

DoS ile Mücadele

written by Mert SARICA | 1 March 2019

2009 yılından bu yana “Bilgi güçtür ve paylaşıldıkça artar!” sloganıyla siber güvenlik alanına meraklılar için özveriyle yaşatmaya çalıştığım blogumun bundan yıllar yıllar önce bir DDoS saldırısı ile karşı karşıya kaldığını ve daha sonrasında blogumu Cloudflare arkasına taşıdığımı anımsıyorum. Son birkaç yıldır ise DDoS ve DoS benzeri bir saldırı ile karşılaşmamış olmanın verdiği rehabet ile ne işletim sistemi üzerinde OSSEC gibi bir HIDS’i, ne Cloudflare, ne WordPress ne de Nginx üzerinde bu tür saldırılara karşı (Wordfence rate limiting, modsecurity vs.) bir koruma özelliğini devreye almamıştım.

15 Temmuz 2018 tarihinde blogumu ziyaret etmeye çalıştığımda web sunucusunun geç yanıt verdiğini farkettilim. "Acaba bu defa neyi bozdum ?" diye kendi kendime homurdandıktan hemen sonra ssh ile sunucuma bağlandığımda herhangi bir yavaşlık farketmedim. Web sunucusundaki yavaşlığın uygulama katmanı ve/veya veritabanı ile ilgili olduğunu düşünerek top komutunu bile çalıştırmadan direk web sunucusunun erişim kayıtlarına (access.log) göz attığımda, 14 Temmuz saat 14:03 itibariyle bloguma rastgele parametreler içeren çok sayıda istekte bulunduğunu farkettilim. (HTTP DoS)

[illegible]

HTTP DoS saldırısını gerçekleştiren kaynak ip adreslerine bakıp,

Cloudflare'in ip adreslerini gördüğümde, rehabetin bedelinden çıkardığım ilk ders (lessons learned) Cloudflare'e yapılan istekleri web sunucuma göndermek (CF-Connecting-IP başlığı) oldu. :) Bir yandan saldırının tam olarak hangi ülkeden kaynaklandığını öğrenmek için Cloudflare'in Analytics sayfasındaki tehdit haritasına baktığımda DoS saldırısının Rusya'dan yapıldığını öğrendim. (Bu cümleyi yazdıktan sonra nedense aklıma Red Alert 2'deki Tanya geldi. :))

```
GNU nano 2.5.3 File: nginx.conf
user www-data;
worker_processes auto;
pid /run/nginx.pid;

events {
    worker_connections 768;
    # multi_accept on;
}

http {
    include /etc/nginx/cloudflare;

    ##
    # Basic Settings
    ##
    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 2048;
    server_tokens off;
    client_max_body_size 40M;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    ##
    # SSL Settings
    ##
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2; # Dropping SSLv3, ref: POODLE
    ssl_prefer_server_ciphers on;

    ##
    # Logging Settings
    ##
    access_log /var/log/nginx/access.log;
    error_log /var/log/nginx/error.log;

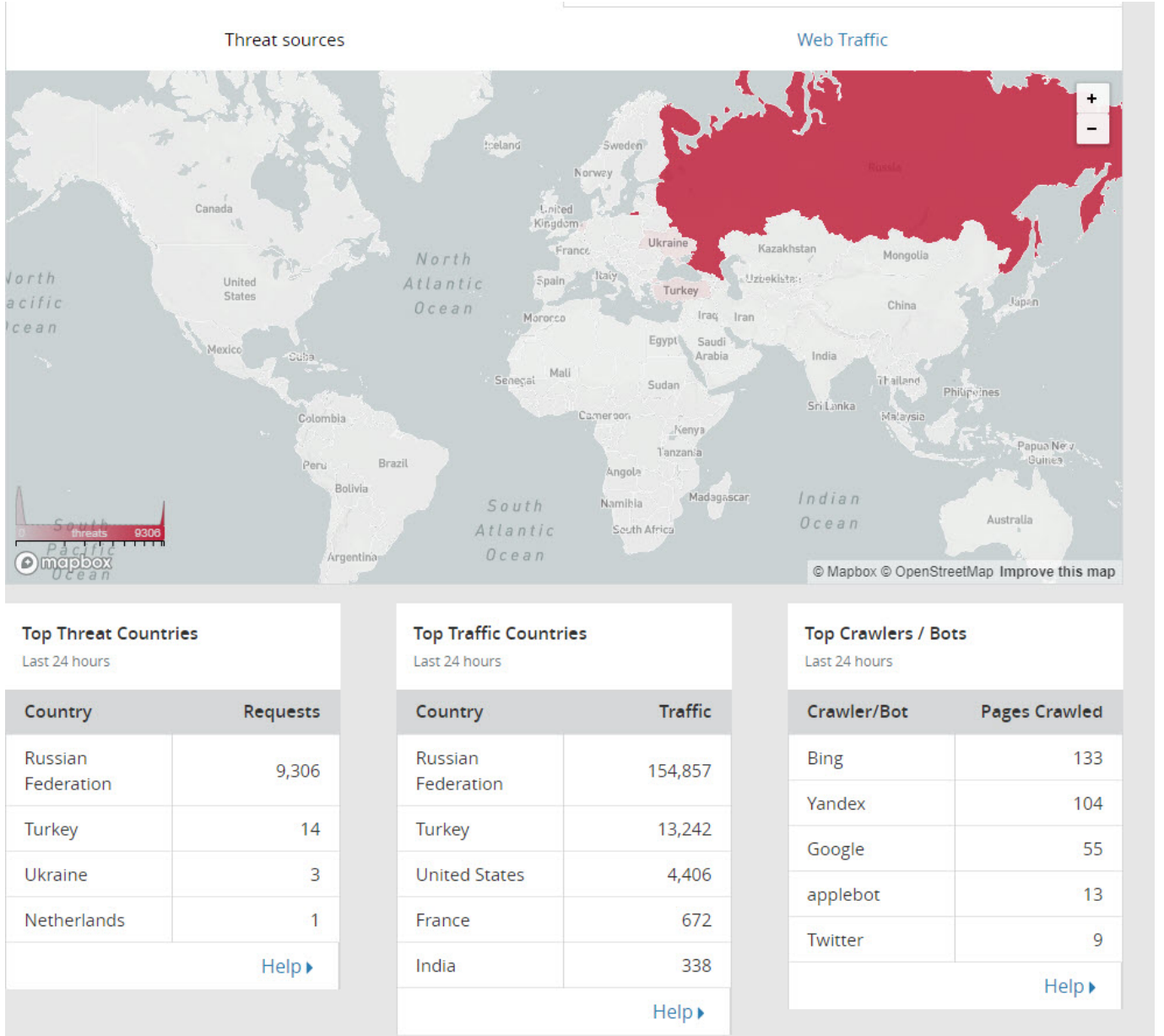
    ##
    # Gzip Settings
    ##
    gzip on;
    gzip_disable "msie6";

    # gzip_vary on;
    # gzip_proxied any;
    # gzip_comp_level 6;
    # gzip_buffers 16 8k;
    # gzip_http_version 1.1;
    # gzip_types text/plain text/css application/json application/javascript text/xml application/xml application/xml+rss text/javascript;

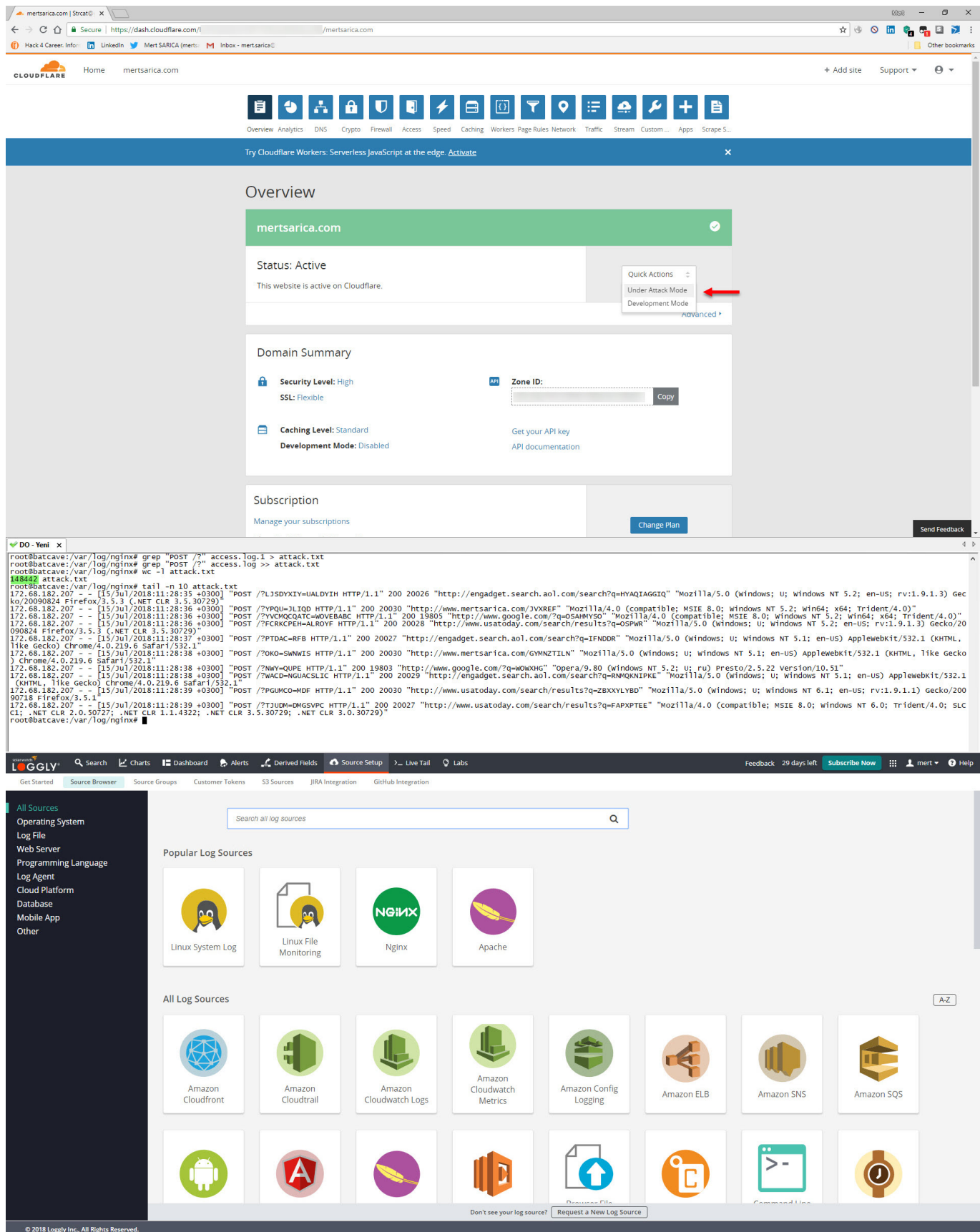
    ##
    # Virtual Host Configs
    ##
    include /etc/nginx/conf.d/*.conf;
    include /etc/nginx/sites-enabled/*;
}

set_real_ip_from 103.21.244.0/22;
set_real_ip_from 103.22.200.0/22;
set_real_ip_from 103.31.4.0/22;
set_real_ip_from 104.16.0.0/12;
set_real_ip_from 108.162.192.0/18;
set_real_ip_from 131.0.72.0/22;
set_real_ip_from 141.101.64.0/18;
set_real_ip_from 162.158.0.0/15;
set_real_ip_from 172.64.0.0/13;
set_real_ip_from 173.245.48.0/20;
set_real_ip_from 188.114.96.0/20;
set_real_ip_from 190.93.240.0/20;
set_real_ip_from 197.234.240.0/22;
set_real_ip_from 198.41.128.0/17;
set_real_ip_from 2400:cb00::/32;
set_real_ip_from 2606:4700::/32;
set_real_ip_from 2803:f800::/32;
set_real_ip_from 2405:b500::/32;
set_real_ip_from 2405:8100::/32;
set_real_ip_from 2c0f:f248::/32;
set_real_ip_from 2a06:98c0::/29;

# use any of the following two
real_ip_header CF-Connecting-IP;
#real_ip_header X-Forwarded-For;
```

