AWS Web Application Deployment with CloudFront and WAF - Walkthrough Summary

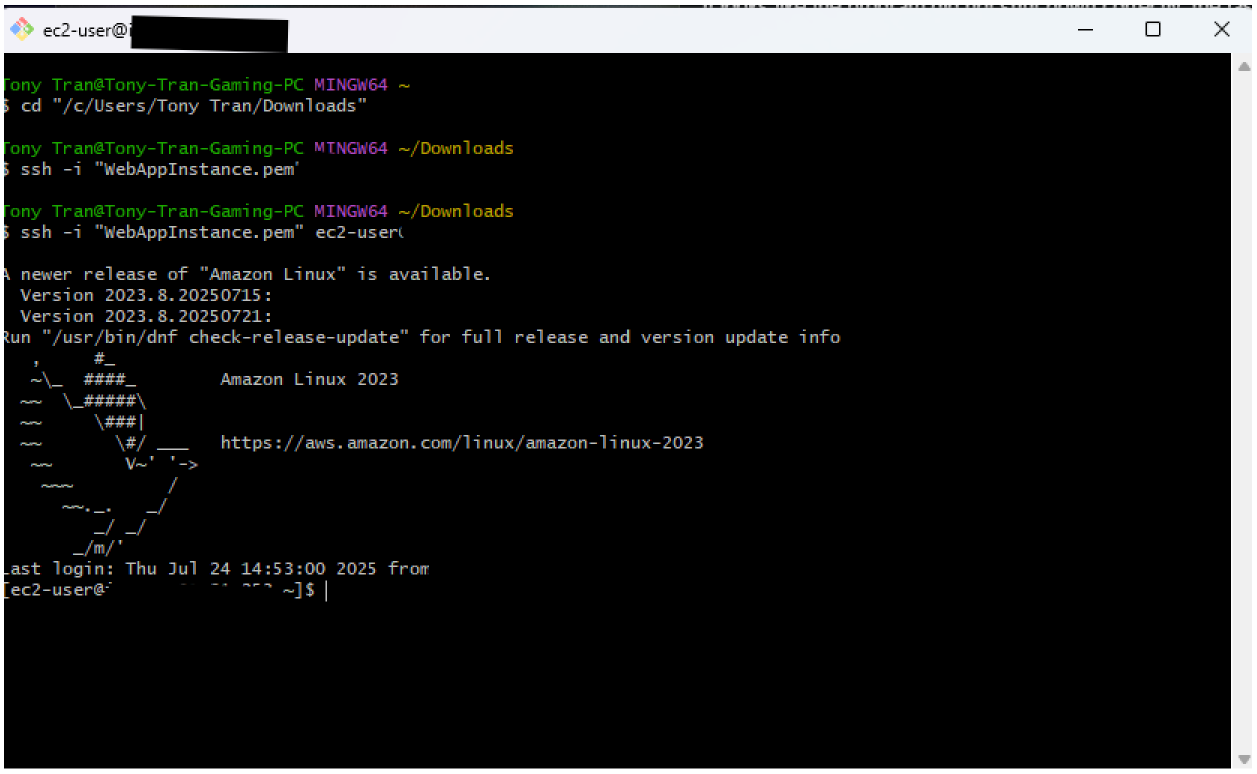
Author: Tony Tran
Project Goal: Deploy a secure web application using EC2, CloudFront, and AWS WAF with OWASP protection.

---

1. EC2 Setup

- **Launch EC2 instance** using Amazon Linux 2023.
- Open ports 22 (SSH) and 80 (HTTP) in the **Security Group**.
- Connect to the EC2 instance using `.pem` key with:

  ```
  ssh -i "WebAppInstance.pem" ec2-user@<EC2 Public DNS>
  ```



```
ec2-user@i[redacted]

Tony Tran@Tony-Tran-Gaming-PC MINGW64 ~
$ cd "/c/Users/Tony Tran/Downloads"

Tony Tran@Tony-Tran-Gaming-PC MINGW64 ~/Downloads
$ ssh -i "WebAppInstance.pem'

Tony Tran@Tony-Tran-Gaming-PC MINGW64 ~/Downloads
$ ssh -i "WebAppInstance.pem" ec2-user[redacted]

A newer release of "Amazon Linux" is available.
  Version 2023.8.20250715:
  Version 2023.8.20250721:
Run "/usr/bin/dnf check-release-update" for full release and version update info
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Thu Jul 24 14:53:00 2025 from [redacted]
[ec2-user@[redacted] ~]$ |
```

