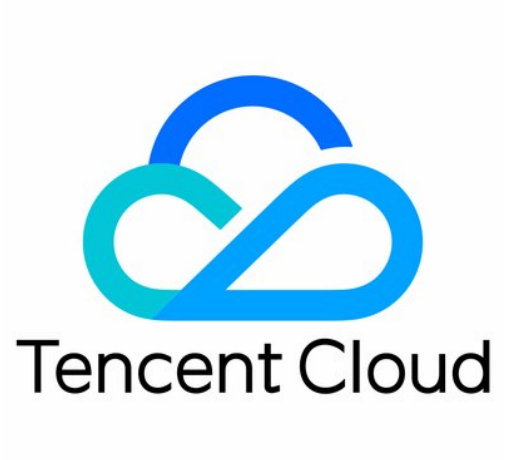
Web Application Firewall

Best Practice

Product Documentation



Contents

Best Practice
WAF CCP Overview

Last updated：2023-12-29 14:52:34

WAF meets the major standards of CCP 2.0. According to Information security technology – Baseline for classified protection of cybersecurity (GB/T 22239-2019), WAF meets the security requirements at level 3.

No.	CCP Chapter	CCP No.	CCP Standard Content	Feature Description
1	Access control	8.1.3.2 e)	Access control based on application protocol and content should be implemented for inbound/outbound data flows.	Access control policies at the application layer are configured to implement access control based on application protocol and content for inbound/outbound data flows.
2	Intrusion protection	8.1.3.3 a)	Externally initiated network attacks should be detected, prevented, or blocked on key network nodes.	WAF is deployed on the perimeters to detect and trigger alarms for various attacks and scans.
3	Intrusion protection	8.1.3.3 c)	Technical measures should be adopted to analyze network behaviors, especially new types of network attack behaviors.	WAF can check and block web traffic in real time and supports AI + rule dual-engine protection to prevent zero-day and other new unknown attacks.
4	Intrusion protection	8.1.3.3 d)	When an attack behavior is detected, the attack source IP, type, target, and event should be logged, and alarms should be triggered for serious intrusions.	WAF can detect and block HTTP and HTTPS traffic attacks and log information such as attack type, URL, content, and source IP, hit rule name and ID, risk level, attack time, target host, and executed action.
5	Malicious code protection	8.1.3.4 a)	Malicious code should be detected and cleared on key network nodes, and the malicious code protection mechanism should be upgraded and updated promptly.	WAF basic security and rule engine modules can implement this feature.
6	Security audit	8.1.3.5	Security audit should be performed	Intrusion events are audited on

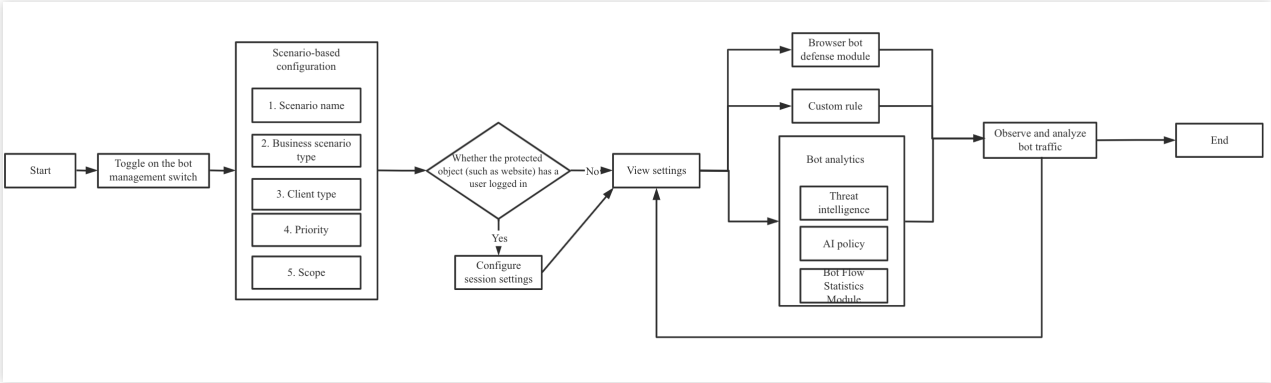
		a)	on the network perimeters and key network nodes and cover every user to audit key user behaviors and security events.	the perimeters.
7	Security audit	8.1.3.5 c)	Audit logs should be protected and regularly backed up to prevent unexpected log deletion, modification, and overwriting.	Logs are retained for at least six months, during which tenants cannot delete or tamper with them.

Bot Management
Best Practices of Scenario-Based Bot Configuration

Last updated：2023-12-29 14:52:50

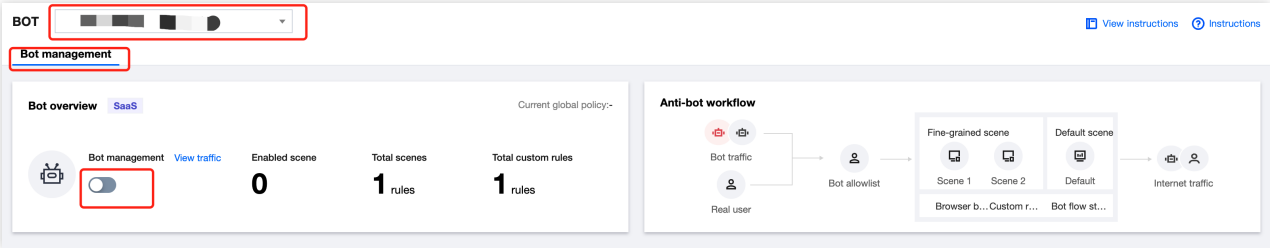
Overview

With bot and application security, you can enable and configure modules in bot management, observe and analyze traffic through bot traffic analysis and access logs. Then, you can set refined policies based on the session status to protect core website APIs and businesses from bot attacks.
Bot management supports configuration of bot scenario types, client risk identification (browser bot defense module), threat intelligence module, AI evaluation module, bot flow statistics module, action score, custom rules, token configuration, and legitimate bots. You can configure these modules for refined bot management as shown below:



Prerequisites

To connect to bot traffic management, you need to purchase a WAF instance extra pack.
On the **Bot and application security** page, you have selected the target domain name and enabled bot traffic management.



Scenario-Based Bot Configuration

Leveraging Tencent's years of expertise in bot governance, this feature offers client risk identification (browser bot defense module), threat intelligence module, AI policy module, bot analytics module, action score, session management, legitimate bots, and custom rules specifically for flash sales, price/content crawling, and login scenarios. It simplifies configuration and makes everything easy to use.

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.
3. On the **Bot management** tab, click **Create scenario**.
4. In the pop-up window, configure parameters and click **Create now**.

Note:

The flash sales, login, or price/content crawling scenario and custom scenario are mutually exclusive.

Parameter description:

Scenario name: Scenario name, which can contain up to 50 characters.

Business scenario type: You can select multiple ones, including flash sales, login, price/content crawling, and custom scenarios.

Client type: Type of the client accessing the protected object.

Priority: Scenario execution priority, which is an integer between 1–100. The smaller the value, the higher the priority.

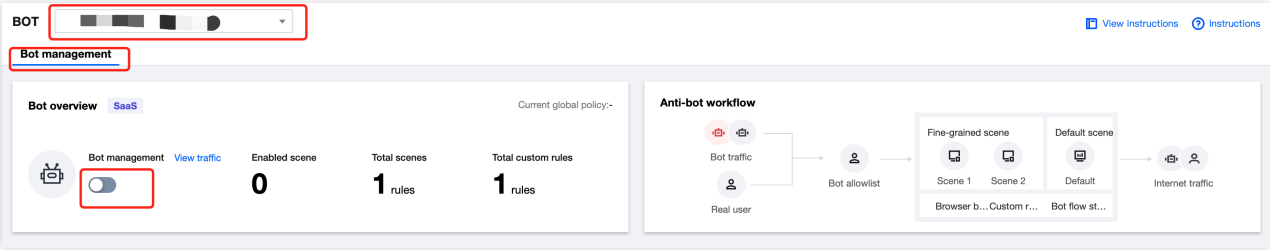
Scope: The scenario scope under the domain name, which can be **All scopes** or **Custom scope**.

5. The scenario-based management list will display the data of the created scenario card, which can be further configured.

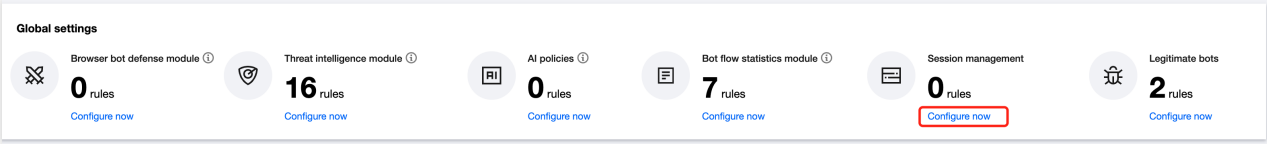
Session Management

This feature allows you to configure the token location of a session to differentiate between access behaviors of different users through the same IP. Therefore, you can precisely handle a user with abnormal access behavior without affecting other users.

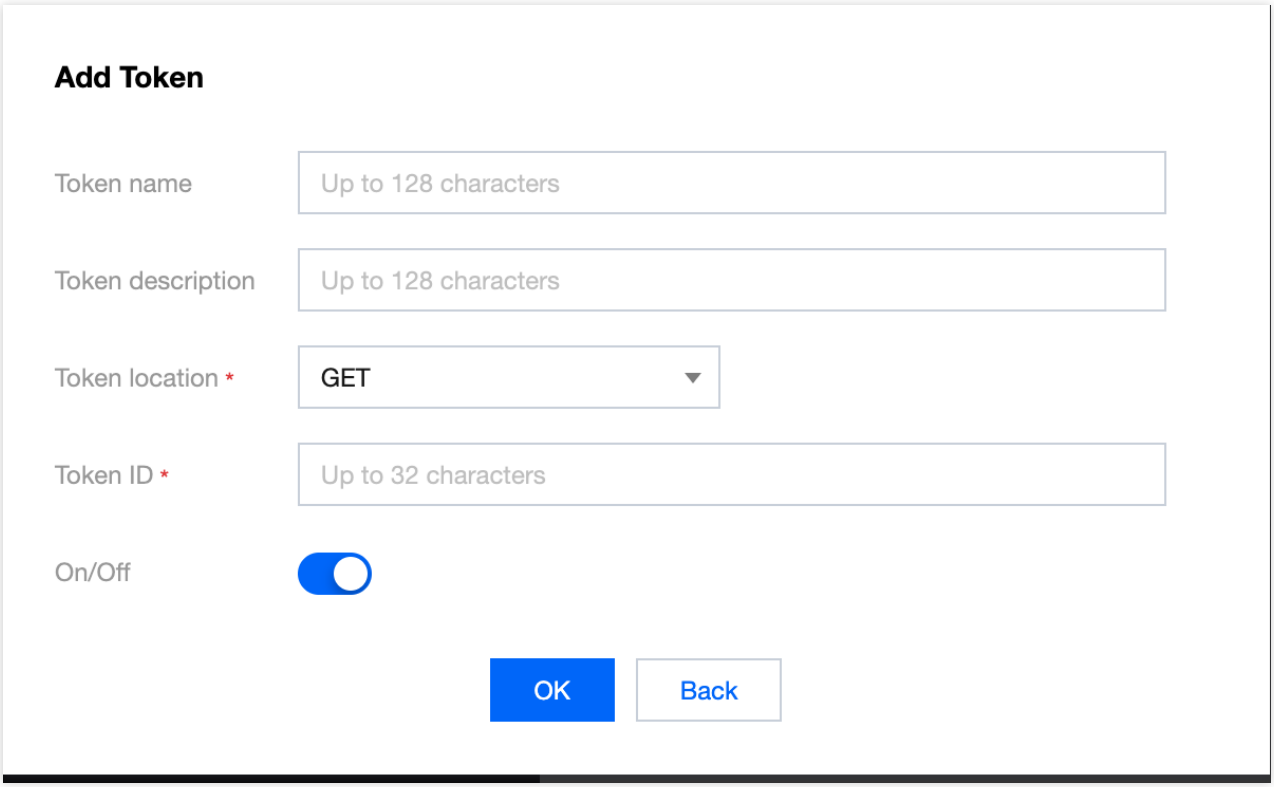
1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **Session management module** section.



4. On the **Session management** page, click **Add a configuration**, configure parameters, and click **OK**.



Parameter description:

Token name: Custom name, which can contain up to 128 characters.

Token description: Custom description, which can contain up to 128 characters.

Token location: It can be **HEADER**, **COOKIE**, **GET**, or **POST**. Here, **GET** and **POST** are HTTP request content parameters rather than HTTP header information.

Token ID: Token ID.

Client Risk Identification (Browser Bot Defense Module)

The client risk identification feature uses the dynamic identity verification technology and generates a unique ID for each client's business request to detect possible bots and malicious crawlers in the access to websites or HTML5 pages.

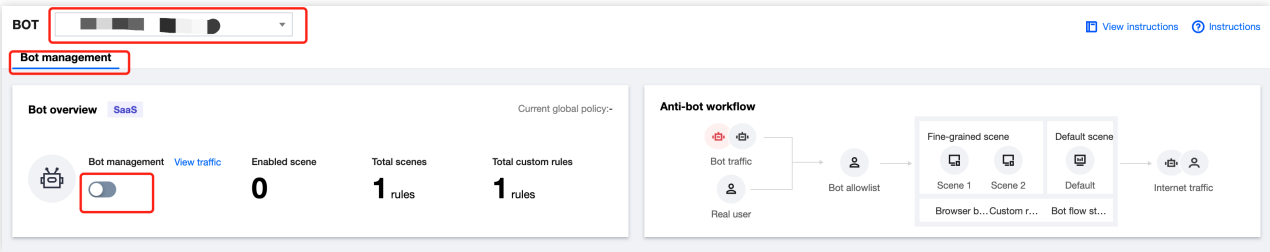
Note:

This feature **does not support CLB-WAF, wildcard domain names, and applications**. It applies only to websites and HTML5 pages. If non-dynamic verification is involved, the automated API script needs to be first added to the allowlist.

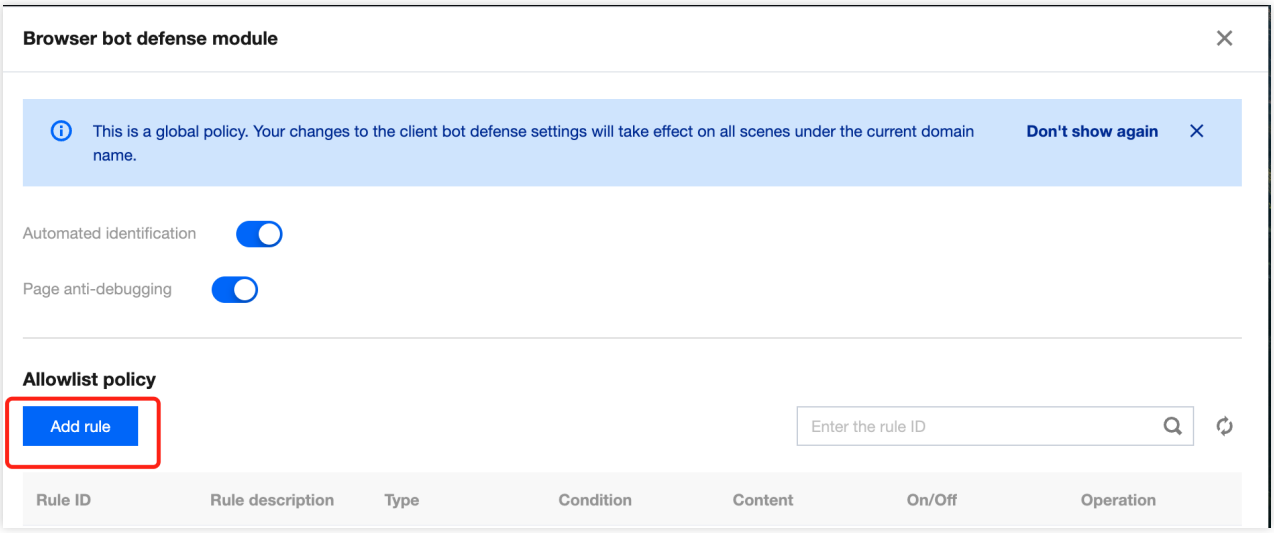
Adding to allowlist

The allowlist is mainly used to allow APIs that don't need to be set.

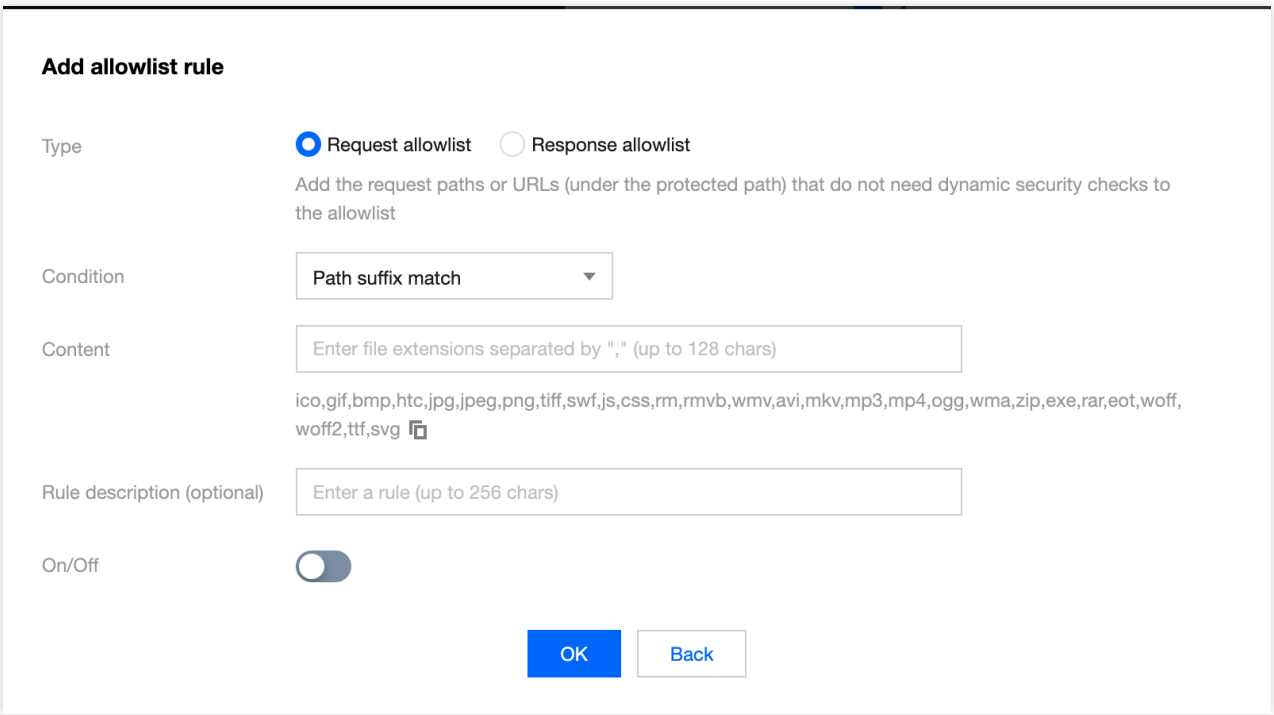
1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **Browser bot defense module** section.
4. On the **Browser bot defense module** page, click **Add rule**.



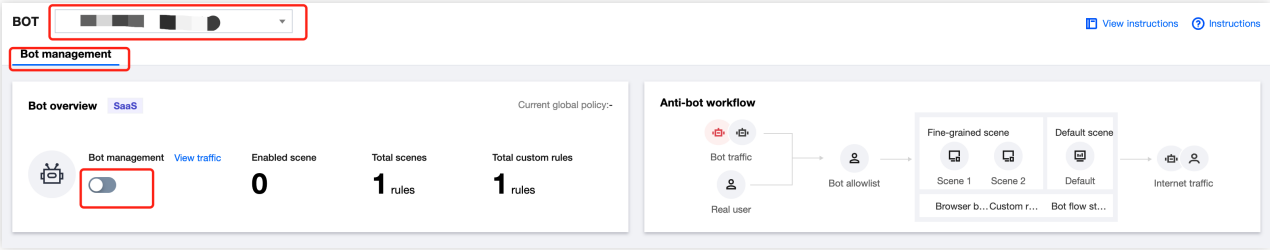
5. In the **Add allowlist rule** pop-up window, configure parameters and click **OK**.



Case 1: A large number of requests from automated scripts

There are a large number of requests from automated scripts. In this case, you can block `CURL`, `SOAPUI`, `JMETER`, `POSTMAN`, and similar requests.

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.

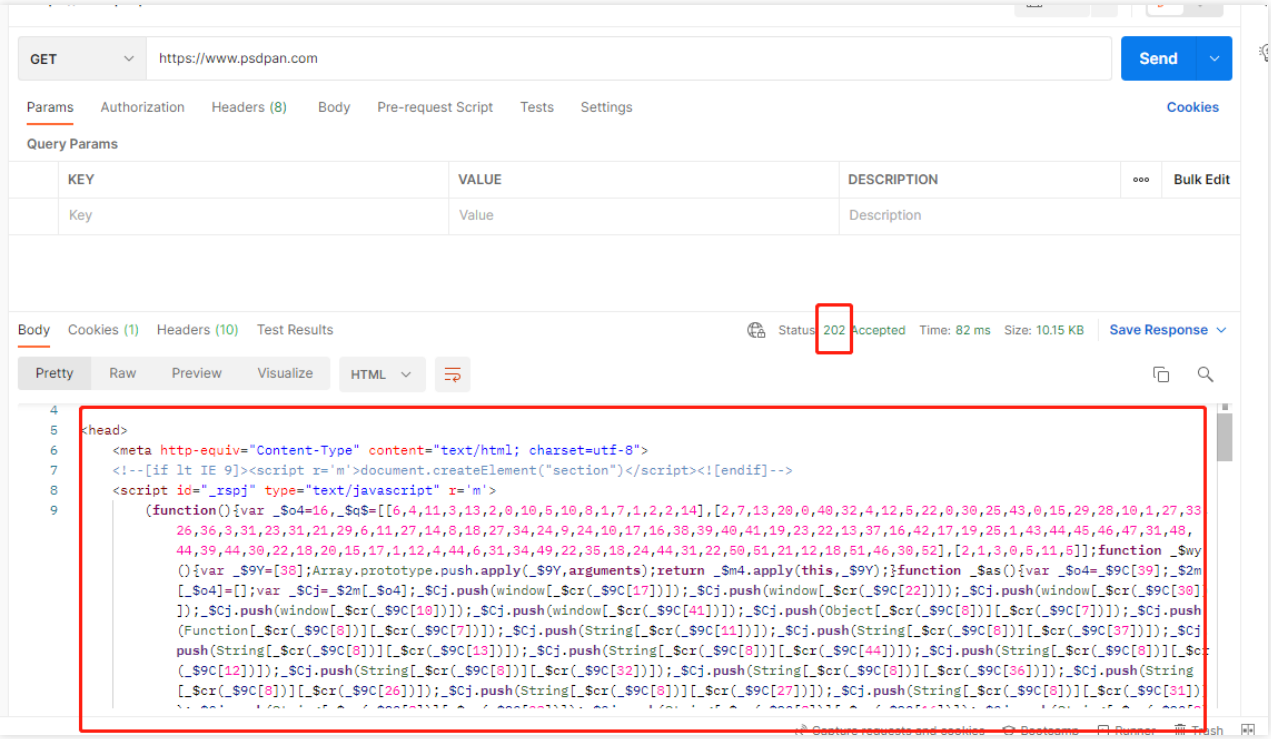


3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **Browser bot defense module** section.
4. Click

of **Automated identification** to confirm the allowlist.
5. On the configuration page of a certain scenario, click **Browser bot defense module**, click

, and select **Block** for **Defense mode**.
6. Below are the results of the `CURL` , `SELENIUM` , and `POSTMAN` requests:

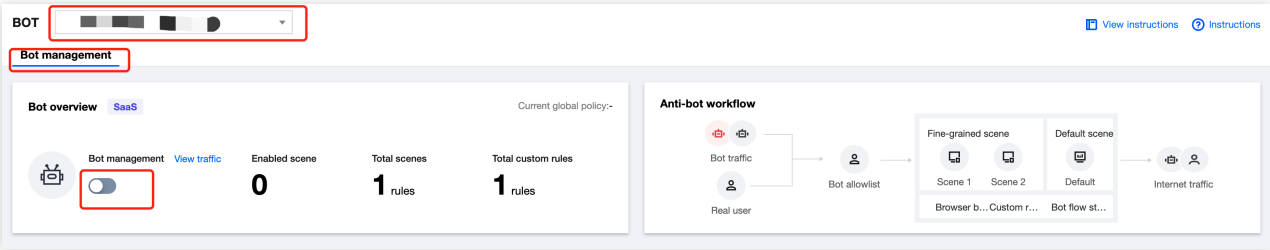
```
$1f.length;_$uD++){_$1f[_$uD]^=_$2_[Math.abs(_$uD)%16];}}return;}}}else if(_$WA*122>1830&&32-_$WA>0){if(-100<_$WA-123&&_$
WA*122<3416){if(150===126+_$WA){_$hL+=7;}else if(92*_$WA===2300){_$hL+=-13;}else if(-24===_$WA-50){_$uD.push("MI7Fp9DreuO
OxmJnUWnNuL");}else{var _$wr=_$mt[19];}}else if(_$WA*79>1501&&24-_$WA>0){if(131===111+_$WA){_$uD.push(59);}else if(2*_$WA
===42){_$12=_$2_[_$8B(_$mt[5])];}else if(16===_$WA-6){_$12=_$vc&&_$vc[_$8B(_$mt[3])];}else{_$12=_$lt[_$8B(_$mt[6])](_$8B(
_$mt[0]));}}else if(15-_$WA<0&&20>_$WA){if(43===27+_$WA){_$vc[_$8B(_$mt[3])][_$8B(_$mt[29])](_$vc);}else if(91*_$WA===154
7){return;}else if(-63===_$WA-81){_$uD.push(4);}else{_$y2();}}else{if(91===63+_$WA){_$hL+=3;}else if(58*_$WA===1682){_$12
=_$lt[_$8B(_$mt[4])];}else if(-73===_$WA-103){_$hL+=13;}else{_$lt[_$8B(_$mt[6])](_$8B(_$mt[0]))[0][_$8B(_$mt[2])](_$2_);}
}}else if(16>_$WA){if(-55<_$WA-62&&_$WA*124<1488){if(41===33+_$WA){_$uD.push("{qqqr0_RGVjOpxCUk]_YuJlqTLN1fSxoDg7spVZxTZL
8nQ_UPWqqr7Dccebc0qqr4r0qqr0c22qq.CB.7K7RYGXUTYmj9XtdgeOmTyesaZg_0qSWhVgawzuE|)YCaXKuVdJsy7mbq1rpWv1s3eM9Q8EDa6WpVpsslQMr
2bWDEEql0G1qflJhlQzcHTRdpfCCuWWBV44bufWBL4gCuHMzar5pkcMJLaN16Qh_90_Ok8MJZM4SuJiZZaak6jqdqK4ShNMzEy6buhxWqeSkhViW0Suu4pWNa
VnCKpRxxfvD4jMWzB2CIJ3wRPPDFmMHL1auBDHz9GOk1KlQlwarciRdNy6OsMJwSvZob.xH9u4pDZoMZAZC8r8F70.kKNRdzyNPsCHR29eOtjJ33oLDuvxRgp
7DDZDRzP_PIPqtEcBDtbEy9_zaqqqqqqr0QQSplxlW7APrrh9L71n2ct0EXAP2hqqVHiGJ6GOIcJ0J6GuVkwZi{Mq32FZPH784zx4TOjpdyq5PdDM_J0nToIp
7rGzuBJM5xzec66wHRzzCdNMIlzZosuM3SBTD6ihQT0nD6gIENB0b6lUtWqqh7QQHsrGZiGac64qqr0HQNywdloZpR9Ua20qch7eki6z9Dm5AqqqYW9hjv3RC
iGJkEOxVPeGt4c64qql4096qqqhQAM3Ma8MO_wkRbQqqk162HmCGbKcppEmgBVn3qqt1083179040lrrL.");}else if(67*_$WA===603){_$vc._$Ui=_$
iP;}else if(-21===_$WA-31){_$uD.push("7V00tRWGFA");}else{if( !_$12)_$hL+=1;}}else if(_$WA*116>348&&8-_$WA>0){if(62===58+
_$WA){_$2_.id=_$1f;}else if(118*_$WA===590){_$2_[_$8B(_$mt[35])]=_$eV;}else if(-117===_$WA-123){var _$2_=_$lt[_$8B(_$mt[2
3])](_$8B(_$mt[24]));}else{_$uD.push(4);}}else if(4>_$WA){if(101===101+_$WA){_$lt[_$8B(_$mt[4])][_$8B(_$mt[2])](_$2_);}el
se if(52*_$WA===52){if( !_$12)_$hL+=2;}else if(-69===_$WA-71){_$uD.push("Vk_yxby7sIG");}else{var _$1f=_$m0;}}else{if(134=
==122+_$WA){_$2_[_$8B(_$mt[34])]=_$mt[33];}else if(74*_$WA===962){_$2_.src=_$13;}else if(-4===_$WA-18){_$lt[_$8B(_$mt[0])
][_$8B(_$mt[2])](_$2_);}else{_$uD.push("R.ldTebdfga");}}}else{if(-17<_$WA-64&&_$WA*22<1144){if(85===37+_$WA){return 0;}el
se if(51*_$WA===2499){return Math.abs(arguments[1]) % 16;}else if(36===_$WA-14){return 10;}else{return 8;}}else{return 1;
}}}function _$rY(_$vs){var _$wr,_$uD,_$KD=_$vs,_$vc=_$Fb[2];while(1){_$uD=_$vc[_$KD++];if(-16>_$uD-20){if(3===_$uD){_$wr=
_$2_[_$8B(_$mt[5])]==_$8B(_$mt[15])||_$2_[_$8B(_$mt[5])]==_$8B(_$mt[42]);}else if(120===119+_$uD){_$PU(_$1f);}else if(70*
_$uD===140){_$2_[_$8B(_$mt[46])]=null;}else{if( !_$wr)_$KD+=2;}}else{return;}}}}})()</script></head>
<body>
<input type="hidden" id="__onload__" name="cDLJ.6zflivja8RAGWSNtmGchMfTmH_nrcvrZ2rWMSsSfm3KWkWRvkmWb1UdoYcTl8J_iPk.XCM_z7
XBKK8HwG" value="g.bsDjQpVCmzPMoeR.dbDA">

</body>
</html>
psdpan@psdpandeMacBook-Pro ~ % curl http://www.psdpan.com -I
HTTP/1.1 202 Accepted
Content-Type: text/html; charset=utf-8
Connection: keep-alive
Set-Cookie: Cc2838679FS=5ffyjNUVxUtd.BOCnqlHHKmk7AhiBH.OtxKdMrzQg1gG.T8yHY8c.A2gLxFTip_ohj9ld.vaZwWDWfo_OuKvQ4G; Path=/;
expires=Tue, 02 Mar 2032 09:11:53 GMT; HttpOnly
Expires: Sat, 05 Mar 2022 09:11:53 GMT
Date: Sat, 05 Mar 2022 09:11:53 GMT
Server: ******
Cache-Control: no-store
Pragma: no-cache
```



Case 2: Prohibiting webpage debugging

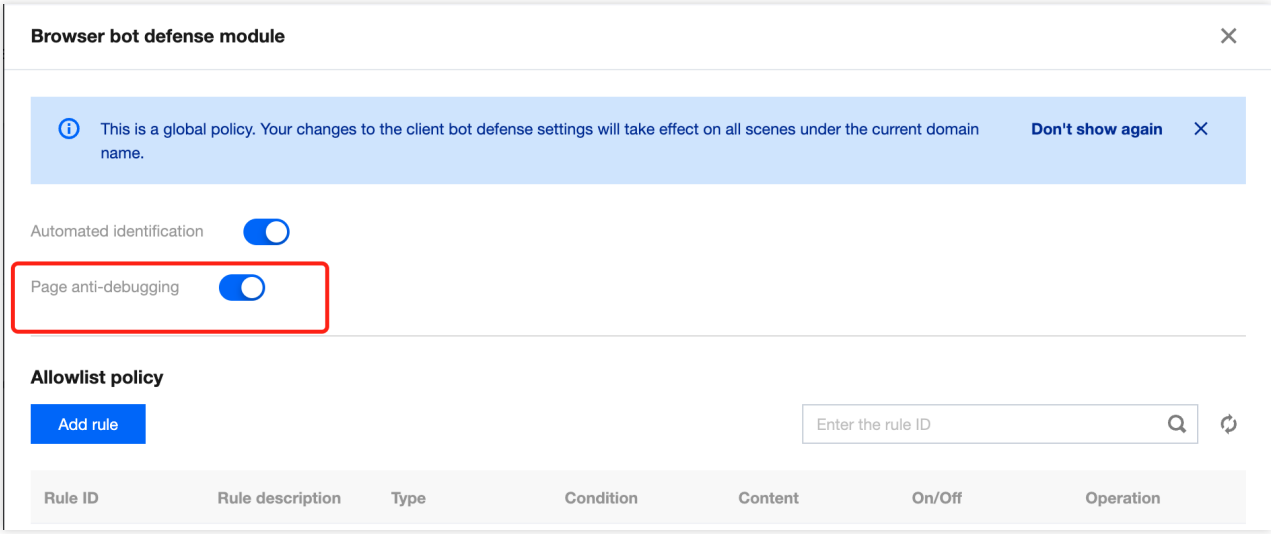
Prohibit webpage debugging to avoid targeted crawler writing.

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **Browser bot defense module** section.
4. Click

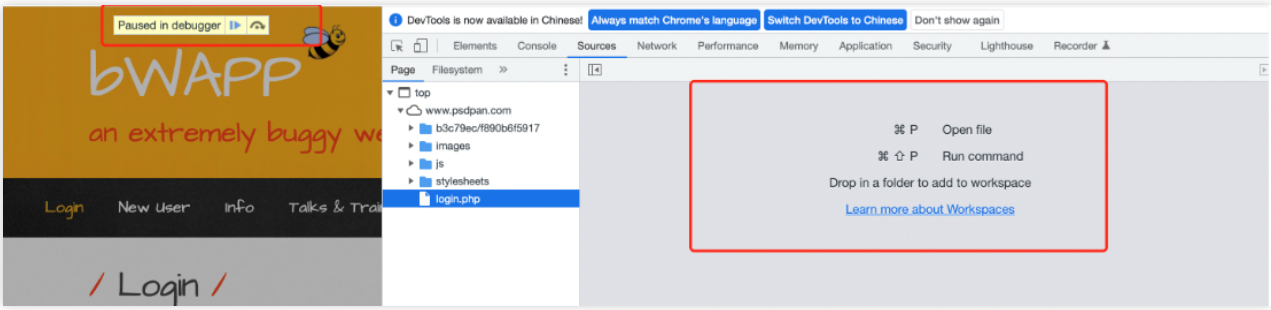
of **Page anti-debugging** to confirm the allowlist.



5. On the configuration page of a certain scenario, click **Browser bot defense module**, click

, and select **Block** for **Defense mode**.

6. Below is the result of the Chrome request:



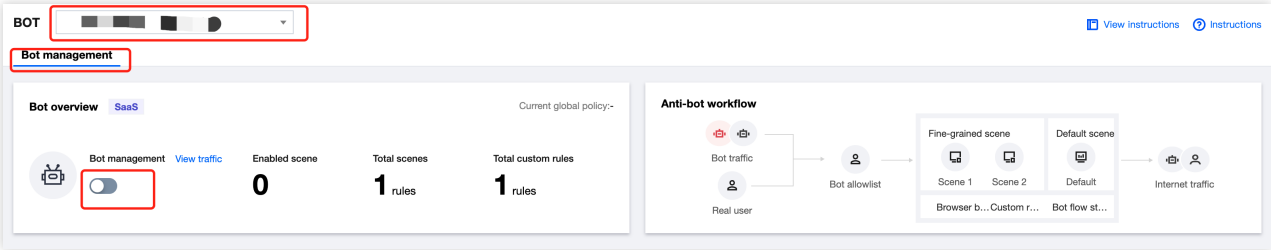
Threat Intelligence Module

The threat intelligence module feature is built on Tencent's nearly 20 years of experience in cybersecurity and big data intelligence. It determines the status of an IP in real time and uses a scoring mechanism to quantify a risk. It precisely identifies the access from a malicious dynamic IP and IDC. In addition, it intelligently identifies the features of a malicious crawler to cope with risky access requests from malicious crawlers, distributed crawlers, proxies, credential stuffing, and bargain hunting.

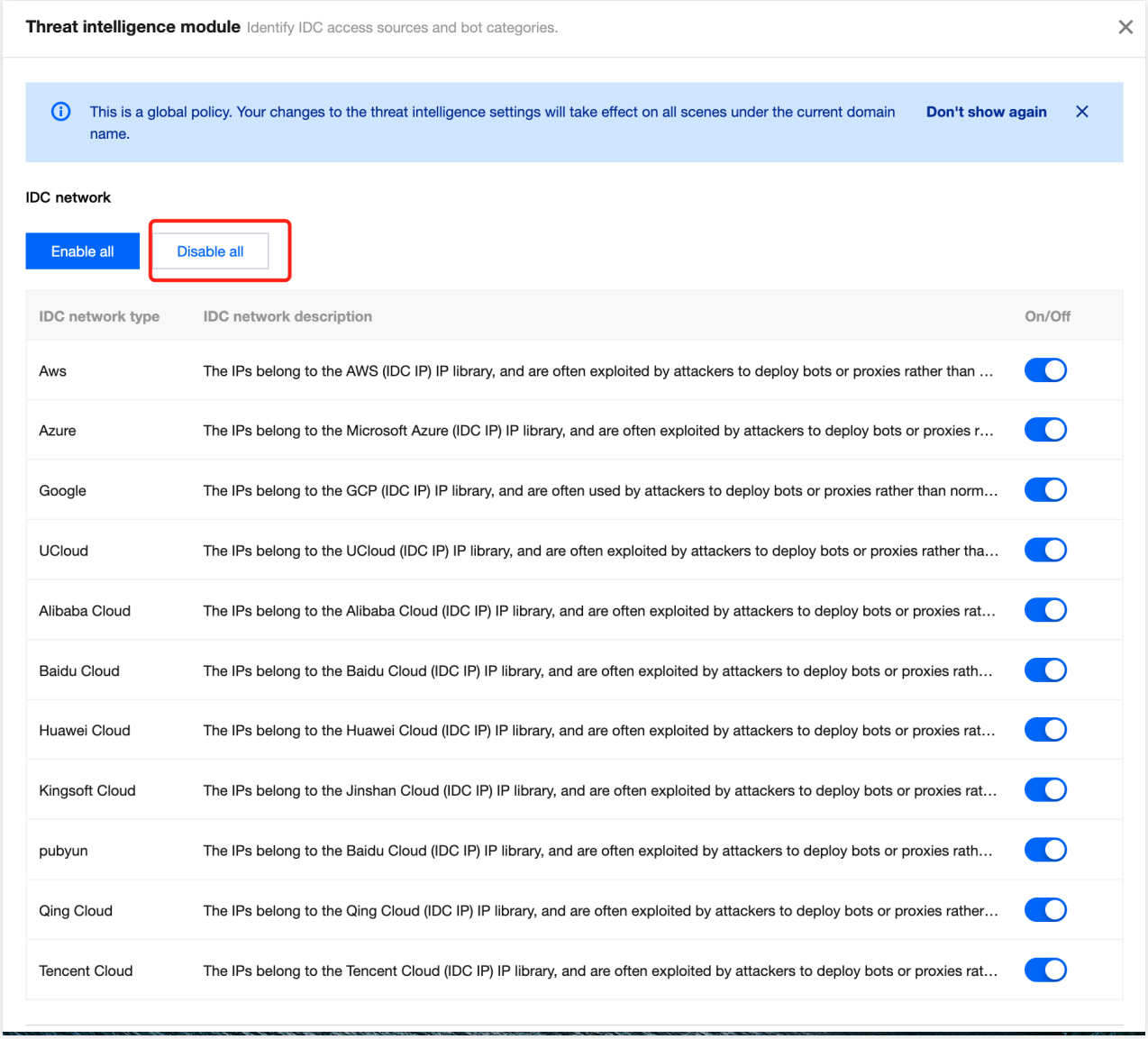
Note:

Before enabling the threat intelligence module feature, you need to check whether the business has IDC traffic access, and if so, disable IDC before enabling threat intelligence module.

