Amazon WAF Security Automations

Implementation Guide

latest
Amazon Web Services

Table of Contents

Welcome

Cost

Architecture overview

Considerations

Template

Automated deployment

Contributors

Revision

Notices

Welcome

Automatically deploy a single web access control list that filters web-based attacks with Amazon WAF Security Automations

Publication date: September 2016 (last update: April 2022)

Amazon WAF helps protect web applications from common exploits that can affect application availability, compromise security, or consume excessive resources. Amazon WAF allows you to define customizable web security rules, and control which traffic to allow to web applications and APIs deployed on Amazon CloudFront, an Application Load Balancer, or Amazon API Gateway.

Configuring WAF rules can be challenging, especially for organizations that do not have dedicated security teams. To simplify this process, Amazon Web Services offers the Amazon WAF Security Automations solution, which automatically deploys a single web access control list (web ACL) with a set of Amazon WAF rules that filters web-based attacks. During initial configuration the Amazon CloudFormation template, you can specify which protective features to include. After this solution is deployed, Amazon WAF inspects web requests to existing CloudFront distributions or Application Load Balancer, and blocks them if applicable.

This implementation guide discusses architectural considerations and configuration steps for deploying the Amazon WAF Security Automations solution in the Amazon Web Services Cloud. It includes links to Amazon CloudFormation templates that launch, configure, and run the compute, network, storage, and other services required to deploy this solution on Amazon Web Services, using Amazon Web Services best practices for security and availability.

The information in this guide assumes working knowledge of services such as Amazon WAF, Amazon CloudFront, Application Load Balancers, and Amazon Lambda. It also requires basic knowledge of common web-based attacks, and mitigation strategies.

The guide is intended for IT Managers, Security Engineers, DevOps Engineers, Developers, Solutions Architects, and Website Administrators.

Cost

Cost

You are responsible for the cost of the Amazon Web Services used while running the Amazon WAF Security Automations solution. The total cost for running this solution depends on the protection activated and the amount of data ingested, stored, and processed.

We recommend creating a budget through Amazon Cost Explorer to help manage costs. For full details, refer to the pricing webpage for each service used in this solution.

The following tables are example cost breakdowns for running this solution in the Ningxia Region (excludes free tier). Prices are subject to change.

Example 1: Turn on Reputation List Protection, Bad Bot Protection, and Lambda Log Parser for HTTP Flood Protection and Scanner & Probe Protection.

Service	Dimensions/Month	Cost/ Month
Amazon Kinesis Data Firehose	100 GB	~¥26.60
Amazon Simple Storage Service	100 GB	~¥17.55
Amazon Lambda	128 MB: 3 functions, total of 1M invocations and average 500 millisecond duration per Lambda run	~¥38.18
	512 MB: 2 functions, total of 1M invocations and average 500 millisecond duration per Lambda run	
Amazon API Gateway	1M requests	~¥7.06
Total		~¥89.39

Example 2: Turn on Reputation List Protection, Bad Bot Protection, and Athena Log Parser for HTTP Flood Protection and Scanner & Probe Protection

Service	Dimensions/Month	Cost/ Month
Amazon Kinesis Data Firehose	100 GB	~¥26.60
Amazon Simple Storage Service	100 GB	~¥17.55
Amazon Lambda	128 MB: 3 functions, total of 1M invocations and average 500 millisecond duration per Lambda run	~¥8.67
	512 MB: 2 functions, total of 7560 invocations and average 500 millisecond duration per Lambda run	
Amazon API Gateway	1M requests	~¥7.06
Amazon Athena	1.2M ALB requests per day that generates a 500 byte log record per hit/request	~¥29.67
Total		~¥89.55

Example 3: Turn on IP retention on Allowed and Denied IP sets

Service	Dimensions/Month	Cost/ Month
Amazon DynamoDB	1K writes, 1MB data storage	~¥0
Amazon Lambda	128 MB: 1 function, total of 2K invocations and average 500 millisecond duration per Lambda run	~¥0.07
	512 MB: 1 function, total of 2K invocations and average 500 millisecond duration per Lambda run	
Amazon CloudWatch	2K events	~¥0.00
Total		~¥0.07

There are Amazon services used in this solution, such as Amazon Lambda, that generate Amazon CloudWatch logs. These logs incur charges. We recommend deleting or archiving old logs to reduce the cost. For log archive detail, refer to Exporting log data to Amazon S3 in the Amazon CloudWatch Logs User Guide.

