Web Application Firewall

Best Practices

HUAWEI TECHNOLOGIES CO., LTD.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Website Access Configuration

1.1 Connecting a Website Without a Proxy to WAF in CNAME Access Mode

Application Scenarios

With the deepening of digital applications, web applications are widely used by most enterprises. Many web applications, such as enterprise websites, online shopping malls, and remote office systems, are publicly accessible. They are becoming major targets of hackers. According to historical data analysis, about 75% of information security attacks target web applications. In addition, web applications and components have more vulnerabilities than others. The critical Log4j vulnerability affected most web applications adversely.

This topic walks you through on how to add your website to WAF in cloud CNAME access mode when no proxies, such as anti-DDoS or CDN products, are used in front of WAF for your website.

Architecture

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.

Figure 1-1 No proxy used
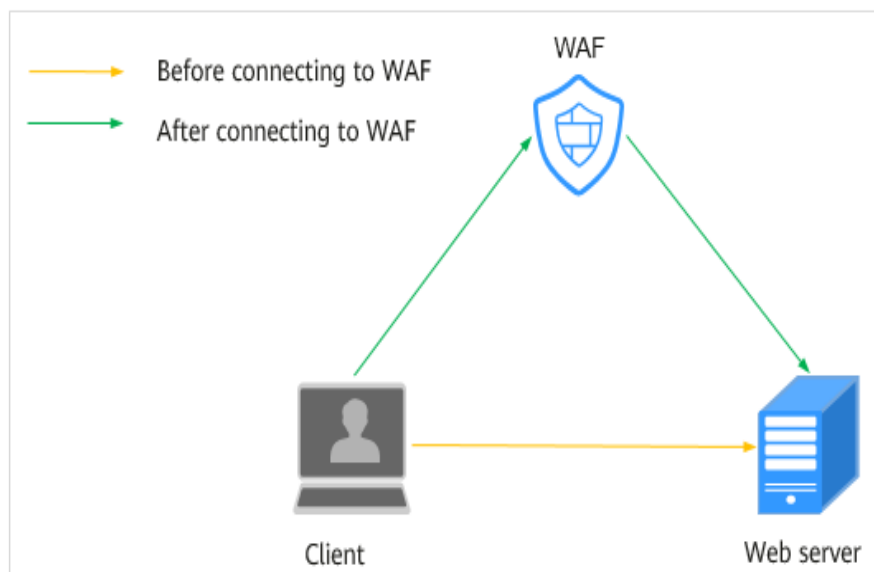
Advantages

After you enable cloud WAF for your website, the website traffic goes through WAF first. WAF examines HTTP/HTTPS requests to identify and block attacks such as SQL injections, cross-site scripting, web shells, command/code injections, file inclusion, sensitive file access, third-party application vulnerability exploits, CC attacks, malicious crawlers, and cross-site request forgery. Then, WAF forwards only legitimate traffic origin servers. In this way, WAF helps keep your website services secure and stable.

Step 1: Buy the Standard Edition Cloud WAF

Step 1 Log in to Huawei Cloud management console.

Step 2 On the management console page, choose **Security** > **Web Application Firewall**.

Step 3 In the upper right corner of the page, click **Buy WAF**. On the purchase page displayed, select **Cloud Mode** for **WAF Mode**.

- **Region**: Select the region nearest to your services WAF will protect.
- **Edition**: Select **Standard**.
- **Expansion Package** and **Required Duration**: Set them based on site requirements.

Step 4 Confirm the product details and click **Buy Now** in the lower right corner of the page.

Step 5 Check the order details and read the *WAF Disclaimer*. Then, select the box and click **Pay Now**.

Step 6 On the payment page, select a payment method and pay for your order.

----End

Step 2: Add Website Information to WAF

Step 1 In the navigation pane on the left, choose **Website Settings**.

Step 2 In the upper left corner of the website list, click **Add Website**.

Step 3 Configure website information as prompted.

Figure 1-2 Configuring basic information
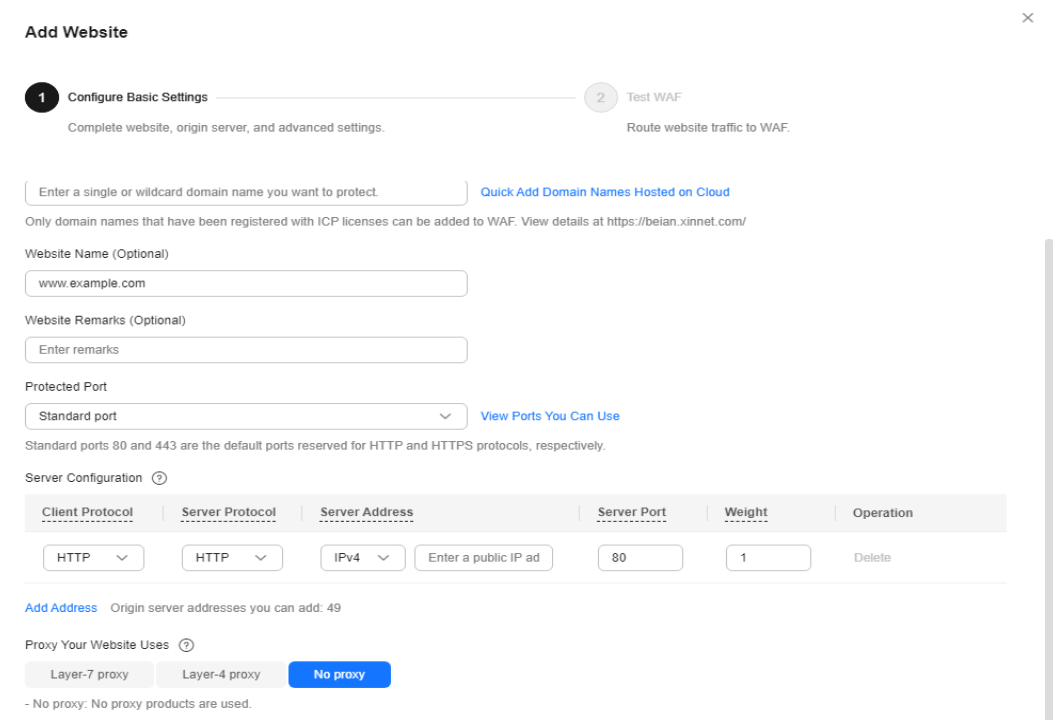
Table 1-1 Key parameters

Parameter	Description	Example Value
Domain Name	Domain name you want to add to WAF. • The domain name has an ICP license. • You can enter a single domain name (for example, top-level domain name **example.com** or level-2 domain name **www.example.com**) or a wildcard domain name (***.example.com**).	www.example.com
Protected Port	The port over which the website service traffic goes	Standard ports

Parameter	Description	Example Value
Server Configuration	Web server address settings. You need to configure the client protocol, server protocol, server weights, server address, and server port. ● **Client Protocol**: protocol used by a client to access a server. The options are **HTTP** and **HTTPS**. ● **Server Protocol**: protocol used by WAF to forward client requests. The options are **HTTP** and **HTTPS**. ● **Server Address**: public IP address (generally corresponding to the A record configured for the domain name on the DNS) or domain name (generally corresponding to the CNAME record configured for the domain name on the DNS) of the web server that a client accesses. ● **Server Port**: service port over which the WAF instance forwards client requests to the origin server. ● **Weight**: Requests are distributed across backend origin servers based on the load balancing algorithm you select and the weight you assign to each server.	**Client Protocol**: Select **HTTP**. **Server Protocol**: **HTTP** **Server Address**: *IPv4 XXX.XXX.1.1* **Server Port**: **80**
Proxy Your Website Uses	You need to configure whether you deploy other proxies in front of WAF. In this example, select **No proxy**.	No proxy

Step 4 Click **Next**. Then, and as prompted.

Figure 1-3 Domain name added to WAF
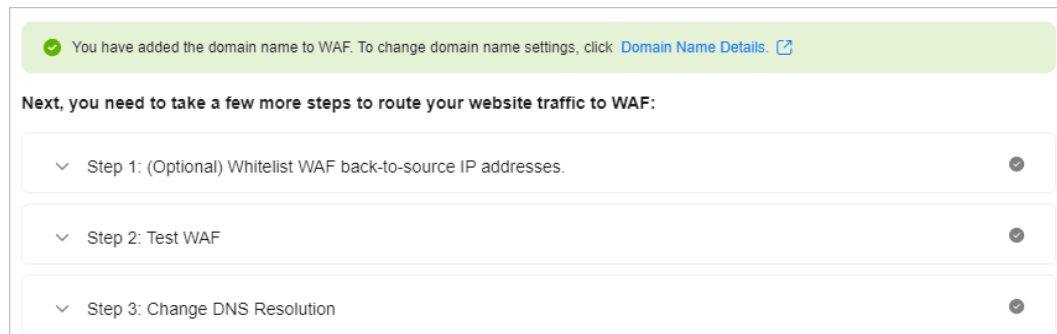
----**End**

Step 3: Complete 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 **Security** > **Web Application Firewall** to go to the **Dashboard** page.

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 1-4 Basic Information
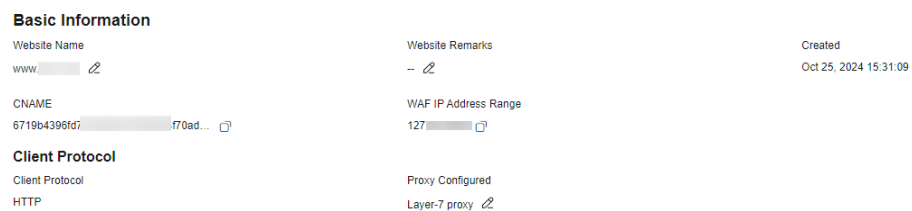
5. In the **CNAME** row, click 📋 to copy the CNAME record.

Step 2 Change the DNS settings.

1. Access the DNS resolution page, as shown in **Figure 1-5**.

Figure 1-5 DNS page
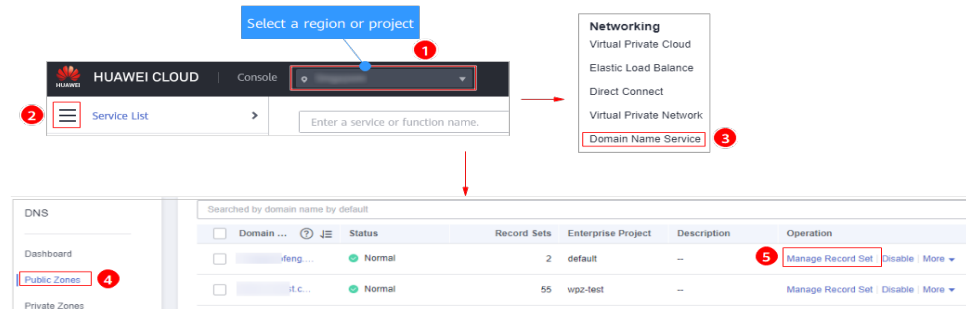
2. In the **Operation** column of the target domain name, click **Modify**. The **Modify Record Set** page is displayed.

