

WEB

APPLICATION

FIREWALL



Small hand book for cybersecurity knowledge

by Chaos



⚠ warning

**This document is only for
educational purposes.**

**The author will approve of
no abusage.**



فهرست

۳	WAF مفهوم
۴	نحوه کارکرد WAF
۵	انواع WAF
۶	مزایای WAF
۷	WAF اهمیت
۷	فروشنده‌گان سرویس‌های ابری WAF
۸	سرویس WAF در مقایسه با Firewall و IPS
۱۰	کاهش آسیب پذیری‌ها در WAF
۱۲	روش‌های گذر کردن از WAF
۱۶	ابزار‌های عالی برای رد شدن از WAF



WAF مفهوم

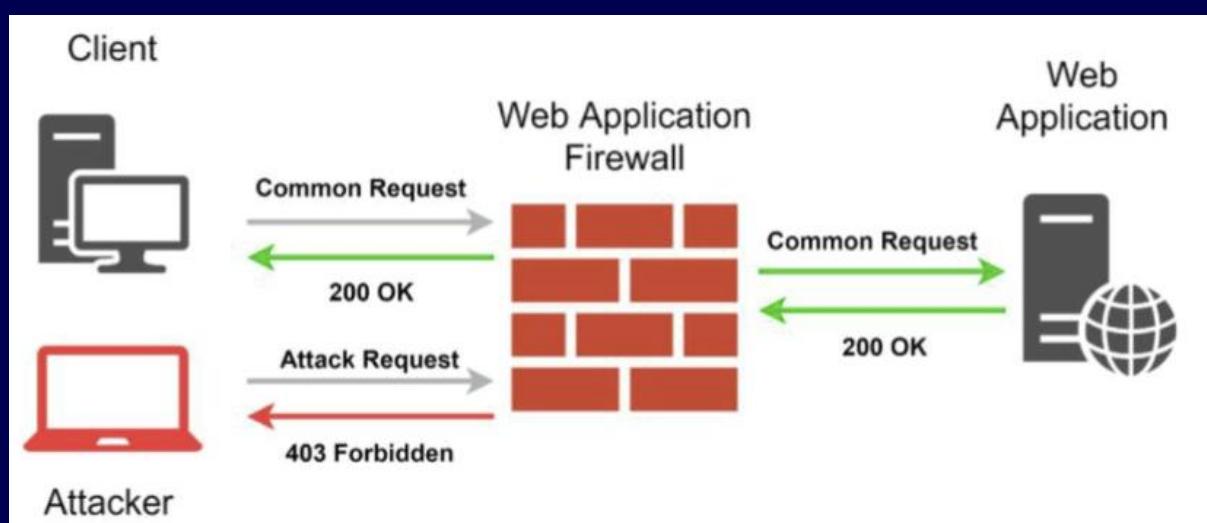
«فایروال برنامه وب (WAF)» یک فایروال برنامه برای برنامه‌های HTTP است. یک WAF از طریق مجموعه‌ای از قوانین که اغلب سیاست نامیده می‌شوند، عمل می‌کند. این سیاست‌ها با WAF فیلتر کردن ترافیک مخرب، از آسیب‌پذیری‌های برنامه محافظت می‌کنند. با استقرار یک WAF در مقابل یک برنامه وب، سپری بین برنامه وب و اینترنت قرار می‌گیرد. در حالی که یک پروکسی سرور با استفاده از یک واسطه از هویت دستگاه کلاینت محافظت می‌کند، WAF نوعی پروکسی معکوس است که با عبور کلاینت‌ها از WAF قبل از رسیدن به سرور، از سرور در برابر افشا محافظت می‌کند. این فایروال از برنامه‌های وب در برابر انواع حملات لایه برنامه مانند اسکریپتنویسی بین سایتی (XSS)، تزریق SQL و مسمومیت کوکی و موارد دیگر محافظت می‌کند. حملات به برنامه‌ها علت اصلی نقض‌ها هستند – آنها دروازه‌ای به داده‌های ارزشمند شما هستند. با استفاده از WAF مناسب، می‌توانید مجموعه‌ای از حملاتی را که با هدف استخراج آن داده‌ها با به خطر انداختن سیستم‌های شما انجام می‌شوند، مسدود کنید.