---

2. Install and Configure Web Server

- Install Apache web server:

  ```
  sudo yum update -y
  sudo yum install httpd -y
  ```

```
sudo systemctl start httpd
sudo systemctl enable httpd
```

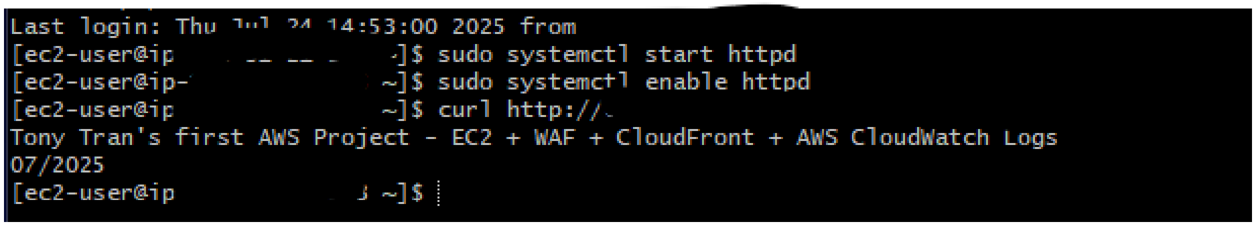
- Modify `index.html`:

```
echo "Hello from Tony's secure AWS web app!" | sudo tee /var/www/html/i
ndex.html
```

- Verify via EC2 public DNS:

```
curl http://<EC2 Public DNS>
```

Output:

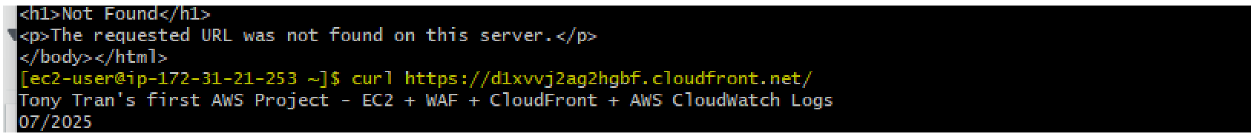


```
Last login: Thu Jul 24 14:53:00 2025 from [redacted]
[ec2-user@ip[redacted]]$ sudo systemctl start httpd
[ec2-user@ip-[redacted] ~]$ sudo systemctl enable httpd
[ec2-user@ip[redacted] ~]$ curl http://[redacted]
Tony Tran's first AWS Project - EC2 + WAF + CloudFront + AWS CloudWatch Logs
07/2025
[ec2-user@ip[redacted]3 ~]$
```

---

3. CloudFront Configuration

- Create a **CloudFront distribution**:
  - **Origin:** EC2 Public DNS (e.g., `ec2-34-228-71-196.compute-1.amazonaws.com`)
  - **Origin Protocol Policy:** HTTP only
  - **Viewer Protocol Policy:** Redirect HTTP to HTTPS
  - Enable **WAF security protections**
- Wait until **Status = Deployed**
- **CloudFront URL: https://d1xvvj2ag2hgbf.cloudfront.net/**



```
<h1>Not Found</h1>
<p>The requested URL was not found on this server.</p>
</body></html>
[ec2-user@ip-172-31-21-253 ~]$ curl https://d1xvvj2ag2hgbf.cloudfront.net/
Tony Tran's first AWS Project - EC2 + WAF + CloudFront + AWS CloudWatch Logs
07/2025
```