Ben bu araştırmaları yaparken bir yandan DoS saldırısı devam ettiği için saldırıları durdurmak için Cloudflare'in panelinden Under Attack özelliğini aktif hale getirdim ve saldırı anında etkisiz hale geldi. Akabinde saldırı ile ilgili istatistiki bilgiler toplamak için öncelikle komut satırından ardından da SolarWinds'in Loggly isimli web uygulamasından faydalandım. Loggly dışında çevrimdışı analiz için de bir yandan Apache Logs Viewer aracının çıktılarını inceledim. Analiz sonucuna göre ~24 saatte bloguma yaklaşık ~150.000 istek yapılmış ve bunlardan ~60.000 tanesi (502 HTTP Status) blogumu erişilemez hale getirmişti.



File upload

1 Step 1: Upload Your File

You can send up to a max of 100MB per file and 1MB per event. The file should be line-separated text data, and the events can be up to 7 days old. Compressed files are not supported. You can upload a log file using common file types: .txt, .log, .csv and .json and quickly see what Loggly can do for you.

Remove any text in the filename after the file type extension. For example, a filename such as nginx-access.log is acceptable. But a filename such as nginx-access.log-20171109 will be rejected, so please rename such a file and then upload it.

The following file types are not supported:

The following file types are not supported:
 .bin, .exe, .lib, .dat, .mp3, .gif, .jpg, .jpeg, .bmp, .zip, .gz, .tar, .dll, .png, .img, .db, .lib, .pdf, .mp4, .mpeg, .mov, .jar, .pem, .crt, .doc, .xls, .xlsx, .docx, .ppt, .pptx, .dmg, .rar, .apk, and .7z

2 Step 2: Verify

✔ Congratulations! We received your logs and they have been automatically parsed.

3 Step 3: Next Steps

[View in Search](#)
[File Monitoring Setup](#)

