Web Application Firewall

Best Practices

Issue	48
Date	2024-02-01

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 WAF Cloud Mode Access Configuration

1.1 Preparations

To enable WAF protection, you need to add domain names of your web services to WAF and route website traffic to WAF. Before you start, get familiar with what you want to protect with WAF.

Website Service Review

Sort out all website services you want to protect with WAF. This helps you learn about status quo and specific data for making right decisions in configuring protection policies.

Table 1-1 Website services

Item	Description
Website and Service Information	
Daily peak traffic of website/web application services, including the bandwidth (in Mbit/s) and QPS	Use it as the basis for selecting the service bandwidth and QPS specifications. **NOTE** If your website traffic peak exceeds the maximum QPS specifications you are using, WAF will stop checking the traffic and directly forward it to the origin server. There is no protection for your website or applications.
Major user group (for example, major area that the requests originate from)	Determine the attack source and then set geolocation access control rules to block users from these areas.
Whether the service is a C/S architecture	If yes, check whether there is an app client, Windows client, Linux client, code callback, or any other client.
Location where the origin server is deployed	Decide which region to buy.

Item	Description
Operating system (Linux or Windows) and web service middleware (Apache, Nginx, or IIS) of the origin server	Check whether access control is enabled for the origin server. If yes, whitelist WAF back-to-source IP addresses.
Domain protocol	Check whether WAF supports the communication protocol used by your site. **NOTE** WAF can protect your website only when **Client Protocol** and **Server Protocol** are configured based on the real situation of your website. • **Client Protocol**: the protocol used by a client (for example, a browser) to access your website. You can select **HTTP** or **HTTPS**. • **Server Protocol**: the protocol used by WAF to forward requests from the client (such as a browser) to the origin server. You can select **HTTP** or **HTTPS**.
Service port	Check whether your service ports are within the port range supported by WAF. • Standard ports – Port 80: default port when the client protocol is set to HTTP – Port 443: default port when the client protocol is set to HTTPS • Non-standard ports Ports other than ports 80 and 443 **NOTE** If your website uses a non-standard port, make sure the WAF edition you plan to buy can protect the non-standard port. For details, see **Which Non-Standard Ports Does WAF Support?**
Whether TLS v1.0 or weak encryption suite is supported	Check whether WAF supports the encryption suite used by your site.
Whether advanced anti-DDoS, CDN, or other proxy services are deployed in front of WAF.	Check whether a proxy is used and whether domain name is resolved to a correct address.
Whether the client supports Server Name Indication (for HTTPS services)	If your domain name supports HTTPS, the client and server must support Server Name Indication (SNI).
Service interaction	Understand the service interaction process and service processing logic to facilitate subsequent configuration of protection policies.

Item	Description
Active users	Determine the severity of an attack event to take a low-risk measure to respond it.
Services and Attacks	
Service types and features (such as games, cards, websites, or apps)	Help analyze the attack signatures.
Inbound traffic range and connection status of a single user or a single IP address	Help determine whether a rate limiting policy can be configured per IP address.
User group attribute	For example, individual users, Internet cafe users, or proxy users
Whether your website experienced large-volumetric attacks, the attack type, and maximum peak traffic	Determine whether a DDoS protection service is required and determine the DDoS protection specifications based on the peak attack traffic.
Whether your website experienced CC attacks and the maximum peak QPS in a CC attack	Configure the protection policies based on attack signatures.
Whether the pressure test has been performed	Evaluate the request processing performance of the origin server to determine whether service anomaly occurs due to attacks.

Prerequisites

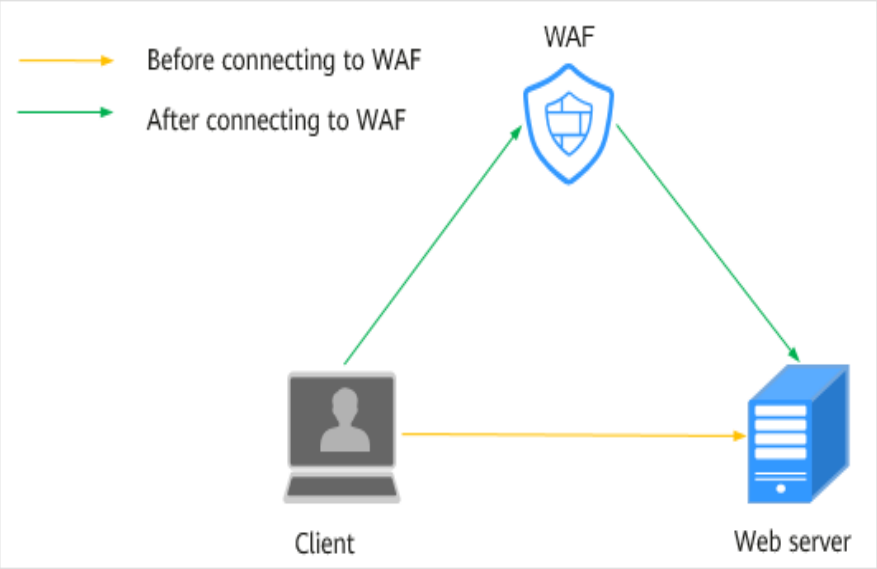
- The domain name information, such as the IP address and port for the origin server, has been added to WAF in **Cloud - CNAME** access mode.
- An administrator account is available for you to change DNS records for WAF to take effect.
- The pressure test has been performed.
- The IP addresses of trusted clients have been whitelisted if your website has trusted clients (such as certain monitoring systems, APIs invoked by internal IP addresses or IP address ranges, and program clients).

1.2 Connecting a Domain Name to WAF for Websites with no Proxy Used

If your website is not added to WAF, DNS resolves your domain name to the IP address of the origin server. If your website is added to WAF, DNS resolves your domain name to the CNAME of WAF. In this way, the traffic passes through WAF. WAF inspects every traffic coming from the client and filters out malicious traffic. This section describes how to change DNS settings for WAF to take effect.

Schematic Diagram

Figure 1-1 No proxy used



Prerequisites

- Website domain names are available.
- **WAF has been purchased**.
- The website information (such as the IP address and port number of the origin server) **has been added to WAF**.
- The account to update the DNS configuration is available.
- (Optional) You have whitelisted WAF back-to-source IP addresses. If other security software is used on the origin server, whitelist the WAF back-to-source IP addresses to prevent normal traffic from being blocked. For details, see **Configuring an Access Control Policy on an ECS or ELB to Protect Origin Servers**.
- (Optional) You have tested WAF before changing DNS settings. This can prevent service interruption due to incorrect configurations. For details, see **Testing WAF**.

Scenario

- If the **Type** of the domain name host record added on DNS is **CNAME - Map one domain to another**, complete the configuration based on the instructions in **CNAME Access**.

For details, see **Record Set Types and Configuration Rules**.

CNAME Access

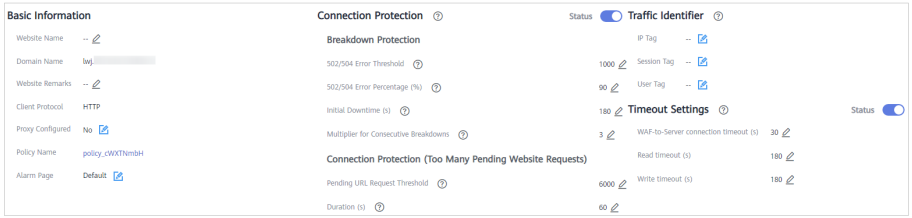
If the **Type** of the domain name host record added on DNS is **CNAME - Map one domain to another**, add the domain name to WAF by following the steps below.

The methods to change DNS records on different DNS platforms are similar. The following example is based on our Domain Name Service (DNS).

Step 1 Obtain the CNAME record.

1. Click in the upper left corner of the management console and select a region or project.
2. Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.
3. In the navigation pane, choose **Website Settings**.
4. In the **Domain Name** column, click the target domain name to go to the **Basic Information** page.

Figure 1-2 Basic Information

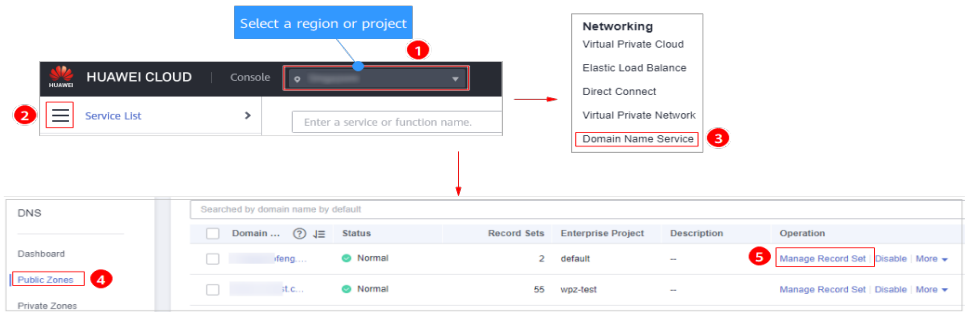


5. In the **CNAME** row, click to copy the CNAME record.

Step 2 Change the DNS settings.

1. Access the DNS resolution page, as shown in **Figure 1-3**.

Figure 1-3 DNS page



2. In the **Operation** column of the target domain name, click **Modify**. The **Modify Record Set** page is displayed.
3. In the displayed **Modify Record Set** dialog box, change the record.
   - **Name**: Domain name configured in WAF
   - **Type**: Select **CNAME - Map one domain to another**.
   - **Line**: **Default**
   - **TTL (s)**: The recommended value is **5 min**. A larger TTL value will make it slower for synchronization and update of DNS records.
   - **Value**: Change it to the copied CNAME value from WAF.
   - Keep other settings unchanged.

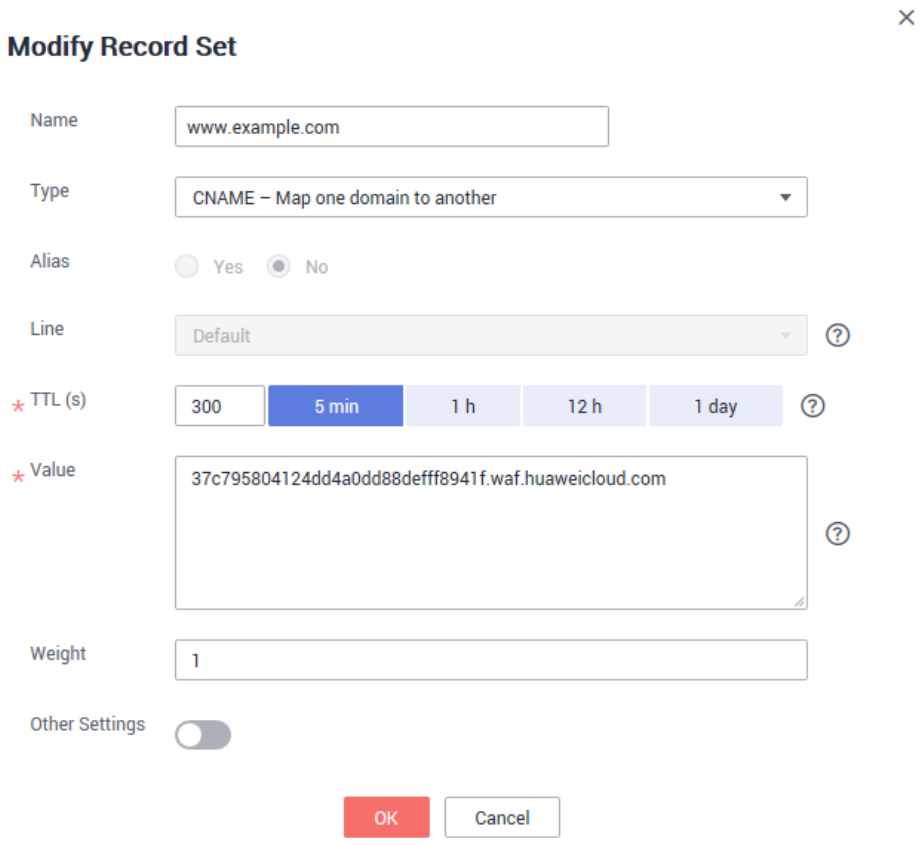
NOTE

About modifying the resolution record:

- The CNAME record must be unique for the same host record. The existing CNAME record must be changed to the WAF CNAME record.
- Record sets of different types in the same zone may conflict with each other. For example, for the same host record, the CNAME record conflicts with another record, such as the A record, MX record, or TXT record. If the record type cannot be changed, you can delete the conflicting records and add a CNAME record. Deleting other records and adding a CNAME record should be completed in as short time as possible. If no CNAME record is added after the A record is deleted, domain resolution may fail.

For details about the restrictions on domain name resolution types, see **Why Is a Message Indicating Conflict with an Existing Record Set Displayed When I Add a Record Set?**

Figure 1-4 Modifying a record set



4. Click **OK**.

Step 3 (Optional) Ping the IP address of your domain name to check whether the new DNS settings take effect.

NOTE

It takes some time for the new DNS settings to take effect. If ping fails, wait for 5 minutes and ping again.

----End

2 Best Practices for Website Protection

If you are a first-time user, you may not know how to configure website protection policies after adding a website to WAF. This topic describes how Web Application Firewall (WAF) works and helps you get familiar with the protection rules in WAF in many scenarios.

Prerequisites

- **The website has been connected to WAF**.
- The WAF edition you purchased supports the corresponding protection functions. For details about function differences between WAF editions, see **Edition Differences**.

Overview

This document provides suggestions on website protection settings from the perspectives of different roles or service requirements. You can select a scenario that best meets your actual requirements to learn about related protection settings.

- **I'm a novice. I know little about security and have no special requirements.**
- **I am a professional in security O&M, and I need comprehensive website protection operations.**
- **My services have strict requirements on security. Every attack must be killed even at the expense of more false positives.**
- **My business is often harassed by crawlers or faces data leakage and tampering risks.**

I'm a novice. I know little about security and have no special requirements.

You may have purchased WAF to meet security and compliance requirements or to improve the security for your organization to a higher level. In this case, you can use the default basic protection settings of WAF. The default protection capability provided by WAF is sufficient to defend websites against most basic web threats.

You can watch out for **Dashboard** and **Events** pages on the WAF console to learn about your services and their security status. For more details, see:

- **Dashboard**
- **Viewing Protection Events**

I am a professional in security O&M, and I need comprehensive website protection operations.

The following protection settings are recommended to you:

- **Basic Web Protection**: Defends against common web attacks, such as SQL injection, XSS, remote overflow attacks, file inclusion, Bash vulnerability exploits, remote command execution, directory traversal, sensitive file access, and command/code injection. It can also identify escape attacks in depth, check all fields in a request header, check Shiro encryption, and detect web shells.

  Operation: On the **Policies** page, click a policy name. On the displayed page, select **Basic Web Protection**, select **Block** or **Log only**, and enable all check items. For details, see **Configuring Basic Web Protection Rules**.
- **Custom protection policies**: You can create custom protection rules and add them to a policy to give your website comprehensive and tailored protection.

  Operation: On the **Policies** page, perform related configurations. For details, see **How to Configure WAF Protection**.

My services have strict requirements on security. Every attack must be killed even at the expense of more false positives.

To meet your requirements, the following protection configurations are recommended:

- **Basic Web Protection (block mode)**: Defends against common web attacks, such as SQL injection, XSS, remote overflow attacks, file inclusion, Bash vulnerability exploits, remote command execution, directory traversal, sensitive file access, and command/code injection. It can also identify escape attacks in depth, check all fields in a request header, check Shiro encryption, and detect web shells.

  Operation: On the **Policies** page, click a policy name. On the displayed page, select **Basic Web Protection**, select **Block**, and enable all check items. For details, see **Configuring Basic Web Protection Rules**.
- **CC attack protection (block mode)**: Helps precisely identify and block CC attacks by limiting the access rate of a single visitor based on its IP address, cookie, or referer.

  Operation: On the **Policies** page, click a policy name. On the displayed page, click the **CC Attack Protection** area, add a rule, and set the **Protective Action** to **Block**. For more details, see **Configuring a CC Attack Protection Rule**.
- **Precise Protection**: You can create custom protection rules by combining HTTP headers, cookies, URLs, request parameters, and client IP addresses to give your website more precise protection.

  Operation: On the **Policies** page, click a policy name. On the displayed page, click the **Precise Protection** area, add a rule, and set the **Protective Action** to **Block**. For details, see **Configuring a Precise Protection Rule**.

- **Blacklist and Whitelist (block mode)**: You can block IP addresses and IP address ranges irrelevant to your services with ease.

  Operation: On the **Policies** page, click a policy name. On the displayed page, click the **Blacklist and Whitelist** area, add a rule, and set the **Protective Action** to **Block**. For more details, see **Configuring an IP Blacklist or Whitelist Rule**..
- **Geolocation Access Control (block mode)**: You can configure geolocation access control rules to block or allow requests from a specific location. If there are too many malicious requests from a specific region, this type of rule helps block every single request from the region. A geolocation access control rule allows you to allow or block requests from IP addresses from specified countries or regions.

  Operation: On the **Policies** page, click a policy name. On the displayed page, click the **Geolocation Access Control** area, add a rule, and set the **Protective Action** to **Block**. For more details, see **Configuring a Geolocation Access Control Rule**.

My business is often harassed by crawlers or faces data leakage and tampering risks.

To meet your requirements, the following protection configurations are recommended:

- **Web Tamper Protection**:WAF caches the pages you want to protect and returns cached pages to visitors so that your website visitors will view right pages all the time even if a web page was tampered with.

  Operation: On the **Policies** page, click the policy name. On the displayed page, click the **Web Tamper Protection** area, add a rule, and complete related settings. For details, see **Configuring a Web Tamper Protection Rule**.
- **Information Leakage Prevention**: Helps mask sensitive information, such as ID numbers, phone numbers, and email addresses, on web pages when those pages are returned to visitors.

  Operation: On the **Policies** page, click the policy name. On the displayed page, click the **Information Leakage Prevention** area, add a rule, and complete related settings. For details, see **Configuring an Information Leakage Prevention Rule**.
- **Anti-Crawler Protection**
  - **Feature Library**: You can allow access requests from legitimate crawlers (such as Googlebot and Baiduspider) but block crawler attacks from most scripts and automation programs.
  - **JavaScript**: If you enable this protection, WAF checks JavaScript. You can also add custom rules to prevent JavaScript crawlers.

  Operation: On the **Policies** page, click the policy name. On the displayed page, click the **Anti-Crawler** area, add a rule, and complete related settings. For details, see **Configuring Anti-Crawler Rules**.

3 Mitigating Web Security Vulnerabilities

3.1 Java Spring Framework Remote Code Execution Vulnerability

Spring Framework is a lightweight open-source application framework for developing enterprise Java applications. A remote code execution (RCE) vulnerability was disclosed in the Spring framework and classified as critical. This vulnerability can be exploited to attack Java applications running on JDK 9 or later versions.

Vulnerability Name

Zero-Day RCE Vulnerability in the Spring Framework

Affected Versions

- JDK 9 or later
- Applications developed using the Spring Framework or derived framework

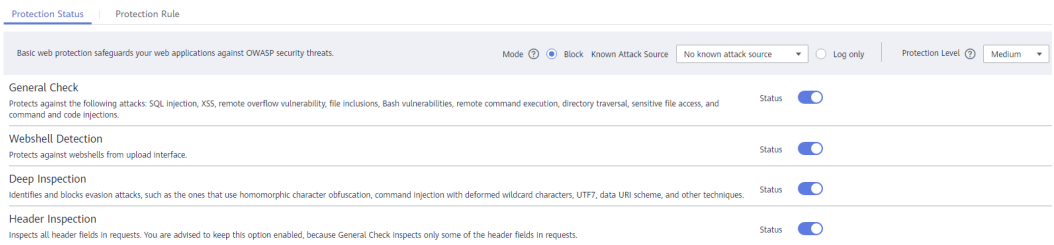
Mitigation

Step 1 **Buy WAF**.

Step 2 Add the website domain name to WAF and connect it to WAF. For details, see **Adding a Domain Name to WAF**.

Step 3 In the **Basic Web Protection** configuration area, set **Mode** to **Block**. For details, see **Configuring Basic Web Protection Rules**.

Figure 3-1 Basic Web Protection



NOTICE

There are two types of malicious payload in this vulnerability. Whether to enable **Header Inspection** depends on the type of payloads in your services.

- Type 1: Malicious payloads are included in submitted parameters. In this situation, **Header Inspection** can be disabled.
- Type 2: Malicious payloads are included in a custom header field. In this situation, **Header Inspection** must be enabled to block attacks.

Type 2 malicious payloads depend on Type 1 malicious payloads so whether to enable **Header Inspection** is determined by your service requirements.

----End

3.2 Apache Dubbo Deserialization Vulnerability

On February 10, 2020, Apache Dubbo officially released the CVE-2019-17564 vulnerability notice, and the vulnerability severity is medium. Unsafe deserialization occurs within a Dubbo application which has HTTP remoting enabled. An attacker may submit a POST request with a Java object in it to completely compromise a Provider instance of Apache Dubbo, if this instance enables HTTP. Now, Huawei Cloud WAF provides protection against this vulnerability.

Affected Versions

This vulnerability affects Apache Dubbo 2.7.0 to 2.7.4, 2.6.0 to 2.6.7, and all 2.5.*x* versions.

Mitigation Version

Apache Dubbo 2.7.5

Solutions

Upgrade Apache Dubbo to version 2.7.5.

If a quick upgrade is not possible or you want to defend against more vulnerabilities, use Huawei Cloud WAF. The procedure is as follows:

Step 1 **Buy WAF**.

Step 2 Add the website domain name to WAF and connect it to WAF. For details, see **Adding a Domain Name**.

Step 3 In the **Basic Web Protection** configuration area, set **Mode** to **Block**. For details, see **Configuring Basic Web Protection Rules**.

----End

3.3 DoS Vulnerability in the Open-Source Component Fastjson

On September 3, 2019, the Huawei Cloud security team detected a DoS vulnerability in multiple versions of the widely used open-source component Fastjson. An attacker can exploit this vulnerability to construct malicious requests and send them to the server that uses Fastjson. As a result, the memory and CPU of the server are used up, and the server breaks down, causing service breakdown. Huawei Cloud WAF provides protection against this vulnerability.

Affected Versions

Versions earlier than Fastjson 1.2.60

Mitigation Version

Fastjson 1.2.60

Official Solution

Upgrade the open-source component Fastjson to 1.2.60.

Mitigation

WAF can detect and defend against this vulnerability. The procedure is as follows:

Step 1 **Buy WAF**.

Step 2 Add the website domain name to WAF and connect it to WAF. For details, see **Adding a Domain Name**.

Step 3 In the **Basic Web Protection** configuration area, set **Mode** to **Block**. For details, see **Enabling Basic Web Protection**.

----End

3.4 Remote Code Execution Vulnerability of Fastjson

On July 12, 2019, the Huawei Cloud Emergency Response Center detected that the open-source component Fastjson had a remote code execution vulnerability. This vulnerability is an extension of the deserialization vulnerability of Fastjson 1.2.24 detected in 2017 and can be directly used to obtain server permissions, causing serious damage.

Affected Versions

Versions earlier than Fastjson 1.2.51

Mitigation Version

Fastjson 1.2.51 or later

Official Solution

Upgrade Fastjson to 1.2.51 or the latest 1.2.58 version.

Mitigation

The built-in protection rules of Huawei Cloud WAF can defend against this vulnerability. The procedure is as follows:

Step 1 **Buy WAF**.

Step 2 Add the website domain name to WAF and connect it to WAF. For details, see **Adding a Domain Name**.

Step 3 In the **Basic Web Protection** configuration area, set **Mode** to **Block**. For details, see **Enabling Basic Web Protection**.

----End

3.5 Oracle WebLogic wls9-async Deserialization Remote Command Execution Vulnerability (CNVD-C-2019-48814)

On April 17, 2019, Huawei Cloud Emergency Response Center found that China's National Vulnerability Database (CNVD) released a security notice on Oracle WebLogic wls9-async component. It revealed that the Oracle WebLogic wls9-async component had a deserialization vulnerability. Unauthorized remote attackers can use this vulnerability to implement remote code execution and gain server permissions.

Vulnerability ID

CNVD-C-2019-48814

Vulnerability Name

Oracle WebLogic wls9-async Deserialization Remote Command Execution Vulnerability

Vulnerability Description

The WebLogic wls9-async component has a defect. The website built on the WebLogic Server has security risks. Attackers can construct HTTP requests to obtain the permission of the target server and execute arbitrary code remotely without authorization.

Affected Products

- Oracle WebLogic Server 10.X
- Oracle WebLogic Server 12.1.3

Official Solution

The patch for fixing this vulnerability has not been released.

Mitigation

Configure precise protection rules to restrict access from the URLs whose prefixes are **/_async/** or **/wls-wsat/** by referring to **Figure 3-2** and **Figure 3-3** and block remote code execution requests initiated by exploiting this vulnerability. For details, see **Configuring a Precise Protection Rule**.

Figure 3-2 async configuration

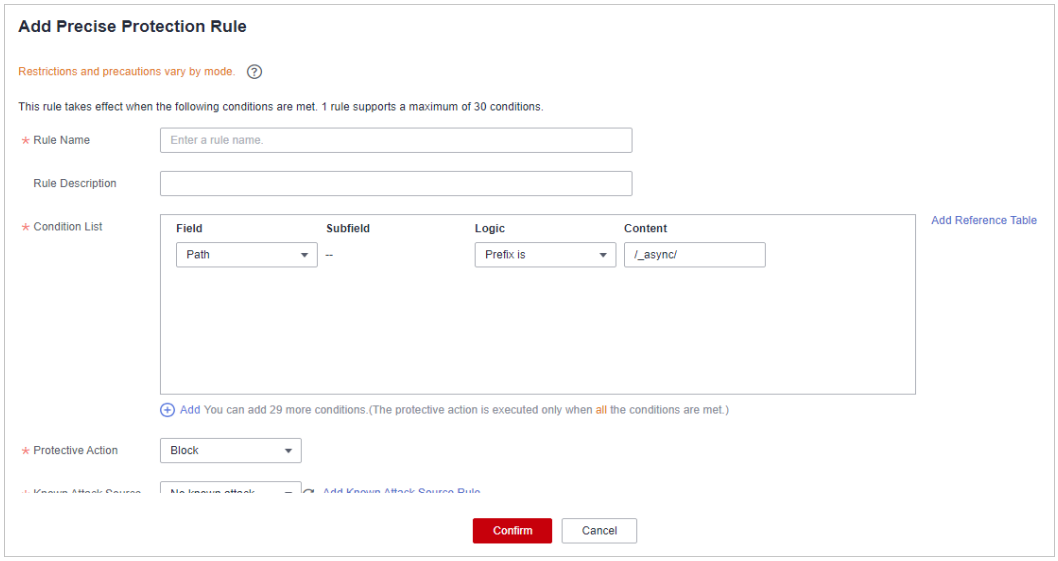
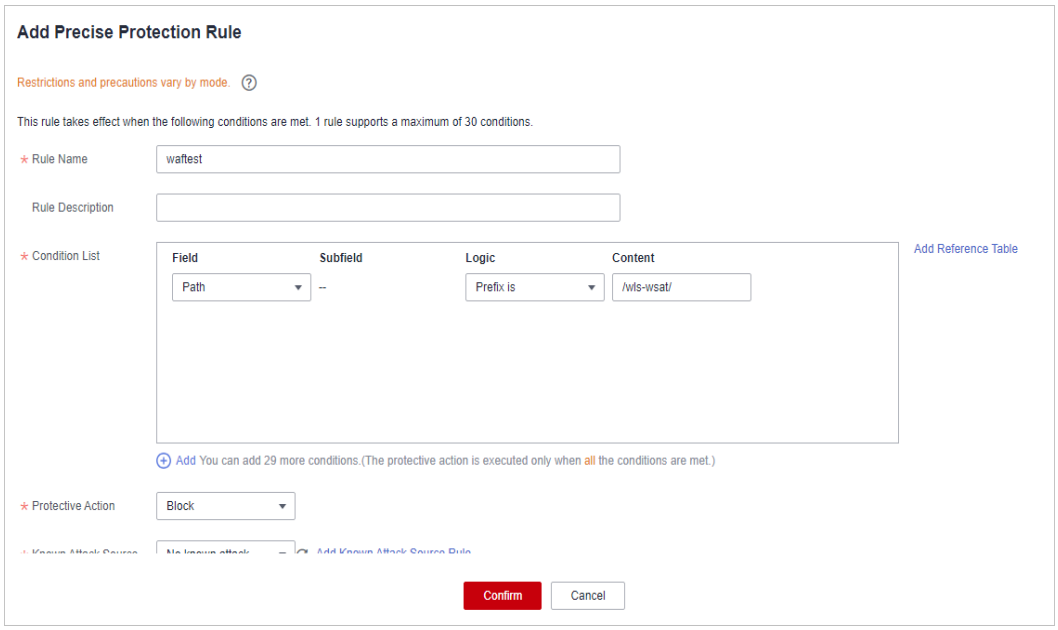


Figure 3-3 wls-wsat configuration



4 Configuring Protection Policies

4.1 Configuring Basic Web Protection

This topic describes best practices in basic web protection.