نحوه کارکرد WAF

• یک WAF با فیلتر کردن، نظارت و مسدود کردن هرگونه ترافیک HTTP/S مخرب که به برنامه وب منتقل می‌شود، از برنامه‌های وب شما محافظت می‌کند و از خروج هرگونه داده غیرمجاز از برنامه جلوگیری می‌کند. این کار را با رعایت مجموعه‌ای از سیاست‌ها انجام می‌دهد که به تعیین اینکه کدام ترافیک مخرب و کدام ترافیک ایمن است، کمک می‌کند. همانطور که یک سرور پروکسی به عنوان واسطه‌ای برای محافظت از هویت یک کلاینت عمل می‌کند، یک WAF نیز به روشهای مشابه عمل می‌کند اما برعکس - که پروکسی معکوس نامیده می‌شود - به عنوان واسطه‌ای عمل می‌کند که سرور برنامه وب را از یک کلاینت بالقوه مخرب محافظت می‌کند.

• WAF‌ها می‌توانند به شکل نرم‌افزار، یک دستگاه یا به صورت سرویس ارائه شوند. سیاست‌ها را می‌توان برای برآورده کردن نیازهای منحصر به فرد برنامه وب یا مجموعه‌ای از برنامه‌های وب شما سفارشی کرد. اگرچه بسیاری از WAF‌ها از شما می‌خواهند که سیاست‌ها را به طور منظم برای رفع آسیب‌پذیری‌های جدید به‌روزرسانی کنید، پیشرفت‌ها در یادگیری ماشینی برخی از WAF‌ها را قادر می‌سازد تا به طور خودکار به‌روزرسانی شوند. این اتوماسیون با افزایش پیچیدگی و ابهام چشم‌انداز تهدید، حیاتی‌تر می‌شود.



انواع WAF

یک WAF می‌تواند به یکی از سه روش مختلف پیاده‌سازی شود که هر کدام مزايا و معایب خاص خود را دارند:

❖ یک WAF مبتنی بر شبکه (Network – based) عموماً مبتنی بر سخت‌افزار نامیده می‌شود. از آنجایی که به صورت محلی نصب می‌شوند، تأخیر را به حداقل می‌رسانند، اما WAF‌های مبتنی بر شبکه گران‌ترین گزینه هستند و همچنین به ذخیره‌سازی و نگهداری تجهیزات فیزیکی نیاز دارند.

❖ یک WAF مبتنی بر میزبان (Host – based) ممکن است به طور کامل در نرم‌افزار یک برنامه که به آن نرم‌افزار محور نیز گفته می‌شود، ادغام شود. این راه حل ارزان‌تر از WAF مبتنی بر شبکه است و قابلیت سفارشی‌سازی بیشتری را ارائه می‌دهد. نقطه ضعف WAF مبتنی بر میزبان، مصرف منابع سرور محلی، پیچیدگی پیاده‌سازی و هزینه‌های نگهداری است. این اجزا معمولاً به زمان مهندسی نیاز دارند و ممکن است پرهزینه باشند.

❖ WAF‌های مبتنی بر فضای ابری (Cloud – based)، گزینه‌ای مقرن به صرفه ارائه می‌دهند که پیاده‌سازی آن بسیار آسان است. آنها معمولاً نصب کلید در دست را ارائه می‌دهند که به سادگی تغییر DNS برای هدایت ترافیک است. WAF‌های مبتنی بر ابر همچنین هزینه اولیه کمی دارند، زیرا کاربران ماهانه یا سالانه برای امنیت به عنوان یک سرویس هزینه پرداخت می‌کنند. WAF‌های مبتنی بر ابر همچنین می‌توانند راهکاری ارائه دهند که به طور مداوم به روزرسانی می‌شود تا در برابر جدیدترین تهدیدات محافظت کند، بدون اینکه هیچ کار یا هزینه اضافی از طرف کاربر تحمیل شود.



WAF مزایای

یک WAF نسبت به فایروال‌های سنتی مزیتی دارد زیرا دید بیشتری به داده‌های حساس برنامه‌های کاربردی که با استفاده از لایه برنامه HTTP منتقل می‌شوند، ارائه می‌دهد. این فایروال می‌تواند از حملات لایه برنامه کاربردی که معمولاً از فایروال‌های شبکه سنتی عبور می‌کنند، از جمله موارد زیر، جلوگیری کند:

- حملات اسکریپتنویسی بین سایتی (XSS) به مهاجمان امکان می‌دهد اسکریپتهاي مخرب را در مرورگر کاربر دیگر تزریق و اجرا کنند.
 - حملات تزریق زبان‌های کوئری دار ساختاریافته (SQL) می‌تواند بر هر برنامه‌ای که از پایگاه داده SQL استفاده می‌کند، تأثیر بگذارد و به مهاجمان امکان دسترسی و تغییر بالقوه داده‌های حساس را می‌دهد.
 - Web session hacking به مهاجمان امکان می‌دهد تا شناسه جلسه را بدزدند و خود را به عنوان یک کاربر مجاز جا بزنند. شناسه جلسه معمولاً در یک کوکی یا URL ذخیره می‌شود.
 - حملات (DDoS) با پر کردن شبکه با ترافیک، آن را تا زمانی که قادر به ارائه خدمات به کاربران خود نباشد، تحت الشعاع قرار می‌دهند. هم فایروال‌های شبکه و هم WAF‌ها می‌توانند این نوع حمله را مدیریت کنند، اما از لایه‌های مختلف به آن نزدیک می‌شوند.
- یکی دیگر از مزایای WAF این است که می‌تواند از برنامه‌های مبتنی بر وب بدون نیاز به دسترسی به کد منبع برنامه، دفاع کند. در حالی که یک WAF مبتنی بر میزبان ممکن است در کد برنامه ادغام شود، یک WAF مبتنی بر ابر قادر است بدون دسترسی به کد برنامه، از آن دفاع کند. علاوه بر این، استقرار و مدیریت یک WAF ابری آسان است و راه حل‌های وصله‌گذاری مجازی سریعی را ارائه می‌دهد که کاربران را قادر می‌سازد تا به سرعت تنظیمات خود را برای سازگاری با تهدیدهای تازه شناسایی شده سفارشی کنند.



WAF اهمیت

یک WAF برای تعداد فزاینده‌ای از شرکت‌هایی که محصولات خود را از طریق اینترنت ارائه می‌دهند - از جمله بانکداران آنلاین، ارائه دهنده‌گان پلتفرم رسانه‌های اجتماعی و توسعه دهنده‌گان برنامه‌های تلفن همراه - مهم است زیرا به جلوگیری از نشت داده‌ها کمک می‌کند.

بسیاری از داده‌های حساس، مانند داده‌های کارت اعتباری و سوابق مشتری، در پایگاه‌های داده پشتیبان ذخیره می‌شوند که از طریق برنامه‌های وب قابل دسترسی هستند. مهاجمان اغلب این برنامه‌ها را برای دسترسی به داده‌های مرتبط هدف قرار می‌دهند.

فروشنده‌گان سرویس‌های ابری WAF



سرویس WAF در مقایسه با IPS و Firewall

IPS یک سیستم پیشگیری از نفوذ، WAF یک فایروال برنامه وب و NGFW یک فایروال نسل بعدی است. تفاوت بین همه آنها چیست؟ IPS یک محصول امنیتی با تمرکز گسترده‌تر است. معمولاً مبتنی بر امضا و سیاست است - به این معنی که می‌تواند آسیب‌پذیری‌های شناخته شده و بردارهای حمله را بر اساس پایگاه داده امضا و سیاست‌های تعیین شده بررسی کند. IPS استانداردی را بر اساس پایگاه داده و سیاست‌ها ایجاد می‌کند، سپس در صورت انحراف هرگونه ترافیک از استاندارد، هشدارهایی ارسال می‌کند. امضاها و سیاست‌ها با گذشت زمان و با شناخته شدن آسیب‌پذیری‌های جدید، رشد می‌کنند. به طور کلی، IPS از ترافیک در طیف وسیعی از انواع پروتکل‌ها مانند DNS، RDP، TELNET، SMTP و FTP محافظت می‌کند. معمولاً در لایه‌های ۳ و ۴ عمل می‌کند و از لایه‌های شبکه و جلسه محافظت می‌کند، اگرچه برخی ممکن است در لایه برنامه (لایه ۷) محافظت محدودی ارائه دهند.

یک فایروال نسل بعدی (NGFW) ترافیک خروجی به اینترنت را - از طریق وب‌سایتها، حساب‌های ایمیل و SaaS - نظارت می‌کند. به عبارت ساده، از کاربر (در مقابل برنامه وب) محافظت می‌کند. یک NGFW سیاست‌های مبتنی بر کاربر را اجرا می‌کند و علاوه بر افزودن ویژگی‌هایی مانند فیلتر کردن URL، آنتی‌ویروس/ ضد بدافزار و احتمالاً سیستم‌های پیشگیری از نفوذ (IPS) خود، زمینه را به سیاست‌های امنیتی اضافه می‌کند. در حالی که یک WAF معمولاً یک پروکسی معکوس (مورد استفاده سرورها) است، NGFW‌ها اغلب پروکسی‌های رو به جلو (مورد استفاده کلاینت‌هایی مانند مرورگر) هستند.