Advanced Options

- Windows File Monitoring
- Linux File Monitoring
- Timestamps

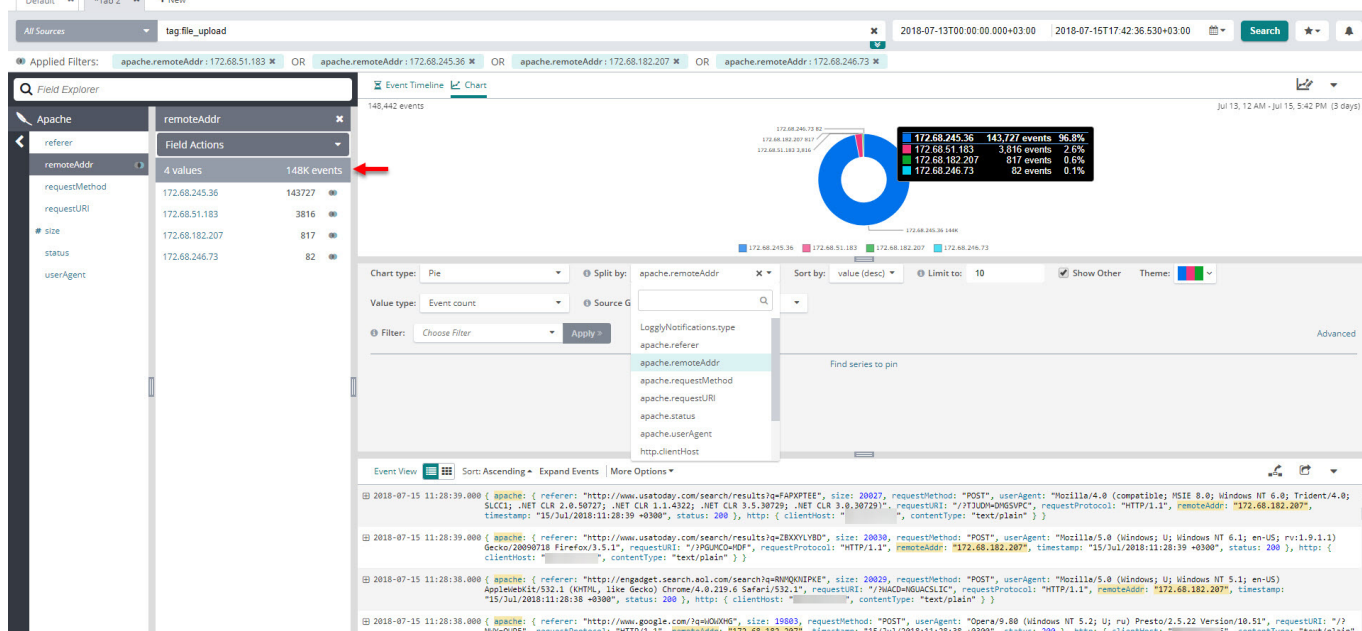
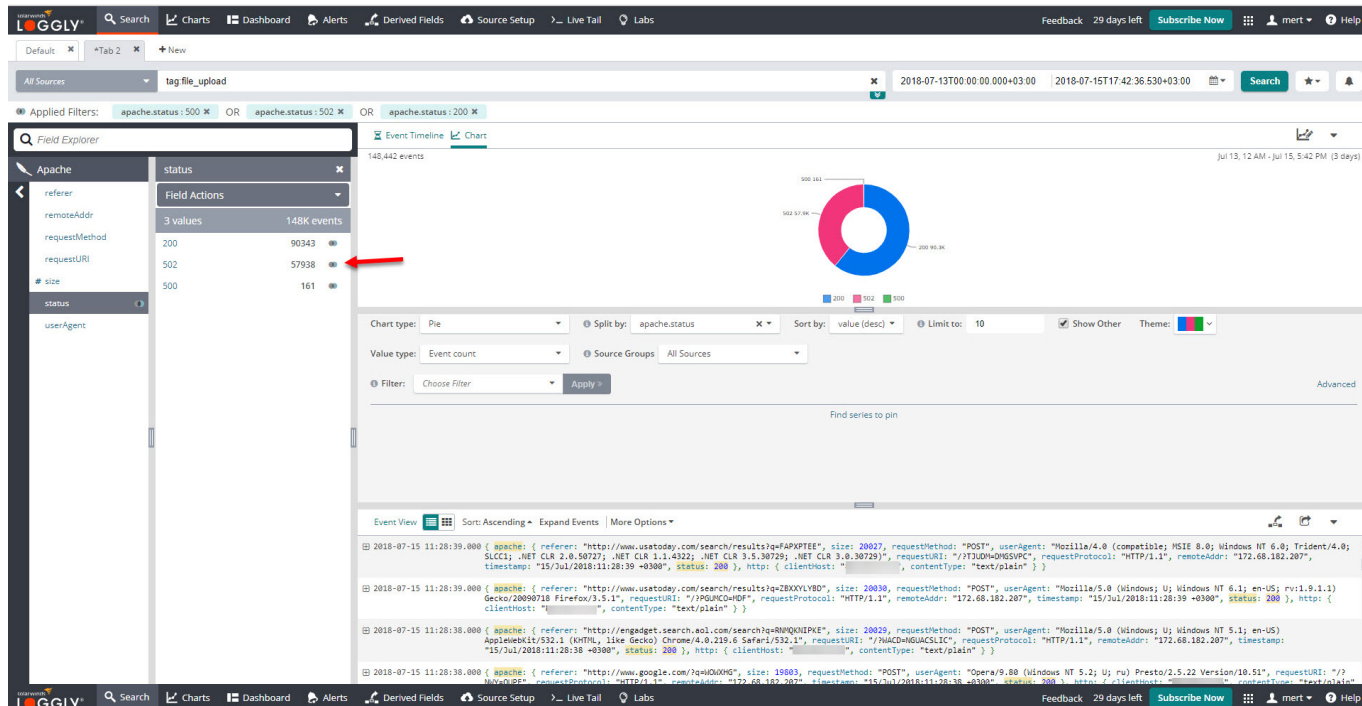
Troubleshooting

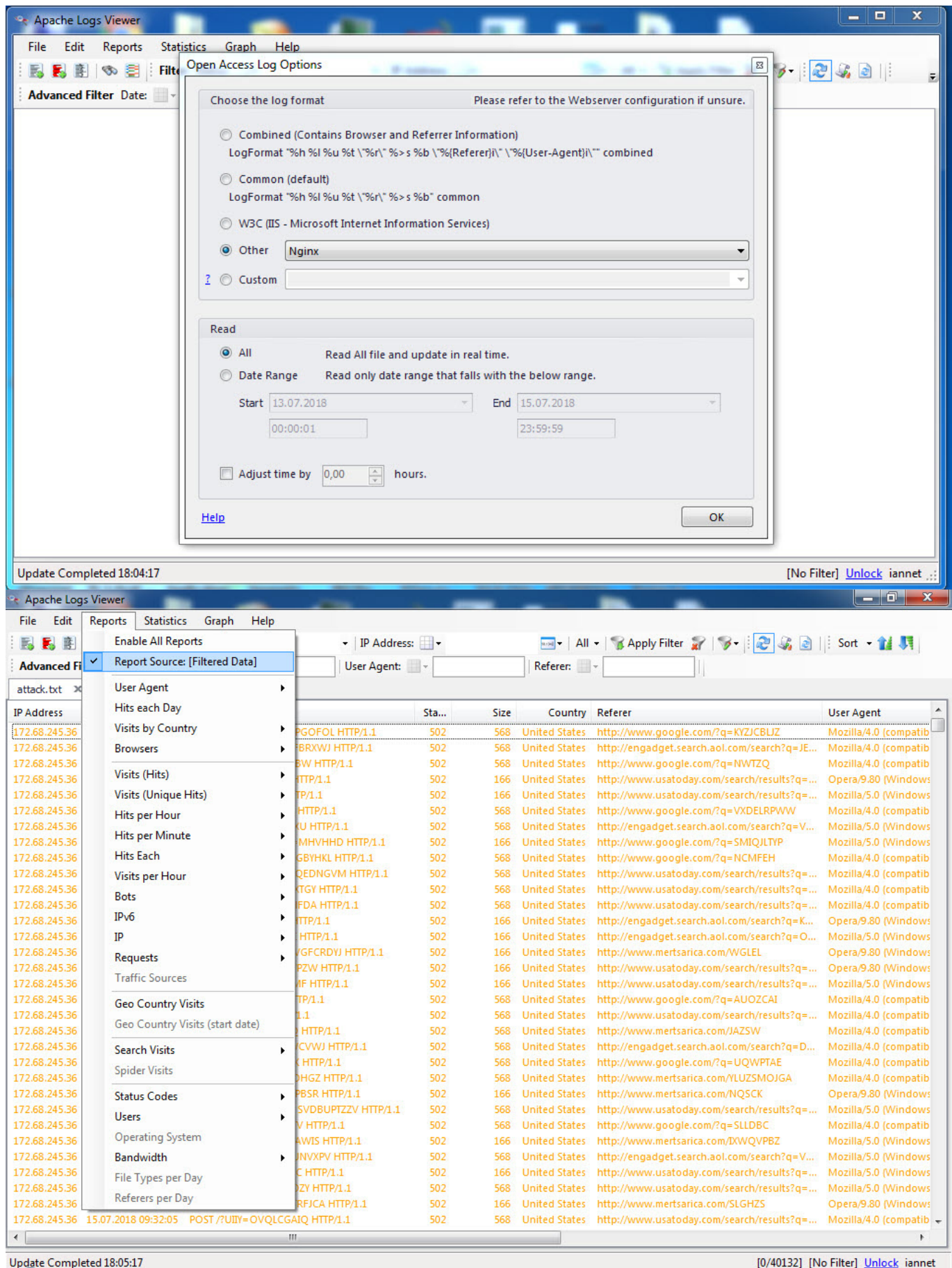
[Troubleshoot File Upload](#)
[Community Forum](#)