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.

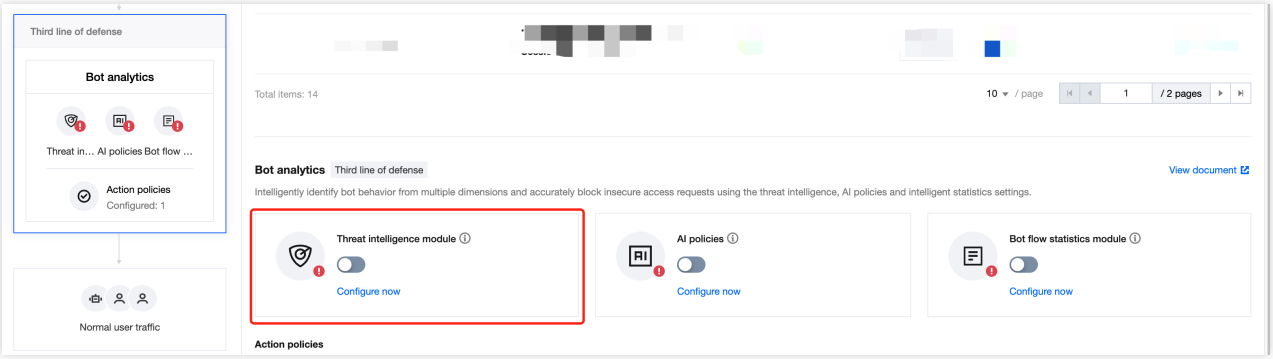


3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **Threat intelligence module** section.
4. On the **Threat intelligence module** page, check whether there is IDC traffic access, and if so, click **Disable all** of **IDC network**.



5. If there is no IDC traffic access, click the configuration page of a certain scenario, click **Bot flow statistics module**, and click

in the **Threat intelligence module** section.



AI Evaluation Module

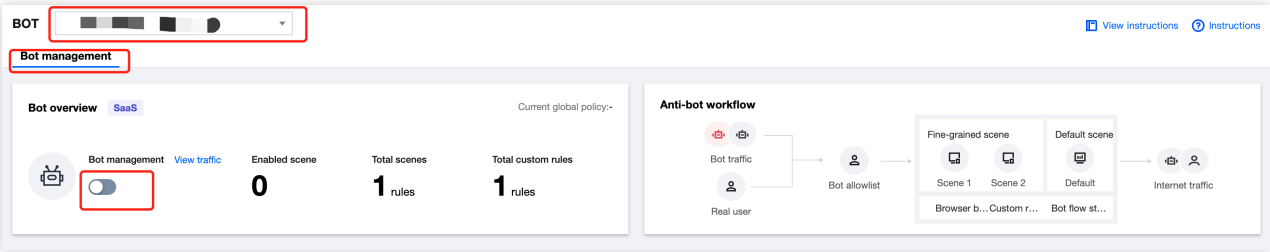
The AI evaluation module feature builds AI evaluation models from AI technologies and Tencent's experiences in controlling risks and fighting cybercrimes. Through big data analysis and AI modeling of access traffic, it quickly identifies malicious requesters and defends against risky access requests from APT and hidden threat bots.

Note:

The AI evaluation module implements automatic learning based on AI modeling and can be directly enabled. If there is a false positive, add the URL to the allowlist.

Enabling the AI evaluation module

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.

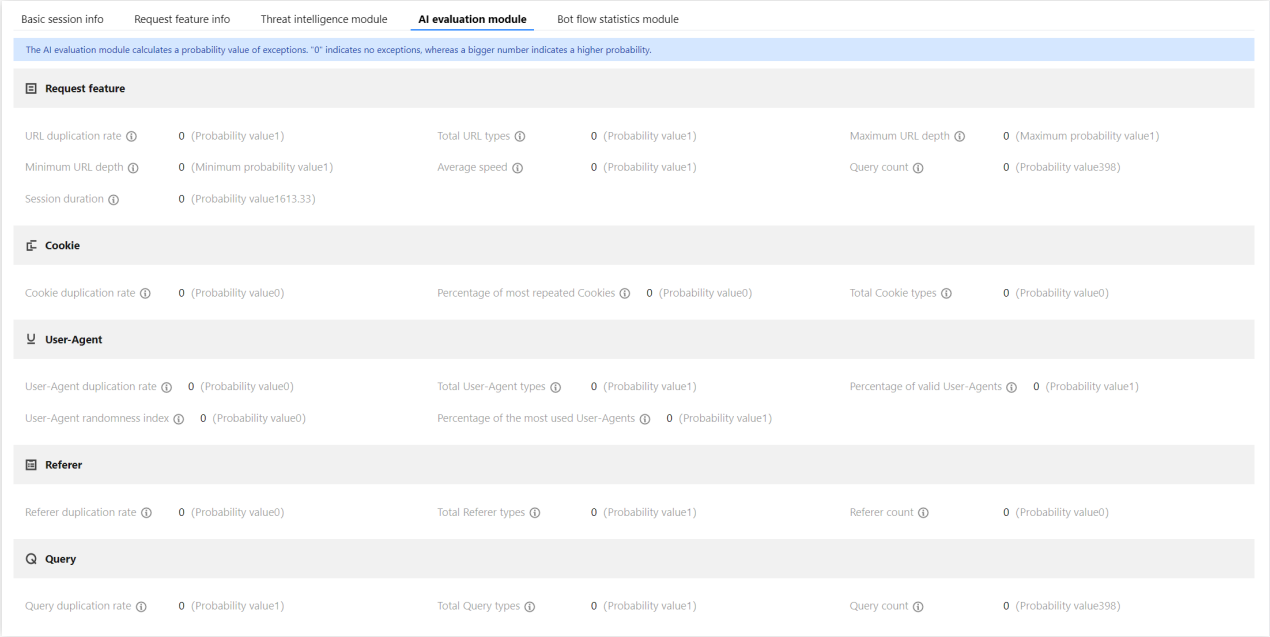


3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **AI policy module** section.

Adding to allowlist

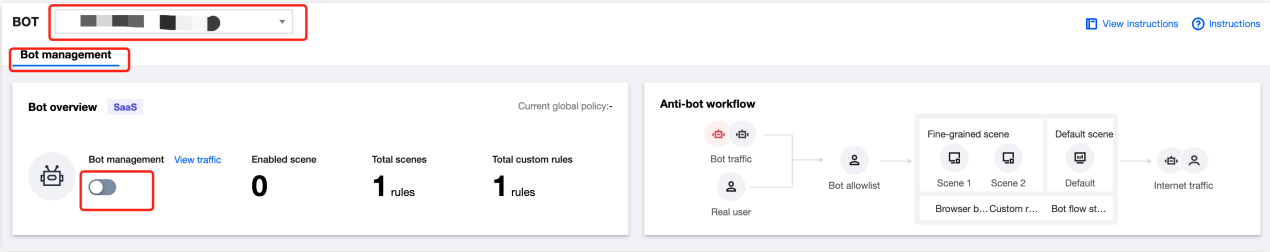
Background

On the **AI evaluation module** tab, the request is normal but reported as abnormal.

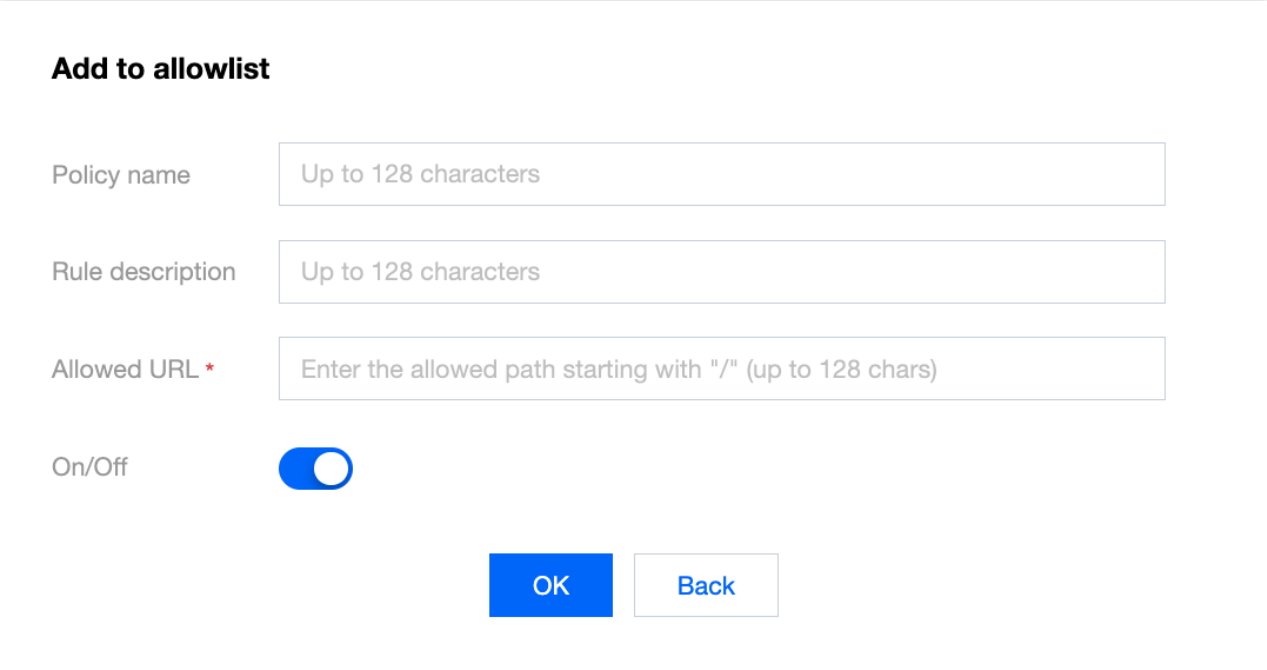


Directions

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



3. In **Global settings** on the **Bot management**tab, click **Configure now** in the **AI evaluation module** section.
4. On the **AI evaluation module** page, click **Add to allowlist**, enter the name, description, and URL, and click **OK**.



5. Click the configuration page of a certain scenario, click **Bot flow statistics module**, and click

in the **AI policy module** section.

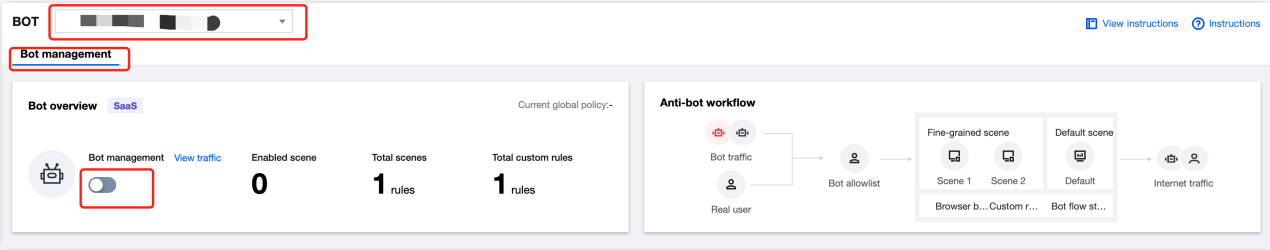
Bot Flow Statistics Module

Based on big data analysis, the bot flow statistics module feature automatically classifies customer traffic by feature and identifies abnormal and malicious traffic. It automatically adjusts the malicious traffic threshold and handles risky access requests from general and high-frequency bots. With auto-adjustment modeling, it resolves most of the bot behavior bypasses.

Note:

You can directly enable the bot flow statistics module. The smart mode is recommended.

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **AI evaluation module** section.

Action Policy

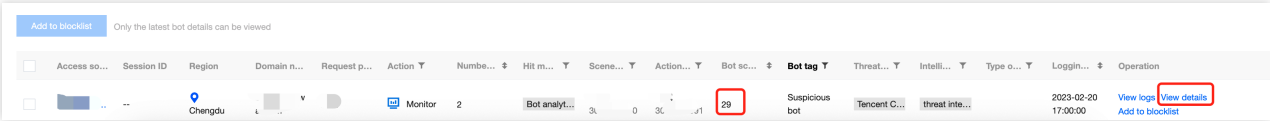
The action score feature leverages the threat intelligence module, AI evaluation module, and bot flow statistics module to provide a comprehensive score ranging from 0 to 100 for the risk level of an access request to a website. The higher the score, the more likely it is from a bot, and the higher the risk level. With the score provided by bot analytics, the risk level of an access request is intelligently identified, and you can precisely block a risky access request by configuring different action policies, the scope of each action policy, and actions in different score ranges.

Background

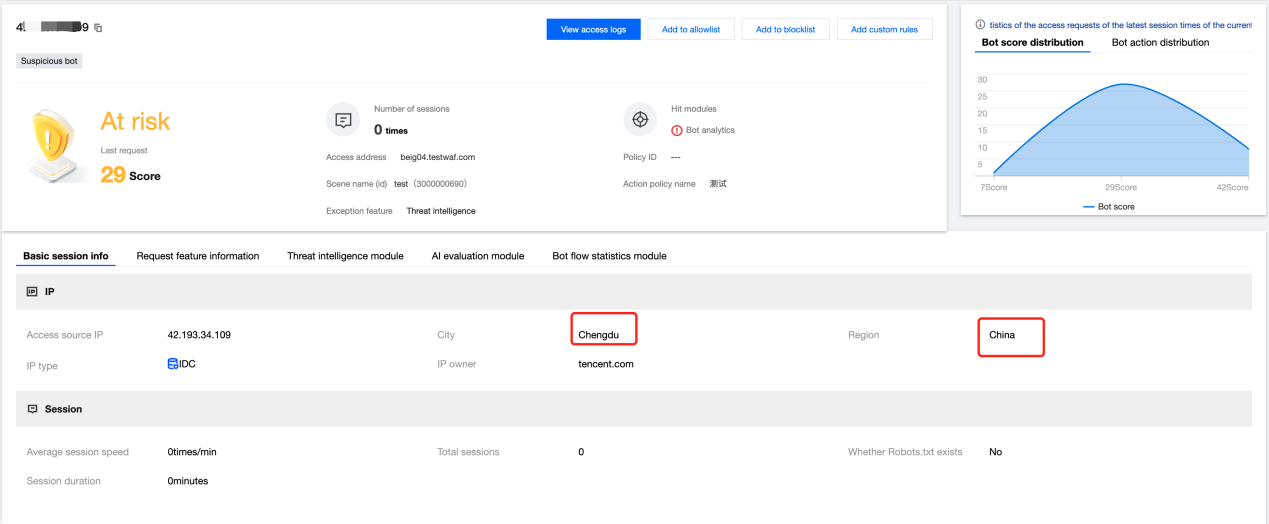
When the threat intelligence module, AI evaluation module, or bot flow statistics module identifies a large amount of traffic, you can customize actions for configuration analysis, as the default configuration cannot implement precise blocking.

Directions

1. Log in to the WAF console and select **Bot traffic analysis** on the left sidebar.
2. On the **Bot traffic analysis** page, select the target domain name in the top-left corner, select the target access source, and click **View details**.

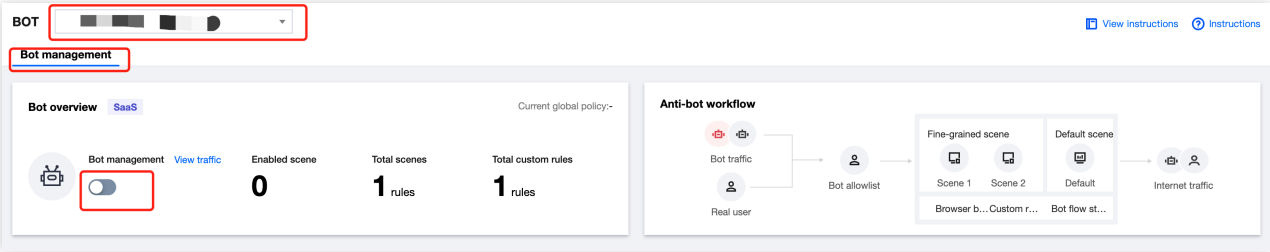


3. In the **Basic session info** section on the details page, view the region and IP region.



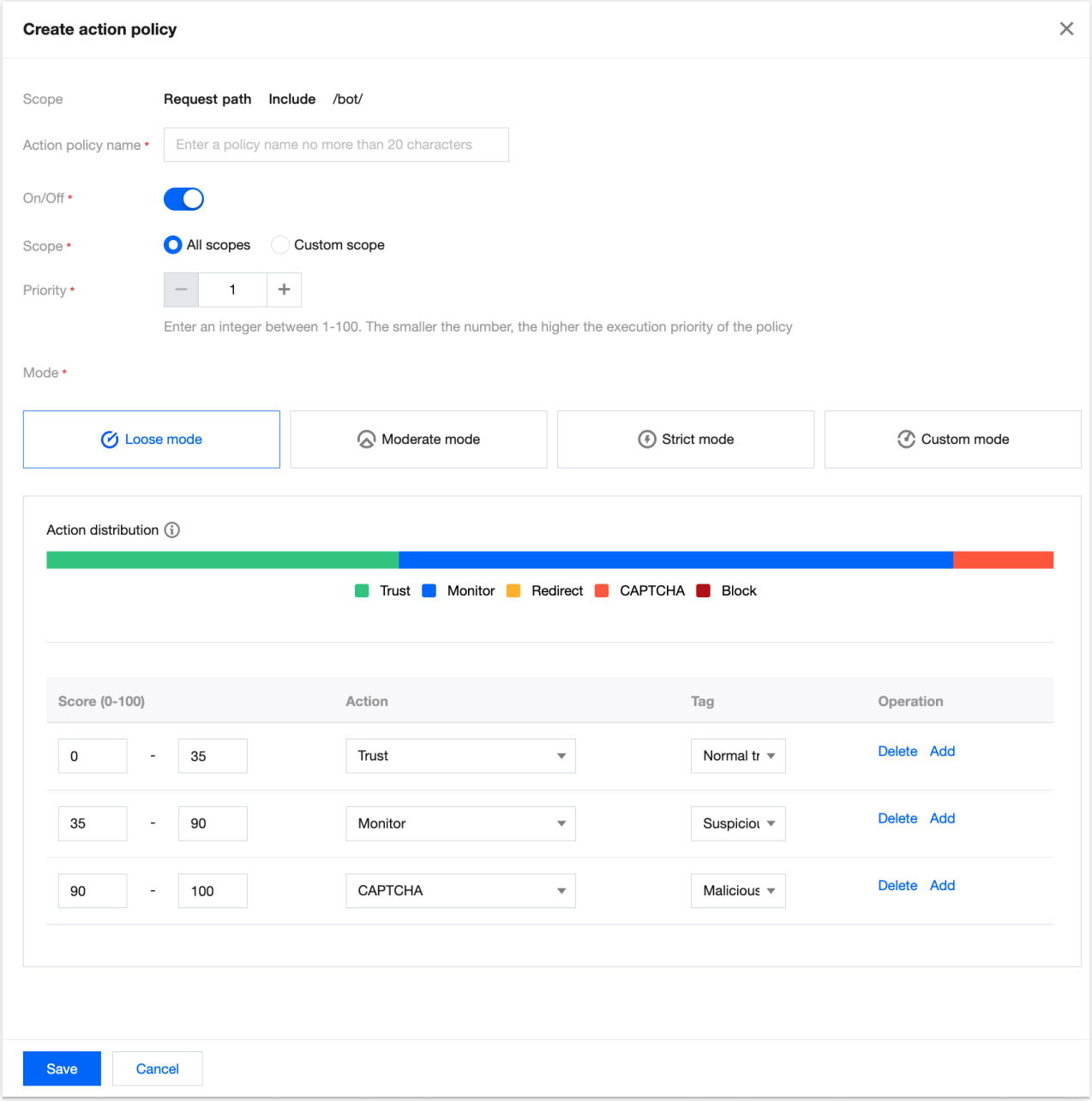
4. If the business doesn't have traffic in that region, the score is abnormal. Then, you can customize an action for more precise settings.

5. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



6. Click the configuration page of a certain scenario, click **Bot flow statistics module**, and click **Add action policy** in the **Action policy module** section.

7. On the displayed page, configure parameters and click **Publish**.



Parameter description:

Policy name: Enter name of the action policy.

On/Off: Specify whether to apply the current action policy.

Scope: The scope of the current action policy.

Priority: Action policy execution priority, which is an integer between 1–100. The smaller the value, the higher the priority.

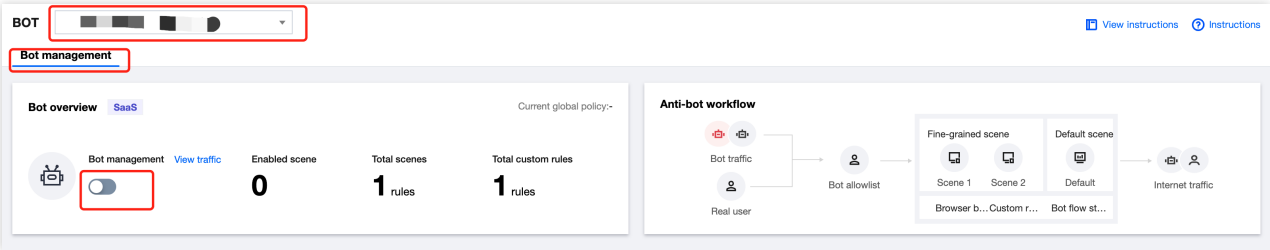
Mode: By default, there are loose, moderate, strict, and custom modes. The first three modes are preset, representing different recommended categories and handling policies for bots at different risk levels in bot traffic management. Once modified, they become the custom mode.

Score range: A score ranges from 0 to 100. Ten score entries can be added to each range, which is left-closed and right-open and cannot be overlapped. You can set a range to null, and then no action will be processed in it.
Action: You can set an action to **Trust**, **Monitor**, **Redirect** (to a certain website URL), **CAPTCHA**, or **Block**.
Tag: You can set a tag of **Friendly bots**, **Malicious bots**, **Normal traffic**, or **Suspicious bots**.
Friendly bots: The bot is friendly and legal for the website by default.
Suspicious bots: The system finds the access source traffic suspicious but cannot determine if it is malicious to the website.
Normal traffic: The access traffic is regarded as from a real user.
Malicious bots: The bot has malicious traffic and is unfriendly to the website.

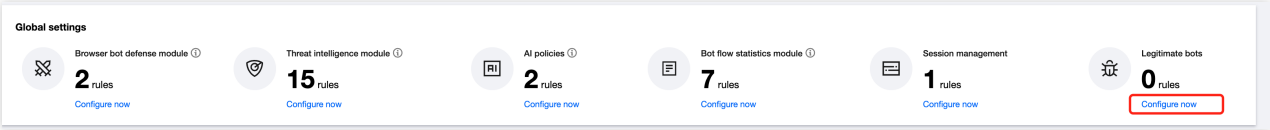
Legitimate Bot

This feature allows legitimate bots (such as search engines and feed bots) to get website data so that the website can be normally indexed.

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.

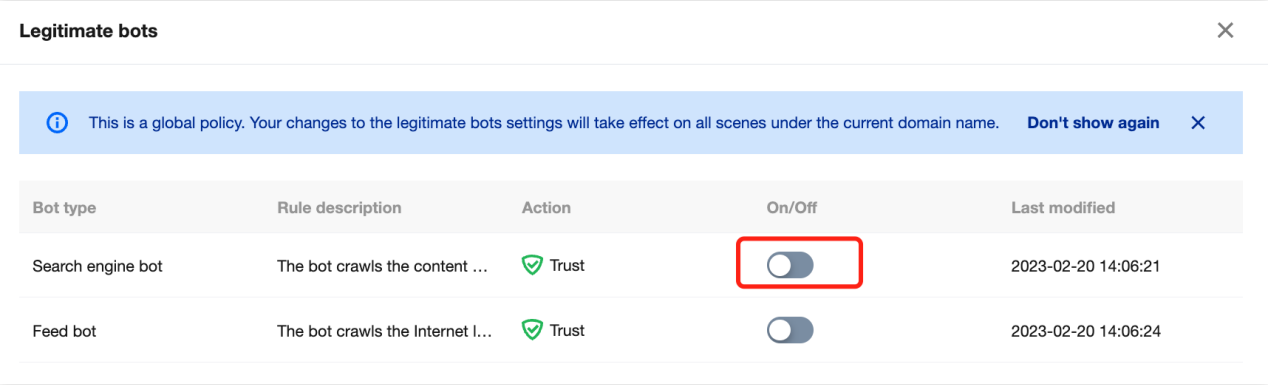


3. In **Global settings** on the **Bot management** tab, click **Configure now** in the **Legitimate bots** module section.



4. On the **Legitimate bots** page, click

to enable the feature.



Custom Rule

This feature allows you to precisely handle compliant crawlers and access requests with different features.

Note:

Currently, when you are creating a scenario-based bot rule, a custom rule set has been preset for the scenario.

This feature analyzes data mainly from bot traffic analysis.

The content **is for reference only and cannot be used as the standard business configuration**. Web crawlers fall into diverse categories and generally vary by business type.

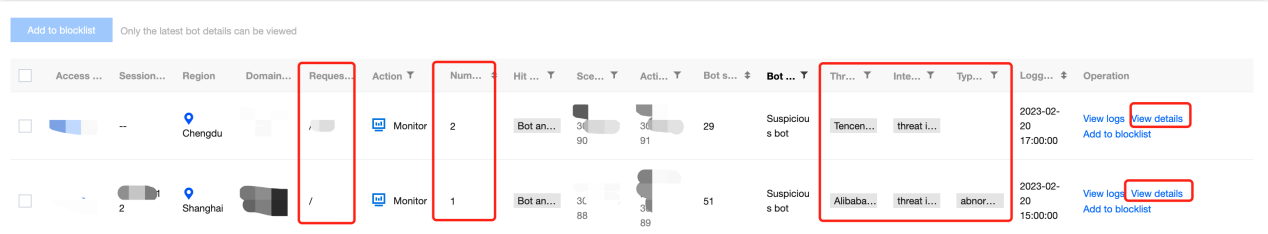
Case details

If requests cannot be blocked by setting an action score, you need to set the abnormal behavior characteristics. After identifying the exception in **Bot traffic analysis**, click **Details** to view the exception data and compare it with normal business data.

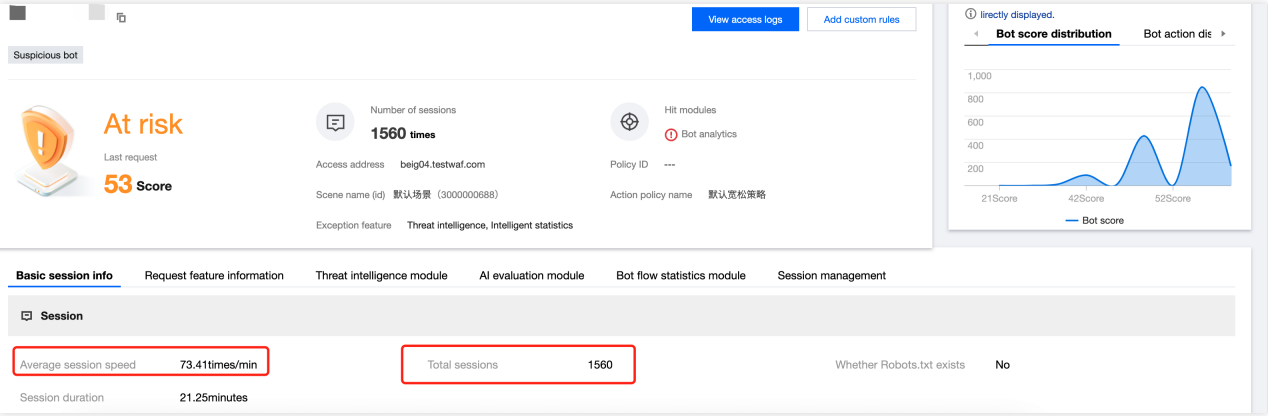
For example, if the URL duplication is `1` , the number of sessions is 100 per minute, and User-Agents are misused , you need to check whether there are similar requests or proxies in the business, and if not, there is a malicious attack. Then, you can view the exception and configure the blocking policy as follows.

Case study

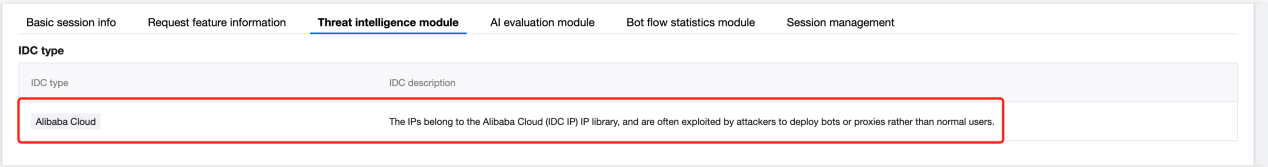
1. Log in to the WAF console and click **Bot traffic analysis** on the left sidebar.
2. On the **Bot traffic analysis** page, select the target domain name in the top-left corner and select the target access source. You can see that the IP request is fast, there is a single URL, and the threat intelligence is IDC.



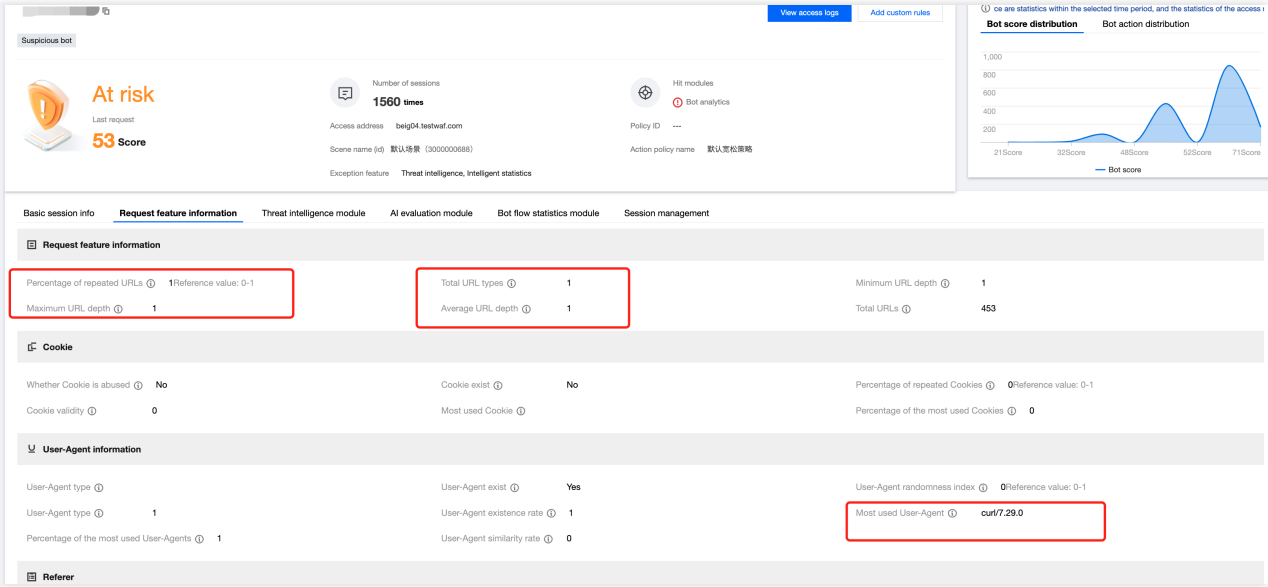
3. Click **View details**. In the **Basic session info** tab, you can view the average number of sessions per minute and the total number of sessions. Then, set the policy accordingly.



4. On the **Threat intelligence** tab, check whether the IP has been used by a real user based on the intelligence data.

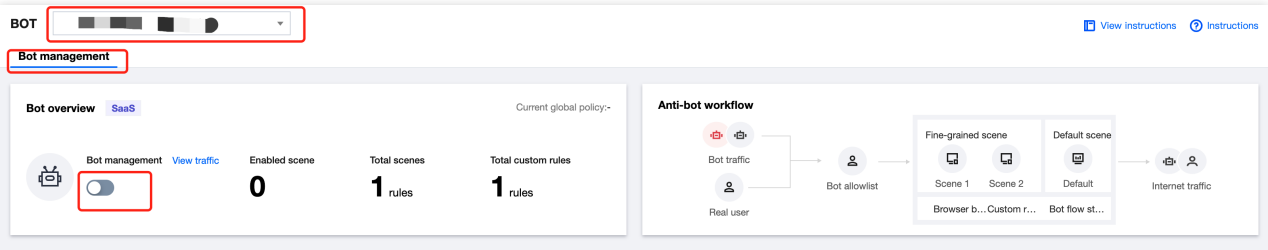


5. On the **Request feature info** tab, view the request details.

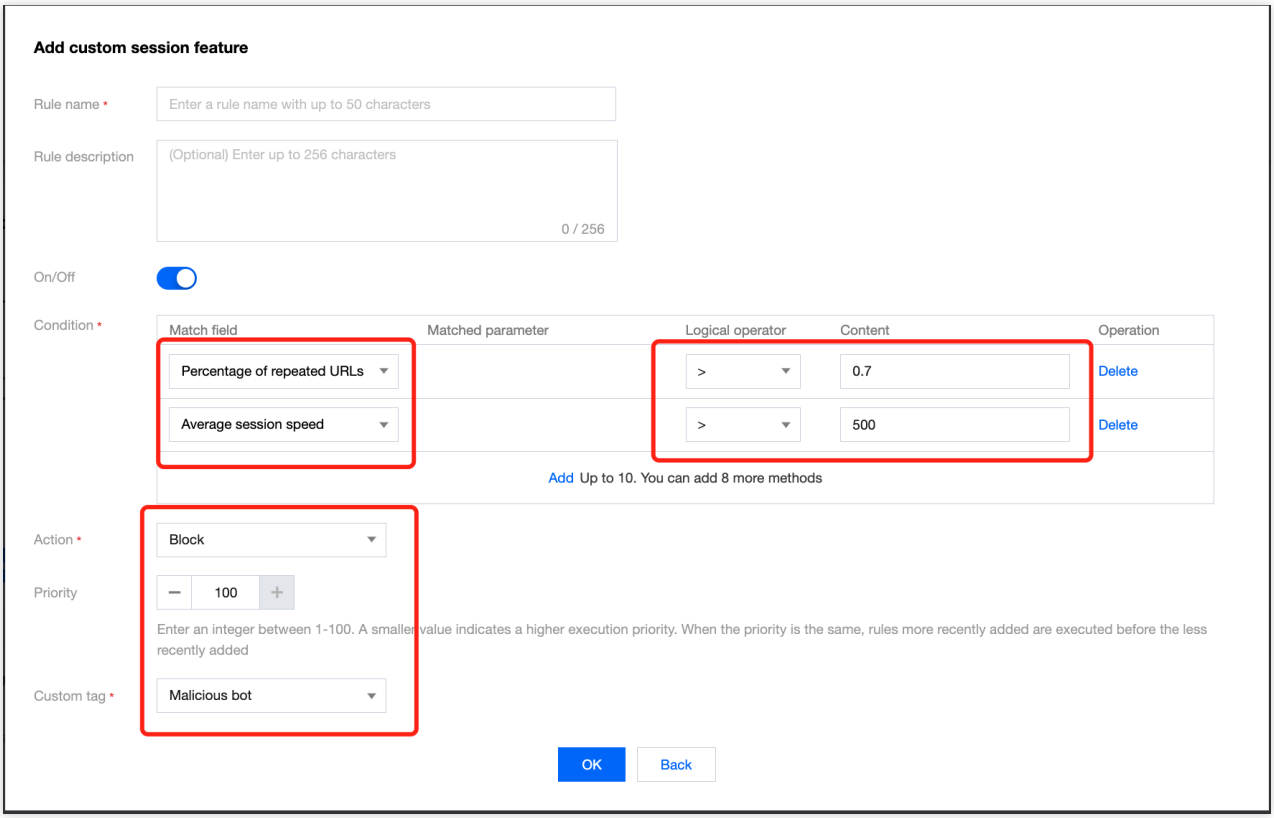


Policy configuration

1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



3. Click the configuration page of a certain scenario and click **Custom rules**.
4. On the **Custom rules** page, click **Add a configuration**. Based on the above analysis, set the percentage of repeated URLs to a value greater than 0.7 (no other data exceeds this value during the process) and the number of sessions per minute to a value greater than 500. Then, click **OK**.



Note:

Currently, when you are creating a scenario-based bot rule, a custom rule set has been preset for the scenario.

API Security
WAF Working with API Gateway

Last updated：2023-12-29 14:53:05

This document describes how to configure WAF to protect APIs on API Gateway.

Prerequisite

You have activated WAF.
You have published an API on API Gateway as instructed in Getting Started.

Directions

Step 1. Bind a custom domain name in the API Gateway console

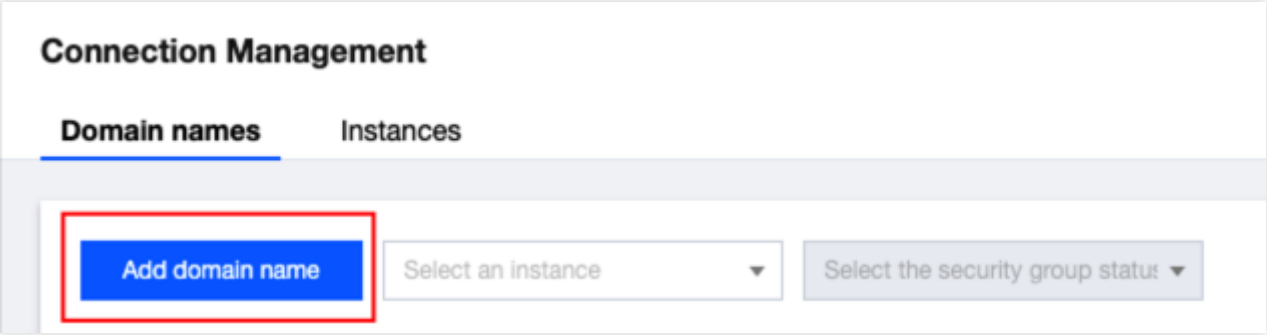
For more information about how to bind a custom domain name in the API Gateway console, see Configuring a Custom Domain Name .

