Cloudflare

Associate Solutions Engineer Take Home Test

Technical Requirements Part 1:

To view the application, please visit:

racherla.info
racherla.info/headers (to view request headers)

Implementation Summary:

The first thing I did was set up a hosted server using Render, deploying the HTTPBin application to simulate an endpoint that reflects HTTP headers. This met the requirement for an origin server capable of responding with headers.

I then forked the HTTPBin GitHub repository and linked it to my Render sever. After resolving minor issues with the deployment process (including auto deploy tweaks and minor code updates), the server went live.

The next step was to then purchase a domain (racherla.info) through Namecheap. Next, I created a Cloudflare account and added the domain. After updating the nameservers in Namecheap to point to my Cloudflare account, I set up CNAME records in Cloudflare's DNS to route traffic to my Render server.

Once DNS propagation was completed, I added racherla.info and www.racherla.info as custom domains in Render, linking them to the deployed app. After verification, HTTPS TLS certificates were issued automatically by Render, completing the TLS 1.2+ requirement.

Using Cloudflare's Web Application Firewall (WAF), I configured a country based challenge rule. Requests from the US and India receive an interactive CAPTCHA challenge, while any other requests from other countries are blocked. These rules were tested successfully from different IPs.

This assignment was a completely new experience for me. I learned the end to end process of deploying a web service, linking a custom domain, configuring DNS, securing traffic with TLS and setting up WAF protection. I also gained hands on experience with Render, Cloudflare and HTTPBin. More importantly, I now understand how these services can be used to build and secure public web applications.

How I Filled Knowledge Gaps:

I had very little background in domain linking, DNS record types like CNAME or A or WAF rule configurations. To fill in the gaps in my knowledge, I used official Cloudflare & Render documentation, community forums (Reddit, Stack Overflow and Quora), Youtube tutorials and technical blogs and walkthroughs. When I ran into errors, I figured them out by checking the logs, reading the error messages, researching online and using a trial and error approach till I could find an appropriate solution.

Issues I Faced & How I Solved Them:

- Render App Failing to Deploy: I encountered multiple errors when linking my forked HTTPBin repo to Render. These were resolved by editing requirements.txt and adjusting Render's settings.
- DNS Verification Failing in Render: Despite proper configuration, domain verification stalled. I resolved this by deleting and adding the custom domain.
- WAF Not Triggering Initially: After setting rules, no CAPTCHA appeared. This was due to proxy status being set to "DNS only." Enabling Cloudflare proxy fixed it.

Explaining Cloudflare Products in Simple Language:

Imagine your website is like an office building. Render is the office inside where all the work happens. Namecheap is like your address sign that tells people where to find you. Cloudflare is the security system that protects your building and helps people get in safely. It handles DNS (like a phonebook) routing which sends visitors to the right place. TLS encryption (like talking in a secret language) keeps the conversations private. The WAF works like the front desk/security, only letting in visitors you trust. And Cloudflare Tunnels are like secret underground paths that connect straight to your office. All of this keeps your site safe and running smoothly.

The idea of this analogy is from a Youtube video that I saw while researching.

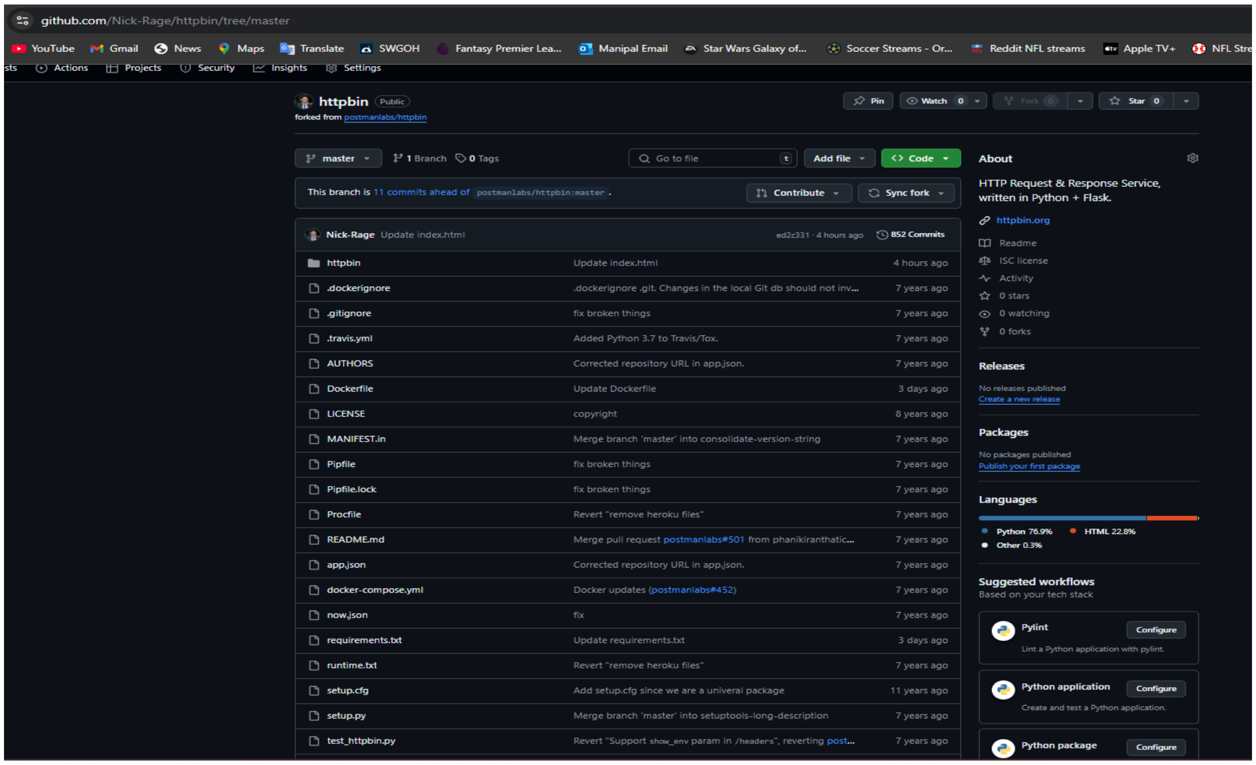
Would HTTP Response Headers Be Different Without Cloudflare?

Yes. Without Cloudflare the headers would only include data from the origin server (Render). With Cloudflare as a proxy, additional headers like Cf-Ray, Cf-Visitor, and Cf-Ipcountry appear. These show Cloudflare's presence and routing.

How Would a Customer Experience This?

If a customer followed a streamlined guide or had prior knowledge of web hosting and DNS, they would find the process smooth and easy. For beginners, especially with a non-technical background, the number of moving parts (Render, GitHub, DNS, WAF, TLS, tunnels) may be overwhelming, but could be managed relatively quickly with good documentation or guidance.

Screenshots Showcasing Work:



github.com/Nick-Rage/httpbin/tree/master

httpbin Public
forked from postmanlabs/httpbin

This branch is 11 commits ahead of `postmanlabs/httpbin:master`.

Nick-Rage Update index.html ed2c331 · 4 hours ago 852 Commits

httpbin	Update index.html	4 hours ago
.dockerignore	.dockerignore .git. Changes in the local Git db should not inv...	7 years ago
.gitignore	fix broken things	7 years ago
.travis.yml	Added Python 3.7 to Travis/Tox.	7 years ago
AUTHORS	Corrected repository URL in app.json.	7 years ago
Dockerfile	Update Dockerfile	3 days ago
LICENSE	copyright	8 years ago
MANIFEST.in	Merge branch 'master' into consolidate-version-string	7 years ago
Pipfile	fix broken things	7 years ago
Pipfile.lock	fix broken things	7 years ago
Procfile	Revert "remove heroku files"	7 years ago
README.md	Merge pull request postmanlabs#501 from phanikiranthatic...	7 years ago
app.json	Corrected repository URL in app.json.	7 years ago
docker-compose.yml	Docker updates (postmanlabs#452)	7 years ago
now.json	fix	7 years ago
requirements.txt	Update requirements.txt	3 days ago
runtime.txt	Revert "remove heroku files"	7 years ago
setup.cfg	Add setup.cfg since we are a univeral package	11 years ago
setup.py	Merge branch 'master' into setuptools-long-description	7 years ago
test_httpbin.py	Revert "Support show_env param in /headers", reverting post...	7 years ago

About

HTTP Request & Response Service, written in Python + Flask.

httpbin.org

Readme
ISC license
Activity
0 stars
0 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