If you choose to use the Athena log parser on installation, this solution schedules a query to run against the WAF and/or application access logs in your Amazon S3 bucket(s) as configured. You are charged based on the amount of data scanned by each query. Partitioning is applied to logs and queries to keep costs low. By default, application access logs are moved from their original S3 location to a partitioned folder structure. You have the option to keep original logs as well but you will be charged for duplicated log storage. This solution uses Workgroups to segment workloads and these can be configured to manage query access and costs. Refer to Cost estimate of Amazon Athena for a sample cost estimate calculation. For more information, refer to Amazon Athena Pricing.

Architecture overview

Protection capabilities

Web applications are vulnerable to a variety of attacks. These attacks include specially crafted requests designed to exploit a vulnerability or take control of a server; volumetric attacks designed to take down a website; or bad bots and scrapers programmed to scrape and steal web content.

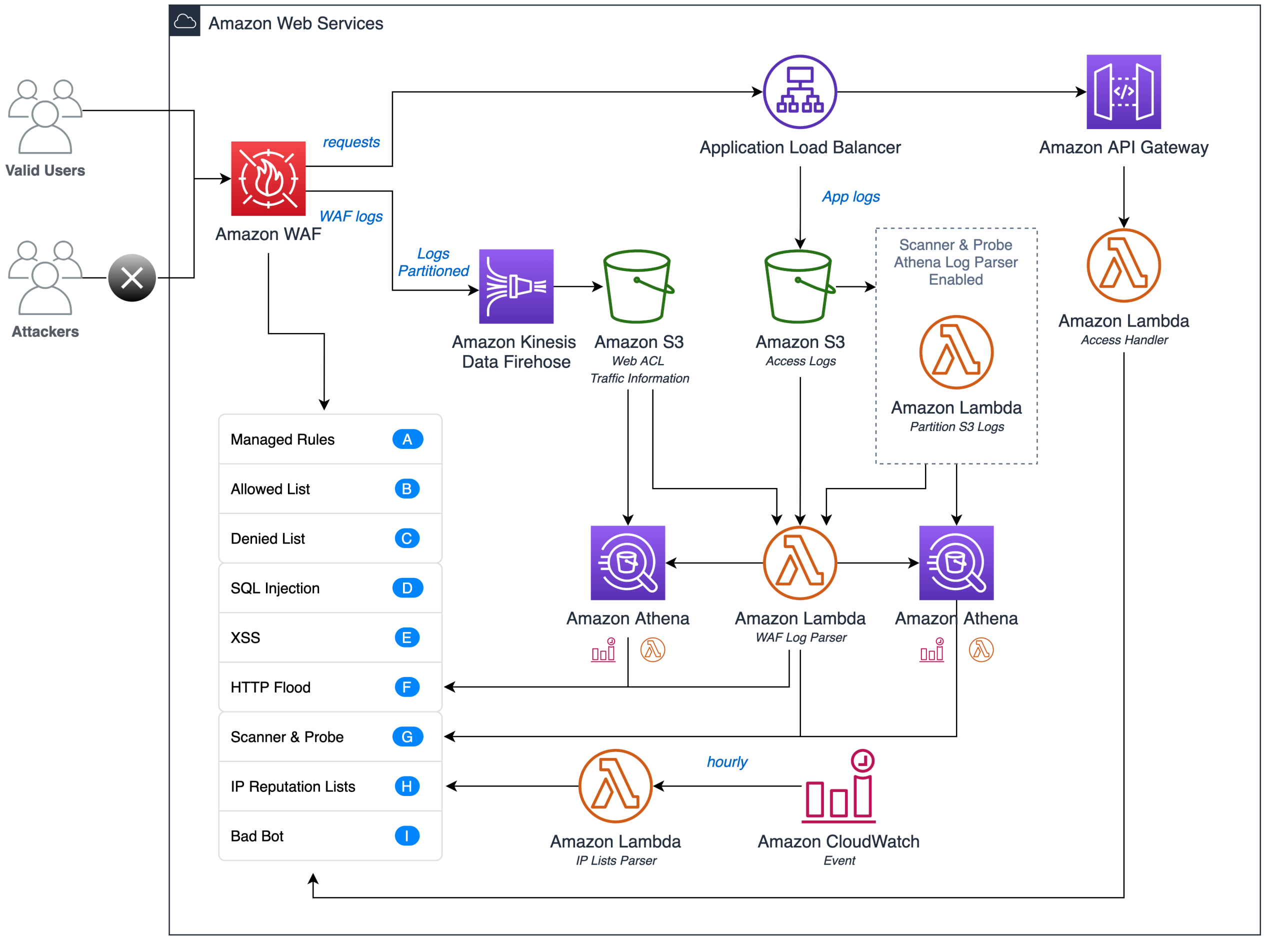
This solution uses Amazon CloudFormation to configure Amazon WAF rules to block the following common attacks:

- SQL injection: Attackers insert malicious SQL code into web requests in an effort to extract data from your database. This solution is blocks web requests that contain potentially malicious SQL code.
- Cross-site scripting: Also known as XSS, attackers use vulnerabilities in a benign website as a vehicle to inject malicious client-site scripts into a legitimate user's web browser. This solution inspects commonly explored elements of incoming requests to identify and block XSS attacks.
- HTTP floods: Web servers and other backend resources are at risk of Distributed Denial of Service (DDoS) attacks, such as HTTP floods. This solution automatically triggers a rate-based rule when web requests from a client exceed a configurable threshold. Alternatively, enforce this threshold by processing Amazon WAF logs using an Amazon Lambda function or an Amazon Athena query.
- Scanners and probes: Malicious sources scan and probe Internet-facing web applications for vulnerabilities, by sending a series of requests that generate HTTP 4xx error codes. You can use this history to help identify and block malicious source IP addresses. This solution creates an Amazon Lambda function or an Amazon Athena query that automatically parses Amazon CloudFront or Application Load Balancer access logs, counts the number of bad requests from unique source IP addresses per minute, and updates Amazon WAF to block further scans from addresses with high error rate – the ones that reached the defined-error threshold.

- Known attacker origins (IP reputation lists): A number of organizations maintain reputation lists of IP addresses operated by known attackers, such as spammers, malware distributors, and botnets. This solution leverages the information in these reputation lists to help you block requests from malicious IP addresses.
- Bots and scrapers: Operators of publicly accessible web applications have to trust that the clients accessing their content identify themselves accurately, and that they will use services as intended. However, some automated clients, such as content scrapers or bad bots, misrepresent themselves to bypass restrictions. This solution helps you identify and block bad bots and scrapers.

Architecture overview

Deploying this solution with the default parameters builds the following environment in the Amazon Web Services Cloud.



At the core of the design is an Amazon WAF web ACL, which acts as the central inspection and decision point for all incoming requests to a web application. During initial configuration of the Amazon CloudFormation stack, you define the protective components to activate. Each component operates independently and adds different rules to the web ACL.

The components of this solution can be grouped into the following areas of protection:

- Amazon Web Services Managed Rules (A): This component contains a set of Amazon WEb Services managed core rules that are generally applicable to web applications. It provides protection against exploitation of a wide range of common application vulnerabilities or other unwanted traffic, including those described in OWASP publications, without having to write your own rules.
- Manual IP lists (B and C): This component creates two specific Amazon WAF rules that allow you to manually insert IP addresses that you want to allow or deny. You can configure IP retention and remove expired IP addresses on allowed or denied IP sets. For information, refer to IP Retention on allowed and denied WAF IP sets.
- SQL injection (D) and XSS (E): This solution configures two native Amazon WAF rules that are designed to protect against common SQL injection or cross-site scripting (XSS) patterns in the URI, query string, or body of a request.
- HTTP flood (F): This component protects against attacks that consist of a large number of requests from a particular IP address, such as a web-layer DDoS attack or a brute-force login attempt. With this rule, you set a threshold that defines the maximum number of incoming requests allowed from a single IP address within a five-minute period. Once this threshold is breached, additional requests from the IP address are temporarily blocked. You can implement this rule by using an Amazon WAF rate-based rule or by processing Amazon WAF logs using an Amazon Lambda function or an Amazon Athena query. For more information about the tradeoffs related to HTTP flood mitigation options, refer to Log parser options.
- Scanners and Probes (G): This component parses application access logs searching for suspicious behavior, such as an abnormal amount of errors generated by an origin. It then blocks those suspicious source IP addresses for a customer-defined period of time. You can implement this rule using an Amazon Lambda function or an Amazon Athena query. For more information about the tradeoffs related to Scanners and Probes mitigation options, refer to Log parser options.
- IP Reputation Lists (H): This component is the IP Lists Parser Amazon Lambda function which checks third-party IP reputation lists hourly for new ranges to block. These lists

include the Spamhaus Don’t Route Or Peer (DROP) and Extended DROP (EDROP) lists, the Proofpoint Emerging Threats IP list, and the Tor exit node list.

- Bad Bots (I): This component automatically sets up a honeypot, which is a security mechanism intended to lure and deflect an attempted attack. This solution’s honeypot is a trap endpoint that you can insert in your website to detect inbound requests from content scrapers and bad bots. If a source accesses the honeypot, the Access Handler Amazon Lambda function will intercept and inspect the request to extract its IP address, and then add it to an Amazon WAF block list.

Each of the three custom Amazon Lambda functions in this solution publish execution metrics to Amazon CloudWatch. For more information on these Lambda functions, refer to Component details.

Considerations

The following sections provide constraints and considerations for implementing this solution.

> **Note**
>
> The included Amazon CloudFormation template should be used as a starting point for implementing Amazon WAF rules. We recommend adding custom rules, applying log analysis, and leveraging Amazon WAF managed rules, based on your company's needs.

Amaozn WAF

Web ACL rules

The web ACL that this solution generates is designed to offer comprehensive protection for web applications. The default configuration adds nine Amazon WAF rules to this solution's web ACL. You can manually modify the web ACL to add further rules, which is subject to the Amazon WAF service limits. This solution also supports including Amazon Managed Rules as the first priority before all additional eight custom Amazon WAF rules. To include Amazon Web Services Managed Rules, choose yes for the relevant box in the parameter list when launching the CloudFormation stack.

IP match conditions

Amazon WAF can block a maximum of 10,000 IP address ranges in Classless Inter-Domain Routing (CIDR) notation per IP match condition. Each list is subject to this limit. The allow list, deny list (manual IP lists component), and third-party IP block list (IP list parsing component) are separate lists, each with a 10,000 IP address limit. Refer to Amazon WAF quotas (formerly called limits) in the Amazon WAF Developer Guide for more information. Starting from version 3.0, this solution creates two IP sets to attach to each rule, one for IPv4 and one for IPv6.

Regions and multiple deployments

This solution includes an Amazon CloudFormation template for web applications. The template contains two nested templates: one that deploys the Web ACL and a separate template that includes resources related to Amazon Glue, Amazon Athena, and Amazon

Kinesis Data Firehose. This solution chooses which nested template to deploy based on the user selected input template parameters. Refer to the parameters table under Step 1, for details about services dependencies.

Customers can deploy the Amazon WAF Security Automations solution in different Regions, or deploy it multiple times in the same Region. Note that each unique deployment will incur additional charges.

> **Note**
>
> If you plan to configure multiple instances of this solution in the same Region, you must use a unique Amazon CloudFormation stack name and Amazon S3 bucket name for each deployment.