Note

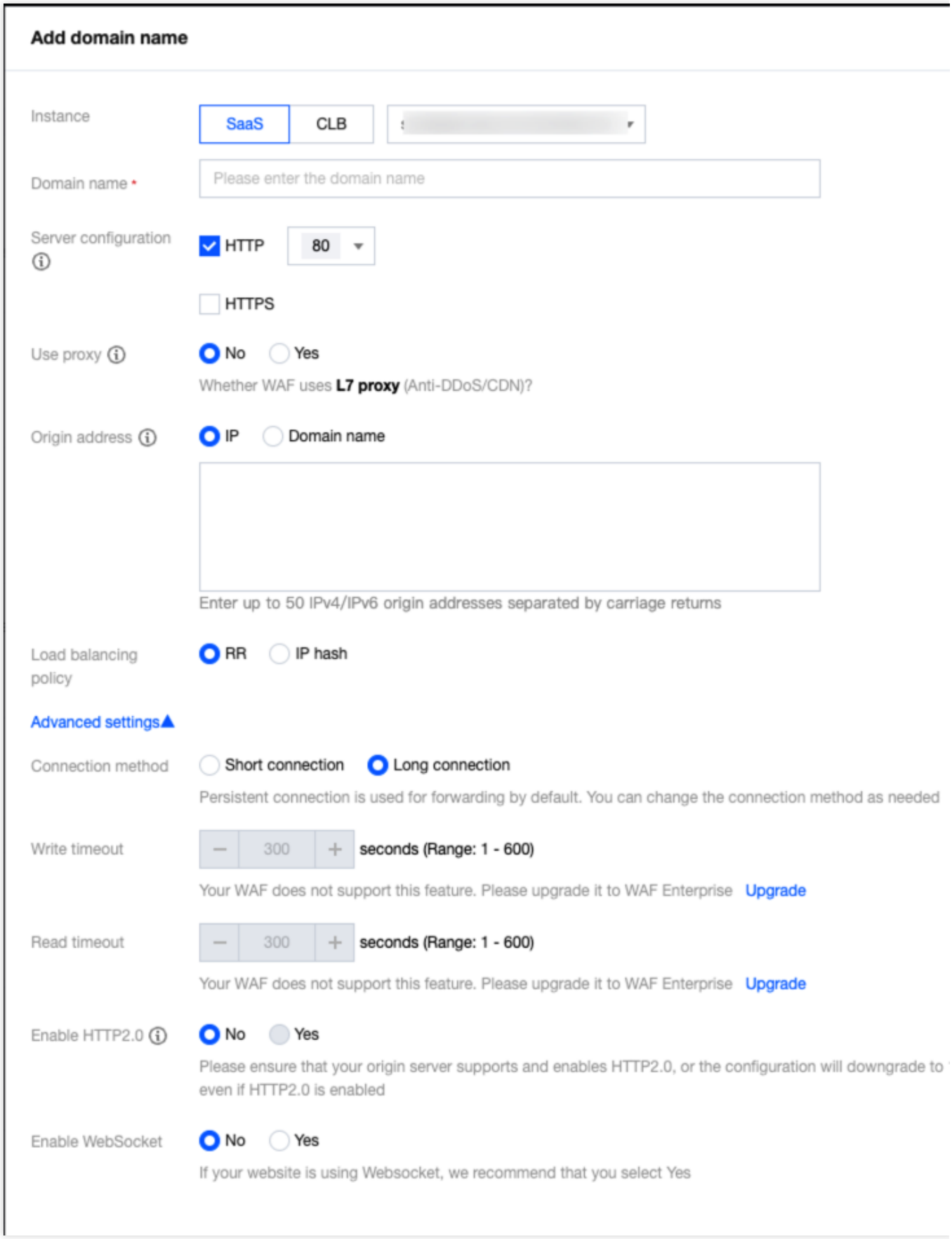
When a custom domain name is bound to API Gateway, the system will check whether you have configured CNAME and resolved it to the service subdomain name. Therefore, you need to configure CNAME and resolve the custom domain name to the subdomain name of API Gateway, modify the DNS record, and point the custom domain name to the WAF CNAME domain name.

Step 2. Configure WAF

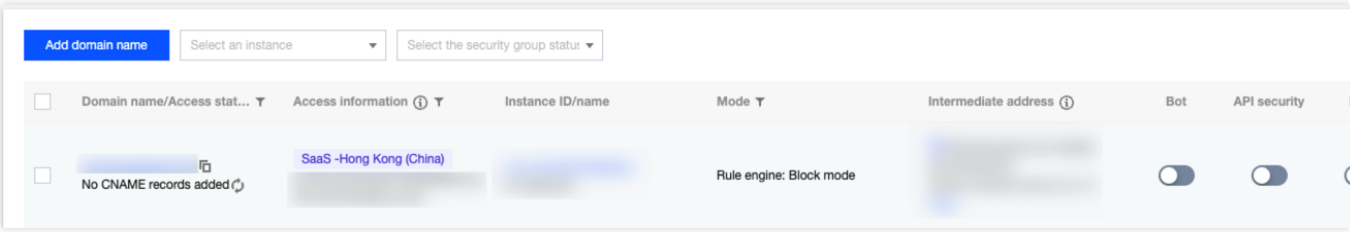
1. Log in to the WAF console and select **Connection Management** on the left sidebar.
2. On the page that appears, click **Add domain name**.



3. Configure required parameters and click **OK**.



4. The domain name should now be in the **No CNAME records added** status.



Step 3. Modify the CNAME record

1. Modify the CNAME record at your DNS service provider and resolve the custom domain name to the WAF domain name.

2. Log in to the WAF console, select **Connection Management** on the left sidebar and then the **Domain names** tab.

API Capacity Protection

Last updated：2023-12-29 14:53:17

Why capacity protection is necessary for APIs?

APIs are designed for automated scheduling and thus vulnerable to network attacks caused by automated scheduling.
Attackers attempt to use replays to automatically send volumes of business traffic with different authentication credentials, resulting in data leakage.
By using automated tools to launch Layer-7 DDoS attacks, attackers initiate continuous requests and occupy the bandwidth of the server and upstream and downstream computing and storage resources, resulting in business instability.
Fuzz testing tools can be also used to conduct targeted attacks and bypass security measures.
In addition, attackers can write automated programming tools to perform resource exhaustion attacks.

Given these threats, APIs can be protected by the following modules.
API capacity protection
API security protection
API asset management
API lifecycle management
This article describes how to implement API capacity protection. Note that during the development lifecycle, the API system stability can be protected and boosted by using **caching, downgrading, and rate limiting** measures.
Cache
Degrade
Rate limits
Increase system access speed and system processing capacity.
When the service or the core process is affected, temporarily block the API access, and unblock after the peak time or the problem is solved.
The system is protected by limiting the rate of concurrent access requests or the rate of requests within a time window. Once the rate limit is reached, services can be denied, queued or waited, and downgraded.
Although these effective protection measures can be implemented in the process of development, operation and deployment, they are too cost-consuming and throughout the lifecycle of API security, it is necessary to provide API capacity protection for all API assets.
Therefore, adjustments need to be made for each API, leading to exponentially increased workload. You can quickly protect the capacity of business APIs with the following methods.
Note
API analytics is currently in beta and only supports 3 domain names. Submit a ticket if you need to use it.

How to protect the capacity of APIs?

When protecting API capacity, in addition to the measures described above, you can also use the API capacoty module in WAF. This article explains the following 9 methods for target APIs.

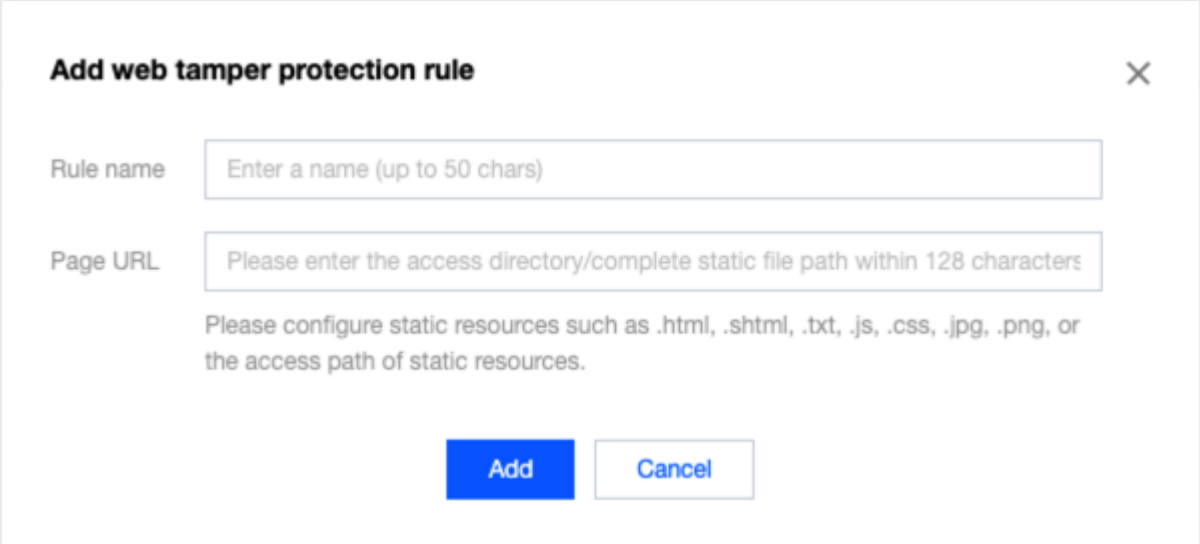
Protection Method	Description
API content caching	Cache static API resources.
API access downgrade	Block API exceptional traffic to protect business system stability.
API rate limiting	Limit the overall access request rate of the API.
API scheduling rate limiting	Limit the access speed of the client scheduling API.
Protection for API sensitive calls	Protect sensitive APIs from scheduling abuse and ensure no data breach.
Protection for API resources	Protect API resources from being overused.
Protection for key APIs	Perform 2FA/MFA authentication when key APIs are scheduled.
API signature verification	Verify that the client is a real client for access.
API exception scheduling protection	Protect the API from being accessed by abnormal resources.

API Content Caching

Public APIs are frequently called to return content using a lot of resources. If the content will not be continuously updated for a period of time, the content can be cached to reduce computing and bandwidth resources of the API server.

Here you can use the **Web tamper protection** module in Basic Security to quickly cache the API content.

1. On the page displayed, click **Add rule**, and the rule adding window will pop up.
2. In the pop-up window, configure relevant fields and click **OK**.



Field description:

Rule name: The rule name can be up to 50 characters. You can search for rules by name in attack logs.

Page path: Path of the page to be protected from tampering. You need to enter a specific URL rather than a path.

Note

The specified page is limited to static resources such as .html, .shtml, .txt, .js, .css, .jpg, and .png.

After the rule is added, when a user accesses this page for the first time, WAF will cache the page, and subsequent access requests will be directed to the WAF-cached page.

3. After the tamper protection rule is added, it will be enabled by default.

API Rate Limiting

API rate limiting involves two parts:

Limiting API speed

If API speed limits are imposed on the server, some clients may be unable to access business. When APIs are attacked by a large amount of traffic and the API speed is limited on the backend, most of the access traffic will be considered exceptional and blocked. So it is recommended to limit the **client calls**.

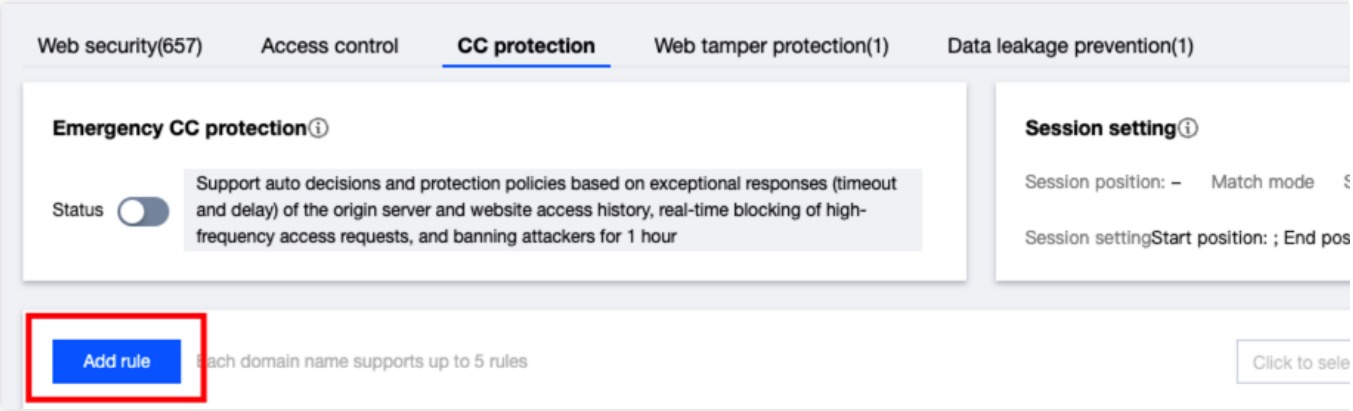
Limiting API calls

The API calls allowed for each client can be restricted through CC protection and bot management.

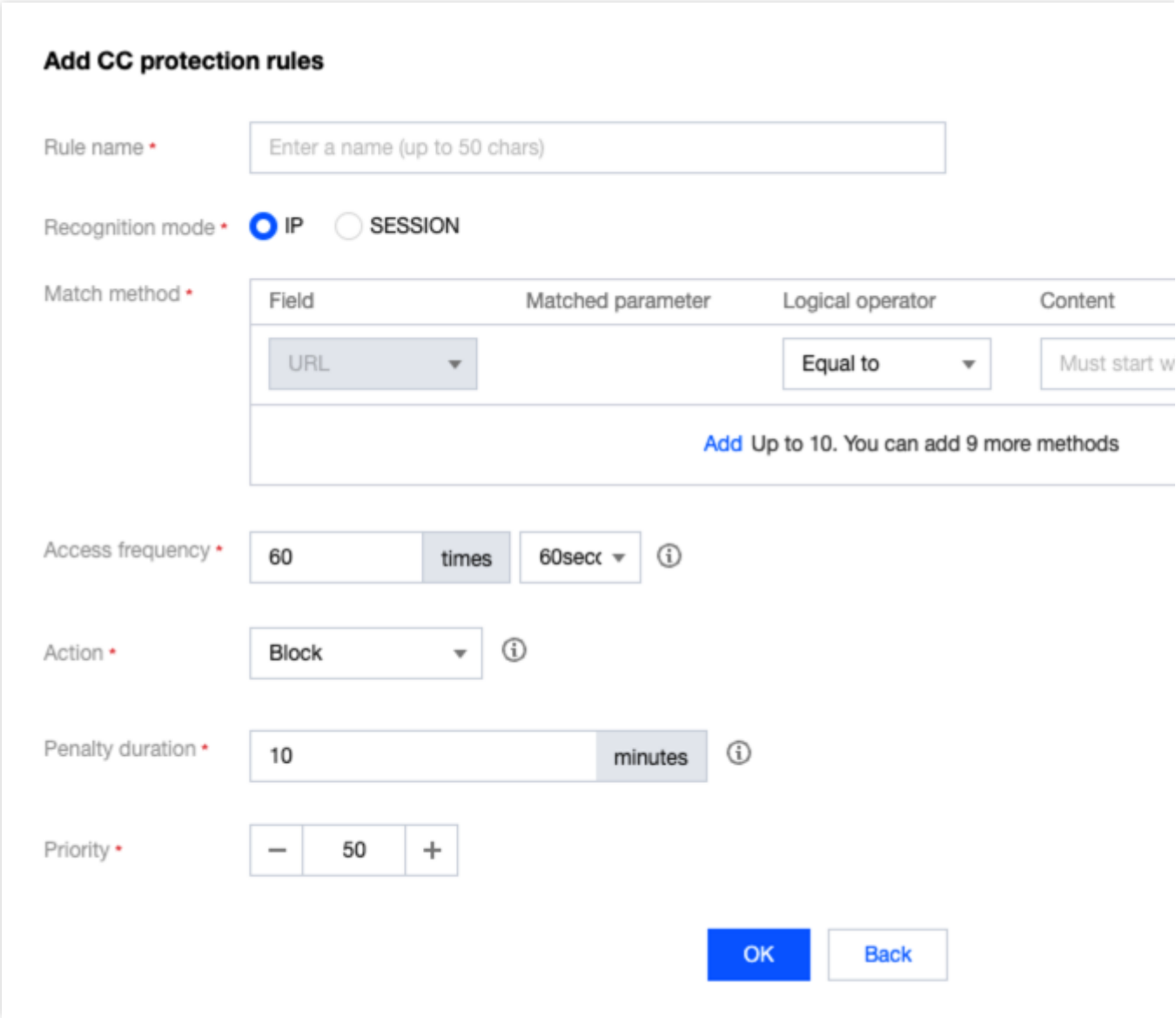
CC protection settings

With CC protection, you can set the overall access frequency of each client. Once the client exceeds the expected limit, it will be handled as configured.

1. On the CC protection page, click **Add rule**.



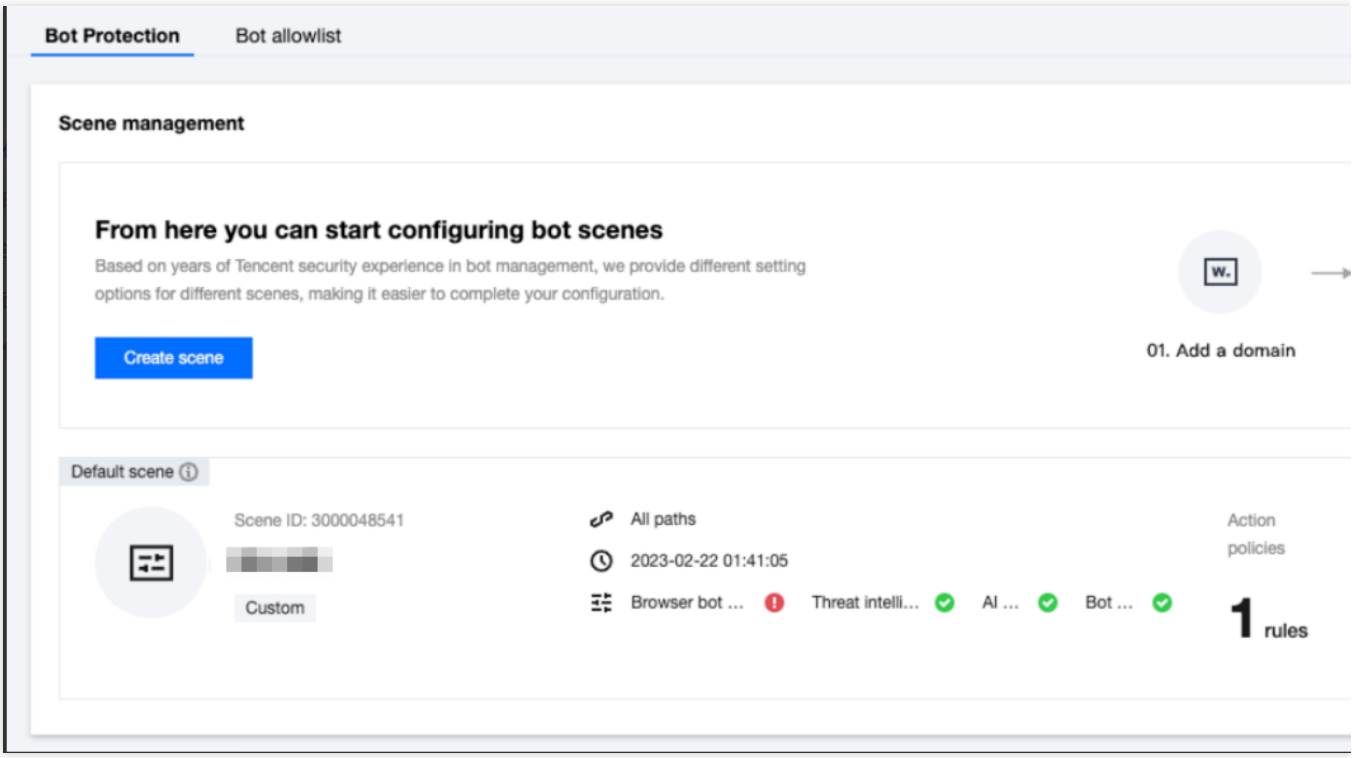
2. In the **Add rule** window displayed, configure the parameters and click **OK**.



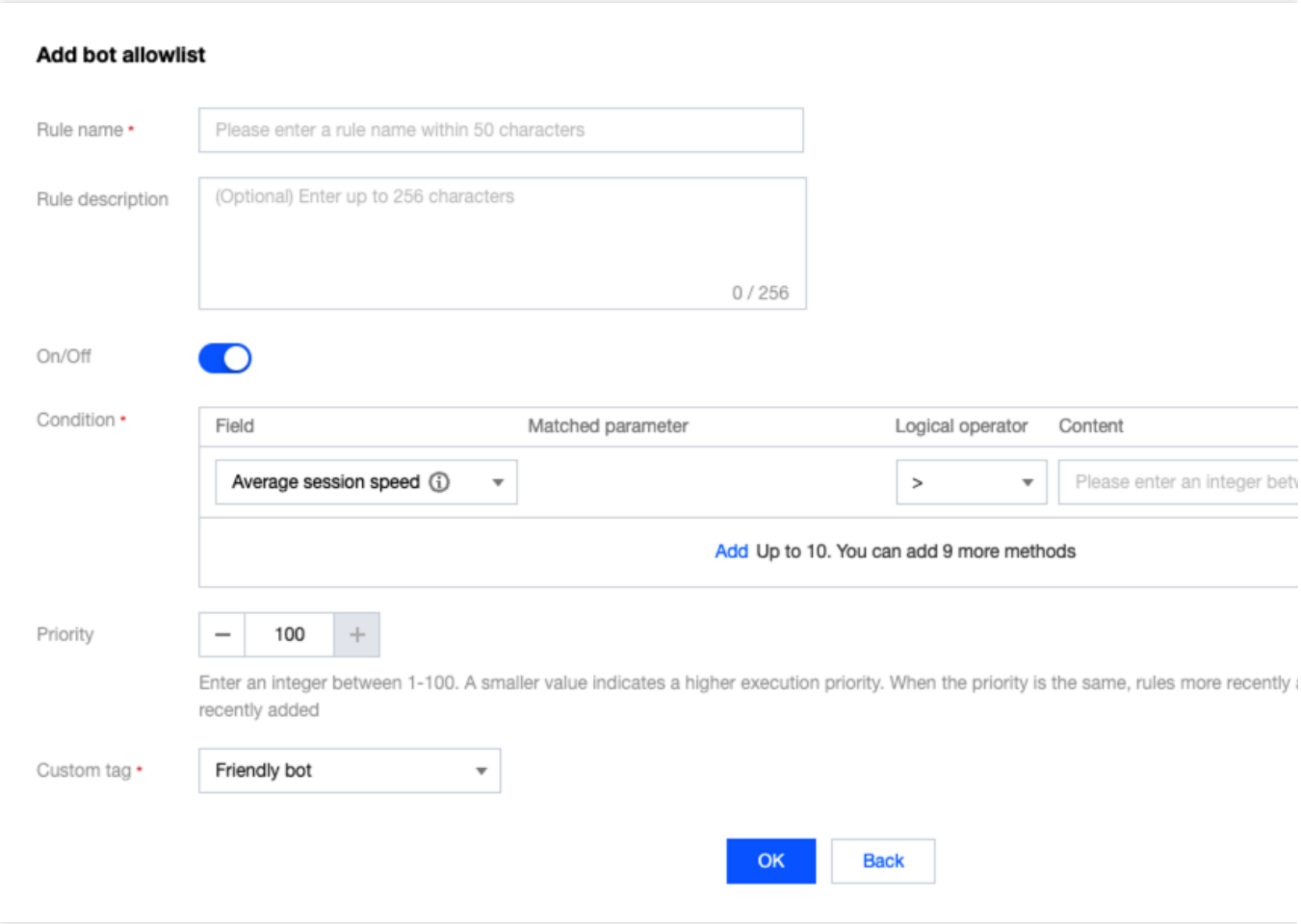
Bot management settings

Go to Bot management > **Bot protection**, configure the average session speed to control the continuous access speed of each client.

1. In the **Scene management** module, view the target scene by clicking **View configuration**.



2. Click **Add rule**, configure parameters, and click **OK**.



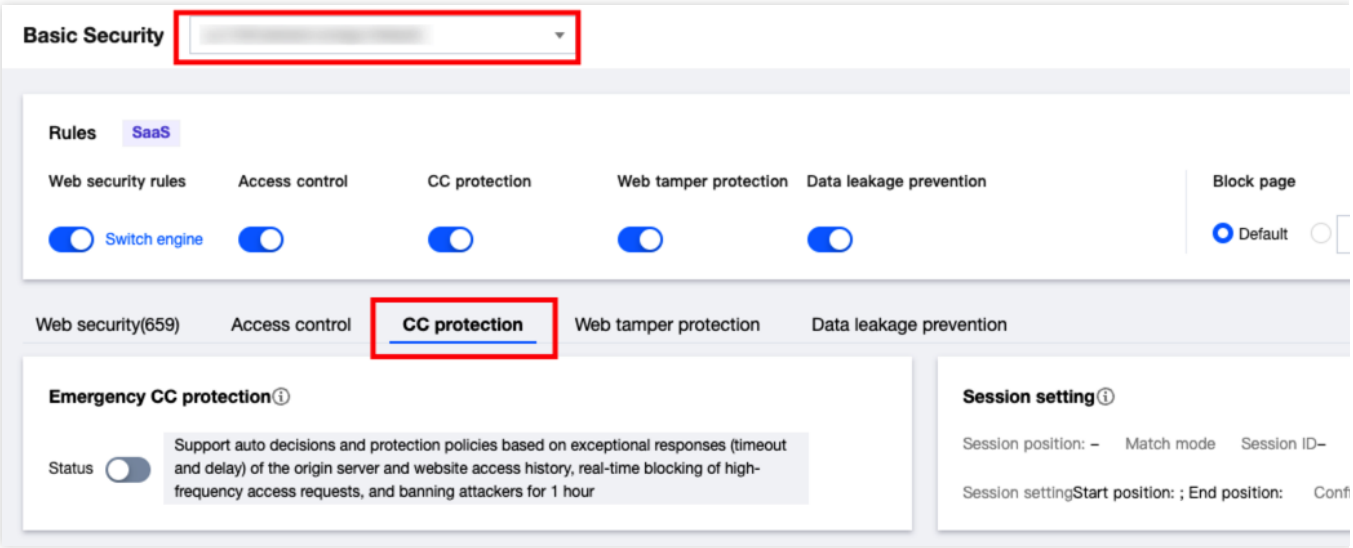
Session settings

With the dramatically growing number of IPv4 IPs in the current network, many IP operators have started using a NAT IP, which allows multiple business clients to use one public IP. If rate limits are only enforced on business IPs that share one NAT IP, IP rate limiting can be easily triggered with false positives. However, restricting the number of requests made will be much less effective if the rate limits are set too high.

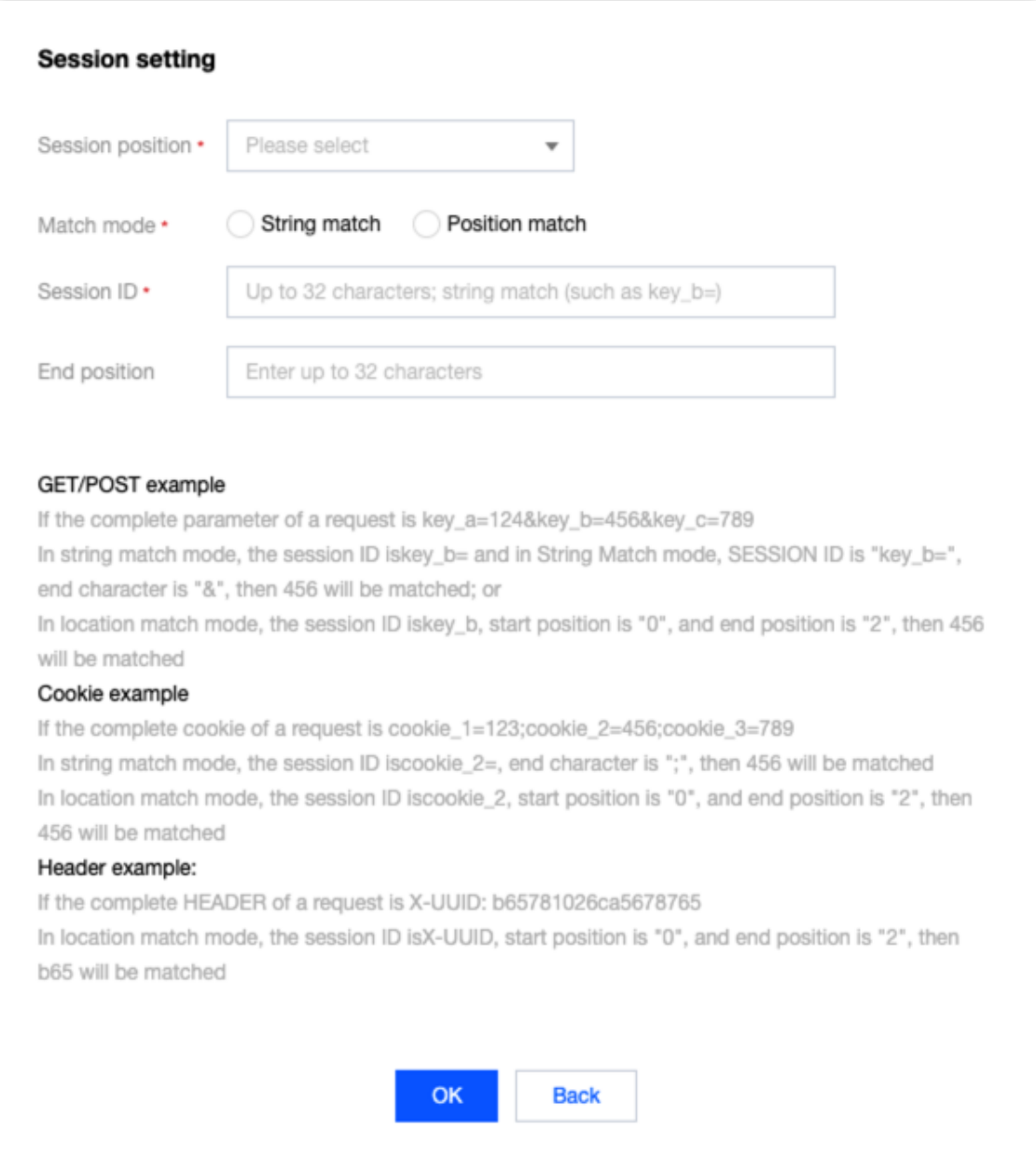
Therefore, you can configure session settings, which can **automatically distinguish different clients under the same IP and impose business rate limits** for a single client.

Session settings

1. Log in to the WAF console and select **Basic Security** on the left sidebar.
2. On the basic security page, select the target domain name in the top-left corner and click **CC protection**.



3. In the **Session settings** module, click **Set**.
4. Configure parameters and click **OK**.



Parameter description:

Session position: Select HEADER, COOKIE, GET, or POST, where GET and POST are HTTP request parameters rather than HTTP headers.

Match mode: Except HEADER (only supports position match), all support matching by string pattern or position.

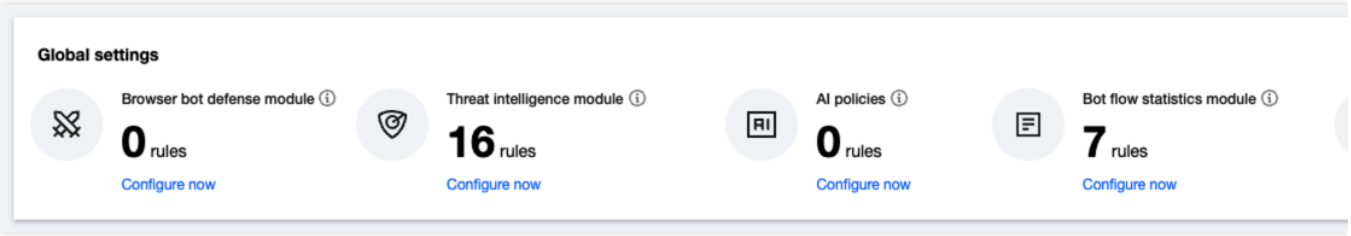
Session ID: The identifier of the session. It can be up to 32 characters.

Start position: Specify the start of the string or the position. It is an integer between 1 and 2048 and only up to 128 characters can be extracted.

End position: Specify the end of the string or the position. It is an integer between 1 and 2048 and only up to 128 characters can be extracted.

Conversation settings

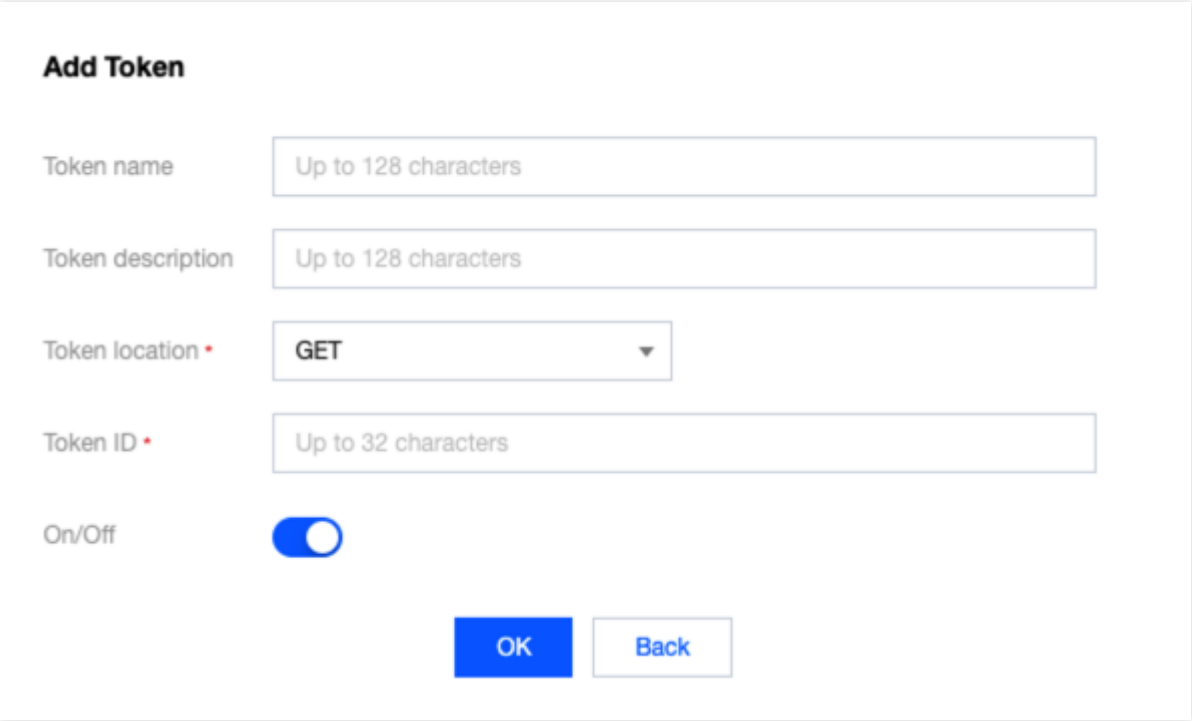
1. Navigate to Bot management > **Advanced settings**, click **Configure now**.



2. On the session management page, click **Add a configuration**, configure parameters and click **OK**.

Note

A token ID should be a continuous tracking ID, such as the value of `set-cookies` after login.



Parameter description:

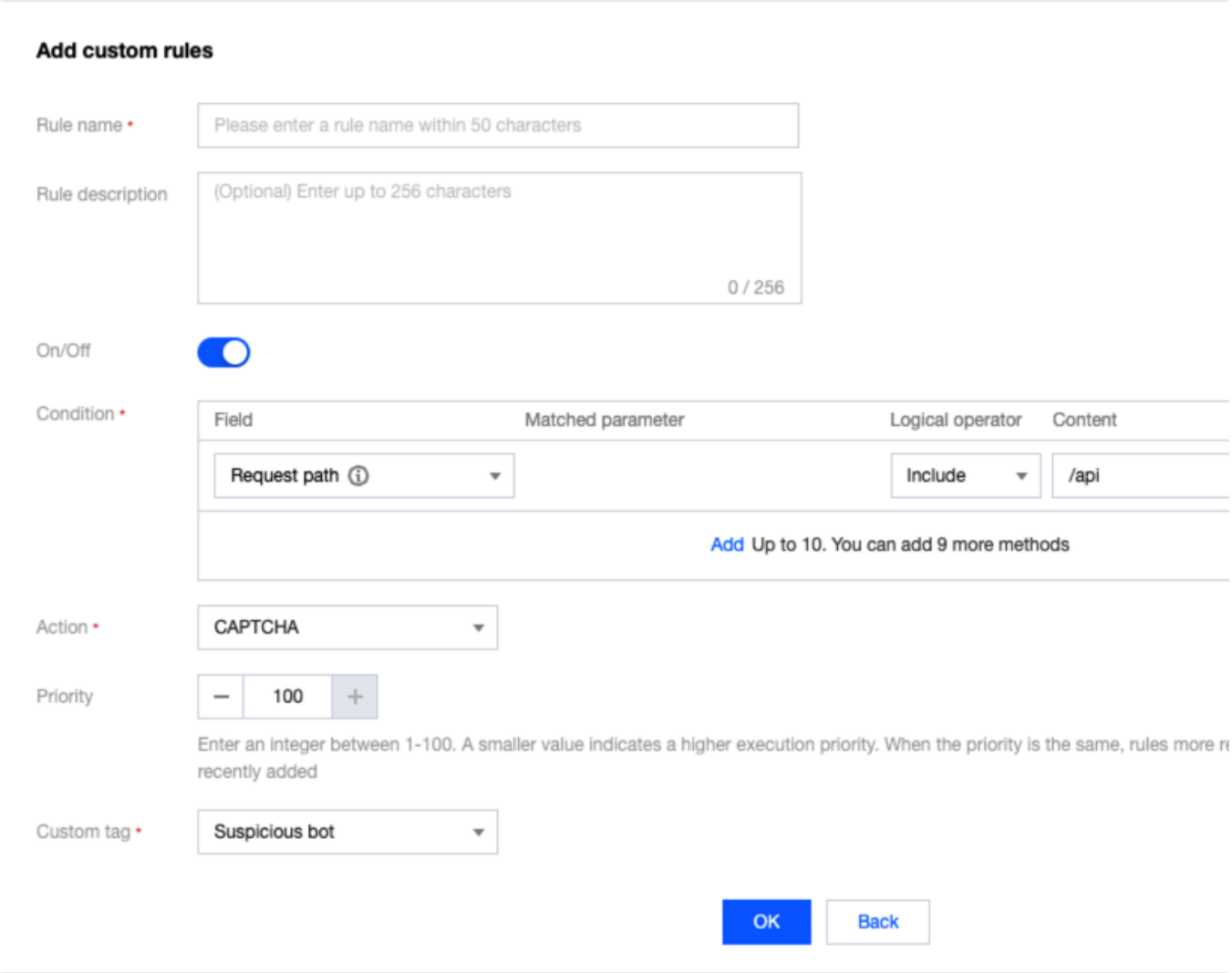
Token location: Select HEADER, COOKIE, GET, or POST, where GET and POST are HTTP request parameters rather than HTTP headers.