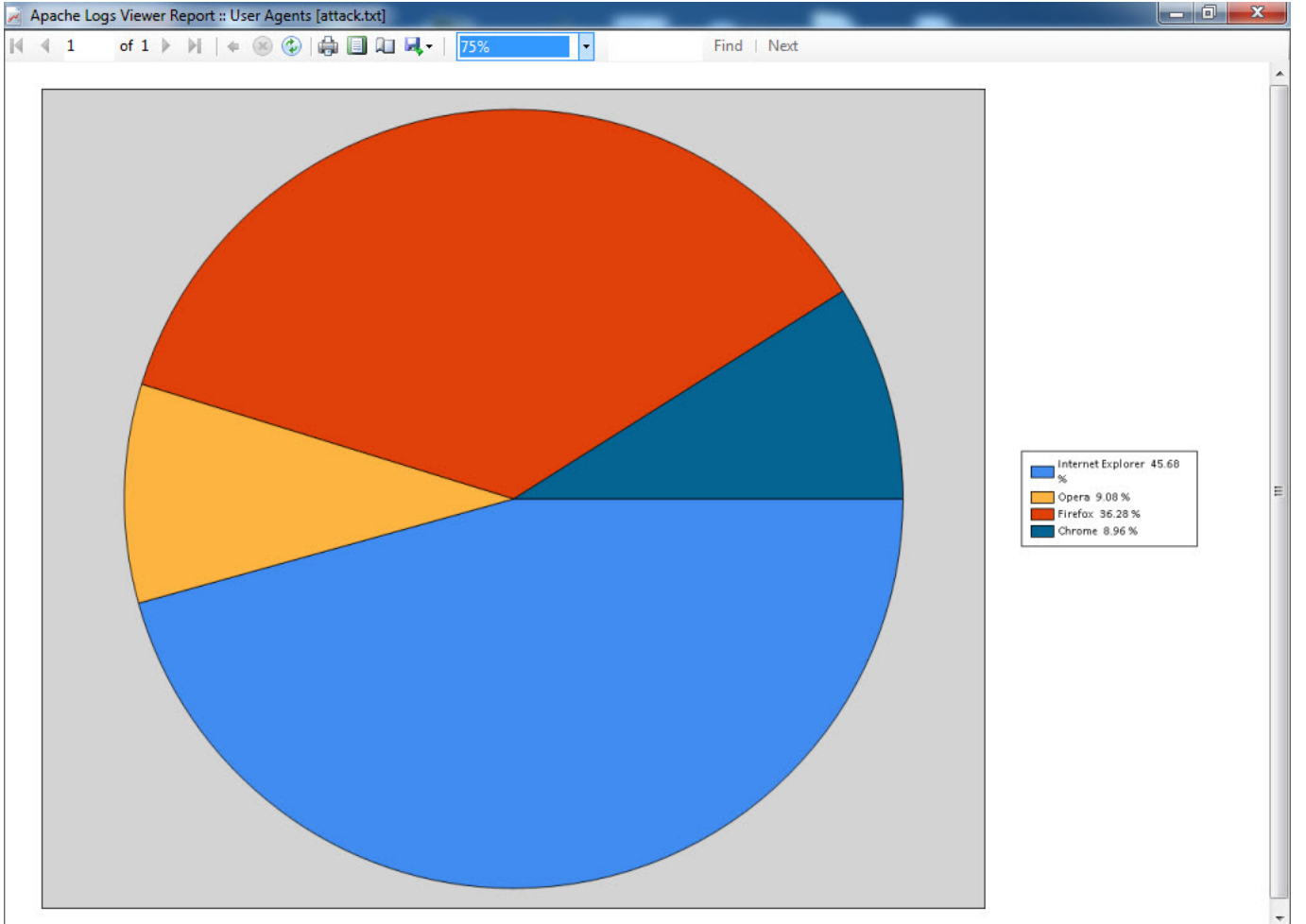
The screenshot displays the Kibana interface for analyzing Apache logs. On the left, the **Field Explorer** shows the **requestURI** field with 148K events. The **Field Actions** dropdown is open, showing options like **Field Actions**, **300 values**, and **148K events**. The **requestURI** field is selected, and the **Field Actions** dropdown is open, showing options like **Field Actions**, **300 values**, and **148K events**.

On the right, the **Event Timeline** view shows a line chart of event counts over time. The chart is filtered by **requestURI** and **source**. The chart shows a peak in event counts around July 13, 2018, at 12 AM. The chart is titled "2,916 events" and "Jul 13, 12 AM - Jul 15, 5:42 PM (3 days)".

The chart is titled "2,916 events" and "Jul 13, 12 AM - Jul 15, 5:42 PM (3 days)". The chart shows a peak in event counts around July 13, 2018, at 12 AM. The chart is titled "2,916 events" and "Jul 13, 12 AM - Jul 15, 5:42 PM (3 days)".