Template

CloudFormation Template Resource

This solution uses Amazon CloudFormation to bootstrap Amazon Web Services infrastructure and automate the deployment of Amazon WAF Security Automations on the Amazon Web Services Cloud. It includes the following Amazon CloudFormation template.

aws-waf-security-automations.template: Use this template to launch the Amazon WAF Security Automations solution for web applications. The default configuration deploys an Amazon WAF web ACL with eight preconfigured rules, but you can also customize the template based on your specific needs.

Automated deployment

Before you launch the automated deployment, please review the architecture, configuration, and other considerations discussed in this guide. Follow the step-by-step instructions in this section to configure and deploy Instance Scheduler into your account.

Time to deploy: Approximately 15 minutes

Prerequisites

This solution is designed to work with web applications deployed with an Application Load Balancer. If you don't already have one of these resources configured, complete the applicable task before you launch this solution.

Configure an Application Load Balancer

Complete the following steps to configure an Application Load Balancer to distribute incoming traffic to your web application. Refer to the Application Load Balancer Guide for detailed instructions.

What we'll cover

The procedure for deploying this architecture consists of the following steps. For detailed instructions, follow the links for each step.

Step 1. Launch the stack

- Launch the Amazon CloudFormation template into your Amazon Web Services account.
- Enter values for the required parameters: Stack Name, Application Access Log Bucket Name.
- Review the other template parameters, and adjust if necessary.

Step 2. Modify the Allowed and Denied sets (Optional)

- Manually add applicable IP addresses to the Amazon WAF accept list and deny list.

Step 3. Embed the Honeypot link in your web application (Optional)

- Embed the hidden trap endpoint in your application.

Step 4. Associate the web ACL with your web application

- Associate your Application Load Balancers with the web ACL that this solution generates. You can associate as many load balancers you want.

Step 5. Configure web access logging

- Enable web access logging for your Amazon Application Load Balancer, and send log files to the appropriate Amazon S3 bucket.

Step 1. Launch the stack

> **Important**
>
> This solution includes an option to send anonymous operational metrics to Amazon Web Services. We use this data to better understand how customers use this solution and related services and products. Amazon owns the data gathered though this survey. Data collection is subject to the Amazon Privacy Policy. To opt out of this feature, download the template, modify the Amazon CloudFormation mapping section, and then use the Amazon CloudFormation console to upload your template and deploy the solution. For more information, refer to the Collection of operational metrics section of this guide.

This automated Amazon CloudFormation template deploys the Amazon WAF Security Automations solution on the Amazon Web Services Cloud.

> **Note**
>
> You are responsible for the cost of the services used while running this solution. For full details, refer to the pricing webpage for each service you will be using in this solution.

1. Sign in to the Amaozn Web Services Management Console and click the link to launch the aws-waf-security-automations CloudFormation stack. You can also download the template as a starting point for your own implementation.
2. The template is launched in the Ningxia Region by default. To launch this solution in a different Region, use the region selector in the console navigation bar.
3. On the Specify template page, verify that you selected the correct template and choose Next.
4. On the Specify stack details page, assign a name to Amazon WAF configuration in the Stack name field. This will also be the name of the web ACL that the template creates.

5. Under Parameters, review the parameters for the template, and modify them as necessary. To opt out of a particular feature, choose none or no as applicable. This solution uses the following default values.

Parameter	Default	Description
Stack Name	WAFSecurityAutomations	The stack name cannot contain spaces and must be unique within your Amazon account. This will also be the name of the web ACL that the template creates.
Protection List		
Activate Managed Rules Protection	no	Choose *yes* to turn on the component designed to add Managed Rules to the top of the Web ACL priority list.
Activate SQL Injection Protection	yes	Choose *yes* to turn on the component designed to block common SQL injection attacks.
Activate Cross-site Scripting Protection	yes	Choose *yes* to turn on the component designed to block common XSS attacks.
Activate HTTP Flood Protection	yes - Amazon WAF rate-based rule	Select the component used to block HTTP flood attacks. Refer to Log parser options for more information about the tradeoffs related the mitigation options.
Activate Scanner and Probe Protection	yes - Lambda log parser	Select the component used to block scanners and probes. Refer to Log parser options for more information about the tradeoffs related the mitigation options.
Activate Reputation List Protection	yes	Choose *yes* to block requests from IP addresses on third-party reputation lists (supported lists: spamhaus, torproject, and emergingthreats).
Activate Bad Bot Protection	yes	Choose *yes* to turn on the component designed to block bad bots and content scrapers.