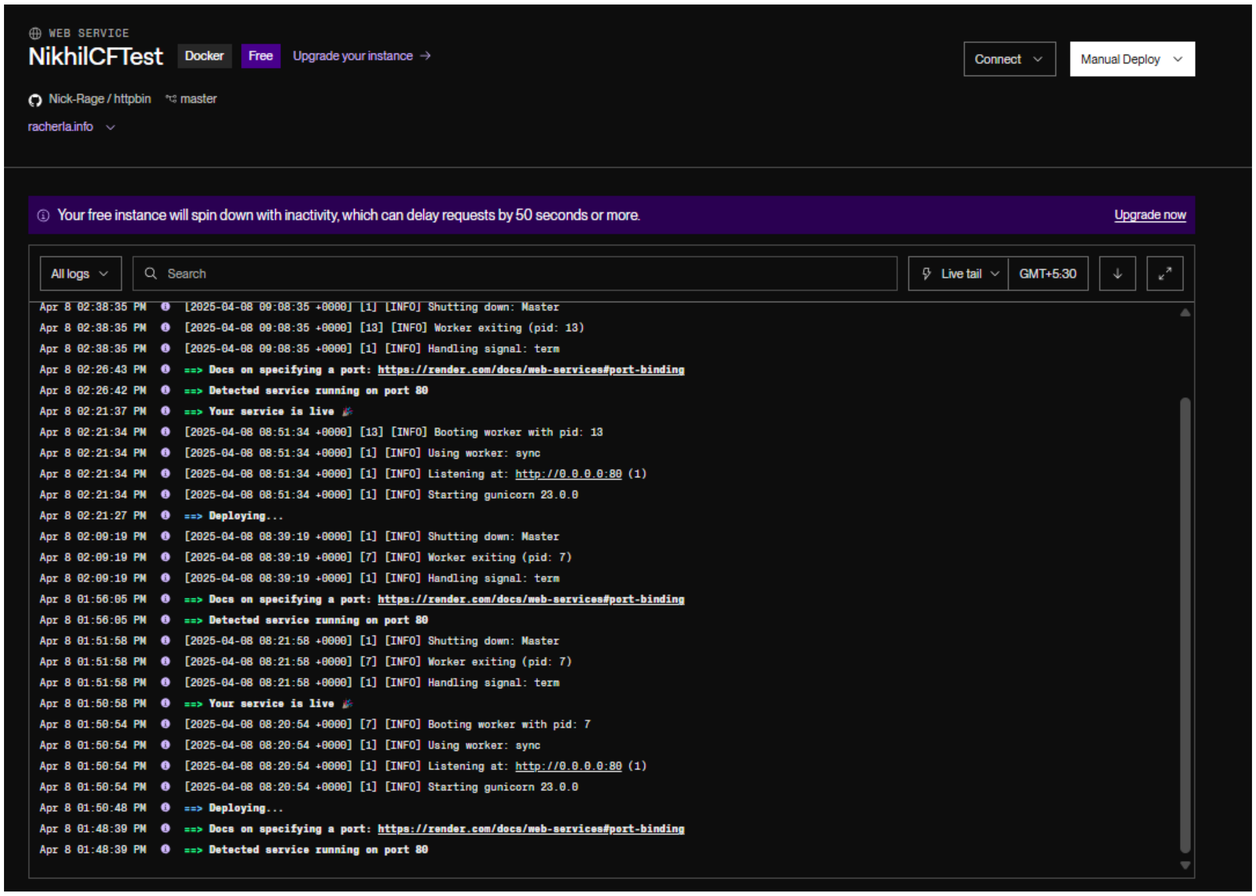
● Python 76.9% ● HTML 22.8%
● Other 0.3%

Suggested workflows
Based on your tech stack

Pylint — Configure
Lint a Python application with pylint.

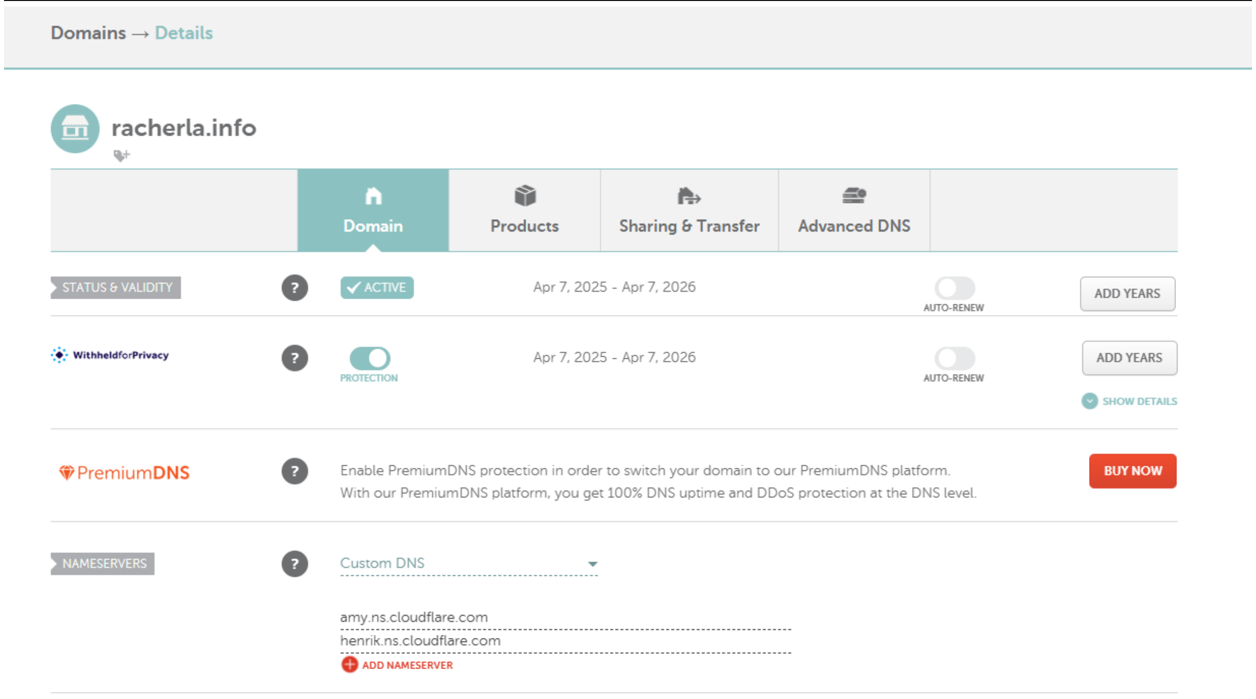
Python application — Configure
Create and test a Python application.