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 1-6 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

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

Application Scenarios

With the deepening of digital applications, web applications are widely used by most enterprises. Many web applications, such as enterprise websites, online shopping malls, and remote office systems, are publicly accessible. They are becoming major targets of hackers. According to historical data analysis, about 75% of information security attacks target web applications. In addition, web applications and components have more vulnerabilities than others. The critical Log4j vulnerability affected most web applications adversely.

If your website has used CDN already, you can use WAF as well to give extra protection to the website.

Architecture

- 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) injections, 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).

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

Figure 1-7 WAF configuration when a proxy is used
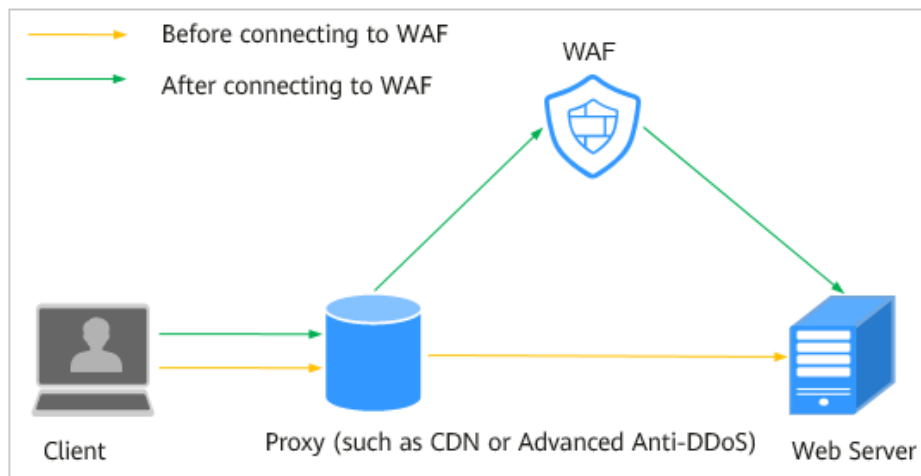
After you deploy CDN and WAF for your website, traffic is accelerated by CDN and then forwarded to WAF. WAF checks received traffic and forwards only the normal traffic to the origin server. The combination protects the website against attacks while improving the website response speed and availability.

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.

Advantages

With both CDN and WAF deployed, your website will be accelerated with CDN while being protected with WAF. This combination will shorten website content access delay, speed up website response, and improve website availability. You can stop worrying about low network bandwidth, large user access traffic, and uneven distribution of branches. Besides that, this combination will protect your website from web application attacks, such as SQL injections, cross-site scripting (XSS), web shells, command/code injections, file inclusion, sensitive file access, third-party application vulnerability exploits, CC attacks, malicious crawlers, and cross-site request forgery.

Resource and Cost Planning

Table 1-2 Resources and costs

Resource	Description	Monthly Fee
CDN	• **Billing Mode**: Select **Pay-per-use**. • Resource packages can be used.	For details about billing rules, see **Billing Description**.
Web Application Firewall	Cloud - Standard edition • Billing mode: Yearly/ Monthly • Domain name quota: 10, including a maximum of one top-level domain name • QPS quota: 2,000 QPS • Peak bandwidth: 100 Mbit/s inside the cloud and 30 Mbit/s outside the cloud	For details about pricing rules, see **Billing Description**.

Constraints

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

Step 1: Buy the Standard Edition Cloud WAF

The following describes how to buy the standard edition cloud WAF.

Step 1　Log in to Huawei Cloud management console.

Step 2　On the management console page, choose **Security** > **Web Application Firewall**.

Step 3　In the upper right corner of the page, click **Buy WAF**. On the purchase page displayed, select **Cloud Mode** for **WAF Mode**.

- **Region**: Select the region nearest to your services WAF will protect.
- **Edition**: Select **Standard**.
- **Expansion Package** and **Required Duration**: Set them based on site requirements.

Step 4　Confirm the product details and click **Buy Now** in the lower right corner of the page.

Step 5　Check the order details and read the *WAF Disclaimer*. Then, select the box and click **Pay Now**.

Step 6　On the payment page, select a payment method and pay for your order.

----End

Step 2: Add Website Information to WAF

The following example shows how to add a website information to WAF in cloud CNAME access mode.

- For details about the cloud load balancer access mode, see .
- For details about the dedicated mode, see .

Step 1　In the navigation pane on the left, choose **Website Settings**.

Step 2　In the upper left corner of the website list, click **Add Website**.

Step 3　Configure website information as prompted.

Figure 1-8 Configuring basic information
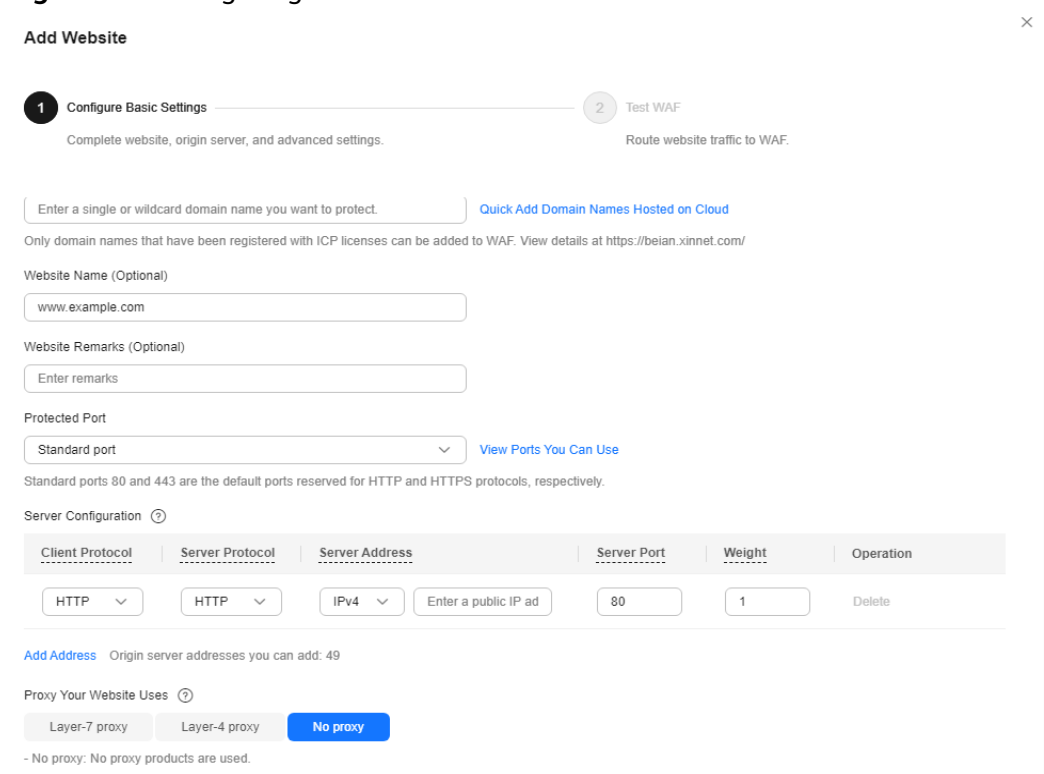
Table 1-3 Key parameters

Parameter	Description	Example Value
Domain Name	Domain name you want to add to WAF for protection. ● The domain name has an ICP license. ● You can enter a single domain name (for example, top-level domain name **example.com** or level-2 domain name **www.example.com**) or a wildcard domain name (***.example.com**).	www.example.com
Protected Port	The port over which the website traffic goes	Standard ports

Parameter	Description	Example Value
Server Configuration	Web server address settings. You need to configure the client protocol, server protocol, server weights, server address, and server port. ● **Client Protocol**: protocol used by a client to access a server. The options are **HTTP** and **HTTPS**. ● **Server Protocol**: protocol used by WAF to forward client requests. The options are **HTTP** and **HTTPS**. ● **Server Address**: public IP address (generally corresponding to the A record of the domain name configured on the DNS) or domain name (generally corresponding to the CNAME record of the domain name configured on the DNS) of the web server that a client accesses. ● **Server Port**: service port over which the WAF instance forwards client requests to the origin server. ● **Weight**: Requests are distributed across backend origin servers based on the load balancing algorithm you select and the weight you assign to each server.	**Client Protocol**: Select **HTTP**. **Server Protocol**: **HTTP** **Server Address**: *IPv4 XXX.XXX.1.1* **Server Port**: **80**
Proxy Your Website Uses	You need to configure whether you deploy other proxies in front of WAF. In this example, select **Layer-7 proxy**.	Layer-7 proxy

Step 4 Click **Next**. Then, and as prompted.

Figure 1-9 Domain name added to WAF
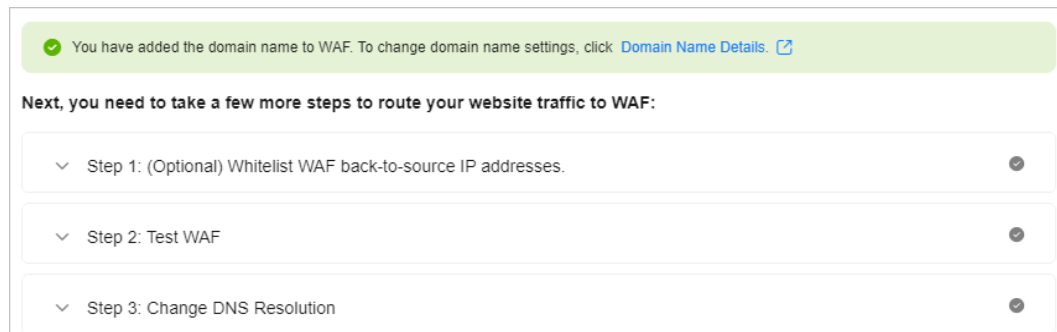
----End

Step 3: Resolve the Domain Name

On the CDN page, add the CNAME record of WAF to let the traffic pass through WAF.

How to Configure in Cloud Mode

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. Choose **Security** > **Web Application Firewall** to go to the **Dashboard** page.

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 basic information page for the domain name, click in the **CNAME** row and copy the CNAME records. On the top of the page, click next to **Inaccessible**. In the dialog box displayed, copy the 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 1-10**.