Parameter	Default	Description
Log Monitoring Settings		
Endpoint Type	ALB	Select the type of resource being used.
Application Access Log Bucket Name	"Require input"	If you select *yes* for the Activate Scanner & Probe Protection parameter, enter the name of the Amazon S3 bucket where you want to store access logs for your Application Load Balancer. To deactivate this protection, ignore this parameter. If you use an existing S3 bucket for this parameter, it must be located in the same Region where you are deploying the Amazon CloudFormation template. You cannot use the same Amazon S3 bucket for multiple deployments in the same Region.
Error Threshold	50	If you chose *yes* for the Activate Scanner & Probe Protection parameter, enter the maximum acceptable bad requests per minute per IP address. If you chose to deactivate this protection, ignore this parameter.
Request Threshold	100	If you chose *yes* for the Activate HTTP Flood Protection parameter, enter the maximum acceptable requests per five (5) minutes per IP address. The minimum acceptable value is 100 for the rate-based rule. If you are using Athena or a Lambda log parser, it can be any value. To deactivate this protection, ignore this parameter.

Parameter	Default	Description
WAF Block Period	240	If you chose *yes Athena or Lambda log parser* for the Activate Scanner & Probe Protection or Activate HTTP Flood Protection parameters, enter the period (in minutes) to block applicable IP addresses. To deactivate log parsing, ignore this parameter.
Keep Data in Original S3 location	No	If you chose *Amazon Athena log parser* for the Activate Scanners & Probes Protection parameter, partitioning will be applied to application access log files and Athena queries. By default, log files will be moved from their original location to a partitioned folder structure in Amazon S3. Choose *yes* if you also want to keep a copy of the logs in their original location. Choosing yes will duplicate your log storage. If you did not choose to activate Athena log parsing, ignore this parameter.
IP Retention Settings		
Retention Period (Minutes) for Allowed IP Set	-1	If you want to activate IP retention for the Allowed IP set, enter a number (15 or above) as the retention period (minutes). IP addresses reaching the retention period will expire and be removed from the IP set. A minimum 15-minute retention period is supported. If you enter a number between 0 and 15, it will be treated as 15. Leave it as the default value -1 to turn off IP retention.

Parameter	Default	Description
Retention Period (Minutes) for Denied IP Set	-1	If you want to activate IP retention for the Denied IP set, enter a number (15 or above) as the retention period (minutes). IP addresses reaching the retention period will expire and be removed from the IP set. A minimum 15-minute retention period is supported. If you enter a number between 0 and 15, it will be treated as 15. Leave it to default value -1 to disable IP retention.
Email for receiving notifcation upon Allowed or Denied IP Sets expiration		If you activated the IP retention period parameter and want to receive an email notification when IP addresses expire, enter a valid email address. If you did not activate IP retention or want to turn off email notifications, leave it blank (default).

1. Choose Next.
2. On the Configure stack options page, you can specify tags (key-value pairs) for resources in your stack and set additional options, and then choose Next.
3. On the Review page, review and confirm the settings. Check the boxes acknowledging that the template will create Amazon Identity and Access Management (IAM) resources and any additional capabilities required.
4. Choose Create to deploy the stack.

View the status of the stack in the Amazon CloudFormation console in the Status column. You should receive a status of CREATE_COMPLETE in approximately 15 minutes.

Note

In addition to the Log Parser, IP Lists Parser, Access Handler Amazon Lambda functions, this solution includes the helper and custom-resourceLambda functions, which run only during initial configuration or when resources are updated or deleted.

> When running this solution, you will see all functions in the Amazon Lambda console, but only the three primary solution functions are regularly active. However, do not delete the other two functions, as they are necessary to manage associated resources.

1. To see details for the stack resources, choose the Outputs tab. This will include the BadBotHoneypotEndpoint value, which is the API Gateway honeypot endpoint. Note this value because you will use it in Step 3.