---

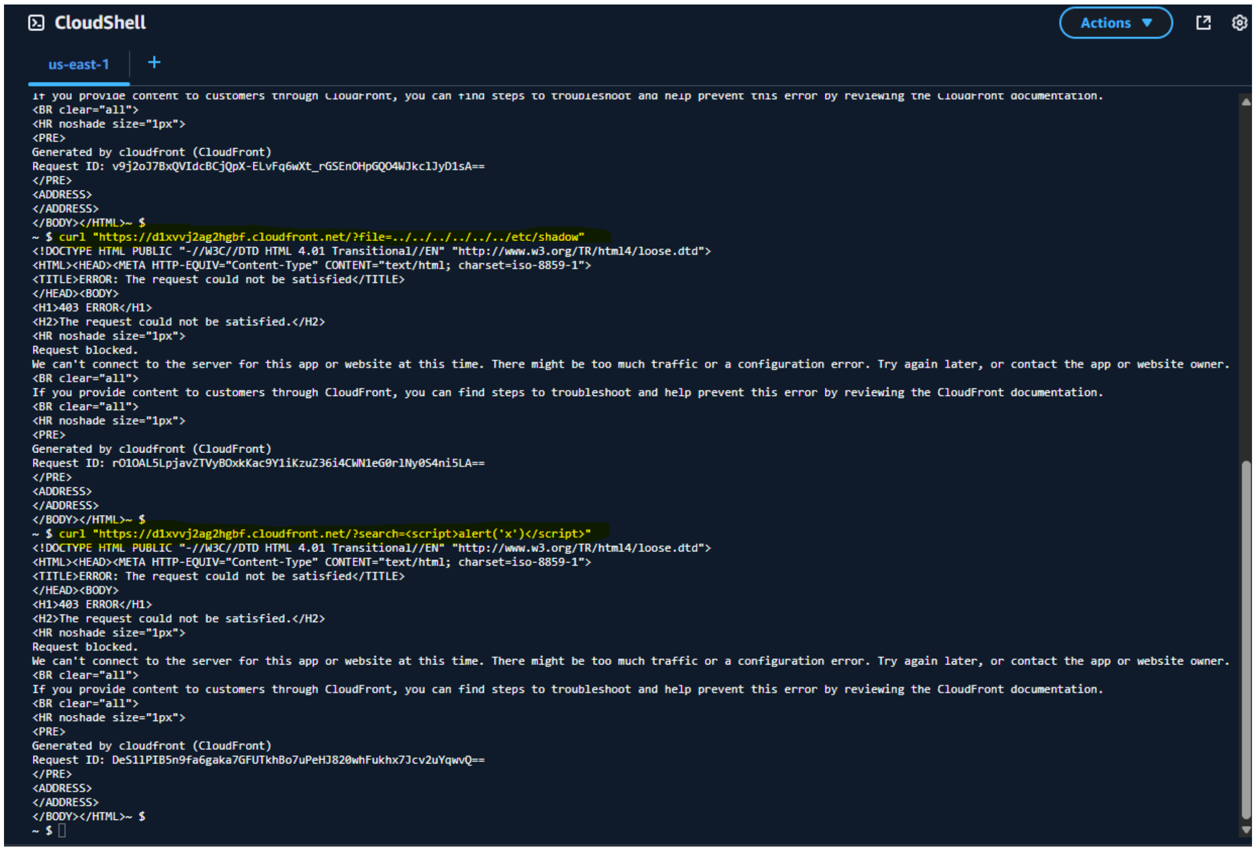
4. AWS WAF Setup

- Create a **Web ACL** in AWS WAF:
  - Scope: CloudFront
  - Associate with the CloudFront distribution
  - Add managed rule group: **AWSManagedRulesCommonRuleSet** (OWASP protection)