Figure 1-10 DNS page
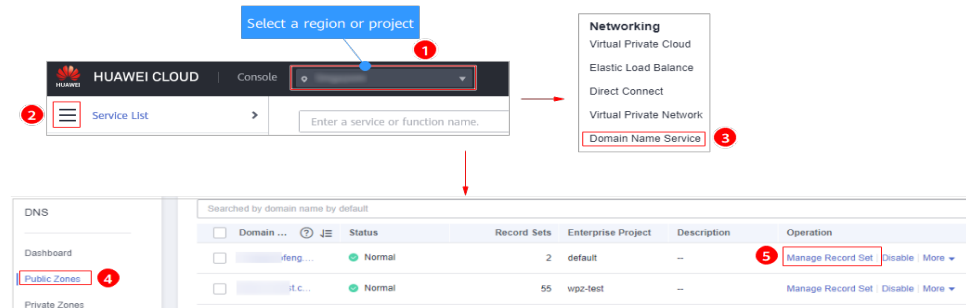
2. In the upper right corner of the page, click **Add Record Set**. The **Add Record Set** page is displayed. **Figure 1-11** shows an example.

    –   **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 1-11 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

Configuration of Dedicated WAF

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 **Server Address** text box, **enter the EIP you bind to the load balancer**.

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.

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?**

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

Application Scenarios

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**.

Architecture

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.
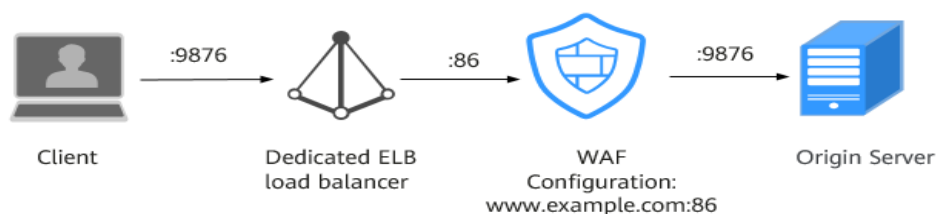
Advantages

This solution makes it possible for WAF to protect your services over any ports.

Resource and Cost Planning

Table 1-4 Resources and costs

Resource	Description	Monthly Fee
Elastic Load Balance (ELB)	- Billing mode: Yearly/Monthly - Instance type: Dedicated - Specifications: application type (HTTP/HTTPS); small II **Billed By**: **Bandwidth** **Bandwidth**: 10 Mbit/s	For details about billing rules, see **Billing Description**.

Resource	Description	Monthly Fee

Web Application Firewall	Dedicated mode:
	
	● **Billing Mode**: Select **Pay-per-use**.
	
	● Number of domain names: 2,000 (2,000 top-level domain names are supported.)
	
	● Specifications: WI-500. Referenced performance:
	
	  – HTTP services - Recommended QPS: 5,000. Maximum QPS: 10,000.
	
	  – HTTPS services - Recommended QPS: 4,000. Maximum QPS: 8,000.
	
	  – WebSocket service - Maximum concurrent connections: 5,000
	
	  – Maximum WAF-to-server persistent connections: 60,000
	
	● Specifications: WI-100. Referenced performance:
	
	  – HTTP services - Recommended QPS: 1,000. Maximum QPS: 2,000.
	
	  – HTTPS services - Recommended QPS: 800. Maximum QPS: 1,600
	
	  – WebSocket service - Maximum concurrent connections: 1,000
	
	  – Maximum WAF-to-server persistent connections: 60,000
		-

Prerequisites

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

  📖 **NOTE**

  Dedicated WAF instances issued before April 2023 cannot be used with dedicated network load balancers. If you use a dedicated network (TCP/UDP) load balancer, make sure your dedicated WAF instance has been upgraded to the latest version (version later than 202304). You can check the version you are using in the **Version** column of the target dedicated WAF instance on the **Dedicated Engine** page.

- 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.

Procedure

Step 1 Log in to Huawei Cloud management console.

Step 2 On the management console page, choose **Security** > **Web Application Firewall**.

Step 3 In the navigation pane on the left, choose **Website Settings**.

Step 4 Click **Add Website**.

Step 5 Connect www.example.com to WAF by referring to **Adding a Website to WAF (Dedicated Mode)**. Select any non-standard port as the protected port, for example, port 86, set **Server Port** to **9876**, and set **Proxy Your Website Uses** to **Layer-7 proxy**.

Step 6 Add listeners and backend server groups to the load balancer.

1. Click ☰ in the upper left corner of the page and choose **Elastic Load Balance** under **Networking** to go to the **Load Balancers** page.

2. Click the name of the load balancer in the **Name** column to go to the **Basic Information** page.

3. 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 1-12 Configuring a listener

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

Figure 1-13 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**.

---

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

Step 7 Add the WAF instance to the load balancer.

1. Choose **Security** > **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.

Figure 1-14 Dedicated engine list

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** based on **Step 6**.

Figure 1-15 Add to ELB

5. 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 5**.

Figure 1-16 Configuring Backend Port

6. Click **Confirm**.

Step 8 **Bind an EIP to a Load Balancer**.

Step 9 **Whitelist IP addresses of your dedicated WAF instances**.

----End

2 Website Protection Configuration Suggestions

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
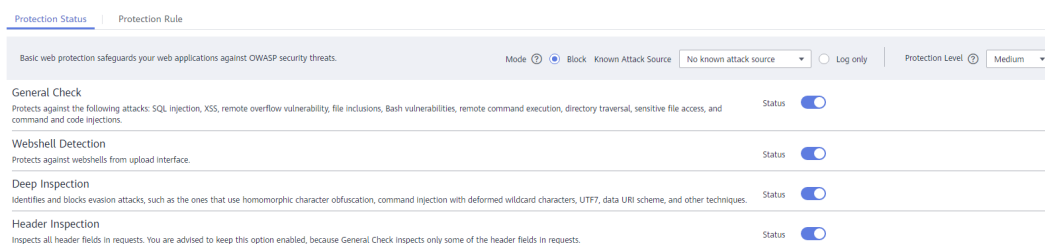
Mitigation

Step 1 **Buy a WAF instance**.

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** and complete the configuration. 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 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 **Configuring Basic Web Protection Rules**.

----End

3.4 Remote Code Execution Vulnerability of Fastjson

On July 12, 2019, the 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 **Configuring Basic Web Protection Rules**.

----End

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

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 Defending Against Challenge Collapsar (CC) Attacks

4.1 Limiting Accesses Through 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 rules 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 details about how to connect a domain name to WAF, see **Adding a Domain Name**.

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 4-1 CC Attack Protection configuration area
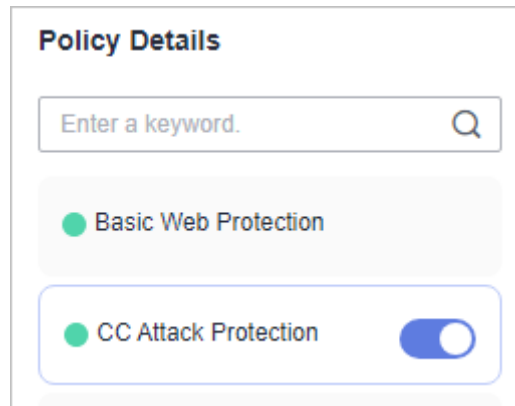
4. Then, add a CC attack protection rule to limit the rate of request traffic destined for the domain name. Set **Rate Limit Mode** to **Per IP address**, **Rate Limit** based on your service features, and **Protective Action** to **Verification code** to prevent blocking legitimate users.

   – **Rate Limit Mode**: Select **Per IP address** to distinguish a single web visitor based on IP addresses.

   – **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.

   – **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.

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

   – Set other parameters based on site requirements.

Figure 4-2 Per IP address
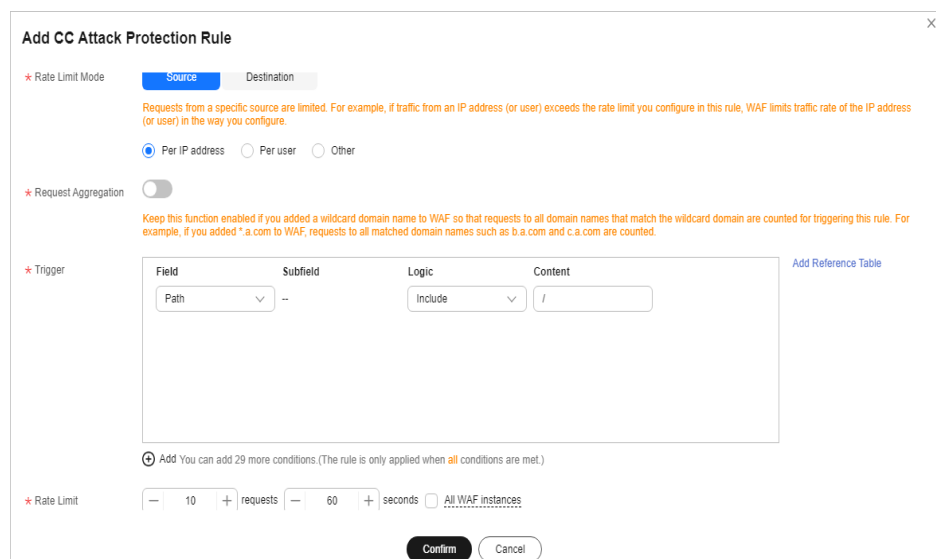
5. In the navigation pane on the left, choose **Events**. You can view details about attack events.

4.2 Limiting Accesses Through Cookie Field Configuration

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 was hit by CC attacks.

2. Log in to the management console and route website traffic to WAF. For details about how to connect a domain name to WAF, see **Adding a Domain Name**.

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 4-3 CC Attack Protection configuration area
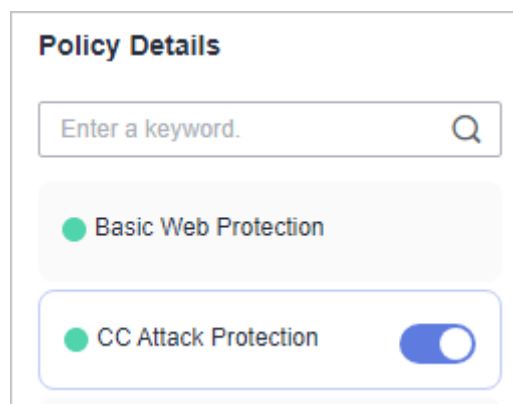
4. Add a CC attack protection rule, as shown in **Figure 4-4**.
   – **Rate Limit Mode**: Select **Per user** to distinguish a single web visitor based on cookies.

- **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.

- **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**.

Figure 4-4 Add CC Attack Protection Rule

5. In the navigation pane on the left, choose **Events**. You can view details about attack events.

4.3 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. For details, see **Adding a Domain Name to WAF**.