یک فایروال برنامه وب (WAF) از لایه برنامه محافظت می‌کند و به طور خاص برای تجزیه و تحلیل هر درخواست HTTP/S در لایه برنامه طراحی شده است. این فایروال معمولاً از کاربر، جلسه و برنامه آگاه است و از برنامه‌های وب پشت آن و خدماتی که ارائه می‌دهند آگاه است. به همین دلیل، می‌توانید WAF را به عنوان واسطه‌ای بین کاربر و خود برنامه در نظر بگیرید که تمام ارتباطات را قبل از رسیدن به برنامه یا کاربر تجزیه و تحلیل می‌کند. WAF‌های سنتی تضمین می‌کنند که فقط اقدامات مجاز (بر اساس سیاست امنیتی) می‌توانند انجام شوند. برای بسیاری از سازمان‌ها، WAF‌ها یک خط دفاعی مطمئن و اولیه برای برنامه‌های کاربردی هستند، به خصوص برای محافظت در برابر ۱۰ مورد برتر OWASP - فهرست اساسی آسیب پذیری‌های برنامه کاربردی که بیشترین دیده شده را دارند.



کاهش آسیب پذیری ها در WAF

بنابراین، دقیقاً چگونه یک WAF همه این آسیب پذیری ها را کاهش می دهد؟ سه روش اصلی وجود دارد که یک WAF برای شناسایی و جلوگیری از حملات وب استفاده می کند:

deny/allow requests, inspect and reject, and signatures.

بیایید هر کدام را بررسی کنیم، موافقید؟

Deny/Allow Requests ➤

این روش درخواست ها بسیار شبیه مدل درگاه سنتی است که توسط فایروال های شبکه استفاده می شود. درخواست ها بر اساس اطلاعات موجود یا رد می شوند یا مجاز. این اطلاعات ممکن است ساده باشند - مانند یک آدرس IP - یا ممکن است پیچیده تر و مختص HTTP باشند، مانند .METHODS یا OPTIONS

Signatures ➤

امضاها یکی دیگر از روش های محافظتی رایج در بسیاری از راه حل های امنیتی مختلف هستند. سرویس های آنتی ویروس و ضد بدافزار به امضاها یی متکی هستند که آنها را قادر می سازد تا به سرعت شواهدی از ویروس ها و بدافزارها را اسکن کرده و آنها را علامت گذاری کنند. IPS/IDS نیز مانند WAF به شدت به این روش متکی هستند. دو نوع امضا وجود دارد: تعریف شده توسط کاربر و مدیریت شده توسط فروشنده.



Inspection ➤

در نهایت، بازرسی برای اطمینان از کنترل کامل بر درخواست‌ها (و پاسخ‌ها) گنجانده شده است. بازرسی درخواست‌ها به WAF اجازه می‌دهد تا اطلاعات موجود در درخواست را با رشته‌ها و مقادیر خوب یا بد شناخته شده مقایسه کند تا مشخص شود که آیا درخواست مخرب است یا مشروع. برای برنامه‌های HTTP (که به معنای اکثر آنها در اینترنت امروزی است) این مهمترین قابلیتی است که یک WAF باید ارائه دهد. اگر یک WAF بازرسی قابل برنامه‌ریزی ارائه نمی‌دهد، باید در این انتخاب تجدید نظر کنید. از آنجا که HTTP مبتنی بر متن و قابل توسعه است، عملً هیچ راهی برای ارائه یک لیست جامع "چکباکس" از گزینه‌ها و روش‌هایی که می‌توانید برای بازرسی درخواست‌ها استفاده کنید، وجود ندارد. تعداد بسیار کمی از هدرهای HTTP گزینه‌های محدود شده دارند، که محدود کردن آنچه می‌توان و نمی‌توان در آن قرار داد را بسیار دشوار می‌کند. این بدان معناست که بازرسی اغلب برای شناسایی کد مخرب تعییه شده در هدرها یا در خود payload مورد نیاز است. دو راه برای استفاده از بازرسی وجود دارد: هدرهای شناخته شده و payload.



