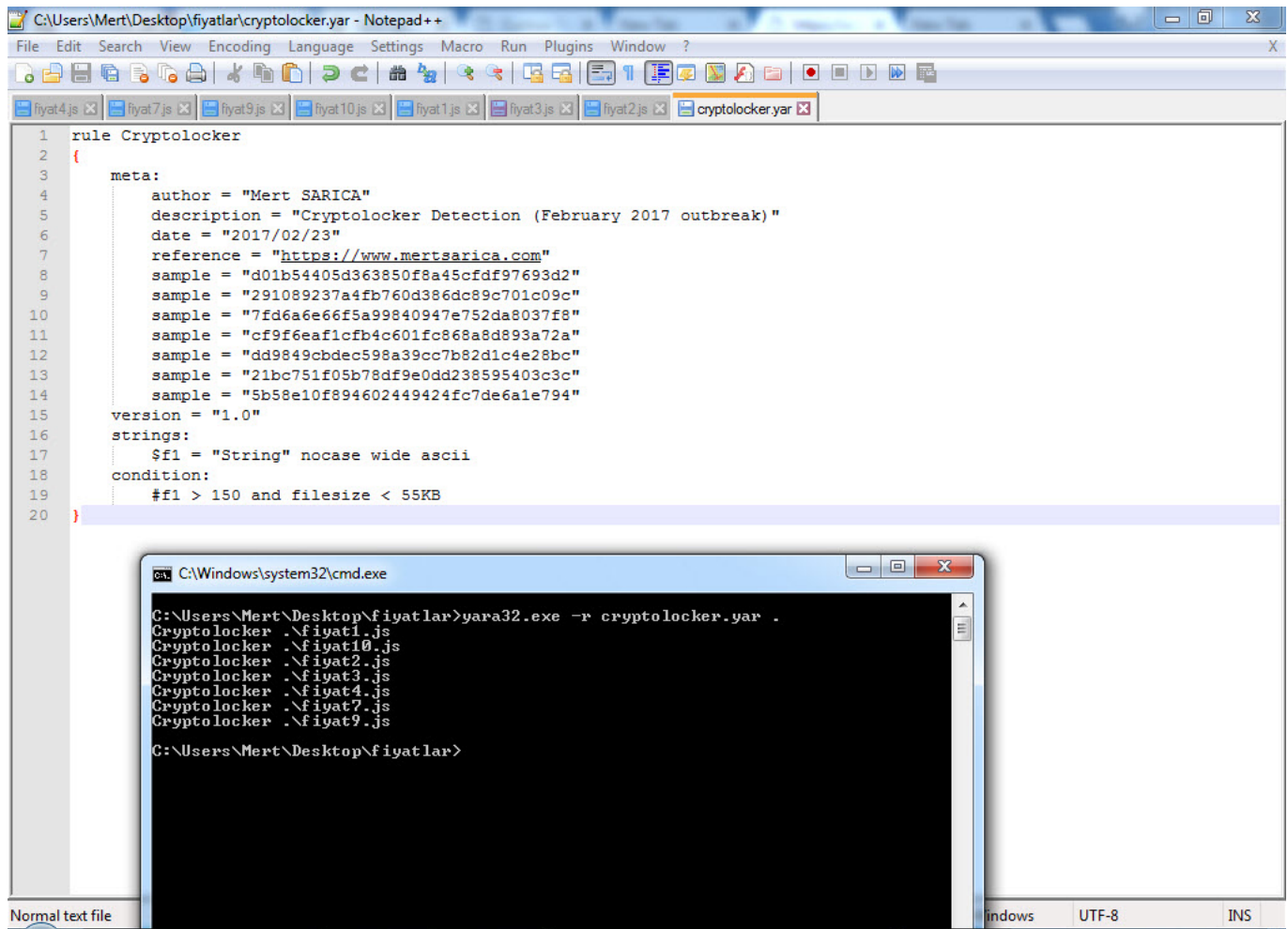
C:\Users\Mert\Desktop\fiyatlar>grep "String" fiyat10.js ! wc -l
448

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

The File Explorer window shows the following list of files:

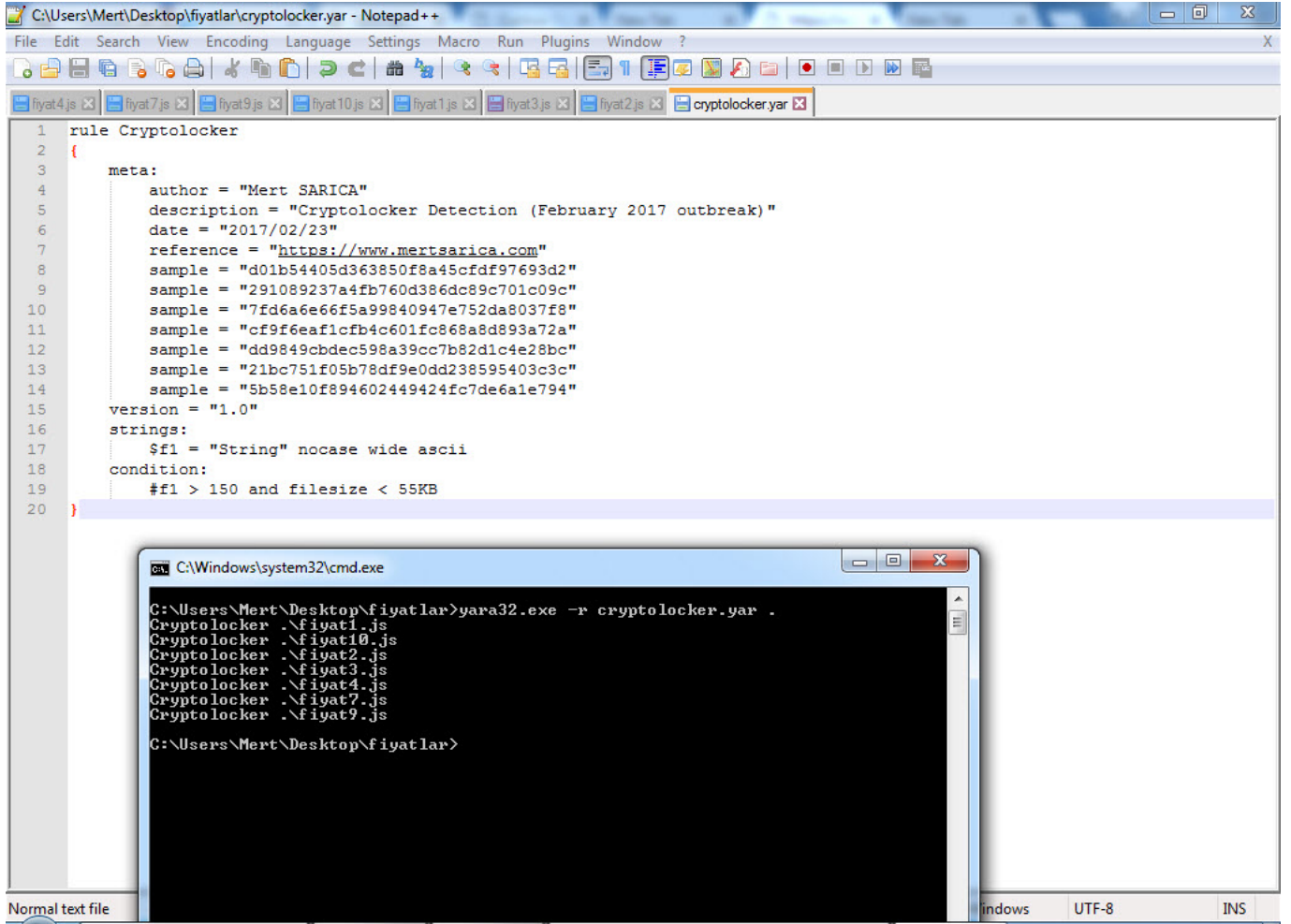
File Name	Date	Time	File Type	Size
fiyat1.js	22.02.2017	11:11	JScript Script File	51 KB
fiyat2.js	22.02.2017	18:20	JScript Script File	38 KB
fiyat3.js	22.02.2017	11:14	JScript Script File	51 KB
fiyat4.js	22.02.2017	11:12	JScript Script File	51 KB
fiyat7.js	22.02.2017	11:08	JScript Script File	49 KB
fiyat9.js	22.02.2017	11:08	JScript Script File	50 KB
fiyat10.js	22.02.2017	11:14	JScript Script File	51 KB
grep.exe	14.04.2003	00:00	Application	79 KB
wc.exe	10.11.1999	23:00	Application	29 KB