Python package — Configure

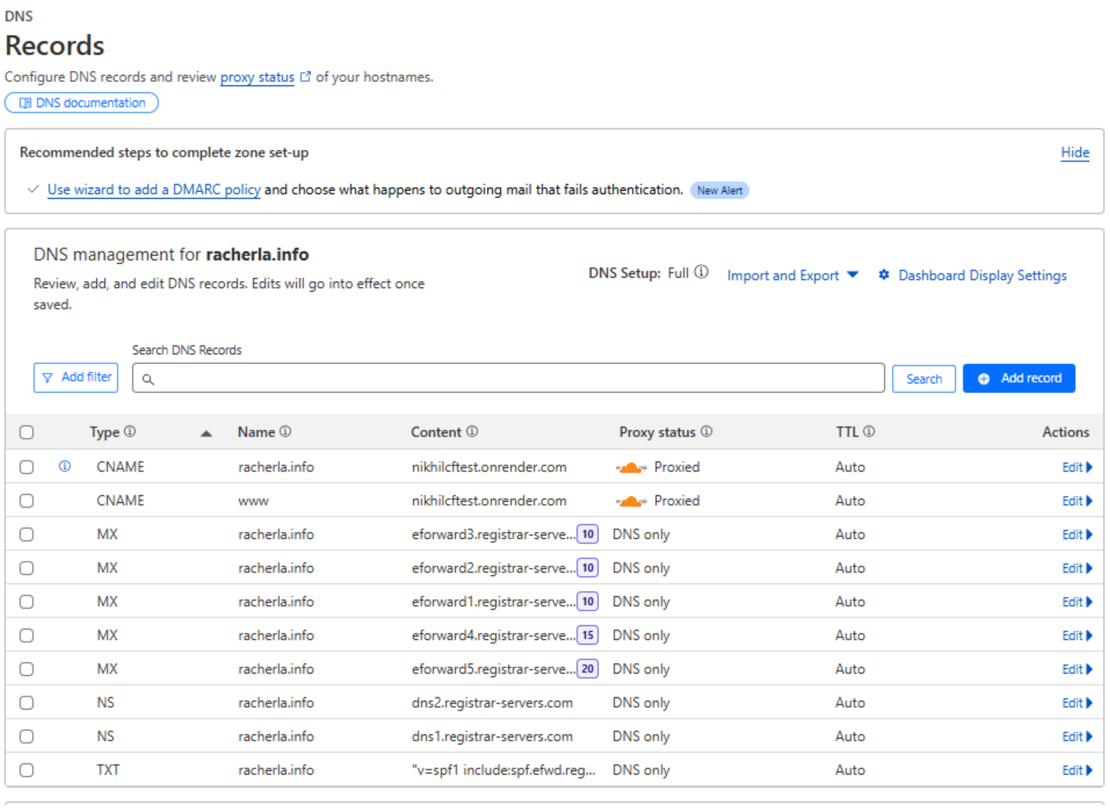


Your free instance will spin down with inactivity, which can delay requests by 50 seconds or more.

```
Apr 8 02:38:35 PM  [2025-04-08 09:08:35 +0000] [1] [INFO] Shutting down: Master
Apr 8 02:38:35 PM  [2025-04-08 09:08:35 +0000] [13] [INFO] Worker exiting (pid: 13)
Apr 8 02:38:35 PM  [2025-04-08 09:08:35 +0000] [1] [INFO] Handling signal: term
Apr 8 02:26:43 PM  ==> Docs on specifying a port: https://render.com/docs/web-services#port-binding
Apr 8 02:26:42 PM  ==> Detected service running on port 80
Apr 8 02:21:37 PM  ==> Your service is live 🎉
Apr 8 02:21:34 PM  [2025-04-08 08:51:34 +0000] [13] [INFO] Booting worker with pid: 13
Apr 8 02:21:34 PM  [2025-04-08 08:51:34 +0000] [1] [INFO] Using worker: sync
Apr 8 02:21:34 PM  [2025-04-08 08:51:34 +0000] [1] [INFO] Listening at: http://0.0.0.0:80 (1)
Apr 8 02:21:34 PM  [2025-04-08 08:51:34 +0000] [1] [INFO] Starting gunicorn 23.0.0
Apr 8 02:21:27 PM  ==> Deploying...
Apr 8 02:09:19 PM  [2025-04-08 08:39:19 +0000] [1] [INFO] Shutting down: Master
Apr 8 02:09:19 PM  [2025-04-08 08:39:19 +0000] [7] [INFO] Worker exiting (pid: 7)
Apr 8 02:09:19 PM  [2025-04-08 08:39:19 +0000] [1] [INFO] Handling signal: term
Apr 8 01:56:05 PM  ==> Docs on specifying a port: https://render.com/docs/web-services#port-binding
Apr 8 01:56:05 PM  ==> Detected service running on port 80
Apr 8 01:51:58 PM  [2025-04-08 08:21:58 +0000] [1] [INFO] Shutting down: Master
Apr 8 01:51:58 PM  [2025-04-08 08:21:58 +0000] [7] [INFO] Worker exiting (pid: 7)
Apr 8 01:51:58 PM  [2025-04-08 08:21:58 +0000] [1] [INFO] Handling signal: term
Apr 8 01:50:58 PM  ==> Your service is live 🎉
Apr 8 01:50:54 PM  [2025-04-08 08:20:54 +0000] [7] [INFO] Booting worker with pid: 7
Apr 8 01:50:54 PM  [2025-04-08 08:20:54 +0000] [1] [INFO] Using worker: sync
Apr 8 01:50:54 PM  [2025-04-08 08:20:54 +0000] [1] [INFO] Listening at: http://0.0.0.0:80 (1)
Apr 8 01:50:54 PM  [2025-04-08 08:20:54 +0000] [1] [INFO] Starting gunicorn 23.0.0
Apr 8 01:50:48 PM  ==> Deploying...
Apr 8 01:48:39 PM  ==> Docs on specifying a port: https://render.com/docs/web-services#port-binding
Apr 8 01:48:39 PM  ==> Detected service running on port 80
```



STATUS & VALIDITY	✓ ACTIVE	Apr 7, 2025 - Apr 7, 2026
WithheldforPrivacy	PROTECTION	Apr 7, 2025 - Apr 7, 2026
PremiumDNS	Enable PremiumDNS protection in order to switch your domain to our PremiumDNS platform. With our PremiumDNS platform, you get 100% DNS uptime and DDoS protection at the DNS level.	
NAMESERVERS	Custom DNS	amy.ns.cloudflare.com henrik.ns.cloudflare.com



DNS

Records

Configure DNS records and review proxy status of your hostnames.

DNS documentation

Recommended steps to complete zone set-up

Hide

✓ Use wizard to add a DMARC policy and choose what happens to outgoing mail that fails authentication. New Alert

DNS management for **racherla.info**

Review, add, and edit DNS records. Edits will go into effect once saved.

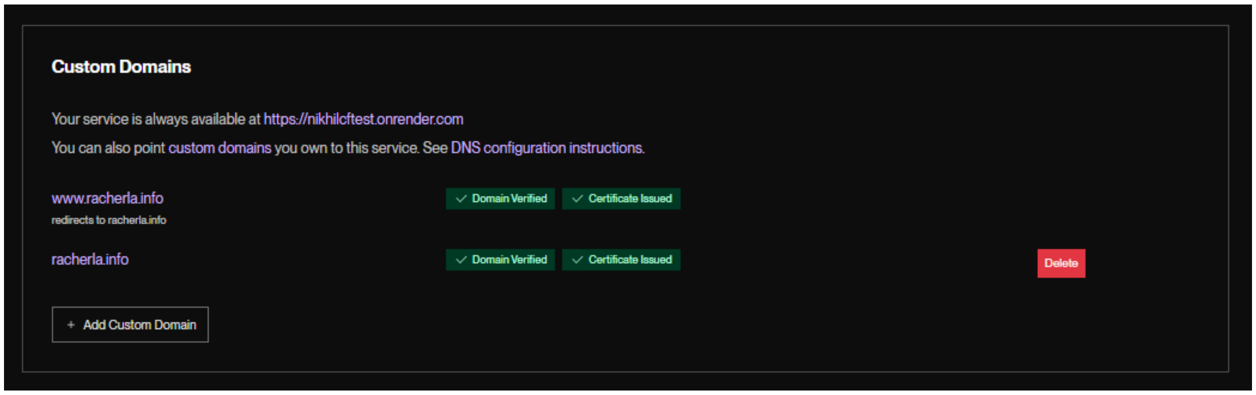
DNS Setup: Full

Import and Export

Dashboard Display Settings

Search DNS Records

Type	Name	Content	Proxy status	TTL	Actions
CNAME	racherla.info	nikhilcftest.onrender.com	Proxied	Auto	Edit
CNAME	www	nikhilcftest.onrender.com	Proxied	Auto	Edit
MX	racherla.info	eforward3.registrar-serve... 10	DNS only	Auto	Edit
MX	racherla.info	eforward2.registrar-serve... 10	DNS only	Auto	Edit
MX	racherla.info	eforward1.registrar-serve... 10	DNS only	Auto	Edit
MX	racherla.info	eforward4.registrar-serve... 15	DNS only	Auto	Edit
MX	racherla.info	eforward5.registrar-serve... 20	DNS only	Auto	Edit
NS	racherla.info	dns2.registrar-servers.com	DNS only	Auto	Edit
NS	racherla.info	dns1.registrar-servers.com	DNS only	Auto	Edit
TXT	racherla.info	"v=spf1 include:spf.efwd.reg...	DNS only	Auto	Edit



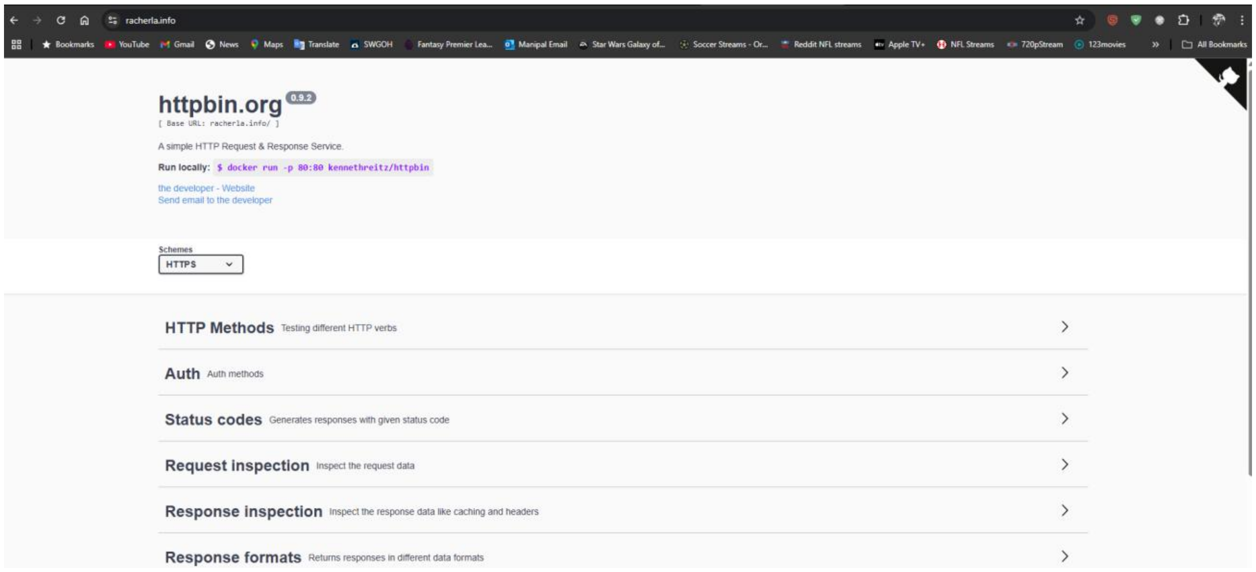
Custom Domains

Your service is always available at https://nikhilcftest.onrender.com

You can also point custom domains you own to this service. See DNS configuration instructions.