Application Scenarios

Web Application Firewall (WAF) keeps web services stable and secure. It examines all HTTP and HTTPS requests to detect and block the following attacks: Structured Query Language (SQL) injection, cross-site scripting (XSS), web shells, command and code injections, file inclusion, sensitive file access, third-party vulnerability exploits, Challenge Collapsar (CC) attacks, malicious crawlers, and cross-site request forgery (CSRF).

Protection Policy

Step 1 **Log in to the management console**.

Step 2 Click  in the upper left corner of the management console and select a region or project.

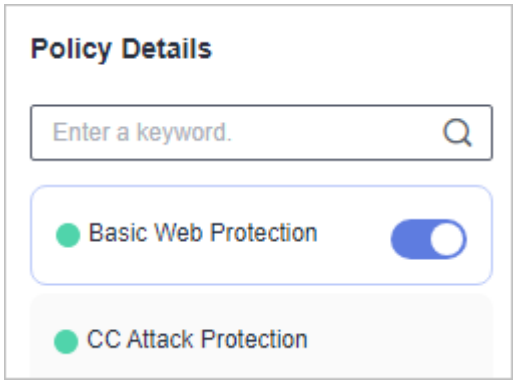
Step 3 Click  in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Website Settings**.

Step 5 In the **Policy** column of the row containing the domain name, click the number to go to the **Policies** page.

Step 6 In the **Basic Web Protection** configuration area, change its status if needed.

Figure 4-1 Basic Web Protection configuration area

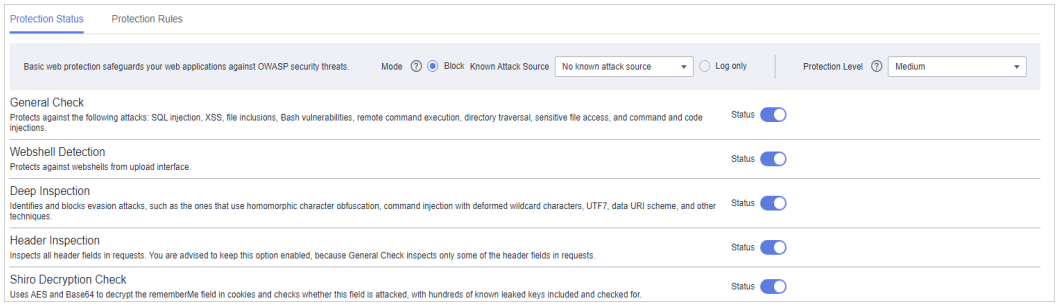


By default, **Basic Web Protection** is enabled and its mode is **Log only**.

- Protection status
  - : **Basic Web Protection** is enabled.
  - : **Basic Web Protection** is disabled.
- Protection mode: block or log only
  - **Block**: WAF blocks and logs the detected attacks.
  - **Log only**: WAF only logs the detected attacks.

Step 7 Go to the **Basic Web Protection** page.

Figure 4-2 Basic web protection



- **Protection Level**: high, medium, and low. The default level is **Low**.

Table 4-1 Protection levels

Protection Level	Description
Low	WAF only blocks the requests with obvious attack signatures. If a large number of false alarms are reported, **Low** is recommended.
Medium	The default level is **Medium**, which meets a majority of web protection requirements.

Protection Level	Description
High	WAF blocks the requests with no attack signature but have specific attack patterns. **High** is recommended if you want to block SQL injection, XSS, and command injection attacks.

- Specify the protection type.

  By default, **General Check** is enabled in WAF. You can enable other protection types to meet your business needs.

----End

Usage Instructions

- If you are not familiar with your website's traffic pattern, select the **Log only** mode for one to two weeks and analyze the logs for those days.
  - If no record of blocking legitimate requests is found, switch to the **Block** mode.
  - If legitimate requests are blocked, adjust the protection level or configure global protection whitelist rules to prevent legitimate requests from being blocked.
- Note the following points in your operations:
  - Do not transfer the original SQL statement or JavaScript code in a legitimate HTTP request.
  - Do not use special keywords (such as UPDATE and SET) in a legitimate URL. For example, **https://www.example.com/abc/update/mod.php?set=1**.
  - Use Object Storage Service (OBS) or other secure methods to upload files that exceed 50 MB rather than via a web browser.

Protection Effect

To check whether basic web protection takes effect, enter a test domain name in the address bar of your browser and simulate an SQL injection attack. If WAF blocks the attack, the configuration works. You can view attack event logs on the **Dashboard** page. **Figure 4-4** shows an example.

Figure 4-3 Blocking SQL attacks

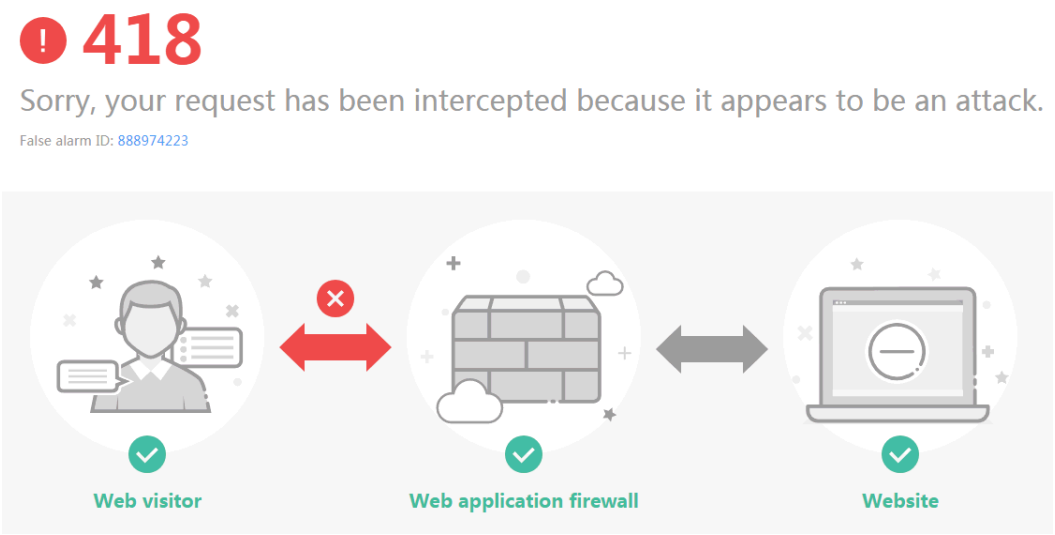
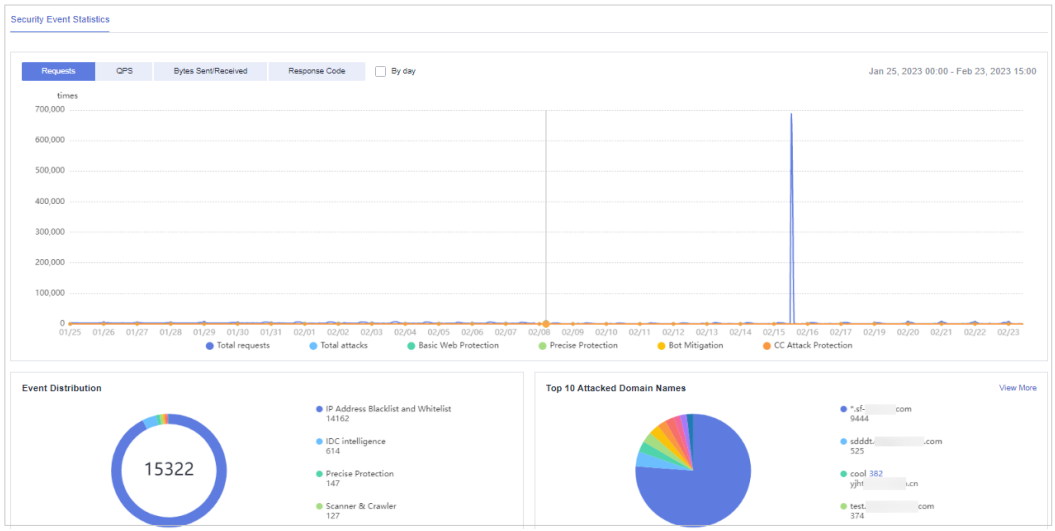


Figure 4-4 Security Event Statistics



You can also view protection logs generated in yesterday, today, past 3 days, past 7 days, 30 days, or user-defined time range on the **Events** page. Alternatively, you can go to the event list, locate the row containing the specific event, and click **Details** in the **Operation** column

4.2 Configuring CC Attack Protection

4.2.1 Overview

This section guides you through configuring IP address-based rate limiting and cookie-based protection rules against Challenge Collapsar (CC) attacks.

How Can We Know Whether a CC Attack Occurs?

If you find that the website processing speed decreases and the network bandwidth usage is high, your website may suffer from CC attacks. In this case, check whether the number of access logs or network connections increases significantly. If yes, your website is suffering from CC attacks. Then you can configure a protection rule to protect your website from CC attacks.

NOTE

- WAF protects application-layer traffic against DoS attacks, such as HTTP GET attacks.
- WAF does not protect your website at or below layer 4 against DDoS traffic, such as ACK Flood and UDP flood attacks. Anti-DDoS and Advanced Anti-DDoS (AAD) are recommended to defend against such attacks.

4.2.2 Configuring CC Attack Protection for Common Scenarios

This topic introduces how CC attack protection rules are used in certain scenarios.

Overview

You can have a quick glance to learn how to set WAF protection in the similar scenarios to protect your services.

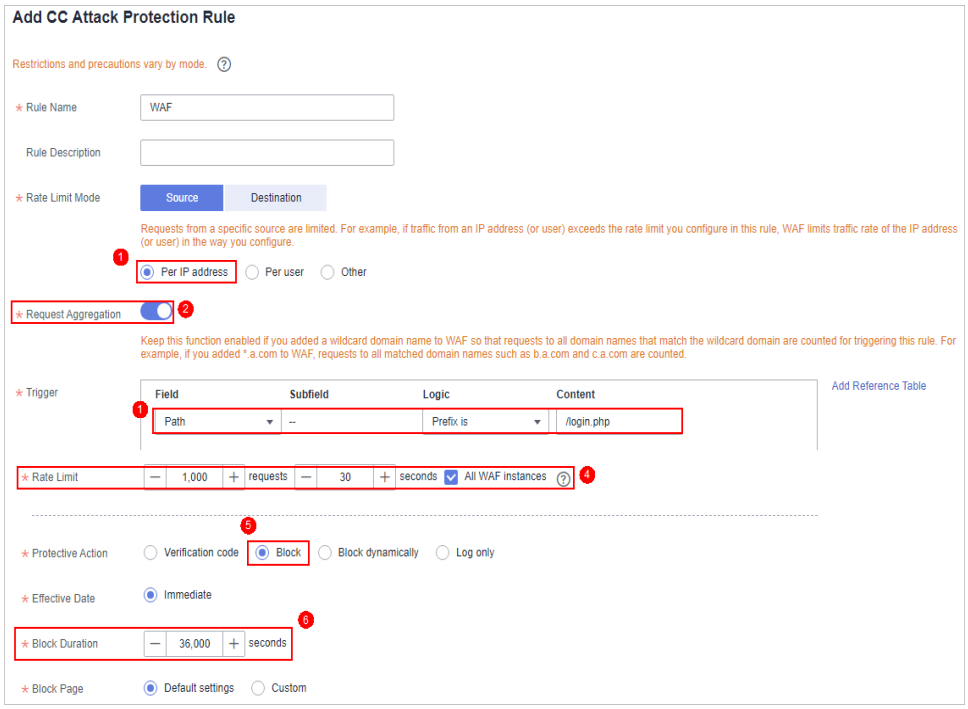
- **Heavy-traffic CC attacks**
- **The request features are malformed or improper.**

Heavy-traffic CC attacks

In large-scale CC attacks, a single zombie server can send far more packets than a common user does. In this scenario, a rate limiting rule is the most effective method to against this type of CC attacks. We recommend IP address-based rate limiting CC attack protection rules. For details, see **IP Address-based Rate Limiting**.

Configuration example: You can configure such a CC rule to mitigate CC attacks. If an IP address accessed any path under the current domain name more than 1000 times within 30 seconds, this rule will block requests from the IP address for 10 hours. This rule can be used as a preventive configuration for common small and medium-sized websites

To get improved and refined protection, you need to adjust rate limit settings and specify an appropriate protective action based on your service requirements. For example, if you need to prevent the login interface from being affected by crazy credential stuffing attacks, use the **prefix is** logical operator and set the matching content to the specific login path, such as **/login.php**.



NOTE

- **Request Aggregation**: Keep this function enabled so that requests to all domain names that match a protected wildcard domain are counted for triggering this rule. For example, if you added *.a.com to WAF, requests to all matched domain names such as b.a.com and c.a.com are counted.
- **All WAF instances**: This parameter is supported only in cloud mode. By default, requests to each WAF instance are counted. If you enable this, WAF will count requests to all your WAF instances for triggering this rule.

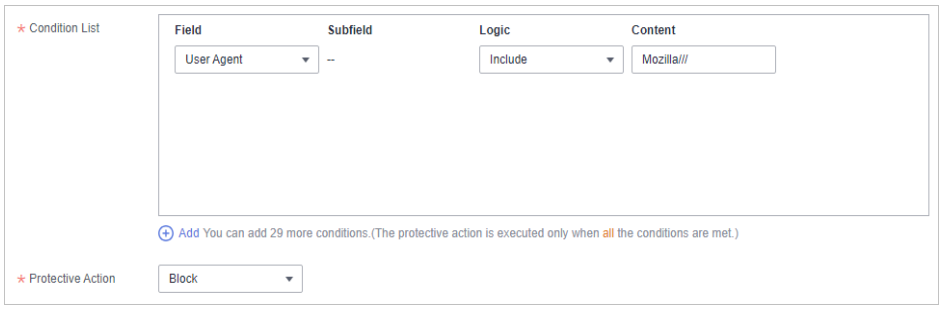
The request features are malformed or improper.

Many CC attack requests are constructed by attackers. After analyzing logs, it is found that these requests have many malformed packet features that do not match normal requests. The following protection rules are recommended to defend against requests having common malformed packets:

The following protection configurations are implemented through precise protection rules. For details, see **Configuring a Precise Protection Rule**.

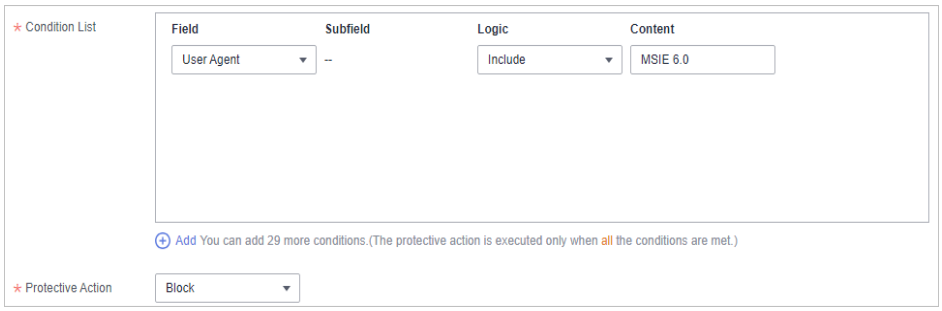
- Abnormal or malformed User-Agent: Invalid User-Agent (for example, Mozilla///), improper User-Agent (for example, www.example.com), and User Agent containing automation tool features If a request matches the features, it will be blocked.

  Configuration example: Blocking requests with User Agent containing Mozilla///.



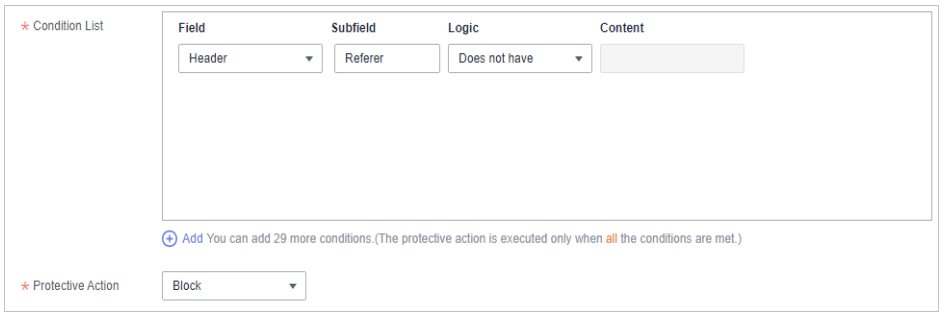
- Improper User-Agent: For example, for HTML5 pages promoted by WeChat, normal users should initiate access through WeChat. It obviously does not make sense if the request User-Agent comes from a Windows desktop browser (for example, MSIE 6.0). If a request matches the features, it will be blocked.

  Configuration example: Blocking requests with User Agent containing MSIE 6.0.



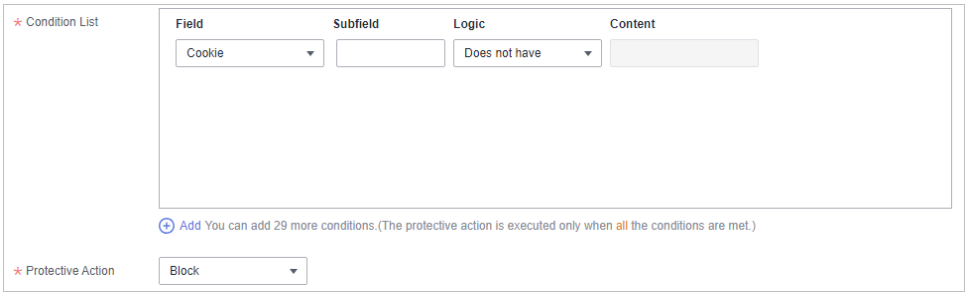
- Abnormal Referer: For example, if a request does not contain a Referer or the Referer is fixed and comes from an unauthorized website, the request can be blocked (except when the website home page is accessed or the page is accessed for the first time). For URLs that can be accessed only through an internal address, you can analyze the behavior by checking the Referer field and determine whether to block the URLs.

  Configuration example: Blocking requests having no Referer



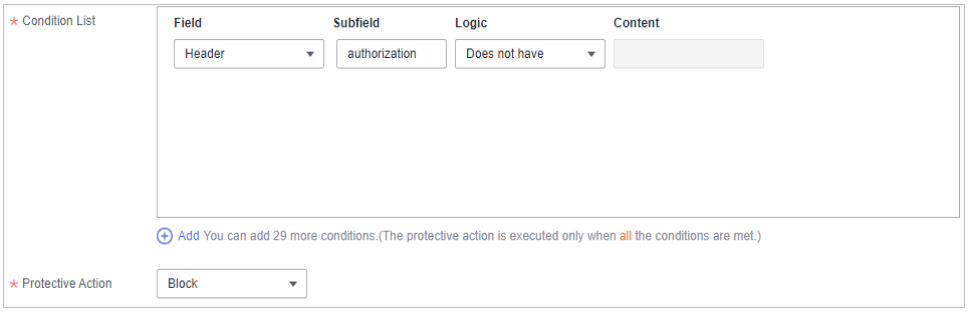
- Abnormal Cookie: A normal request usually carries cookies that belong to the service set of the website (except when the user accesses the page for the first time). In most cases, CC attack packets do not carry any cookie. So you can block access requests having no cookies.

  Configuration example: Blocking requests having no cookies



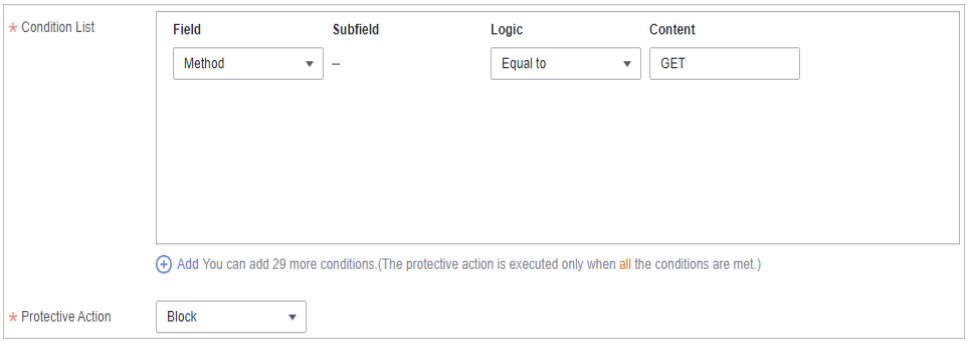
- Lack of some HTTP headers. For example, a common user will have the authentication header required by some services carried in the request, but attack packets do not.

  Configuration example: Blocking requests with no authorization headers included.



- Incorrect request method: For example, if an interface designed for only POST requests is attacked by a large number of GET requests, you can directly block GET requests.

  Configuration example: Blocking GET requests



4.2.3 IP Address-based Rate Limiting

If no proxy is used between WAF and web visitors, limiting source IP addresses is an effective way to detect attacks. IP address-based rate limiting policies are recommended.

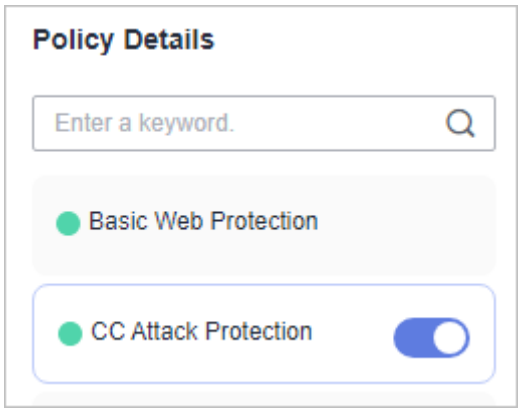
Use Cases

Attackers use several hosts to continuously send HTTP POST requests to website **www.example.com**. Those malicious requests will use up website resources, such as the website connections and bandwidth. As a result, the website fails to respond to normal requests and its competitiveness decreases sharply.

Protective Measures

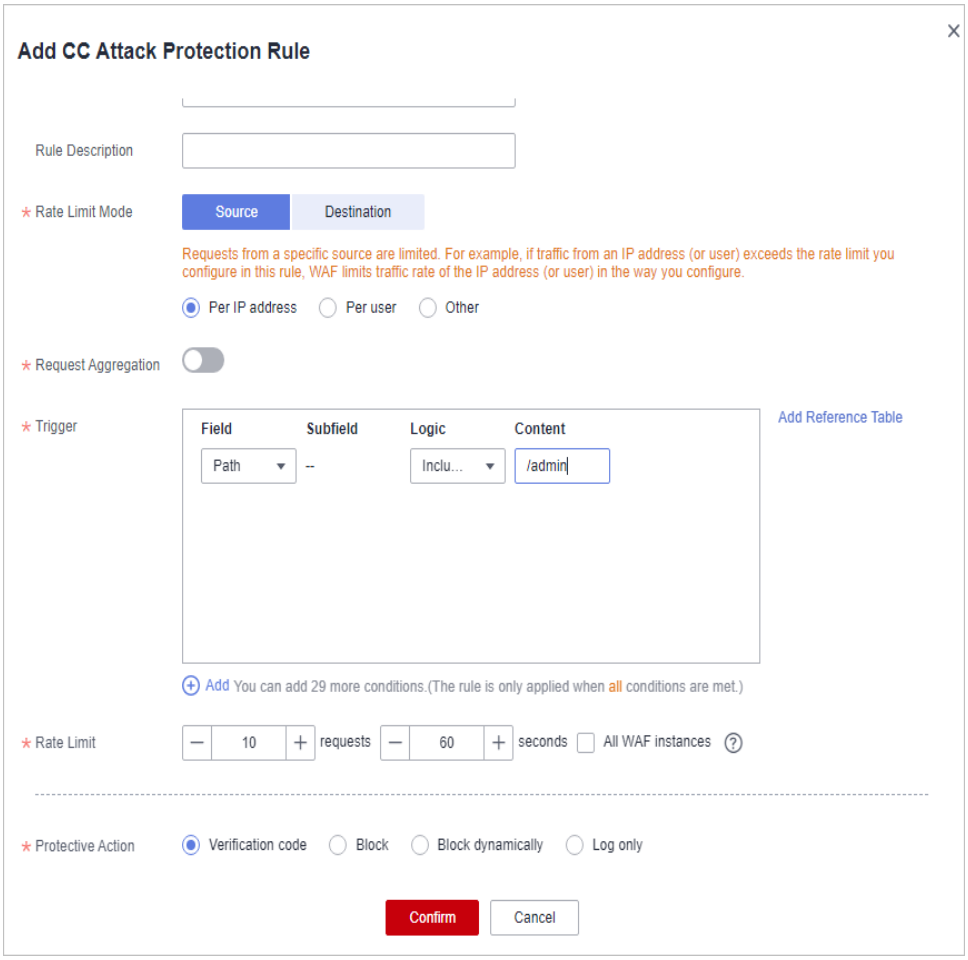
1. Based on the access statistics, check whether a large number of requests were sent from a specific IP address. If yes, it is likely that the website was hit by CC attacks.
2. Log in to the management console and route website traffic to WAF. For more details, see **Adding a Domain Name to WAF**.
3. In the **Policy** column of the row containing the target domain name, click the number of enabled protection rules. On the displayed **Policies** page, keep the **Status** toggle on ( ) for **CC Attack Protection**.

   Figure 4-5 CC Attack Protection configuration area

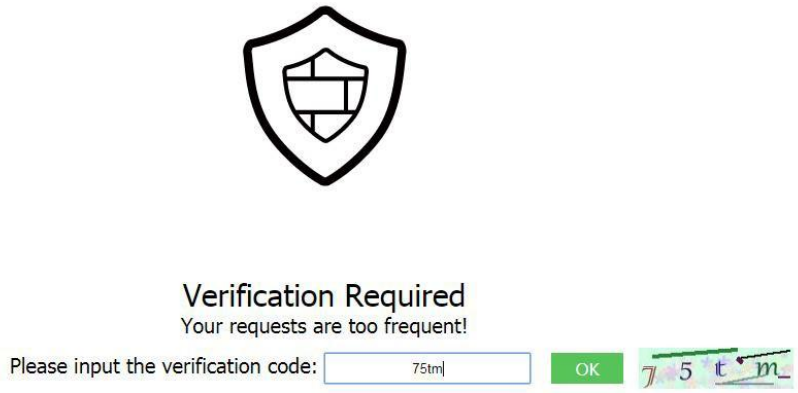
4. Then, add a CC attack protection rule to limit the rate of request traffic destined for the domain name. Set **Rate Limit Mode** to **Per IP address**, **Rate Limit** based on your service features, and **Protective Action** to **Verification code** to prevent blocking legitimate users. **Figure 4-6** shows the settings.