---

5. Test WAF Protection

- Test payloads via CloudFront URL:
- `# XSS Payload`
- `curl "https://d1xvvj2ag2hgbf.cloudfront.net/?search=<script>alert('x')</script>"`
- `# SQL Injection Payload`
- `curl "https://d1xvvj2ag2hgbf.cloudfront.net/?id=1' OR '1'='1"`
- `# Path Traversal Payload`
- `curl "https://d1xvvj2ag2hgbf.cloudfront.net/?search=../../etc/passwd"`
- `# Admin Enumeration Payload`
- `curl "https://d1xvvj2ag2hgbf.cloudfront.net/?admin=true"`
- `# Command Injection (for completeness)`
- `curl "https://d1xvvj2ag2hgbf.cloudfront.net/?cmd=ls%20-la"`
- `# Local File Inclusion (LFI) variant`
- `curl https://d1xvvj2ag2hgbf.cloudfront.net/?file=../../../../../../etc/shadow`
- All return:



```
If you provide content to customers through CloudFront, you can find steps to troubleshoot and help prevent this error by reviewing the CloudFront documentation.
<BR clear="all">
<HR noshade size="1px">
<PRE>
Generated by cloudfront (CloudFront)
Request ID: v9j2oJ7BxQVIdcBCjQpX-ELvFq6wXt_rGSEnOHpGQO4WJkclJyD1sA==
</PRE>
<ADDRESS>
</ADDRESS>
</BODY></HTML>~ $
~ $ curl "https://d1xvvj2ag2hgbf.cloudfront.net/?file=../../../../../../etc/shadow"
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<HTML><HEAD><META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<TITLE>ERROR: The request could not be satisfied</TITLE>
</HEAD><BODY>
<H1>403 ERROR</H1>
<H2>The request could not be satisfied.</H2>
<HR noshade size="1px">
Request blocked.
We can't connect to the server for this app or website at this time. There might be too much traffic or a configuration error. Try again later, or contact the app or website owner.
<BR clear="all">
If you provide content to customers through CloudFront, you can find steps to troubleshoot and help prevent this error by reviewing the CloudFront documentation.
<BR clear="all">
<HR noshade size="1px">
<PRE>
Generated by cloudfront (CloudFront)
Request ID: rO1OAL5LpjavZTVyBOxkKac9Y1iKzuZ36i4CWN1eG0rlNy0S4ni5LA==
</PRE>
<ADDRESS>
</ADDRESS>
</BODY></HTML>~ $
~ $ curl "https://d1xvvj2ag2hgbf.cloudfront.net/?search=<script>alert('x')</script>"
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<HTML><HEAD><META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<TITLE>ERROR: The request could not be satisfied</TITLE>
</HEAD><BODY>
<H1>403 ERROR</H1>
<H2>The request could not be satisfied.</H2>
<HR noshade size="1px">
Request blocked.
We can't connect to the server for this app or website at this time. There might be too much traffic or a configuration error. Try again later, or contact the app or website owner.
<BR clear="all">
If you provide content to customers through CloudFront, you can find steps to troubleshoot and help prevent this error by reviewing the CloudFront documentation.
<BR clear="all">
<HR noshade size="1px">
<PRE>
Generated by cloudfront (CloudFront)
Request ID: DeS11PIB5n9fa6gaka7GFUTkhBo7uPeHJ820whFukhx7Jcv2uYqwvQ==
</PRE>
<ADDRESS>
</ADDRESS>
</BODY></HTML>~ $
~ $
```

```
403 ERROR: The request could not be satisfied. Request blocked.
```

---

6. Troubleshooting Notes

- If CloudFront returns `504 Gateway Timeout`:
  - Confirm EC2 is running
  - Ensure EC2 allows traffic on port 80
  - Check CloudFront origin settings and wait for deployment