Token ID: The identifier of the Token.

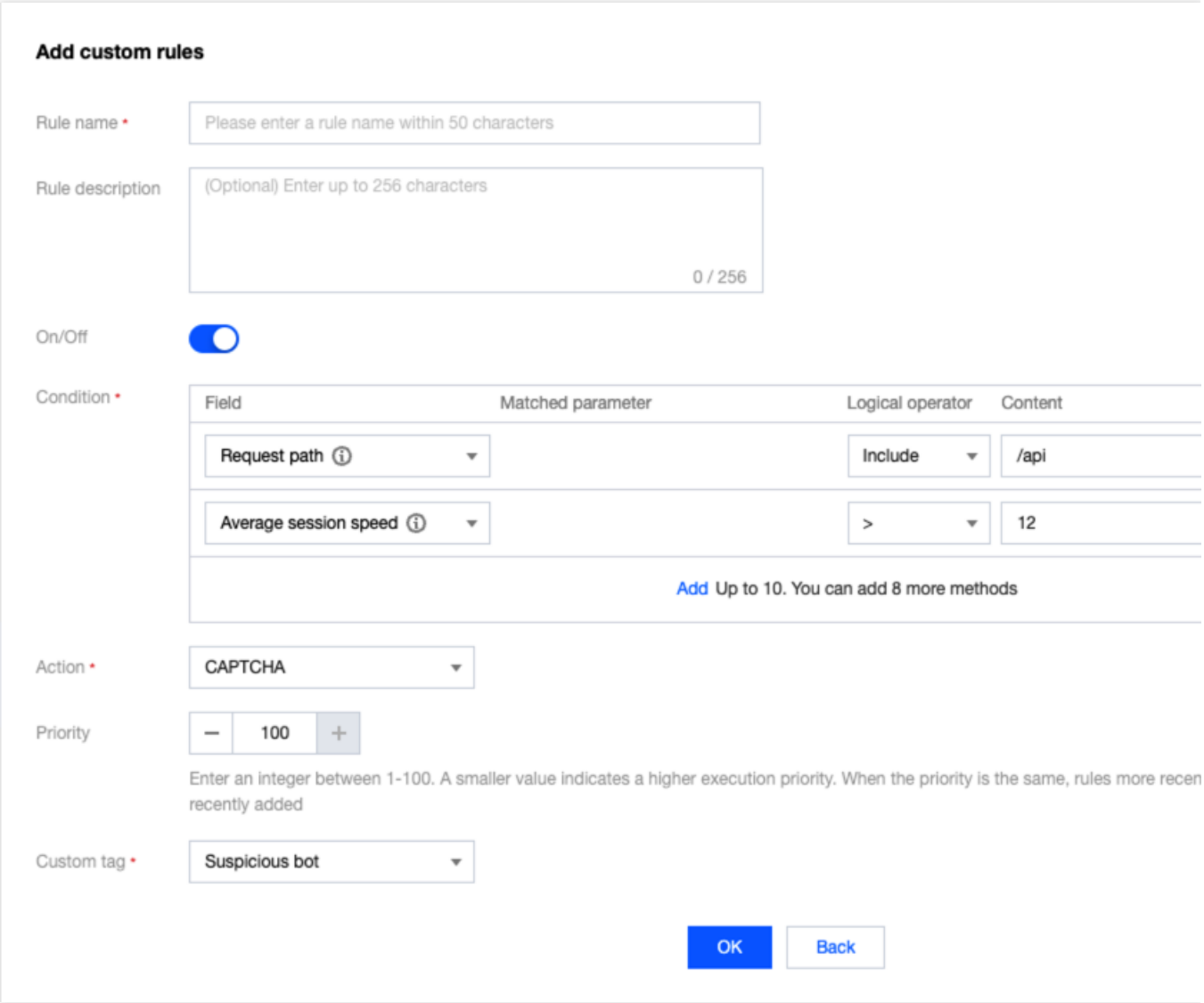
Limiting API calls

Each sensitive API should have a limit on the number of calls. For example, if the SMS API service is not rate-limited, the APIs could suffer abusive consumption and incur excessive charges. If these sensitive APIs are verified by 2FA/MFA or other authentication techniques before being called, abnormal API scheduling can be effectively reduced. You can limit API calls in Bot management > **Bot protection**.

Performing authentication before sensitive API calls



Limiting the total API calls per client can make within a session



How to authenticate the client access to APIs?

There are many ways to verify the client's signature, including but not limited to:

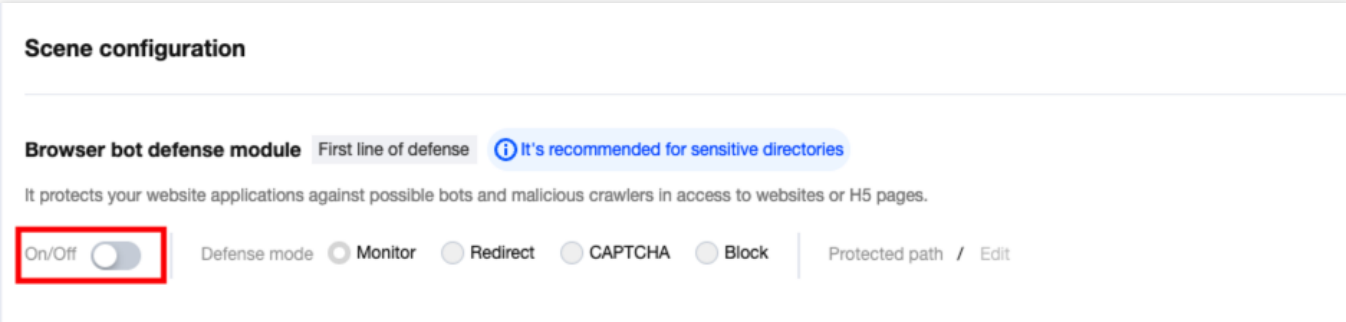
Mutual TLS authentication.

Client signature verification.

Client challenge authentication.

Authentication can be enhanced by applying mTLS and client signature challenges, etc.

Meanwhile, browser bot defence can be enabled in WAF to authenticate API data on the client side. For more details, see Client Risk Identification.



API Data Security and Enhancement

Last updated：2023-12-29 14:53:30

APIs allow all computer platforms and operating systems to access data in different formats, such as tracking APIs that can enable users to track the location of goods purchased online.
Many organizations focus more on fast delivery of APIs and applications rather than safeguarding security, contributing to API attacks and data breaches in recent years.
The table lists three API call scenarios:

API Type	Description	Security Status Quo
Public API	Public APIs are exposed on the Internet, allowing anyone to access services from anywhere. Callers can schedule data and processes by passing necessary fields into APIs. Such APIs require the highest level of security and usability monitoring.	While there are few restrictions on public APIs, such as authorization restrictions, loopholes are frequent to detect in business authentication logic, and attackers prefer to target and bypass these APIs through automated fuzz testing and targeted testing.
Internal API	Internal APIs are usually deployed and operated in a data center or private cloud network for internal use, mainly for operation management and internal services.	Using internal APIs has more restrictions, such as authentication restrictions, with low authentication and security strength. Such APIs are vulnerable to targeted attacks and thus have become the culprit for data breaches.
Channel API	Channel APIs are usually deployed and operated in a data center or private cloud network, providing specific external partners and suppliers with limited access to internal APIs to extract and manage data. Such APIs are more sensitive to data leakage than data extraction.	The access control level is higher than internal APIs but lower than external APIs. It's the same case with security control, which is guaranteed mainly through API gateway. When supply chain attacks happen, channel APIs are easily utilized for data abuse due to the lack of monitoring and supervision mechanisms.

Why API Sensitive Data Discovery Matters

According to the Salt Labs State of API Security Report, Q1 2023, 43% considered zombie APIs the most concerning API security risk and 22% were worried about account takeover/abuse; 83% lacked confidence in organizations' API inventory.

Enterprises are so concerned about API assets as security risks are often hidden in the unknown zombie APIs, unknown shadow APIs, and unknown sensitive data exposure, all rooted in the lack of comprehensive asset visibility. Through such APIs, attackers are likely to launch targeted attacks to extract and expose sensitive data, and even expand the attack surface to gain unauthorized access to servers and databases.

Even if enterprises have begun managing zombie APIs, zombie parameters can be easily overlooked and pose a huge security threat. Zombie parameters may exist in APIs and can be called by attackers even though they are not exposed in the API release. Common zombie parameters include debugging parameters and system property parameters configured during the development and testing cycle. Once attackers successfully exploit vulnerabilities such as batch allocation to obtain unauthorized responses, enormous amounts of business data and user data can be easily collected.

Directions

Step 1: Discover API assets

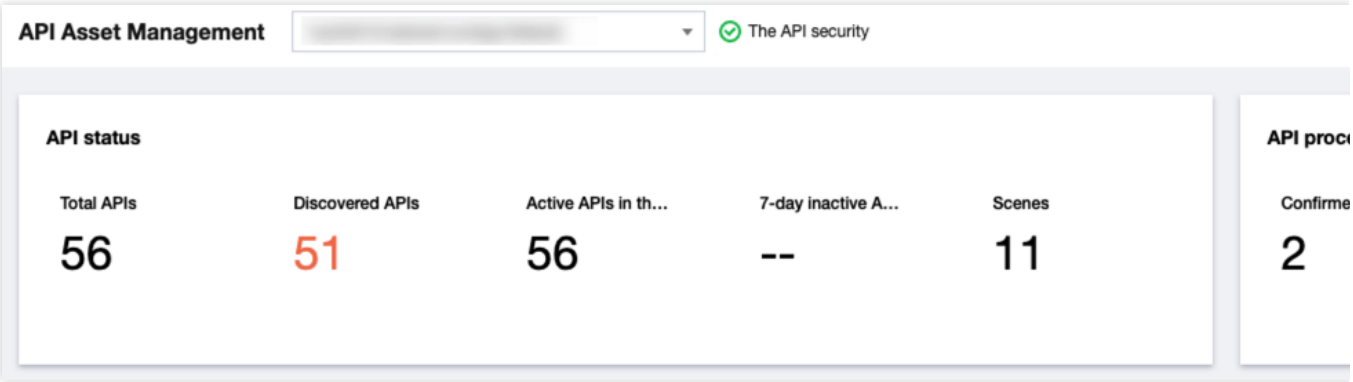
1. Log in to the WAF console and select **API Analytics** on the left sidebar.

Notes

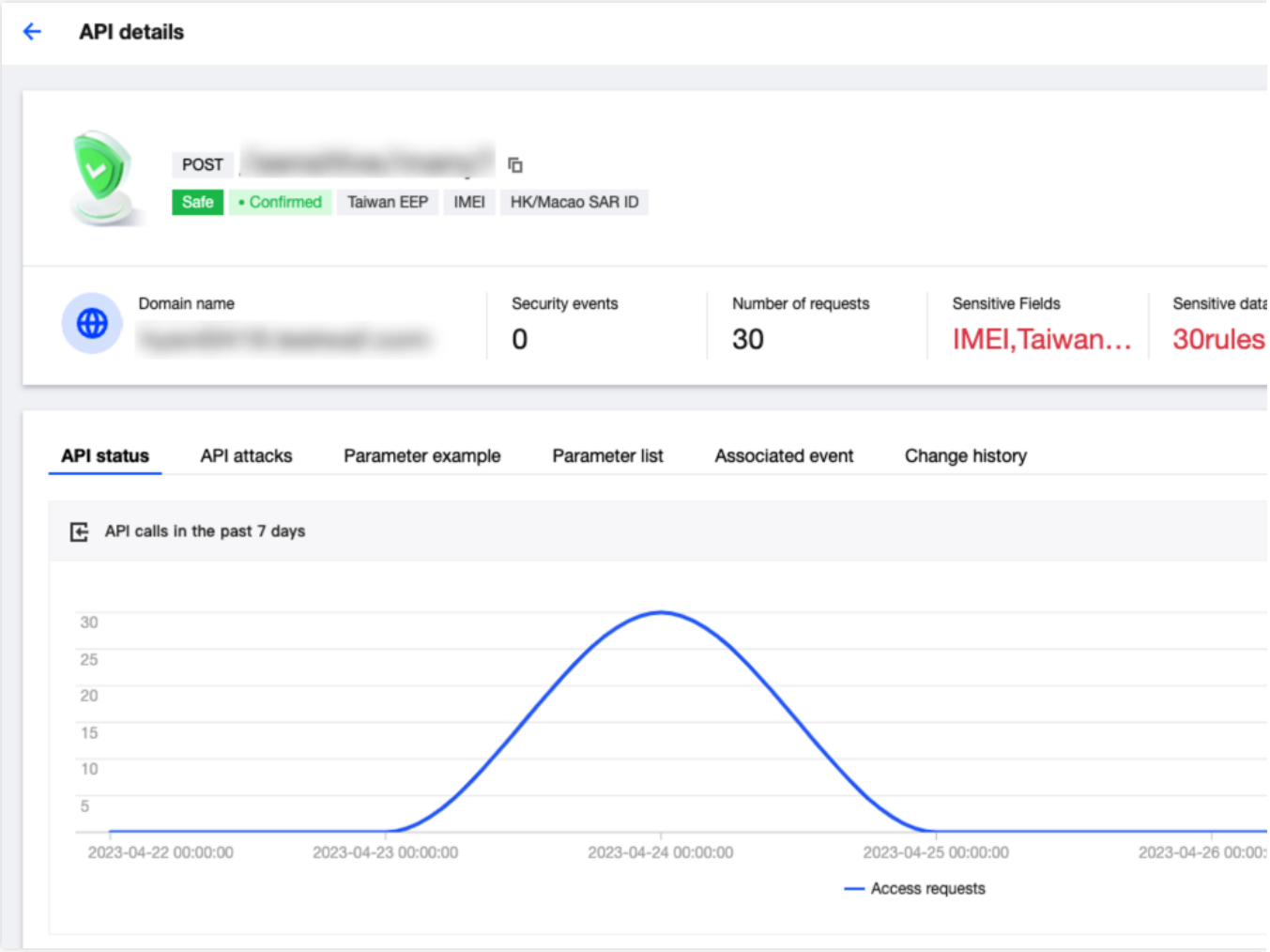
API Analytics is currently in beta testing and only supports 3 domain names. To use this feature, submit a ticket.

2. On the page that appears, select a domain name to protect and toggle on the switch

.

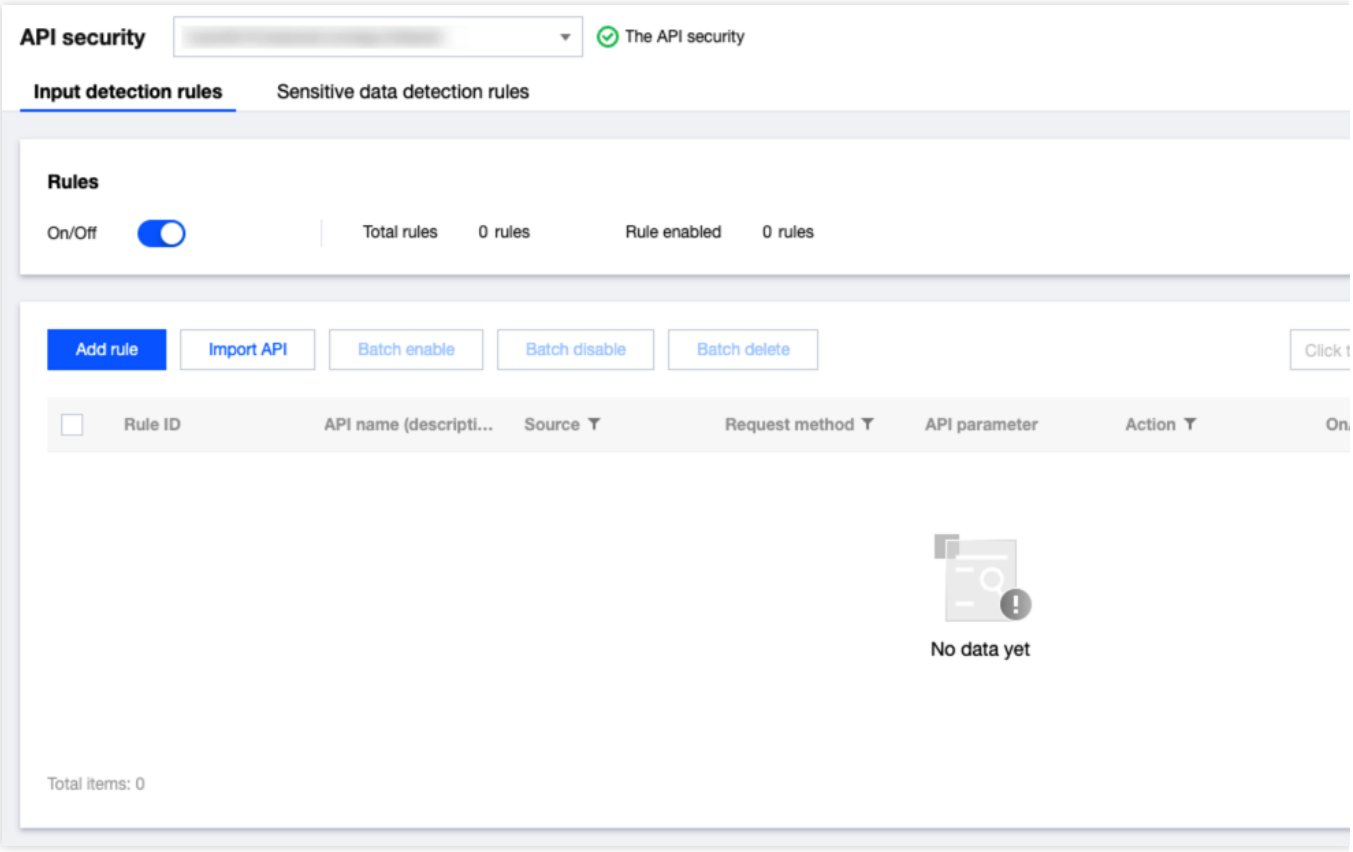


3. When it's on, you can view related information on the **API details** page.

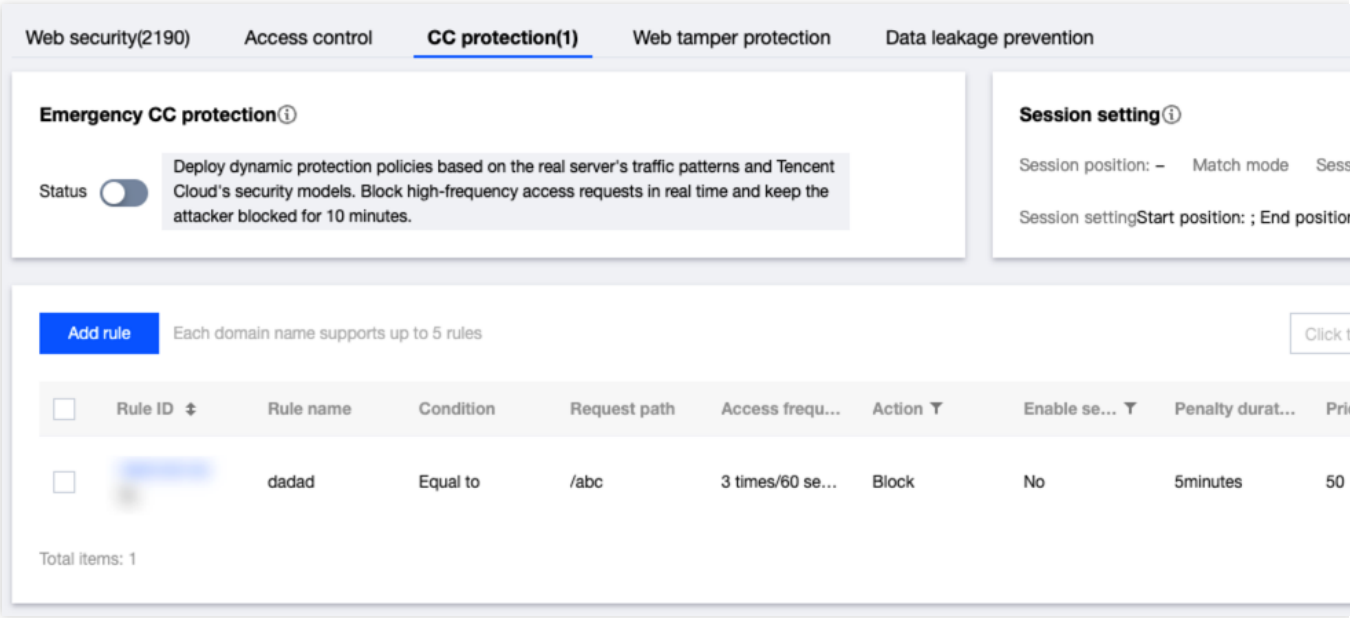


Step 2: Enhance API security

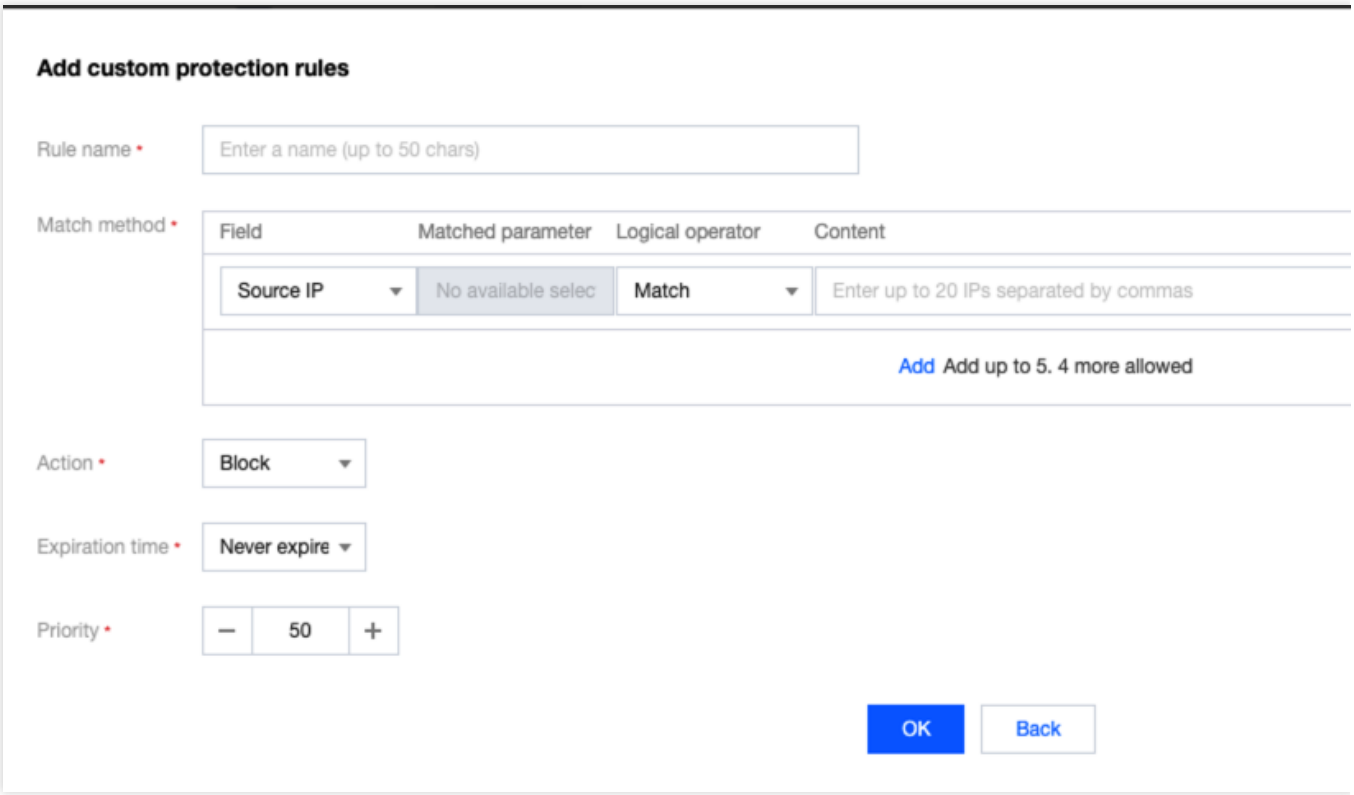
1. On the Basic Security page, select the **API security** tab and create rules.



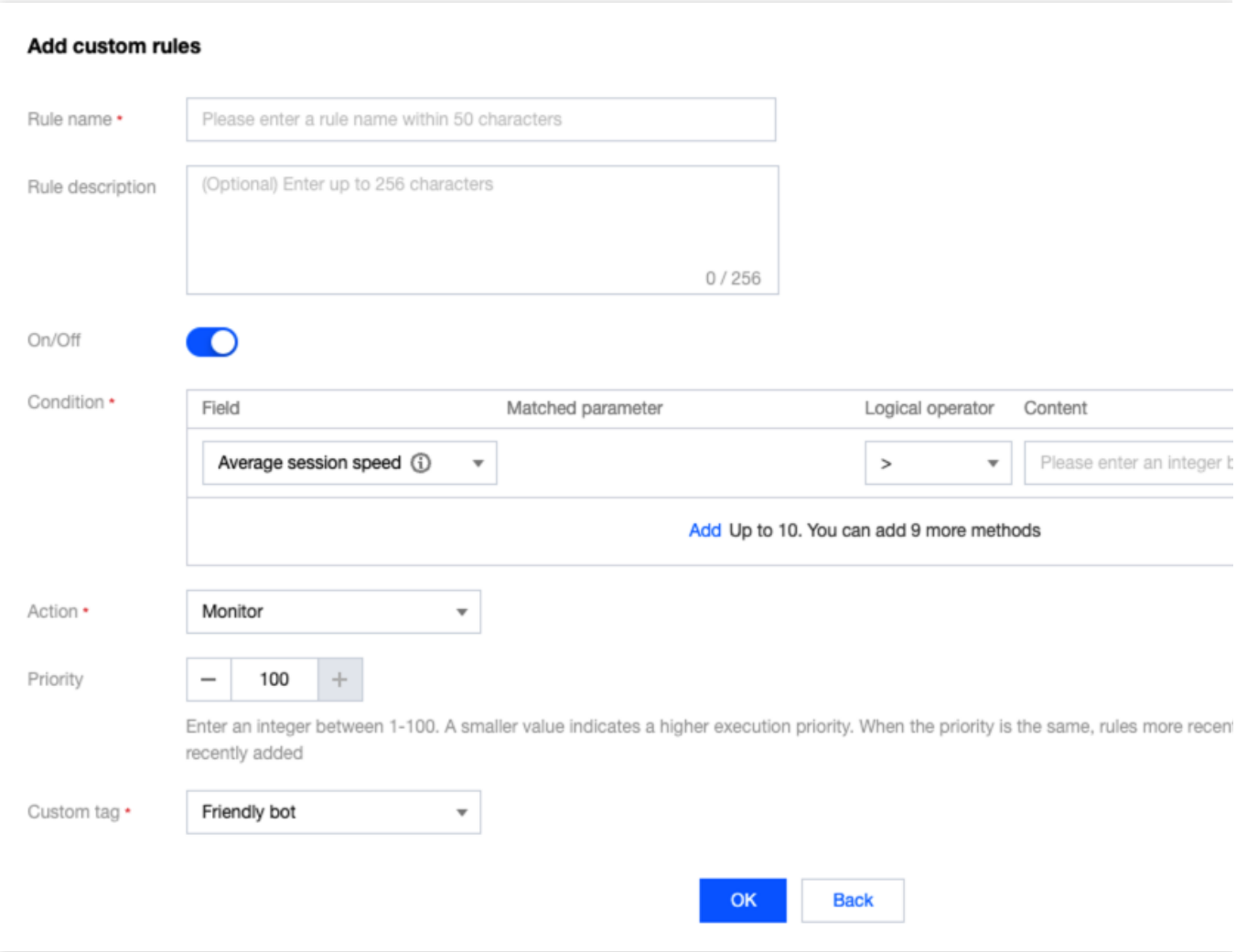
2. On the **CC protection** tab, configure capacity protection settings based on relevant APIs.



3. On the **Access control** tab, click **Add rule** to implement protection for sensitive operations based on relevant APIs.

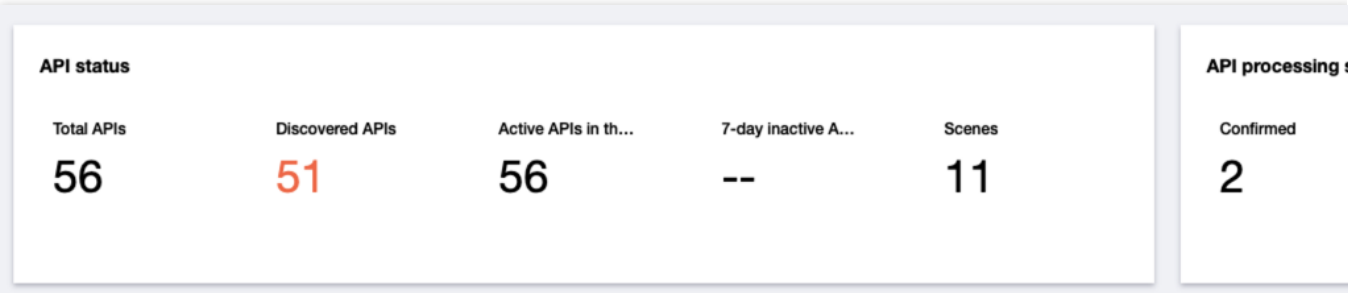


4. On the **Bot and Application Security** page, configure settings to detect API behavior exceptions.

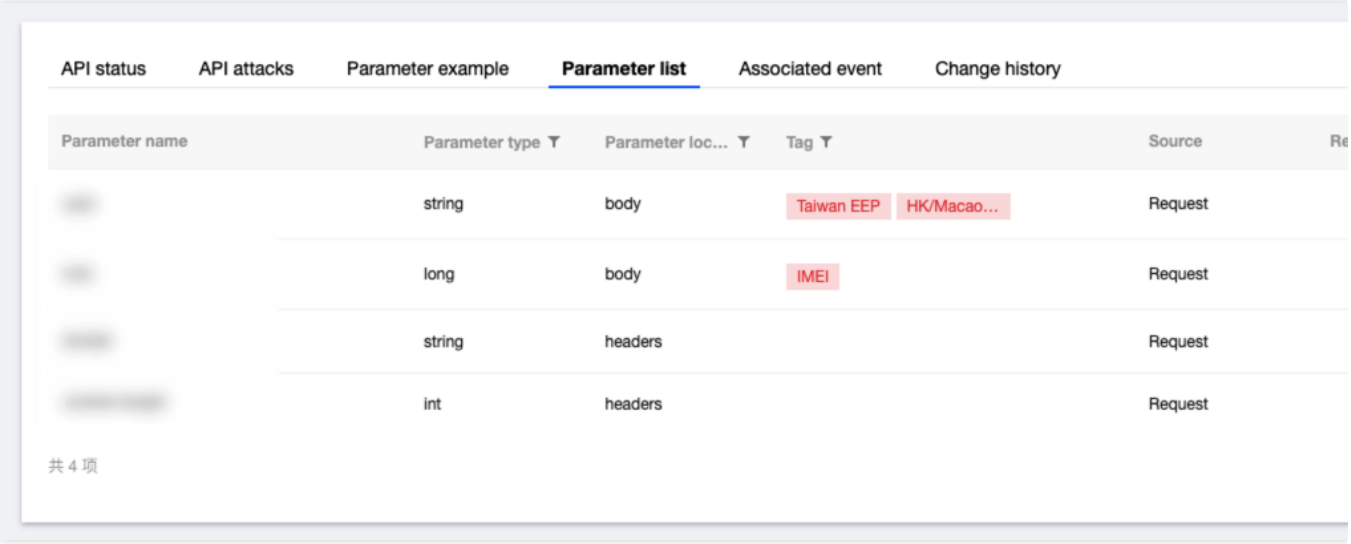


Step 3: Manage API lifecycle

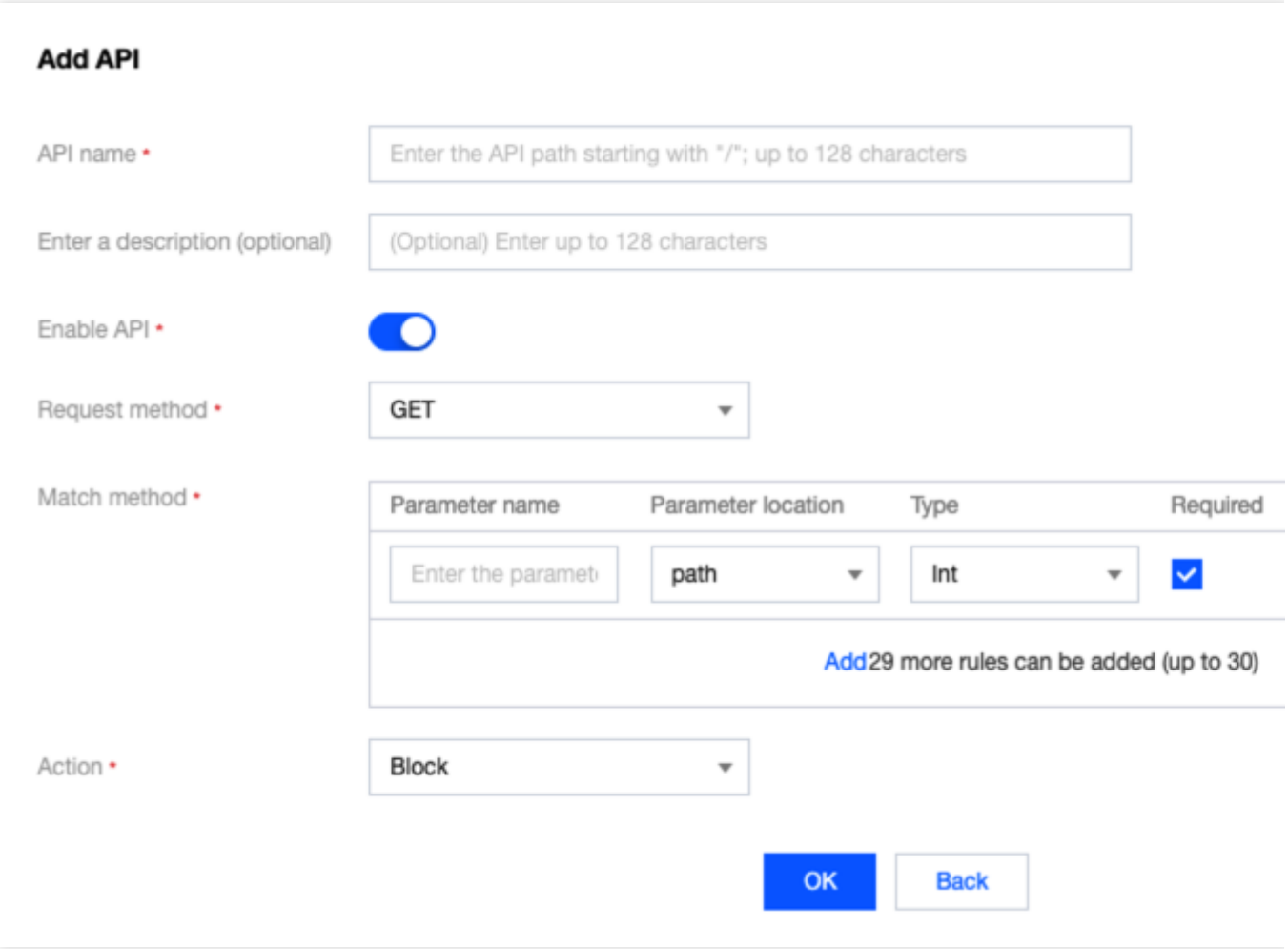
1. Keep track of the number and status of APIs.



2. Detect updates of API parameters.



3. Reprocess APIs when they are no longer in use.



API Exposure Management

Last updated：2023-12-29 14:53:43

Background

Though most of today's digital experiences are empowered by APIs, API security remains a top concern for most CISOs. With the spread of digital transformation across industries and the rise of malicious threats targeting APIs, there is a big gap between API security and actual needs, leaving organizations plagued by incomprehensible attack surfaces and a lack of proper security measures.
APIs are now at the center of digital experience, giving support for core features of mobile and web applications, micro-service architecture and regulations. According to Akamai's statistics, API requests account for 83% of all application requests and the number of hits is expected to reach 42 trillion in 2024. However, APIs have become a prime target for attackers as they are more vulnerable to attacks compared with traditional web forms. A prediction from Gartner that API abuse would be the most common attack type by 2022 also highlights the seriousness of API security issues, which arise from these challenges:

Migrating applications to the cloud increases attack surfaces

As cloud computing has come into widespread use, SaaS applications are increasingly migrated to the cloud and reaching more users, exposing APIs to the cloud. Compared with traditional data centers working in a single-point mode, both East-West and North-South traffic may become the attack surface of APIs.

API security is neglected to fuel innovation

Agile development is a popular method that focuses on individuals and interactions, working software, customer cooperation and response to changes. Although innovation efficiency and flexibility are increased, proper measures to ensure API security are ignored when building software.

Attack risks are incurred due to API invisibility

Since APIs are written by programmers, few people realize the existence and maintenance. On the other hand, unprotected APIs are vulnerable to attacks that could be triggered by network traffic, reverse code, and security vulnerabilities.

Security measures are missing due to underestimation of API risks

The likelihood and impact of API risks are seriously underestimated when running applications and thus APIs including third-party APIs are not adequately protected.
To implement API governance, proper management of API assets and attack surface need to be prioritized.

About API Exposure

API exposure can be classified into two types:

Type	Description
Data exposure through APIs	Data exposure occurs through internal APIs.
	Data exposure occurs through partner APIs.
	Data exposure occurs through zombie APIs.
	Data exposure occurs through external APIs.
	Data exposure occurs through trial APIs.
Data exposure through parameters	Data exposure occurs through sensitive parameters in APIs.
	Data exposure occurs through backend parameters in APIs.