Figure 4-6 Per IP address



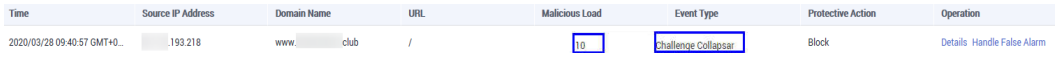
- **Rate Limit Mode**: Select **Source** and then **Per IP address** to distinguish a single web visitor based on IP addresses.
- **Rate Limit**: Number of requests allowed from a website visitor in the rate limiting period. The visitor's access request is denied if the limit is reached.
- **Protective Action**: To prevent legitimate requests from being blocked, select **Verification code**.
  - **Verification code**: A verification code is required if your website visitor's requests reaches **Rate Limit** you configured. WAF allows requests that trigger the rule as long as the website visitors complete the required verification.
  - **Block**: Requests are blocked if the number of requests exceeds the configured rate limit.
  - **Log only**: Requests are logged only but not blocked if the number of requests exceeds the configured rate limit.

If the number of access requests exceeds the configured rate limit, the visitors are required to enter a verification code to continue the access.



Go to the **Events** page and view details about attack events.

Figure 4-7 Querying CC attack event logs



4.2.4 Cookie-based CC Attack Protection

In some cases, it may be difficult for WAF to obtain real IP addresses of website visitors. For example, if a website uses proxies that do not use the **X-Forwarded-For** HTTP header field, WAF is unable to obtain the real access IP addresses. In this situation, the cookie field should be configured to identify visitors and **All WAF instances** should be enabled for precise user-based rate limiting.

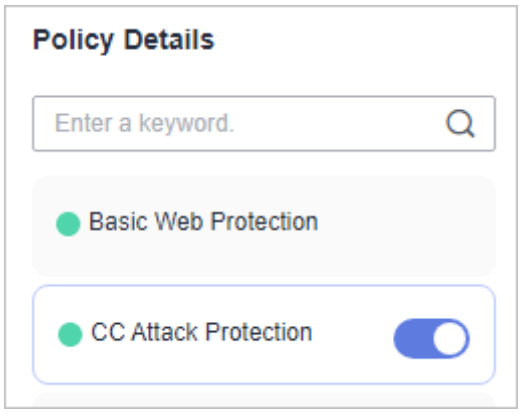
Use Cases

Attackers may control several hosts and disguise as normal visitors to continuously send HTTP POST requests to website **www.example.com** through the same IP address or many different IP addresses. As a result, the website may respond slowly or even fails to respond to normal requests as the attackers exhausted website resources like connections and bandwidth.

Protective Measures

1. Based on the access statistics, check whether a large number of requests are sent from a specific IP address. If yes, it is likely that the website is hit by CC attacks.
2. Log in to the management console and route website traffic to WAF. For more details, see **Adding a Domain Name to WAF**.
3. In the **Policy** column of the row containing the target domain name, click the number of enabled protection rules. On the displayed **Policies** page, keep the **Status** toggle on ( ) for **CC Attack Protection**.

Figure 4-8 CC Attack Protection configuration area

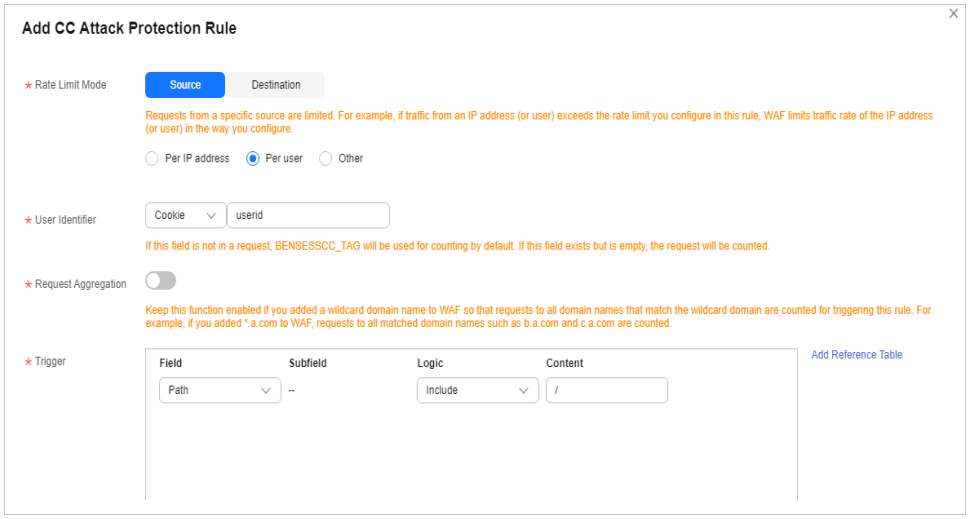


4. Add a CC attack protection rule. Set **Rate Limit Mode** to **Per user** and enter the user identifier, which is the variable in the cookie field. To identify visitors more effectively, use **sessionid** or **token**.

   NOTE

   With a CC attack protection rule, you can configure **Protective Action** to **Block** and specify a block duration. Then, once an attack is blocked, the attacker will be blocked until the block duration expires. These settings are recommended if your applications have high security requirements.

Figure 4-9 Add CC Attack Protection Rule



- **Rate Limit Mode**: Select **Source** and then **Per user** to distinguish a single web visitor based on cookies.
- **User Identifier**: To identify visitors more effectively, use **sessionid** or **token**.
- **Rate Limit**: Number of requests allowed from a web visitor in the rate limiting period. The visitor's access request is denied if the limit is reached.
- **All WAF instances**: Requests to on one or more WAF instances will be counted together according to the rate limit mode you select. By default, requests to each WAF instance are counted. If you enable this, WAF will

count requests to all your WAF instances for triggering this rule. To enable user-based rate limiting, **Per user** or **Other** (**Referer** must be configured) instead of **Per IP address** must be selected for **Rate Limit Mode**. This is because IP address-based rate limiting cannot limit the access rate of a specific user. However, in user-based rate limiting, requests may be forwarded to one or more WAF instances. Therefore, **All WAF instances** must be enabled for triggering the rule precisely.

- **Protective Action**: Select **Block**. Then specify **Block Duration**. Once an attack is blocked, the attacker will be blocked until the block duration expires. These settings are recommended if your applications have high security requirements.
  - **Verification code**: A verification code is required if your website visitor's requests reaches **Rate Limit** you configured. WAF allows requests that trigger the rule as long as the website visitors complete the required verification.
  - **Block**: Requests are blocked if the number of requests exceeds the configured rate limit.
  - **Log only**: Requests are logged only but not blocked if the number of requests exceeds the configured rate limit.
- **Block Page**: Select **Default settings** or **Custom**.

4.2.5 Restricting Malicious Requests in Promotions by Using Cookies and HWWAFSESID

This topic describes how to configure cookies and HWWAFSESID fields in CC attack protection rules to restrict malicious requests in promotions.

Application Scenarios

- **Scenario 1**: To steal extra bonus (such as goods in promotions or downloads), a malicious actor may use the same account to send requests to a website by changing IP addresses or terminals.

  Protective measures: **Using Cookies (or User IDs) to Configure a Path-based CC Attack Protection Rule**
- **Scenario 2**: To steal extra bonus (such as goods in promotions or downloads), a malicious actor may use multiple accounts to send requests to a website through the same PC by frequently changing its IP address.

  Protective measures: **Using HWWAFSESID to Configure a CC Attack Protection Rule**

Using Cookies (or User IDs) to Configure a Path-based CC Attack Protection Rule

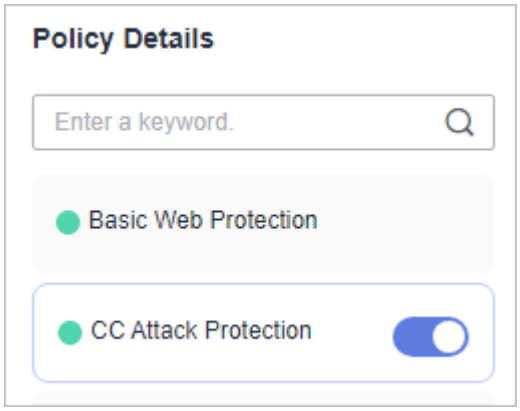
Step 1 Log in to the management console and connect your website to WAF.

- Cloud WAF: **Adding a Domain Name to WAF (Cloud Mode)**
- Dedicated WAF instances: **Adding a Website to WAF (Dedicated Mode)**

Step 2 In the **Policy** column of the row containing the domain name, click the number to go to the **Policies** page.

Step 3 In the **CC Attack Protection** configuration area, toggle **CC Attack Protection** on if needed.

Figure 4-10 CC Attack Protection configuration area

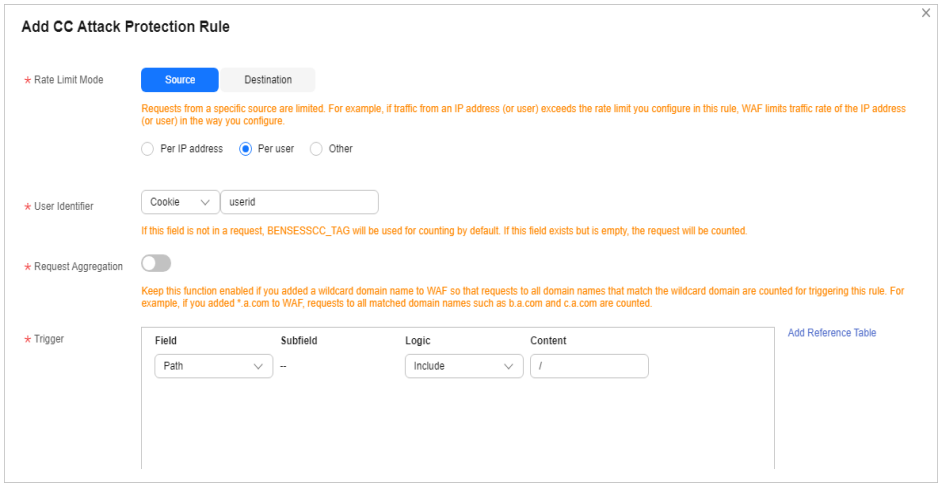


Step 4 In the upper left corner of the **CC Attack Protection** page, click **Add Rule**.

Step 5 Configure a CC attack protection rule using a cookie or user ID to limit traffic to the path. **Figure 4-11** shows an example.

Set the following parameters based on site requirements:

Figure 4-11 Configuring service cookies



Step 6 Click **Confirm**.

----End

Using HWWAFSESID to Configure a CC Attack Protection Rule

Step 1 Log in to the management console and connect your website to WAF.

- Cloud WAF: **Adding a Domain Name to WAF (Cloud Mode)**
- Dedicated WAF instances: **Adding a Website to WAF (Dedicated Mode)**

Step 2 In the **Policy** column of the row containing the domain name, click the number to go to the **Policies** page.

Step 3 In the **CC Attack Protection** configuration area, toggle **CC Attack Protection** on ( ) if needed.

Figure 4-12 CC Attack Protection configuration area

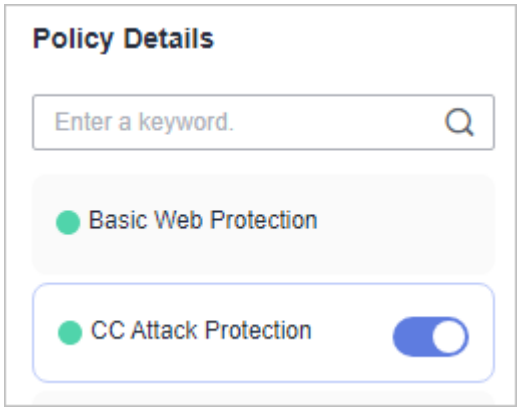
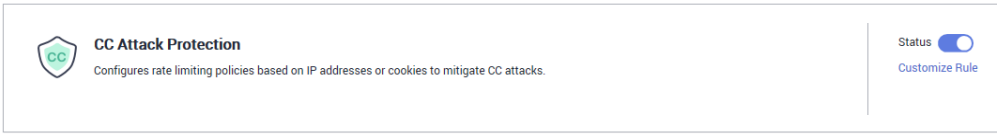


Figure 4-13 CC Attack Protection configuration area

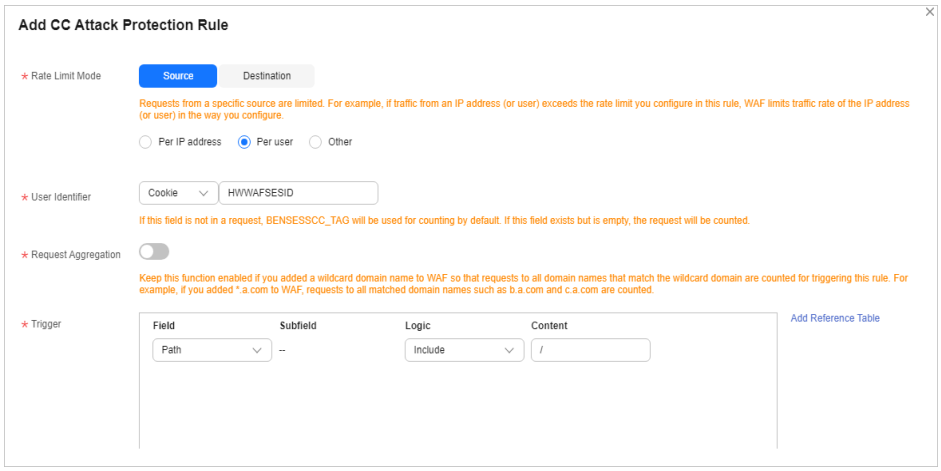


Step 4 In the upper left corner of the **CC Attack Protection** page, click **Add Rule**.

Step 5 Configure a CC attack protection rule using HWWAFSESID to limit traffic to the path. For details, see **Figure 4-14**.

- **User Identifier**: Select **Cookie** and set it to **HWWAFSESID**.
- Other parameters: Set them to meet your service requirements.

Figure 4-14 HWWAFSESID-based rate limiting



Step 6 Click **Confirm**.

----End

4.3 Configuring Anti-Crawler Rules to Prevent Crawler Attacks

Web crawlers make network information collection and query easy, but they also introduce the following negative impacts:

- Web crawlers always consume too much server bandwidth and increase server load as they use specific policies to browser as much information of high value on a website as possible.
- Bad actors may use web crawlers to launch DoS attacks against websites. As a result, websites may fail to provide normal services due to resource exhaustion.
- Bad actors may use web crawlers to steal mission-critical data on your websites, which will damage your economic interests.

WAF provides three anti-crawler policies, bot detection by identifying User-Agent, website anti-crawler by checking browser validity, and CC attack protection by limiting the access frequency, to comprehensively mitigate crawler attacks against your websites.

Prerequisites

The domain name has been connected to WAF.

Enabling Robot Detection to Identify User-Agent

If you enable robot detection, WAF can detect and block threats such as malicious crawlers, scanners, and web shells.

Step 1 **Log in to the management console**.

Step 2 Click  in the upper left corner of the management console and select a region or project.

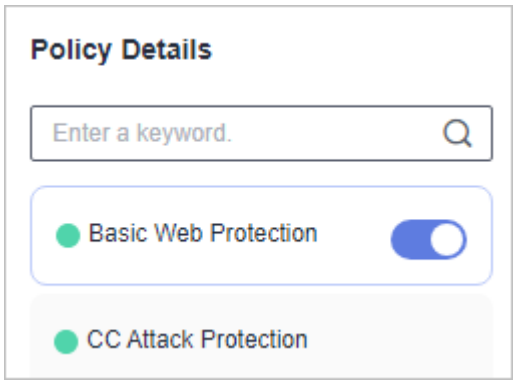
Step 3 Click  in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Website Settings**.

Step 5 In the **Policy** column of the row containing the domain name, click the number to go to the **Policies** page.

Step 6 Ensure that **Basic Web Protection** is enabled (status: ).

Figure 4-15 Basic Web Protection configuration area



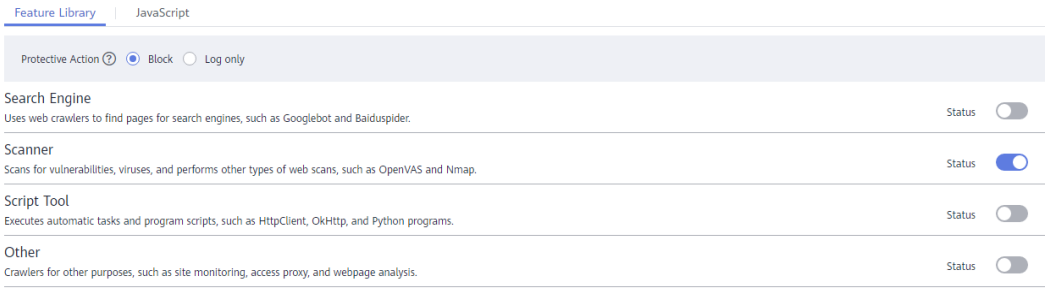
Step 7 On the **Protection Status** page, enable **General Check** and **Webshell Detection**.

Step 8 Click the **Anti-Crawler** configuration area and toggle it on.

- : enabled.
- : disabled.

Step 9 On the **Feature Library** page, enable protection functions based on your business needs.

Figure 4-16 Feature Library



----End

If WAF detects that a malicious crawler or scanner is crawling your website, WAF immediately blocks it and logs the event. You can view the crawler protection logs on the **Events** page.

Enabling Anti-Crawler Protection to Verify Browser Validity

If you enable anti-crawler protection, WAF dynamically analyzes website service models and accurately identifies crawler behavior based on data risk control and bot identification approaches.

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click ☰ in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Website Settings**.

Step 5 In the **Policy** column of the row containing the domain name, click the number to go to the **Policies** page.

Step 6 Click the **Anti-Crawler** configuration area and toggle it on.

-  : enabled.
-  : disabled.

Step 7 Select the **JavaScript** tab and change **Status** if needed.

JavaScript anti-crawler is disabled by default. To enable it, click  and then click **Confirm** in the displayed dialog box to toggle on  .

Protective Action: **Block**, **Verification code**, and **Log only**.

NOTE

Verification code: If the JavaScript challenge fails, a verification code is required.

NOTICE

- Cookies must be enabled and JavaScript supported by any browser used to access a website protected by anti-crawler protection rules.
- If your service is connected to CDN, exercise caution when using the JS anti-crawler function.

  CDN caching may impact JS anti-crawler performance and page accessibility.

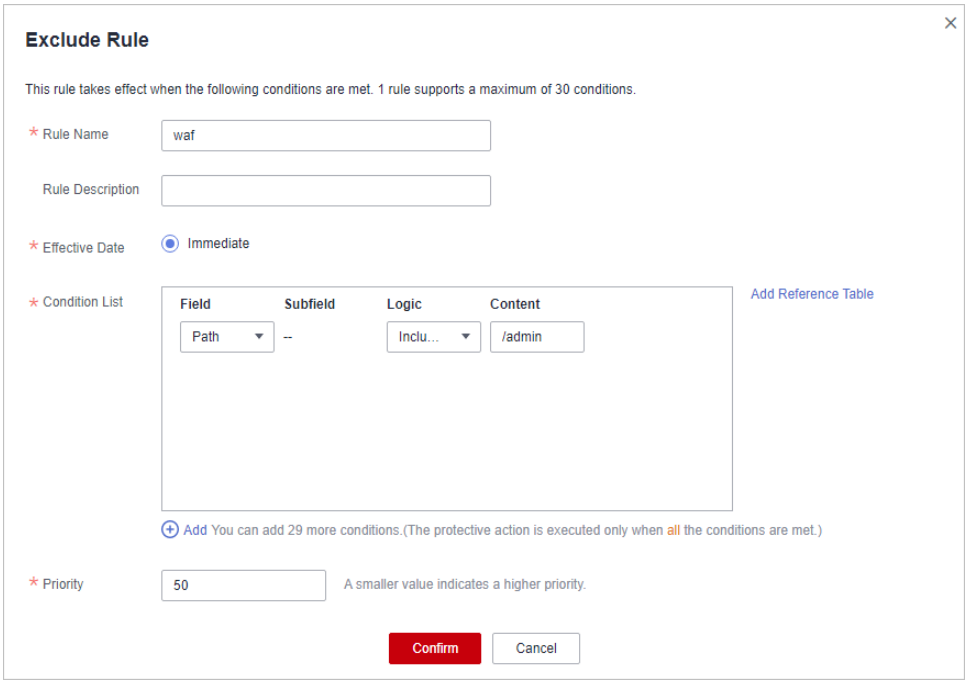
Step 8 Configure a JavaScript-based anti-crawler rule by referring to **Table 4-2**.

Two protective actions are provided: **Protect all requests** and **Protect specified requests**.

- To protect all requests except requests that hit a specified rule

  Set **Protection Mode** to **Protect all requests**. Then, click **Exclude Rule**, configure the request exclusion rule, and click **Confirm**.

Figure 4-17 Exclude Rule



- To protect a specified request only

  Set **Protection Mode** to **Protect specified requests**, click **Add Rule**, configure the request rule, and click **Confirm**.

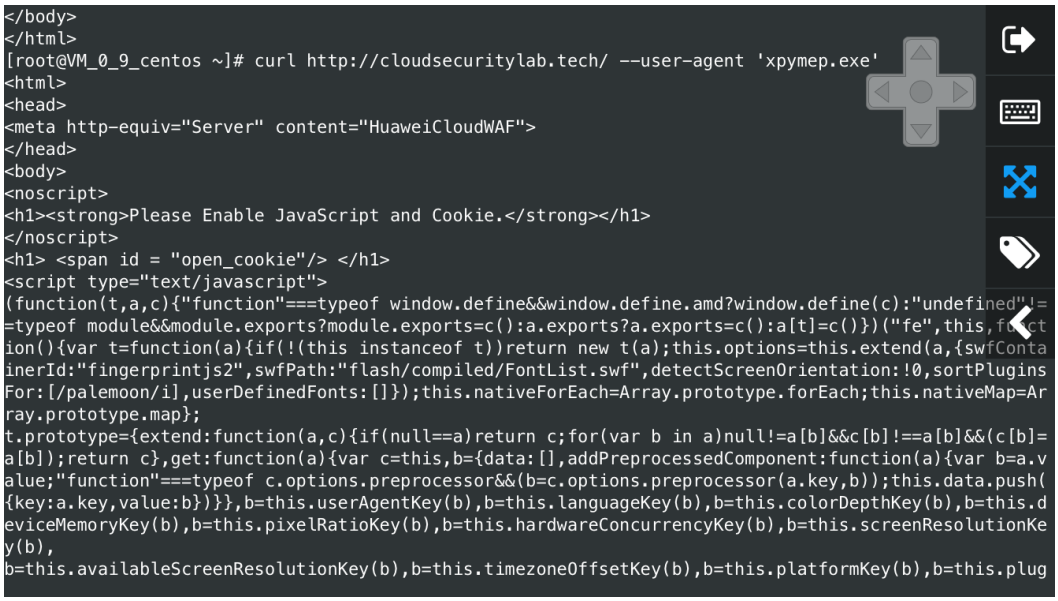
Table 4-2 Parameters of a JavaScript-based anti-crawler protection rule

Parameter	Description	Example Value
Rule Name	Name of the rule	waf
Rule Description	A brief description of the rule. This parameter is optional.	-
Effective Date	Time the rule takes effect.	Immediate

Parameter	Description	Example Value
Condition List	Parameters for configuring a condition are as follows: • **Field**: Select the field you want to protect from the drop-down list. Currently, only **Path** and **User Agent** are included. • **Subfield** • **Logic**: Select a logical relationship from the drop-down list. **NOTE** If you set **Logic** to **Include any value**, **Exclude any value**, **Equal to any value**, **Not equal to any value**, **Prefix is any value**, **Prefix is not any of them**, **Suffix is any value**, or **Suffix is not any of them**, you need to select a reference table. • **Content**: Enter or select the content that matches the condition.	**Path Include /admin**
Priority	Rule priority. If you have added multiple rules, rules are matched by priority. The smaller the value you set, the higher the priority.	5

----End

If you enable anti-crawler, web visitors can only access web pages through a browser.



Configuring CC Attack Protection to Limit Access Frequency

A CC attack protection rule uses a specific IP address, cookie, or referer to limit the access to a specific path (URL), mitigating the impact of CC attacks on web services.

Step 1 Log in to the management console.

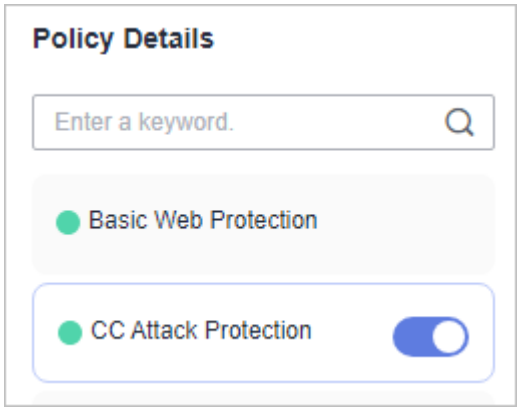
Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Website Settings**.

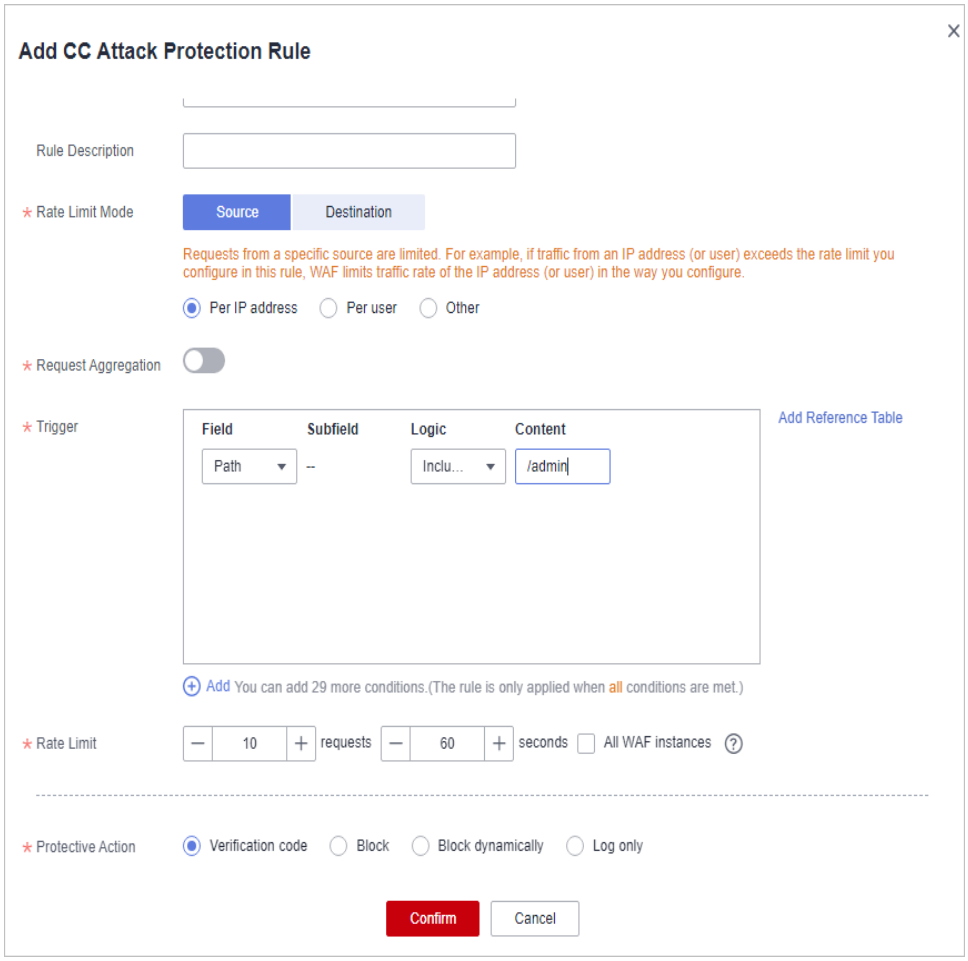
Step 5 In the **Policy** column of the row containing the target domain name, click the number of enabled protection rules. On the displayed **Policies** page, keep the **Status** toggle on ( ) for **CC Attack Protection**.