---

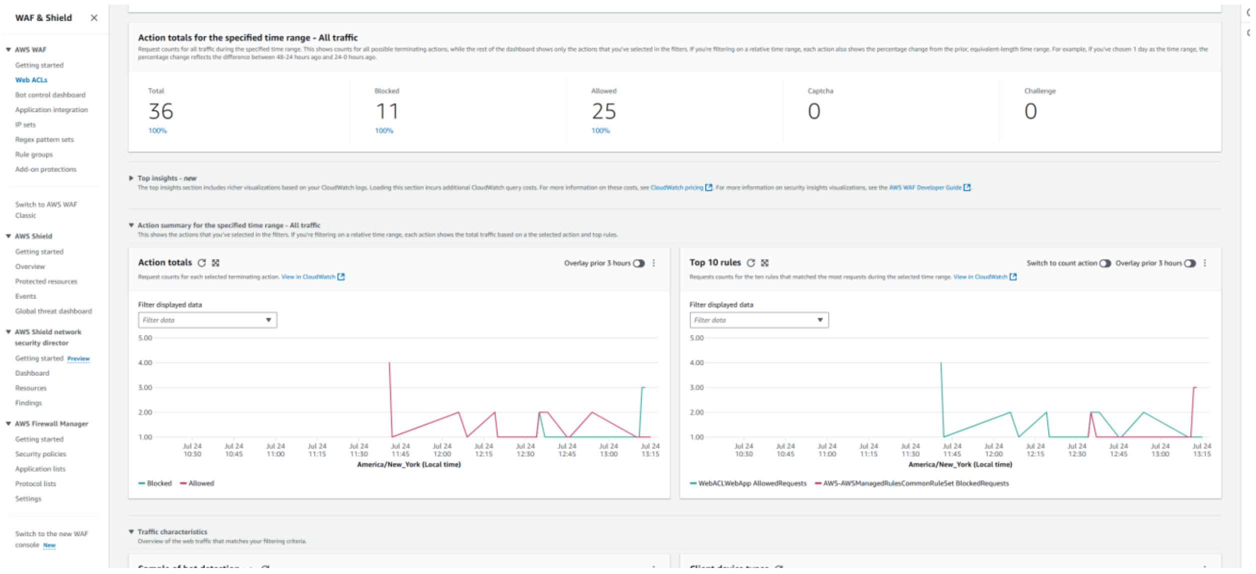
7. Key Services Used

- **EC2:** Hosts the web application
- **CloudFront:** CDN layer with performance and caching
- **AWS WAF:** Blocks malicious input using managed rules
- **Security Groups:** Controls inbound/outbound EC2 traffic

---

8. Outcome

- Successfully deployed a public web app with **CloudFront CDN** and **WAF protection**.
- Verified defense against **OWASP Top 10** payloads.



Action totals for the specified time range - All traffic

Request counts for all traffic during the specified time range. This shows counts for all possible terminating actions, while the rest of the dashboard shows only the actions that you've selected in the filters. If you're filtering on a relative time range, each action also shows the percentage change from the prior, equivalent-length time range. For example, if you've chosen 1 day as the time range, the percentage change reflects the difference between 48-24 hours ago and 24-0 hours ago.

Total	Blocked	Allowed	Captcha	Challenge
36	11	25	0	0
100%	100%	100%		

Top insights - *new*

The top insights section includes richer visualizations based on your CloudWatch logs. Loading this section incurs additional CloudWatch query costs. For more information on these costs, see CloudWatch pricing. For more information on security insights visualizations, see the AWS WAF Developer Guide.

Action summary for the specified time range - All traffic

This shows the actions that you've selected in the filters. If you're filtering on a relative time range, each action shows the total traffic based on a the selected action and top rules.