API exposure makes way for attackers to exploit insufficiently protected APIs, leading to unexpected security incidents such as data and permission leakage and API abuse.
Meanwhile, sensitive and backend parameters in open APIs can also be easily targeted and utilized by attackers.

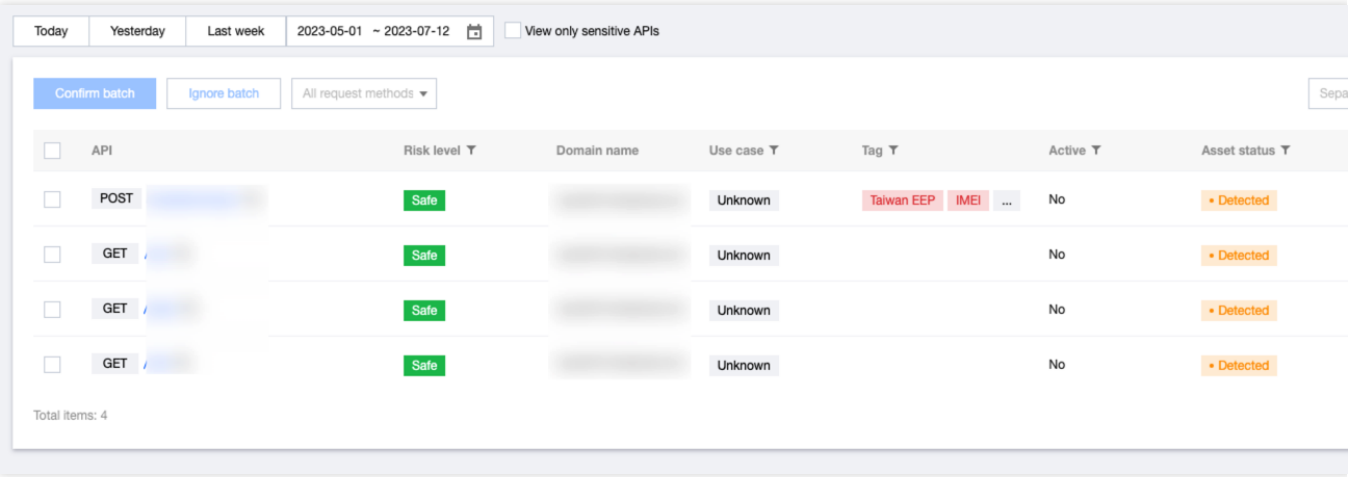
Detecting API Exposure

1. Reduce risk exposure by automatic identification of API call relationships and comprehensive and continuous inventory of all APIs.
2. Reduce data exposure by continuous monitoring of sensitive data flows and custom sensitive data detection.
3. Identify unsafe operations by continuous sorting of access accounts and multi-dimensional recording of their behaviors.

The cornerstone of exposure detection is API discovery, which can be achieved using API Analytics. It enables you to discover and manage APIs, monitor exposure surface as well as view comprehensive information about sensitive assets (such as tag, risk level and status).

Note

API Analytics is currently in beta testing and only supports 3 domain names. To use this feature, submit a ticket.



API Behavior Control

Last updated：2023-12-29 14:53:54

Background

Thriving in the era where everything can be an API, it is necessary to know how to quickly deliver products and services in response to customer needs for digital enterprises. Meanwhile, APIs provide access to increasingly complex applications and massive sensitive data, so they've become a primary target for hackers.
In recent years, many well-known international enterprises have suffered a huge blow due to negligence with API security. There has been a 681% increase in attackers in the past 12 months, and 95% of organizations have experienced API security incidents, according to the State of API Security Report Q1 2022 released by Salt Labs. However, most organizations are not prepared to deal with these challenges, with over a third (34%) having no API security strategy.
Using APIs involves the transfer of large amounts of data. Through WAF, you can secure data access by categorizing and desensitizing data, and prevent data theft by identifying data leakage and blocking abnormal access and connection.

Exceptional API Behaviors

Launch attacks without obvious features.
Abnormal access to services.
Transfer of large amounts of data.
Access from abnormal sources.
Exploit outdated or zombie APIs.
Overexpose data.

Handling API Exceptions

Detecting and investigating abnormal API access behaviors is the best way to find and fix security vulnerabilities in daily security operations. In the WAF console, you can use **API Analytics** and **Bot Analytics** to quickly identify API exceptions, so as to enable rapid closed-loop security operations

Note

API Analytics is currently in beta testing and only supports 3 domain names. To use this feature, submit a ticket.
Detect and investigate API abnormal access behaviors as follows:

1. Detect exceptional requests.

On the Attack Logs page, identify abnormal access behaviors in logs and track their activity.

On the API Analytics page, identify abnormal APIs, check API logs and track their activity.

On the Bot Analytics page, identify API access requests assigned with abnormal scores and track their activity.

2. Get the unique UUID of the abnormal access request and examine the incident scope by the UUID.

After **Access Logs** is enabled, each log entry has a unique UUID, which allows you to analyze and track user activity, API access logs as well as bot behaviors.

3. Identify typical user behavior anomalies.

User access behaviors are inconsistent across different APIs. For instance, it is highly likely to cause an exception to login APIs when there are too many access attempts.

4. Identify whether there are any exceptions from access.

Check whether the access source and login location is abnormal and whether the calls are made from the business side.

5. Identify whether there are any exceptions from returned content.

Check whether the accessed parameters (such as body size) are exceptional.

Check whether the returned content is exceptional.

6. Check the relevant API and user information.

Handle exceptions after identifying abnormal access behaviors, user and API information.

Integration
Combined Application of WAF and Anti-DDoS Pro

Last updated：2023-12-29 14:54:08

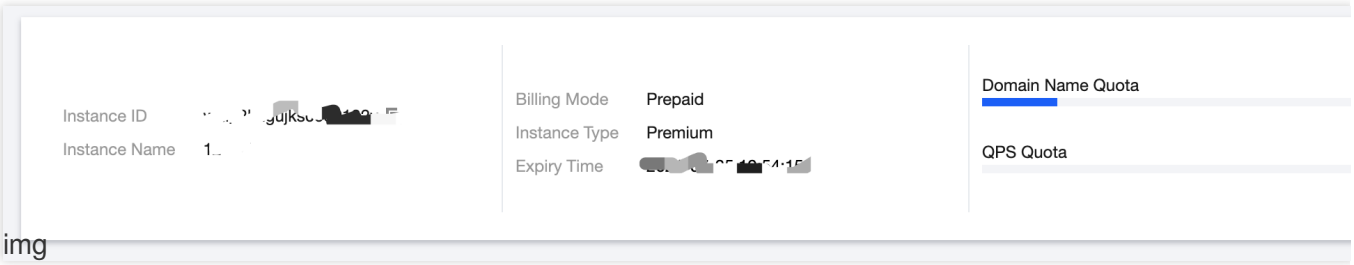
Scenarios

Web Application Firewall (WAF) is able to defeat CC attacks. WAF can work with Anti-DDoS Pro to provide an all-out protection against non-HTTP requests.
With DDoS protection capability of hundreds of Gbps, Anti-DDoS Pro can easily deal with DDoS attacks and ensure the availability of your business.
WAF can block web attacks in real time to ensure the security of your business data and information.

Directions

Step 1. Configure WAF

1. Log in to the WAF Console and select **Instance Management** -> **Instance List** on the left sidebar to enter the instance list.
2. On the page, select an instance, and click **Domain Name Connection** to add a domain name.



img

3. On the domain name connection page, click **Add Domain Name** and configure the following parameters as needed:

Domain Name Configuration

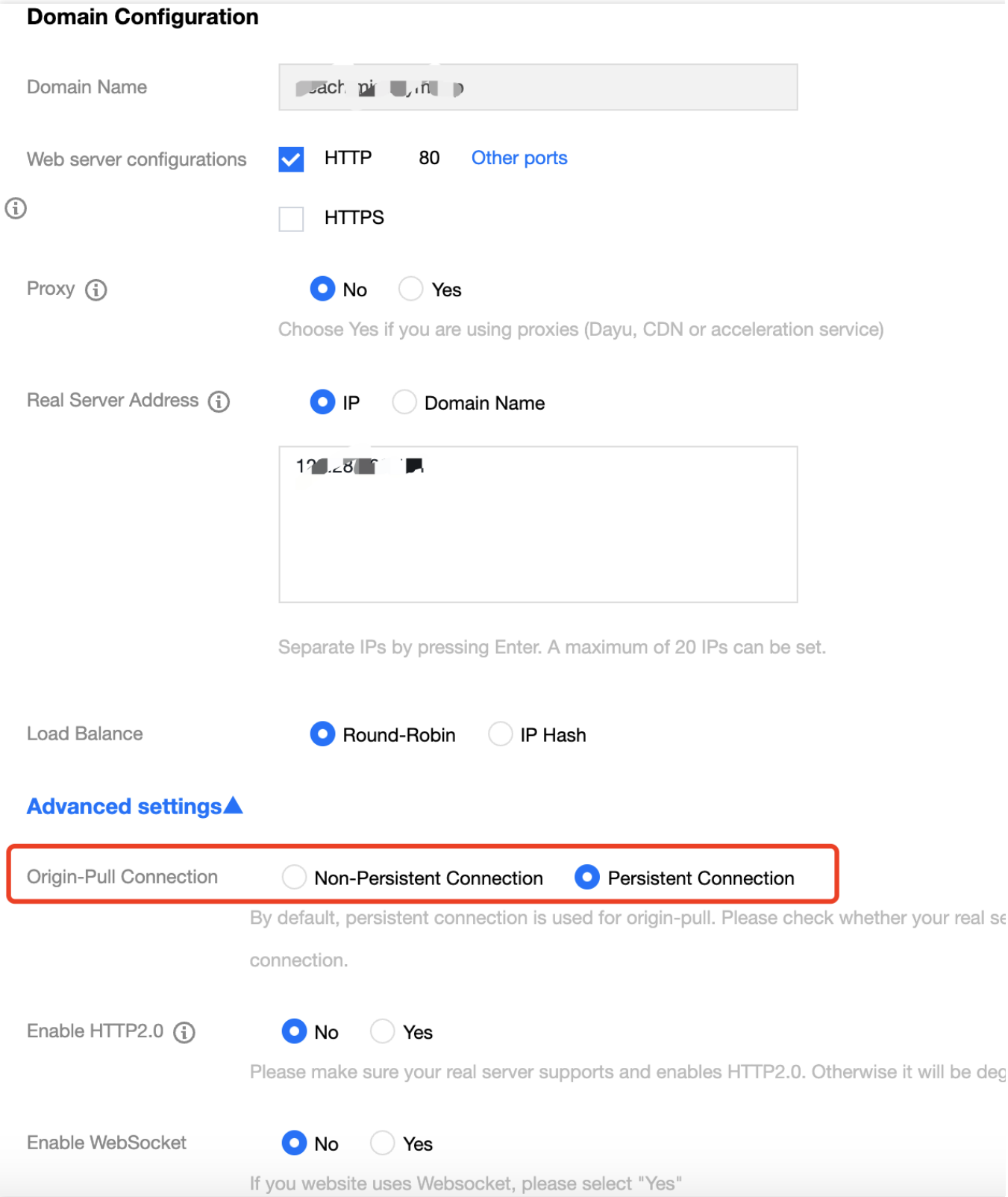
Domain Name: enter the domain name to be protected.
Web Server Configurations: select a protocol type and port as needed.
Enable HTTP 2.0: select according to your situation.
Server Port: select according to your situation.

Origin Server Address: enter the real IP address of the origin server of the website to be protected, which is the public IP of the origin server.

Other Configurations

Proxy: select "No". If WAF works with Anti-DDoS Advanced, select "Yes".

Enable WebSocket and Load Balancer: select according to your situation.



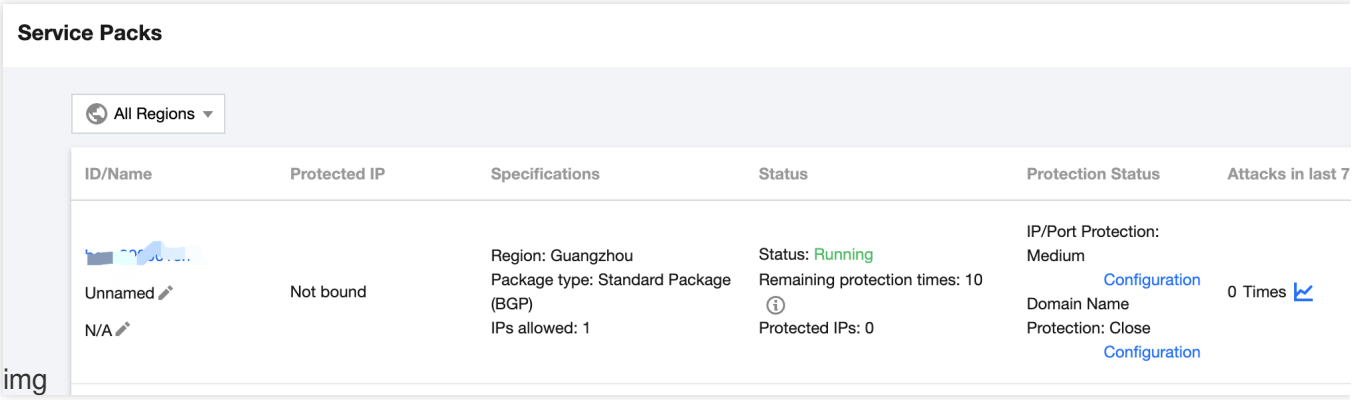
Note:

If the real server has multiple intermediate IPs, choose a load balancing strategy as needed. The round-robin strategy will distribute requests of the source IP across real servers in order, while the IP hash strategy will forward requests of the source IP to the same real server. Round-robin is used by default.

4. After the configuration, click **Save**.

Step 2. Configure Anti-DDoS Pro

1. Log in to Anti-DDoS Pro Console and select **Anti-DDoS Pro** > **Service Packs** on the left sidebar.
2. Select a region of the target Anti-DDoS Pro instance and click **Protected Resource** on the right of the instance.

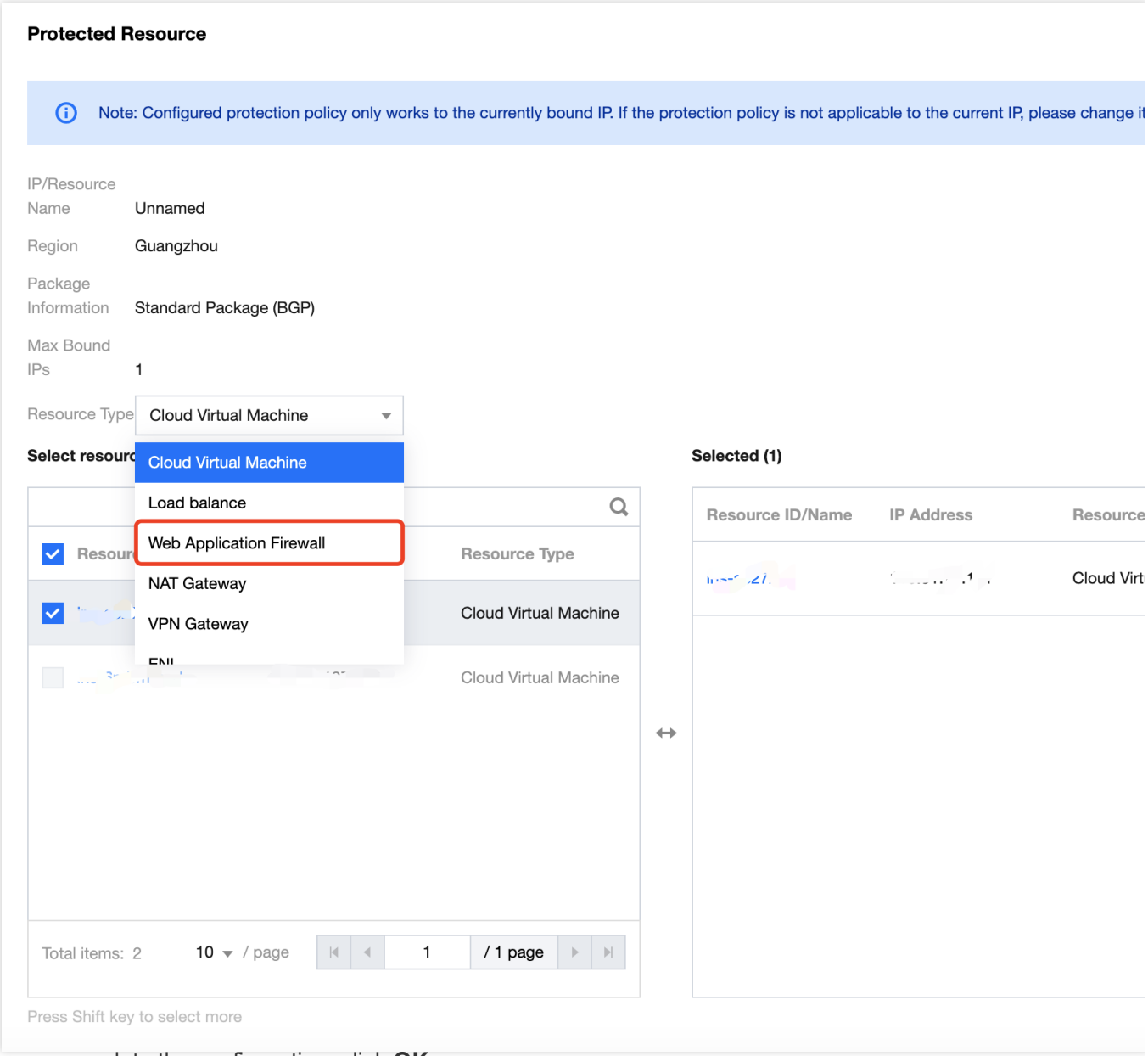


img

3. Select "Web Application Firewall" as the resource type, and set the IP address of the WAF instance.

Note:

For a CLB WAF instance, select "Load Balancing" as the resource type, and set the public IP address of the instance.



4. After you complete the configuration, click **OK**.

Applying for and Using Free HTTPS Certificates

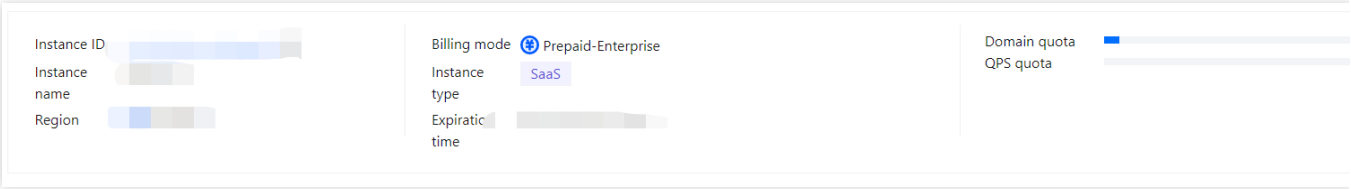
Last updated：2023-12-29 14:54:19

Prerequisites

WAF supports the configuration and protection of HTTPS access to domain names. If your website has not been altered for the HTTPS protocol, you can apply for a DV certificate free of charge in the SSL Certificate Service console. After your application is approved, you can associate the certificate in the WAF console and then easily implement access and client connection to the entire website over HTTPS without modifying the real server.

Associating HTTPS Certificate

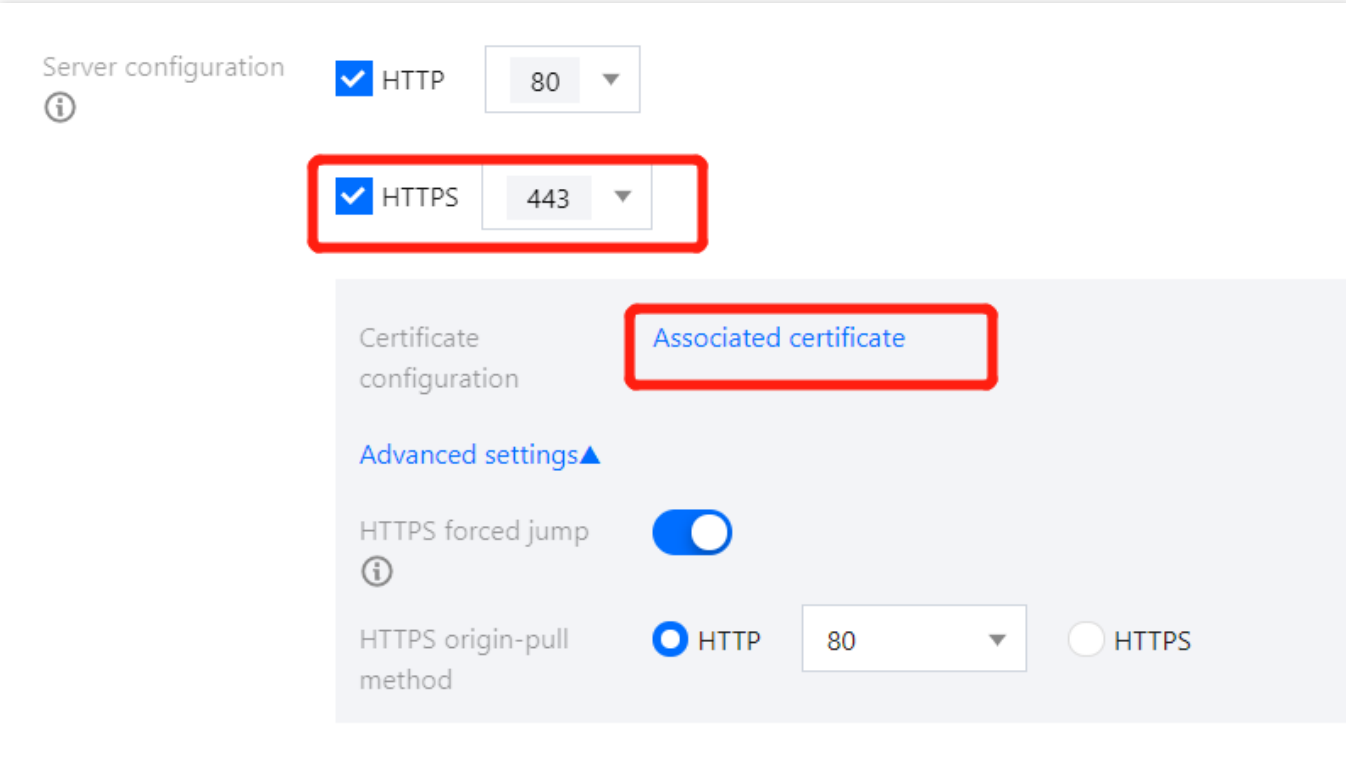
1. Log in to the WAF console and select **Instance management** > **Instance list** on the left sidebar.
2. On the **Instance list** page, select the target instance and click **Domain name connection**.



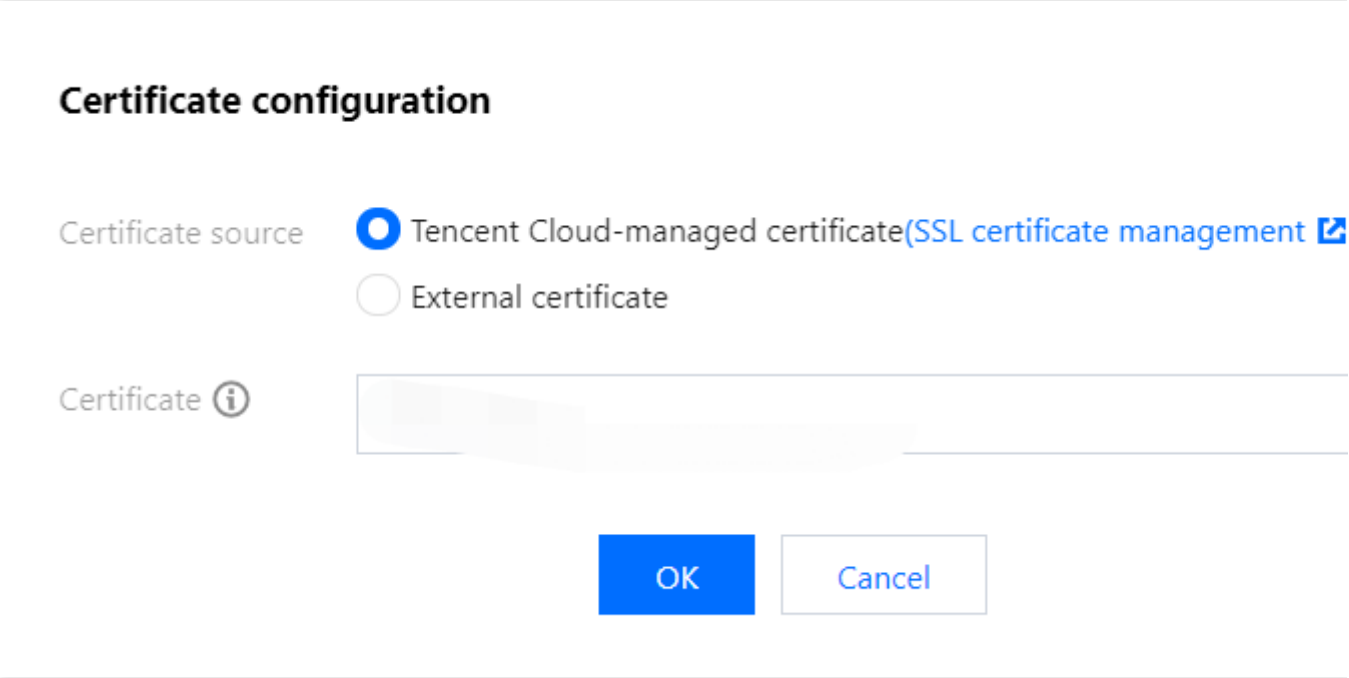
3. On the **Domain name connection** page, click **Add domain name**.
4. In **Server configuration** of the domain name configuration, select **HTTPS**. In **Certificate configuration**, click **Associated certificate**.

Note:

The certificate format should be PEM and the content should be text.



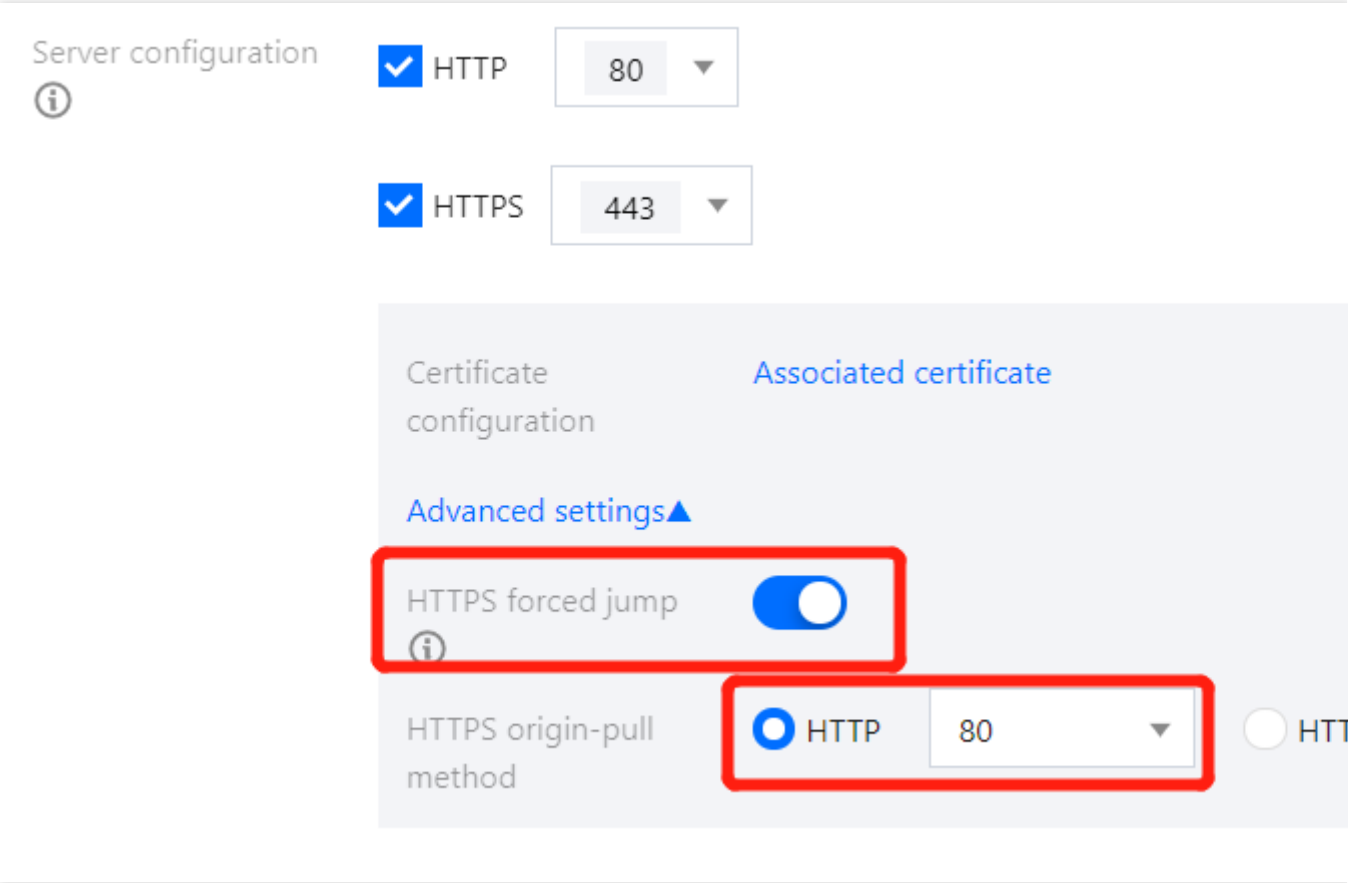
5. Select **Tencent Cloud-managed certificate** as the **Certificate source**. Then, WAF will automatically associate an available certificate of the domain name. After the configuration is completed, click **Save**.



6. Enable **HTTPS forced jump** and select the **HTTP** access protocol above. Select **HTTP** for **HTTPS origin-pull method** and set other parameters as needed; then, your website will support HTTPS access.

Note:

To enable **HTTPS forced jump**, you need to select both **HTTP** and **HTTPS** access protocols.



Obtaining Real Client IPs

Last updated：2023-12-29 14:54:31

Getting Real Client IP in WAF

WAF uses a reverse proxy to protect your website. When you access a WAF-protected domain name, a `X-Forwarded-For` record will be added to the HTTP header field to record your real IP, such as `X-Forwarded-For:user IP`. If the accessed domain name has proxies at multiple levels, WAF will record the IP of the proxy server just before WAF, for example:

Scenario 1: User > WAF > real server, with `X-Forwarded-For` recorded as `X-Forwarded-For:user's real IP`

Scenario 2: User > CDN > WAF > real server, with `X-Forwarded-For` recorded as `X-Forwarded-For:user's real IP,X-Forwarded-For:CDN origin-pull address`

Note:

In scenario 2, you need to select **Yes** for **Use proxy** when adding a domain name in WAF. After the proxy is connected, the client IP may be forged, but this will not be the case if Tencent Cloud CDN is used, as it will reset the `X-Forwarded-For` information and enter only the client IP it has obtained. (If a proxy is used, attackers can launch attacks only if they can send requests directly to the WAF VIP address. When the proxy is connected, the WAF VIP address cannot be detected by users. Be sure to keep the WAF VIP confidential.)

For more information on CLB WAF connection, see Obtaining Real Client IPs over IPv4 CLBs.

Below are commonly used `X-Forwarded-For` configuration schemes for application servers:

IIS 7 Configuration Scheme

Apache Configuration Scheme

NGINX Configuration Scheme

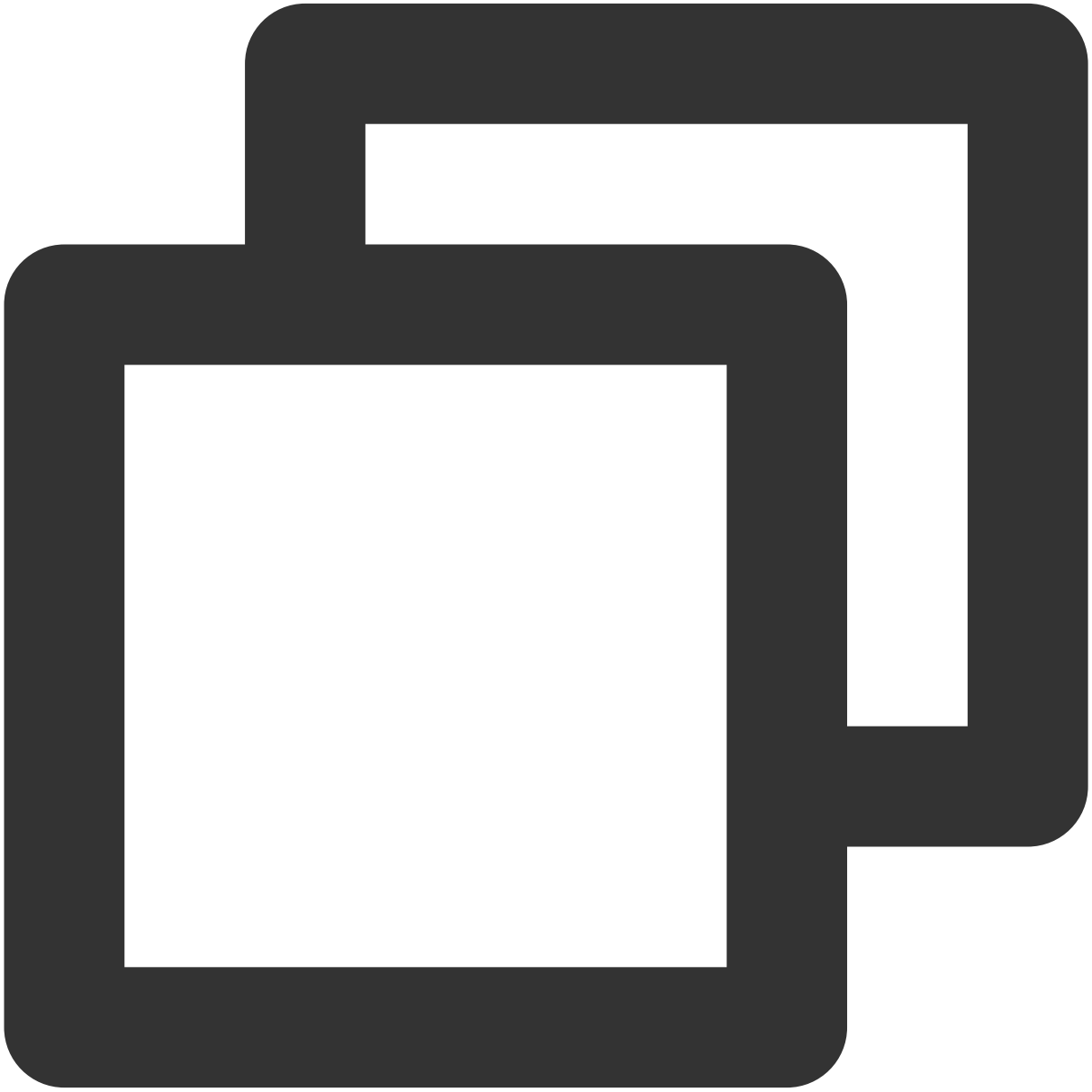
IIS 7 Configuration Scheme

1. Download and install the F5XForwardedFor plugin module, copy `F5XFFHttpModule.dll` and `F5XFFHttpModule.ini` in the `x86\\Release` or `x64\\Release` directory based on your server OS to a certain directory (such as `C:\\F5XForwardedFor`), and make sure that the IIS process has read permission to this directory.
2. Select **IIS Server** and double-click **Modules**.
3. Click **Configure Native Modules**.
4. In the pop-up box, click **Register**.
5. Add the downloaded DLL files.

6. After adding the files, check them and click **OK**.
7. Add the above two DLL files in "ISAPI and CGI Restrictions" and set the restrictions to "Allow".
8. Restart the IIS server for the configuration to take effect.

Apache Configuration Scheme

1. Install the Apache "mod_rpaf" module using the following commands:



```
wget http://stderr.net/apache/rpaf/download/mod_rpaf-0.6.tar.gz
```

```
tar zxvf mod_rpaf-0.6.tar.gz
cd mod_rpaf-0.6
/usr/bin/apxs -i -c -n mod_rpaf-2.0.so mod_rpaf-2.0.c
```

2. Modify the Apache configuration file `/etc/httpd/conf/httpd.conf` by adding the following to the end of the file:

<pre>

LoadModule rpaf_module modules/mod_rpaf-2.0.so

RPAFenable On

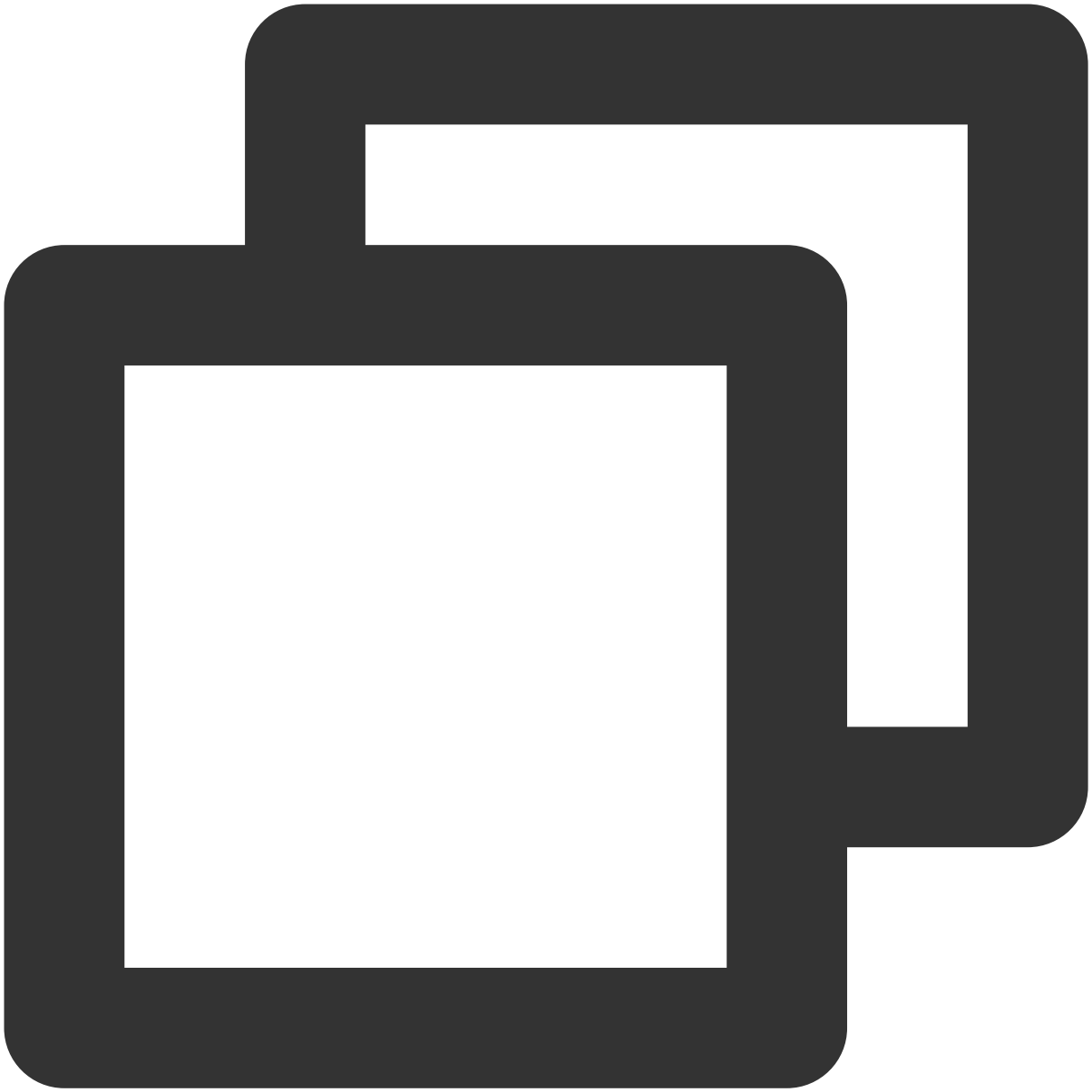
RPAFsethostname On

RPAFproxy_ips IP // The IP address is the origin-pull IP address of the WAF-protected domain name. You can view it in the protected domain name list in the WAF console or in the backend logs of the server. You only need to enter all the IP addresses that need to be viewed.