Figure 4-18 CC Attack Protection configuration area

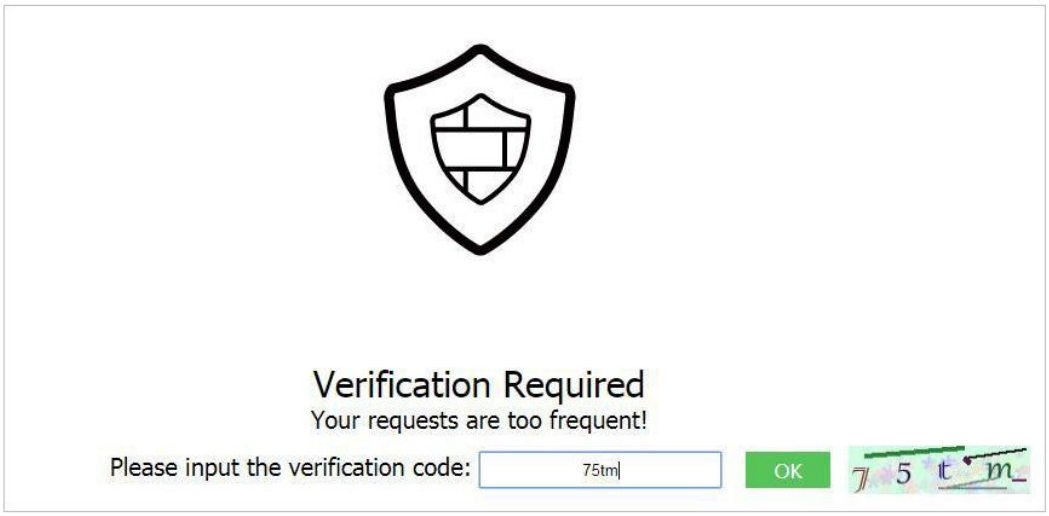


Step 6 In the upper left corner above the **CC Attack Protection** rule list, click **Add Rule**. The following uses IP address-based rate limiting and human-machine verification as examples to describe how to add an IP address-based rate limiting rule, as shown in **Figure 4-19**.

Figure 4-19 Per IP address



If the number of access requests exceeds the configured rate limit, the visitors are required to enter a verification code to continue the access.



----End

4.4 Handling False Alarms to Get Improved Basic Web Protection

After you connect your website to Web Application Firewall (WAF) and enable basic web protection, WAF detects and blocks requests that match the rules you configured. If a normal request matches a basic web protection rule and is blocked by WAF, you can handle the event as false alarm. In this way, WAF will no longer block the same type of request.

Prerequisites

You can view false alarm events on the **Events** page.

Constraints

An event can only be handled as a false alarm once.

Application scenarios

Sometimes normal service requests may be blocked by WAF. For example, suppose you deploy a web application on an ECS and then add the public domain name associated with that application to WAF. If you enable basic web protection for that application, WAF may block the access requests that match the basic web protection rules. As a result, the website cannot be accessed through its domain name. However, the website can still be accessed through the IP address. In this case, you can handle the false alarms to allow normal access requests to the application.

Impact on the System

- The event will not be displayed on the **Events** page and you will not receive any alarm notifications about the event.
- If an event is handled as a false alarm, the rule hit will be added to the global protection whitelist (formerly false alarm masking) rule list. You can go to the **Policies** page and then switch to the **Global Protection Whitelist (Formerly False Alarm Masking)** page to manage the rule, including querying, disabling, deleting, and modifying the rule.

Procedure

Step 1 **Log in to the management console**.

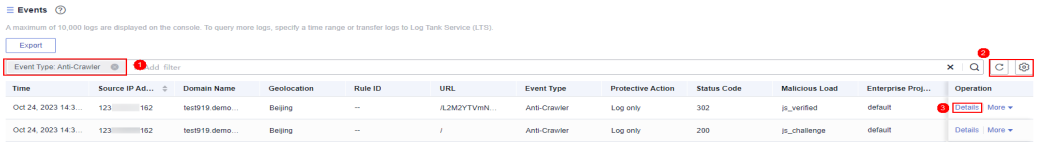
Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Click  in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Events**.

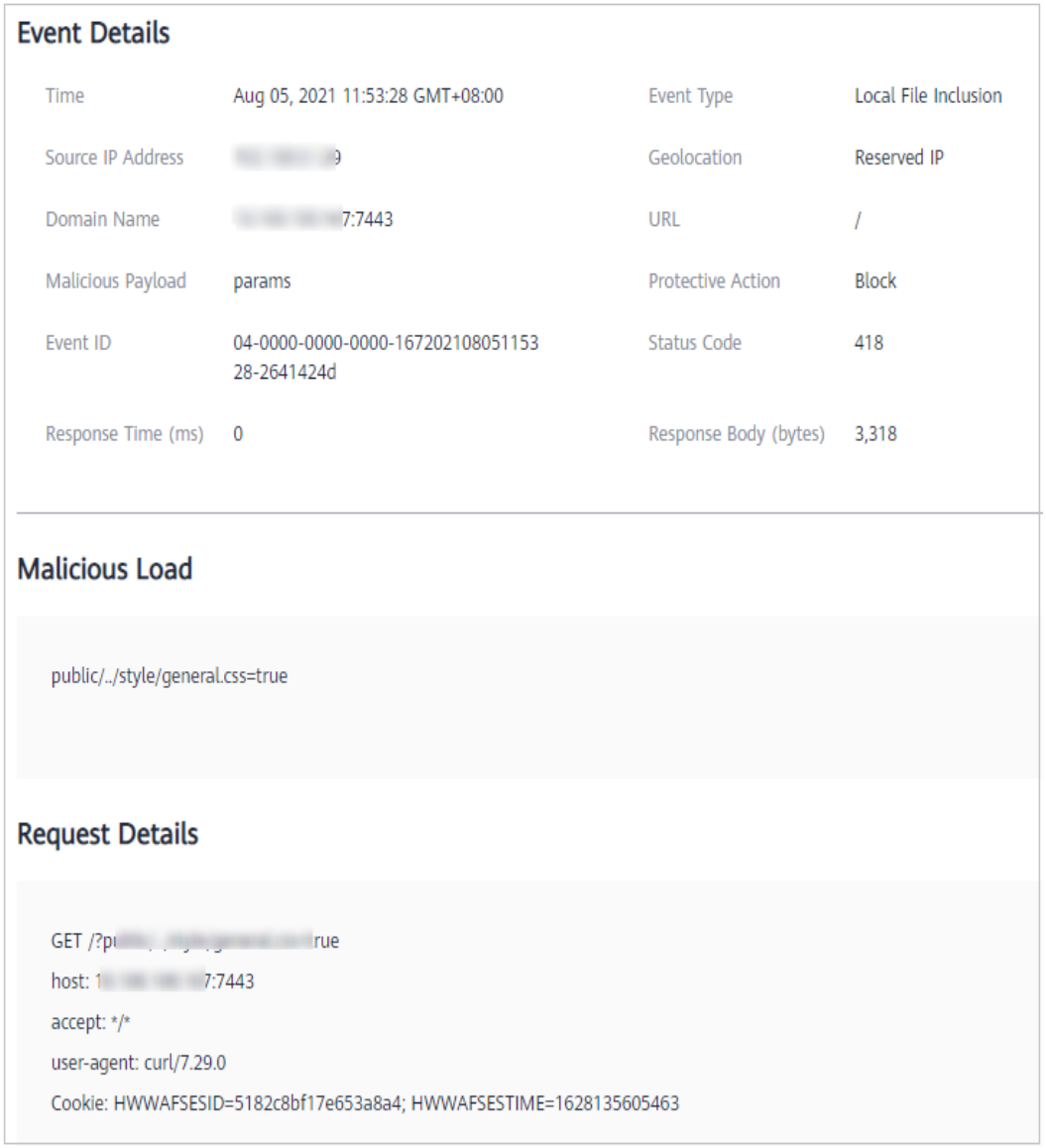
Step 5 In the event list, search for false alarms by protected website, event type, source IP address, and URL.

Figure 4-20 Events



Step 6 In the **Operation** column of an event you consider as a false alarm, click **Details**. On the displayed page, confirm that the event is a false alarm.

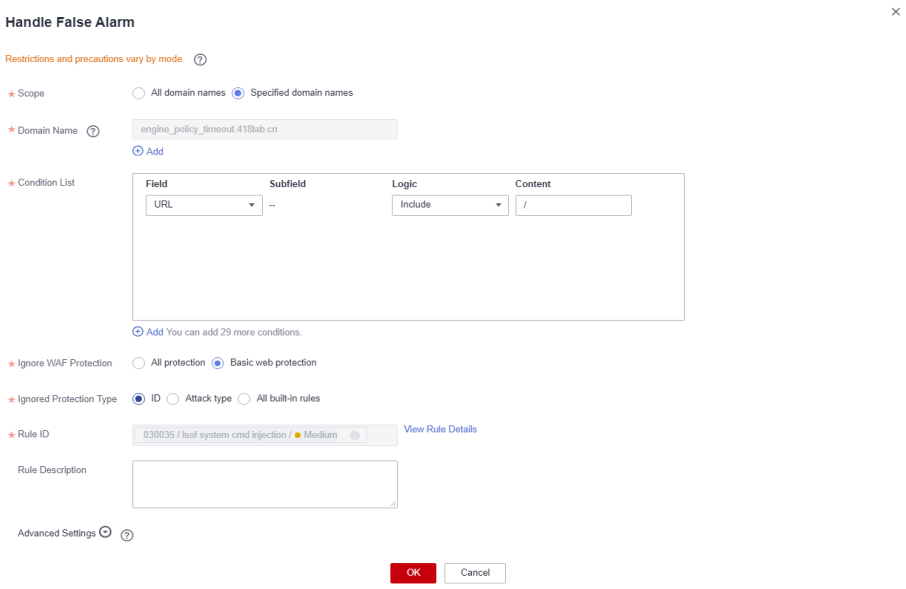
Figure 4-21 Event Details



Step 7 In the row containing the event, click **Handle as False Alarm**.

Step 8 In the displayed dialog box, add a false alarm handling policy.

Figure 4-22 Add Global Protection Whitelist Rule



----End

Verification

A false alarm will be deleted within about a minute after the handling configuration is done. It will no longer be displayed in the event list. You can clear the cache, refresh the browser, and access the page again to verify whether the false alarm was successfully handled. If the requested page responds normally, the configuration takes effect.

Basic Web Protection Check Items

WAF basic web protection defends against common Open Web Application Security Project (OWASP) security threats. WAF uses built-in semantic analysis and regular expression engines for basic web protection to detect and block threats such as malicious scanners, IP addresses, and web shells. You can enable all protection rules in basic web protection or only the ones you want. For details, see **Table 4-3**.

Table 4-3 Protection types

Type	Description
General Check	Defends against attacks such as SQL injections, XSS, remote overflow vulnerabilities, file inclusions, Bash vulnerabilities, remote command execution, directory traversal, sensitive file access, and command/code injections. SQL injection attacks are mainly detected based on semantics. **NOTE** If you enable **General Check**, WAF checks your websites based on the built-in rules.

Type	Description
Webshell Detection	Protects against web shells from upload interface. **NOTE** If you enable **Webshell Detection**, WAF detects web page Trojan horses inserted through the upload interface.
Deep Inspection	Identifies and blocks evasion attacks, such as the ones that use homomorphic character obfuscation, command injection with deformed wildcard characters, UTF7, data URI scheme, and other techniques. **NOTE** If you enable **Deep Inspection**, WAF detects and defends against evasion attacks in depth.
Header Inspection	This function is disabled by default. When it is disabled, General Check will check some of the header fields, such as User-Agent, Content-type, Accept-Language, and Cookie. **NOTE** If you enable this function, WAF checks all header fields in the requests.
Shiro Decryption Check	This function is disabled by default. After this function is enabled, WAF uses AES and Base64 to decrypt the rememberMe field in cookies and checks whether this field is attacked. There are hundreds of known leaked keys included and checked for. **NOTE** If your website uses Shiro 1.2.4 or earlier, or your website uses Shiro 1.2.5 or later but AES is not configured, it is strongly recommended that you enable Shiro decryption detection to prevent attackers from using leaked keys to construct attacks.

Basic Web Protection Levels

WAF provides three basic web protection levels, **Low**, **Medium**, and **High**. The default level is **Medium**. The lower the protection level, the higher the false negative rate and the lower the false positive rate. For details, see **Table 4-4**.

Table 4-4 Protection levels

Protection Level	Description
Low	WAF only blocks the requests with obvious attack signatures. If a large number of false alarms are reported, **Low** is recommended.
Medium	The default level is **Medium**, which meets a majority of web protection requirements.
High	WAF blocks the requests with no attack signature but have specific attack patterns. **High** is recommended if you want to block SQL injection, XSS, and command injection attacks.

4.5 Verifying a Global Protection Whitelist (Formerly False Alarm Masking) Rule by Simulating Requests with Postman

After your website is connected to WAF, you can use an API test tool to send HTTP/HTTPS requests to the website and verify that WAF protection rules take effect. This topic uses Postman as an example to describe how to verify a global protection whitelist (formerly false alarm masking) rule.

Example

Assume that your workloads are deployed in the **/product** directory, and parameter ID contains scripts or rich text submitted by your customers. To ensure service running and improve WAF protection accuracy, you plan to mask false alarms generated for content submitted by the customers.

Prerequisites

- You have connected the website you want to protect to WAF.
- **Basic Web Protection** has been enabled and its **Mode** is **Block**. **General Check** has been enabled.

Procedure

Step 1 **Download** and install Postman.

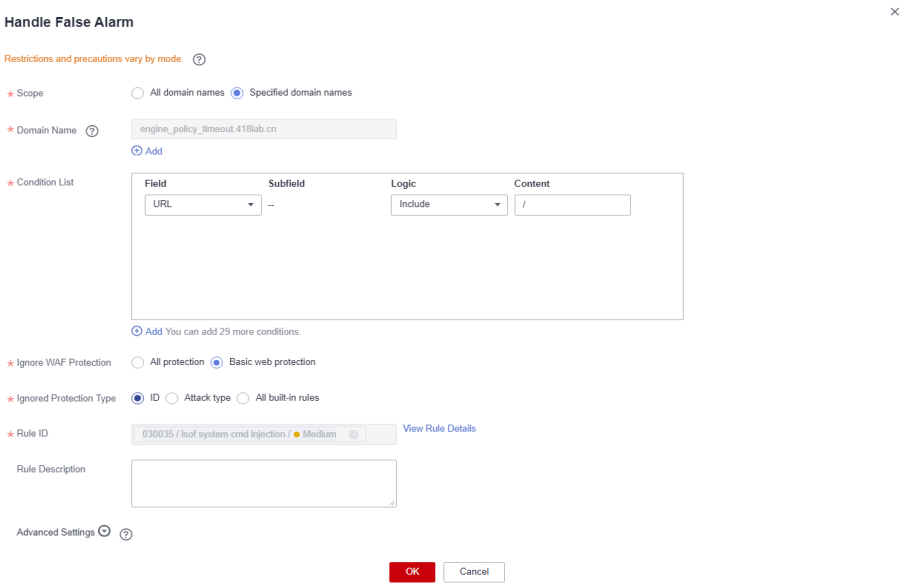
Step 2 On Postman, set the request path to **/product** and parameter ID to a common test script and send the request. The access request to the protected website is blocked.

Step 3 Handle the false alarm.

1. **Log in to the management console**.

2. Click in the upper left corner of the management console and select a region or project.
3. Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.
4. In the navigation pane on the left, choose **Events**.
5. On the **Events** page, WAF **010000** rule for **XSS Attack** is hit.
6. In the row containing the event, click **More** > **Handle as False Alarm** in the **Operation** column.
7. In the **Handle False Alarm** dialog box, add a global protection whitelist rule as shown in **Figure 4-23**.

Figure 4-23 Add Global Protection Whitelist Rule



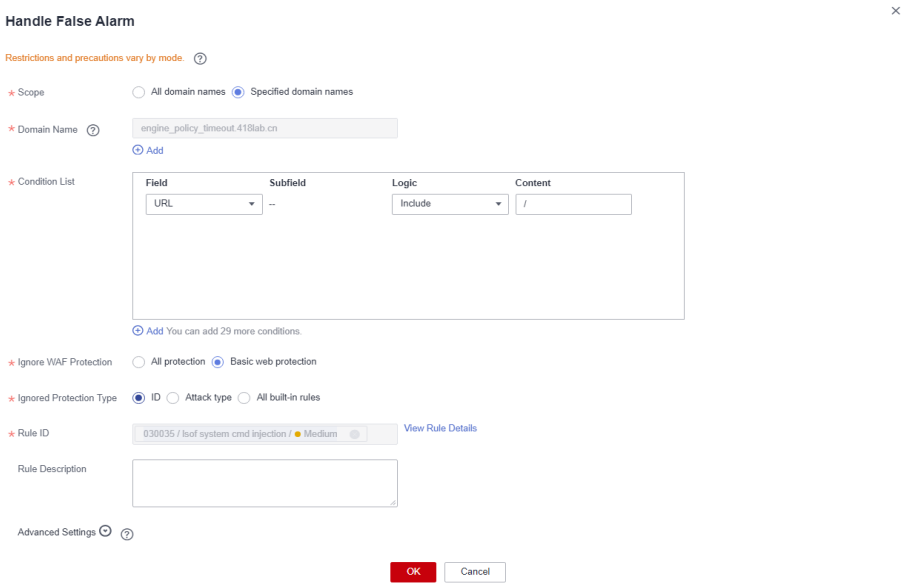
8. Click **OK**.

   It takes about 5 minutes for a protection rule to take effect.

Step 4 On Postman, set the request path to **/product** and parameter ID to a common test script and send the request again. The access request to the protected website is blocked again.

Step 5 Handle the false alarms that hit the **110053 XSS attack** rule by referring to **Step 3**.

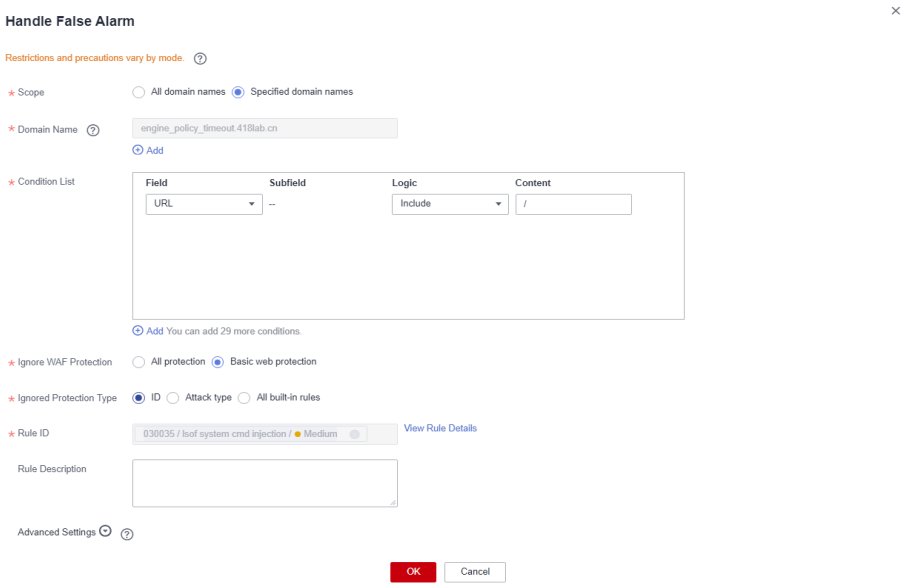
Figure 4-24 Add Global Protection Whitelist Rule



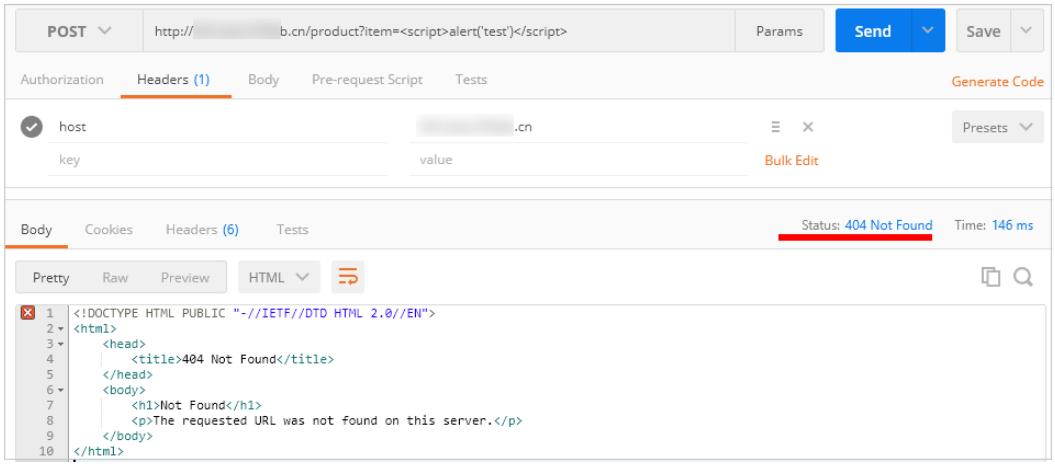
Step 6 On Postman, set the request path to **/product** and parameter ID to a common test script and send the request third time. The access request to the protected website is still blocked.

Step 7 Handle the false alarm that hits the **110060** rule for **XSS attack** by referring to **Step 3**.

Figure 4-25 Add Global Protection Whitelist Rule



Step 8 On Postman, set the request path to **/product** and the parameter ID to a common test script and send the request forth time. In this case, the access request to the protected website is not blocked. All global protection whitelist rules have taken effect.



Go to the **Event** page, no new XSS attack event is displayed.

Step 9 Simulate an attack on Postman to verify that the configured global protection whitelist (formerly false alarm masking) rules do not stop WAF from blocking XSS attacks against other parameters.

1. On Postman, set the request path to **/product** and parameter **item** to a common test script and send the request. The access request to the protected website is blocked.
2. On the **Events** page, view the XSS attack against parameter **item**.

Step 10 Simulate an attack on Postman to verify that the configured global protection whitelist (formerly false alarm masking) rules do not stop WAF from blocking XSS attacks against other paths.

1. On Postman, set the request path to **/order** and parameter ID to a common test script and send the request. The access request to the protected website is blocked.
2. On the **Events** page, view the event generated for blocked XSS attack against **/order** (**URL**) and parameter ID.

----End

5 Configuring Origin Server Security

5.1 Configuring the Minimum TLS Version and Cipher Suite to Better Secure Connections

HTTPS is a network protocol constructed based on Transport Layer Security (TLS) and HTTP for encrypted transmission and identity authentication. When you **add a domain name to WAF**, set **Client Protocol** to **HTTPS**. Then, you can configure the minimum TLS version and cipher suite to harden website security. The details are as follows:

- Minimum TLS version

  The minimum TLS version that can be used by a client to access the website. After you configure the minimum TLS version, only the requests over the connections secured with the minimum TLS version or the later version can access your website. This helps you meet security requirements for industrial websites.

  NOTE

  - Up to now, three TLS versions (TLS v1.0, TLS v1.1, and TLS v1.2) have been released, among which TLS v1.0 and TLS v1.1 have been released for a long time. Some encryption algorithms (such as SHA1 and RC4) used by TLS v1.0 and TLS v1.1 are vulnerable to attacks. TLS v1.0 and TLS v1.1 cannot meet the geometric growth of data transmission encryption requirements, which might bring potential security risks. To secure the communication and meet the Payment Card Industry Data Security Standard (PCI DSS), PCI Security Standards Council (PCI SSC) stated that it no longer accepted TLS v1.0 as of June 30, 2018. Vendors of mainstream browsers, such as Mozilla Firefox, Apple Safari, Google Chrome, and Microsoft Edge, also declared that they would stop supporting TLS v1.0 and TLS v1.1 by 2020.
  - You can query the TLS version supported by the website through other tools.

- Cipher suites

  A cipher suite is a set of algorithms that help secure a network connection through TLS. A more secure cipher suite can better secure the confidentiality and data integrity of websites.

Recommended Minimum TLS Versions for Different Scenarios

The default minimum TLS version configured in WAF is **TLS v1.0**. To better secure your website, configure an appropriate TLS version. **Table 5-1** lists the recommended minimum TLS versions for different scenarios.

Table 5-1 Recommended minimum TLS versions

Scenario	Minimum TLS Version (Recommended)	Protection Effect
Websites that handle critical business data, such as sites used in banking, finance, securities, and e-commerce.	TLS v1.2	WAF automatically blocks website access requests that use TLS v1.0 or TLS v1.1.
Websites with basic security requirements, for example, small- and medium-sized enterprise websites.	TLS v1.1	WAF automatically blocks website access requests that use TLS v1.0.
Client applications with no special security requirements	TLS v1.0	Requests using any TLS protocols can access the website.

Recommended Cipher Suites

The default cipher suite in WAF is **Cipher suite 1**. Cipher suite 1 offers a good mix of browser compatibility and security. For details about each cipher suite, see **Table 5-2**.

NOTE

The cipher suites whose configuration value starts with **!** are not supported. For example, ! MD5 indicates that the MD5 algorithm is not supported.

Table 5-2 Description of cipher suites

Cipher Suite Name	Cipher Suite Configuration Value	Description
Default cipher suite	• ECDHE-RSA-AES256-SHA384 • AES256-SHA256 • RC4 • HIGH • !MD5 • !aNULL • !eNULL • !NULL • !DH • !EDH • !AESGCM	• Compatibility: Good. A wide range of browsers are supported. • Security: Average
Cipher suite 1	• ECDHE-ECDSA-AES256-GCM-SHA384 • HIGH • !MEDIUM • !LOW • !aNULL • !eNULL • !DES • !MD5 • !PSK • !RC4 • !kRSA • !SRP • !3DES • !DSS • !EXP • !CAMELLIA • @STRENGTH	Recommended configuration. • Compatibility: Good. A wide range of browsers are supported. • Security: Good

Cipher Suite Name	Cipher Suite Configuration Value	Description
Cipher suite 2	• EECDH+AESGCM • EDH+AESGCM	• Compatibility: Average. Strict compliance with forward secrecy requirements of PCI DSS and excellent protection, but browsers of earlier versions may be unable to access the website. • Security: Excellent
Cipher suite 3	• ECDHE-RSA-AES128-GCM-SHA256 • ECDHE-RSA-AES256-GCM-SHA384 • ECDHE-RSA-AES256-SHA384 • RC4 • HIGH • !MD5 • !aNULL • !eNULL • !NULL • !DH • !EDH	• Compatibility: Average. Earlier versions of browsers may be unable to access the website. • Security: Excellent. Multiple algorithms, such as ECDHE, DHE-GCM, and RSA-AES-GCM, are supported.
Cipher suite 4	• ECDHE-RSA-AES256-GCM-SHA384 • ECDHE-RSA-AES128-GCM-SHA256 • ECDHE-RSA-AES256-SHA384 • AES256-SHA256 • RC4 • HIGH • !MD5 • !aNULL • !eNULL • !NULL • !EDH	• Compatibility: Good. A wide range of browsers are supported. • Security: Average. The GCM algorithm is supported.