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 4-5 CC Attack Protection configuration area
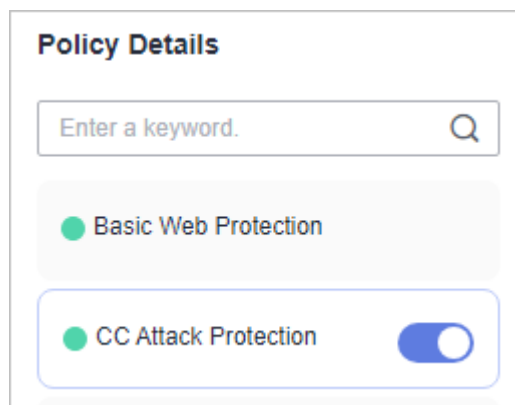
Step 4 In the upper left corner of the **CC Attack Protection** page, click **Add Rule**.

Step 5 Configure a CC attack protection rule using a cookie or user ID to limit traffic to the path. **Figure 4-6** shows an example.

- **Rate Limit Mode**: Select **Source** and then **Per user**.

- **User Identifier**: Select **Cooke** and enter the User ID as the key value.

- **Trigger**: Set **Field** to **Path**, and set **Logic** and **Content** based on site requirements.

- Other parameters: Set them to meet your service requirements.

Figure 4-6 Configuring service cookies

Step 6 Click **Confirm**.

----End

Using HWWAFSESID to Configure a CC Attack Protection Rule

HWWAFSESID: session ID. WAF inserts HWWAFSESID (session ID) into the cookie of a customer request. WAF uses this field to count client requests. If the number of requests reaches the threshold, the CC attack protection rule will be triggered. Now, let's see how to use this field to configure a CC attack protection rule.

Step 1 Log in to the management console and connect your website to WAF. For details, see **Adding a Domain Name to WAF**.

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 4-7 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 4-8**.

- **Rate Limit Mode**: Select **Source** and then **Per user**.
- **User Identifier**: Select **Cookie** and set it to **HWWAFSESID**.
- **Trigger**: Set **Field** to **Path**, and set **Logic** and **Content** based on site requirements.
- Other parameters: Set them to meet your service requirements.

Figure 4-8 HWWAFSESID-based rate limiting

Step 6 Click **Confirm**.

----End

5 Using WAF to Block Crawler Attacks

Application Scenarios

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.

To comprehensively mitigate crawler attacks against websites, WAF provides three anti-crawler policies, general check and web shell detection by identifying User-Agent, website anti-crawler by checking browser validity, and CC attack protection by limiting the access traffic rate.

Overview

Figure 5-1 shows how WAF detects crawlers, where step 1 and step 2 are called JS challenges and step 3 is called JS authentication.

Figure 5-1 JavaScript Anti-Crawler protection process
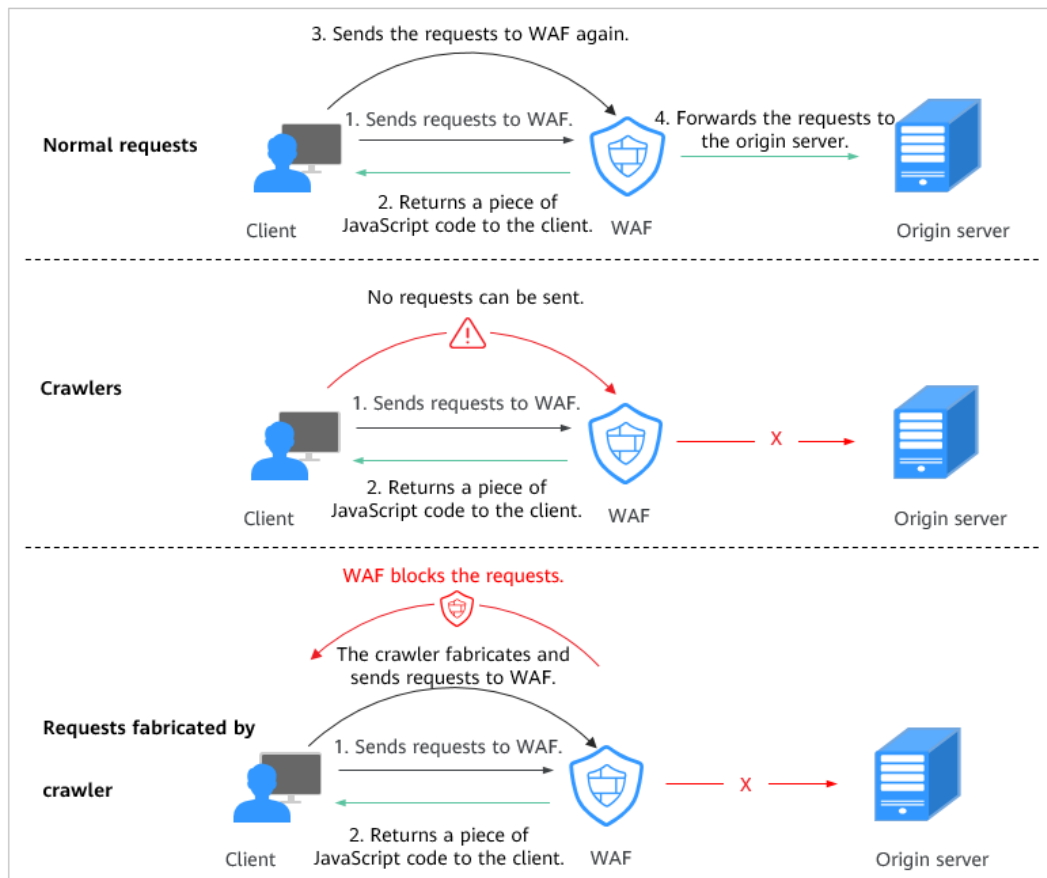
If JavaScript anti-crawler is enabled when a client sends a request, WAF returns a piece of JavaScript code to the client.

● If the client sends a normal request to the website, triggered by the received JavaScript code, the client will automatically send the request to WAF again. WAF then forwards the request to the origin server. This process is called JavaScript verification.

● If the client is a crawler, it cannot be triggered by the received JavaScript code and will not send a request to WAF again. The client fails JavaScript authentication.

● If a client crawler fabricates a WAF authentication request and sends the request to WAF, the WAF will block the request. The client fails JavaScript authentication.

By collecting statistics on the number of JavaScript challenges and authentication responses, the system calculates how many requests the JavaScript anti-crawler defends. In **Figure 5-2**, the JavaScript anti-crawler has logged 18 events, 16 of which are JavaScript challenge responses, and 2 of which are JavaScript authentication responses. **Other** indicates the number of WAF authentication requests fabricated by the crawler.

Figure 5-2 Parameters of a JavaScript anti-crawler protection rule
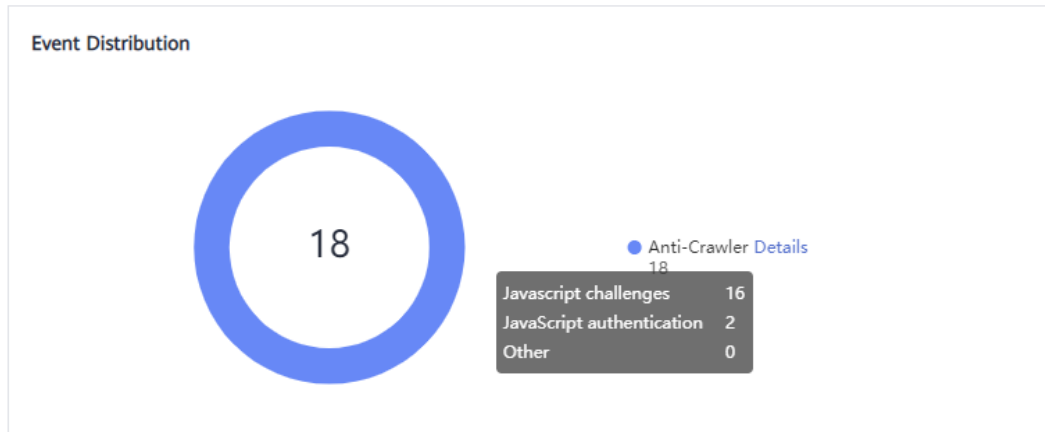
Limitations and Constraints

- 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.

Resource and Cost Planning

Table 5-1 Resources and costs

Resource	Description	Monthly Fee
Web Application Firewall	Cloud - professional edition: • Billing mode: Yearly/Monthly • Domain name quota: 50, including a maximum of five top-level domain names • QPS quota: 5,000 QPS • Peak bandwidth: 200 Mbit/s inside the cloud and 50 Mbit/s outside the cloud	For details about pricing rules, see **Billing Description**.

Step 1: Buy the Professional Edition Cloud WAF

The following describes how to buy the standard edition cloud WAF.

Step 1 Log in to Huawei Cloud management console.

Step 2 On the management console page, choose **Security** > **Web Application Firewall**.

Step 3 In the upper right corner of the page, click **Buy WAF**. On the purchase page displayed, select **Cloud Mode** for **WAF Mode**.

- **Region**: Select the region nearest to your services WAF will protect.

- **Edition**: Select **Profesional**.

- **Expansion Package** and **Required Duration**: Set them based on site requirements.

Step 4 Confirm the product details and click **Buy Now** in the lower right corner of the page.

Step 5 Check the order details and read the *WAF Disclaimer*. Then, select the box and click **Pay Now**.

Step 6 On the payment page, select a payment method and pay for your order.

----End

Step 2: Add Website Information to WAF

The following example shows how to add a website information to WAF in cloud CNAME access mode.

Step 1 In the navigation pane on the left, choose **Website Settings**.

Step 2 In the upper left corner of the website list, click **Add Website**.

Step 3 Select **Cloud - CNAME** and click **Configure Now**.

Step 4 Configure website information as prompted.

Figure 5-3 Configuring basic information

Table 5-2 Key parameters

Parameter	Description	Example Value
Domain Name	Domain name you want to add to WAF for protection. ● The domain name has an ICP license. ● You can enter a single domain name (for example, top-level domain name **example.com** or level-2 domain name **www.example.com**) or a wildcard domain name (***.example.com**).	www.example.com
Protected Port	The port over which the website traffic goes	Standard ports

Parameter	Description	Example Value
Server Configuration	Web server address settings. You need to configure the client protocol, server protocol, server weights, server address, and server port. ● **Client Protocol**: protocol used by a client to access a server. The options are **HTTP** and **HTTPS**. ● **Server Protocol**: protocol used by WAF to forward client requests. The options are **HTTP** and **HTTPS**. ● **Server Address**: public IP address (generally corresponding to the A record of the domain name configured on the DNS) or domain name (generally corresponding to the CNAME record of the domain name configured on the DNS) of the web server that a client accesses. ● **Server Port**: service port over which the WAF instance forwards client requests to the origin server. ● **Weight**: Requests are distributed across backend origin servers based on the load balancing algorithm you select and the weight you assign to each server.	**Client Protocol**: Select **HTTP**. **Server Protocol**: **HTTP** **Server Address**: *IPv4 XXX.XXX.1.1* **Server Port**: **80**
Proxy Your Website Uses	You need to configure whether you deploy other proxies in front of WAF. Set this parameter based on your website deployment.	Layer-7 proxy

Step 5 Click **Next**. Then, , , and as prompted.

