Integrating Cequence Bot Defense SaaS with Amazon CloudFront

Contents

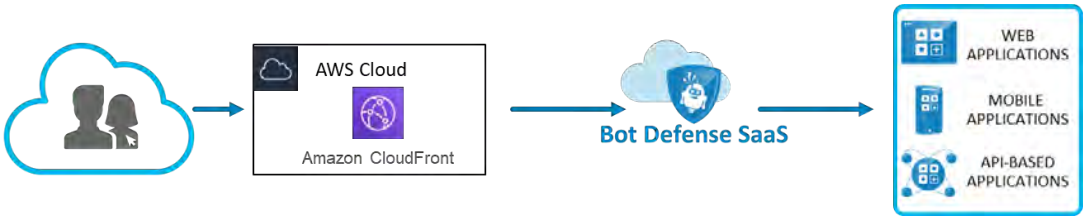
About Cequence Bot Defense SaaS and Amazon CloudFront

Bot Defense SaaS uses an ML-based approach to eliminate avenues of fraud caused by automated attacks targeted at your web, mobile and API-based applications deployed on AWS. Using a SaaS deployment model reduces the operation efforts associated with deploying Bot Defense to prevent account takeovers and API-based business logic abuse.

Amazon CloudFront, the highly secure and programmable content delivery network (CDN) integrates with Bot Defense SaaS, allowing you to analyze your public-facing application transactions to determine malicious or benign intent. The findings are then used to enforce policy or exported via a REST-based API to an existing component of your security infrastructure.

Traffic flow without Bot Defense SaaS:

Traffic flow with Bot Defense SaaS:



The steps required to integrate Bot Defense SaaS with Amazon CloudFront are relatively straightforward. Selected traffic that terminates on Amazon CloudFront will be routed to Bot Defense SaaS for inspection before it is forwarded to the application origin.

Step 1: Configure Application Availability

Application availability must be ensured with the addition of Bot Defense SaaS to the traffic flow between Amazon CloudFront and application origin. In the rare event where the Bot Defense SaaS becomes unavailable (determined via a health check), a fail-open must kick in and all application traffic from Amazon CloudFront must get routed directly to the application origin, bypassing Bot Defense SaaS completely. Such a fail-open scenario can be configured with a failover routing policy configuration.

To create a failover routing policy, either one of the below solutions can be leveraged:

- Amazon Route 53
- Bot Defense SaaS Traffic Manager (for customers that don't use Amazon Route 53)

The snapshots below show an Amazon Route 53 failover routing policy example, where the DNS Hostname test-cq.emadisonisland.com is pointing at two CNAME records:

1. Bot Defense SaaS origin: test.s.cequence.cloud (set as primary)
2. Sample application origin: origin-www.emadisonisland.com (set as secondary)

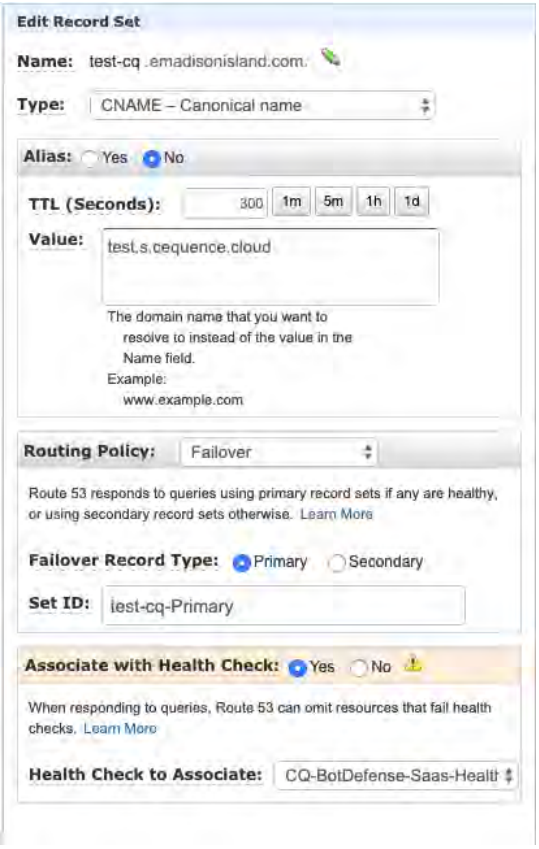


Image 1: Bot Defense SaaS origin set as primary with associated health check

Image 2: Application origin set as secondary

The DNS hostname, test-cq.emadisonisland.com in this example, will be set as the origin hostname for forwarding traffic to Bot Defense SaaS on the Amazon CloudFront configuration.