Cipher Suite Name	Cipher Suite Configuration Value	Description
Cipher suite 5	• AES128-SHA:AES256-SHA • AES128-SHA256:AES256-SHA256 • HIGH • !MEDIUM • !LOW • !aNULL • !eNULL • !EXPORT • !DES • !MD5 • !PSK • !RC4 • !DHE • @STRENGTH	Supported algorithms: RSA-AES-CBC only
Cipher suite 6	• ECDHE-ECDSA-AES256-GCM-SHA384 • ECDHE-RSA-AES256-GCM-SHA384 • ECDHE-ECDSA-AES128-GCM-SHA256 • ECDHE-RSA-AES128-GCM-SHA256 • ECDHE-ECDSA-AES256-SHA384 • ECDHE-RSA-AES256-SHA384 • ECDHE-ECDSA-AES128-SHA256 • ECDHE-RSA-AES128-SHA256	• Compatibility: Average • Security: Good

The cipher suites provided by WAF are compatible with the latest browsers and clients, but are incompatible with some browsers of earlier versions. Compatible browsers or clients of a certain cipher suite may vary depending on the TLS version configured. Using TLS v1.0 as an example, **Table 5-3** describes the browser and client compatibility.

NOTICE

It is recommended that compatibility tests should be carried out on the service environment to ensure service stability.

Table 5-3 Incompatible browsers and clients for cipher suites under TLS v1.0

Browser/Client	Default Cipher Suite	Cipher Suite 1	Cipher Suite 2	Cipher Suite 3	Cipher Suite 4
Google Chrome 63 /macOS High Sierra 10.13.2	Not compatible	Compatible	Compatible	Compatible	Not compatible
Google Chrome 49/ Windows XP SP3	Not compatible	Not compatible	Not compatible	Not compatible	Not compatible
Internet Explorer 6 /Windows XP	Not compatible	Not compatible	Not compatible	Not compatible	Not compatible
Internet Explorer 8 /Windows XP	Not compatible	Not compatible	Not compatible	Not compatible	Not compatible
Safari 6/iOS 6.0.1	Compatible	Compatible	Not compatible	Compatible	Compatible
Safari 7/iOS 7.1	Compatible	Compatible	Not compatible	Compatible	Compatible
Safari 7/OS X 10.9	Compatible	Compatible	Not compatible	Compatible	Compatible
Safari 8/iOS 8.4	Compatible	Compatible	Not compatible	Compatible	Compatible
Safari 8/OS X 10.10	Compatible	Compatible	Not compatible	Compatible	Compatible
Internet Explorer 7/Windows Vista	Compatible	Compatible	Not compatible	Compatible	Compatible
Internet Explorer 8, 9, or 10 /Windows 7	Compatible	Compatible	Not compatible	Compatible	Compatible

Browser/Client	Default Cipher Suite	Cipher Suite 1	Cipher Suite 2	Cipher Suite 3	Cipher Suite 4
Internet Explorer 10 /Windows Phone 8.0	Compatible	Compatible	Not compatible	Compatible	Compatible
Java 7u25	Compatible	Compatible	Not compatible	Compatible	Compatible
OpenSSL 0.9.8y	Not compatible	Not compatible	Not compatible	Not compatible	Not compatible
Safari 5.1.9/OS X 10.6.8	Compatible	Compatible	Not compatible	Compatible	Compatible
Safari 6.0.4/OS X 10.8.4	Compatible	Compatible	Not compatible	Compatible	Compatible

Configuring the Minimum TLS Version and Cipher Suite

The following describes how to configure TLS v1.2 and cipher suite 1 as the minimum TLS version and how to verify that the configuration takes effect.

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

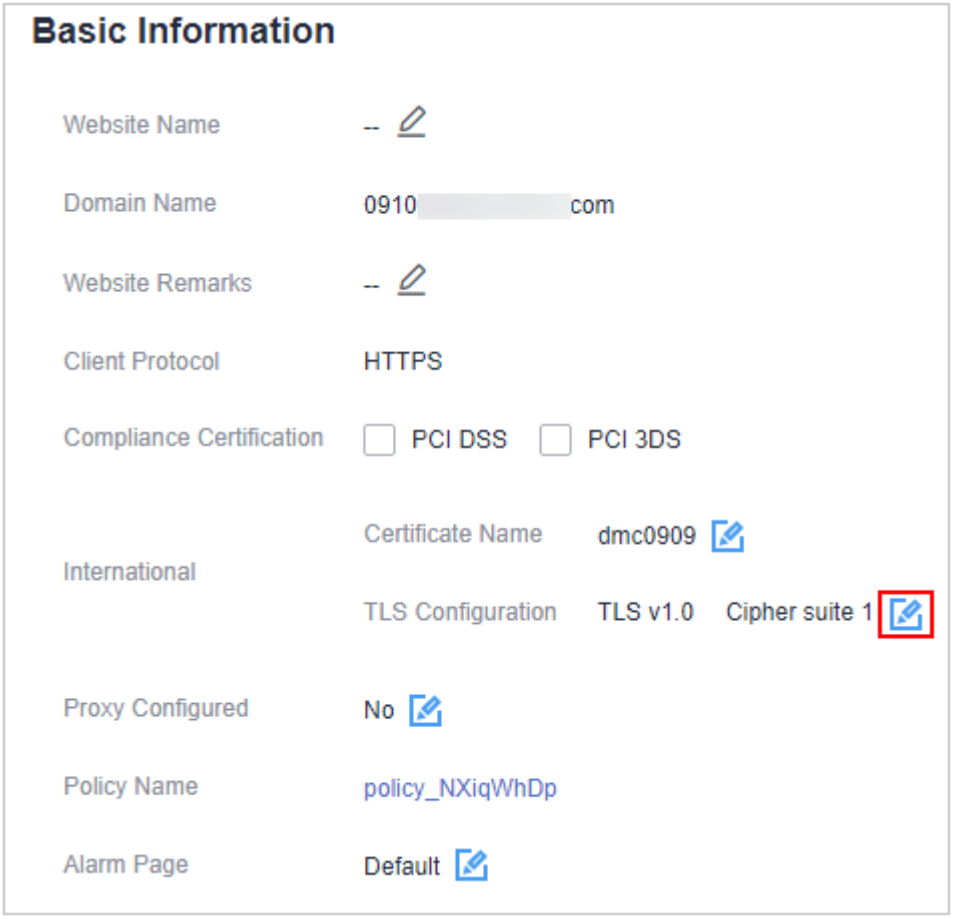
Step 3 Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Website Settings**.

Step 5 In the **Domain Name** column, click the website domain name to go to the basic information page.

Step 6 Click in the **TLS Configuration** row.

Figure 5-1 TLS configuration modification



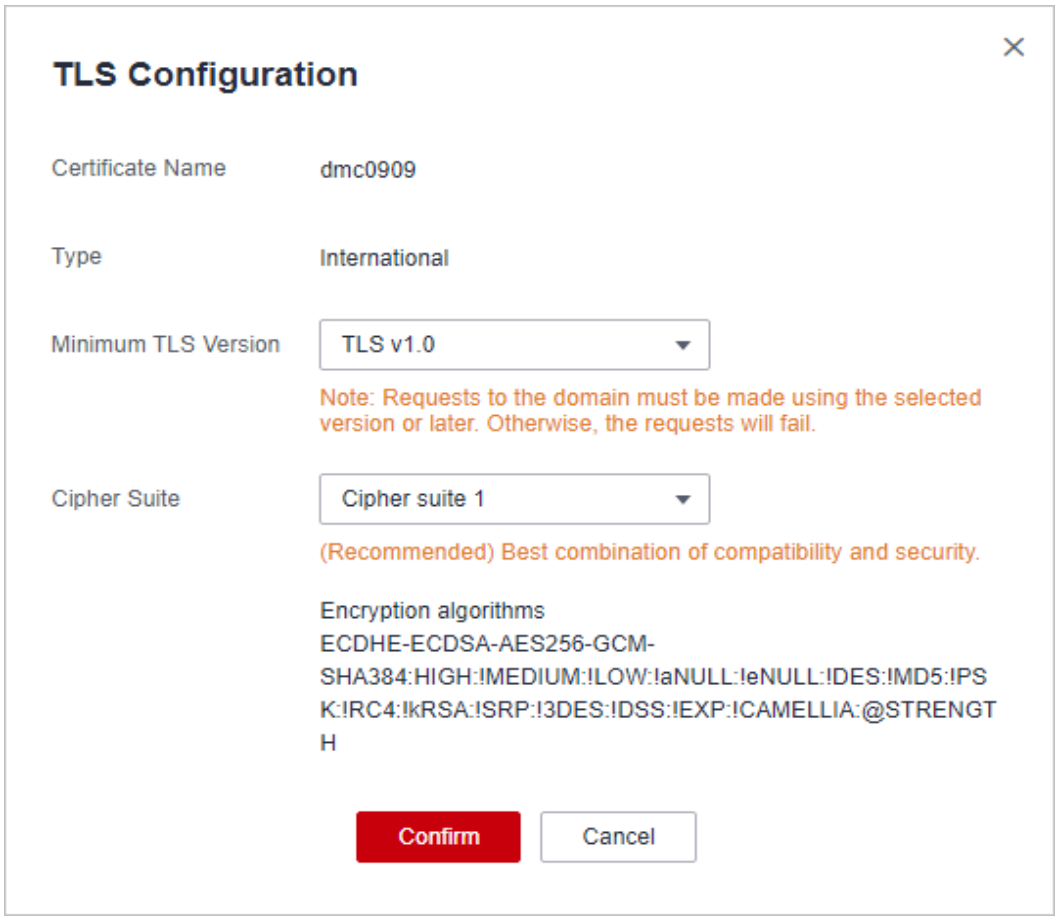
NOTE

WAF allows you to enable PCI DSS and PCI 3-Domain Secure (3DS) compliance certification checks with just a few clicks. After they are enabled, WAF will configure the minimum TLS version in accordance with the PCI DSS and PCI 3DS compliance certification requirements.

- If you enable the PCI DSS certification check:
  - The minimum TLS version and cypher suite are automatically set to **TLS v1.2** and **EECDH+AESGCM:EDH+AESGCM**, respectively, and cannot be changed.
  - To change the minimum TLS version and cipher suite, disable the check.
- If you enable the PCI 3DS certification check:
  - The minimum TLS version is automatically set to **TLS v1.2** and cannot be changed.
  - The check cannot be disabled.

Step 7 In the displayed **TLS Configuration** dialog box, select **TLS v1.2** as the minimum TLS version and **Cipher suite 1**.

Figure 5-2 TLS Configuration



Step 8 Click **Confirm**.

----End

Verification

If the **Minimum TLS Version** is set to **TLS v1.2**, verify that the website can be accessed over connections secured by TLS v1.2 or later but cannot be accessed over connections secured by TLS v1.1 or earlier.

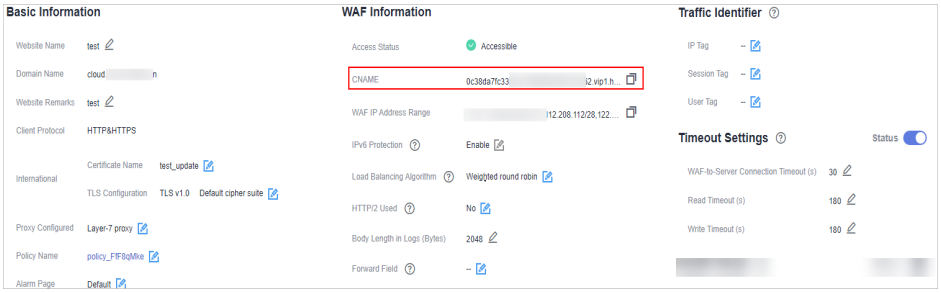
You can run commands on the local PC to check whether the TLS is configured successfully. Before the verification, ensure that **OpenSSL** has been installed on your local PC.

Step 1 Copy the CNAME record of the protected domain name and use the CNAME record to obtain WAF back-to-source IP addresses.

1. **Log in to the management console**.
2. Click in the upper left corner of the management console and select a region or project.
3. Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.
4. In the navigation pane on the left, choose **Website Settings**.

5. In the **Domain Name** column, click the domain name of the website to go to the basic information page.
6. In the **CNAME** row, click to copy the CNAME record.

Figure 5-3 Copying the CNAME record



Step 2 Obtain the WAF back-to-source IP addresses.

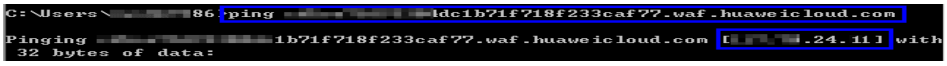
- Cloud mode

  In the command line interface (CLI) of the Windows OS, run the following command to obtain WAF back-to-source IP addresses:

  ping *CNAME record*

  The command output displays WAF back-to-source IP addresses. **Figure 5-4** shows an example.

  Figure 5-4 ping cname

- Dedicated mode

  a. In the navigation pane on the left, choose **Instances Management** > **Dedicated Engine** to go to the dedicated WAF instance page.

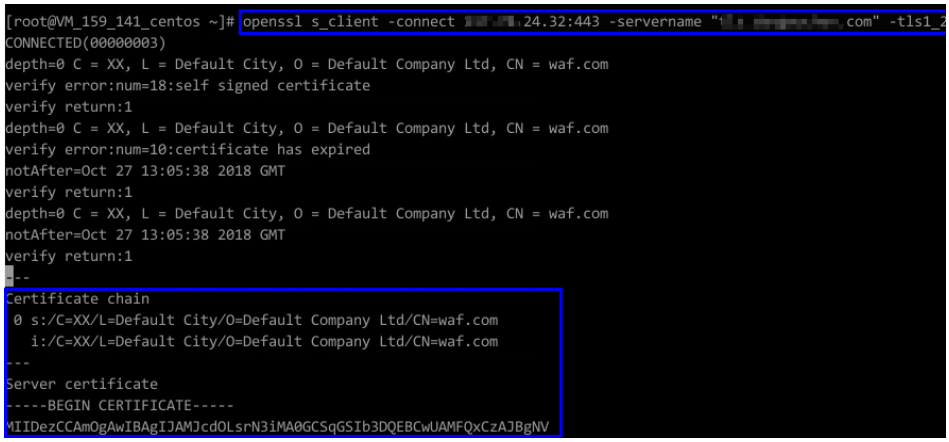
  b. In the **IP Address** column, obtain the subnet IP addresses of all dedicated WAF instances. Those subnet IP addresses are back-to-source IP addresses of dedicated WAF instances.

Step 3 Run the following command to verify that the protected website can be accessed using TLS v1.2.

openssl s_client -connect *WAF back-to-source IP address* **-servername** "*Domain name of the protected website*" **-tls1_2**

If the certificate information similar to the one shown in **Figure 5-5** is displayed, the website can be accessed using TLS v1.2.

Figure 5-5 Verifying TLS v1.2

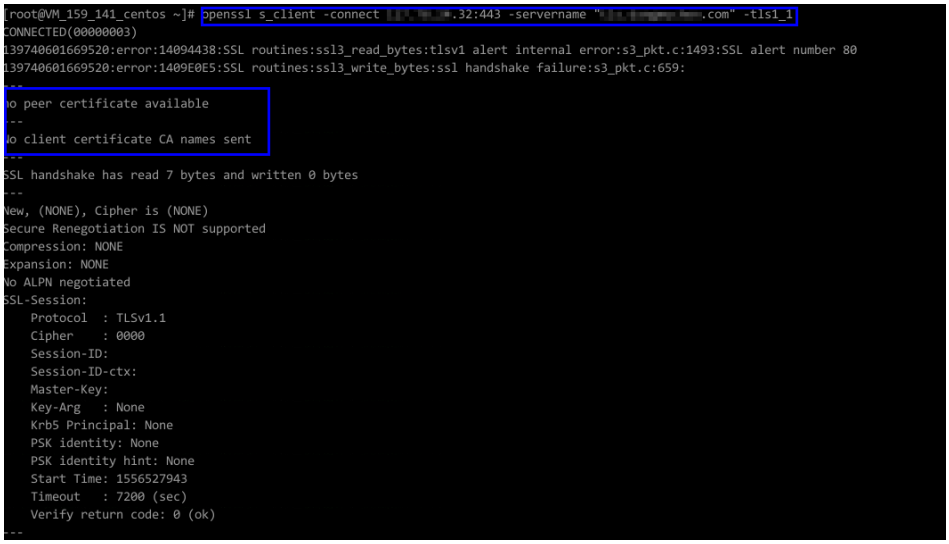


Step 4 Run the following command to verify that the protected website cannot be accessed using TLS v1.1.

openssl s_client -connect *WAF back-to-source IP address* **-servername** "*Protected domain name*" **-tls1_1**

If no certificate information is displayed, as shown in **Figure 5-6**, WAF has blocked the access that used TLS v1.1.

Figure 5-6 Verifying TLS v1.1



----End

5.2 Configuring an Access Control Policy on an ECS or ELB to Protect Origin Servers

After you connect your website to Web Application Firewall (WAF), configure an access control policy on your origin server to allow only the WAF back-to-source IP addresses. This prevents hackers from obtaining your origin server IP addresses and then bypassing WAF to attack origin servers.

This topic walks you through how to check whether origin servers have exposure risks and how to configure access control policies. This topic applies to scenarios where your origin servers are deploying on ECSs or have been added to backend servers of an ELB load balancer.

NOTE

- WAF will forward incoming traffic destined for the origin servers no matter whether you configure access control rules on the origin servers. However, if you have no access control rules configured on origin servers, bad actors may bypass WAF and directly attack your origin servers once they obtain your origin server IP addresses.
- If you use an NAT gateway before an ECS for forwarding data, you also need to configure an inbound rule in the security group the ECS belongs to by referring to **Configuring an Inbound Rule for an ECS**. This rule allows only WAF IP addresses to access origin servers to keep them secure.

Precautions

- Before configuring an access control policy on an origin server, ensure that you have connected all domain names of websites hosted on Elastic Cloud Server (ECS) or having Elastic Load Balance (ELB) deployed to WAF.
- The following issued should be considered when you configure a security group:
  - If you enable the WAF bypassed mode for your website but do not disable security group and network ACL configurations, the origin server may become inaccessible from the Internet.
  - If new WAF back-to-source IP addresses are assigned to WAF after a security group is configured for your website, the website may respond 5xx errors frequently.

How Do I Check Whether the Origin Server IP Address Is Exposed?

In a non-Huawei Cloud environment, use a Telnet tool to establish a connection over the service port of the public IP address of your origin server (or enter the IP address of your web application in the browser). Then, check whether the connection is established.

- Connection established

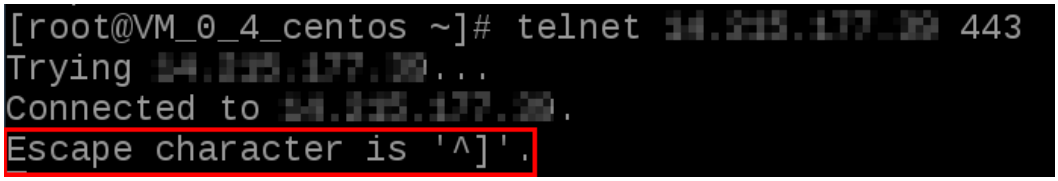
  The origin server has exposed to the public. Once a hacker obtains the public IP address of the origin server, the hacker can bypass WAF and directly attack the origin server.

- Connection not established

  The origin server is hidden from the public and there is no exposure risk.

For example, to check whether the origin server is exposed, check whether the origin server IP address that has been protected by WAF can be connected over port 443. If information similar to that shown in **Figure 5-7** is displayed, the connection is established and the origin server IP address is exposed.

Figure 5-7 Testing



Obtaining WAF Back-to-Source IP Addresses

A back-to-source IP address is a source IP address used by WAF to forward client requests to origin servers. To origin servers, all web requests come from WAF and all source IP addresses are WAF back-to-source IP addresses. The real client IP address is encapsulated into the HTTP X-Forwarded-For (XFF) header field. For more details, see **How Do I Whitelist the WAF Back-to-Source IP Address Ranges?**

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Website Settings**.

Step 5 In the upper right corner above the website list, click the **WAF Back-to-Source IP Addresses** link.

NOTE

WAF back-to-source IP addresses are periodically updated. Whitelist the new IP addresses in time to prevent those IP addresses from being blocked by origin servers.

Step 6 In the displayed dialog box, click **Copy** to copy all the addresses.

----End

Configuring an Inbound Rule for an ECS

If your origin server is deployed on an ECS, perform the following steps to configure a security group rule to allow only the WAF back-to-source IP addresses to access the origin server.

NOTICE

Ensure that all WAF back-to-source IP addresses are whitelisted by an inbound rule of the security group configured for the ECS. Otherwise, website may become inaccessible.

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner of the page and choose **Compute** > **Elastic Cloud Server**.

Step 4 Locate the row containing the ECS you want. In the **Name/ID** column, click the ECS name to go to the ECS details page.

Step 5 Click the **Security Groups** tab. Then, click **Change Security Group**.

Step 6 Click the security group ID and view the details.

Step 7 Click the **Inbound Rules** tab and click **Add Rule**. Then, specify parameters in the **Add Inbound Rule** dialog box. For details, see **Table 5-4**.

Figure 5-8 Add Inbound Rule

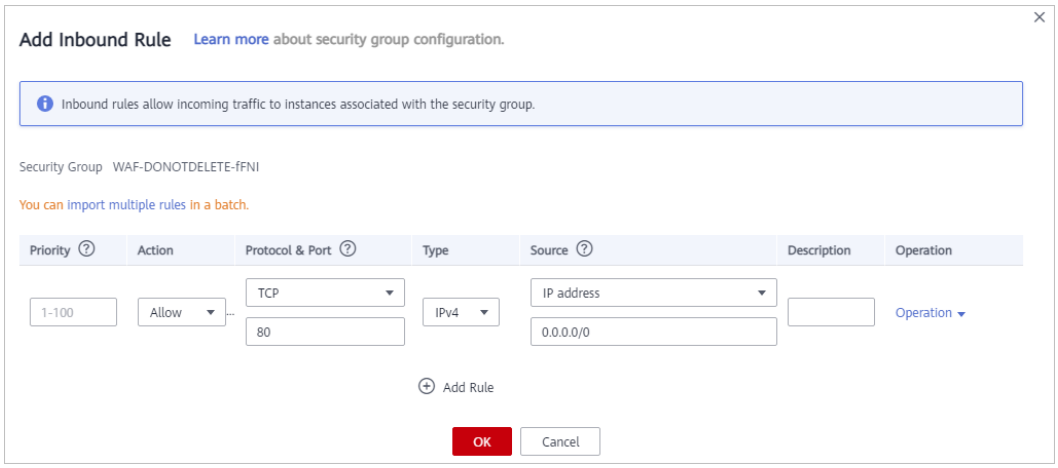


Table 5-4 Inbound rule parameters

Parameter	Description
Protocol & Port	Protocol and port for which the security group rule takes effect. If you select **TCP (Custom ports)**, enter the origin server port number in the text box below the TCP box.
Source	Add all WAF back-to-source IP addresses copied in **Step 6** one by one. **NOTE** One IP address is configured in a rule. Click **Add Rule** to add more rules. A maximum of 10 rules can be added.

Step 8 Click **OK**.

Then, the security group rules allow all inbound traffic from the WAF back-to-source IP addresses.

To check whether the security group rules take effect, refer to **How Do I Check Whether the Origin Server IP Address Is Exposed?** If a connection cannot be

established over the service port but the website is still accessible, the configuration takes effect.

----End

Enabling ELB Access Control

If your origin server is deployed on backend servers of an ELB load balancer, perform the following steps to configure an access control list to allow only the WAF back-to-source IP addresses to access the origin server.

Step 1 **Log in to the management console**.

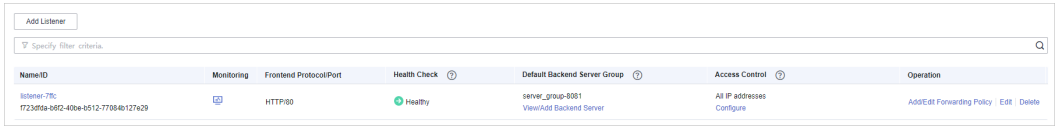
Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner of the page and choose **Networking** > **Elastic Load Balance**.

Step 4 Locate the load balancer you want. In the **Listener** column, click the listener name to go to the details page.

Step 5 In the **Access Control** row of the target listener, click **Configure**.

Figure 5-9 Listener list



Step 6 In the displayed dialog box, select **Whitelist** for **Access Control**.

1. Click **Create IP Address Group** and add the dedicated WAF instance IP addresses obtained in **Step 6** to the group being created.
2. Select the IP address group created in **Step 6.1** from the **IP Address Group** drop-down list.

Step 7 Click **OK**.

To check whether the security group rules take effect, refer to **How Do I Check Whether the Origin Server IP Address Is Exposed?** If a connection cannot be established over the service port but the website is still accessible, the configuration takes effect.

----End

6 Analyzing WAF Logs with LTS

6.1 Using LTS to Quickly Query and Analyze WAF Access Logs

After you authorize WAF to access Log Tank Service (LTS), you can use the WAF logs recorded by **LTS** for quick and efficient real-time analysis, device O&M management, and analysis of service trends.

This practice uses the access log stream **lts-waf-access** of log group **lts-waf** as an example to describe how to use LTS to quickly query and analyze logs.

Prerequisites

- You have connected the website you want to protect to WAF.
- You have **enabled LTS for WAF logging**.

Procedure

Step 1 **Log in to the management console**.

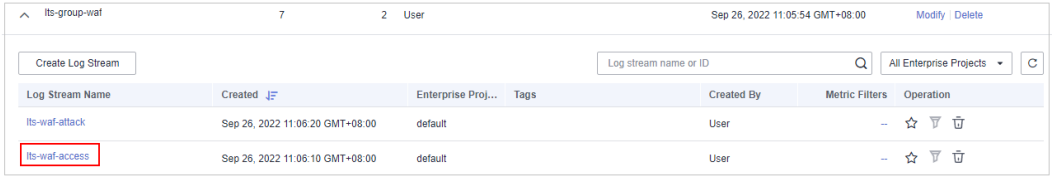
Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 4 In the **Log Group Name** column, click the name of the target log group (for example, **lts-waf**) to go the log stream page.

Step 5 In the **Log Stream Name** column, click the name of the log stream used for WAF access logs (for example, **lts-waf-access**). Then, select the **Log Stream** tab.

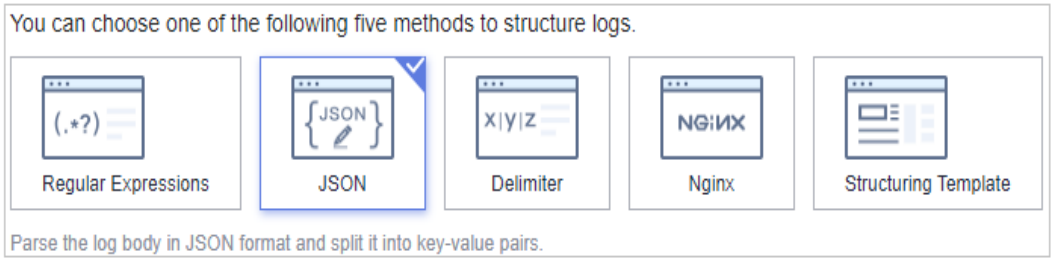
Figure 6-1 Accessing the log stream page



Step 6 On the log stream details page, click  in the upper right corner. On the page displayed, click the **Cloud Structured Parsing** tab.