Figure 5-4 Domain name added to WAF
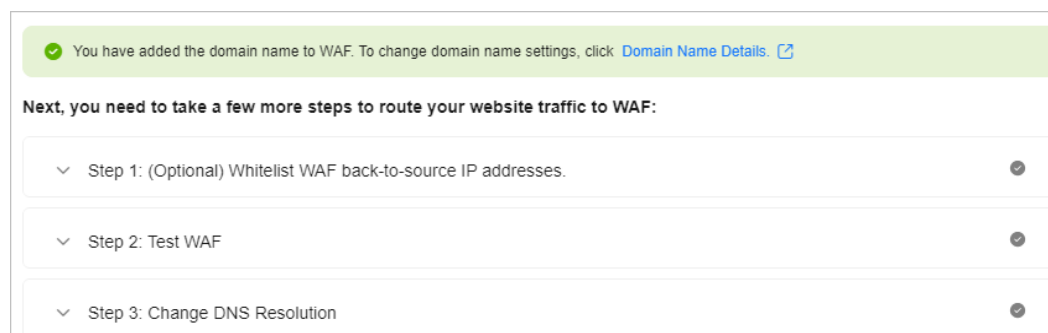
----**End**

Step 3: Enable General Check and Web Shell Detection (Identifying User-Agent)

General check and web shell detection in WAF can help detect and block threats such as malicious crawlers and web shells.

Step 1 Log in to the management console.

Step 2 Choose **Security** > **Web Application Firewall** to go to the **Dashboard** page.

Step 3 In the navigation pane on the left, choose **Website Settings**.

Step 4 In the **Policy** column of the row containing the domain name, click the number to go to the **Policies** page.

Step 5 Ensure that **Basic Web Protection** is enabled (status:).

Figure 5-5 Basic Web Protection configuration area
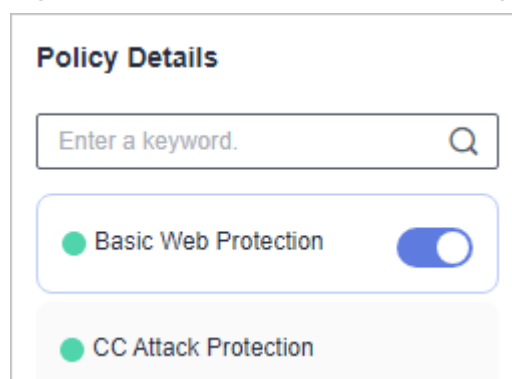
Step 6 On the **Protection Status** page, enable **General Check** and **Webshell Detection**.

Figure 5-6 Protection configuration
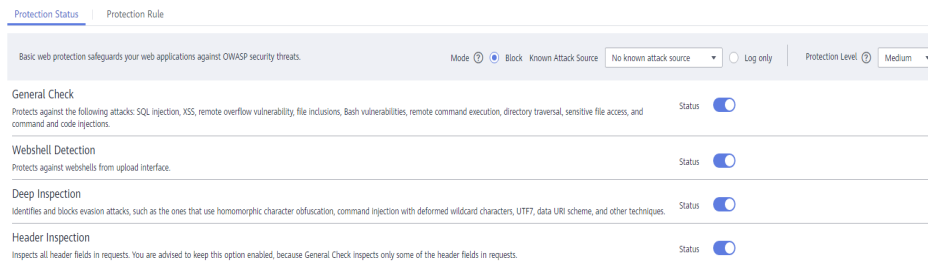
If WAF detects that a malicious crawler is crawling your website, WAF immediately blocks it and logs the event. You can view the crawler protection logs on the **Events** page.

----End

Step 4: Enable 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 Click the **Anti-Crawler** configuration area and toggle it on.

- : enabled.

- : disabled.

Step 2 On the **Feature Library** page, enable **Scanner** and set other parameters based on your service requirements.

Figure 5-7 Feature Library
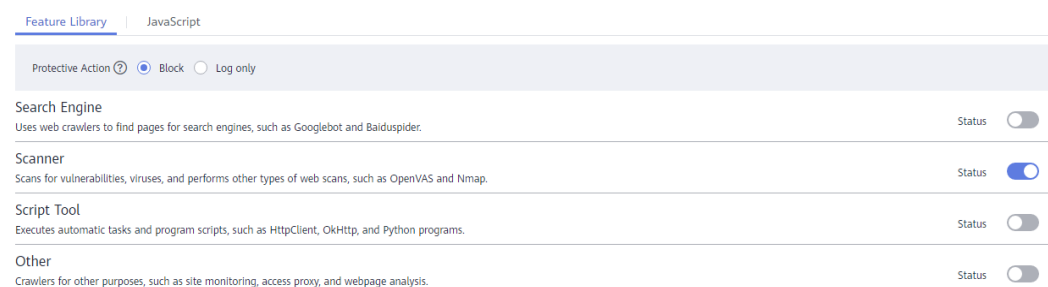
Step 3 Select the **JavaScript** tab and change **Status** if needed.

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

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 4 Configure a JavaScript-based anti-crawler rule by referring to **Table 5-3**.

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

- To protect all paths except a specified path

  Set **Protection Mode** to **Protect all paths**. Then, click **Exclude Path**, configure protected paths, and click **Confirm**.

  Figure 5-8 Exclude Rule

- To protect a specified path only

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

Figure 5-9 Add Rule

Table 5-3 Parameters of a JavaScript-based anti-crawler protection rule

Parameter	Description	Example Value
Rule Name	Name of the rule	wafjs

Parameter	Description	Example Value
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
Logic	Select a logical relationship from the drop-down list.	Include
Rule Description	A brief description of the rule.	None
Effective Date	Immediate	Immediate

----End

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

```
</body>
</html>
[root@VM_0_9_centos ~]# curl http://cloudsecuritylab.tech/ --user-agent 'xpymep.exe'
<html>
<head>
<meta http-equiv="Server" content="HuaweiCloudWAF">
</head>
<body>
<noscript>
<h1><strong>Please Enable JavaScript and Cookie.</strong></h1>
</noscript>
<h1> <span id = "open_cookie"/> </h1>
<script type="text/javascript">
(function(t,a,c){"function"===typeof window.define&&window.define.amd?window.define(c):"undefined"!=
=typeof module&&module.exports?module.exports=c():a.exports?a.exports=c():a[t]=c()})("fe",this,funct
ion(){var t=function(a){if(!(this instanceof t))return new t(a);this.options=this.extend(a,{swfConta
inerId:"fingerprintjs2",swfPath:"flash/compiled/FontList.swf",detectScreenOrientation:!0,sortPlugins
For:[/palemoon/i],userDefinedFonts:[]});this.nativeForEach=Array.prototype.forEach;this.nativeMap=Ar
ray.prototype.map};
t.prototype={extend:function(a,c){if(null==a)return c;for(var b in a)null!=a[b]&&c[b]!==a[b]&&(c[b]=
a[b]);return c},get:function(a){var c=this,b={data:[],addPreprocessedComponent:function(a){var b=a.v
alue;"function"===typeof c.options.preprocessor&&(b=c.options.preprocessor(a.key,b));this.data.push(
{key:a.key,value:b})}},b=this.userAgentKey(b),b=this.languageKey(b),b=this.colorDepthKey(b),b=this.d
eviceMemoryKey(b),b=this.pixelRatioKey(b),b=this.hardwareConcurrencyKey(b),b=this.screenResolutionKe
y(b),
b=this.availableScreenResolutionKey(b),b=this.timezoneOffsetKey(b),b=this.platformKey(b),b=this.plug
```

Step 5: Configure 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 Ensure that the **Status** of **CC Attack Protection** is **enabled** ( ).

Figure 5-10 CC Attack Protection configuration area
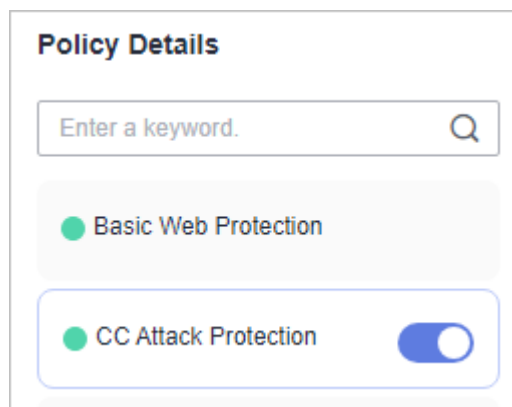
Step 2 In the upper left corner above the **CC Attack Protection** rule list, click **Add Rule**. The following uses IP address-based rate limiting and human-machine verification as examples to describe how to add an IP address-based rate limiting rule, as shown in **Figure 5-11**.