DoS saldırısı ile ilgili olarak, erişim kayıtları (access.log) üzerinden elde ettiğim bilgilere istinaden yapmış olduğum araştırmalar sonucunda birbirine benzeyen 3 farklı Python aracı (hulk.py , doser.py , attack.py) dikkatimi çekti. Bu araçların kaynak kodları ile bloguma yapılan DoS saldırısını karşılaştırdığımda, gelen isteklerin POST isteği olması sebebiyle doser.py aracı ile bu saldırının gerçekleştirildiği fikri ağır basmış oldu.

```
ddos/attack.py at master · firebug/ddos · GitHub
Secure | https://github.com/firebug/ddos/blob/master/attack.py
Hack 4 Career: Info LinkedIn Mert SARICA (mert) Inbox - mertsarica@

25 def useragent_list():
26     global headers_useragents
27     headers_useragents = []
28     headers_useragents.append('Mozilla/5.0 (X11; U; Linux x86_64; en-US; rv:1.9.1.3) Gecko/20090913 Firefox/3.5.3')
29     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 6.1; en; rv:1.9.1.3) Gecko/20090824 Firefox/3.5.3 (.NET CLR 3.5.30721)')
30     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 5.2; en-US; rv:1.9.1.3) Gecko/20090824 Firefox/3.5.3 (.NET CLR 3.5.30721)')
31     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1')
32     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; AppleWebKit/532.1 (KHTML, like Gecko) Chrome/4.0.219.6 Safari/532.1)')
33     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; SLCC2; .NET CLR 2.0.50727; InfoPath.2)')
34     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Win64; x64; Trident/4.0)')
35     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; SV1; .NET CLR 2.0.50727; InfoPath.2)')
36     headers_useragents.append('Mozilla/5.0 (Windows; U; MSIE 7.0; Windows NT 6.0; en-US)')
37     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 6.1; Windows XP)')
38     headers_useragents.append('Opera/9.80 (Windows NT 5.2; U; ru) Presto/2.5.22 Version/10.51')
39     return(headers_useragents)
40
41 # generates a referer array
42 def referer_list():
43     global headers_referers
44     headers_referers = []
45     headers_referers.append('http://www.usatoday.com/search/results?q=')
46     headers_referers.append('http://engadget.search.aol.com/search?q=')
47     headers_referers.append('http:// ' + host + '/')
48     return(headers_referers)
49
50 def handler(signum, _):
51     if signum == signal.SIGALRM:
52         print "Time is up !"
53         print "Attack finished !"
54         sys.exit()
55
56 #builds random ascii string
57 def buildblock(size):
58     out_str = ''
59     for i in range(0, size):
60         a = random.randint(65, 90)
61         out_str += chr(a)
62     return(out_str)
63
64 def send_packet(host, param_joiner):
65     request = urllib2.Request(url = param_joiner + buildblock(random.randint(3,10)) + ".html" + buildblock(random.randint(3,10)))
66     request.add_header('User-Agent', random.choice(headers_useragents))
67     request.add_header('Cache-Control', 'no-cache')
68     request.add_header('Accept-Charset', 'ISO-8859-1,utf-8;q=0.7;q=0.7')
69     request.add_header('Referer', random.choice(headers_referers) + buildblock(random.randint(5,10)))
70     request.add_header('Keep-Alive', random.randint(10,120))
71     request.add_header('Connection', 'keep-alive')
72     request.add_header('Host', host)
```

```
doserpy/doser.py at master · Quitten/doserpy · GitHub
Secure | https://github.com/Quitten/doserpy/blob/master/doser.py
Hack 4 Career: Info LinkedIn Mert SARICA (mert) Inbox - mertsarica@

98 pass
99
100 class SendPOSTThread(threading.Thread):
101     def run(self):
102         try:
103             while True:
104                 global url, payload
105                 sendPOST(url, payload)
106         except:
107             pass
108
109 # TODO:
110 # check if the site stop responding and alert
111
112 def main(argv):
113     parser = argparse.ArgumentParser(description='Sending unlimited amount of requests in order to perform DoS attacks. Written by Baran')
114     parser.add_argument('-g', help='Specify GET request. Usage: -g <url>')
115     parser.add_argument('-p', help='Specify POST request. Usage: -p <url>')
116     parser.add_argument('-d', help='Specify data payload for POST request', default=None)
117     parser.add_argument('-ah', help='Specify additional header/s. Usage: -ah <Content-type: application/json> <User-Agent: Doser>')
118     parser.add_argument('-t', help='Specify number of threads to be used', default=500, type=int)
119     args = parser.parse_args()
120
121     global url, payload, additionalHeaders
122     additionalHeaders = args.ah
123     payload = args.d
124
125     if args.g:
126         url = args.g
127         for i in range(args.t):
128             t = SendGETThread()
129             t.start()
130
131     if args.p:
132         url = args.p
133         for i in range(args.t):
134             t = SendPOSTThread()
135             t.start()
136
137     if len(sys.argv)==1:
138         parser.print_help()
139         exit()
140
141 if __name__ == "__main__":
142     main(sys.argv[1:])
143
```

```
hulk/hulk.py at master · grafov/hulk/blob/master/hulk.py
Secure | https://github.com/grafov/hulk/blob/master/hulk.py
Hack 4 Career: Info LinkedIn Mert SARICA (mert) Inbox - mertsarica@

33 def set_safe():
34     global safe
35     safe=1
36
37 # generates a user agent array
38 def useragent_list():
39     global headers_useragents
40     headers_useragents.append('Mozilla/5.0 (X11; U; Linux x86_64; en-US; rv:1.9.1.3) Gecko/20090913 Firefox/3.5.3')
41     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 6.1; en; rv:1.9.1.3) Gecko/20090824 Firefox/3.5.3 (.NET CLR 3.5.3072)')
42     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 5.2; en-US; rv:1.9.1.3) Gecko/20090824 Firefox/3.5.3 (.NET CLR 3.5.3072)')
43     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1')
44     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; AppleWebKit/532.1 (KHTML, like Gecko) Chrome/4.0.219.6 S')
45     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; SLCC2; .NET CLR 2.0.50727; InfoPa')
46     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.0; Trident/4.0; SLCC1; .NET CLR 2.0.50727; .NET CLR 1.1.4')
47     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Win64; x64; Trident/4.0)')
48     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; SV1; .NET CLR 2.0.50727; InfoPath.2)')
49     headers_useragents.append('Mozilla/5.0 (Windows; U; Windows 7.0; Windows NT 6.0; en-US)')
50     headers_useragents.append('Mozilla/4.0 (compatible; MSIE 6.1; Windows XP)')
51     headers_useragents.append('Opera/9.80 (Windows NT 5.2; U; ru) Presto/2.5.22 Version/10.51')
52     return(headers_useragents)
53
54 # generates a referer array
55 def referer_list():
56     global headers_referers
57     headers_referers.append('http://www.google.com/?q=')
58     headers_referers.append('http://www.usatoday.com/search/results?q=')
59     headers_referers.append('http://engadget.search.aol.com/search?q=')
60     headers_referers.append('http://' + host + '/')
61     return(headers_referers)
62
63 #builds random ascii string
64 def buildblock(size):
65     out_str = ''
66     for i in range(0, size):
67         a = random.randint(65, 90)
68         out_str += chr(a)
69     return(out_str)
70
71 def usage():
72     print '.....'
73     print 'USAGE: python hulk.py <url>'
74     print 'you can add "safe" after url, to autoshut after dos'
75     print '.....'
76
77
78 #http request
79 def httpcall(url):
80     .....
81     .....
82     .....
83     .....
84     .....
85     .....
86     .....
87     .....
88     .....
89     .....
90     .....
91     .....
92     .....
93     .....
94     .....
95     .....
96     .....
97     .....
98     .....
99     .....
100    .....
```

Blogumun WordPress yönetici paneline de göz gezdirdiğimde, Aranılan Terimler kısmındaki i_love_brother_This_is_not_an_attack_This_is_a_test arama terimi dikkatimi çekti. Bu terim ile ilgili göndermiş olduğum bir tweete yapılan yorumda da olduğu gibi, blogumun DDoS testlerinde akla ilk gelen blog olabileceğini de dikkate alarak Cloudflare'in yönetim paneli üzerinden Rusya ip adreslerini engelledim.



Mert SARICA @MertSARICA · 17 Tem
Sevginin de böylesi işte :)

Dimensi ▾

eşit ▾

Uygula

18-06-18 – 17-07-18 ▾

Aranılan Terimler

➞ Hacker avı

➞ x

➞ i_love_brother_This_is_not_an_attack_This_is_a_test

➞ linux

➞ hack

1 13



musa şana
@musa_sana

Takip et

@MertSARICA adlı kullanıcıya yanıt olarak

ddos testlerinde akla ilk gelen blog :)

22:08 - 17 Tem 2018

3 Beğeni



17 Temmuz tarihinde yapılan kısa süreli benzer bir saldırıda bu defa kaynak IP adreslerini görebildiğim için DoS saldırısının Fransa'da bulunan bir ip adresinden (51.254.122.14) gerçekleştirildiğini gördüm ve bu sefer de Fransa ip adreslerini Cloudflare'in yönetim paneli üzerinden engelledim.

```
root@batcave:/var/log/nginx# grep "POST /?" access.log.1 | cut -d " " -f 1 | sort | uniq -i
```

```
51.254.122.14
```

```
root@batcave:/var/log/nginx#
```

Whois domain sorgula: X

www.whois.com.tr/domain/51.254.122.14

Hack 4 Career: Info LinkedIn Mert SARICA (mert...) Inbox - mertsarica

DomainAI Seoport SiteAnaliz Sosyal Video Indir

Whois'de reklam ver ÜYELİK

IP Whois Kaydı (51.254.122.14)