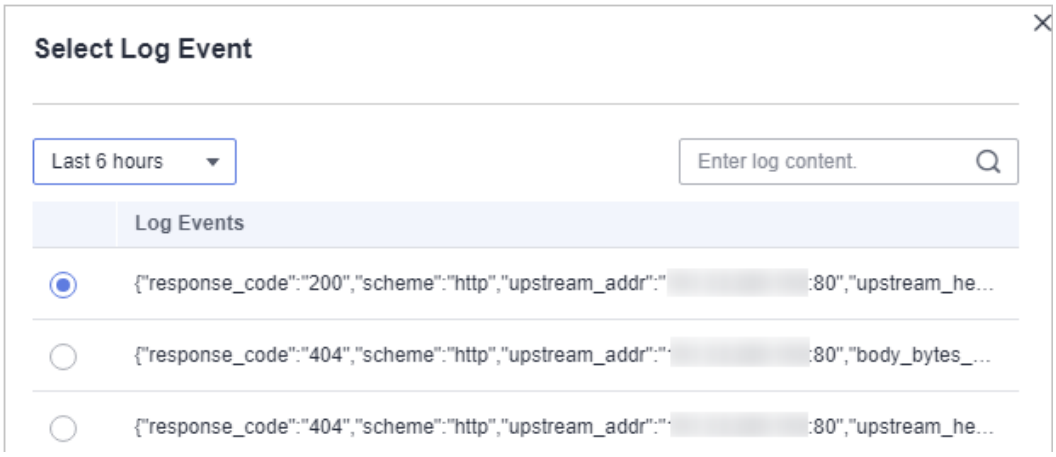
Step 7 Select **JSON** as the log structure, as shown in **Figure 6-2**.

Figure 6-2 JSON



Step 8 In the **Step 1 Select a sample log event.** area, click **Select from existing log events**. In the displayed **Select Log Event** dialog box, select a log and click **OK**.

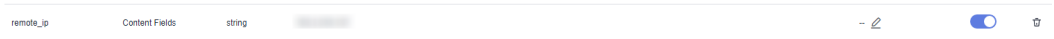
Figure 6-3 Select Log Event



Step 9 In the **Step 2 Extract fields** area, click **Intelligent Extraction** and enable quick analysis for the log field you want to analyze (for example, **remote_ip**).

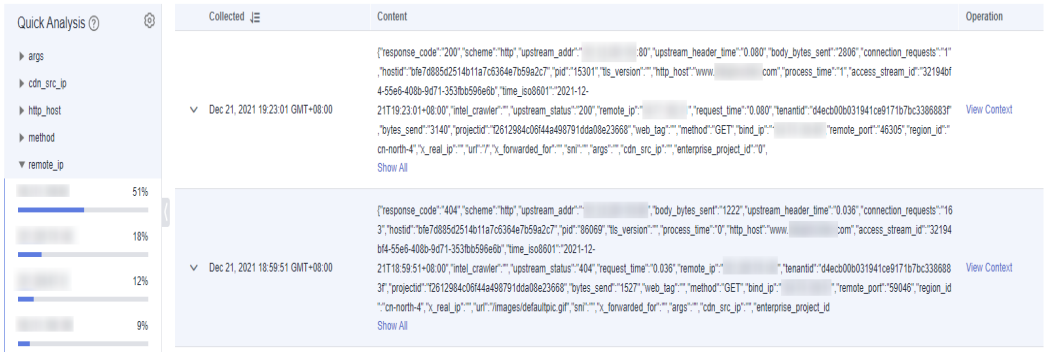
remote_ip: IP address of a client from which the request originates.

Figure 6-4 Selecting log fields for quick analysis



Step 10 Click **Save**. Then, LTS will start a quick analysis and do statistics for logs collected in a certain period.

Figure 6-5 Quickly analysis of access logs



Step 11 In the navigation pane, choose **Visualization**. On the right pane, select a log query time range, enter an SQL statement in the search box, and click **Query** to query the specified log.

You can enter either of the following SQL statements in the search box to query logs of a specified IP address:

select * where remote_ip = 'xx.xx.xx.xx' or **select * where remote_ip like 'xx.xx.xx%'**

----End

6.2 Using LTS to Analyze How WAF Blocks Spring Core RCE Vulnerability in Real Time

After you authorize WAF to access Log Tank Service (LTS), you can use the attack logs recorded by **LTS** for quick and efficient real-time analysis, device O&M management, and analysis of service trends.

This topic walks you through on how to enable the LTS quick analysis for WAF attack logs and use the Spring rule ID to quickly query and analyze the logs of the blocked Spring Core RCE vulnerabilities.

Prerequisites

- You have connected the website you want to protect to WAF.
- You have **enabled LTS for WAF logging**.
- You have obtained the Spring rule ID.

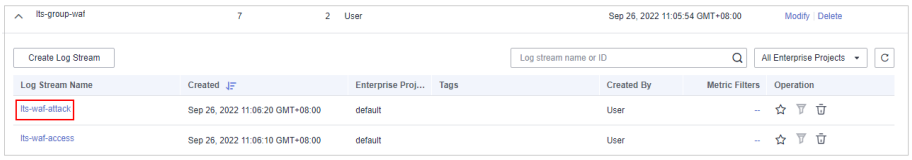
Procedure

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Figure 6-6 Log stream name configured for attack logs



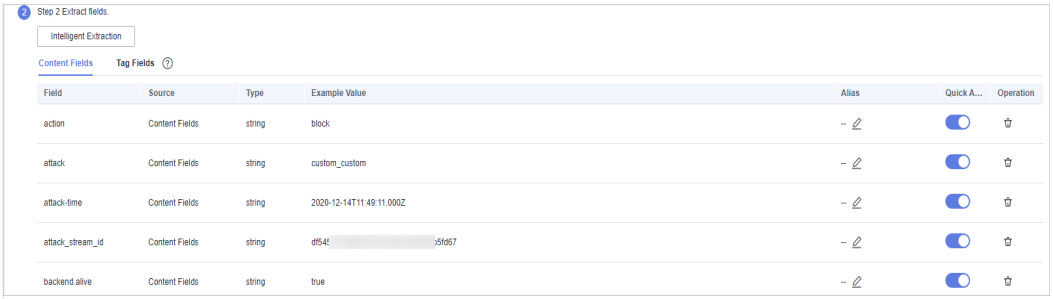
Step 4 In the log group list, expand the WAF log group and choose log stream **attack**.

Step 5 On the log stream details page, click in the upper right corner. On the page displayed, click the **Cloud Structured Parsing** tab.

Step 6 Select **JSON** for log structuring. Then, click **Select from existing events** and select a log in the dialog box displayed on the right.

Step 7 Click **Intelligent Extraction** to find the fields you want to analyze quickly. Enable these fields in the **Quick Analysis** column. After this, you can collect and analyze periodic logs.

Figure 6-7 Log extraction field



Step 8 Find the **category** field, click in the **Alias** column, change the field name, and click to save the settings.

NOTE

There is already a built-in **category** field in the system so you need to change the alias name of the **category** field, or your settings cannot be saved.

Step 9 In the lower right corner of the list, click **Save**. LTS quickly analyzes and collects statistics on logs in the specified period.

Step 10 In the navigation pane on the left, choose **Visualization**. Enter the following command and click **Query** to view the logs of the blocked Spring core RCE vulnerability.

select rule, hit_data where rule IN('XX','XX','XX','XX',)

Figure 6-8 Visualization query

----End

6.3 Using LTS to Configure Block Alarms for WAF Rules

After you authorize WAF to access Log Tank Service (LTS), you can use the attack logs recorded by **LTS** for quick and efficient real-time analysis, device O&M management, and analysis of service trends.

This topic walks you through how to enable LTS quick analysis for WAF attack logs and configure alarm rules to analyze WAF attack logs and generate alarms. In this way, you can gain insight into the protection status of your workloads in WAF in real time and make informed decisions.

Prerequisites

- You have connected the website you want to protect to WAF.
- You have enabled WAF attack log stream in LTS.
- You have enabled Simple Message Notification (SMN).

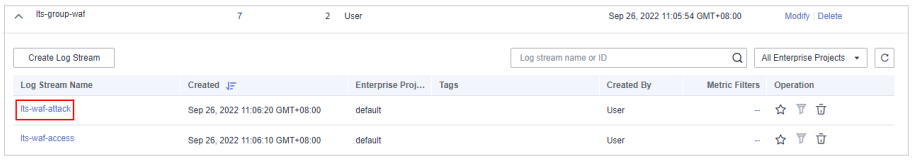
Quickly Analyzing Rule Block Logs

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Figure 6-9 Log stream name configured for attack logs



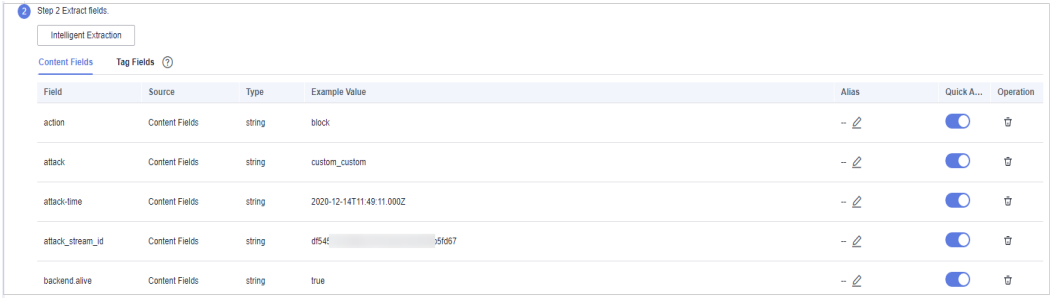
Step 4 In the log group list, expand the WAF log group and choose log stream **attack**.

Step 5 On the log stream details page, click in the upper right corner. On the page displayed, click the **Cloud Structured Parsing** tab.

Step 6 Select **JSON** for log structuring. Then, click **Select from existing events** and select a log in the dialog box displayed on the right.

Step 7 Click **Intelligent Extraction** to find the fields you want to analyze quickly. Enable these fields in the **Quick Analysis** column. After this, you can collect and analyze periodic logs.

Figure 6-10 Log extraction field



Step 8 Find the **category** field, click in the **Alias** column, change the field name, and click to save the settings.

NOTE

There is already a built-in **category** field in the system so you need to change the alias name of the **category** field, or your settings cannot be saved.

Step 9 In the lower right corner of the list, click **Save**. LTS quickly analyzes and collects statistics on logs in the specified period.

Step 10 In the navigation pane, choose **Visualization**. On the right pane, select a log query time range, enter an SQL statement in the search box, and click **Query**.

You can group logs by rule and URI. Enter the following SQL statement in the search box to query logs of a specified rule:

select rule, uri, count(*) as cnt where action = 'block' group by rule, uri order by cnt desc

----End

Creating an Alarm Rule

Step 1 Click in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 2 In the navigation pane on the left, choose **Alarms** > **Alarm Rules**.

Step 3 Click **Create**. In the dialog box displayed on the right, specify related parameters. **Table 6-1** describes the parameters.

Figure 6-11 Create Alarm Rule

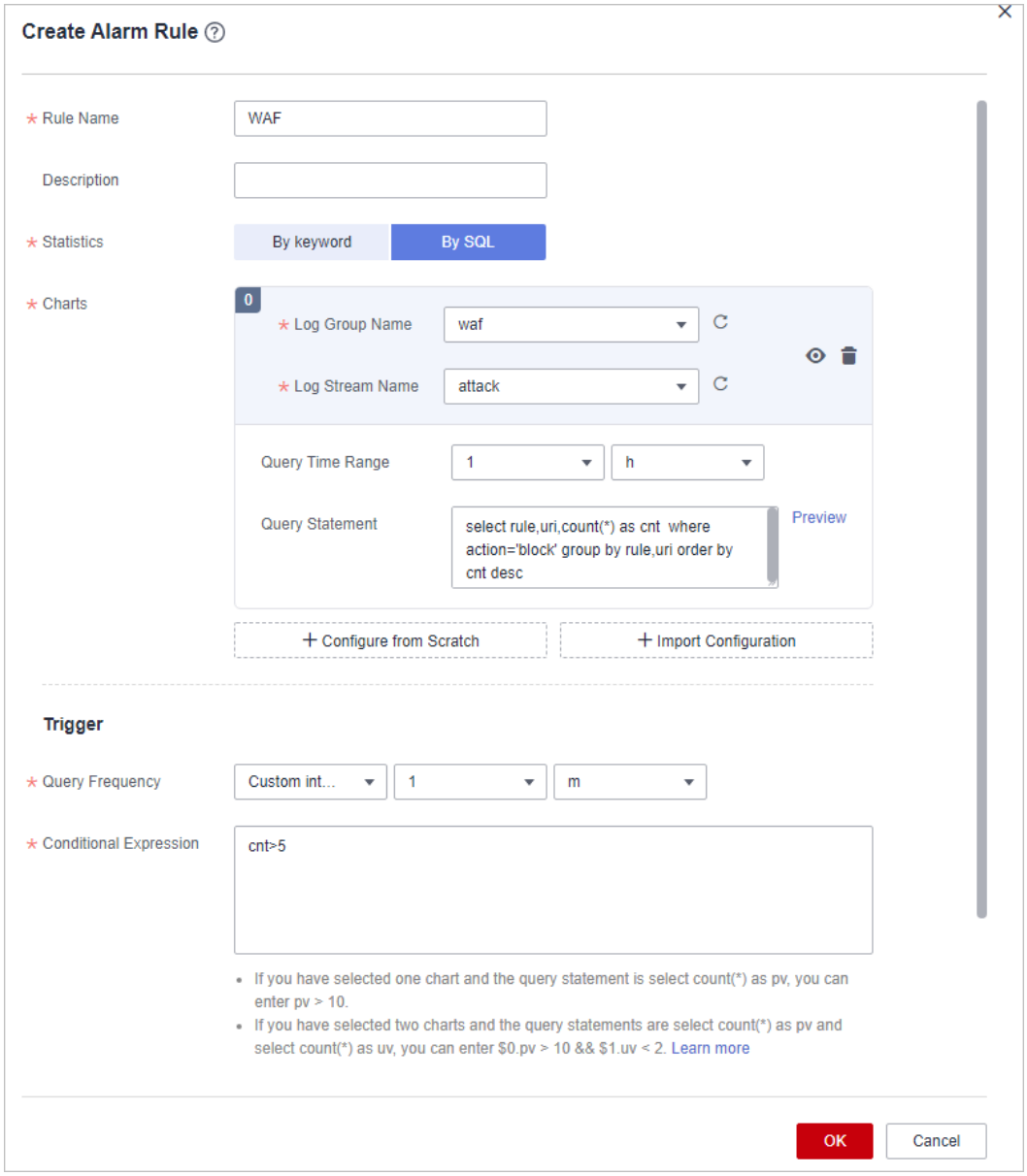


Table 6-1 Parameter description

Parameter	Description	Example Value
Rule Name	Name of the custom rule	WAF alarms
Statistics	Select **By SQL**.	By SQL

Parameter	Description	Example Value
Charts	Click **Configure from Scratch**. • Specify **Log Group Name** and **Log Stream Name**. • **Query Time Range**: Time range for log statistics • **Query Statement**: Enter the SQL statement configured in **Step 10**, for example, **select rule,uri,count(*) as cnt where action='block' group by rule,uri order by cnt desc**.	None
Query Frequency	Frequency which triggers alarms Generally, a fixed custom interval of 5 minutes is selected.	Custom interval 5 minutes
Conditional Expression	Alarm threshold	cnt>5
Alarm Severity	Select an alarm severity based on the blocking emergency of the rule. The options are **critical**, **major**, **minor**, and **info**.	Major
Send Notification	Select **Yes**.	Yes
SMN Topic	Select a topic from the drop-down list or create a topic. If there are no topics, click **View Topic** and perform the following steps to create a topic: 1. Create a topic. For details, see **Creating a Topic**. 2. Add one or more subscriptions to the topic. You will need to provide a phone number, email address, function, platform application endpoint, DMS endpoint, or HTTP/HTTPS endpoint for receiving alarm notifications. For details, see **Adding a Subscription**. 3. Confirm the subscription. After the subscription is added, confirm the subscription. For details about topics and subscriptions, see the *Simple Message Notification User Guide*.	None

Parameter	Description	Example Value
Time Zone/ Language	You can modify the language and time zone for receiving messages.	None
Message Templates	Select an existing template from the drop-down list box or click **Create Message Template** and create a template.	sql_template

Step 4 Confirm all parameters and click **OK**. The alarm rule is configured. When the alarm rule is triggered, you will receive an alarm email or SMS message.

----End

7 Configuring Collaborative Protection

7.1 Combining CDN and WAF to Get Improved Protection and Load Speed

How the Combination Works

- When a user accesses a website that uses Huawei Cloud CDN, the local DNS server will redirect all domain requests to CDN using CNAME records. **CDN** uses a group of predefined policies (such as the content type, geographical location, and network load status) to respond visitors with the nearest CDN IP address so that visitors can obtain requested website content as quickly as possible.

  Objects supported by CDN: domain names of web applications on Huawei Cloud, other cloud platforms, or on-premises data centers

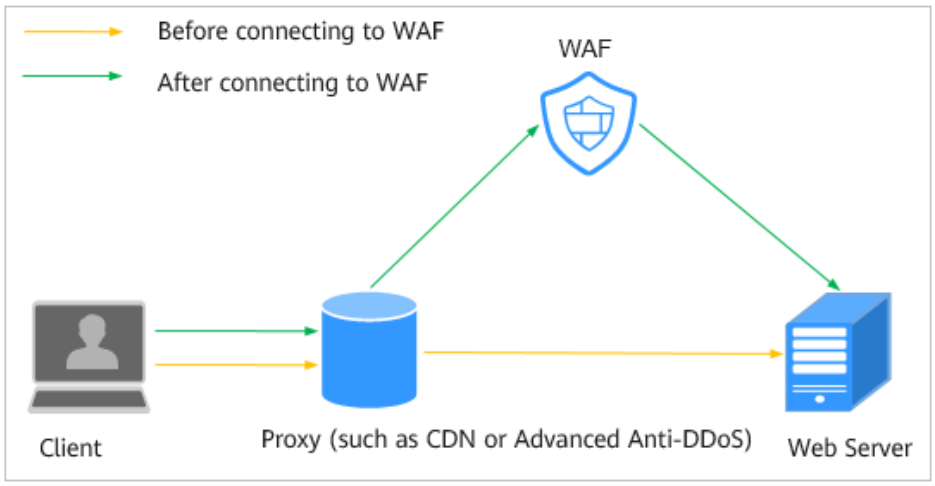
- Web Application Firewall (WAF) keeps web services stable and secure. It examines all HTTP and HTTPS requests to detect and block the following attacks: Structured Query Language (SQL) injection, cross-site scripting (XSS), web shells, command and code injections, file inclusion, sensitive file access, third-party vulnerability exploits, Challenge Collapsar (CC) attacks, malicious crawlers, and cross-site request forgery (CSRF).

  When adding a website to WAF, you can select **Cloud - CNAME**, **Cloud - Load balancer**, or **Dedicated** for **Protection**. Before you start, get familiar with the following differences:

  - **Cloud - CNAME**: protects your web applications that have domain name and are deployed on any clouds or in on-premises data centers.
  - **Cloud - Load balancer**: protects your web applications that are deployed on Huawei Cloud and accessible over domain names or IP addresses.
  - **Dedicated**: protects your web applications that are deployed on Huawei Cloud and accessible over domain names or IP addresses.

The combination of CDN and WAF can protect websites on Huawei Cloud, other clouds, or on-premises and improve website response time. **Figure 7-1** shows the configuration diagram.

Figure 7-1 WAF configuration when a proxy is used



After you deploy CDN and WAF for your website, traffic is accelerated by CDN and then forwarded to WAF. WAF checks received traffic and forwards only the normal traffic to the origin server. The combination protects the website against attacks while improving the website response speed and availability.

Point your website domain name to CDN and then change the CDN back-to-source address to the WAF CNAME record. After that, you can also add a WAF subdomain name and TXT record on your DNS management platform in case others have connected the website domain name to WAF before you configure CDN.

The configurations are as follows:

- Cloud - CNAME

  Point your website domain name to CDN and then change the CDN back-to-source address to the WAF CNAME record. After that, you can also add a WAF subdomain name and TXT record on your DNS management platform in case others have connected the website domain name to WAF before you configure CDN.
- Cloud -Load balancer

  Point your website domain name to CDN and change the CDN back-to-source IP address to the EIP bound to the load balancer configured for your load-balancing WAF instance.
- Dedicated mode

  Point your website domain name to CDN and change the CDN back-to-source IP address to the EIP bound to the load balancer configured for your dedicated WAF instance.

Constraints

If your website uses proxies such as anti-DDoS, Content Delivery Network (CDN), and cloud acceleration services, select **Per user** for **Rate Limit Mode** and enable **All WAF instances** for your CC attack protection rules.

Prerequisites

- **WAF has been purchased**.
- You have **added the website domain name to WAF** and configured other details, including origin server IP address and port.
- You have **connected the website domain name to CDN**.
- You have obtained the permissions from the DNS service provider to add domain names.
- (Optional) You have whitelisted WAF back-to-source IP addresses. If non-Huawei Cloud security software is used on the origin server, whitelist the WAF back-to-source IP addresses to prevent normal traffic from being blocked. For details, see **Configuring an Access Control Policy on an ECS or ELB to Protect Origin Servers**.

Cloud WAF Configuration

The following uses Huawei Cloud CDN as an example to describe how to configure domain name resolution. If you use Huawei Cloud CDN, perform the following steps directly. If you use non-Huawei Cloud CDN, configure domain name resolution on non-Huawei Cloud CDN based on the instructions in the following steps.

Step 1 Obtain settings of **CNAME**, **Subdomain Name**, and **TXT Record**.

1. Log in to the management console.
2. Click  in the upper left corner of the management console and select a region or project.
3. Click  in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.
4. In the navigation pane on the left, choose **Website Settings**.
5. In the row containing the desired domain name, click the domain name to go to the **Basic Information** page.
6. On the page for configuring basic domain information, click  in the row where the CNAME is located to copy **CNAME**. In the row of **Access Status**, click **Make Accessible**. In the **Access Guide** dialog box, copy **Subdomain Name** and **TXT Record**.

Step 2 **Change the origin server domain name of the primary origin server of CDN to the CNAME of WAF.**

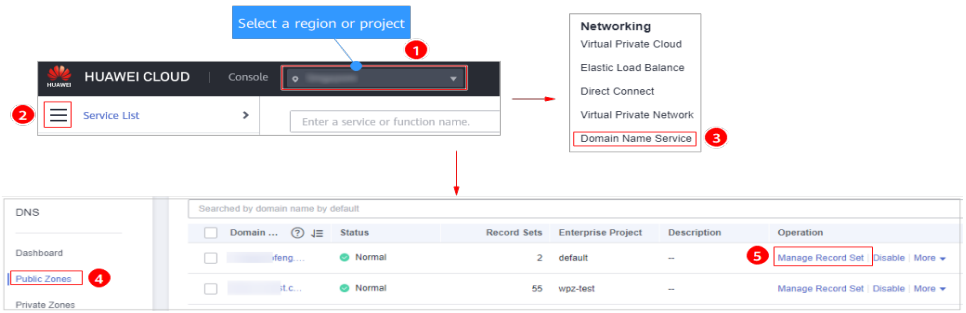
Step 3 (Optional) Add a WAF subdomain name and TXT record at your DNS provider.

NOTE

To prevent others from configuring your domain names on WAF in advance (this will cause interference on your domain name protection), this step is recommended.

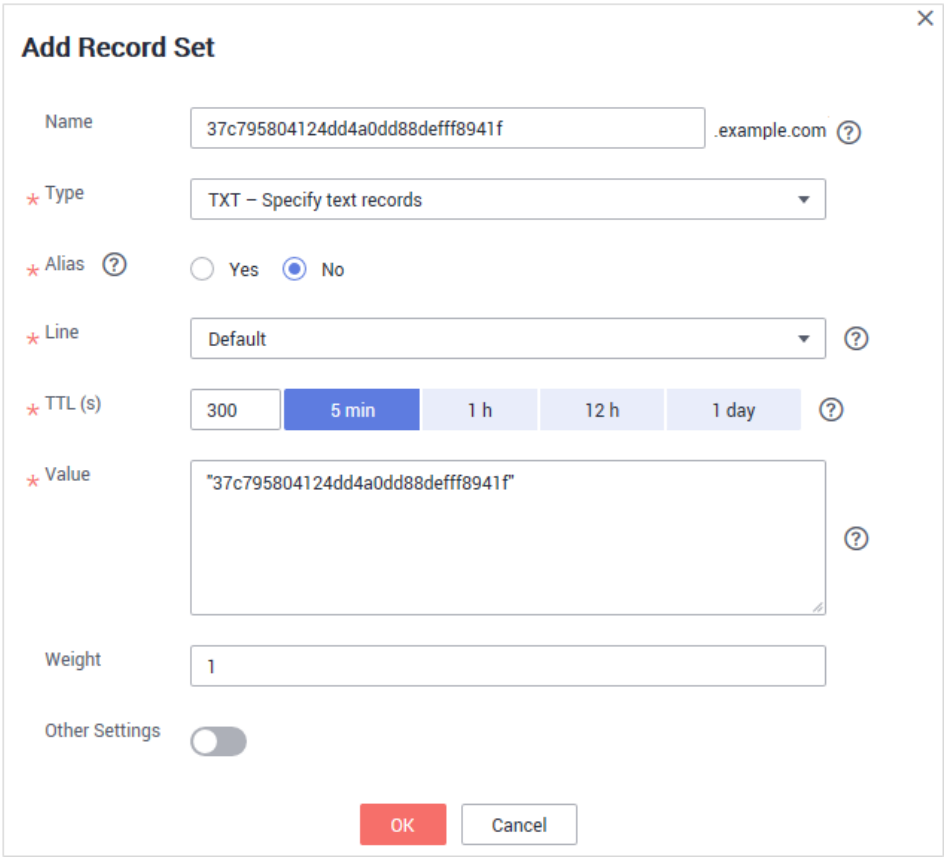
1. Access the DNS resolution page, as shown in **Figure 7-2**.

   Figure 7-2 DNS page

2. In the upper right corner of the page, click **Add Record Set**. The **Add Record Set** page is displayed. **Figure 7-3** shows an example.
   - **Name**: TXT record copied in **Step 1.6**.
   - **Type**: Select **TXT – Specify text records**.
   - **Alias**: Select **No**.
   - **Line**: **Default**
   - **TTL (s)**: The recommended value is **5 min**. A larger TTL value will make it slower for synchronization and update of DNS records.
   - **Value**: Add quotation marks to the TXT record copied in **Step 1.6** and paste them in the text box, for example, **TXT record**.
   - Keep other settings unchanged.

Figure 7-3 Adding a record set



3. Click **OK**.

Step 4 (Optional) Ping the IP address of your domain name to check whether the new DNS settings take effect.

NOTE

It takes some time for the new DNS settings to take effect. If ping fails, wait for 5 minutes and ping again.

----End

Dedicated/ELB WAF Configuration

Perform the following steps to complete configurations on Huawei Cloud CDN:

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner of the page and choose **Content Delivery & Edge Computing** > **Content Delivery Network**.

Step 4 In the navigation pane on the left, choose **Domains**.

Step 5 In the domain list, click the target domain name or click **Configure** in the **Operation** column.

Step 6 Click the **Basic Settings** tab. In the **Origin Server Settings** area, click **Edit**.