Figure 5-11 Per IP address
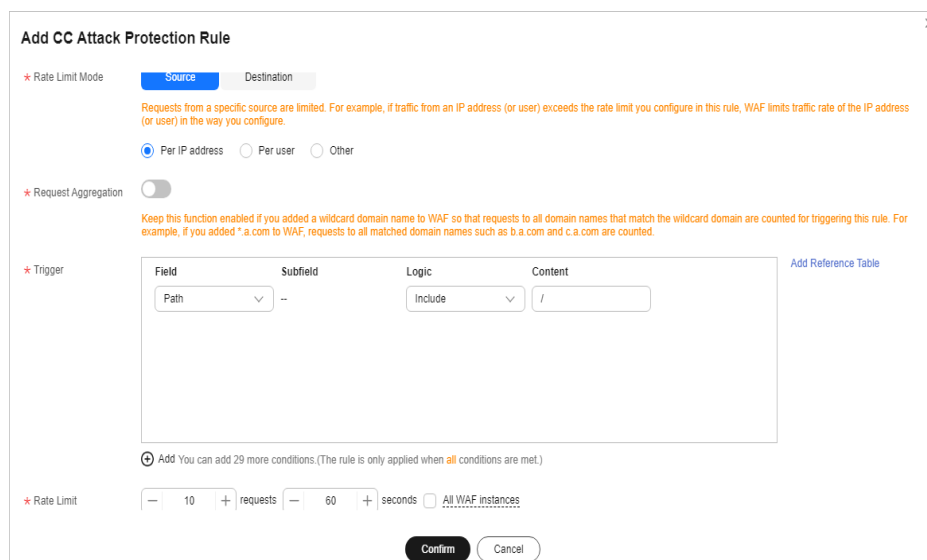
If the number of access requests exceeds the configured rate limit, the visitors are required to enter a verification code to continue the access.
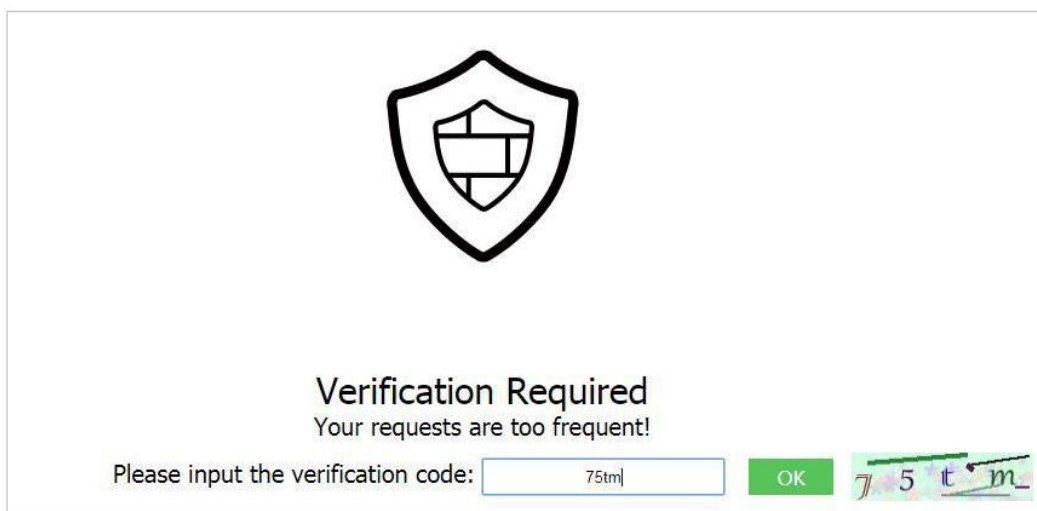
----End

6 Verifying a Global Protection Whitelist Rule by Simulating Requests with Postman

Application Scenarios

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 rule.

Example

Assume that your workloads are deployed in the **/product** directory, and parameter ID contains scripts or text in Rich Text Format (RTF) 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.

Resource and Cost Planning

Table 6-1 Resources and costs

Resource	Description	Monthly Fee
Web Application Firewall	Cloud - Standard edition ● Billing mode: Yearly/Monthly ● Domain name quota: 10, including a maximum of one top-level domain name ● QPS quota: 2,000 QPS ● Peak bandwidth: 100 Mbit/s inside the cloud and 30 Mbit/s outside the cloud	For details about pricing rules, see **Billing Description**.

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.

Step 1: Buy the Standard Edition Cloud WAF

The following describes how to buy the standard edition cloud WAF.

Step 1 Log in to Huawei Cloud management console.

Step 2 On the management console page, choose **Security** > **Web Application Firewall**.

Step 3 In the upper right corner of the page, click **Buy WAF**. On the purchase page displayed, select **Cloud Mode** for **WAF Mode**.

- **Region**: Select the region nearest to your services WAF will protect.
- **Edition**: Select **Standard**.
- **Expansion Package** and **Required Duration**: Set them based on site requirements.

Step 4 Confirm the product details and click **Buy Now** in the lower right corner of the page.

Step 5 Check the order details and read the *WAF Disclaimer*. Then, select the box and click **Pay Now**.

Step 6 On the payment page, select a payment method and pay for your order.

----End

Step 2: Add Website Information to WAF

The following example shows how to add a website information to WAF in cloud CNAME access mode.

Step 1 In the navigation pane on the left, choose **Website Settings**.

Step 2 In the upper left corner of the website list, click **Add Website**.

Step 3 Select **Cloud - CNAME** and click **Configure Now**.

Step 4 Configure website information as prompted.

Figure 6-1 Configuring basic information
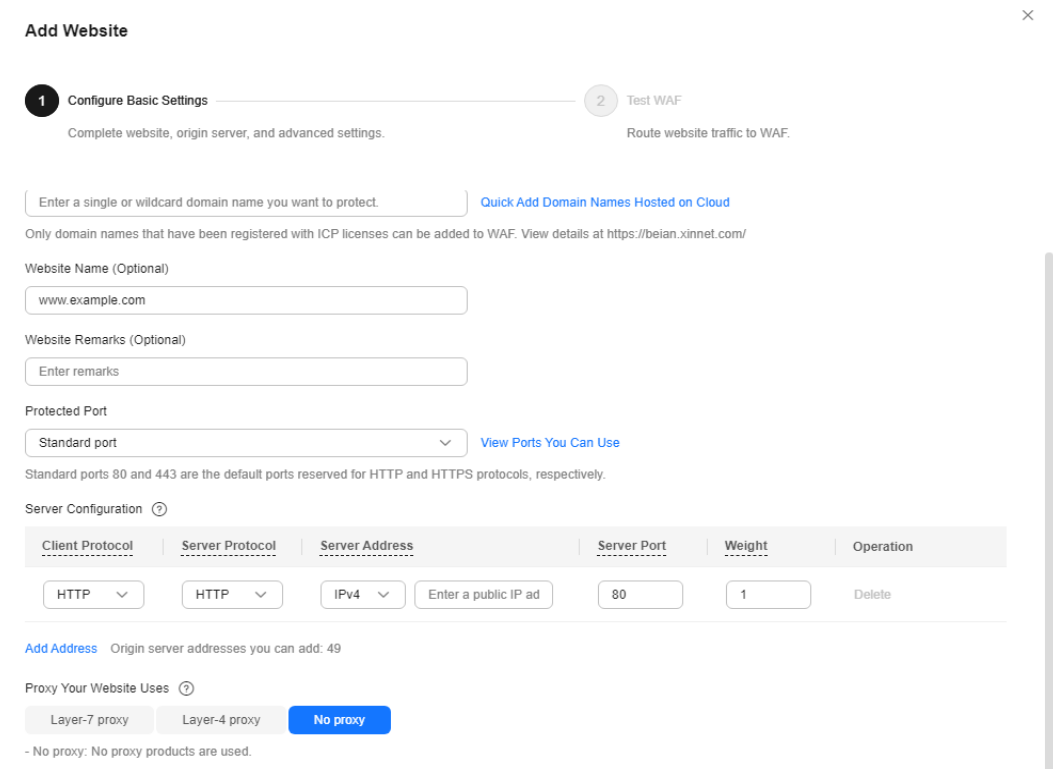
Table 6-2 Key parameters

Parameter	Description	Example Value
Domain Name	Domain name you want to add to WAF for protection. ● The domain name has an ICP license. ● You can enter a single domain name (for example, top-level domain name **example.com** or level-2 domain name **www.example.com**) or a wildcard domain name (***.example.com**).	www.example.com
Protected Port	The port over which the website traffic goes	Standard ports

Parameter	Description	Example Value
Server Configuration	Web server address settings. You need to configure the client protocol, server protocol, server weights, server address, and server port. ● **Client Protocol**: protocol used by a client to access a server. The options are **HTTP** and **HTTPS**. ● **Server Protocol**: protocol used by WAF to forward client requests. The options are **HTTP** and **HTTPS**. ● **Server Address**: public IP address (generally corresponding to the A record of the domain name configured on the DNS) or domain name (generally corresponding to the CNAME record of the domain name configured on the DNS) of the web server that a client accesses. ● **Server Port**: service port over which the WAF instance forwards client requests to the origin server. ● **Weight**: Requests are distributed across backend origin servers based on the load balancing algorithm you select and the weight you assign to each server.	**Client Protocol**: Select **HTTP**. **Server Protocol**: **HTTP** **Server Address**: *IPv4 XXX.XXX.1.1* **Server Port**: **80**
Proxy Your Website Uses	You need to configure whether you deploy other proxies in front of WAF. Set this parameter based on your website deployment.	Layer-7 proxy

Step 5 Click **Next**. Then, , , and as prompted.

Figure 6-2 Domain name added to WAF
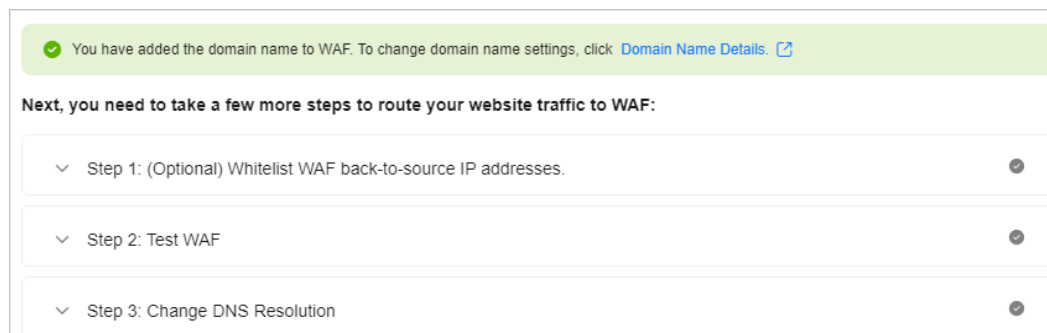
    ----End

Step 3: Verify a Global Protection Whitelist Rule by Simulating Requests with Postman

    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. Choose **Security** > **Web Application Firewall** to go to the **Dashboard** page.

        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 **Handle as False Alarm**.

        7. In the **Handle False Alarm** dialog box, add a global protection whitelist rule as shown in **Figure 6-3**.

Figure 6-3 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 6-4 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 6-5 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.
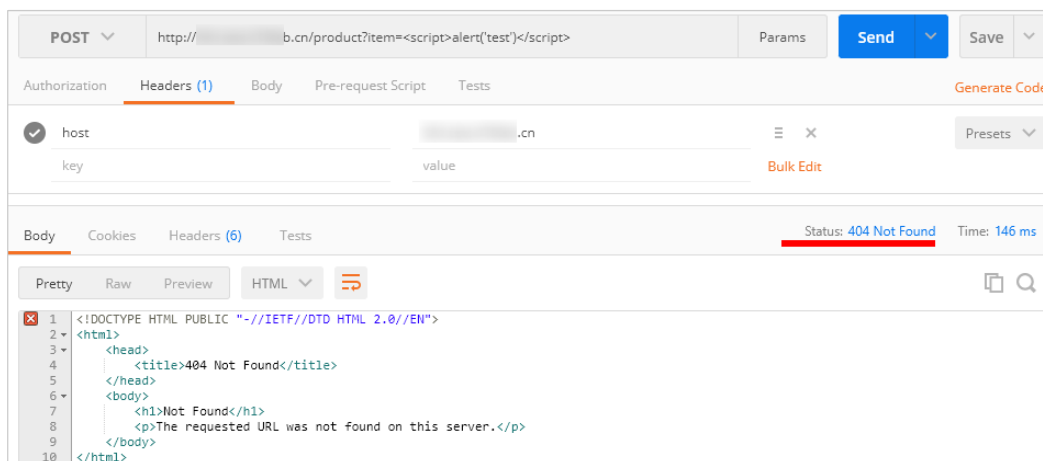
Go to the **Event** page, no new XSS attack event is displayed.