RPAFheader X-Forwarded-For

</pr>

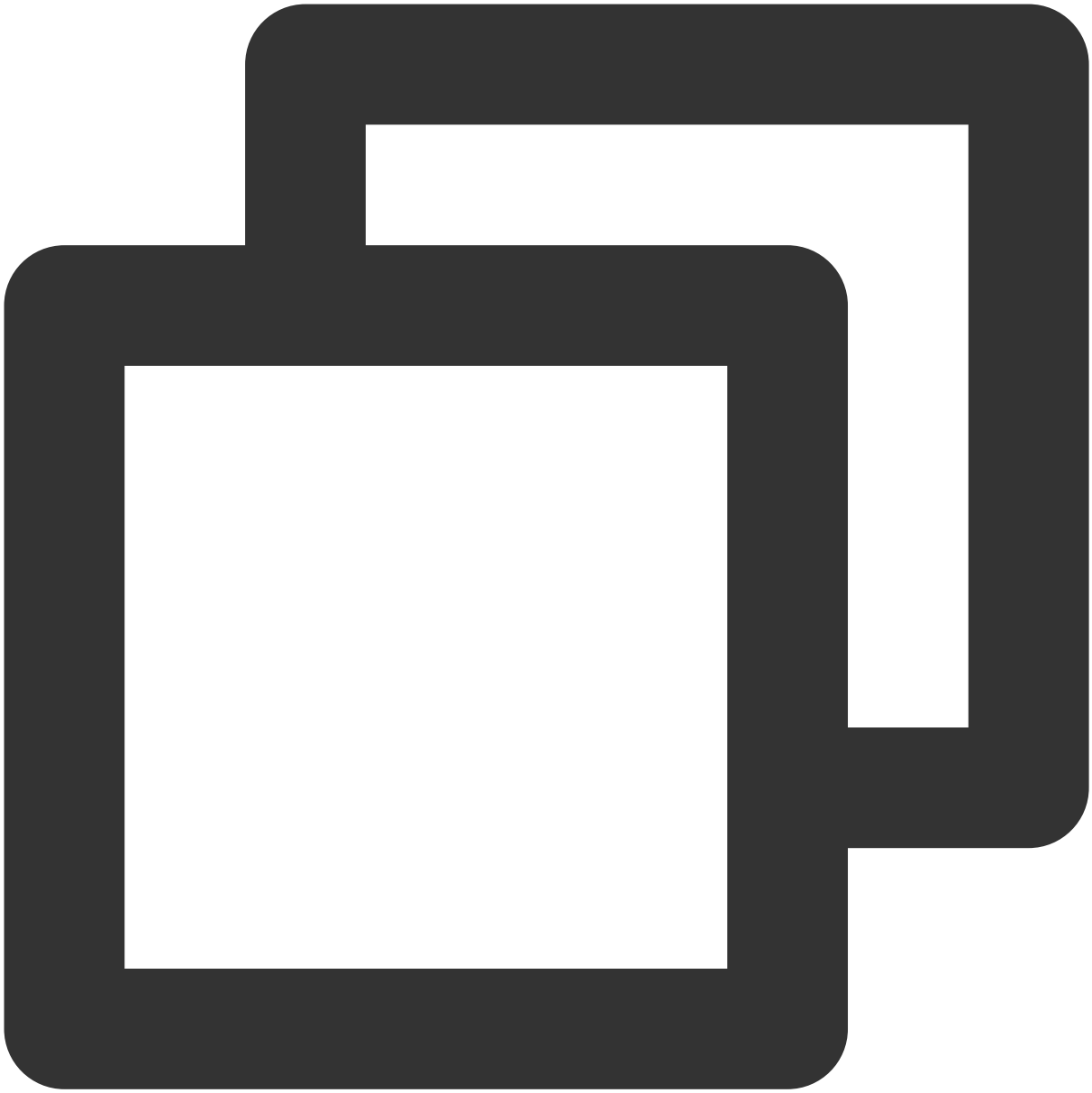
3. After adding the above content, restart Apache.



```
/usr/sbin/apachectl restart
```

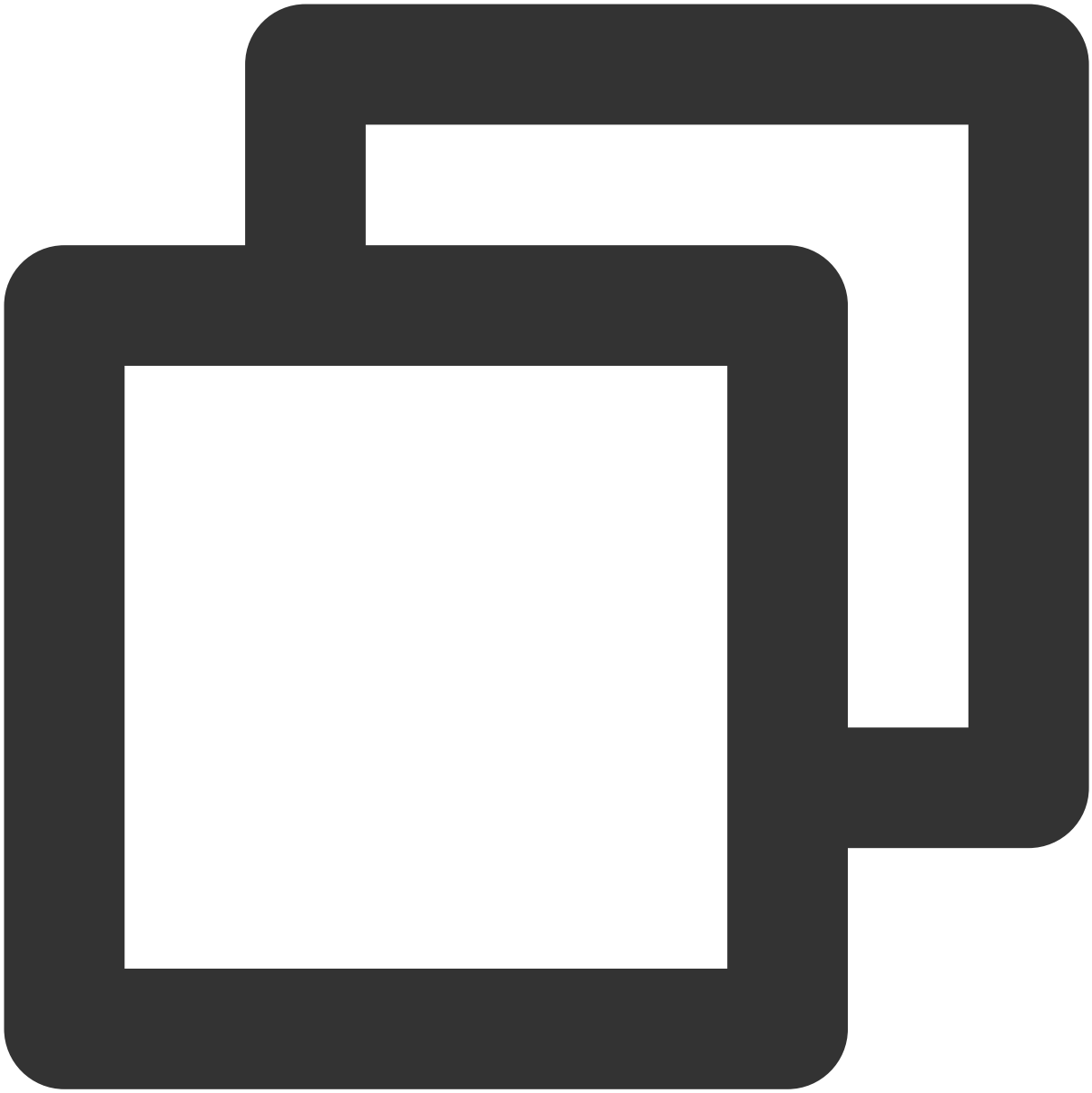
NGINX Configuration Scheme

1. You can use `http_realip_module` to get the real client IP when NGINX is used as the server. However, this module is not installed in NGINX by default, so you need to recompile NGINX to add `--with-http_realip_module`. The code is as follows:



```
wget  http://nginx.org/download/nginx-1.14.0.tar.gz
tar  zxvf nginx-1.14.0.tar.gz
cd nginx-1.14.0
./configure --user=www --group=www --with-http_stub_status_module --without-http-ca
make
make install
```

2. Modify the `nginx.conf` file.



```
vi /etc/nginx/nginx.conf
```

Modify the content in red as shown below:

<pre>

fastcgi connect_timeout 300;

fastcgi send_timeout 300;

fastcgi read_timeout 300;

fastcgi buffer_size 64k;

fastcgi buffers 4 64k;

fastcgi busy_buffers_size 128k;

fastcgi temp_file_write_size 128k;

set_real_ip_from IP; // The IP address is the origin-pull IP address of the WAF-protected domain name. You can view it in the connected domain name list in the WAF console.

real_ip_header X-Forwarded-For;

</pre>

3. Restart NGINX.

<pre>

service nginx restart

</pre>

Replacing Certificate

Last updated：2023-12-29 14:54:43

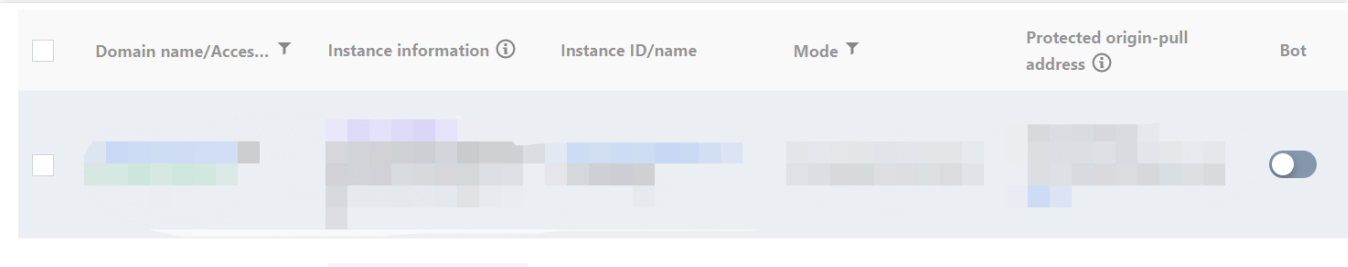
Overview

When users visit your website with an expired certificate, there will be a warning sign displayed; if an API has been called by your domain name, an error will be reported. To avoid business interruption, update your certificate on the console in a timely manner.

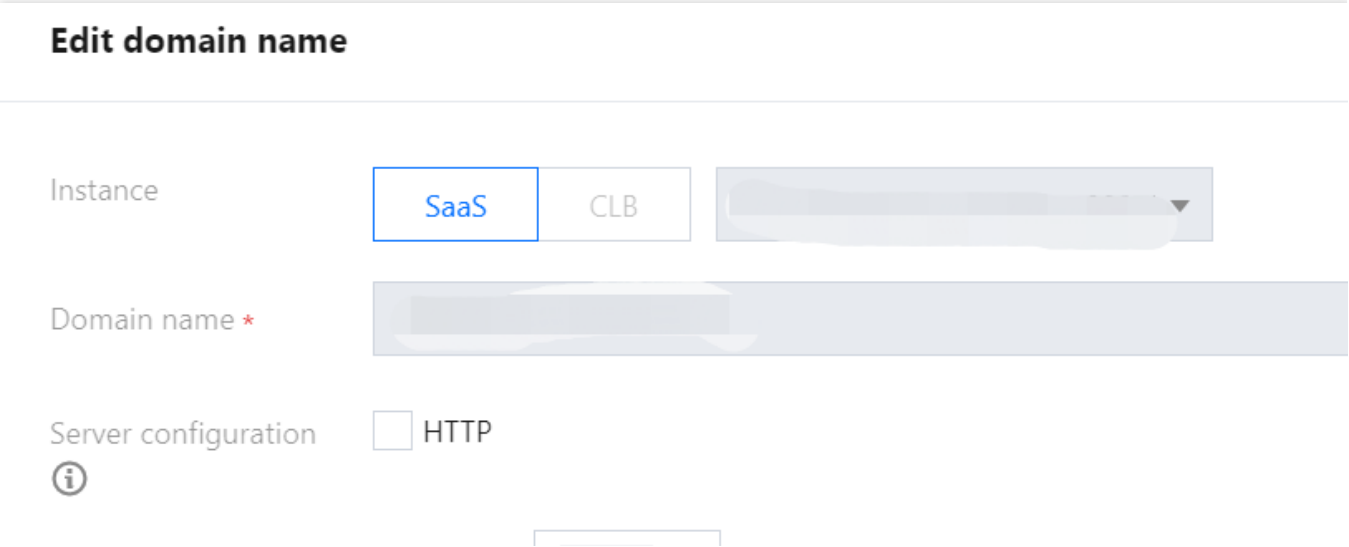
Directions

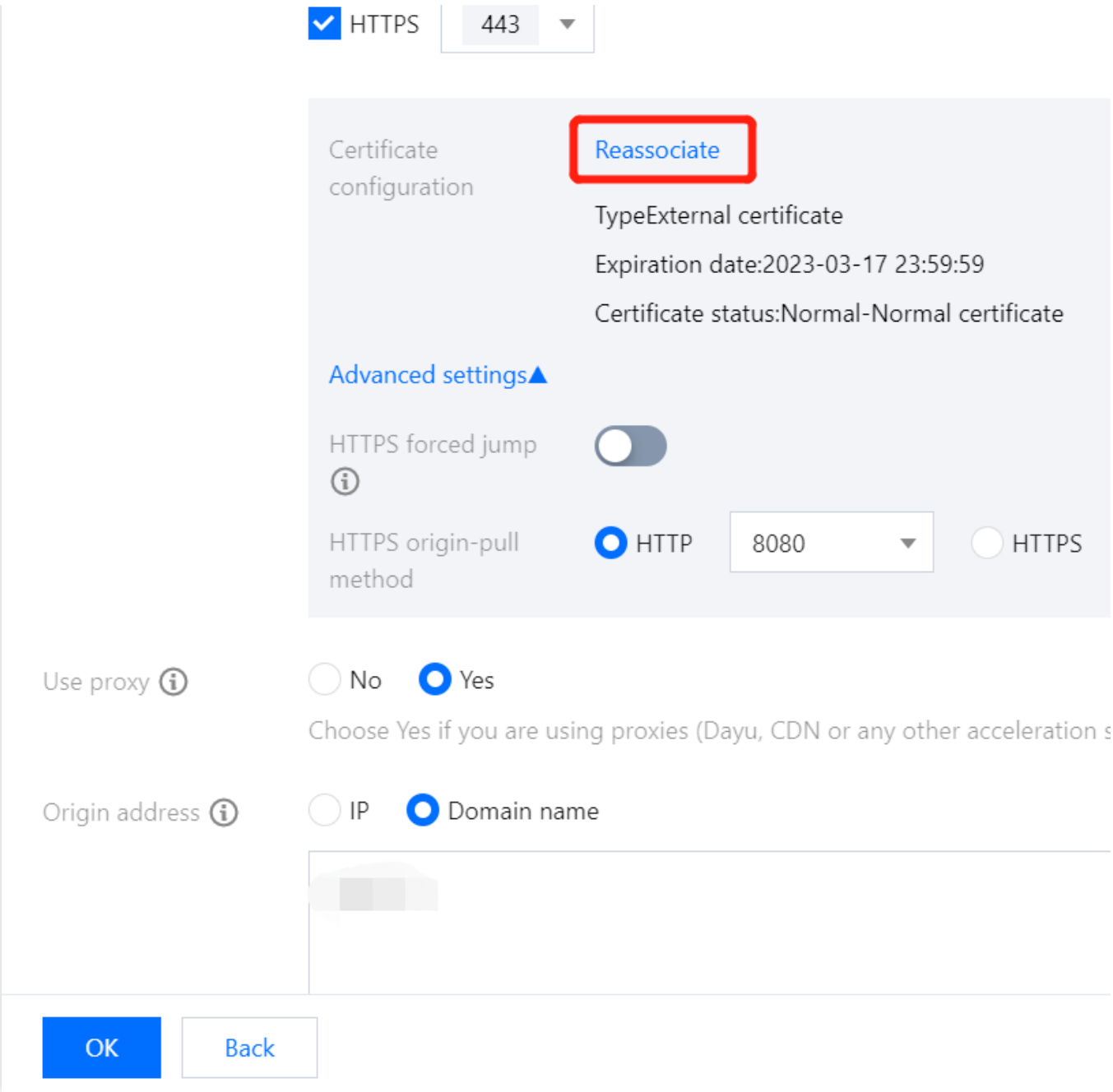
Example 1: External certificate

1. Log in to the WAF console and select **Asset center** > **Domain name list** on the left sidebar.
2. On the **Domain name list** page, select the target domain name and click **Edit**.

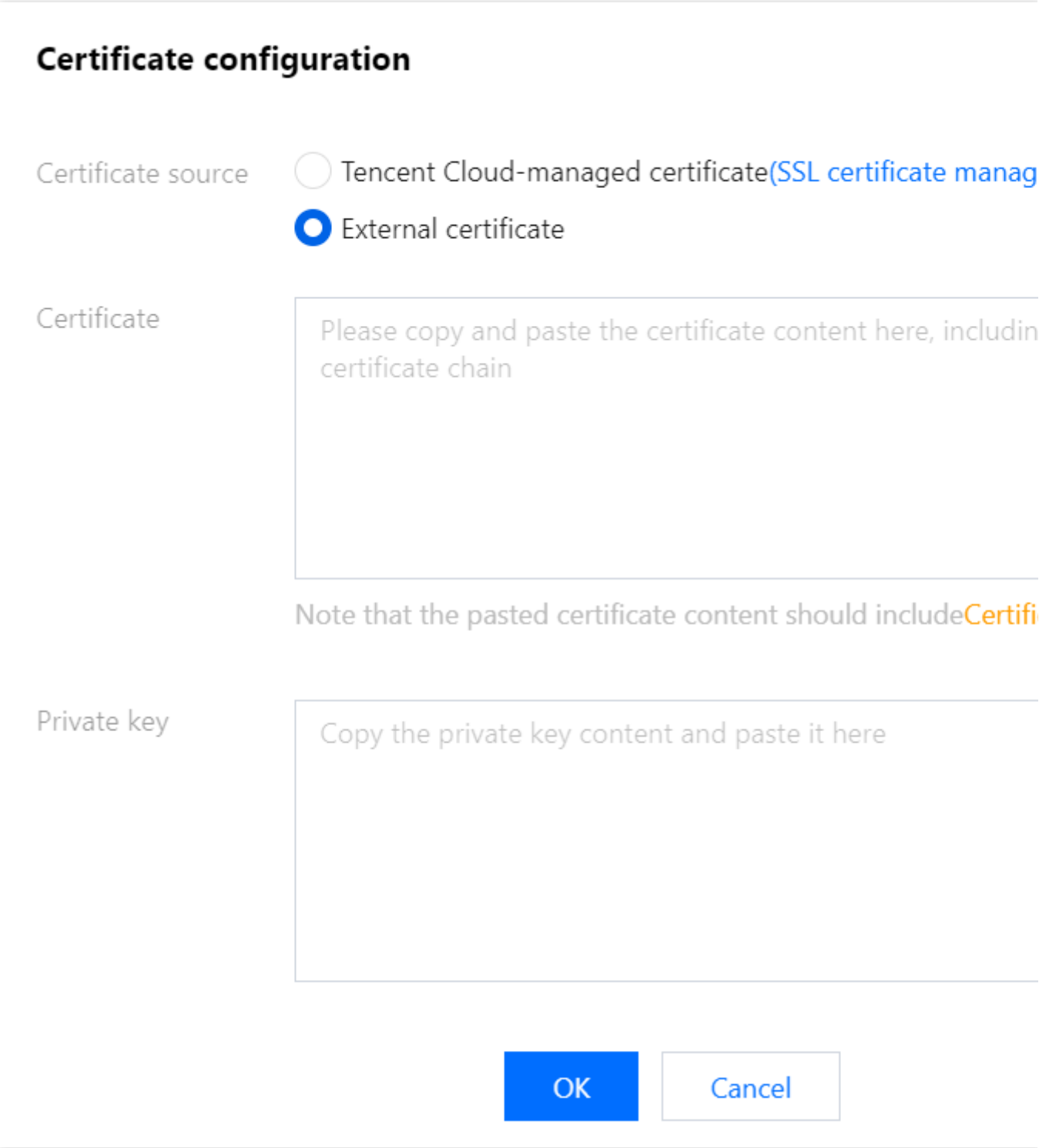


3. On the **Edit domain name** page, click **Reassociate** in **Server configuration** to pop up the **Certificate configuration** window.



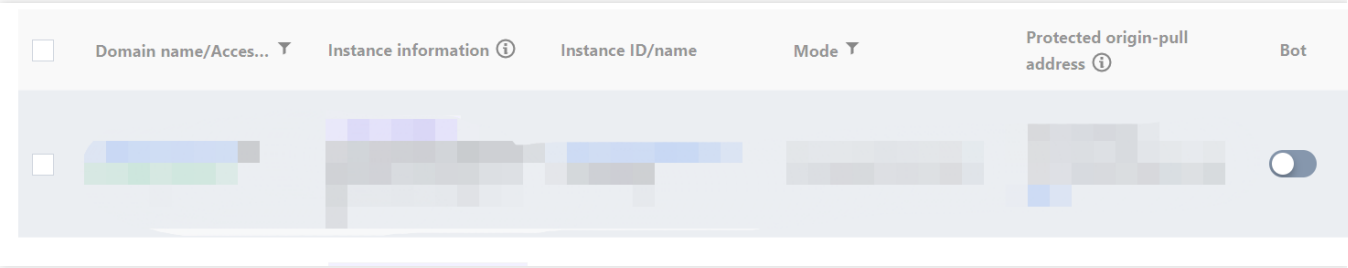


4. In the **Certificate configuration** pop-up window, select **External certificate** for **Certificate source**, enter the certificate and private key, and click **OK**.

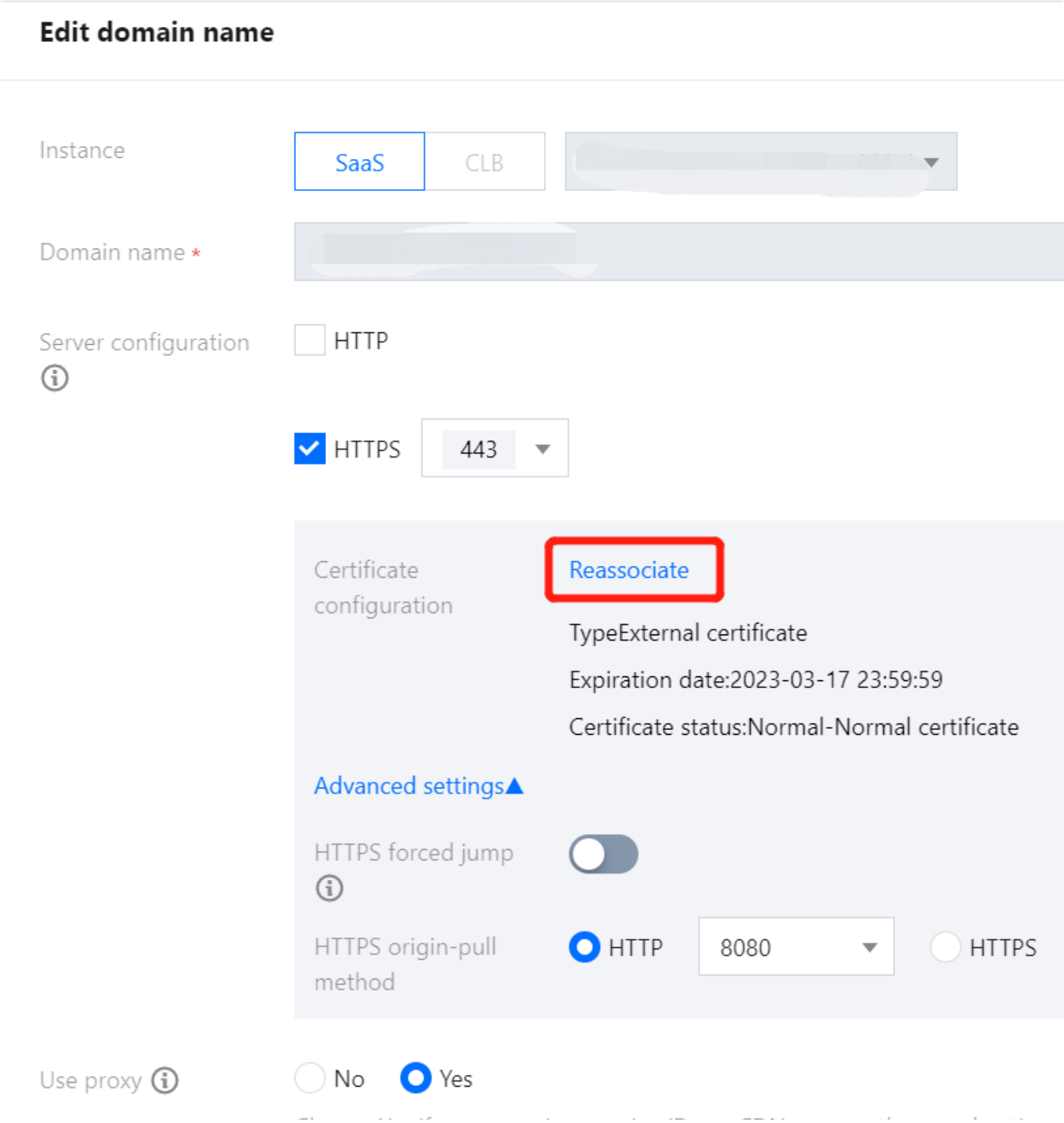


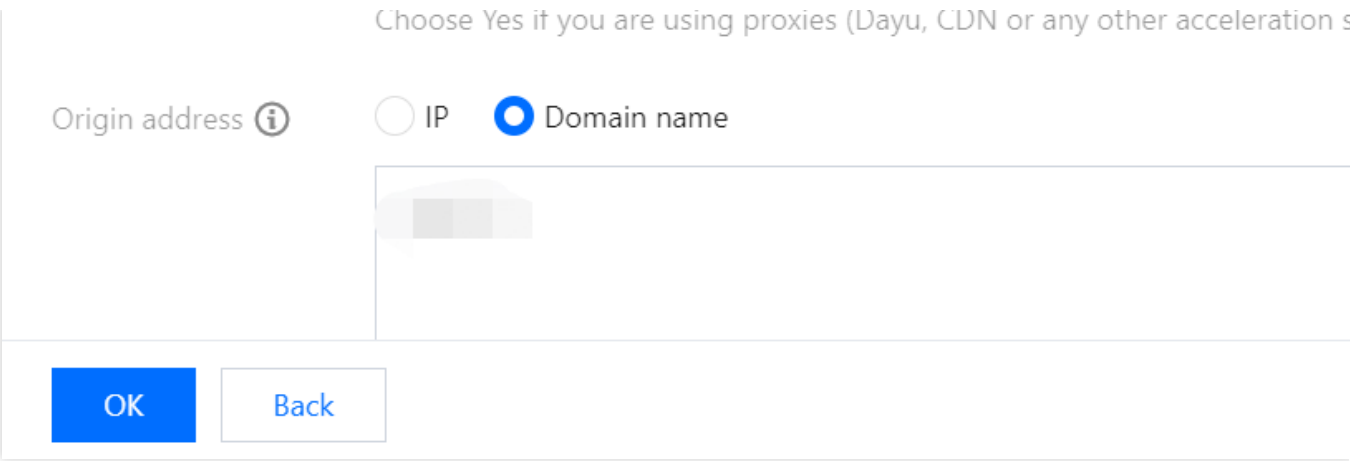
Example 2: Tencent Cloud-managed certificate

1. On the Domain name list page, select the target domain name and click **Edit**.



2. On the **Edit domain name** page, click **Reassociate** in **Server configuration** to pop up the **Certificate configuration** window.

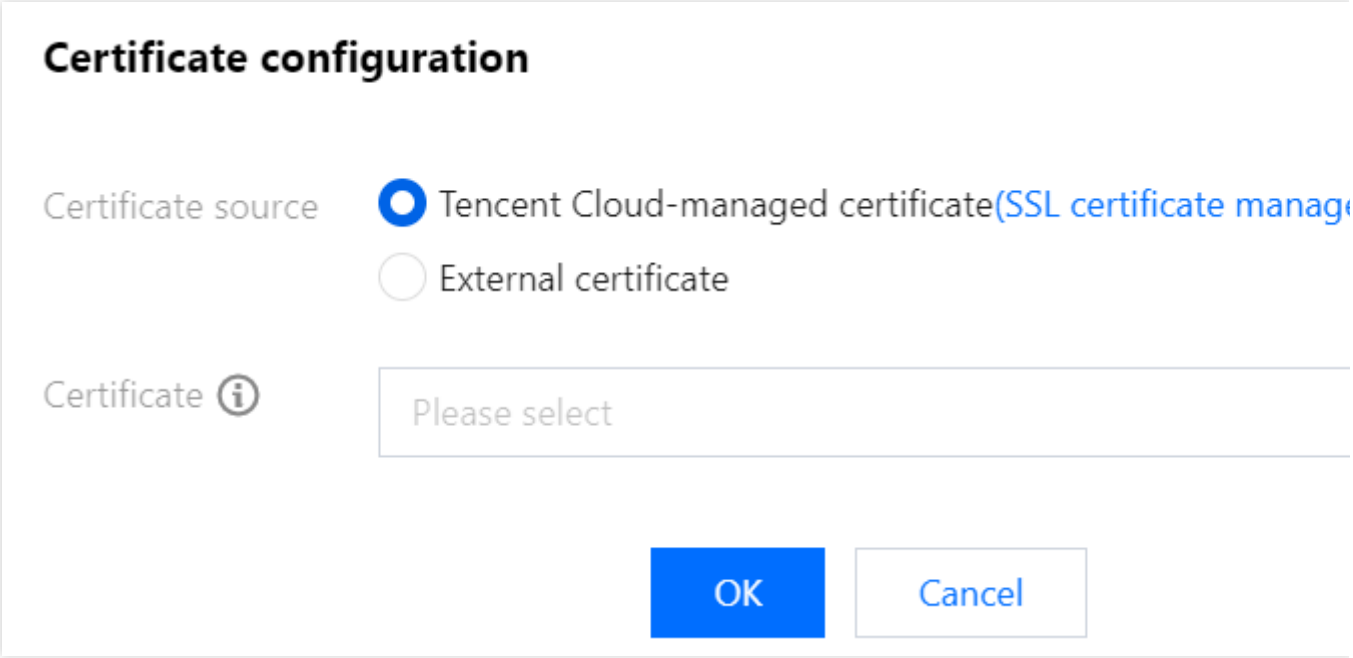




3. In the **Certificate configuration** pop-up window, select **Tencent Cloud-managed certificate** for **Certificate source** and click **OK**.

Note:

This method only applies to certificates that have been uploaded to SSL Certificate Service.



Certificate Validity Check

You can check the effective and expiration dates of the certificate by accessing the domain name via a browser. If the certificate does not take effect, contact us for help.

Protection Configuration
Setting CC Protection

Last updated：2023-12-29 14:55:00

This document describes how to configure CC protection in the WAF console.

Overview

CC protection enables access protection for specified URLs, which supports emergency CC protection and custom CC protection policies.

Note:

Emergency CC protection and custom CC rules cannot be enabled at the same time.

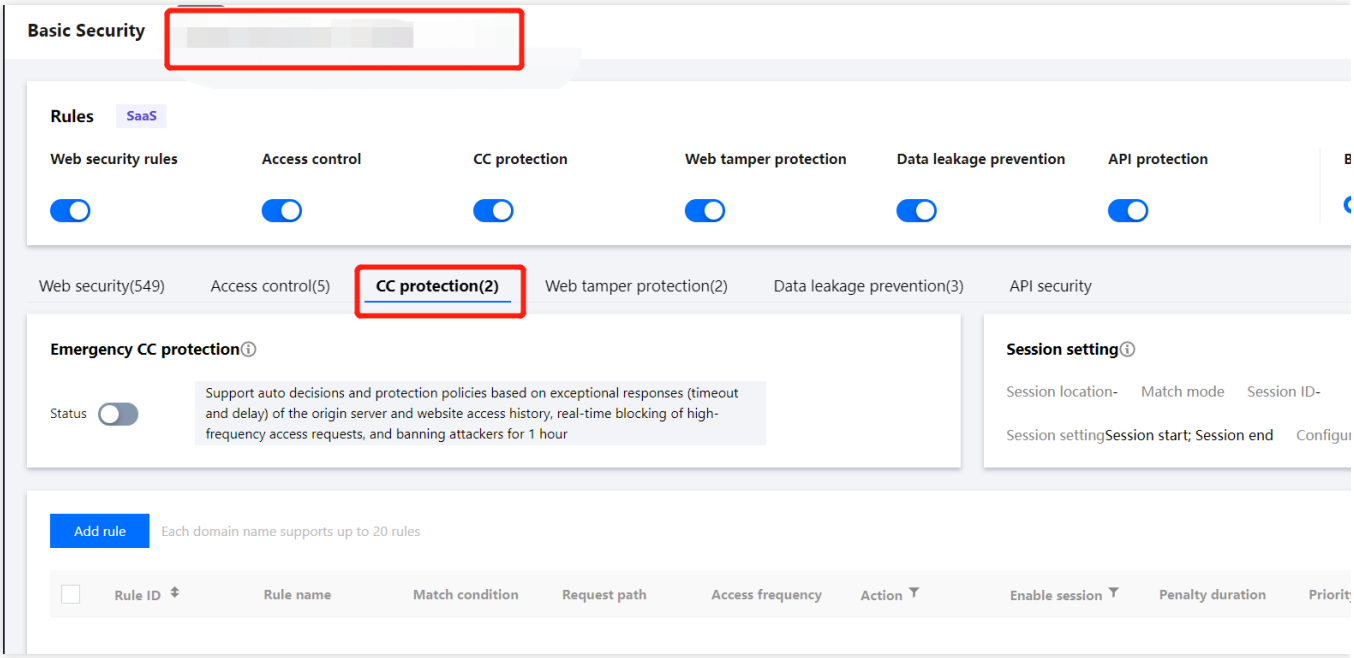
Directions

Example 1: Emergency CC protection settings

Note:

Emergency CC protection is disabled by default. Before enabling it, make sure that the custom CC rule feature is disabled.

1. Log in to the WAF console and select **Basic security** on the left sidebar.
2. On the **Basic security** page, select the target domain name in the top-left corner and click **CC protection**.



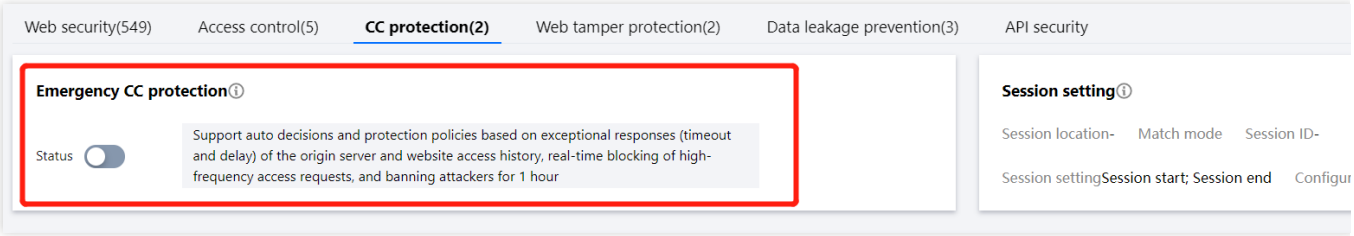
3. In the emergency CC protection module, click

on the right of the status and confirm the operation to enable emergency CC protection.

Note:

After emergency CC protection is enabled, if a website is under massive CC attacks (with a website QPS of 1000 or above), the protection will be automatically triggered. If there are no specific protection paths, we recommend enabling emergency CC protection. As there may be some false alarms, you can enter the blocklist/allowlist in the console to add blocked IPs to the allowlist.