Step 2: Configure Bot Defense SaaS Origin

In this step, configure the Bot Defense SaaS as a new origin:

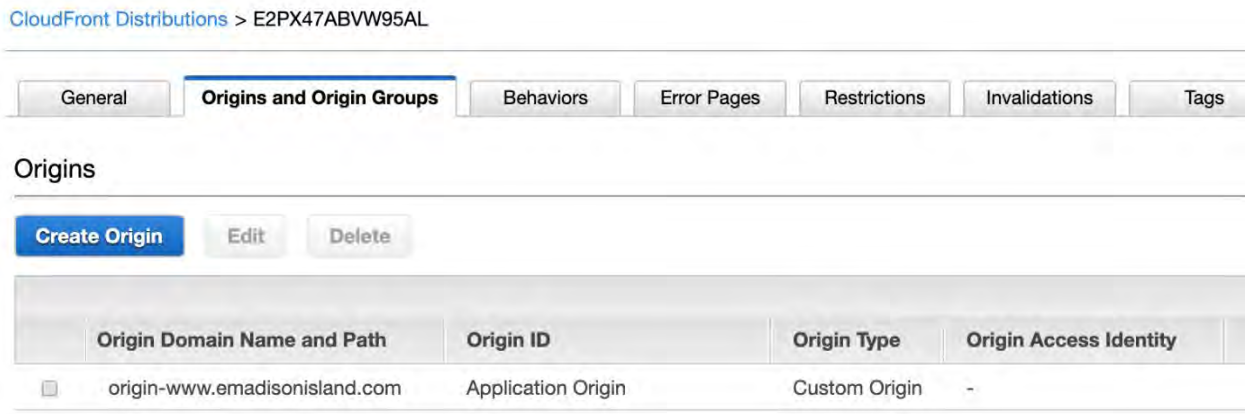


Image 3: The existing application origin configured with Amazon CloudFront

- Go to the Origins and Origin Groups tab and click on Create Origin where you will create and define the new origin settings for Bot Defense SaaS. Change the origin settings configuration as shown in image 2 on the following page.
- In the example, the Origin Domain Name field is where the origin hostname for Bot Defense SaaS created in Step 1 will need to be entered
- All the other settings should be configured exactly as shown in the screenshot. Once complete, click Create.
- This will complete the creation of Bot Defense SaaS origin on the existing Amazon CloudFront distribution.