Step 9 Simulate an attack on Postman to verify that the configured global protection whitelist 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 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

7 Combining WAF and HSS to Improve Web Page Tampering Protection

Application Scenarios

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.

This topic describes how to use the combination of WAF and HSS to protect dynamic and static web pages from being tampered with.

Solution Architecture and Advantages

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.

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.
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

Step 1: Configure a Web Tamper Protection Rule in WAF

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 **Security** > **Web Application Firewall**.

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 7-1 Adding a web tamper protection rule

Add Web Tamper Protection Rule

* Domain Name   www.example.com

* Path   /admin

Rule Description

Confirm   Cancel

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

Step 2: Enable 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 **Server Protection** > **Web Tamper Protection**. On the **Web Tamper Protection** page, click **Add Server**.

Figure 7-2 Adding a protected server
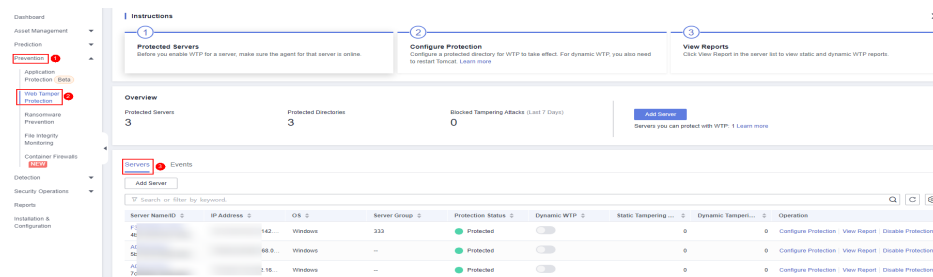
Step 4 On the **Add Server** page, click the **Available servers** tab. Select the target server, select a quota from the drop-down list or retain the default value, and click **Add and Enable Protection**.

Figure 7-3 Selecting a server to enable protection
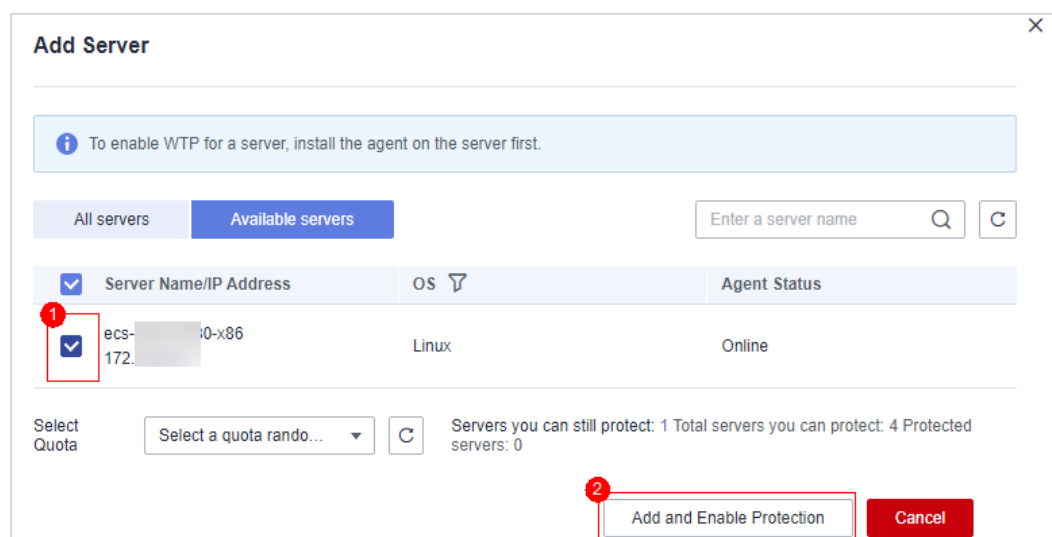
Step 5 View the server status on the **Web Tamper Protection** page.

- Choose **Server Protection** > **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 **Web Tamper Protection**, the WTP edition is enabled.

----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 Configuring Origin Server Security

8.1 Using WAF to Improve Connection Security

Application Scenarios

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.

Resource and Cost Planning

Table 8-1 Resources and costs

Resource	Description	Monthly Fee
Web Application Firewall	Cloud - Standard edition ● Billing mode: Yearly/Monthly ● Domain name quota: 10, including a maximum of one top-level domain name ● QPS quota: 2,000 QPS ● Peak bandwidth: 100 Mbit/s inside the cloud and 30 Mbit/s outside the cloud	For details about pricing rules, see **Billing Description**.

Minimum TLS Versions Supported

The default minimum TLS version configured in WAF is **TLS v1.0**. To better secure your website, configure an appropriate TLS version. **Table 8-2** lists the minimum TLS versions supported for different scenarios.

Table 8-2 Minimum TLS versions supported

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.

Cipher Suites Supported

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 8-3**.

Table 8-3 Description of cipher suites

Cipher Suite Name	Cryptographic Algorithm Supported	Cryptographic Algorithm Not Supported	Description
Default cipher suite **NOTE** By default, **Cipher suite 1** is configured for websites. However, if the request does not carry the server name indication (SNI), WAF uses the **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 • 3DES • DSS • EXP • CAMELLIA	Recommended configuration. • Compatibility: Good. A wide range of browsers are supported. • Security: Good

Cipher Suite Name	Cryptographic Algorithm Supported	Cryptographic Algorithm Not Supported	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	Cryptographic Algorithm Supported	Cryptographic Algorithm Not Supported	Description
Cipher suite 5	- AES128-SHA:AES256-SHA - AES128-SHA256:AES256-SHA256 - HIGH	- MEDIUM - LOW - aNULL - eNULL - EXPORT - DES - MD5 - PSK - RC4 - DHE	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 8-4** 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 8-4 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	Cipher suite 5	Cipher suite 6
Google Chrome 63 /macOS High Sierra 10.13.2	Not compatible	Compatible	Compatible	Compatible	Not compatible	Compatible	√
Google Chrome 49/ Windows XP SP3	Not compatible	Not compatible	Not compatible	Not compatible	Not compatible	Compatible	Compatible
Internet Explorer 6 /Windows XP	Not compatible	Not compatible	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	Not compatible	Not compatible
Safari 6/iOS 6.0.1	Compatible	Compatible	Not compatible	Compatible	Compatible	Compatible	Compatible
Safari 7/iOS 7.1	Compatible	Compatible	Not compatible	Compatible	Compatible	Compatible	Compatible
Safari 7/OS X 10.9	Compatible	Compatible	Not compatible	Compatible	Compatible	Compatible	Compatible
Safari 8/iOS 8.4	Compatible	Compatible	Not compatible	Compatible	Compatible	Compatible	Compatible
Safari 8/OS X 10.10	Compatible	Compatible	Not compatible	Compatible	Compatible	Compatible	Compatible
Internet Explorer 7/Windows Vista	Compatible	Compatible	Not compatible	Compatible	Compatible	Not compatible	√

Browser/Client	Default Cipher Suite	Cipher Suite 1	Cipher Suite 2	Cipher Suite 3	Cipher Suite 4	Cipher suite 5	Cipher suite 6
Internet Explorer 8, 9, or 10 /Windows 7	Compatible	Compatible	Not compatible	Compatible	Compatible	Not compatible	√
Internet Explorer 10 /Windows Phone 8.0	Compatible	Compatible	Not compatible	Compatible	Compatible	Not compatible	√
Java 7u25	Compatible	Compatible	Not compatible	Compatible	Compatible	Not compatible	√
OpenSSL 0.9.8y	Not compatible	Not compatible	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	Not compatible	√
Safari 6.0.4/OS X 10.8.4	Compatible	Compatible	Not compatible	Compatible	Compatible	Not 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 Choose **Security** > **Web Application Firewall** to go to the **Dashboard** page.

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 8-1 TLS configuration modification
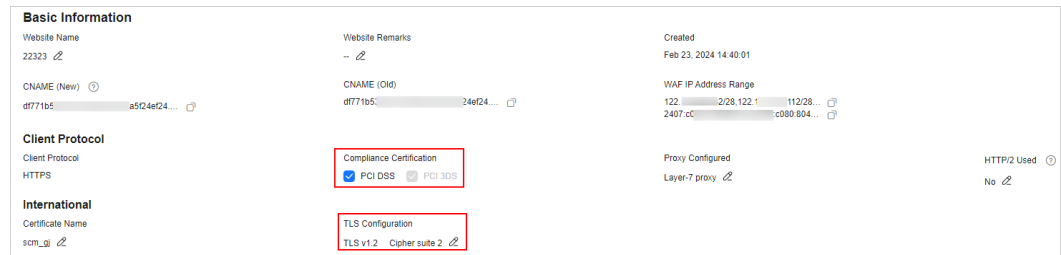
☐ **NOTE**

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 8-2 TLS Configuration
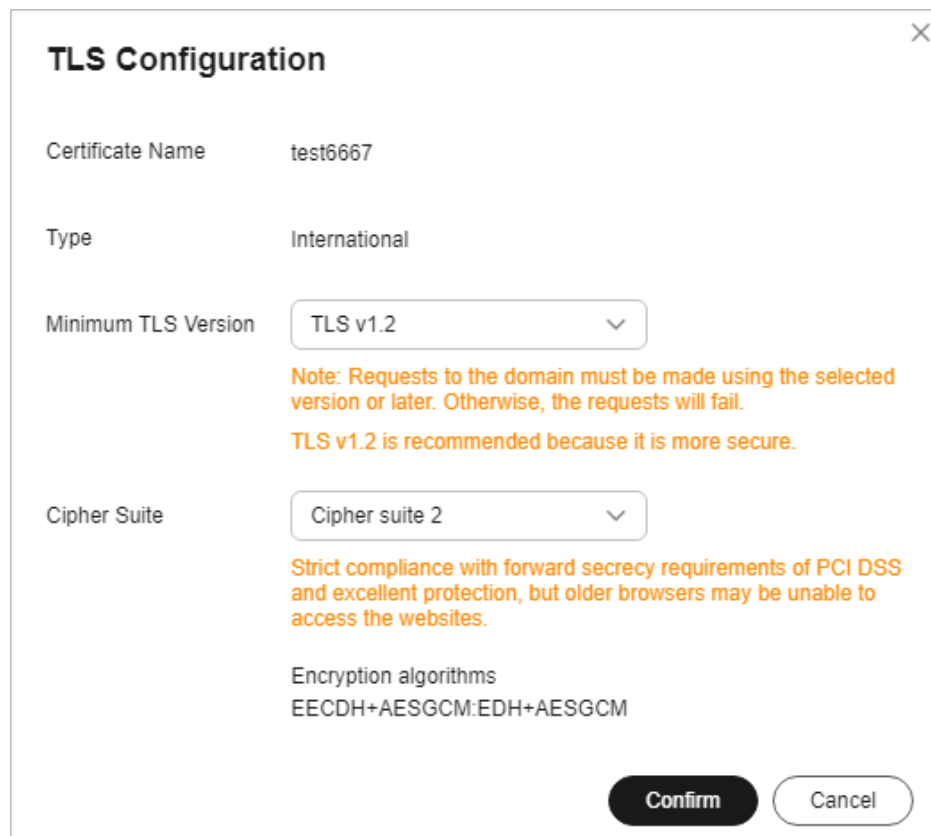
Step 8 Click **Confirm**.