روش های گندر کردن از WAF

۱. تکنیک ترکیب حروف بزرگ و کوچک (Case Toggling)

استفاده از حروف بزرگ و کوچک در کلمات کلیدی باعث دور زدن فایروال هایی می شود که به صورت حساس به حروف (case-sensitive) پیکربندی شده اند.

```
● ● ●  
<ScRIPt>confirm( )</sCRIPt>
```

۲. استفاده از کامنت ها (Using Comments)

با درج کامنت بین توابع یا دستورات، برخی از فایروال ها را فریب داده و عبور از فیلتر را ممکن می سازید.

```
● ● ●  
<!--><script>confirm/**/( )/**/</script>
```

۳. تزریق نویسه (Null Character Injection)

قرار دادن ۰۰٪ قبل از کد مخرب. فایروال ممکن است بقیه رشته را نادیده بگیرد، اما وب سرور همچنان آن را پردازش می کند.

۴. نظرات درون خطی (Inline Comments)

استفاده از /* */ برای عبور از فایروال و رسیدن به پایگاهداده:

```
● ● ●  
/*!SELECT*/ * FROM users;
```



۵. رمزنگاری با حالت (DHE/EDH) Ephemeral

رمزنگاری با کلید موقت باعث می‌شود فایروال نتواند ترافیک را رمزگشایی کرده و بررسی کند. مخصوصاً وقتی فایروال نتواند تبادل کلید را شنود کند.

۶. سریز بافر (Buffer Overflow)

فایروال نیز یک برنامه نرم‌افزاری است. آسیب‌پذیری‌هایی مانند سریز بافر ممکن است باعث از کار افتادن (fail open) آن شود.

۷. آلودگی پارامتر (HTTP Parameter Pollution)

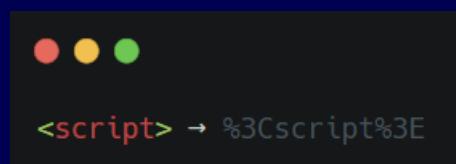
ارسال چند پارامتر با نام یکسان برای گمراه کردن فایروال:



```
● ● ●  
http://example.com?id=1&id=' OR '1'='1
```

۸. رمزگذاری URL (Hex Encoding)

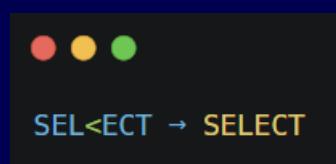
استفاده از مقادیر هگزادسیمال برای نویسه‌ها مثل:



```
● ● ●  
<script> → %3Cscript%3E
```

۹. شکستن کلیدواژه‌ها (Keyword Splitting)

گنجاندن نویسه‌های خاص در میان کلیدواژه‌ها تا پس از حذف توسط فایروال، رشته اصلی ساخته شود:

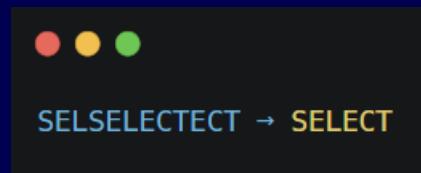


```
● ● ●  
SEL<ECT → SELECT
```



۱۰. جایگزینی کلیدواژه‌ها (Replaced Keywords)

استفاده از تکرار یا پوشاندن کلیدواژه‌ها برای گمراهی فیلتر:



۱۱. نادیده گرفتن کوکی‌ها (Ignoring Cookies)

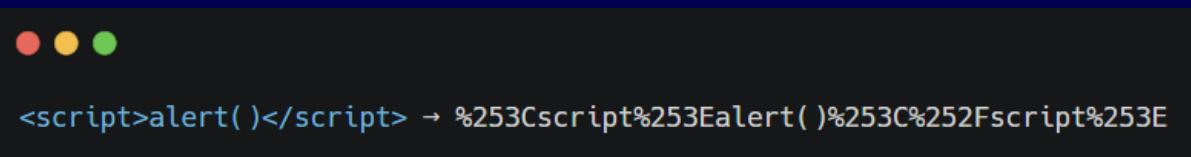
نادیده گرفتن کوکی‌هایی که فایروال به مرورگر می‌دهد تا کاربر مخرب را شناسایی کند.

۱۲. استفاده از Data URI

رمزگذاری محتوای مخرب به صورت URI داده‌ای که توسط مرورگر تفسیر شده ولی ممکن است توسط WAF نادیده گرفته شود.