Lookup results for 51.254.122.14 from whois.ripe.net server:

```
inetnum: 51.254.0.0 - 51.255.255.255
netname: FR-OVH-20150522
descr: OVH SAS
country: FR
admin-c: OTIC2-RIPE
tech-c: OTIC2-RIPE
status: LEGACY
mnt-by: OVH-NIT
created: 2015-05-20T06:55:56Z
last-modified: 2015-05-27T15:52:47Z
source: RIPE
org: ORG-OS3-RIPE
organisation: ORG-OS3-RIPE
org-name: OVH SAS
org-type: LIR
address: 2 rue Kellermann
address: 59100
address: Roubaix
address: FRANCE
phone: +33022010007
abuse-c: AR15333-RIPE
admin-c: OTIC2-RIPE
admin-c: OK217-RIPE
admin-c: G984-RIPE
mnt-ref: OVH-NIT
mnt-ref: RIPE-NCC-HI-NIT
mnt-by: RIPE-NCC-HI-NIT
mnt-by: OVH-NIT
created: 2004-04-17T11:23:17Z
last-modified: 2007-10-30T14:40:06Z
source: RIPE # Filtered
role: OVH Technical Contact
address: OVH SAS
address: 2 rue Kellermann
address: 59100 Roubaix
address: France
admin-c: OK217-RIPE
tech-c: G984-RIPE
tech-c: SL18162-RIPE
nic-hdl: OTIC2-RIPE
abuse-mailbox: abuse@ovh.net
mnt-by: OVH-NIT
created: 2004-01-20T17:42:20Z
last-modified: 2014-09-05T10:47:15Z
source: RIPE # Filtered
route: 51.254.0.0/15
descr: OVH
origin: AS16276
mnt-by: OVH-NIT
created: 2015-05-20T17:50:05Z
last-modified: 2015-05-20T17:50:05Z
source: RIPE
```

Share

Hakkımızda Alan Adları Gizlilik İletişim

Access Rules

Firewall rules can be based on IP address, IP address range, Autonomous System Number (ASN) or country.

Search Access Rules

Enter an IP, IP range, country name, or ASN

Block

This website

Add a note

Add

Value	Applies to	Action
France (FR) HTTP Flood Sebebiyle	This website	Block  
51.254.122.14 HTTP Flood	This website	Block  
Russian Federation (RU) HTTP Flood Sebebiyle	This website	Block  

23 Temmuz tarihinde yapılan yeni bir DoS saldırısında bu defa blogumun arama sayfasına çok sayıda i_love_you_brother arama terimi gönderildiğini gördüm. Bu defa kaynak IP adresini (31.223.24.59) görebildiğim için IP adresinin kayıt bilgilerine baktığımda bu defa saldırının Türkiye'den yapıldığını gördüm.

```
root@batcave:/var/log/nginx# grep 1_love_you access.log | head -n 10
31.223.24.59 - - [23/Jul/2018:20:04:32 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14217 "-" "Mozilla/5.0 (windows NT 6.1) AppleWebKit/537.2 (KHTML, like Gecko) Chrome/22.0.1216.0 Safari/537.2"
31.223.24.59 - - [23/Jul/2018:20:04:32 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14217 "-" "Mozilla/5.0 (windows NT 6.2) AppleWebKit/537.11 (KHTML, like Gecko) Chrome/23.0.1271.26 Safari/537.11"
31.223.24.59 - - [23/Jul/2018:20:04:32 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14218 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_6_8) AppleWebKit/535.19 (KHTML, like Gecko) Chrome/18.0.1025.11 Safari/535.19"
31.223.24.59 - - [23/Jul/2018:20:04:33 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14216 "-" "Mozilla/5.0 (windows NT 6.0; WOW64) AppleWebKit/535.19 (KHTML, like Gecko) Chrome/18.0.1025.45 Safari/535.19"
31.223.24.59 - - [23/Jul/2018:20:04:34 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14217 "-" "Mozilla/5.0 (windows NT 6.2; WOW64) AppleWebKit/537.1 (KHTML, like Gecko) Chrome/19.77.34.5 Safari/537.1"
31.223.24.59 - - [23/Jul/2018:20:04:34 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14218 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_2) AppleWebKit/537.11 (KHTML, like Gecko) Chrome/23.0.1271.6 Safari/537.11"
31.223.24.59 - - [23/Jul/2018:20:04:35 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14216 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_2) AppleWebKit/535.24 (KHTML, like Gecko) Chrome/19.0.1055.1 Safari/535.24"
31.223.24.59 - - [23/Jul/2018:20:04:35 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14217 "-" "Mozilla/5.0 (windows NT 6.2) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1061.1 Safari/536.3"
31.223.24.59 - - [23/Jul/2018:20:04:35 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14218 "-" "Mozilla/5.0 (windows NT 6.1; WOW64) AppleWebKit/537.1 (KHTML, like Gecko) Chrome/22.0.1207.1 Safari/537.1"
31.223.24.59 - - [23/Jul/2018:20:04:35 +0300] "GET /?s=1_love_you_brother HTTP/1.1" 200 14217 "-" "Mozilla/5.0 (windows NT 5.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/35.0.2309.372 Safari/537.36"
root@batcave:/var/log/nginx#
```

WHOIS IP Lookup Tool | X

Secure | <https://www.ultratools.com/tools/ipWhoisLookupResult>

Hack 4 Career: Info | LinkedIn | Mert SARICA (mertsarica) | Inbox - mertsarica

IP Tools

- Decimal IP Calculator
- ASN Information
- CIDR/Netmask
- What's your IP
- IP Geo-location Lookup
- IPWHOIS Lookup

WHOIS IP Lookup Tool

The IPWHOIS Lookup tool finds contact information for the owner of a specified IP address.

Enter a host name or an IP address:

31.223.24.59

Related Tools: [DNS Traversal](#) [Traceroute](#) [Vector Trace](#) [Ping](#) [WHOIS Lookup](#)

```
Source: whois.ripe.net
IP Address: 31.223.24.59

% This is the RIPE Database query service.
% The objects are in RPSL format.
%
% The RIPE Database is subject to Terms and Conditions.
% See http://www.ripe.net/db/support/db-terms-conditions.pdf

% Note: this output has been filtered.
% To receive output for a database update, use the "-b" flag.

% Information related to '31.223.24.0 - 31.223.24.255'
% Abuse contact for '31.223.24.0 - 31.223.24.255' is 'li@turknet.net.tr'

inetnum: 31.223.24.0 - 31.223.24.255
netname: AnkaraPOP_XdslDynamic
descr: TurkNet-DSL
country: TR
admin-c: TL143-RIPE
tech-c: TL143-RIPE
status: ASSIGNED PA
mnt-by: NMT-TURKNET-WMT
created: 2011-07-26T13:53:03Z
last-modified: 2011-07-26T13:53:03Z
source: RIPE

person: TurkNet LIR
address: TurkNet Iletisim Hizmetleri A.S.
address: Buyukdere Cad. Ercan Han No.121
address: Gayrettepe / Istanbul / Turkey
phone: +90 212 355 17 00
nic-hdl: TL143-RIPE
created: 2009-03-05T10:03:41Z
last-modified: 2011-08-24T12:18:33Z
source: RIPE
mnt-by: NMT-TURKNET-WMT

% Information related to '31.223.0.0/17AS12735'

route: 31.223.0.0/17
descr: TurkNet Iletisim Hizmetleri A.S
origin: AS12735
mnt-by: NMT-TURKNET-WMT
created: 2011-05-16T14:17:07Z
last-modified: 2011-05-16T14:17:07Z
source: RIPE

% This query was served by the RIPE Database Query Service version 1.91.2 (WAGYU)
```

Bana olan sevgisini sele dönüştürme konusunda ısrarcı olan bu Rus kardeşim ile sistematik olarak mücadele edebilmek ve benzer durumlarla mücadele etmek durumunda kalanlara yol gösterme adına çeşitli araçlardan faydalanmaya karar verdim.