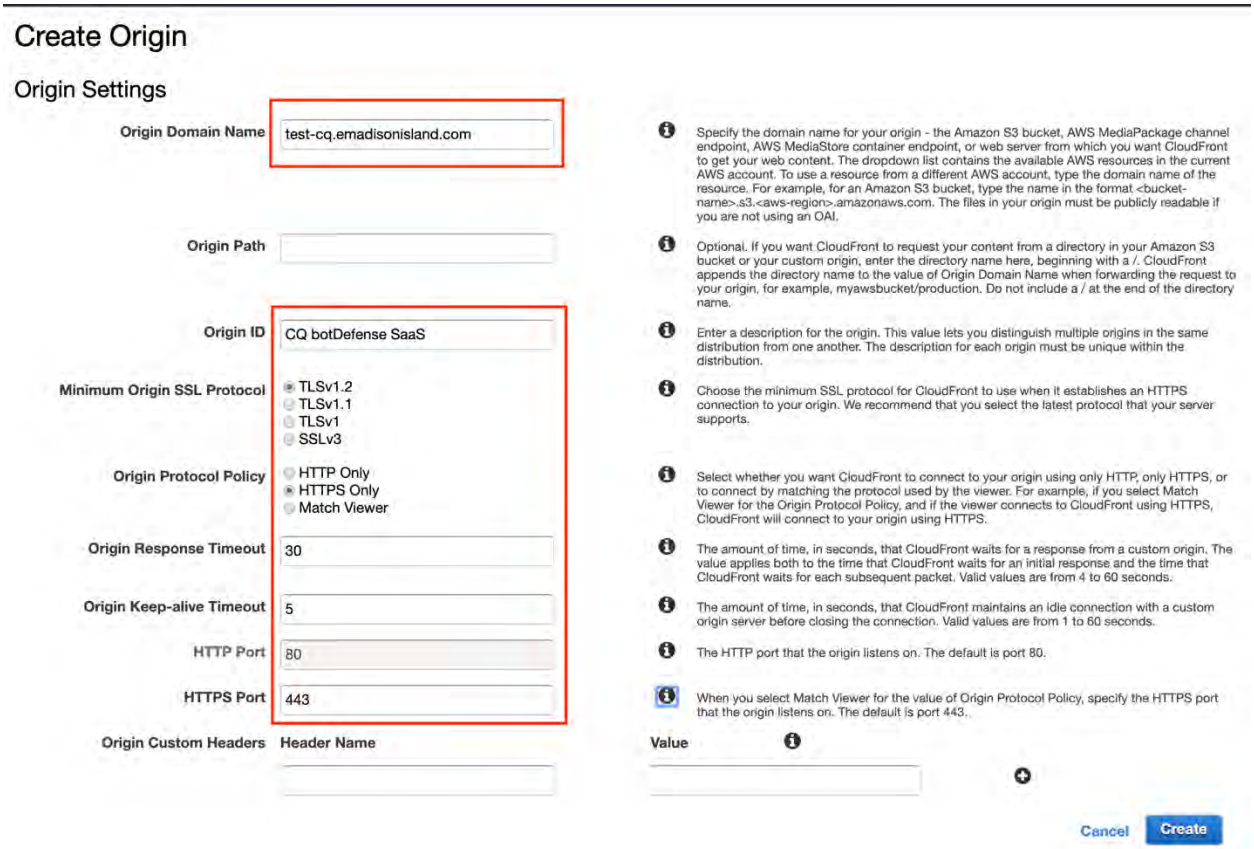


Image 4: Behavior modification

Step 3: Configure Traffic Forwarding to Bot Defense SaaS

To configure forwarding of all application traffic to Bot Defense SaaS origin as shown in the previous step, we will need to make Bot Defense SaaS the default origin.

- Go to the **Behaviors** tab and select the Origin that has the **Path Pattern** of **Default (*)**
- In the example screenshot - Behaviors: screenshot (a), the existing customer application origin **Application Origin** is set with the **Path Pattern** of **Default (*)**

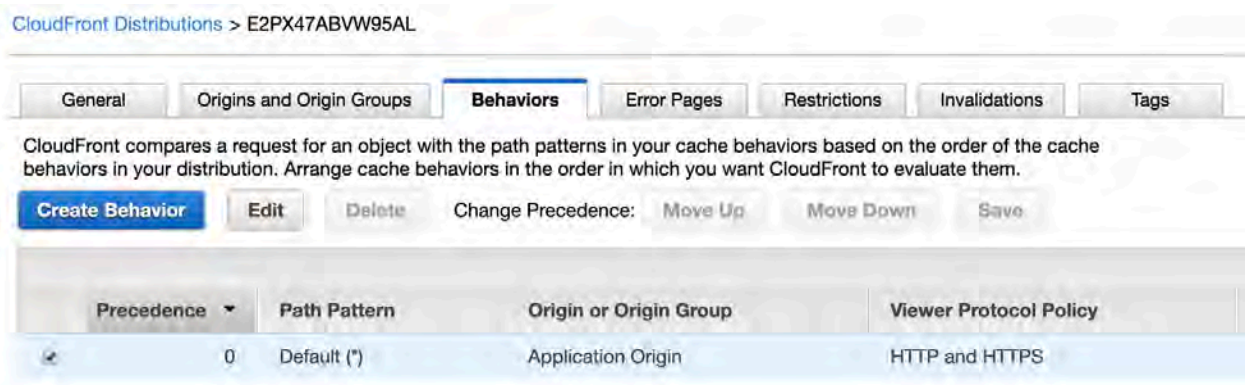


Image 5: Modifying CloudFront behavior (a)

- Click **Edit** to update the default behavior settings
- Select **Bot Defense SaaS** in the **Origin or Origin Group,** as shown in image 7.
- Click **Yes, Edit**
- As shown in Image 6, the **Behaviors** tab will now be updated and reflect **Bot Defense SaaS** as the Origin with **Path Pattern** of **Default (*).**

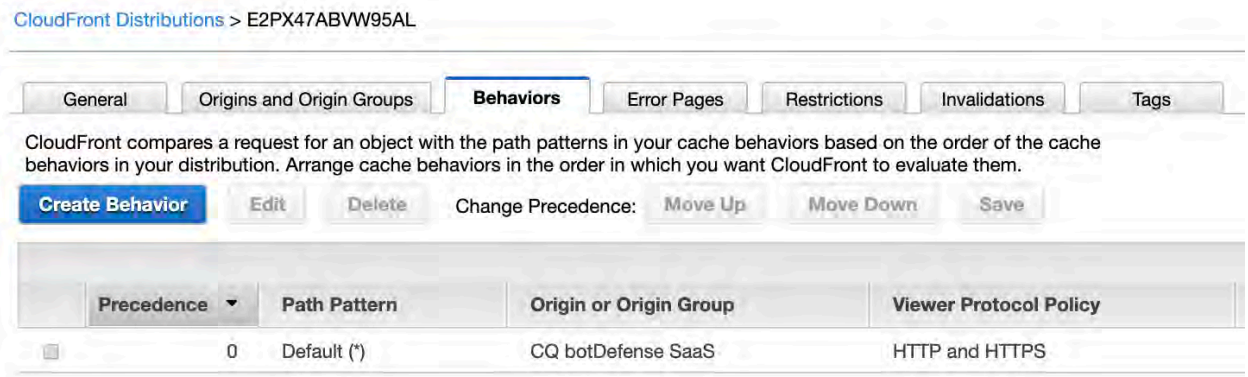


Image 6: Modifying CloudFront behavior (b)