www.racherla.info redirects to racherla.info	✓ Domain Verified ✓ Certificate Issued	
racherla.info	✓ Domain Verified ✓ Certificate Issued	Delete

+ Add Custom Domain



httpbin.org 0.9.2

[Base URL: racherla.info/]

A simple HTTP Request & Response Service.

Run locally: `$ docker run -p 80:80 kennethreitz/httpbin`

the developer - Website
Send email to the developer

Schemes
HTTPS

HTTP Methods Testing different HTTP verbs

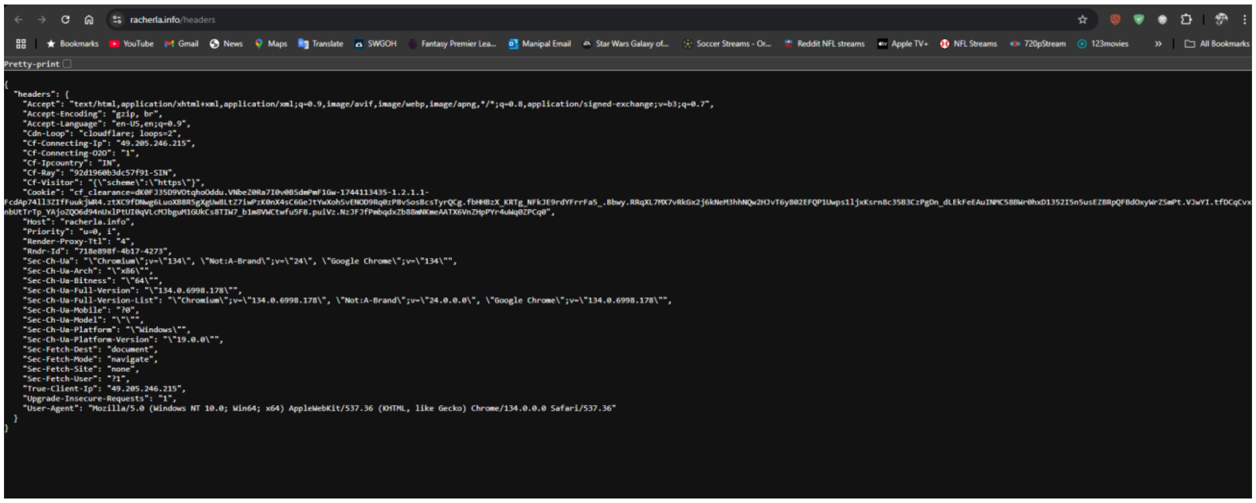
Auth Auth methods

Status codes Generates responses with given status code

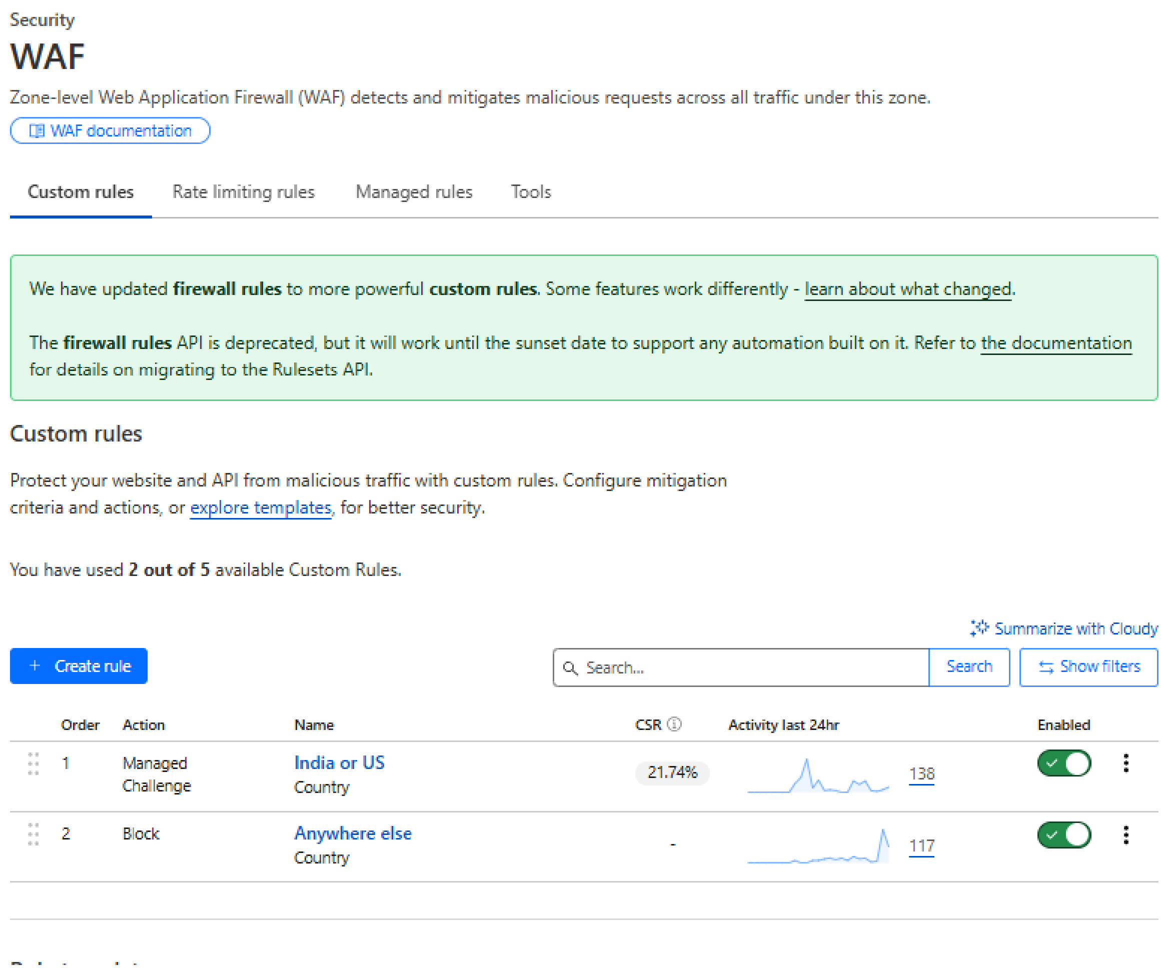
Request inspection Inspect the request data