İlk olarak gerçekleştirilen genel DoS saldırılarından hızlıca haberdar olabilmek için OSSEC HIDS yazılımını işletim sistemine kurdum. Buna ilaveten bu saldırıyı tespit edebilmek için de özel olarak bir kural hazırlayıp bu kuralı /var/ossec/rules/http_dos_rules.xml adı altında kaydettikten sonra kuralı içeren dosyayı da /var/ossec/etc/ossec.conf dosyasına kaydettim. ossec-logtest aracı ile de kuralın düzgün çalıştığını kontrol ettikten sonra OSSEC'i yeniden başlatıp attack.py ile bloguma ufak bir DoS saldırısı gerçekleştirdim. Saldırı sonucunda OSSEC başarıyla bu saldırıyı tespit edip beni e-posta ile hemen uyardı.

```
GNU nano 2.5.3 File: ossec.conf
<include>cmsserver_rules.xml</include>
<include>vpopmail_rules.xml</include>
<include>vmpop3d_rules.xml</include>
<include>courier_rules.xml</include>
<include>web_rules.xml</include>
<include>web_appsec_rules.xml</include>
<include>apache_rules.xml</include>
<include>nginx_rules.xml</include>
<include>php_rules.xml</include>
<include>mysql_rules.xml</include>
<include>postgresql_rules.xml</include>
<include>ids_rules.xml</include>
<include>sqld_rules.xml</include>
<include>firewall_rules.xml</include>
<include>apparmor_rules.xml</include>
<include>cisco-ios_rules.xml</include>
<include>netscreenfw_rules.xml</include>
<include>sonicwall_rules.xml</include>
<include>postfix_rules.xml</include>
<include>sendmail_rules.xml</include>
<include>imap_rules.xml</include>
<include>mailscanner_rules.xml</include>
<include>dovecot_rules.xml</include>
<include>ms-exchange_rules.xml</include>
<include>racon_rules.xml</include>
<include>vpn_concentrator_rules.xml</include>
<include>spand_rules.xml</include>
<include>msauth_rules.xml</include>
<include>mcafee_av_rules.xml</include>
<include>trend-osec_rules.xml</include>
<include>ms-se_rules.xml</include>
<!-- <include>policy_rules.xml</include> -->
<include>zeus_rules.xml</include>
<include>solaris_bsm_rules.xml</include>
<include>vmware_rules.xml</include>
<include>ms_dhcp_rules.xml</include>
<include>asterisk_rules.xml</include>
<include>ossec_rules.xml</include>
<include>attack_rules.xml</include>
<include>dropbear_rules.xml</include>
<include>unbound_rules.xml</include>
<include>sysmon_rules.xml</include>
<include>opensmtpd_rules.xml</include>
<include>exim_rules.xml</include>
<include>openbsd-dhcpd_rules.xml</include>
<include>local_rules.xml</include>
</rules>

<syscheck>
<!-- Frequency that syscheck is executed -- default every 20 hours -->
<frequency>7200</frequency>
<!-- Directories to check (perform all possible verifications) -->
<directories check_all="yes">/etc,/usr/bin,/usr/sbin</directories>
<directories check_all="yes">/bin,/sbin,/boot</directories>
<!-- Files/directories to ignore -->
<ignore>/etc/mtab</ignore>
<ignore>/etc/hosts.deny</ignore>
<ignore>/etc/mail/statistics</ignore>
<ignore>/etc/random-seed</ignore>
<ignore>/etc/random-seed</ignore>
</syscheck>

<!-- @(#) $Id: ./var/ossec/rules/http_dos_rules.xml, 2018/07/23 Mert SARICA Exp $
-
- HTTP DoS attacks/vulns specific rules for OSSEC.
-
- Copyright (C) 2012 Daniel B. Cid (dcid@dcid.me)
- All rights reserved.
-
- This program is a free software; you can redistribute it
- and/or modify it under the terms of the GNU General Public
- License (version 2) as published by the FSF - Free Software
- Foundation.
-
- License details: http://www.ossec.net/en/licensing.html
-->

<!-- Collection of rules for common web attacks that we are seeing in the wild.
- The real goal is to stop bots and automated attacks from doing further damage
- on sites that are not updated.
-->
<group name="web,appsec,attack">

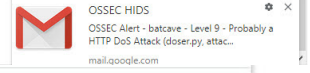
  <!-- Checking POST & GET HTTP DoS Attack
  -->
  <rule id="41501" level="1">
    <if_sid>31100</if_sid>
    <match>GET |POST </match>
    <regex>/?\D+=\D+HTTP/1.1</regex>
    <description>Probably a non-malicious site search</description>
  </rule>

  <!-- If we see frequent /?VXVFII=FXSF HTTP/1.1 then it is probably a HTTP DoS. -->
  <rule id="41502" level="9" frequency="10" timeframe="20">
    <if_matched_sid>41501</if_matched_sid>
    <same_source_ip />
    <description>Probably a HTTP DoS Attack (doser.py, attack.py or hulk.py)</description>
  </rule>

  <!-- Checking GET HTTP DoS Attack
  <rule id="41503" level="9">
    <if_sid>31100</if_sid>
    <match>GET /?s=i_love_you_brother</match>
    <description>HTTP DoS Attack (i_love_you_brother)</description>
  </rule>
  -->
</group>
```



```
root@ubuntu:~/dos# python attack.py http://www.mertsarica.com/
Debug : thread=5 time=30 http://www.mertsarica.com/
Time is up !
Attack Finished !
root@ubuntu:~/dos#
```



OSSEC Alert - batcave - Level 9 - Probably a HTTP DoS Attack (doser.py, attack.py or hulk.py)



OSSEC HIDS <mert.sarica@gmail.com>
to me

11:20 PM (0 minutes ago)



Be careful with this message

Gmail could not verify that it actually came from mert.sarica@gmail.com. Avoid clicking links, downloading attachments, or replying with personal information.

Report As Spam

Report As Phishing



OSSEC HIDS Notification.
2018 Jul 23 23:20:18

Received From: batcave->/var/log/nginx/access.log
Rule: 41502 fired (level 9) -> "Probably a HTTP DoS Attack (doser.py, attack.py or hulk.py)"
Src IP: [REDACTED]
Portion of the log(s):

```
-- [23/Jul/2018:23:20:17 +0300] "GET /?FLEDNO=IQOKGUIUYA HTTP/1.1" 200 19955 "http://www.usatoday.com/search/results?q=MDPNECW" "Mozilla/5.0 (Windows; U; Windows NT 6.1; en; rv:1.9.1.3) Gecko/20090824 Firefox/3.5.3 (NET CLR 3.5.30729)"
-- [23/Jul/2018:23:20:15 +0300] "GET /?ECO=NYNGERU HTTP/1.1" 200 19952 "http://engadget.search.aol.com/search?q=JHYFFX" "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1"
-- [23/Jul/2018:23:20:15 +0300] "GET /?ECO=NYNGERU HTTP/1.1" 200 19957 "http://engadget.search.aol.com/search?q=JHYFFX" "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1"
-- [23/Jul/2018:23:20:15 +0300] "GET /?ECO=NYNGERU HTTP/1.1" 200 19957 "http://engadget.search.aol.com/search?q=JHYFFX" "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1"
-- [23/Jul/2018:23:20:15 +0300] "GET /?NBQRRYKCS=WSHMMVEU HTTP/1.1" 200 19955 "http://engadget.search.aol.com/search?q=VHJAOOBNE" "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Win64; x64; Trident/4.0)"
-- [23/Jul/2018:23:20:15 +0300] "GET /?NBQRRYKCS=WSHMMVEU HTTP/1.1" 200 19956 "http://engadget.search.aol.com/search?q=VHJAOOBNE" "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Win64; x64; Trident/4.0)"
-- [23/Jul/2018:23:20:13 +0300] "GET /?ECO=NYNGERU HTTP/1.1" 200 19955 "http://engadget.search.aol.com/search?q=JHYFFX" "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1"
-- [23/Jul/2018:23:20:13 +0300] "GET /?GRFB=INLOWJKU HTTP/1.1" 200 19956 "http://www.mertsarica.com/HWEARASA" "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1"
-- [23/Jul/2018:23:20:13 +0300] "GET /?NBQRRYKCS=WSHMMVEU HTTP/1.1" 200 19954 "http://engadget.search.aol.com/search?q=VHJAOOBNE" "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Win64; x64; Trident/4.0)"
-- [23/Jul/2018:23:20:13 +0300] "GET /?NBQRRYKCS=WSHMMVEU HTTP/1.1" 200 19957 "http://engadget.search.aol.com/search?q=VHJAOOBNE" "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Win64; x64; Trident/4.0)"
-- [23/Jul/2018:23:20:13 +0300] "GET /?NBQRRYKCS=WSHMMVEU HTTP/1.1" 200 19957 "http://engadget.search.aol.com/search?q=VHJAOOBNE" "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Win64; x64; Trident/4.0)"
-- [23/Jul/2018:23:20:11 +0300] "GET /?GRFB=INLOWJKU HTTP/1.1" 200 19956 "http://www.mertsarica.com/HWEARASA" "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.1) Gecko/20090718 Firefox/3.5.1"
```