- If you use a dedicated WAF instance, in the **Address** text box, **enter the EIP you bind to the load balancer**.
- If you use a cloud WAF instance and select the load balancer access mode, in the **Address** text box, enter the EIP bound to the load balancer selected in **Add a Website to WAF (ELB Mode)**.

Step 7 Click **Save**.

----End

Verification

If **Access Status** is **Accessible**, the traffic destined for your website domain name or IP address is routed to WAF.

NOTICE

- WAF automatically checks the access status of protected websites every hour. If WAF detects that a protected website has received 20 access requests within 5 minutes, it considers that the website has been successfully connected to WAF.
- By default, WAF checks only the **Access Status** of domain names added or updated over the last two weeks. If a domain name was added to WAF two weeks ago and has not been modified in the last two weeks, you can click in the **Access Progress** column to refresh the progress.

If a domain name fails to be connected to WAF, its access status is **Inaccessible**. To fix this issue, see **Why Is the Access Status of a Domain Name or IP Address Inaccessible?**

7.2 Combining WAF and Layer-7 Load Balancers to Protect Services over Any Ports

This topic walks you through how to combine dedicated WAF instances and layer-7 load balancers to protect your services over non-standard ports that cannot be protected with WAF alone. For ports supported by WAF, see **Ports Supported by WAF**.

Protection Scenarios

The following procedure describes how WAF and ELB together protect **www.example.com:9876**. Port 9876 is a non-standard port WAF alone cannot protect.

Prerequisites

- You have purchased a dedicated layer-7 load balancer. For details about ELB load balancer types, see **Differences Between Shared and Dedicated Load Balancers**.

NOTE

Dedicated WAF instances issued before April 2023 cannot be used with dedicated network load balancers. If you use a dedicated network load balancer (TCP/UDP), ensure that your dedicated WAF instance has been upgraded to the latest version (issued after April 2023).

- Related ports have been enabled in the security group to which the dedicated WAF instance belongs.

  You can configure your security group as follows:

  – Inbound rules

    Add an inbound rule to allow incoming network traffic to pass through over a specified port based on your service requirements. For example, if you want to allow access from port 80, add a rule that allows **TCP** and port **80**.

  – Outbound rules

    Retain the default settings. All outgoing network traffic is allowed by default.

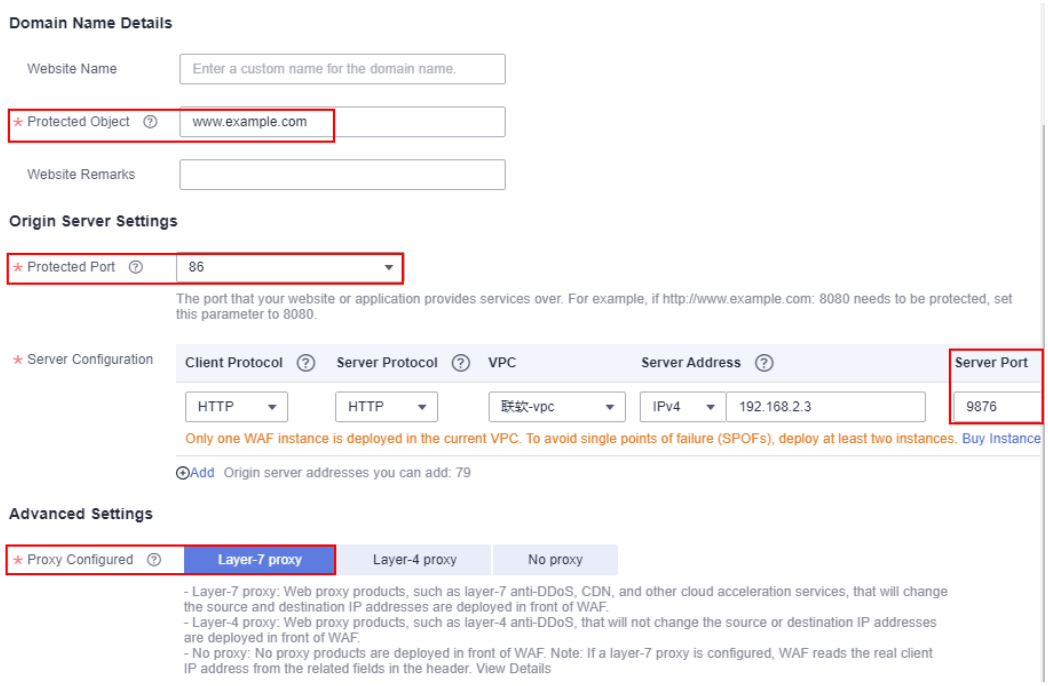
  For more details, see **Adding a Security Group Rule**.

Procedure

Step 1 **Buy a dedicated WAF instance**.

Step 2 Connect www.example.com to WAF by referring to **Adding a Website to WAF (Dedicated Mode)**. Select any non-standard port as protected port, for example, port 86, set **Server Port** to **9876**, and set **Proxy Configured** to **Layer-7 proxy**.

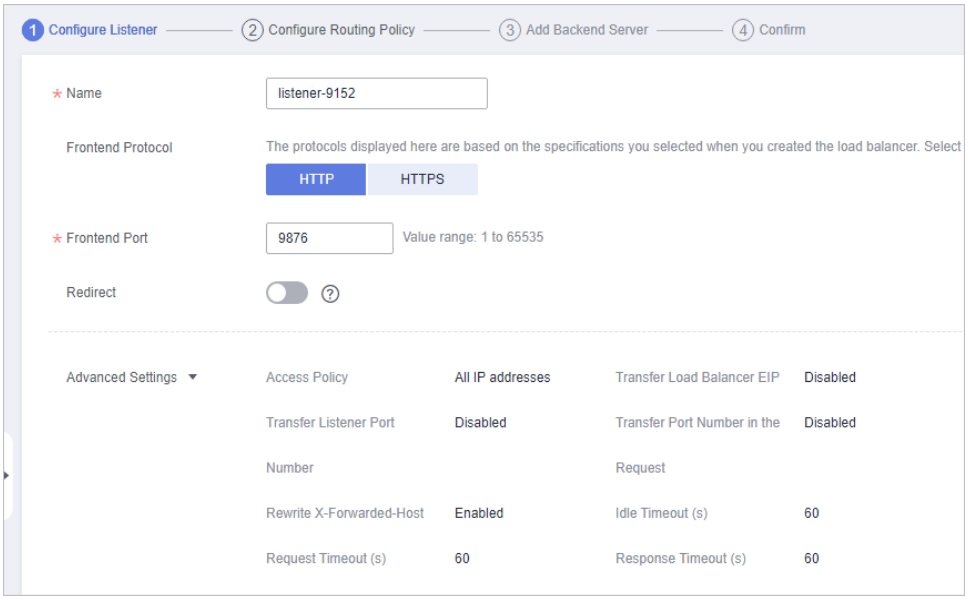
Figure 7-4 Adding a domain name to WAF



Step 3 Add listeners and backend server groups to the load balancer.

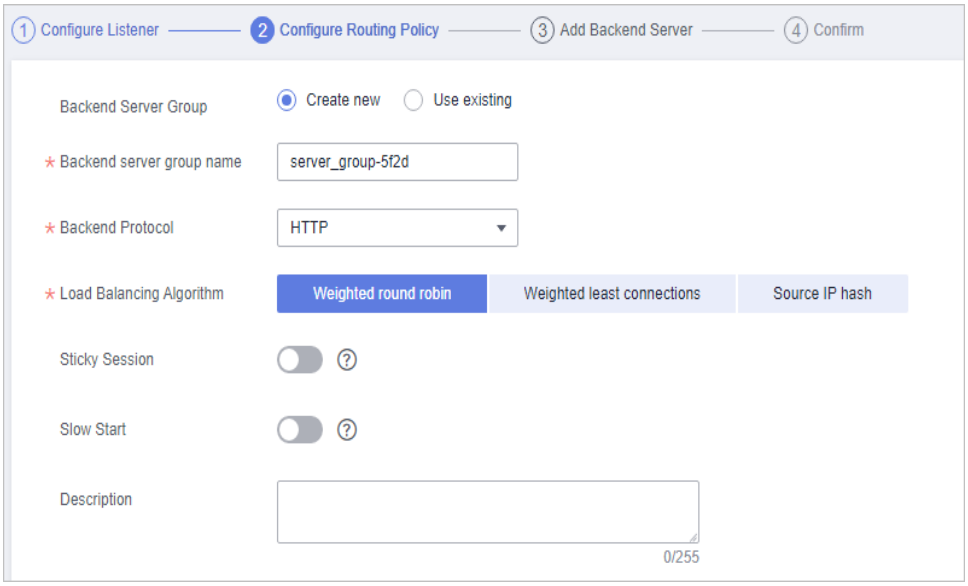
1. **Log in to the management console**.
2. Click in the upper left corner of the management console and select a region or project.
3. Click in the upper left corner of the page and choose **Elastic Load Balance** under **Networking** to go to the **Load Balancers** page.
4. Click the name of the load balancer in the **Name** column to go to the **Basic Information** page.
5. Click the **Listeners** tab and then click **Add Listener**. On the displayed page, configure the listener. In the **Frontend Port** text box, enter the port you want to protect. In this case, enter **9876**.

Figure 7-5 Configuring a listener



6. Click **Next: Configure Request Routing Policy**.

Figure 7-6 Configuring a backend server group



NOTICE

- If you select **Weighted round robin** for **Load Balancing Algorithm**, disable **Sticky Session**. If you enable **Sticky Session**, the same requests will be forwarded to the same dedicated WAF instance. If this instance becomes faulty, an error will occur when the requests come to it next time.
- For details about ELB traffic distribution policies, see **Load Balancing Algorithms**.

7. Click **Next: Add Backend Server** and click **Next: Confirm**.

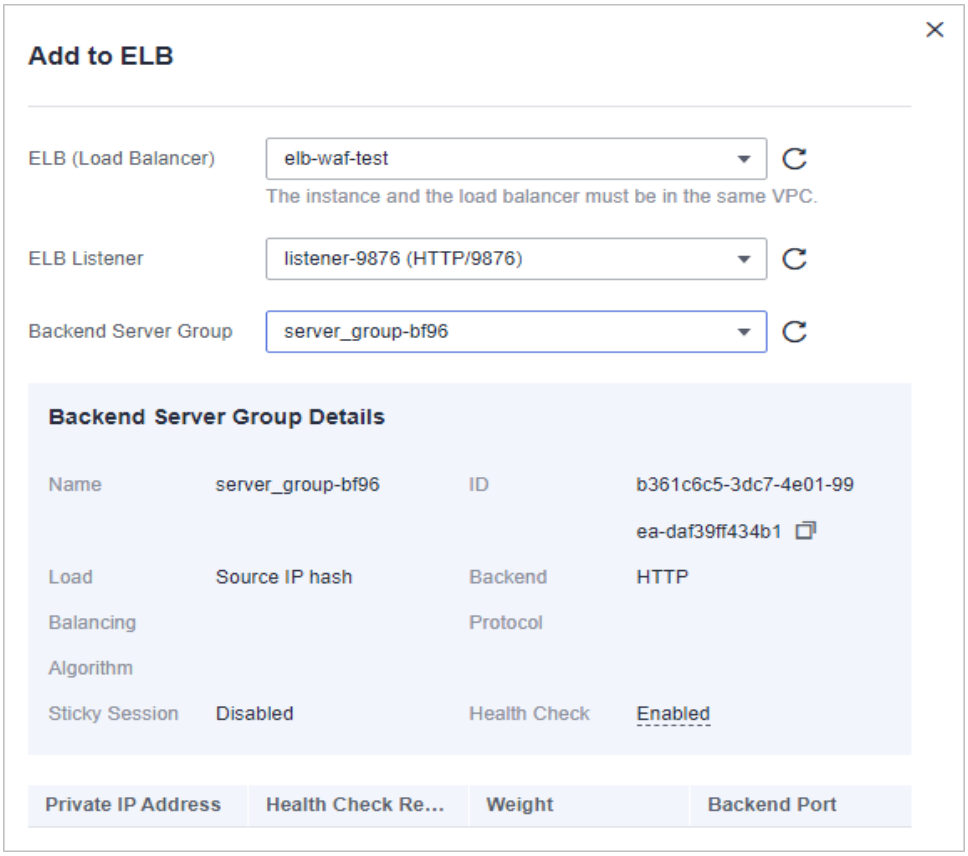
Step 4 Add the WAF instance to the load balancer.

1. **Log in to the management console**.
2. Click in the upper left corner of the management console and select a region or project.
3. Click in the upper left corner, select a region, and choose **Security & Compliance** > **Web Application Firewall** to go to the **Dashboard** page.
4. In the navigation pane on the left, choose **Instance Management** > **Dedicated Engine** to go to the dedicated WAF instance page.

Figure 7-7 Dedicated engine list

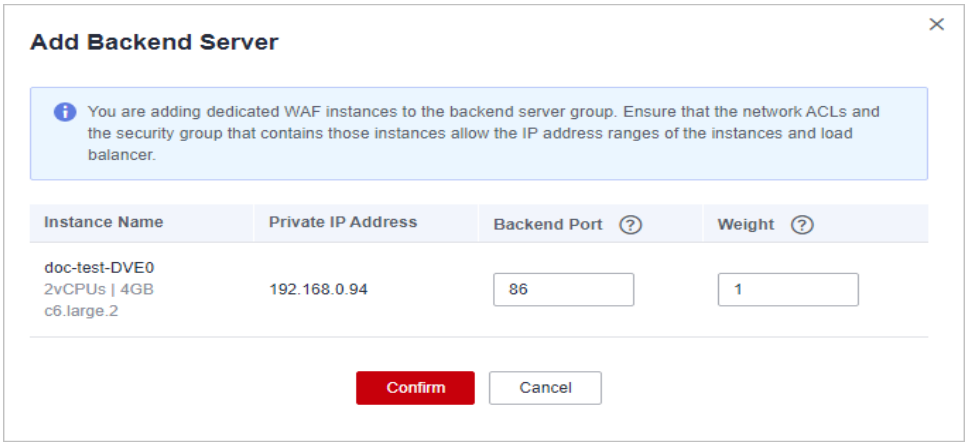
5. Locate the row containing the WAF instance. In the **Operation** column, click **More** > **Add to ELB**.
6. In the **Add to ELB** dialog box, specify **ELB (Load Balancer)**, **ELB Listener**, and **Backend Server Group** based on **Step 3**.

Figure 7-8 Add to ELB



7. Click **Confirm**. Then, configure service port for the WAF instance. In this example, configure **Backend Port** to **86**, which is the one we configured in **Step 2**.

Figure 7-9 Configuring Backend Port



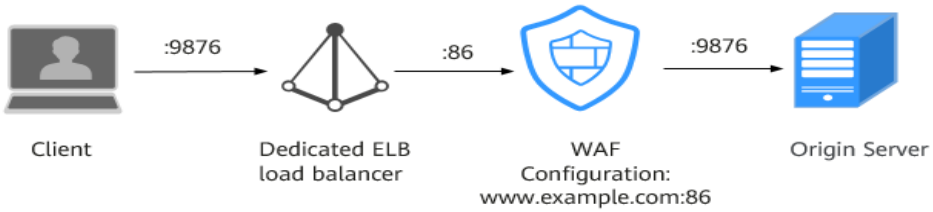
8. Click **Confirm**.

Step 5 **Bind an EIP to a Load Balancer**.

Step 6 **Whitelist the back-to-source IP addresses of your dedicated WAF instances**.

----End

How the Combination Protects Traffic



7.3 Combining WAF and HSS to Get Improved Web Tamper Protection

WAF examines HTTP/HTTPS requests. If an attacker attempts to tamper with web pages using attacks like SQL injection, WAF can identify and block the attacks in a timely manner, so they cannot sneak into or change anything in the OSs of your web servers.

Even if attacks bypass the first layer of protection, HSS WTP provides multi-level defenses. HSS WTP protects files in the web file directories from any unauthorized access. Only your website administrator can update the website content through the privileged process. Apart from that, HSS WTP also backs up web file directories locally and remotely. Once a file is tampered with, it can be quickly restored with backups. For dynamic web pages such as applications on web servers, HSS WTP uses Runtime Application Self-Protection (RASP) to monitor application access. It can detect tampering on dynamic data such as databases and prevent attackers from using applications to tamper with web pages in real time.

With HSS and WAF in place, you can stop worrying about web page tampering.

What Web Tampering Is and Impacts of Web Tampering

Web tampering is a type of cyberattack that exploits vulnerabilities in web applications to tamper with web application content or to insert hidden links. Web tampering attacks are often used to spread malicious information, incite unrest, and steal money.

Links to pornographic or otherwise illegal content may be inserted into normal web pages. Tampered web pages can permanently damage the brand image of your organization.

Differences Between The Web Tamper Protection Functions of HSS and WAF

Table 7-1 Differences between the web tamper protection functions of HSS and WAF

Type	HSS	WAF
Static web pages	Locks files in driver and web file directories to prevent attackers from tampering with them.	Caches static web pages on servers.

Type	HSS	WAF
Dynamic web pages	• Dynamic WTP Protects your data while Tomcat is running, detecting dynamic data tampering in databases. • Privileged process management Allows only privileged processes to modify web pages.	Not supported
Backup and restoration	• Proactive backup and restoration If WTP detects that a file in the protection directory is tampered with, it immediately uses the backup file on the local server to restore the file. • Remote backup and restoration If a file directory or backup directory on the local server becomes invalid, you can use the remote backup service to restore the tampered web page.	Not supported
Protection object	Web tamper prevention. This function is suitable for websites that have high protection requirements.	Websites that only require application-layer protection

Configuring a Web Tamper Protection Rule in WAF

NOTE

- This function is not supported in the starter edition or by ELB-mode WAF instances.
- For more details, see **Configuring a Web Tamper Protection Rule**.

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

Step 4 In the navigation pane on the left, choose **Policies**.

Step 5 Click the name of the target policy to go to the protection configuration page.

Step 6 Click the **Web Tamper Protection** configuration area and toggle it on or off if needed.

- : enabled.
- : disabled.

Step 7 In the upper left corner above the **Web Tamper Protection** rule list, click **Add Rule**.

Step 8 In the displayed dialog box, specify the parameters by referring to **Table 7-2**.

Figure 7-10 Adding a web tamper protection rule

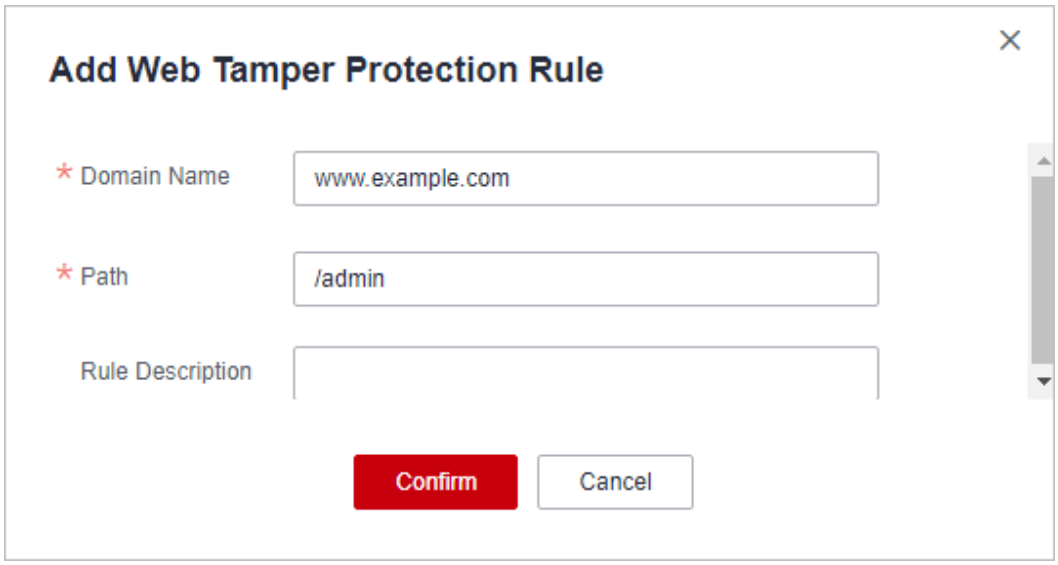


Table 7-2 Rule parameters

Parameter	Description	Example Value
Domain Name	Domain name of the website to be protected	**www.example.com**
Path	A part of the URL, not including the domain name A URL is used to define the address of a web page. The basic URL format is as follows: Protocol name://Domain name or IP address[:Port]/[Path/.../File name]. For example, if the URL is **http://www.example.com/admin**, set **Path** to **/admin**. **NOTE** • The path does not support regular expressions. • The path cannot contain two or more consecutive slashes. For example, **///admin**. If you enter **///admin**, WAF converts **///** to **/**.	**/admin**

Parameter	Description	Example Value
Rule Description	A brief description of the rule. This parameter is optional.	None

Step 9 Click **Confirm**. You can view the rule in the list of web tamper protection rules.

----End

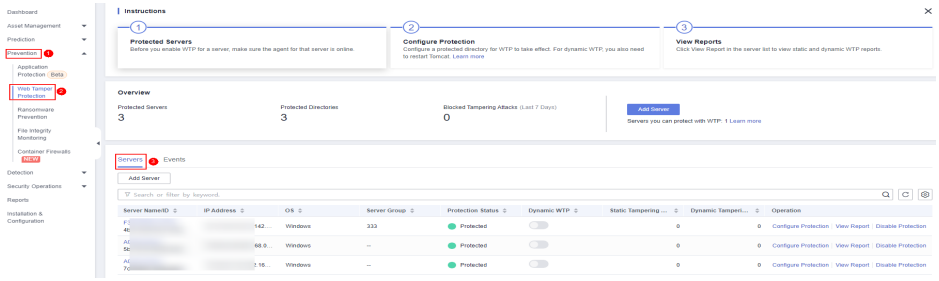
Enabling HSS Web Tamper Protection

Step 1 Log in to the management console.

Step 2 Click ☰ in the upper left corner of the page, select a region, and choose **Security & Compliance** > **HSS** to go to the HSS management console.

Step 3 In the navigation pane, choose **Prevention** > **Web Tamper Protection**. On the **Web Tamper Protection** page, click **Add Server**.

Figure 7-11 Adding a protected server



Step 4 On the **Add Server** page, click the **Available servers** tab. Select the target server, select a quota from the drop-down list or retain the default value, and click **Add and Enable Protection**.

Step 5 View the server status on the **Web Tamper Protection** page.

The premium edition will be enabled when you enable WTP.

- Choose **Prevention** > **Web Tamper Protection**. If the **Protection Status** of the server is **Protected**, WTP has been enabled.
- Choose **Asset Management** > **Servers & Quota** and click the **Servers** tab. If the protection status of the target server is **Enabled** and the **Edition/ Expiration Date** of it is **Premium (included with WTP)**, the premium edition provided by the WTP edition is enabled free of charge.

----End

NOTICE

- Before disabling WTP, perform a comprehensive detection on the server, handle known risks, and record operation information to prevent O&M errors and attacks on the server.
- If WTP is disabled, web applications are more likely to be tampered with. Therefore, you need to delete important data on the server, stop important services on the server, and disconnect the server from the external network in a timely manner to avoid unnecessary losses caused by attacks on the server.
- After you or disable WTP, files in the protected directory are no longer protected. You are advised to process files in the protected directory before performing these operations.
- If you find some files missing after disabling WTP, search for them in the local or remote backup path.
- The premium edition will be disabled when you disable WTP.

8 Upgrading a Dedicated WAF Instance

You can upgrade your dedicated WAF instances on the WAF console to obtain the latest protection performance. To ensure business availability during the upgrade, upgrade your dedicated WAF instances by following the procedure below.

NOTICE

If your workloads have high reliability requirements, at least two dedicated WAF instances should be deployed in dual-active or multi-active architecture. A single dedicated WAF instance may cause single points of failure (SPOFs) once the ECS hosting it becomes faulty.

Prerequisites

You have connected the website to a dedicated WAF instance.

Upgrading a Single Dedicated WAF Instance

If you have deployed only one dedicated WAF instance for your workloads, perform the following operations:

Step 1 **Buy a dedicated WAF instance**.

- The new dedicated WAF instance is of the latest version. So its **Upgrade** button is grayed out.
- The VPC, subnet, security group, and other settings of the new instance must be the same as those of the original one. In this way, the new instance automatically synchronizes all WAF protection configurations of the original instance.

Step 2 Run the curl command on any ECS in the VPC the original dedicated WAF instance locates to check whether the workloads are normal.

- HTTP workloads

  curl http://*IP-address-of-the-dedicated-WAF-instance* **:***Service-port* **-H "host:***Service-domain-name***" -H "User-Agent: Test"**

- HTTPS workloads

curl https://*IP-address-of-the-dedicated-WAF-instance* **:***Service-port* **-H "host:***Service-domain-name***" -H "User-Agent: Test"**

Check whether the service is normal. If the service is normal, go to **Step 3**. If the service is abnormal, fix the issue by referring to **Why Is the Access Status of a Domain Name Inaccessible?** and **How Do I Troubleshoot 404/502/504 Errors?**. After the fault is rectified, go to **Step 3**.

NOTE

To run a curl command, your ECS must meet the following requirements:

- The network communication is normal.
- A curl command line tool has been installed. If you are using a Windows ECS, manually install a **curl** command line tool on it. If you are a using a non-Windows ECS, no such action is required as the curl tool is installed automatically along with the operating system.

Step 3 Add the new dedicated WAF instance to the backend server group of the ELB load balancer you are using.

The following uses a shared load balancer to show how to add an instance to a backend server group.

1. Click  in the upper left corner, select a region, and choose **Security & Compliance** > **Web Application Firewall** to go to the **Dashboard** page.
2. In the navigation pane on the left, choose **Instance Management** > **Dedicated Engine** to go to the dedicated WAF instance page.
3. Locate the row containing the WAF instance. In the **Operation** column, click **More** > **Add to ELB**.
4. In the **Add to ELB** dialog box, specify **ELB (Load Balancer)**, **ELB Listener**, and **Backend Server Group** you configure for the original dedicated instance.
5. Click **Confirm**. Then, configure service port for the WAF instance. In this example, configure **Backend Port** to the one we configured for the original dedicated instance.

Step 4 On the ELB console, set the weight of the original dedicated instance to **0**. For details, see **Changing Backend Server Weights**.

Requests are not forwarded to a backend server if its weight is set to 0.

Step 5 Delete the original dedicated WAF instance during off-peak hours.

View the monitored metrics on Cloud Eye for the dedicated WAF instance, if there are less than five new connections, the traffic to the instance has decreased. For details, see **Viewing Metrics of a Dedicated WAF Instance**.

1. In the navigation pane on the left on the WAF console, choose **Instance Management** > **Dedicated Engine** to go to the dedicated WAF instance page.
2. In the row of the instance, click **More** > **Delete** in the **Operation** column.
3. Click **Confirm**.

   Resources on deleted instance are released and cannot be restored.

----End

Upgrading Multiple Dedicated WAF Instances

If you have deployed multiple dedicated WAF instances for your workloads, perform the following steps to upgrade them:

Step 1 On the ELB console, obtain the weight of a dedicated instance and then change the weight to **0**. For details, see **Changing Backend Server Weights**.

Requests are not forwarded to a backend server if its weight is set to 0.

Step 2 Upgrade the dedicated WAF instance during off-peak hours.

View the monitored metrics on Cloud Eye for the dedicated WAF instance, if there are less than five new connections, the traffic to the instance has decreased. For details, see **Viewing Metrics of a Dedicated WAF Instance**.