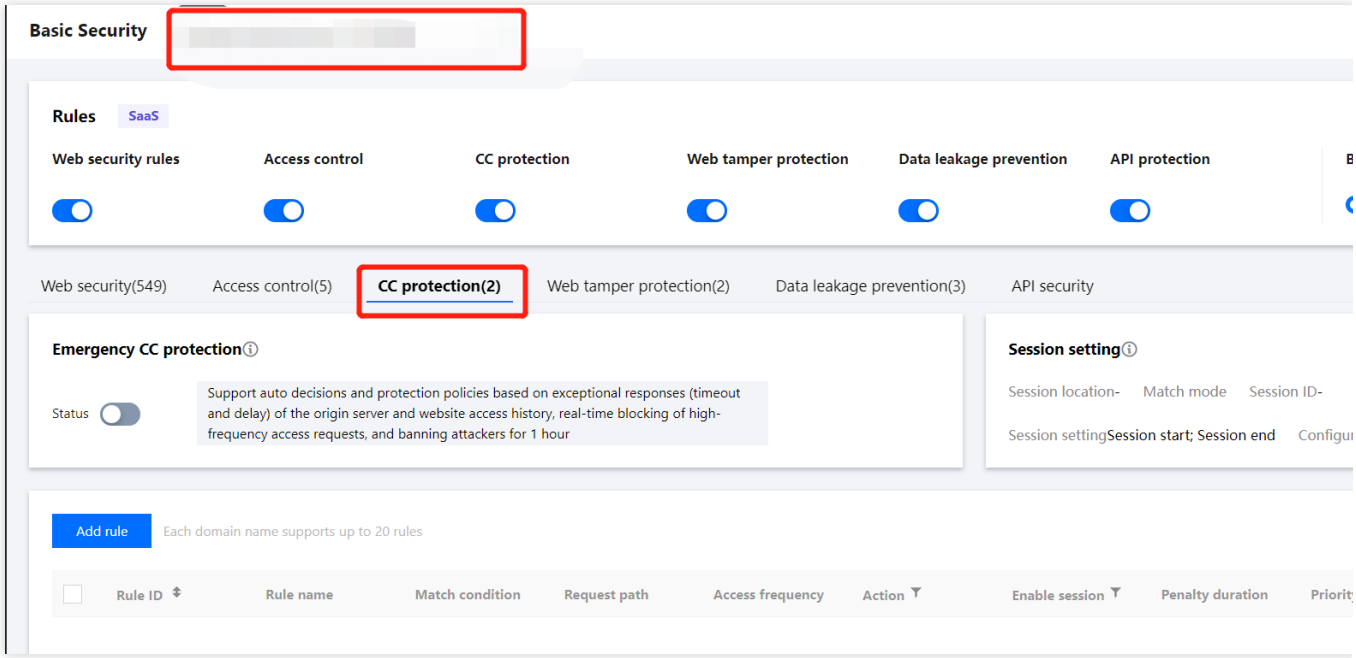
If there are specific protection paths, we recommend using custom CC rules.



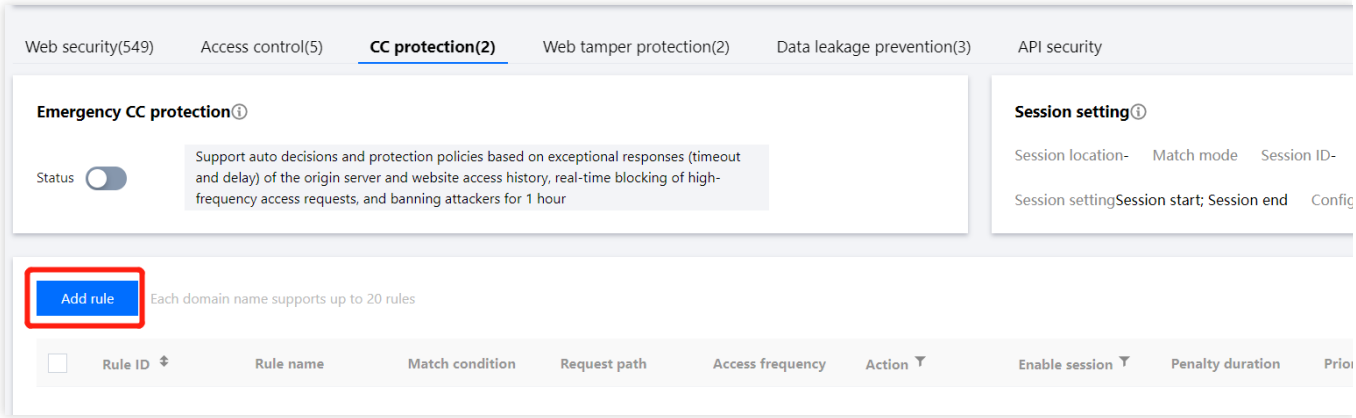
Example 2: Access source IP-based CC protection settings

An IP-based CC protection policy can be directly configured without setting SESSION.

1. Log in to the WAF console and select **Basic security** on the left sidebar.
2. On the **Basic security** page, select the target domain name in the top-left corner and click **CC protection**.



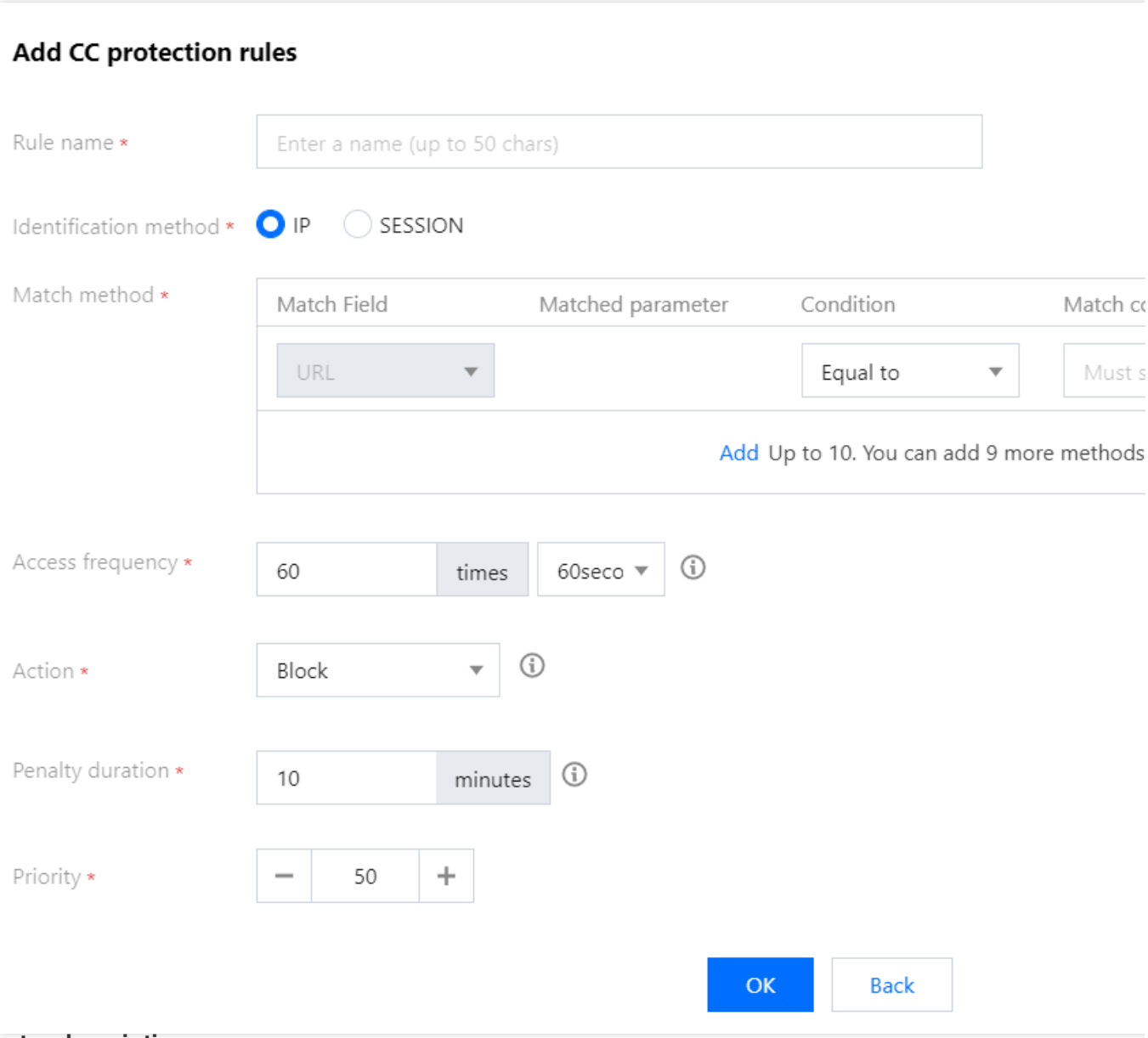
3. On the **CC protection** page, click **Add rule**.



4. In the **Add rule** pop-up window, enter the rule details.

Note:

If **IP** is selected as the recognition mode, after the rule is triggered for blocking, the IP will be blocked across the entire website (i.e., the IP will be blocked when accessing other URLs). But if **SESSION** is selected, blocking will not be global.



Parameter description:

Rule name: Custom name, which can contain up to 50 characters.

Identification method: **IP** or **SESSION**.

Match method: **Equal to**, **Prefix match**, or **Include**.

Advanced match: Filters access with GET and POST form parameters to control the frequency in a more refined manner and increase the hit rate.

Match field: Specifies the request method, which can be GET or POST.

Parameter name: Parameter name in a request field, which can contain up to 512 characters.

Parameter value: Parameter value in a request field, which can contain up to 512 characters.

Note: The three test entries for GET request are as follows: a=1&b=11, a=2&b=12, a=&b=13.

If the parameter name of a GET configuration is `a`, and the parameter value is `1`, then `1` will be hit.

If the parameter name of a GET configuration is `a`, the parameter value is `\\*`, then `1`, `2`, and `3` will be hit.

Access frequency: Set the access frequency based on your business, for which a value 3 to 10 times the common number of access requests is recommended. For example, if your website is accessed averagely 20 times per minute, you can configure the value to 60 to 200 times per minute or adjust it according to the attack severity.

Action: **Observe**, **CAPTCHA**, or **Block**.

Penalty duration: One minute to one week.

Priority: Enter an integer between 1 to 100. A smaller integer indicates a higher action priority for a rule. When the priority is the same, the later a rule is created, the higher its priority.

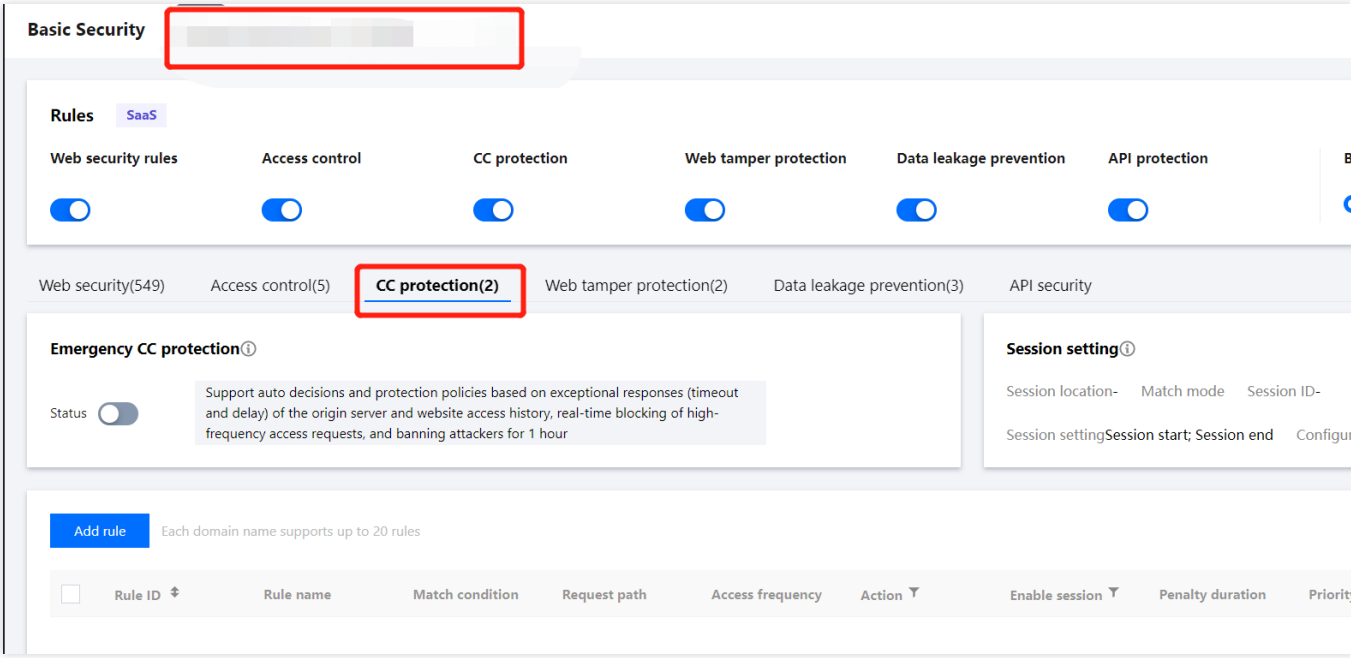
Example 3: Session-based CC protection settings

CC protection based on session access frequency effectively resolves false positive problems that may occur when the same IP egress is used by multiple users in office buildings, stores, supermarkets, and other public Wi-Fi networks.

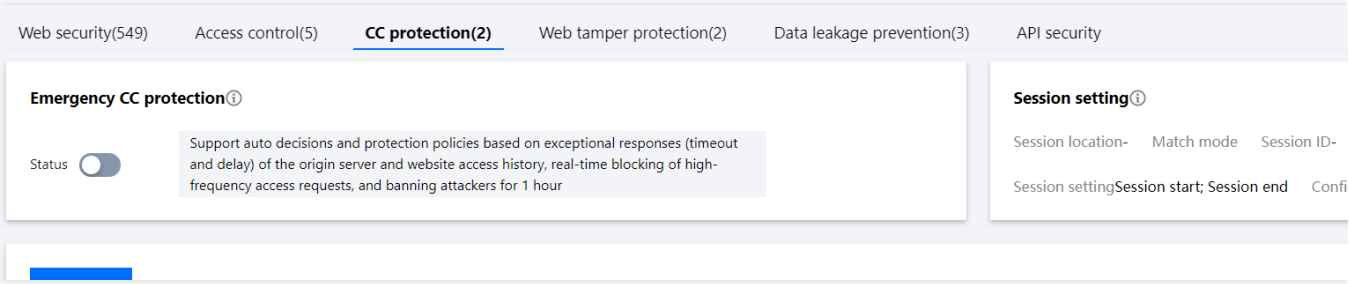
Note:

SESSION must be set before using the session-based CC protection policy. The step 1 to 4 are SESSION setting directions.

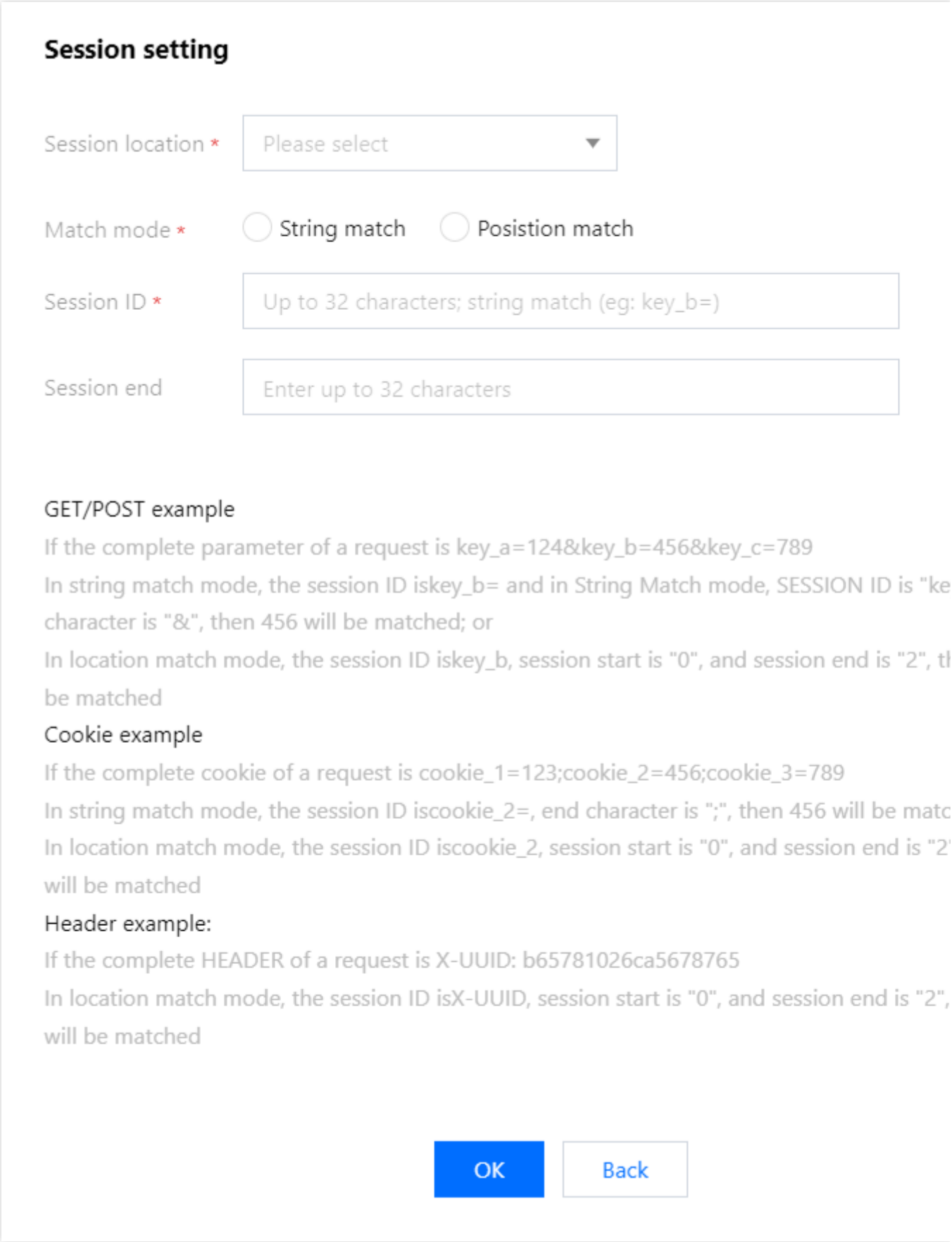
1. Log in to the WAF console and select **Basic security** on the left sidebar.
2. On the **Basic security** page, select the target domain name in the top-left corner and click **CC protection**.



3. In the **Session setting** module, click **Set** to set the session dimension information.



4. In the **Session setting** pop-up window, enter the required information. In this example, a cookie is used as the test object, whose **Session ID** is `security` , **Session start** is `0` , and **Session end** is `9` . After completing the settings, click **OK**.



Parameter description:

Session location: **COOKIE**, **GET**, or **POST**. Here, **GET** and **POST** are HTTP request content parameters rather than HTTP header information.

Match: **Location match** or **String match**.

Session ID: Session ID of up to 32 characters.

Session start: Location where string or location match starts. It is an integer between 0 and 2048.

Session end: Location where string or location match ends. It is an integer between 1 and 2048 and can contain up to 128 characters.

GET/POST example: Assume that the complete parameter content in a request is `key_a = 124&key_b = 456&key_c = 789`, then:

In string match mode, if the session ID is `key_b =`, and the end character is `&`, then the matched content will be `456`.

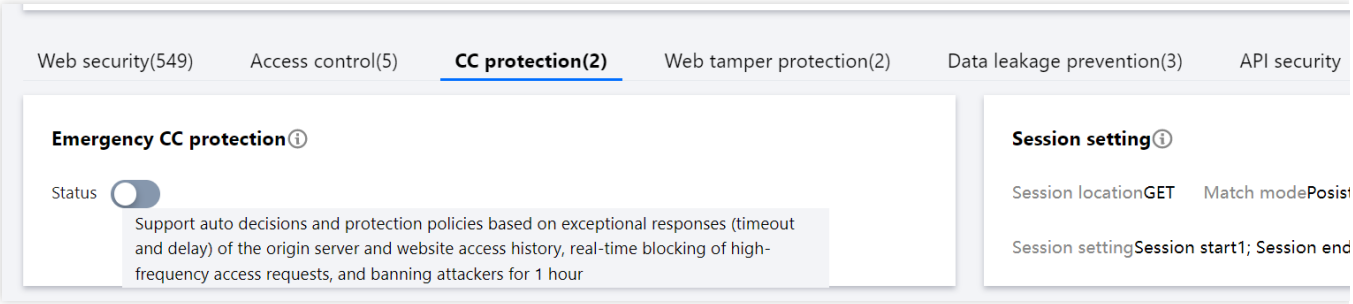
In location match mode, if the session ID is `key_b`, the session start is `0`, and the session end is `2`, then the matched content will be `456`.

Cookie example: Assume that the complete cookie content in a request is `cookie_1 = 123;cookie_2 = 456;cookie_3 = 789`, then:

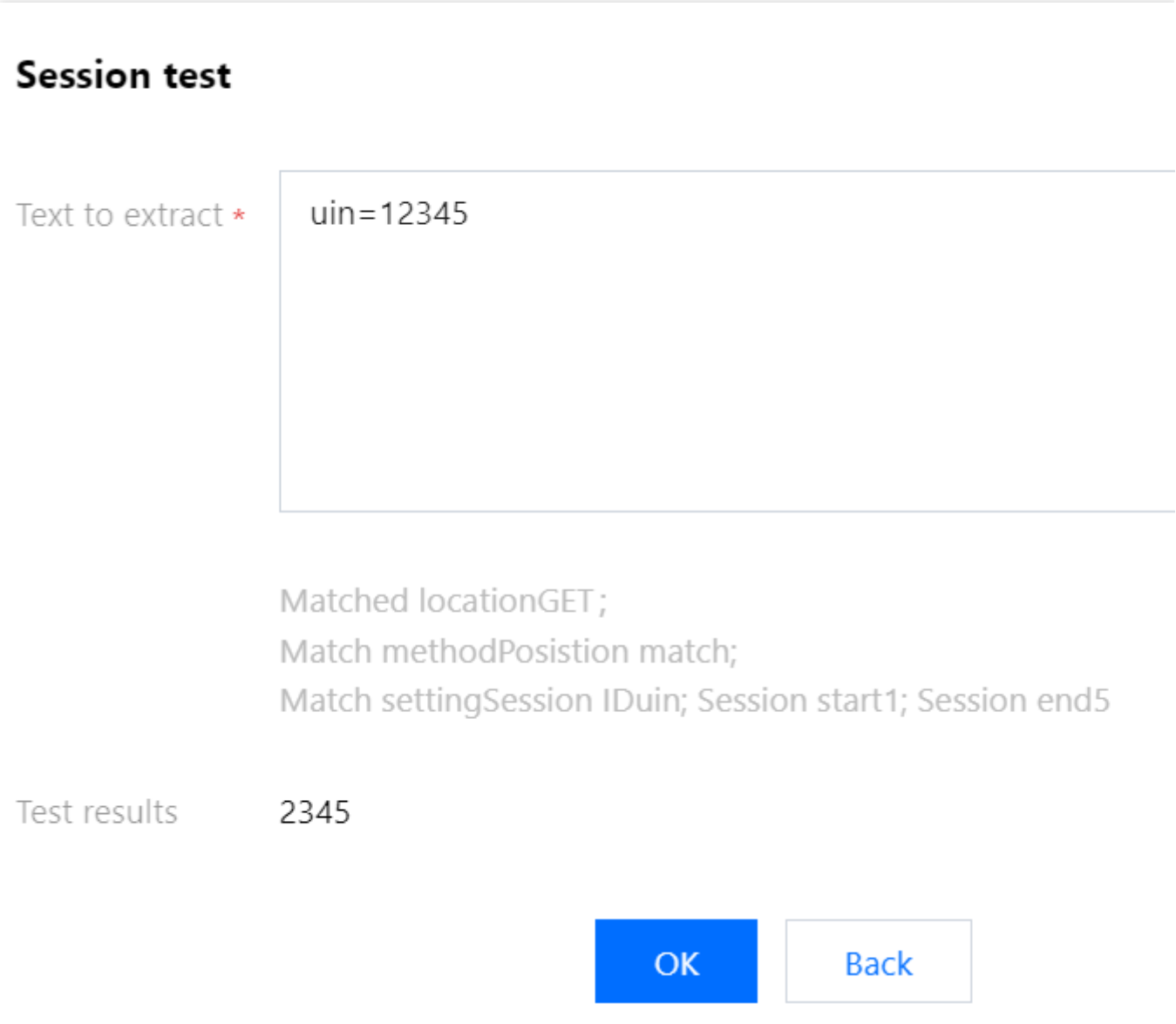
In string match mode, if the session ID is `cookie_2 =`, and the end character is `;`, then the matched content will be `456`.

In location match mode, if the session ID is `cookie_2`, the session start is `0`, and the session end is `2`, then the matched content will be `456`.

5. Click **Test** to test the session information.



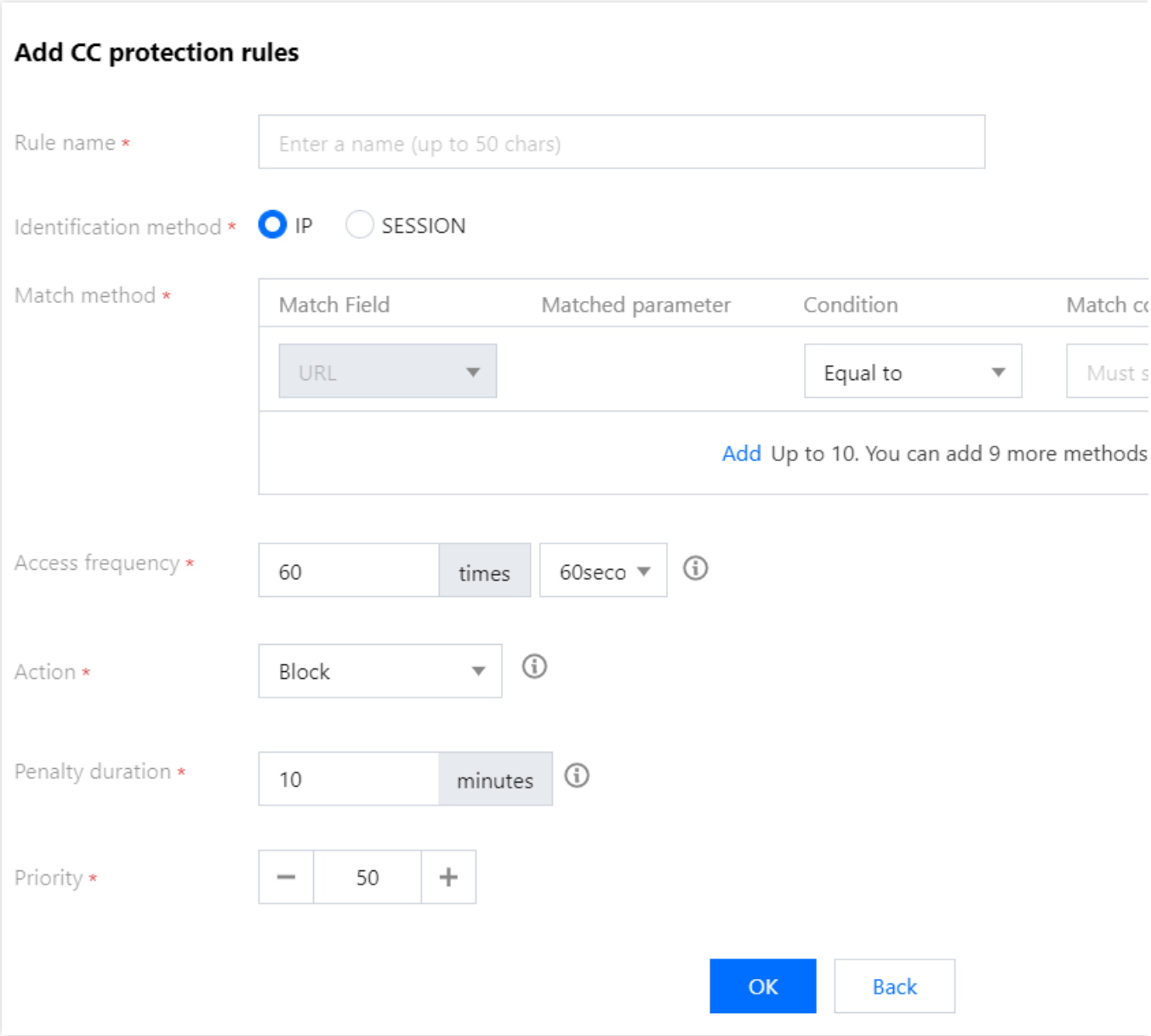
6. Go to the SESSION settings page and set the content to `security = 0123456789`. Then, WAF will use the 10 characters following `security` as the session ID. You can also delete or reconfigure the session information.



7. Set a session-based CC protection policy as instructed in Example 2, but select "SESSION" as the recognition mode.

Note:

If **GET** is selected as the session location in a rule, access with the same session information instead of the IP information will be blocked.



7. After the configuration is completed, the session-based CC protection policy will take effect.

Note:

If you use session-based CC protection, you cannot view IP blocking information in the IP blocking status section.

Connecting Frontend-Backend Separated Site to WAF CAPTCHA

Last updated：2023-12-29 14:55:13

You can connect WAF CAPTCHA to frontend-backend separated sites or app sites to dynamically send CAPTCHAs from such sites.
You can connect a frontend-backend separated site to the WAF CAPTCHA process to dynamically verify human operations for the site in various scenarios, including custom rule hit, CC attack protection, and bot traffic management. Both iOS and Android apps are connected through web frontend HTML5.

Prerequisites

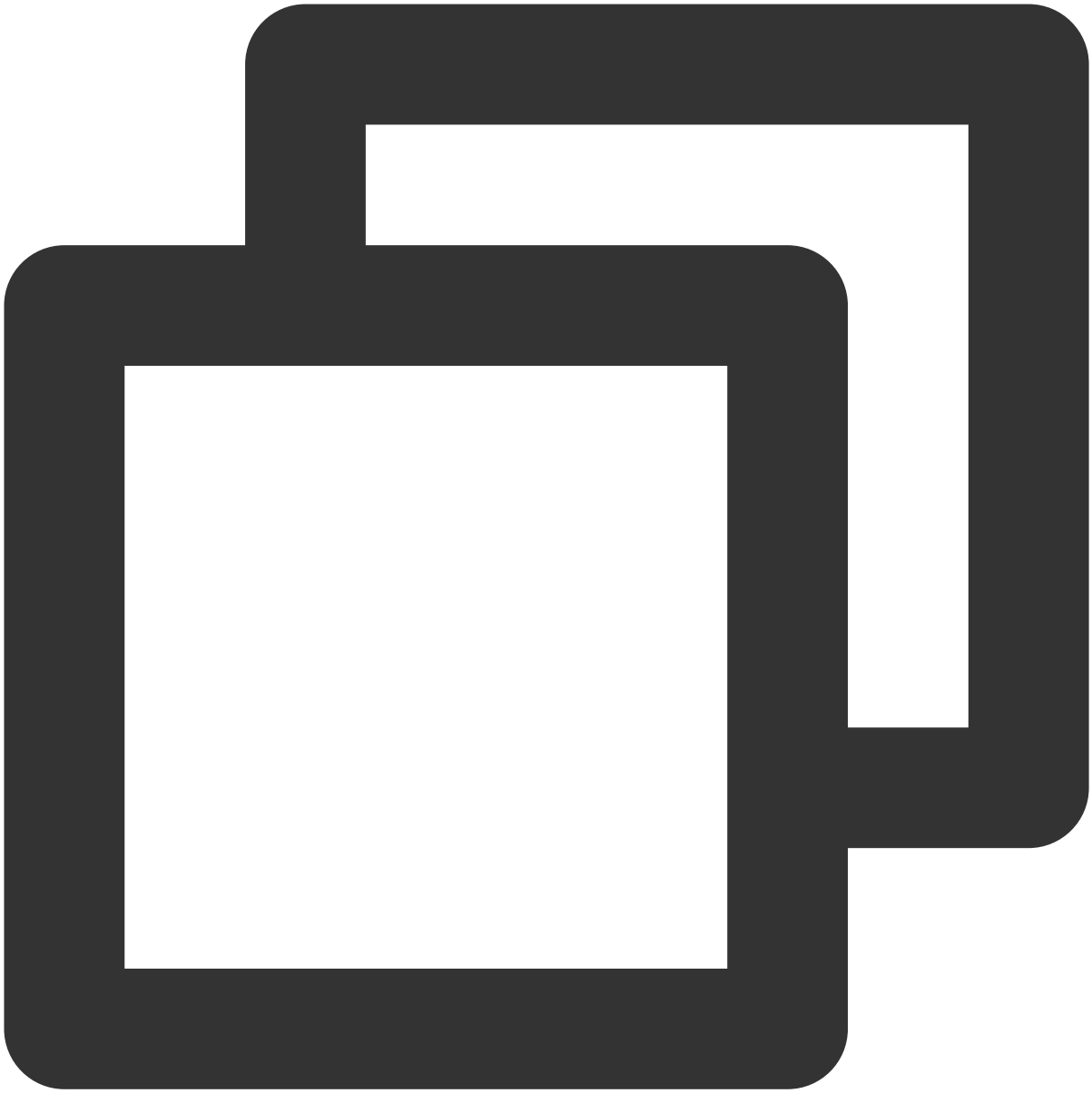
You have purchased WAF (Premium or higher) and connected to it.

How to Detect

This feature dynamically checks whether the packets returned from the server contain the CAPTCHA fields delivered by WAF, and if so, it will render the CAPTCHA at the top floating layer to connect the frontend-backend separated site or app to WAF CAPTCHA.

Directions

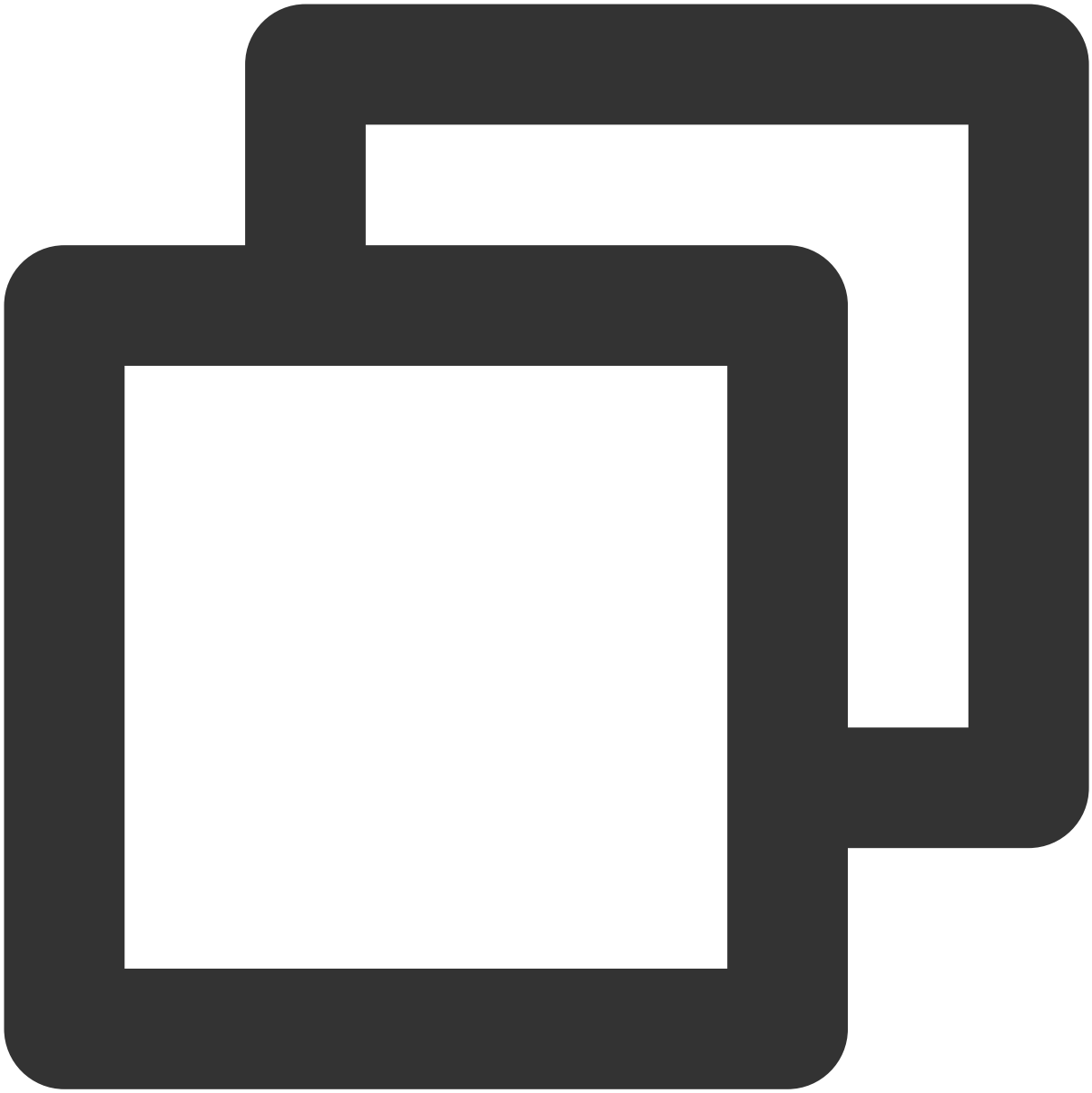
Below is the sample code for WAF CAPTCHA connection (with Axios as an example). You can refer to the following to connect a frontend-backend separated site to WAF CAPTCHA based on your actual use case:
1. Add interceptors to the Axios response.



```
  // Regexes related to WAF CAPTCHA `seqid`
const sig_data = /seqid\\s=\\s"(\\w+)"/g
const waf_id_data = /TencentCaptcha\\((\\'\\d+\\')/g

const service = axios.create({
 baseURL: '/api',
 timeout: 10000,
 withCredentials: true
});

service.interceptors.response.use((response)=>{
```

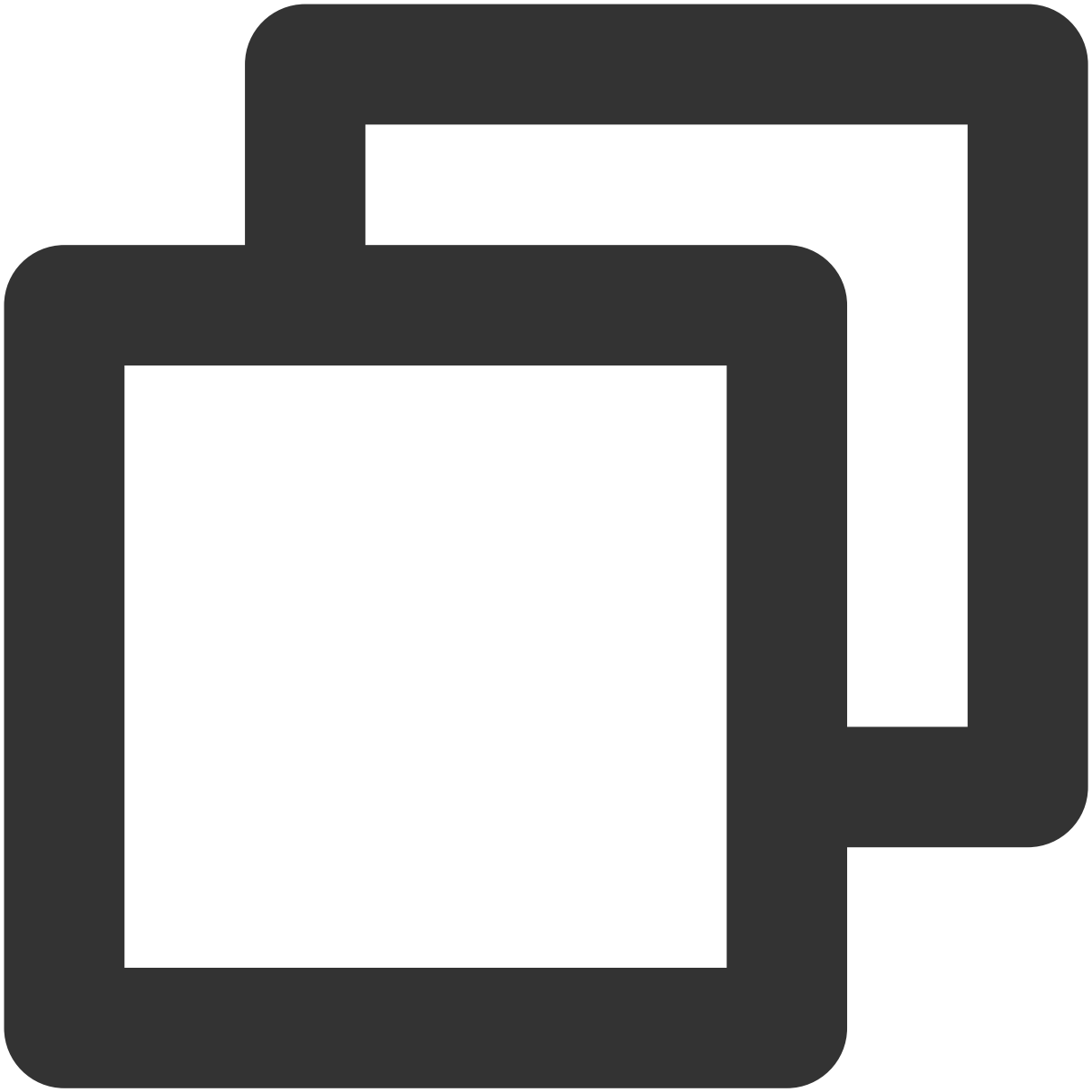
```
  const res = response.data;
  if(res.code === 0){
    return res;
  }else{
    // Capture the error and render the CAPTCHA
    const matches = sig_data.exec(res);
    if(matches){
      // Display the CAPTCHA
      let seqid = matches[1];
        const wid_matches = waf_id_data.exec(res);
      let wid = wid_matches[1]
      var captcha = new TencentCaptcha(wid, function(res){
        var captchaResult = []
        captchaResult.push(res.ret)
        if(res.ret === 0){
            captchaResult.push(res.ticket)
            captchaResult.push(res.randstr)
            captchaResult.push(seqid)
        }
        var content = captchaResult.join('\\n')
        axios.post(
          "/WafCaptcha",content
        ).then().catch();
      });
      captcha.show()
    }else{
      return res;
    }
  }
},()=>{});
export default service;


Vue.prototype.$axios = service;
```

2. Add the Axios response with added interceptors during API call.



```
getTopic:function(){
this.$axios.get("/api.php").then(res => {
 this.topic = res
});
}
```

3. Import the CAPTCHA script globally by adding `<script src="https://ssl.captcha.qq.com/TCaptcha.js"></script>` to `public/index.html`.