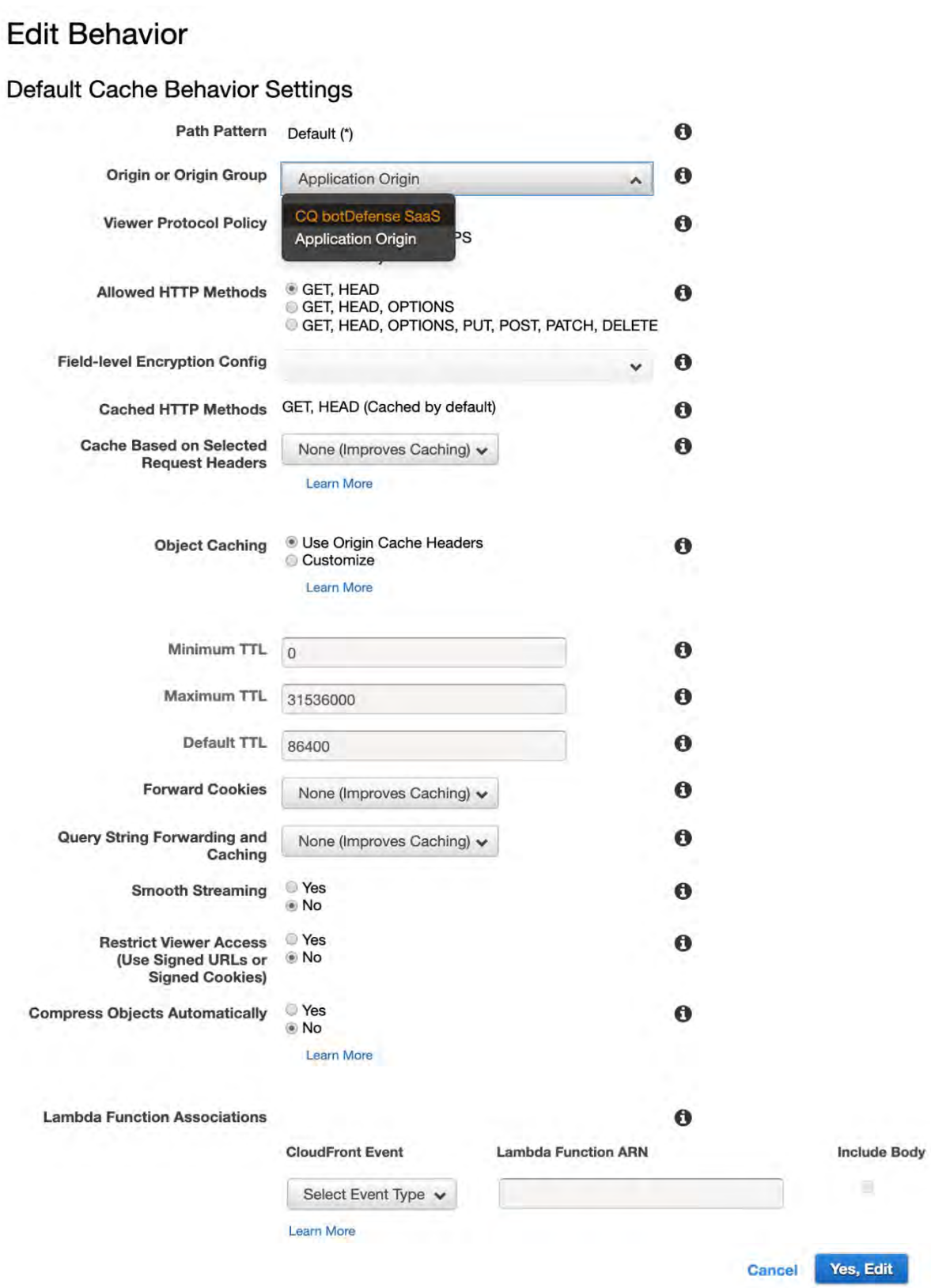


Image 7: Modifying default cache behavior