Response inspection Inspect the response data like caching and headers

Response formats Returns responses in different data formats



```
{
  "headers": {
    "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7",
    "Accept-Encoding": "gzip, br",
    "Accept-Language": "en-US,en;q=0.9",
    "Cdn-Loop": "cloudflare; loops=2",
    "Cf-Connecting-Ip": "49.205.246.215",
    "Cf-Connecting-O2O": "1",
    "Cf-Ipcountry": "IN",
    "Cf-Ray": "92d1960b3dc57f91-SIN",
    "Cf-Visitor": "{\"scheme\":\"https\"}",
    "Cookie": "cf_clearance=dK0FJ35D9VOtqhoOddu.VNbeZ0Ra7I0v0BSdmPmF1Gw-1744113435-1.2.1.1-
FcdAp74ll3ZIfFuukjWR4.ztXC9fDNwg6LuoXB8R5gXgUw8LtZ7iwPzK0nX4sC6GeJtYwXohSvENOD9Rq0zP8vSosBcsTyrQCg.fbHHBzX_KRTg_NFkJE9rdYFrrFa5_.Bbwy.RRqXL7MX7vRkGx2j6kNeM3hhNQw2HJvT6y802EFQP1Uwps1ljxKsrn8c35B3CzPgDn_dLEkFeEAuINMC58BWr0hxD1352I5n5usEZBRpQFBdOxyWrZSmPt.VJwYI.tfDCqCvx
nbUtTrTp_YAjoZQO6d94nUxlPtUI0qVLcMJbguM1GUkCs8TIW7_b1m8VWCtwfu5F8.puiVz.NzJFJfPmbqdxZb88mNKmeAATX6VnZHpPYr4uWq0ZPCq0",
    "Host": "racherla.info",
    "Priority": "u=0, i",
    "Render-Proxy-Ttl": "4",
    "Rndr-Id": "718e898f-4b17-4273",
    "Sec-Ch-Ua": "\"Chromium\";v=\"134\", \"Not:A-Brand\";v=\"24\", \"Google Chrome\";v=\"134\"",
    "Sec-Ch-Ua-Arch": "\"x86\"",
    "Sec-Ch-Ua-Bitness": "\"64\"",
    "Sec-Ch-Ua-Full-Version": "\"134.0.6998.178\"",
    "Sec-Ch-Ua-Full-Version-List": "\"Chromium\";v=\"134.0.6998.178\", \"Not:A-Brand\";v=\"24.0.0.0\", \"Google Chrome\";v=\"134.0.6998.178\"",
    "Sec-Ch-Ua-Mobile": "?0",
    "Sec-Ch-Ua-Model": "\"\"",
    "Sec-Ch-Ua-Platform": "\"Windows\"",
    "Sec-Ch-Ua-Platform-Version": "\"19.0.0\"",
    "Sec-Fetch-Dest": "document",
    "Sec-Fetch-Mode": "navigate",
    "Sec-Fetch-Site": "none",
    "Sec-Fetch-User": "?1",
    "True-Client-Ip": "49.205.246.215",
    "Upgrade-Insecure-Requests": "1",
    "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36"
  }
}
```



Security

WAF

Zone-level Web Application Firewall (WAF) detects and mitigates malicious requests across all traffic under this zone.

We have updated **firewall rules** to more powerful **custom rules**. Some features work differently - learn about what changed.

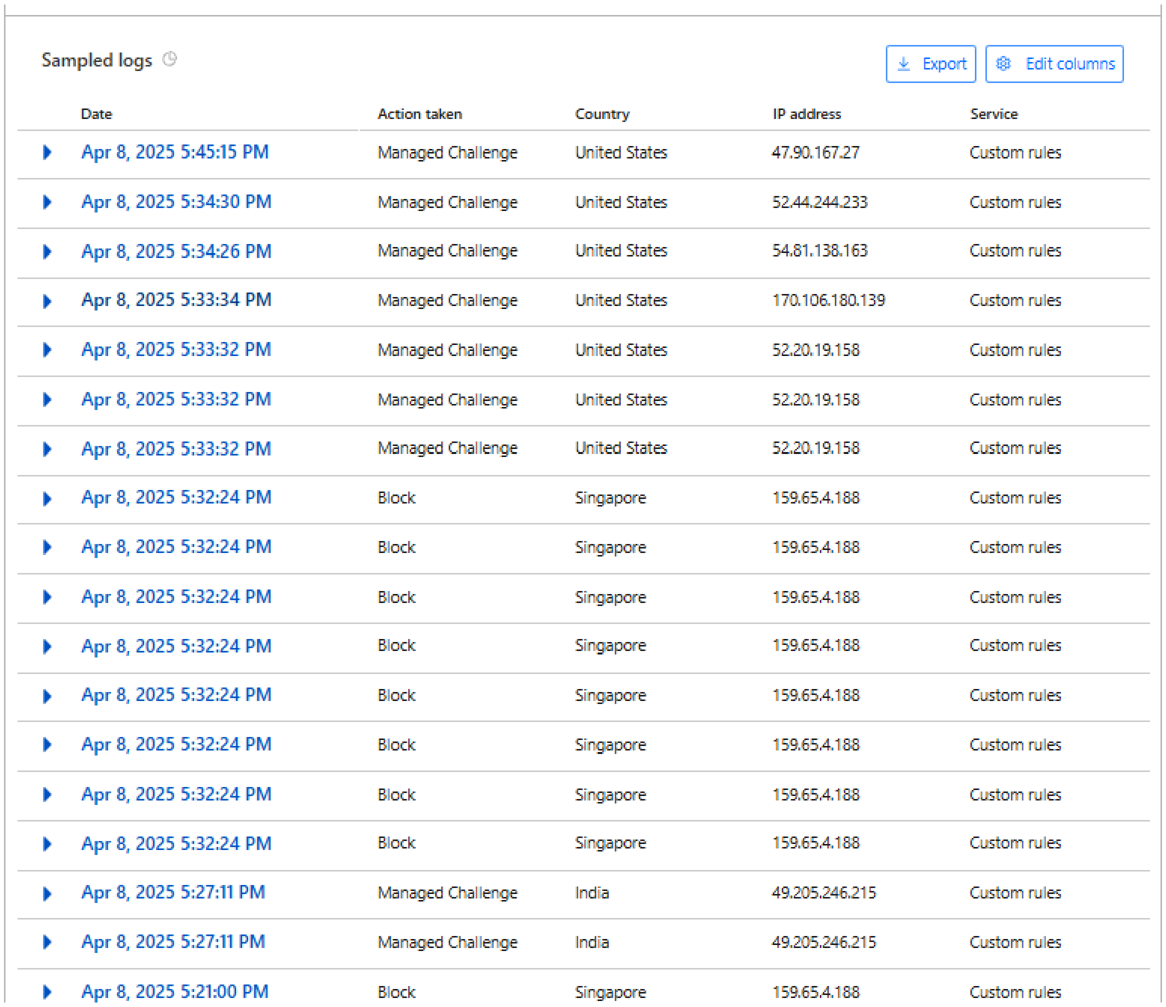
The **firewall rules** API is deprecated, but it will work until the sunset date to support any automation built on it. Refer to the documentation for details on migrating to the Rulesets API.

Custom rules

Protect your website and API from malicious traffic with custom rules. Configure mitigation criteria and actions, or explore templates, for better security.

You have used **2 out of 5** available Custom Rules.

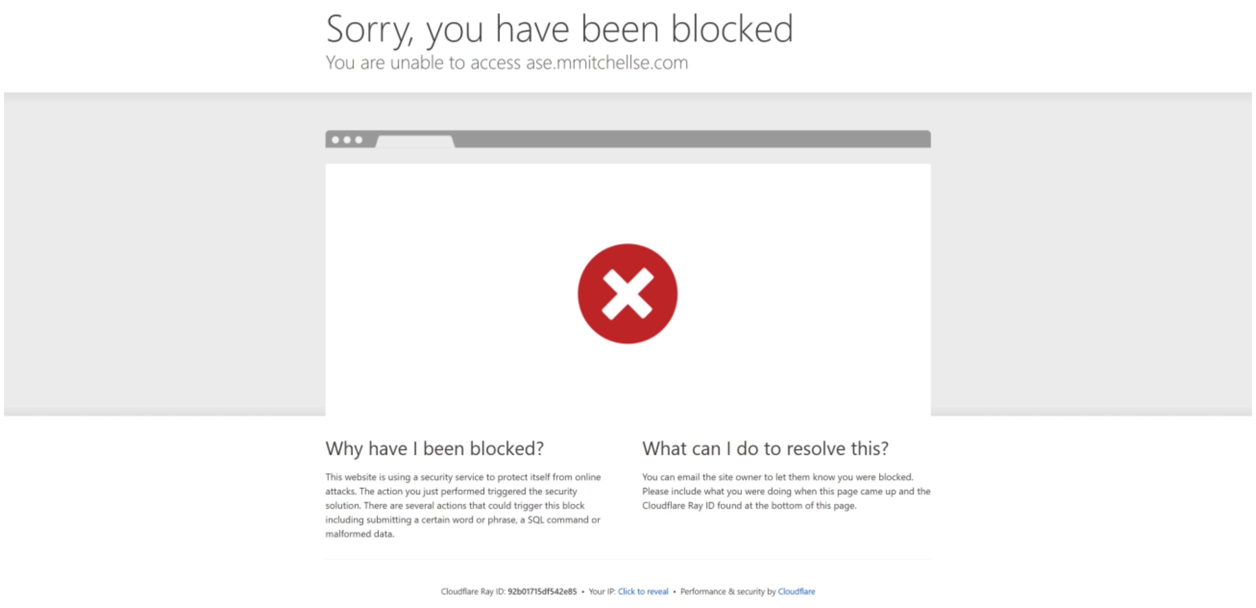
Order	Action	Name	CSR	Activity last 24hr	Enabled
1	Managed Challenge	India or US Country	21.74%	138	✓
2	Block	Anywhere else Country	-	117	✓