Step 2. Modify the Allowed and Denied sets (Optional)

After deploying this solution's Amazon CloudFormation stack, you can manually modify the allowed and denied sets to add or remove IP addresses as necessary.

1. Open the Amazon WAF console, and in the left navigation pane, choose IP addresses.
2. Choose Whitelist Set and add IP addresses from trusted sources.
3. Choose Manual Blacklist Set and add IP addresses you want to block.

Step 3. Embed the Honeypot link in your web application (Optional)

If you chose to activate scanner and probe protection in Step 1, the Amazon CloudFormation template creates a trap endpoint to a low-interaction production honeypot, intended to detect and divert inbound requests from content scrapers and bad bots. Valid users will not attempt to access this endpoint. However, content scrapers and bots, such as malware that scans for security vulnerabilities and scrapes email addresses might attempt to access the trap endpoint. In this scenario, the Access Handler Amazon Lambda function will inspect the request in order to extract its origin, and then update the associated Amazon WAF rule to block subsequent requests from that IP address.

Use the applicable procedure to embed the honeypot link for requests from either a CloudFront distribution or an Application Load Balancer.

Embed the Honeypot Endpoint as an external link

Use this procedure for web applications that are deployed with an Application Load Balancer.

1. Open the Amazon CloudFormation console, choose the stack that you built in Step 1, and then choose the Outputs tab.
2. From the BadBotHoneypotEndpoint key, copy the endpoint URL.

3. Embed this endpoint link in your web content. Use the full URL that you copied in Step 2. Hide this link from your human users. As an example, review the following code sample:

> **Note**
>
> This procedure uses nofollow to instruct robots to not access the honeypot URL. However, because the link is embedded externally, you cannot include a robots.txt file to explicitly disallow the link. It is your responsibility to verify what tags work in your website environment. Do not use rel="nofollow" if your environment doesn't observe it.

Step 4. Associate the web ACL with your web application

Update your Application Load Balancer(s) to activate Amazon WAF and logging using the resources you generated in Step 1.

1. Open the Amazon WAF console and choose the web ACL that you want to use.
2. On the Associated Amazon resources tab, choose Add resources.
3. Under Resource type, choose Application Load Balancer.
4. Select a resource from the list, then choose Add to save your changes.

Step 5. Configure web access logging

Configure your Application Load Balancer to send web access logs to the appropriate Amazon S3 bucket so that this data is available for the Log Parser Amazon Lambda function.

Store web access logs from an Application Load Balancer

1. Open the Amazon Elastic Compute Cloud console.
2. In the navigation pane, choose Load Balancers.
3. Select your web application's Application Load Balancer.
4. On the Description tab, choose Edit attributes.
5. Choose Enable access logs.
6. For S3 location, type the name of the Amazon S3 bucket that you want use to store web access logs (defined in Step 1).
7. Set the log prefix as AWSLogs/. If you enter AWSLogs as prefix but get a message saying prefix cannot start with 'AWSLogs', then remove the prefix. Application Load Balancer will use AWSLogs as the default prefix.
8. Choose Save.

For more information, refer to Access Logs for Your Application Load Balancer in the Elastic Load Balancing User Guide.

Contributors

Contributors

The following individuals contributed to this document:

- Heitor Vital
- Lee Atkinson
- Ben Potter
- Vlad Vlasceanu
- Aijun Peng
- Chaitanya Deolankar
- Paul Li

Revision

Revision

Date	Change
May 2021	Initial release, version 4.1.0
April 2022	Update to version 4.2.0, support IP Retension

Notices

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current Amazon Web Services product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from Amazon Web Services and its affiliates, suppliers or licensors. Amazon Web Services products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of Amazon Web Services to its customers are controlled by Amazon Web Services agreements, and this document is not part of, nor does it modify, any agreement between Amazon Web Services and its customers.

The Amazon WAF Security Automations solution is licensed under the terms of the Apache License Version 2.0 available at https://www.apache.org/licenses/LICENSE-2.0