1. In the navigation pane on the left on the WAF console, choose **Instance Management** > **Dedicated Engine** to go to the dedicated WAF instance page.
2. In the row containing the desired instance, click **Upgrade** in the **Operation** column.
3. Confirm the upgrade conditions and click **Confirm**.

   It takes about 5 minutes for the upgrade to complete.

Step 3 Run the curl command on any ECS in the VPC the dedicated WAF instance locates to check whether the workloads are normal.

- HTTP workloads

  curl http://*IP-address-of-the-dedicated-WAF-instance* **:***Service-port* **-H "host:***Service-domain-name***" -H "User-Agent: Test"**

- HTTPS workloads

  curl https://*IP-address-of-the-dedicated-WAF-instance* **:***Service-port* **-H "host:***Service-domain-name***" -H "User-Agent: Test"**

Check whether the service is normal. If the service is normal, go to **Step 4**. If the service is abnormal, fix the issue by referring to **Why Is the Access Status of a Domain Name Inaccessible?** and **How Do I Troubleshoot 404/502/504 Errors?**. After the fault is rectified, go to **Step 4**.

NOTE

To run a curl command, your ECS must meet the following requirements:

- The network communication is normal.
- A curl command line tool has been installed. If you are using a Windows ECS, manually install a **curl** command line tool on it. If you are a using a non-Windows ECS, no such action is required as the curl tool is installed automatically along with the operating system.

Step 4 On the ELB console, change the weight of the dedicated instance from **0** to the one you obtain in **Step 1**. For details, see **Configuring Weights for Backend Servers**.

Step 5 Upgrade other dedicated WAF instances one by one by referring to **Step 1** to **Step 4**.

----End

9 Obtaining Real Client IP Addresses

A client IP address refers to an IP address of a visitor (or the device a visitor uses to initiate the request). Sometimes, a web application needs to require the client IP address. For example, a voting system needs to obtain the client IP addresses to ensure that each client casts only once.

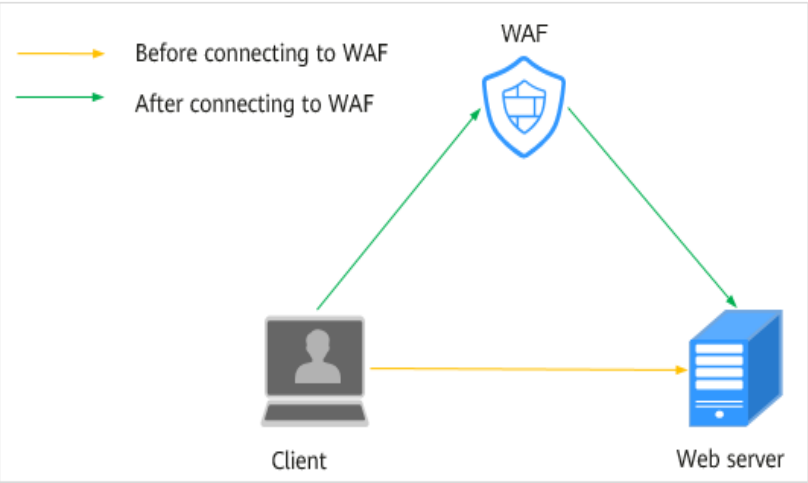
After your website is connected to WAF, WAF works as a reverse proxy between the client and the server. The real IP address of the server is hidden, and only the IP addresses of WAF are visible to web visitors. In this case, you can directly obtain the real IP address of the client through WAF or configure the website server to obtain the real IP address of the client.

The following describes how to obtain the client IP address from WAF and how to configure different types of web application servers, including Tomcat, Apache, Nginx, IIS 6, and IIS 7, to obtain the client IP address.

Background

Generally, a browser request does not directly reach the web server. Proxy servers, such as CDN, WAF, and advanced anti-DDoS, may be deployed between the browser and the origin server. Using WAF as an example, see **Figure 9-1**.

Figure 9-1 WAF deployment diagram



NOTE

- DNS resolves your domain name to the origin server IP address before your website is connected to WAF. Therefore, web visitors can directly access the server.
- After your website is connected to WAF, DNS resolves your domain name to the CNAME record of WAF. In this way, the traffic passes through WAF. WAF then filters out illegitimate traffic and only routes legitimate traffic back to the origin server.

In this case, the access request may be forwarded by multiple layers of security or acceleration proxies before reaching the origin server. So, how does the server obtain the real IP address of the client that initiates the request?

When forwarding HTTP requests to the downstream server, the transparent proxy server adds an **X-Forwarded-For** field to the HTTP header to identify the client IP address in the format of **X-Forwarded-For: client IP address, proxy 1-IP address, proxy 2-IP address, proxy 3-IP address, ……->…**.

Then, you can obtain the client IP address from the **X-Forwarded-For** field, the first IP address in which is the client IP address.

Constraints

- Ensure that **Proxy Configured** is configured correctly when you add the website to the WAF instance, or WAF cannot obtain the real IP address of your website visitors.

  To ensure that WAF obtains real client IP addresses and takes protective actions configured in protection policies, if your website has layer-7 proxy server such as CDN and cloud acceleration products deployed in front of WAF, select **Yes** for **Proxy Configured**. In other cases, select **No** for **Proxy Configured**.
- In normal cases, the first IP address in the **X-Forwarded-For** field is the real IP address of the client. If the length of an IPv6 address exceeds the length limit of the **X-Forwarded-For** field, the IP address cannot be read. In NAT64, the load balancer uses IPv4 listeners, which cannot read IPv6 addresses.

Obtaining the Client IP Address from WAF

After a website is connected to WAF, WAF is deployed between the client and server as a reverse proxy to protect the website. For details, see **From Which Request Field Can WAF Obtain the Real Client IP Address?**

The following describes how WAF uses the X-Forwarded-For and X-Real-IP variables to obtain the real IP address of a client:

- Using the **X-Forwarded-For** field to obtain the client IP address

  The client IP address is placed in the **X-Forwarded-For** HTTP header field. The format is as follows:

  ```
  X-Forwarded-For: Client IP address,Proxy 1-IP address,Proxy 2-IP address,...
  ```

  NOTE

  The first IP address included in the **X-Forwarded-For** field is the client IP address.

  The methods to obtain the **X-Forwarded-For** field by invoking the SDK interface in different programming languages are as follows:

  - **ASP**
    ```
    Request.ServerVariables("HTTP_X_FORWARDED_FOR")
    ```
  - **ASP.NET(C#)**
    ```
    Request.ServerVariables["HTTP_X_FORWARDED_FOR"]
    ```
  - **PHP**
    ```
    $_SERVER["HTTP_X_FORWARDED_FOR"]
    ```
  - **JSP**
    ```
    request.getHeader("HTTP_X_FORWARDED_FOR")
    ```

- Using the **X-Real-IP** field to obtain the client IP address (modifications caused by reverse proxies is considered)

  The methods to obtain the **X-Real-IP** field by invoking the SDK interface in different programming languages are as follows:

  - **ASP**
    ```
    Request.ServerVariables("HTTP_X_REAL_IP")
    ```
  - **ASP.NET(C#)**
    ```
    Request.ServerVariables["HTTP_X_REAL_IP"]
    ```
  - **PHP**
    ```
    $_SERVER["HTTP_X_REAL_IP"]
    ```
  - **JSP**
    ```
    request.getHeader("HTTP_X_REAL_IP")
    ```

How Does Tomcat Obtain the Client IP Address from Access Logs?

If Tomcat is deployed on your origin server, you can enable the X-Forwarded-For function of Tomcat to obtain the client IP address.

Step 1 Open the **server.xml** file in the **tomcat/conf/** directory. Partial information about the AccessLogValue logging function is as follows:

```
<Host name="localhost"  appBase="webapps" unpackWARs="true" autoDeploy="true">
      <Valve className="org.apache.catalina.values.AccessLogValue" directory="logs"
           prefix="localhost_access_log." suffix=".txt"
           pattern="%h %l %u %t "%r" %s %b" />
```

Step 2 Add **%{X-Forwarded-For}i** to **pattern**. Part of the modified **server.xml** file is as follows:

```
<Host name="localhost"  appBase="webapps" unpackWARs="true" autoDeploy="true">
      <Valve className="org.apache.catalina.valves.AccessLogValue" directory="logs"
           prefix="localhost_access_log." suffix=".txt"
           pattern="%{X-Forwarded-For}i %h %l %u %t "%r" %s %b" />
</Host>
```

Step 3 View the **localhost_access_log** file to obtain the client IP address from the **X-Forwarded-For** field.

----End

How Does Apache Obtain the Client IP Address from Access Logs?

If Apache HTTP Server 2.4 or later is deployed on your origin server, you can use the **mod_remoteip.so** file under **remoteip_module** in the Apache installation package to obtain the real client IP address.

- CentOS 7.6
  a. Add the following content to the **httpd.conf** file:


```
LoadModule remoteip_module modules/mod_remoteip.so ##Load the mod_remoteip.so module.
RemoteIPHeader X-Forwarded-For ## Set RemoteIPHeader.
RemoteIPInternalProxy WAF IP address range##Set the WAF back-to-source IP address range.
```

For more details, see **How Do I Whitelist the WAF Back-to-Source IP Address Ranges?**

NOTE

- File **/etc/httpd/conf.modules.d/00-base.conf:46** has been added to the **mod_remoteip.so** module.
- Use spaces to separate multiple back-to-source IP address ranges.

b. Replace **%h** with **%a** in the log format file.

```
LogFormat "%a %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%a %l %u %t \"%r\" %>s %b" common
```

c. Restart the Apache service to make the configuration take effect.

- Ubuntu 20.04.2

a. Add the following content to the **apache2.conf** file:

```
ln -s ../mods-available/remoteip.load /etc/apache2/mods-enabled/remoteip.load  ##Load the mod_remoteip.so module.
RemoteIPHeader X-Forwarded-For   ## Set RemoteIPHeader.
RemoteIPInternalProxy WAF IP address range##Set the WAF back-to-source IP address range.
```

For more details, see **How Do I Whitelist the WAF Back-to-Source IP Address Ranges?**

NOTE

- You can also add the following content to load the **mod_remoteip.so** module:

  LoadModule remoteip_module /usr/lib/apache2/modules/mod_remoteip.so
- Use spaces to separate multiple back-to-source IP address ranges.

b. Replace **%h** with **%a** in the log format file.

```
LogFormat "%a %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%a %l %u %t \"%r\" %>s %b" common
```

c. Restart the Apache service to make the configuration take effect.

If Apache 2.2 or earlier is deployed on your origin server, to obtain the real client IP address, you can run commands to install third-party module **mod_rpaf** of Apache and modify the **http.conf** file

Step 1 Run the following commands to install third-party module **mod_rpaf** for Apache:

```
wget https://github.com/gnif/mod_rpaf/archive/v0.6.0.tar.gz
tar xvfz mod_rpaf-0.6.tar.gz
cd mod_rpaf-0.6
/usr/local/apache/bin/apxs -i -c -n mod_rpaf-2.0.so mod_rpaf-2.0.c
```

Step 2 Open the **httpd.conf** configuration file and modify the file content as follows:

```
LoadModule rpaf_module   modules/mod_rpaf-2.0.so ##Load module mod_rpaf.
<IfModule mod_rpaf.c>
RPAFenable On
RPAFsethostname On
RPAFproxy_ips 127.0.0.1 <Reverse proxy IP address>
RPAFheader X-Forwarded-For
</IfModule>
```

Step 3 Define the log format.

```
LogFormat "%{X-Forwarded-For}i %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" common
```

Step 4 Enable customized logs.


```
CustomLog"[Apache server directory]/logs/$access.log"common
```

Step 5 Restart the Apache server for the configuration to take effect.

```
/[Apache server directory]/httpd/bin/apachectl restart
```

Step 6 View the **access.log** file to obtain the client IP address from the **X-Forwarded-For** field.

----End

How Does Nginx Obtain the Client IP Address from Access Logs?

If an Nginx reverse proxy is deployed on your origin server, you can configure location information on the Nginx reverse proxy so that the backend web server can use similar functions to obtain the client IP address

Step 1 Configure the following information in the corresponding location of the Nginx reverse proxy to obtain the information about the client IP address:

```
Location ^ /<uri> {
    proxy_pass  ....;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
}
```

Step 2 The backend web server obtains the real IP address of your website visitors by defining the Nginx log parameter **$http_x_forwarded_for**.

Example

```
log_format main ' "<$http_Cdn_Src_IP>" "{$http_x_real_ip}" "[$http_x_forwarded_for]" "$remote_addr"
' '$http_user_agent - $remote_user [$time_local] "$request" '    ' $status $body_bytes_sent "$http_referer" ';
```

----End

How Does IIS 6 Obtain the Client IP Address from Access Logs?

If you have deployed an IIS 6 server on your origin server, you can install the **F5XForwardedFor.dll** plug-in and obtain the client IP address from the access logs recorded by the IIS 6 server.

Step 1 Download the **F5XForwardedFor** module.

Step 2 Copy the **F5XForwardedFor.dll** file in the **x86\Release** or **x64\Release** directory to a specified directory (for example, **C:\ISAPIFilters**) based on the operating system version of your server. Ensure that the IIS process has the read permission for the directory.

Step 3 Open the IIS manager, right-click the website that is currently open, and choose **Attribute** from the shortcut menu. The **Attribute** page is displayed.

Step 4 On the **Attribute** page, switch to **ISAPI filter** and click **Add**. In the dialog box that is displayed, configure the following information:

- **Filter Name**: Set this parameter to **F5XForwardedFor**.
- **Executable file**: Set this parameter to the full path of **F5XForwardedFor.dll**, for example, **C:\ISAPIFilters\F5XForwardedFor.dll**.

Step 5 Click **OK** to restart the IIS 6 server.

Step 6 View the access logs recorded by the IIS 6 server (the default log path is **C:\WINDOWS\system32\LogFiles**, and the IIS log file name extension is **.log**). You can obtain client IP address from the **X-Forwarded-For** field.

----End

How Does IIS 7 Obtain the Client IP Address from Access Logs?

If you have deployed an IIS 7 server on your origin server, you can install the **F5XForwardedFor.dll** module and obtain the client IP address from the access logs recorded by the IIS 7 server.

Step 1 Download the **F5XForwardedFor** module.

Step 2 Copy the **F5XFFHttpModule.dll** and **F5XFFHttpModule.ini** files in the **x86\Release** or **x64\Release** directory to a specified directory (for example, **C:\x_forwarded_for\x86** or **C:\x_forwarded_for\x64**) based on the operating system version of your server. Ensure that the IIS process has the read permission for the directory.

Step 3 On the server home page, double-click **Modules** to go to the **Modules** page.

Step 4 Click **Configure Native Module**. In the dialog box displayed, click **Register**.

Step 5 In the displayed dialog box, register the downloaded DLL file according to the operating system, and then click **OK**.

- x86 operating system: registration module **x_forwarded_for_x86**
  - **Name**: **x_forwarded_for_x86**
  - **Path**: **C:\x_forwarded_for\x86\F5XFFHttpModule.dll**
- x64: Register the module **x_forwarded_for_x64**.
  - **Name**: **x_forwarded_for_x64**
  - **Path**: **C:\x_forwarded_for\x64\F5XFFHttpModule.dll**

Step 6 After the registration is complete, select the newly registered module (**x_forwarded_for_x86** or **x_forwarded_for_x64**) and click **OK**.

Step 7 In **ISAPI and CGI restriction**, add the registered DLL files by operating system and change **Restriction** to **Permitting**.

- x86 operating system:
  - **ISAPI or CGI path**: **C:\x_forwarded_for\x86\F5XFFHttpModule.dll**
  - **Description**: **x86**
- x64 operating system:
  - **ISAPI or CGI path**: **C:\x_forwarded_for\x64\F5XFFHttpModule.dll**
  - **Description**: **x64**

Step 8 Restart the IIS 7 server and wait for the configuration to take effect.

Step 9 View the access logs recorded by the IIS 7 server (the default log path is **C:\WINDOWS\system32\LogFiles**, and the IIS log file name extension is **.log**). You can obtain the client IP address from the **X-Forwarded-For** field.

----End

10 Configuring Accept-Encoding Field Forwarding and Disabling Response Packet Compression

The client usually declares that response compression is supported in the Accept-Encoding header, for example, Accept-Encoding: gzip. After the response passes through WAF, WAF considers that the response meets the compression standard and compresses the response. However, after receiving the response, the client cannot automatically decompress the response packet. The website then responds abnormally.

To address this issue, you can rewrite the **Accept-Encoding** header for the domain name on the WAF console so that WAF does not compress the response.

Procedure

Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Click in the upper left corner and choose **Web Application Firewall** under **Security & Compliance**.

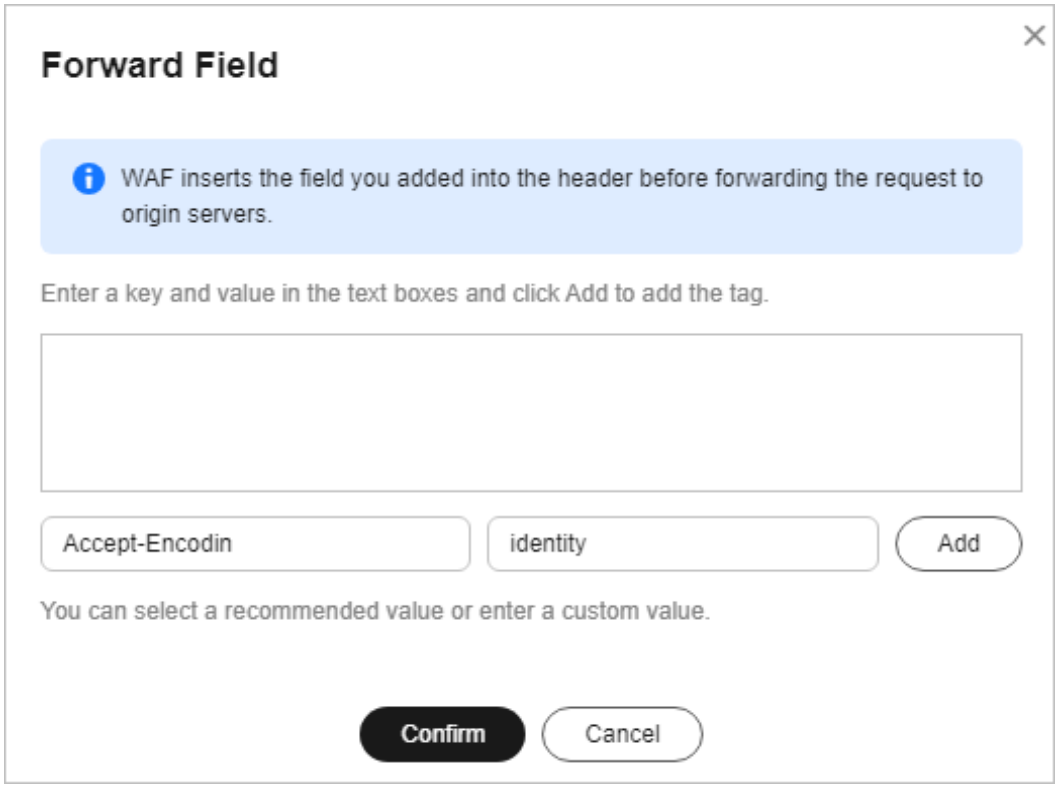
Step 4 In the navigation pane on the left, choose **Website Settings**.

Step 5 In the **Domain Name** column, click the website domain name to go to the basic information page.

Step 6 In the **Forward Field** column, click . In the displayed **Forward Field** dialog box, enter a key/value pair, and click **Add**.

Set **Key** to **Accept-Encoding** and **Value** to **identity**.

Figure 10-1 Forward Field



Step 7 Click **Confirm**.

----End

A Change History

Released On	Description
2024-02-01	This issue is the forty-eighth official release. Modified the following content: • **IP Address-based Rate Limiting** • **Cookie-based CC Attack Protection** • **Configuring Anti-Crawler Rules to Prevent Crawler Attacks** • **Configuring Basic Web Protection** • **Combining CDN and WAF to Get Improved Protection and Load Speed** • **Combining WAF and Layer-7 Load Balancers to Protect Services over Any Ports** • **Configuring Accept-Encoding Field Forwarding and Disabling Response Packet Compression**
2024-01-05	This issue is the forty-seventh official release. Modified the following content: • **Using LTS to Quickly Query and Analyze WAF Access Logs** • **Using LTS to Analyze How WAF Blocks Spring Core RCE Vulnerability in Real Time** • **Using LTS to Configure Block Alarms for WAF Rules**

Released On	Description
2023-11-30	This issue is the forty-sixth official release. • Adjusted the document structure. • Added the following content: – **Best Practices for Website Protection** – **Configuring CC Attack Protection for Common Scenarios** • Modified the following content: – **IP Address-based Rate Limiting** – **Cookie-based CC Attack Protection** – **Configuring Anti-Crawler Rules to Prevent Crawler Attacks** – **Configuring Basic Web Protection** – **Combining CDN and WAF to Get Improved Protection and Load Speed** – **Combining WAF and Layer-7 Load Balancers to Protect Services over Any Ports**
2023-11-15	This issue is the forty-fifth official release. Modified the following content: **Configuring the Minimum TLS Version and Cipher Suite to Better Secure Connections**
2023-08-11	This issue is the forty-fourth official release. Added **Configuring Accept-Encoding Field Forwarding and Disabling Response Packet Compression**.
2023-06-30	This issue is the forty-third official release. Modified **Obtaining Real Client IP Addresses**.
2023-06-07	This issue is the forty-second official release. Modified **Upgrading a Dedicated WAF Instance**.
2023-06-02	This issue is the forty-first official release. Modified **Configuring an Access Control Policy on an ECS or ELB to Protect Origin Servers**.
2023-03-03	This issue is the fortieth official release. Modified the following content: • **Combining WAF and Layer-7 Load Balancers to Protect Services over Any Ports** • **Combining WAF and HSS to Get Improved Web Tamper Protection**

Released On	Description
2022-10-25	This is the thirty-ninth official release. Modified the following topics: **Upgrading a Dedicated WAF Instance**
2022-09-30	This issue is the thirty-eighth official release. Added **Combining WAF and Layer-7 Load Balancers to Protect Services over Any Ports**.
2022-08-11	This issue is the thirty-seventh official release. Added the following content: • **Restricting Malicious Requests in Promotions by Using Cookies and HWWAFSESID**
2022-07-26	This issue is the thirty-sixth official release. Modified **Combining WAF and HSS to Get Improved Web Tamper Protection**.
2022-07-06	This issue is the thirty-fifth official release. Released the function for counting requests to all WAF instances. Modified the following topics: • **Configuring CC Attack Protection** • **Combining CDN and WAF to Get Improved Protection and Load Speed**
2022-07-04	This issue is the thirty-fourth official release. Released the global protection whitelist function. Modified the following topics: • **Handling False Alarms to Get Improved Basic Web Protection** • **Apache Dubbo Deserialization Vulnerability**
2022-05-23	This issue is the thirty-third official release. • Added **Combining WAF and HSS to Get Improved Web Tamper Protection**. • Modified **Obtaining Real Client IP Addresses**.
2022-05-05	This issue is the thirty-second official release. Added constraints in **Obtaining Real Client IP Addresses**.
2022-04-19	This issue is the thirty-first official release. Added the following content: • **Using LTS to Analyze How WAF Blocks Spring Core RCE Vulnerability in Real Time** • **Using LTS to Configure Block Alarms for WAF Rules**

Released On	Description
2022-04-01	This issue is the thirtieth official release. Added **Java Spring Framework Remote Code Execution Vulnerability**.
2022-03-29	This is the twenty-ninth official release. Added descriptions of some parameters in **Preparations**.
2022-02-11	This issue is the twenty-eighth official release. Added the method of obtaining the origin server IP address when Apache 2.4 or later is used in **Obtaining Real Client IP Addresses**.
2021-12-22	This issue is the twenty-seventh official release. • Added **Using LTS to Quickly Query and Analyze WAF Access Logs**. • Optimized descriptions in **Configuring the Minimum TLS Version and Cipher Suite to Better Secure Connections**.
2021-08-19	This issue is the twenty-sixth official release. Updated some screenshots in **Combining CDN and WAF to Get Improved Protection and Load Speed**.
2021-07-20	This issue is the twenty-fifth official release. Modified the entry to the management console.
2021-06-15	This issue is the twenty-fourth official release. Optimized descriptions in **Combining CDN and WAF to Get Improved Protection and Load Speed**.
2020-04-02	This issue is the twenty-third official release. Updated some screenshots.
2020-02-27	This issue is the twenty-second official release. Updated screenshots and descriptions in **Handling False Alarms to Get Improved Basic Web Protection**
2020-02-14	This issue is the twenty-first official release. Added section **Apache Dubbo Deserialization Vulnerability**.
2020-01-03	This issue is the twentieth official release. Modified the title in **Obtaining Real Client IP Addresses**.

Released On	Description
2019-12-19	This issue is the nineteen official release. • Added the method for the IIS server to obtain the real IP address of a visitor in **Handling False Alarms to Get Improved Basic Web Protection**. • Optimized descriptions in **Combining CDN and WAF to Get Improved Protection and Load Speed**.
2019-12-16	This is the eighteenth official release. • Modified the domain name of Huawei Cloud international website. • Updated the operation entry figure.
2019-12-05	This issue is the seventeenth official release. Optimized descriptions in **Obtaining Real Client IP Addresses**.
2019-10-21	This issue is the sixteenth official release. • Optimized descriptions in **Configuring the Minimum TLS Version and Cipher Suite to Better Secure Connections**. • Optimized the description in **CC Attack Defense** • Optimized descriptions in **Handling False Alarms to Get Improved Basic Web Protection**. • Optimized descriptions in **Combining CDN and WAF to Get Improved Protection and Load Speed**.
2019-09-06	This issue is the fifteenth official release. Added section **DoS Vulnerability in the Open-Source Component Fastjson**.
2019-09-04	This issue is the fourteenth official release. Optimized descriptions in **Connecting a Domain Name to WAF for Websites with no Proxy Used**.
2019-08-30	This issue is the thirteenth official release. Optimized descriptions in **Combining CDN and WAF to Get Improved Protection and Load Speed**.
2019-08-27	This issue is the twelfth official release. Optimized descriptions in **Configuring Anti-Crawler Rules to Prevent Crawler Attacks**.
2019-08-01	This issue is the eleventh official release. Added section **Combining CDN and WAF to Get Improved Protection and Load Speed**.

Released On	Description
2019-07-12	This issue is the tenth official release. Added section **Remote Code Execution Vulnerability of Fastjson**.
2019-06-21	This issue is the ninth official release. Added section **Obtaining Real Client IP Addresses**.
2019-06-04	This issue is the eighth official release. • Added section **Handling False Alarms to Get Improved Basic Web Protection**. • Added section **Domain Setup**.
2019-05-16	This issue is the seventh official release. • Added section **Configuring an Access Control Policy on an ECS or ELB to Protect Origin Servers**. • Added section **Configuring Basic Web Protection**.
2019-05-05	This issue is the sixth official release. Optimized descriptions in **Configuring the Minimum TLS Version and Cipher Suite to Better Secure Connections**.
2019-04-28	This issue is the fifth official release. Added **Configuring the Minimum TLS Version and Cipher Suite to Better Secure Connections**.
2019-04-23	This issue is the fourth official release. • Added section **Oracle WebLogic wls9-async Deserialization Remote Command Execution Vulnerability (CNVD-C-2019-48814)**. • Optimized the description in **CC Attack Defense** • Optimized descriptions in **Configuring Anti-Crawler Rules to Prevent Crawler Attacks**.
2018-11-08	This is the third official release. Optimized some descriptions.
2018-10-15	This is the second official release. Updated screenshots and descriptions.
2018-05-11	This is the first official release.