Date	Action taken	Country	IP address	Service
Apr 8, 2025 5:45:15 PM	Managed Challenge	United States	47.90.167.27	Custom rules
Apr 8, 2025 5:34:30 PM	Managed Challenge	United States	52.44.244.233	Custom rules
Apr 8, 2025 5:34:26 PM	Managed Challenge	United States	54.81.138.163	Custom rules
Apr 8, 2025 5:33:34 PM	Managed Challenge	United States	170.106.180.139	Custom rules
Apr 8, 2025 5:33:32 PM	Managed Challenge	United States	52.20.19.158	Custom rules
Apr 8, 2025 5:33:32 PM	Managed Challenge	United States	52.20.19.158	Custom rules
Apr 8, 2025 5:33:32 PM	Managed Challenge	United States	52.20.19.158	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:32:24 PM	Block	Singapore	159.65.4.188	Custom rules
Apr 8, 2025 5:27:11 PM	Managed Challenge	India	49.205.246.215	Custom rules
Apr 8, 2025 5:27:11 PM	Managed Challenge	India	49.205.246.215	Custom rules
Apr 8, 2025 5:21:00 PM	Block	Singapore	159.65.4.188	Custom rules

Technical Requirements Part 2:

Step 1: When I accessed https://ase.mmitchellse.com, I was presented with a Cloudflare "Sorry, you have been blocked" page. At the bottom of the page, I noticed a Ray ID, which I recognized as a unique request identifier that could be used for troubleshooting. This helped me begin my investigation using Cloudflare's dashboard.



Sorry, you have been blocked

You are unable to access ase.mmitchellse.com

Why have I been blocked?

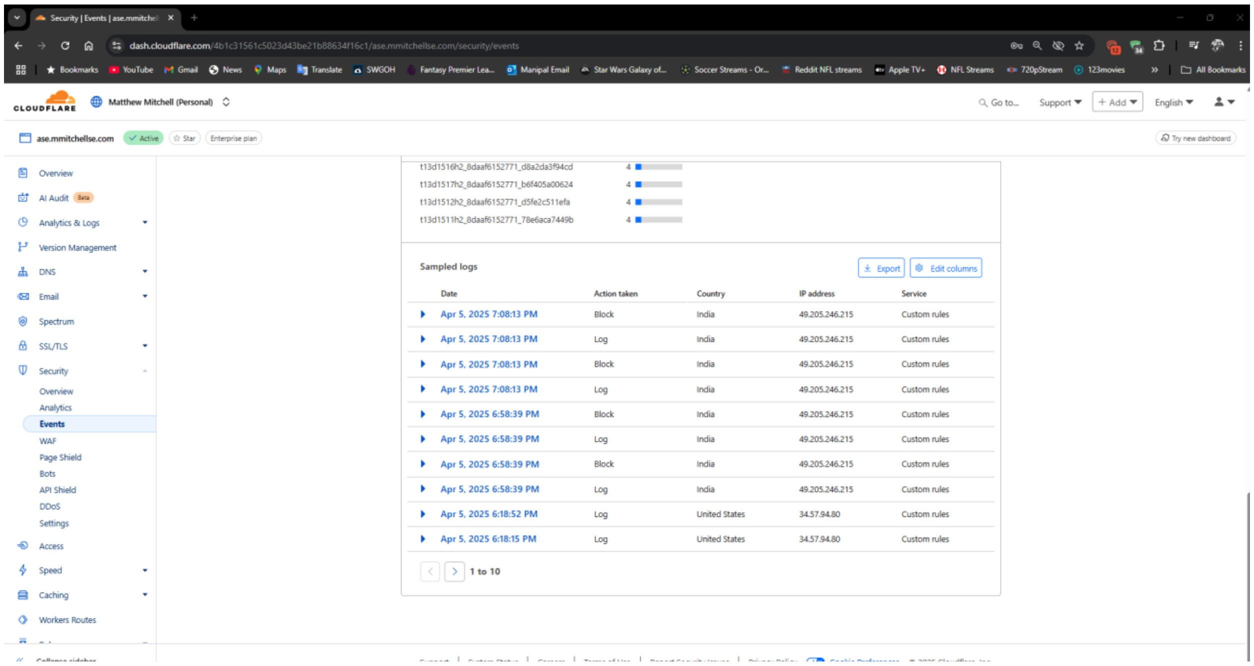
This website is using a security service to protect itself from online attacks. The action you just performed triggered the security solution. There are several actions that could trigger this block including submitting a certain word or phrase, a SQL command or malformed data.

What can I do to resolve this?

You can email the site owner to let them know you were blocked. Please include what you were doing when this page came up and the Cloudflare Ray ID found at the bottom of this page.

Cloudflare Ray ID: 92b01715df542e85 • Your IP: Click to reveal • Performance & security by Cloudflare

Step 2: When logging into the dashboard, I navigated the Events tab and filtered it by my IP address and the Ray ID. I discovered that my requests had been blocked multiple times, all triggered by a custom rule.

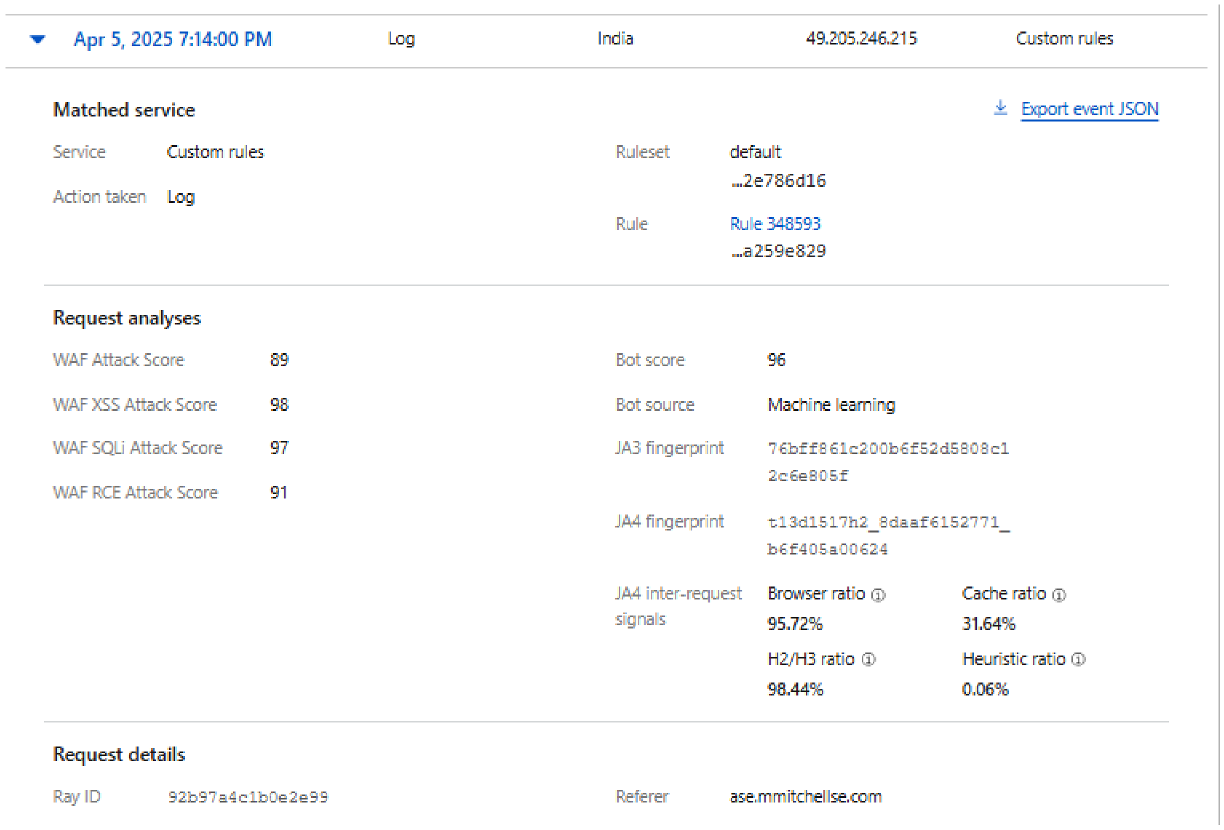


t13d1516h2_8daaf6152771_d8a2da3f94cd	4
t13d1517h2_8daaf6152771_b6f405a00624	4
t13d1512h2_8daaf6152771_d5fe2c511efa	4
t13d1511h2_8daaf6152771_78e6aca7449b	4