Action totals

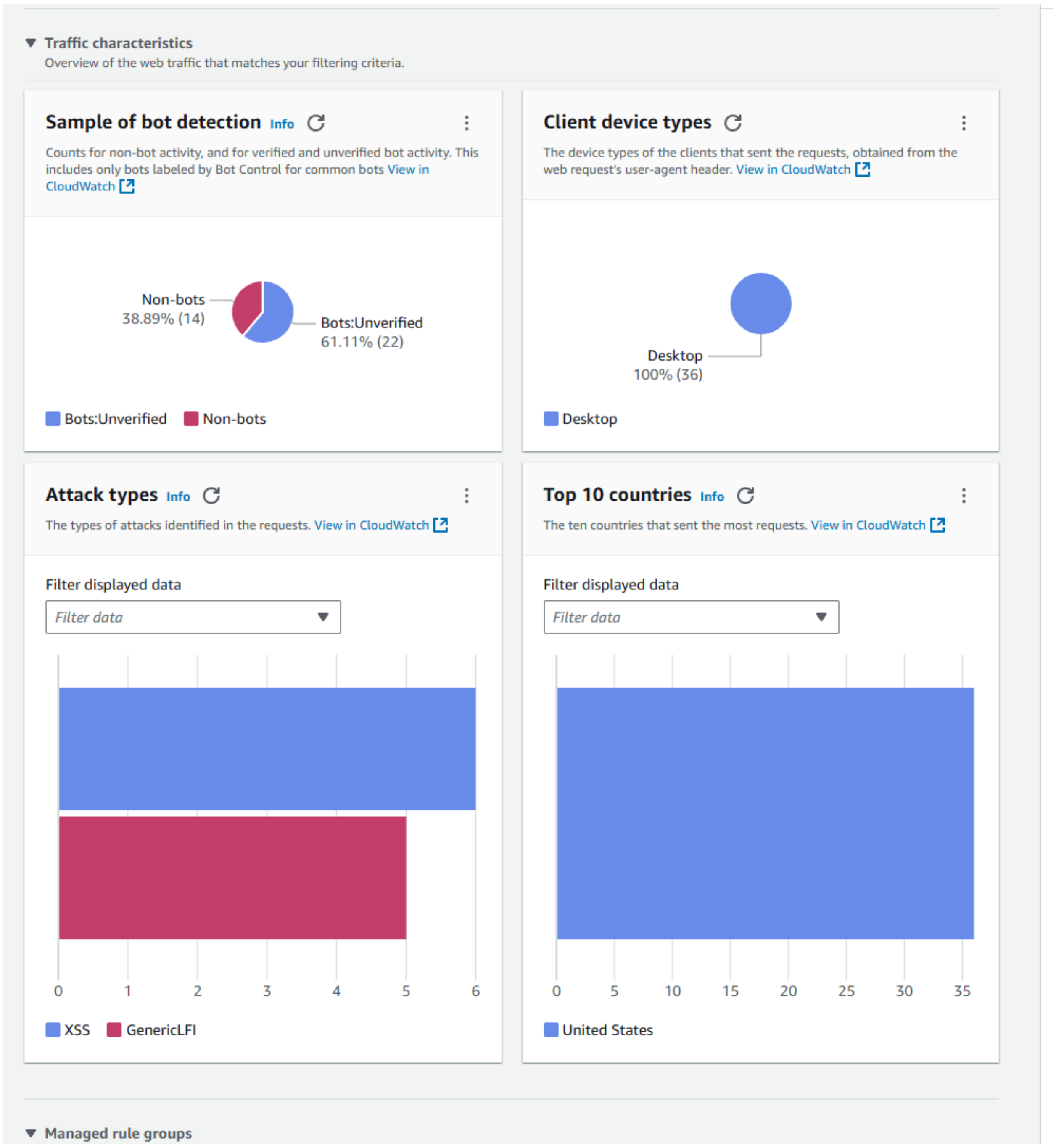
Request counts for each selected terminating action. View in CloudWatch

Top 10 rules

Requests counts for the ten rules that matched the most requests during the selected time range. View in CloudWatch

Traffic characteristics

Overview of the web traffic that matches your filtering criteria.



▼ Traffic characteristics

Overview of the web traffic that matches your filtering criteria.

Sample of bot detection Info

Counts for non-bot activity, and for verified and unverified bot activity. This includes only bots labeled by Bot Control for common bots View in CloudWatch

Client device types

The device types of the clients that sent the requests, obtained from the web request's user-agent header. View in CloudWatch

Attack types Info

The types of attacks identified in the requests. View in CloudWatch

Filter displayed data

Filter data

Top 10 countries Info

The ten countries that sent the most requests. View in CloudWatch

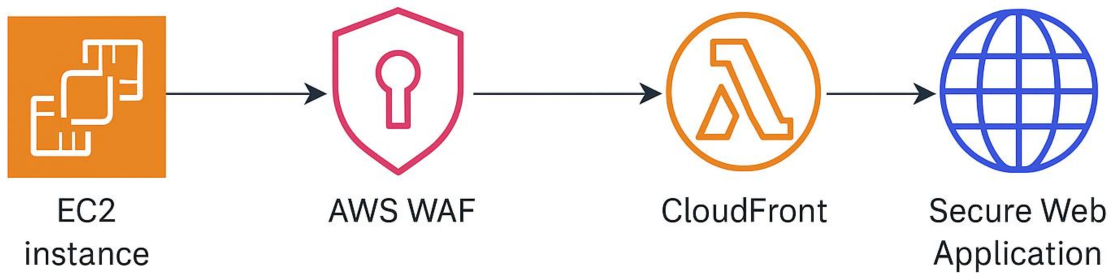
Filter displayed data

Filter data

▼ Managed rule groups

AWS Diagram:

Secure Web Application Deployment on AWS


EC2 instance
AWS WAF
CloudFront
Secure Web Application