Integrating CQ botDefense SaaS with Fastly

Contents

About Cequence CQ botDefense SaaS and Fastly ................................................................. 2

Step 1: Configure CQ botDefense SaaS Origin ................................................................. 3

Step 2: Configure Application Availability ........................................................................ 5

Step 3: Configure Traffic Forwarding to CQ botDefense SaaS ........................................ 8

Step 4: Configure Traffic Forwarding to Application Origin – Loopback only ................ 10
About Cequence CQ botDefense SaaS and Fastly

CQ botDefense SaaS uses an agentless, ML-based approach to eliminate avenues of fraud caused by account takeovers and API business logic abuse.

When integrated with Fastly, traffic is directed to CQ botDefense SaaS where it is analyzed by the CQ AI ML-based automation indicators to determine malicious or benign intent. CQ AI 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 CQ botDefense SaaS:

Traffic flow with CQ botDefense SaaS (option 1):

Traffic flow with CQ botDefense SaaS in a loopback architecture (option 2):

The steps required to integrate CQ botDefense SaaS with Fastly are relatively straightforward. All traffic that terminates on Fastly will be routed to CQ botDefense SaaS first for inspection and then forwarded to the application origin (option 1) or forwarded back to Fastly from where it will be routed to the application origin (option 2).
Step 1: Configure CQ botDefense SaaS Origin

The configuration of CQ botDefense SaaS origin and forwarding traffic to it will be explained using an example scenario where:

- Web Application: test.emadisonisland.com
- Application Origin: ec2-54-188-157-137.us-west-2.compute.amazonaws.com
- CQ botDefense SaaS Origin: test.s.cequence.cloud

Navigate to the Origins > Hosts section to add the CQ botDefense SaaS Origin on an existing Fastly configuration.
Image 2: Add Cequence provided CQ botDefense SaaS origin hostname

Fill out the Host Details as shown below and leave the other options as defaults:

Image 3: Configure CQ botDefense SaaS origin host
The hosts, both for the customer’s Application Origin as well as for CQ botDefense SaaS Origin will be shown as below:

![Image 4: CQ botDefense SaaS origin and customer application origin hosts](image)

**Step 2: Configure Application Availability**

Application availability must be ensured with the addition of CQ botDefense SaaS to the traffic flow between Fastly and Application Origin.

In the rare event where the CQ botDefense SaaS becomes unavailable (determined via a health check) a fail-open must kick in and all application traffic from Fastly must get routed directly to the Application Origin, bypassing CQ botDefense SaaS completely.

Fastly offers the capabilities to set up a fail-open configuration using health checks.

To create the health check for the CQ botDefense SaaS origin, navigate to Origins > Hosts and create a health check under the Health checks section.
The Health Check needs to have the Host header field present in order to allow CQ botDefense SaaS to forward the Health Check traffic onto the Application Origin.

In the example below, the health check traffic is sent to the URI of “/” along with the respective Host Header and a 302 response is expected in order to indicate a success.
Once the Health Check configuration is created, edit the CQ botDefense SaaS Host configuration (test.s.cequence.cloud, in our case) and assign to it the Health Check that was created.
After assignment of the Health check, the Host summary should appear as below:

Step 3: Configure Traffic Forwarding to CQ botDefense SaaS

In the Fastly configuration, the Host that is configured without attaching any condition is treated as the Default Host for forwarding application traffic.

Since all application traffic from Fastly will first need to be forwarded to CQ botDefense SaaS, we do not attach any condition to it.

Instead, a condition will be attached to forward traffic to the customer’s Application Origin. This condition will typically be that of a health check failure to CQ botDefense SaaS in order to trigger a fail-open to the Application Origin.
To set this up, click on Attach a condition for the Customer Application Origin Host, and create the condition as shown below:

![Image 7: See Attach a condition]

Create a new request condition and Save the configuration.

![Image 8: Create a health check fail-open condition]

The below image is indicative of the way Origin > Hosts section would appear after CQ botDefense SaaS has been configured as the Default Origin and a condition has been attached for forwarding traffic to the Application Origin.
Step 4: Configure Traffic Forwarding to Application Origin – Loopback Only

As shown in the loopback architecture traffic flow diagram (option 2 on page 2), Fastly forwards all application traffic to CQ botDefense SaaS, by default.

- CQ botDefense SaaS then adds a pre-shared key in a specialized request header to all the application traffic it processes and forwards to Fastly.

- When this traffic hits Fastly again, using a conditional placed on the presence of the pre-shared key in the specialized request header added by CQ botDefense SaaS, Fastly makes the determination to no longer forward traffic to CQ botDefense SaaS, and instead forwards it to the Application Origin.