Sampled logs

Date	Action taken	Country	IP address	Service
Apr 5, 2025 7:08:13 PM	Block	India	49.205.246.215	Custom rules
Apr 5, 2025 7:08:13 PM	Log	India	49.205.246.215	Custom rules
Apr 5, 2025 7:08:13 PM	Block	India	49.205.246.215	Custom rules
Apr 5, 2025 7:08:13 PM	Log	India	49.205.246.215	Custom rules
Apr 5, 2025 6:58:39 PM	Block	India	49.205.246.215	Custom rules
Apr 5, 2025 6:58:39 PM	Log	India	49.205.246.215	Custom rules
Apr 5, 2025 6:58:39 PM	Block	India	49.205.246.215	Custom rules
Apr 5, 2025 6:58:39 PM	Log	India	49.205.246.215	Custom rules
Apr 5, 2025 6:18:52 PM	Log	United States	34.57.94.80	Custom rules
Apr 5, 2025 6:18:15 PM	Log	United States	34.57.94.80	Custom rules

1 to 10



Apr 5, 2025 7:14:00 PM	Log	India	49.205.246.215	Custom rules

Matched service

Service	Custom rules	Ruleset	default …2e786d16
Action taken	Log	Rule	Rule 348593 …a259e829

Request analyses

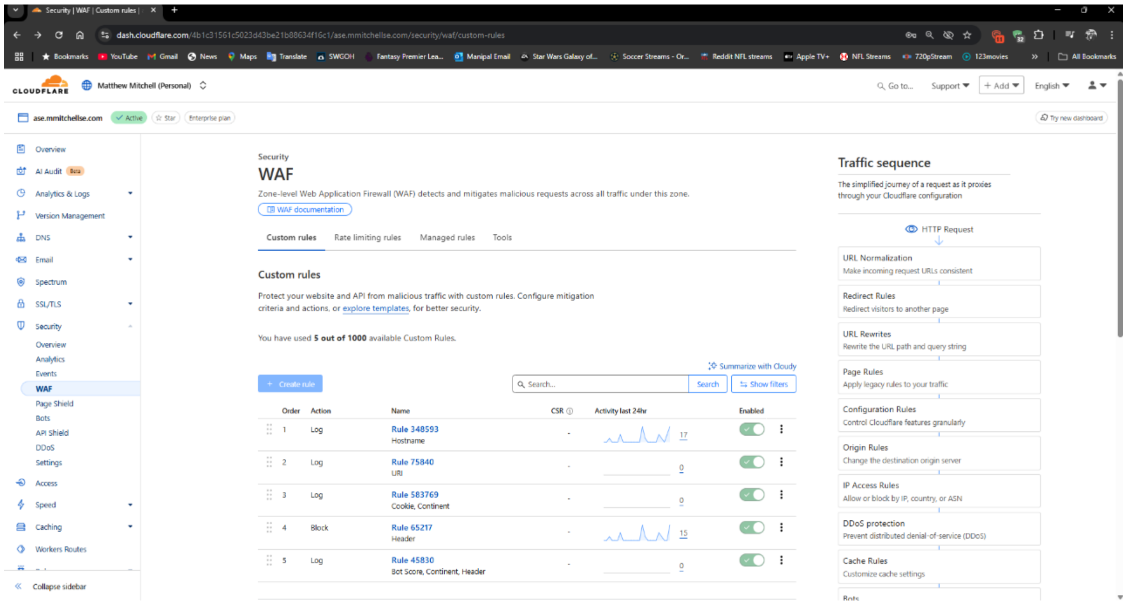
WAF Attack Score	89	Bot score	96
WAF XSS Attack Score	98	Bot source	Machine learning
WAF SQLi Attack Score	97	JA3 fingerprint	76bff861c200b6f52d5808c12c6e805f
WAF RCE Attack Score	91	JA4 fingerprint	t13d1517h2_8daaf6152771_b6f405a00624

JA4 inter-request signals: Browser ratio 95.72%; Cache ratio 31.64%; H2/H3 ratio 98.44%; Heuristic ratio 0.06%

Request details

Ray ID	92b97a4c1b0e2e99	Referer	ase.mmitchellse.com

Step 3: I then proceeded to the WAF section, where five rules were configured. Out of them four were set to log traffic, while one rule (Rule 65217) was set to block.



Security

WAF

Zone-level Web Application Firewall (WAF) detects and mitigates malicious requests across all traffic under this zone.

Custom rules

Protect your website and API from malicious traffic with custom rules. Configure mitigation criteria and actions, or explore templates, for better security.

You have used **5 out of 1000** available Custom Rules.

Order	Action	Name	CSR	Activity last 24hr	Enabled
1	Log	Rule 348593 Hostname	-	17	✓
2	Log	Rule 75840 URI	-	0	✓
3	Log	Rule 583769 Cookie, Continent	-	0	✓
4	Block	Rule 65217 Header	-	15	✓
5	Log	Rule 45830 Bot Score, Continent, Header	-	0	✓

Traffic sequence

The simplified journey of a request as it proxies through your Cloudflare configuration

HTTP Request

URL Normalization — Make incoming request URLs consistent

Redirect Rules — Redirect visitors to another page

URL Rewrites — Rewrite the URL path and query string

Page Rules — Apply legacy rules to your traffic

Configuration Rules — Control Cloudflare features granularly

Origin Rules — Change the destination origin server

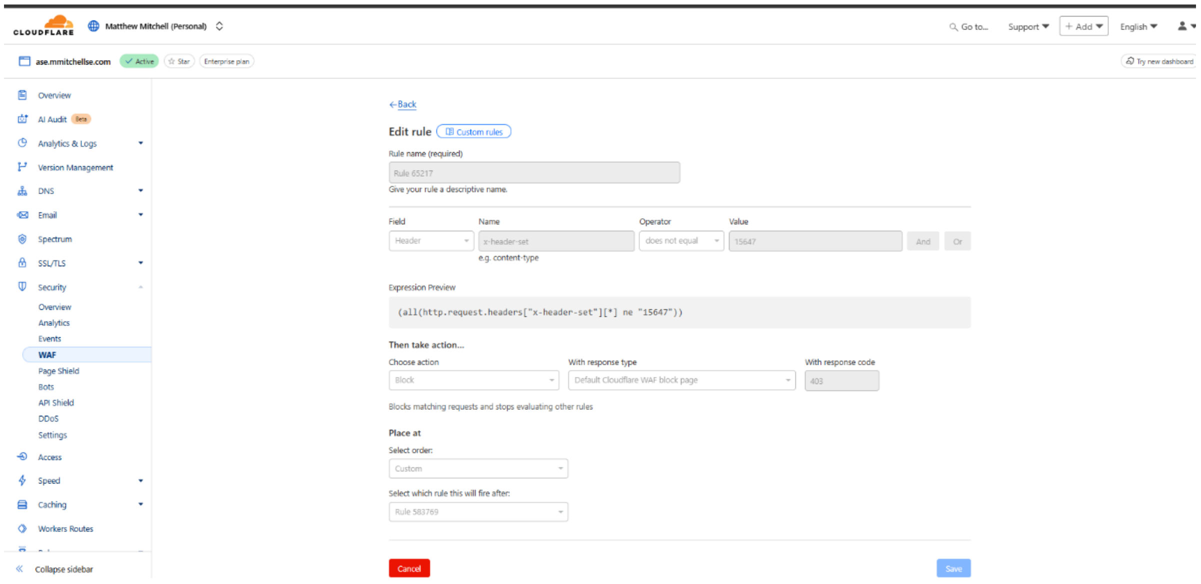
IP Access Rules — Allow or block by IP, country, or ASN

DDoS protection — Prevent distributed denial-of-service (DDoS)

Cache Rules — Customize cache settings

Step 4: Clicking into Rule 65217 revealed that it was blocking requests based on a custom HTTP header condition. Specifically, the expression logic required all requests to include the header x-header-set: 15647.

Any request that did not include this header was automatically blocked before reaching the origin server. Since I had initially accessed the site using a standard browser without that specific header, my request was denied, triggering the 403 error. This confirmed that the root cause of the block was the absence of this specific header.


Edit rule

Rule name (required): Rule 65217

Give your rule a descriptive name.

Field	Name	Operator	Value
Header	x-header-set	does not equal	15647

Expression Preview

```
(all(http.request.headers["x-header-set"][*] ne "15647"))
```

Then take action...