OSSEC HIDS ile benim sevgi dolu Rus kardeşimin DoS saldırısını tespit etmek yeterli olmayacağı için fail2ban aracı ile bu iki tür DoS saldırısını tespit eden özel bir kural yazıp, bu yazıda da belirtildiği şekilde Cloudflare

API'si üzerinden saldırganın ip adresini yasaklayan betikten faydalandım. Hazırladığım kuralı kontrol etmek için fail2ban-regex aracından faydalandıktan sonra attack.py ile bloguma saldırarak Cloudflare tarafından ip adresimin başarıyla yasaklandığını gördüm ve Rus kardeşimin yeni saldırıları için kedi fare oyunundaki yerimi sağlamlaştırdım. :)

```
GNU nano 2.5.3 File: /etc/fail2ban/filter.d/nginx-dos.conf
[Definition]
# Probably a HTTP Dos Attack (doser.py, attack.py or hulk.py)
failregex = ^<HOST> \- .* \"(POST|GET).*(/[A-Z]+= [A-Z]+ HTTP/1.1)\" .+$

# HTTP Dos Attack (i_love_you_brother)
# ^<HOST> \- .* \"(POST|GET).*(/?s=i_love_you_brother HTTP/1.1)\" .+$

# HTTP Dos Attack (this_is_another_test)
# ^<HOST> \- .* \"(POST|GET).*(/?s=).*(_).*(_).*(_) (HTTP/1.1)\" .+$

# HTTP Dos Attack (x_y_z)
# ^<HOST> \- .* \"(POST|GET).*(/?s=).*(_).*(_) (HTTP/1.1)\" .+$

# HTTP Dos Attack (python-requests)
# ^<HOST> \- .* \"python-requests/.*)\" .+$

ignoreregex =

root@batcave:/var/log/nginx# fail2ban-regex /var/log/nginx/access.log /etc/fail2ban/filter.d/nginx-dos.conf

Running tests
=====
Use failregex filter file : nginx-dos, basedir: /etc/fail2ban
Use log file : /var/log/nginx/access.log
Use encoding : UTF-8

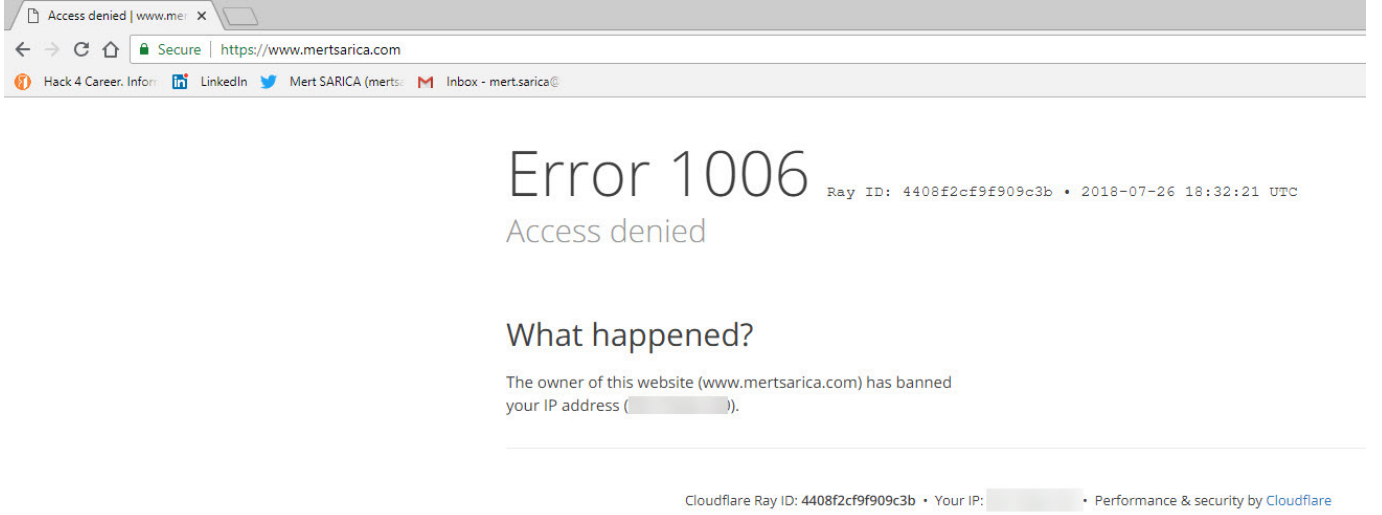
Results
=====
Failregex: 6317 total
|- #) [# of hits] regular expression
| 1) [2006] ^<HOST> \- .* \"(POST|GET).*(/[A-Z]+= [A-Z]+ HTTP/1.1)\" .+$
| 2) [4311] ^<HOST> \- .* \"(POST|GET).*(/?s=).*(_).*(_) (HTTP/1.1)\" .+$
|-

Ignoreregex: 0 total

Date template hits:
|- [# of hits] date format
| [25807] Day(?P<_sep>[-/])MON(?P=_sep)Year[ :]?24hour:Minute:Second(?:\.Microseconds)?(?: Zone offset)?
|-

Lines: 25807 lines, 0 ignored, 6317 matched, 19490 missed [processed in 3.11 sec]
Missed line(s): too many to print. Use --print-all-missed to print all 19490 lines
root@batcave:/var/log/nginx#

root@batcave:~# service fail2ban restart
root@batcave:~# tail -f /var/log/fail2ban.log
2018-07-26 21:44:57,925 fail2ban.jail [12335]: INFO Jail 'nginx-dos' uses pyinotify
2018-07-26 21:44:57,925 fail2ban.filter [12335]: INFO Set jail log file encoding to UTF-8
2018-07-26 21:44:57,929 fail2ban.jail [12335]: INFO Initiated 'pyinotify' backend
2018-07-26 21:44:57,931 fail2ban.filter [12335]: INFO Set maxRetry = 3
2018-07-26 21:44:57,932 fail2ban.filter [12335]: INFO Added logfile = /var/log/nginx/access.log
2018-07-26 21:44:57,934 fail2ban.filter [12335]: INFO Set findtime = 60
2018-07-26 21:44:57,935 fail2ban.actions [12335]: INFO Set banTime = 6000
2018-07-26 21:44:57,935 fail2ban.filter [12335]: INFO Set jail log file encoding to UTF-8
2018-07-26 21:44:57,939 fail2ban.jail [12335]: INFO Jail 'sshd' started
2018-07-26 21:44:57,944 fail2ban.jail [12335]: INFO Jail 'nginx-dos' started
2018-07-26 21:45:24,929 fail2ban.filter [12335]: INFO [nginx-dos] Found
2018-07-26 21:45:24,932 fail2ban.filter [12335]: INFO [nginx-dos] Found
2018-07-26 21:45:24,996 fail2ban.filter [12335]: INFO [nginx-dos] Found
2018-07-26 21:45:25,263 fail2ban.filter [12335]: INFO [nginx-dos] Found
2018-07-26 21:45:25,489 fail2ban.filter [12335]: INFO [nginx-dos] Found
```



Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.