Normal şartlarda 55 KB'dan küçük olan bir dosyada kullanılan String fonksiyonunun sayısının, dosya şüpheli olmadığı sürece 150'den az olacağını varsayarak Yara anahtar kelimelerinden faydalananarak aşağıdaki gibi bir Yara imzası oluşturabiliriz. Yazmış olduğumuz cryptolocker.yar isimli imzanın doğru çalıştığını ve elimizdeki tüm varyantları tespit edebildiğini Yara aracı ile doğruladıktan sonra imzamızı Yara destekleyen tüm güvenlik sistemlerine, teknolojilerine yükleyerek yeni bir salgını, tehdidi tespit etmede önemli bir mesafe katetmiş oluyoruz.



```
1 rule Cryptolocker
2 {
3   meta:
4     author = "Mert SARICA"
5     description = "Cryptolocker Detection (February 2017 outbreak)"
6     date = "2017/02/23"
7     reference = "https://www.mertsarica.com"
8     sample = "d01b54405d363850f8a45cfd97693d2"
9     sample = "291089237a4fb760d386dc89c701c09c"
10    sample = "7fd6a6e66f5a99840947e752da8037f8"
11    sample = "cf9f6eaf1cfb4c601fc868a8d893a72a"
12    sample = "dd9849cbdec598a39cc7b82d1c4e28bc"
13    sample = "21bc751f05b78df9e0dd238595403c3c"
14    sample = "5b58e10f894602449424fc7de6a1e794"
15  version = "1.0"
16  strings:
17    $f1 = "String" nocase wide ascii
18  condition:
19    #f1 > 150 and filesize < 55KB
20 }
```

```
C:\Windows\system32\cmd.exe
C:\Users\Mert\Desktop\fiyatlar>yara32.exe -r cryptolocker.yar .
Cryptolocker .\fiyat1.js
Cryptolocker .\fiyat10.js
Cryptolocker .\fiyat2.js
Cryptolocker .\fiyat3.js
Cryptolocker .\fiyat4.js
Cryptolocker .\fiyat7.js
Cryptolocker .\fiyat9.js
C:\Users\Mert\Desktop\fiyatlar>
```



The image shows a Notepad++ window with a YARA rule named 'Cryptolocker'. The rule includes meta-information such as author, description, date, and several sample hashes. It also defines a condition to match files larger than 150 bytes and smaller than 55KB. Overlaid on the bottom of the Notepad++ window is a Windows command prompt window. The command prompt shows the execution of 'yara32.exe -r cryptolocker.yar .' in the directory 'C:\Users\Mert\Desktop\fiyatlar'. The output lists several JavaScript files that were scanned: 'fiyat1.js', 'fiyat10.js', 'fiyat2.js', 'fiyat3.js', 'fiyat4.js', 'fiyat7.js', and 'fiyat9.js'.

```
1 rule Cryptolocker
2 {
3   meta:
4     author = "Mert SARICA"
5     description = "Cryptolocker Detection (February 2017 outbreak)"
6     date = "2017/02/23"
7     reference = "https://www.mertsarica.com"
8     sample = "d01b54405d363850f8a45cfd97693d2"
9     sample = "291089237a4fb760d386dc89c701c09c"
10    sample = "7fd6a6e66f5a99840947e752da8037f8"
11    sample = "cf9f6eaf1cfb4c601fc868a8d893a72a"
12    sample = "dd9849cbdec598a39cc7b82d1c4e28bc"
13    sample = "21bc751f05b78df9e0dd238595403c3c"
14    sample = "5b58e10f894602449424fc7de6a1e794"
15  version = "1.0"
16  strings:
17    $f1 = "String" nocase wide ascii
18  condition:
19    #f1 > 150 and filesize < 55KB
20 }
```

```
C:\Windows\system32\cmd.exe
C:\Users\Mert\Desktop\fiyatlar>yara32.exe -r cryptolocker.yar .
Cryptolocker .\fiyat1.js
Cryptolocker .\fiyat10.js
Cryptolocker .\fiyat2.js
Cryptolocker .\fiyat3.js
Cryptolocker .\fiyat4.js
Cryptolocker .\fiyat7.js
Cryptolocker .\fiyat9.js
C:\Users\Mert\Desktop\fiyatlar>
```

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.