Choose action: Block

With response type: Default Cloudflare WAF block page

With response code: 403

Blocks matching requests and stops evaluating other rules

Place at

Select order: Custom

Select which rule this will fire after: Rule 583769

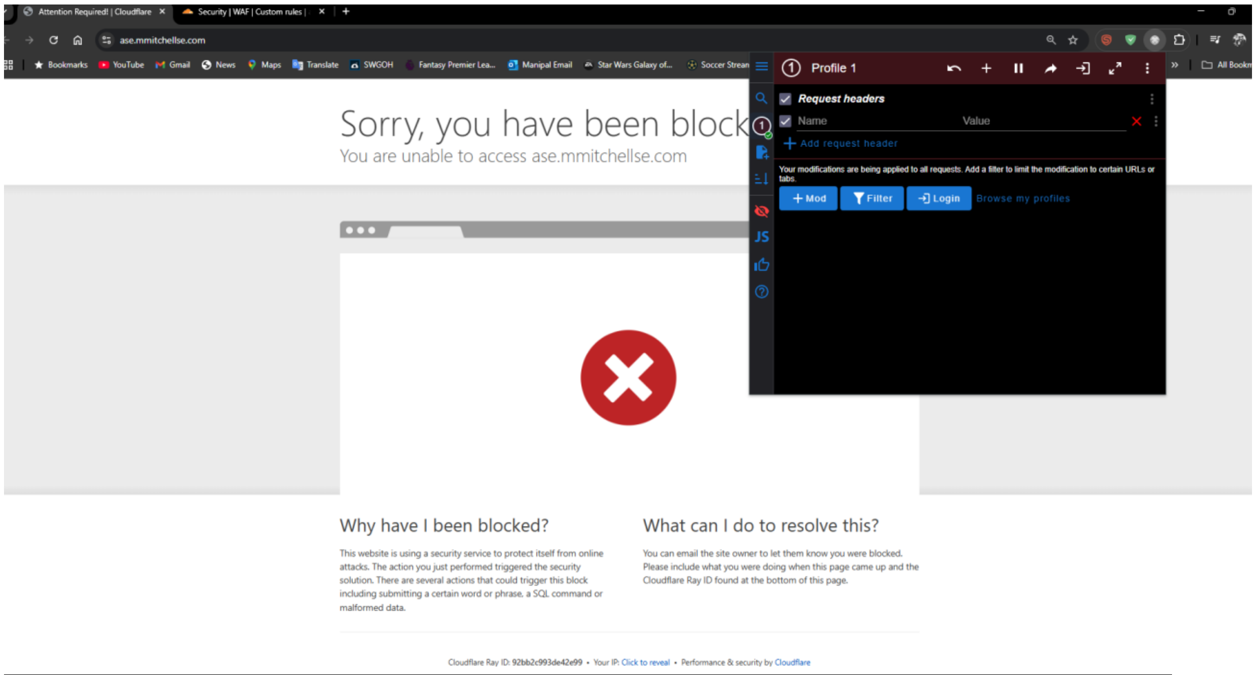
Bonus 1:

This type of security configuration is particularly useful in enterprise environments where lightweight access control is needed. For example, organizations could use similar rules to restrict access to staging environments, internal APIs or sensitive resources. It can serve as a quick authentication layer or a method to block automated scanning tools and bots. However, in production systems, this method should be used alongside other stronger identity based security controls such as tokens or Cloudflare Zero Trust policies.

Bonus 2:

To bypass the block and reach the Congrats Page, I tested two different methods. First, I used the ModHeader browser extension to inject the required header into my request. I added the header x-header-set with the value 15647, then reloaded the page in the browser. This successfully allowed me to view the Congrats Page.

As a second method, I also created a simple PowerShell script using Notepad. The script sent a request to the site with the correct header, saved the HTML response to a local file, and then automatically opened it in the default browser. This was especially useful as a privacy conscious environment and would be ideal where browser extensions are restricted. Both approaches demonstrated a successful bypass and confirmed the custom header requirement.



Sorry, you have been block

You are unable to access ase.mmitchellse.com

Request headers

Name Value

Your modifications are being applied to all requests. Add a filter to limit the modification to certain URLs or tabs.

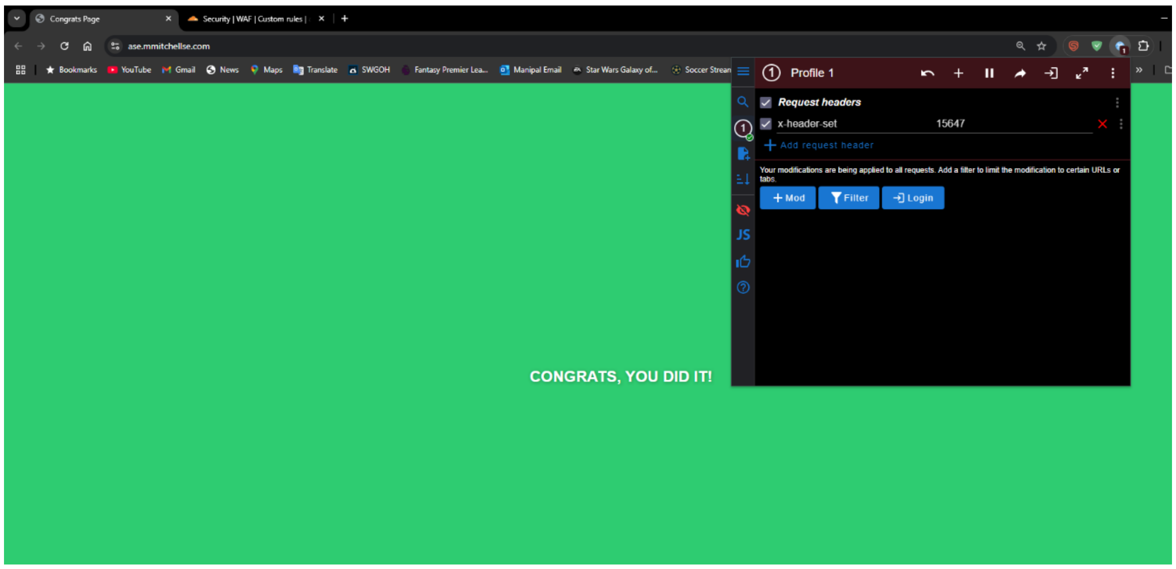
Why have I been blocked?

This website is using a security service to protect itself from online attacks. The action you just performed triggered the security solution. There are several actions that could trigger this block including submitting a certain word or phrase, a SQL command or malformed data.

What can I do to resolve this?

You can email the site owner to let them know you were blocked. Please include what you were doing when this page came up and the Cloudflare Ray ID found at the bottom of this page.

Cloudflare Ray ID: 92bb2c993de42e99 • Your IP: Click to reveal • Performance & security by Cloudflare



Request headers

x-header-set 15647

Your modifications are being applied to all requests. Add a filter to limit the modification to certain URLs or tabs.

CONGRATS, YOU DID IT!

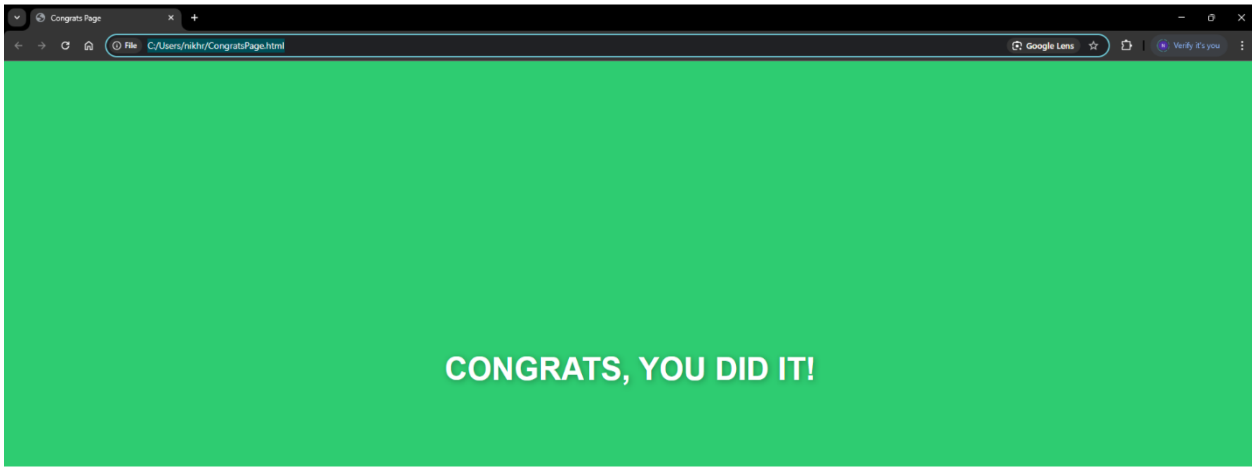
<u>Notepad Script:</u>

$response = Invoke-WebRequest -Uri "https://ase.mmitchellse.com" -Headers @{ "x-header-set" = "15647" }

$outputPath = "$HOME\CongratsPage.html"

$response.Content | Out-File -FilePath $outputPath -Encoding utf8

Start-Process $outputPath



Summary

In conclusion, this investigation gave me hands on experience with identifying WAF rule behavior, understanding custom header logic and simulating how such issues might impact real customers. I was able to not only replicate the issue but also resolve it using practical and privacy friendly approaches.