----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 access 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. Choose **Security** > **Web Application Firewall** to go to the **Dashboard** page.

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 8-3 Copying the CNAME record
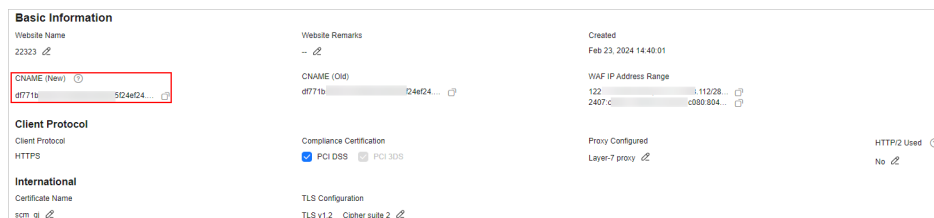
Step 2 Obtain the WAF access IP addresses.

● Cloud mode

In the command line interface (CLI) of Windows OSs, run the following command to obtain WAF access IP addresses:

ping *CNAME record*

The command output displays WAF access IP addresses. **Figure 8-4** shows an example.

Figure 8-4 ping cname
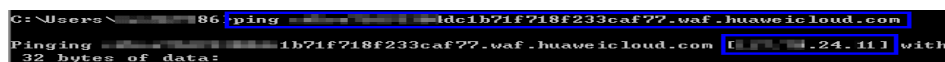
● Dedicated mode

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 access 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 access IP address* **-servername** "*Domain name of the protected website*" **-tls1_2**

If the certificate information similar to the one shown in **Figure 8-5** is displayed, the website can be accessed using TLS v1.2.

Figure 8-5 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 access IP address* **-servername** "*Protected domain name*" **-tls1_1**

If no certificate information is displayed, as shown in **Figure 8-6**, WAF has blocked the access that used TLS v1.1.

Figure 8-6 Verifying TLS v1.1

----End

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

Application Scenarios

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 8-7** is displayed, the connection is established and the origin server IP address is exposed.

Figure 8-7 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.

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 Choose **Security** > **Web Application Firewall** to go to the **Dashboard** page.

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.

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.

Figure 8-8 WAF Back-to-Source IP Addresses dialog box
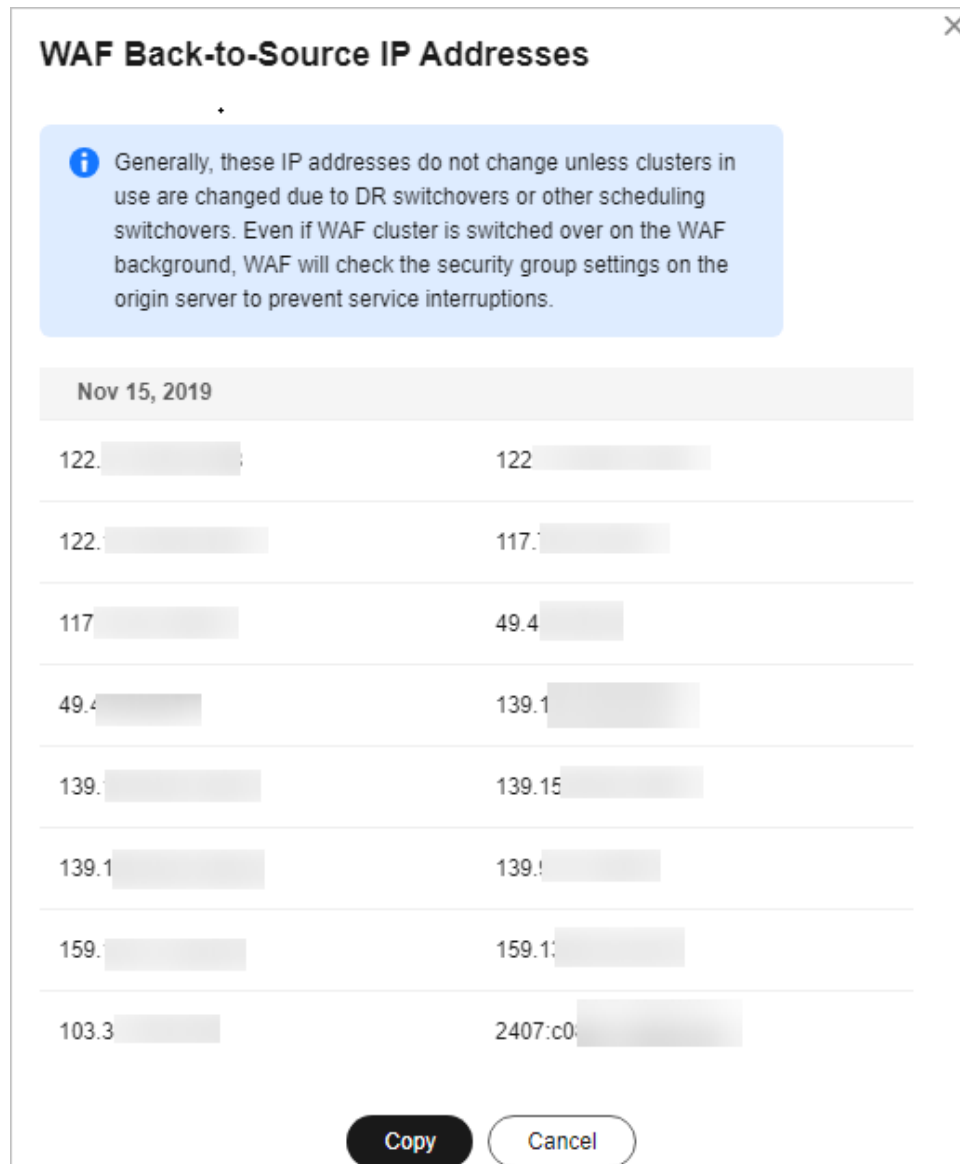
----**End**

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 [icon] in the upper left corner of the management console and select a region or project.

Step 3 Click [icon] 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 8-5**.

Table 8-5 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 8-9 Listener list

Name/ID	Monitoring	Frontend Protocol/Port	Health Check ⑦	Default Backend Server Group ⑦	Access Control ⑦	Operation
listener-7ffc f723dfda-b6f2-40be-b512-77084b127e29	⊡	HTTP/80	● Healthy	server_group-8081 View/Add Backend Server	All IP addresses Configure	Add/Edit Forwarding Policy | Edit | Delete

Step 6 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

9 Obtaining the Real Client IP Addresses

Application Scenarios

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 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.

Architecture

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
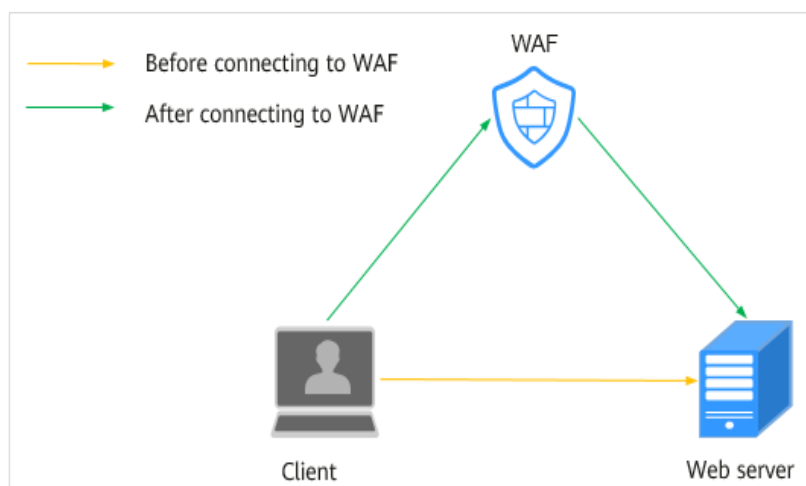
📖 **NOTE**

- 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.

- If you select **Yes** for **Use Layer-7 Proxy** when you add a domain name to WAF, WAF obtains the source IP address in the following sequence:

  a. The source IP header list configured in **upstream** is preferentially used, that is, the IP address tag configured on the basic information page of the domain name. For details, see **Configuring a Traffic Identifier for a Known Attack Source**. If no IP address is available, go to **b**.

     📖 **NOTE**

     If you want to use a TCP connection IP address as the client IP address, set **IP Tag** to **remote_addr**.

  b. Obtain the value of the **cdn-src-ip** field in the source IP header list configured in the **config** file. If no value is obtained, go to **c**.

  c. Obtain the value of the **x-real-ip** field. If no value is obtained, go to **d**.

d. Obtain the first public IP address from the left of the **x-forwarded-for** field. If no public IP address is obtained, go to **e**.

e. Obtain the value of the **remote_addr** field, which includes the IP address used for establishing the TCP connection.

- If no proxy is used, WAF obtains the source IP address from the **remote_ip** field.

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

How Does WAF Obtain the Real Client IP Addresses for Services Deployed in CCE?

If your service is deployed on Cloud Container Engine (CCE), CCE records the real client IP address in the **X-Original-Forwarded-For** field and records the WAF back-to-source address in the **X-Forwarded-For** field. You need to modify the CCE configuration file to enable Ingress to add the real IP address to the **X-Forwarded-For** field. In this way, WAF can obtain the real client IP address.

To modify the CCE configuration file, take the following steps:

Step 1 Run the following command to modify the **kube-system/nginx-configuration** configuration file:

```
kubectl -n kube-system edit cm nginx-configuration
```

Step 2 Add the following information to the configuration file:

```
compute-full-forwarded-for: "true"
forwarded-for-header: "X-Forwarded-For"
use-forwarded-headers: "true"
```

Step 3 Save the configuration file.

The configuration takes effect after you save the file. Ingress adds the real client IP addresses to the **X-Forwarded-For** field.

Step 4 Modify the field for the service to obtain the real client IP addresses to **X-Original-Forwarded-For**.

----End