۱۳. رمزگذاری دوگانه (Double Encoding)

استفاده از دو بار رمزگذاری برای گمراه کردن فیلترهای ضعیف:



۱۴. استفاده از خط جدید (Line Breaks)

قرار دادن کاراکترهای CR/LF (خط جدید) برای شکستن regex فایروال:

```
● ● ●  
<iframe src="%0Aj%0Aa%0Av%0Aa...>
```

۱۵. نویسه‌های بی‌ارزش (Junk Characters)

استفاده از کاراکترهای نامعتبر یا اضافی در ساختار حمله برای فریب فایروال:

```
● ● ●  
<BODY onload!#$%&( )*~+-_.,:;?@[/*]^`=alert()>
```

منابع پیشنهادی برای مطالعه بیشتر:

[zetc.de gist](#) •

[iSec Blog](#) •

[SecJuice #۱](#) , [SecJuice #۲](#) •



ابزارهای عالی برای رد شدن از WAF

ابزارهای شناسایی و تحلیل WAF

◆ WAFW00F

- توسعه‌دهنده : EnableSecurity
- کاربرد : ابزار پیشرفته برای شناسایی نوع WAF با بزرگترین دیتابیس اثرانگشت (Fingerprint).
- ویژگی : قابلیت شناسایی انواع WAF‌های رایج به صورت دقیق.

◆ IdentYwaf

- توسعه‌دهنده : stamparm
- کاربرد : شناسایی کور WAF با استفاده از اثرانگشت‌های از پیش جمع‌آوری شده.
- ویژگی خاص : بدون نیاز به پاسخ مستقیم از فایروال.

◆ WhatWaf

- توسعه‌دهنده : Ekultek
- کاربرد : شناسایی نوع WAF پشت وب‌سایت.
- ویژگی : پشتیبانی از بسیاری از WAF‌های تجاری و متن‌باز.



WAF ابزارهای دور زدن

◆ WAFNinja

• توسعه‌دهنده : khalilbijjou

• کاربرد : تولید خودکار Payload های فاز شده و پیشنهاد روش‌های Bypass

• زبان : Python

◆ WAFTester

• توسعه‌دهنده : Raz0r

• کاربرد : تست فایروال با استفاده از Payload های رمزگذاری شده برای عبور از فیلتر.

◆ SQLMap Tamper Scripts

• توسعه‌دهنده : SQLMap Project

• کاربرد : اسکریپت‌هایی برای تغییر شکل SQL های Payload جهت دور زدن WAF

• کاربردی در : حملات SQLi اتوماتیک با دور زدن فایروال.

◆ Bypass WAF BurpSuite Plugin

• پلاگین Burp Suite

• کاربرد : اضافه کردن هدرهایی مانند X-Forwarded-For برای وامود کردن

WAF درخواست از شبکه داخلی و فریب



WAF ابزارهای تست و بنچمارک

◆ GoTestWAF

• توسعه‌دهنده : Wallarm

• کاربرد : ارزیابی میزان پوشش و دقیقت فایروال‌ها با مجموعه‌های از تست‌های استاندارد.

◆ Lightbulb Framework

• زبان : Python

• کاربرد : چارچوب تست WAF با قابلیت fuzz کردن درخواست‌ها.

◆ WAFBench

• توسعه‌دهنده : Microsoft

• کاربرد : تست عملکرد و قدرت WAF در شرایط مختلف.

◆ FTW – Framework for Testing WAFs

• توسعه‌دهنده : OWASP Core Rule Set Team

• کاربرد : تست قوانین WAF بر اساس CRS v3 به صورت ساخت‌یافته و خودکار.

◆ WAF Testing Framework (Imperva)

• کاربرد : تست فایروال‌های Imperva با ابزار رسمی شرکت.



ابزارهای پیدا کردن IP واقعی پشت WAF

◆ CloudFail

- کاربرد : پیدا کردن آدرس IP واقعی پشت سرویس Cloudflare ویژگی : تحلیل DNS ، اسکن دامنه‌ها و بررسی زیرساخت‌ها.

◆ BypassWAF via DNS History

- کاربرد : جستجوی رکوردهای قدیمی DNS برای شناسایی IP واقعی سرور.
- مناسب برای : دور زدن WAF های Incapsula ، Cloudflare و غیره.

◆ abuse-ssl-bypass-waf

- کاربرد : بررسی cipher های SSL/TLS برای بهره‌برداری در دور زدن WAF

ابزارهای جامع امنیتی با قابلیت WAF Bypass

◆ w3af

- کاربرد : فریمورک تست نفوذ کامل برای اپلیکیشن‌های وب.
- قابلیت‌ها : اسکن آسیب‌پذیری، تست نفوذ، بازپس WAF و مدیریت نشست.

◆ libinjection-fuzzer

- کاربرد : تست و فاز کردن فیلترهای libinjection جهت یافتن مسیرهای دور زدن.