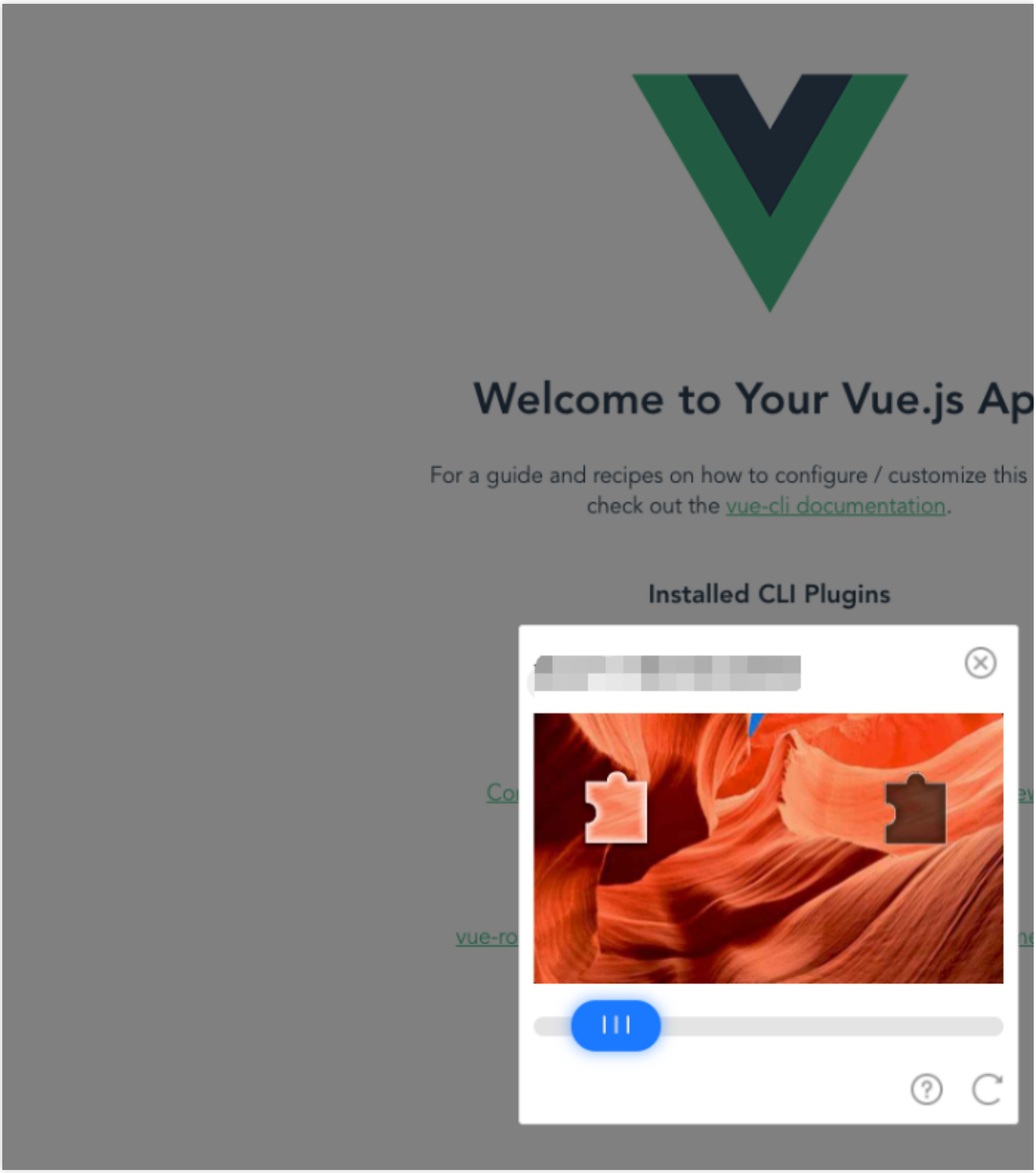


```
<!DOCTYPE html>
<html lang="">
<head>
 <meta charset="utf-8">
 <meta http-equiv="X-UA-Compatible" content="IE=edge">
 <meta name="viewport" content="width=device-width,initial-scale=1.0">
 <link rel="icon" href="<%= BASE_URL %>favicon.ico">
 <title><%= htmlWebpackPlugin.options.title %></title>
</head>
<body>
 <noscript>
```

```
  <strong>We're sorry but <%= htmlWebpackPlugin.options.title %> doesn't work proper
  </noscript>
  <script src="https://ssl.captcha.qq.com/TCaptcha.js"></script>
  <div id="app"></div>
  <!-- built files will be auto injected -->
 </body>
 </html>
```

4. After entering the above code, compile and deploy it on the server.

5. Configure a custom rule in WAF and use an async request to check whether the current page pops up the CAPTCHA window.



Best Practices of Bot Traffic Management Connection

Last updated：2023-12-29 14:55:32

This document describes how to quickly connect to the bot traffic management feature and defend against malicious traffic during routine operations.

Prerequisites

To connect to bot traffic management, you need to purchase an extra pack of WAF.

Note:

Currently, WAF Enterprise and Ultimate users are offered a free trial of the bot traffic management feature to observe how bots affect websites.

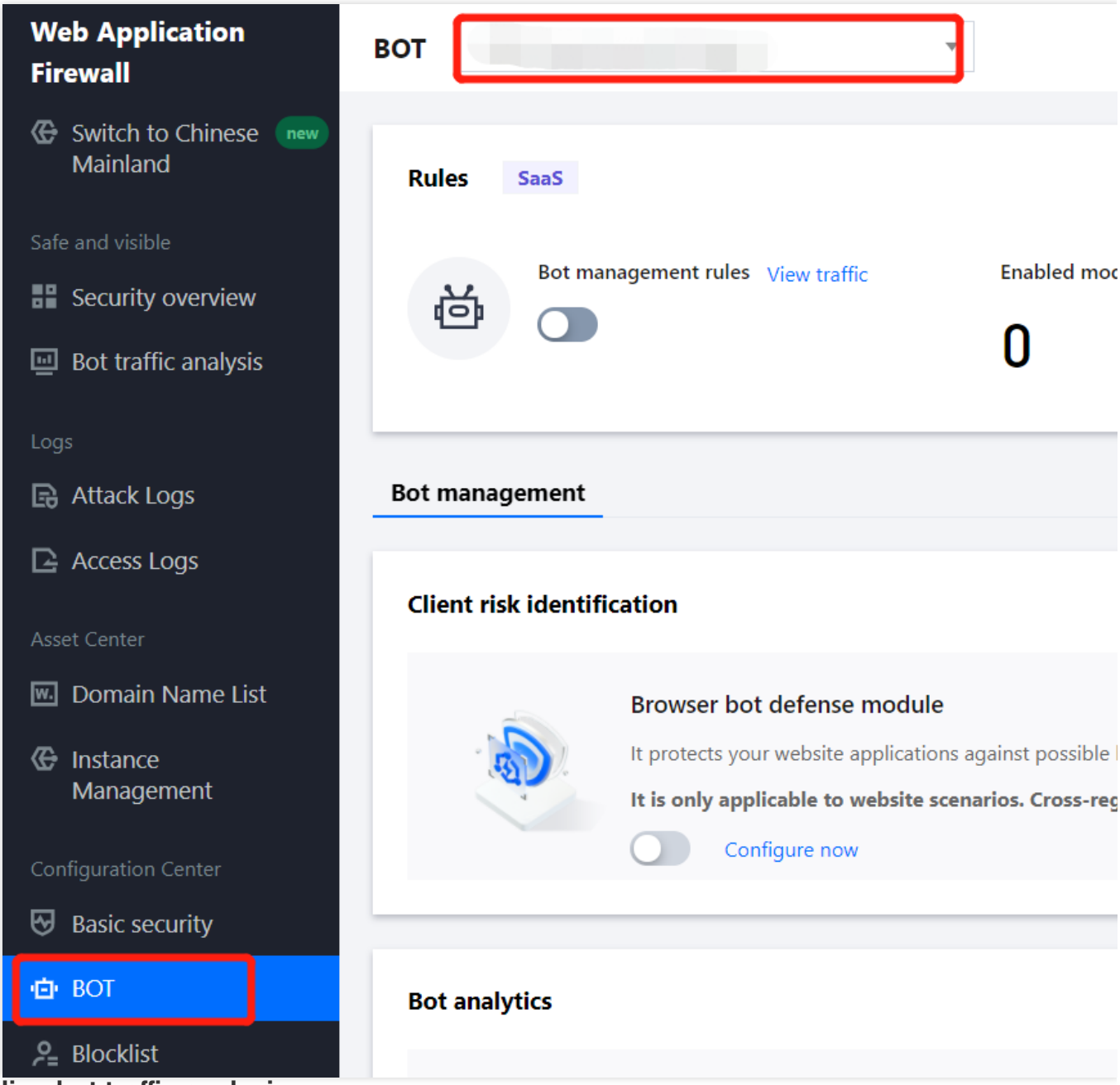
Parsing CAPTCHA

When you use applications, mini programs, and clients as well as cross-domain scheduling, the CAPTCHA issued by the WAF instance cannot be parsed and recognized. Therefore, the bot traffic management feature cannot parse and pop up the CAPTCHA for verification. After multiple CAPTCHAs are triggered, the access requests of normal users will be blocked, affecting the business.

Therefore, when configuring a CAPTCHA action, you need to modify the frontend/client business accordingly as instructed in Connecting Frontend-Backend Separated Site to WAF CAPTCHA.

General Business Connection

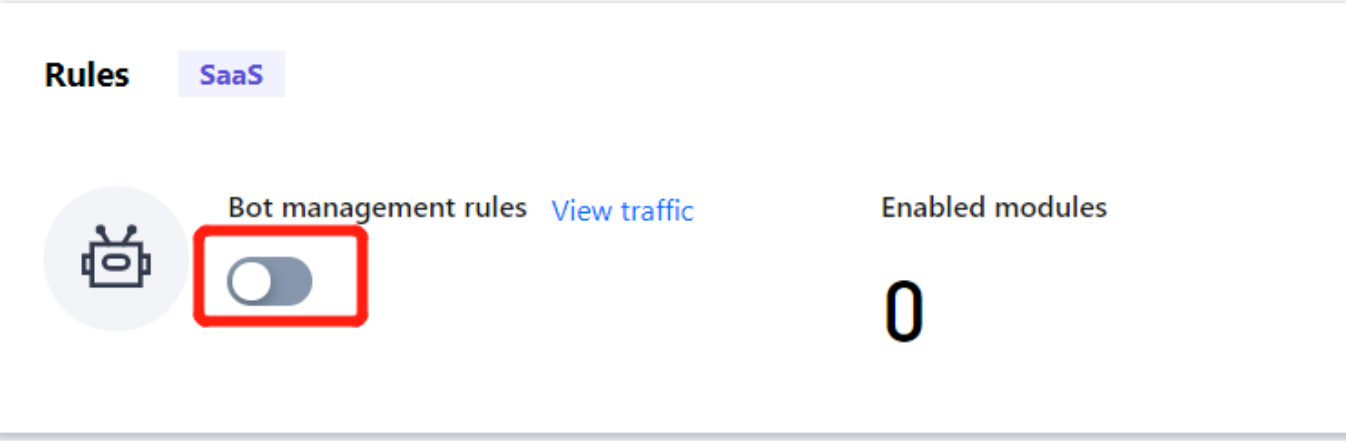
1. Log in to the WAF console and select **Configuration center** > **Bot and application security** on the left sidebar.
2. On the **Bot and application security** page, select the target domain name in the top-left corner and click **Bot management**.



Enabling bot traffic analysis

On the **Bot management** page, click

in the **Rules** section.



Setting browser bot defense module

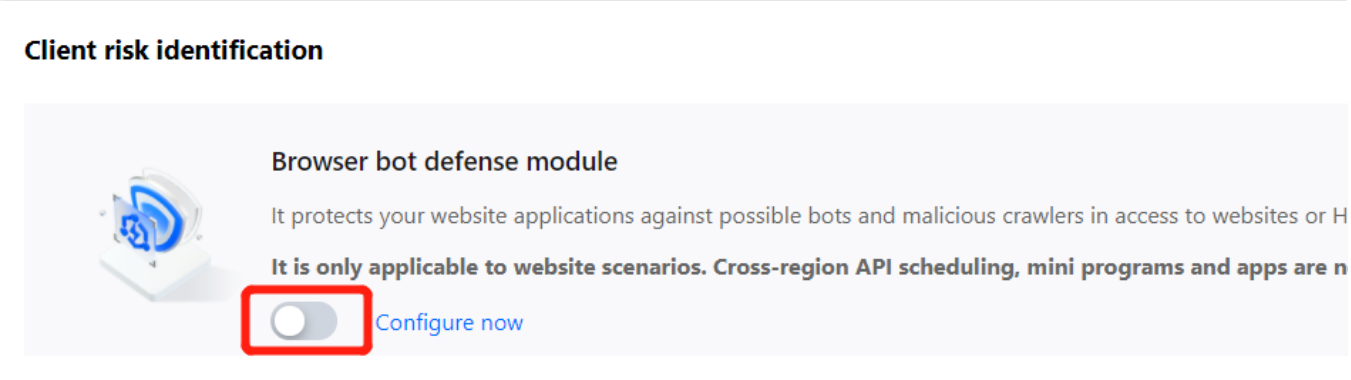
1. In **Browser bot defense module** on the **Bot management** page, click

.

Note:

Make sure that your client is a WeChat Official Account, HTML5 page, application, mini program, or PC client. When you only have a browser, WeChat Official Account, or HTML5 page as the client and need cross-domain scheduling, enable the browser bot defense module to achieve the best protection.

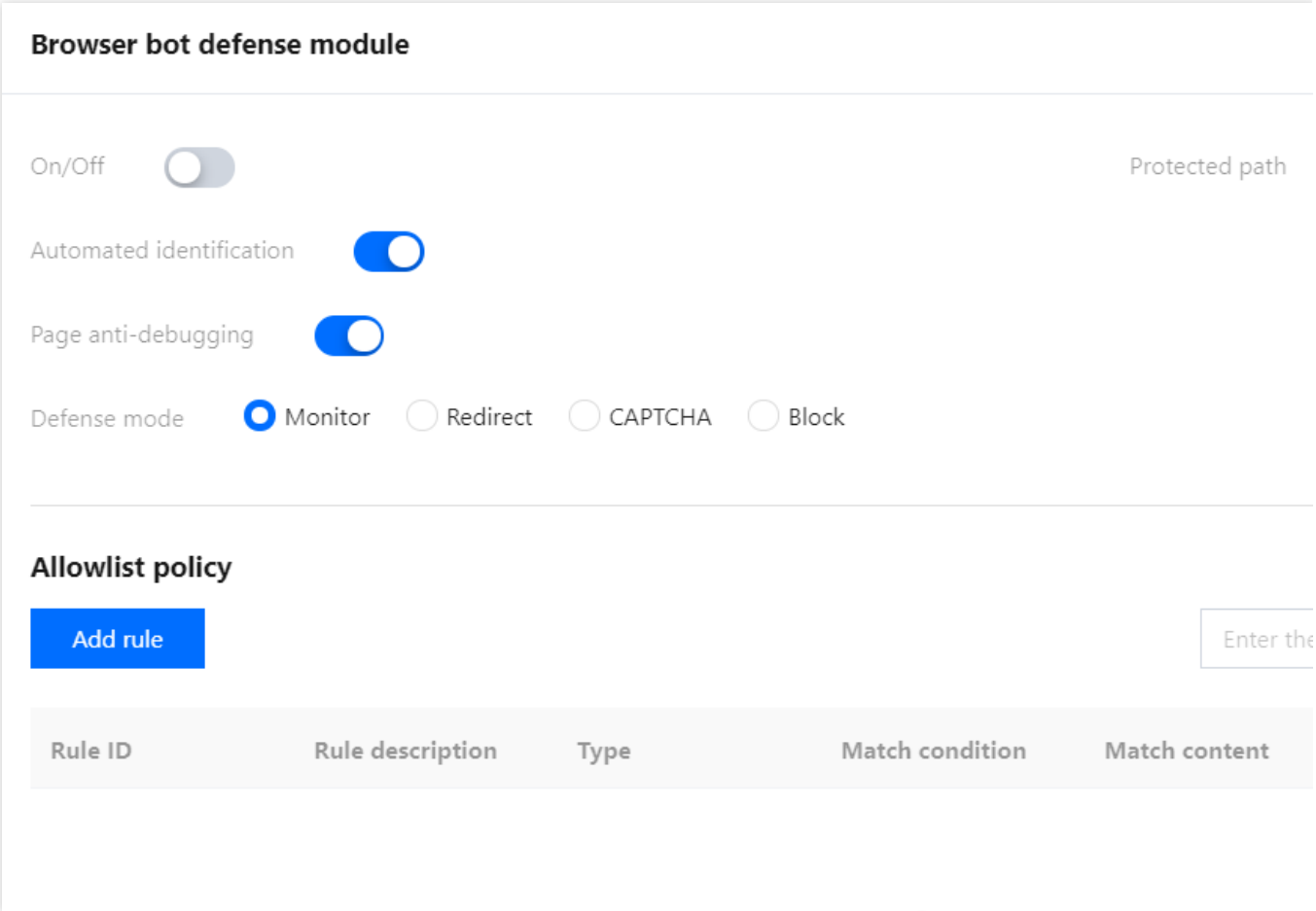
After the browser bot defense module is enabled, when its protection path is accesses, the system will check whether the client is capable of parsing JavaScript. A JavaScript code snippet will be issued to verify whether the client is a real browser. For mini programs, applications, and API calls, the query issued by WAF will not be actively parsed, so the client cannot perform parsing normally.



2. In the browser bot defense module, click **Configure now** to configure protection for key pages.

Note:

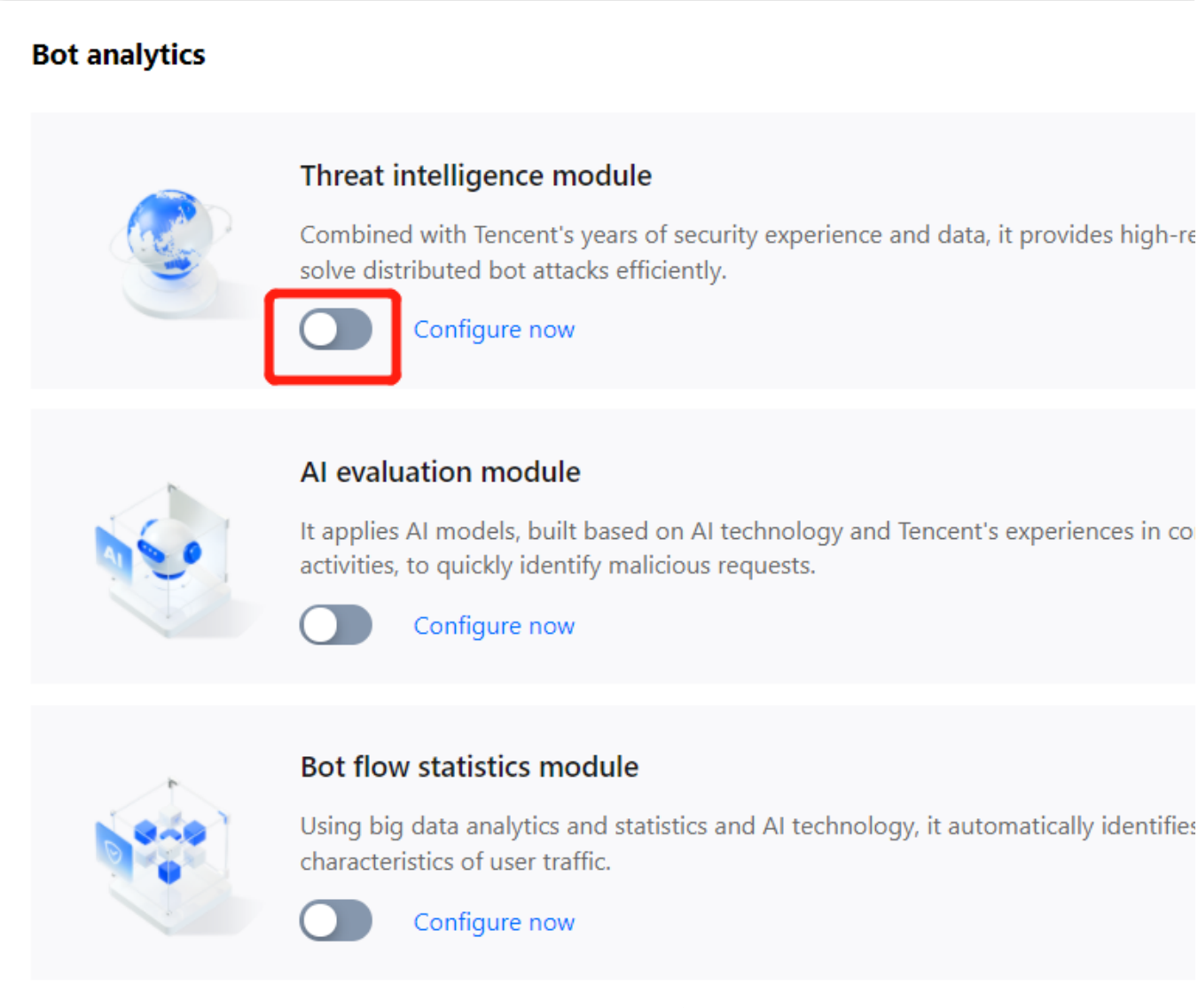
For more information, see Bot Management.



Setting threat intelligence module

1. In **Threat intelligence module** on the **Bot management** page, click

. When the module is enabled for the first time, all recognition items will be enabled. After you enable corresponding items, you can recognize the access sources at different malicious levels from the threat intelligence module and IDC.



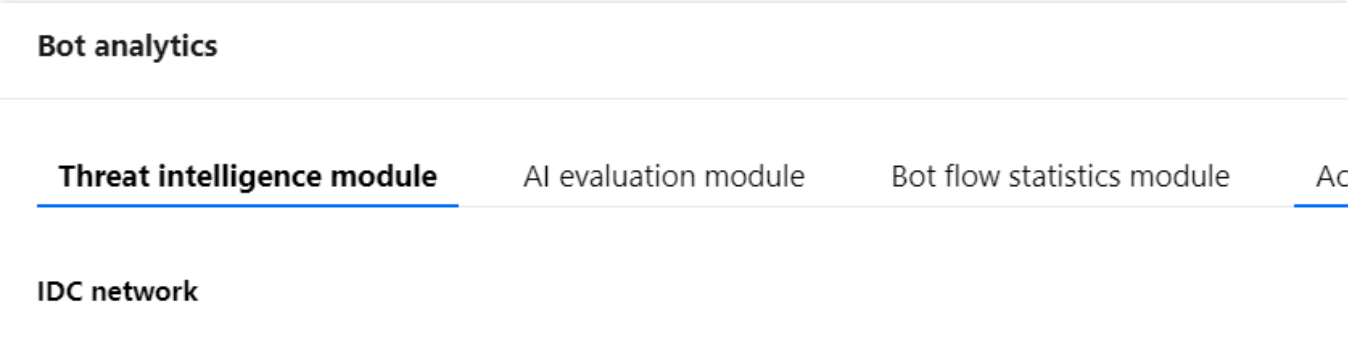
2. In the threat intelligence module, click **Configure now** to set the IDC network and threat intelligence library.

Note:

The current business callback API is in the IDC domain:

If you are not sure about a source IP, contact us to add the IDC to the allowlist, that is, to disable the IDC option in the threat intelligence module for the business.

If you are sure about the current business callback IP, add the source IP to the allowlist in **Custom rules**. For more information, see Precise Allowlist Management.



Enable all Disable all

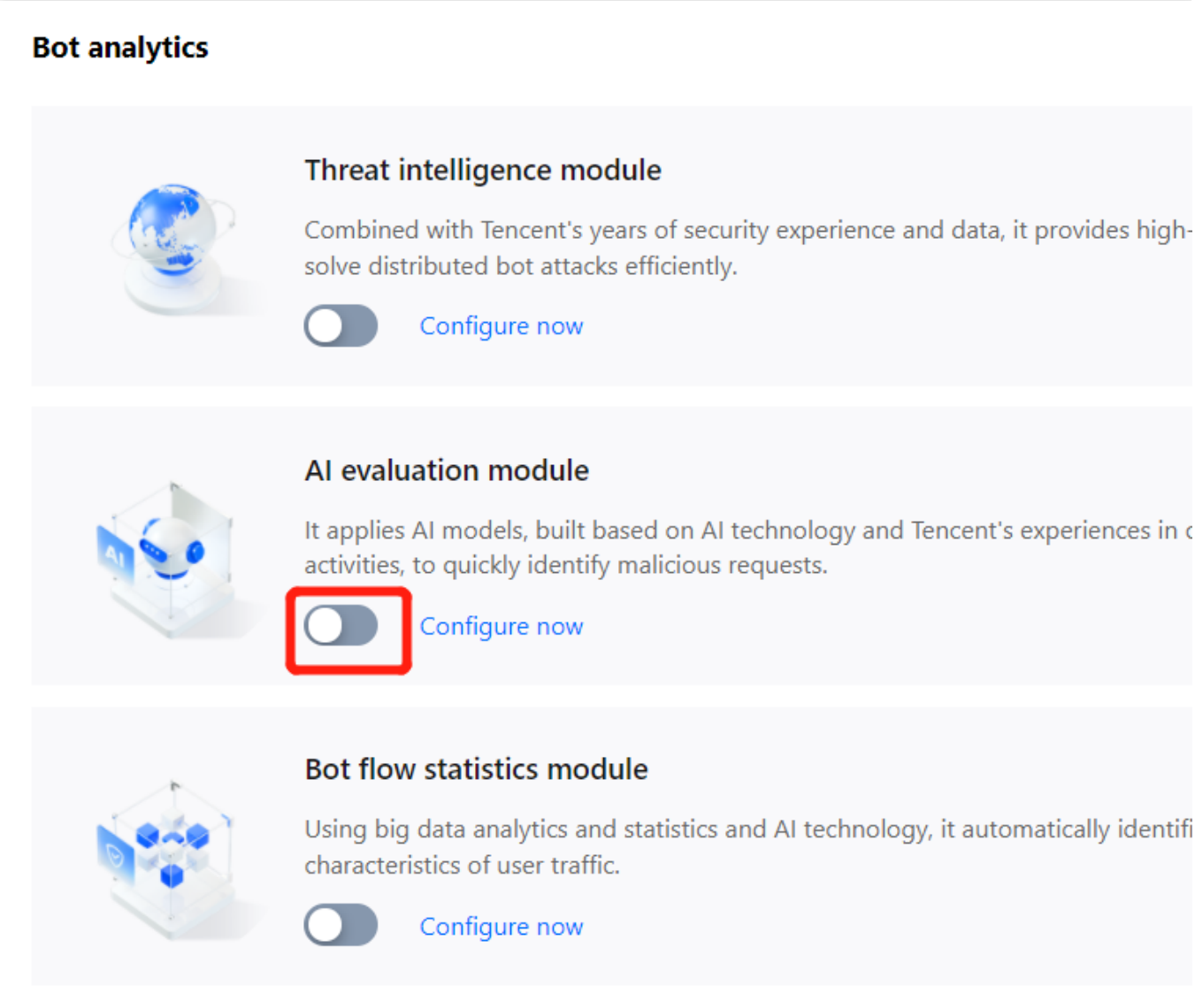
IDC network type	IDC network description
Aws	The IPs belong to the AWS (IDC IP) IP library, and are often exploited by attackers to dep
Azure	The IPs belong to the Microsoft Azure (IDC IP) IP library, and are often exploited by attac
Google	The IPs belong to the GCP (IDC IP) IP library, and are often used by attackers to deploy b
UCloud	The IPs belong to the UCloud (IDC IP) IP library, and are often exploited by attackers to c
Alibaba Cloud	The IPs belong to the Alibaba Cloud (IDC IP) IP library, and are often exploited by attacke
Baidu Cloud	The IPs belong to the Baidu Cloud (IDC IP) IP library, and are often exploited by attackers
Huawei Cloud	The IPs belong to the Huawei Cloud (IDC IP) IP library, and are often exploited by attacke
Kingsoft Cloud	The IPs belong to the Jinshan Cloud (IDC IP) IP library, and are often exploited by attacke
pubyun	The IPs belong to the Baidu Cloud (IDC IP) IP library, and are often exploited by attackers
Qing Cloud	The IPs belong to the Qing Cloud (IDC IP) IP library, and are often exploited by attackers
Tencent Cloud	The IPs belong to the Tencent Cloud (IDC IP) IP library, and are often exploited by attack

Threat intelligence library

Enabling AI evaluation module

In **AI evaluation module** on the **Bot management** page, click

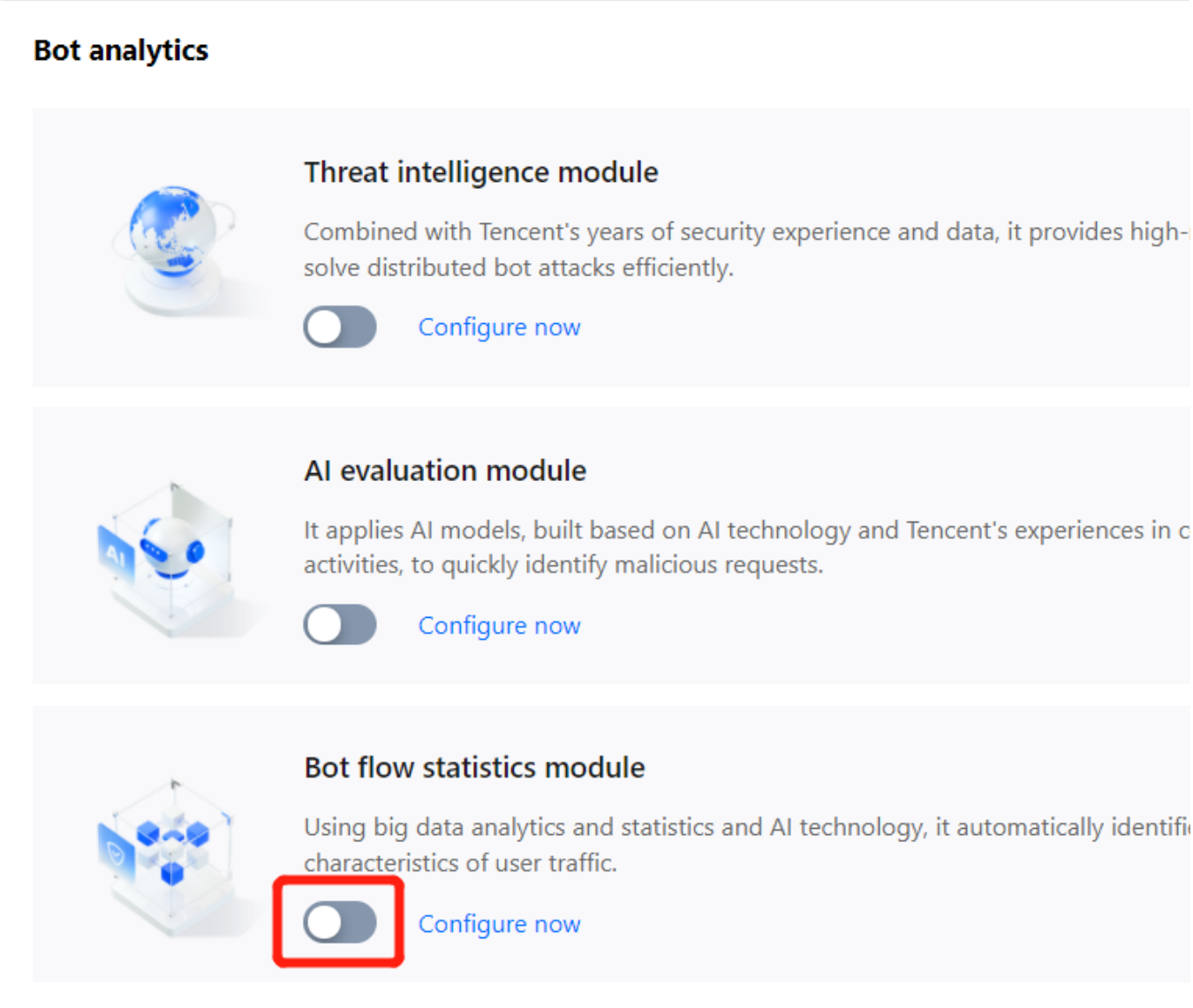
.



Enabling bot flow statistics module

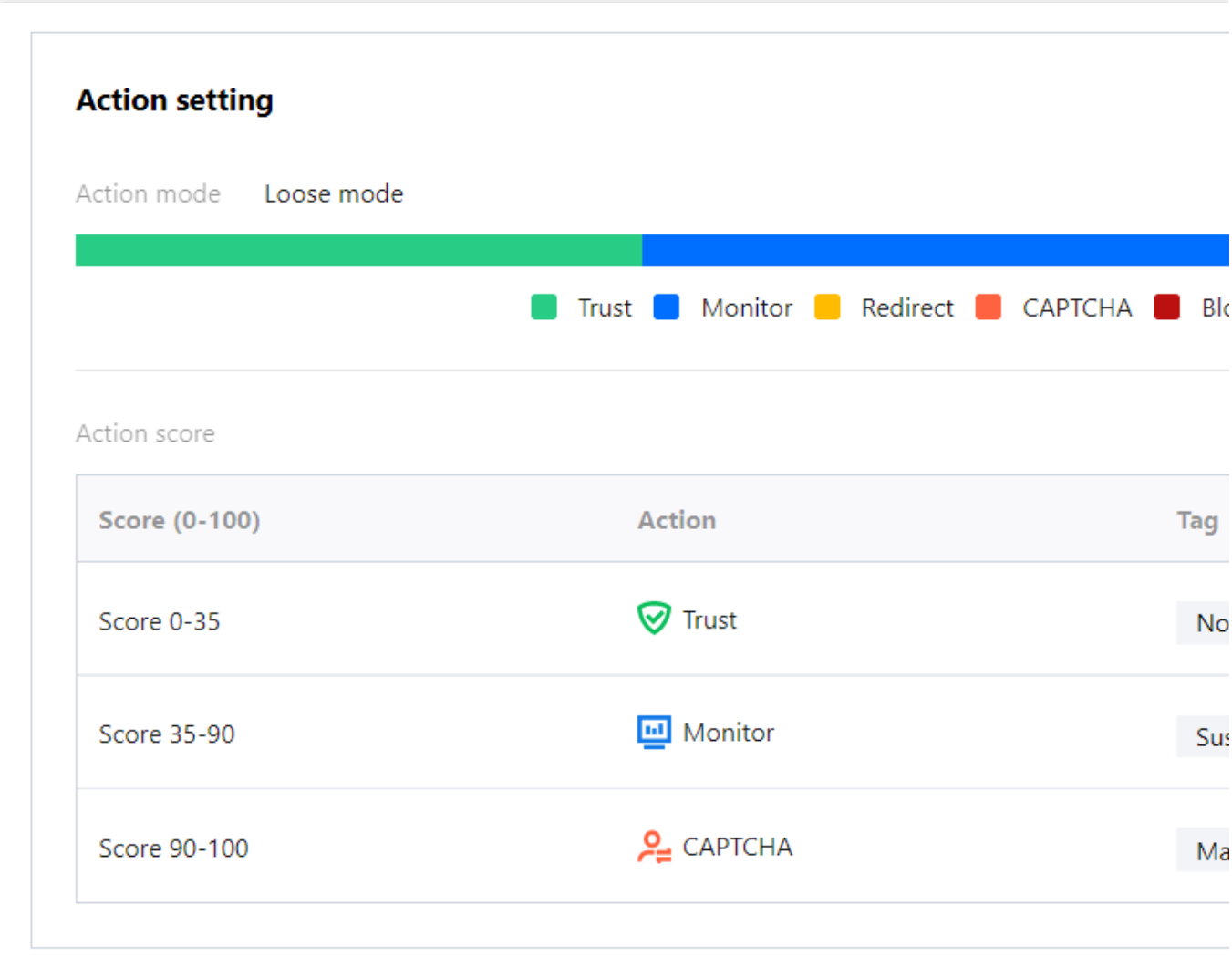
In **Bot flow statistics module** on the **Bot management** page, click

.

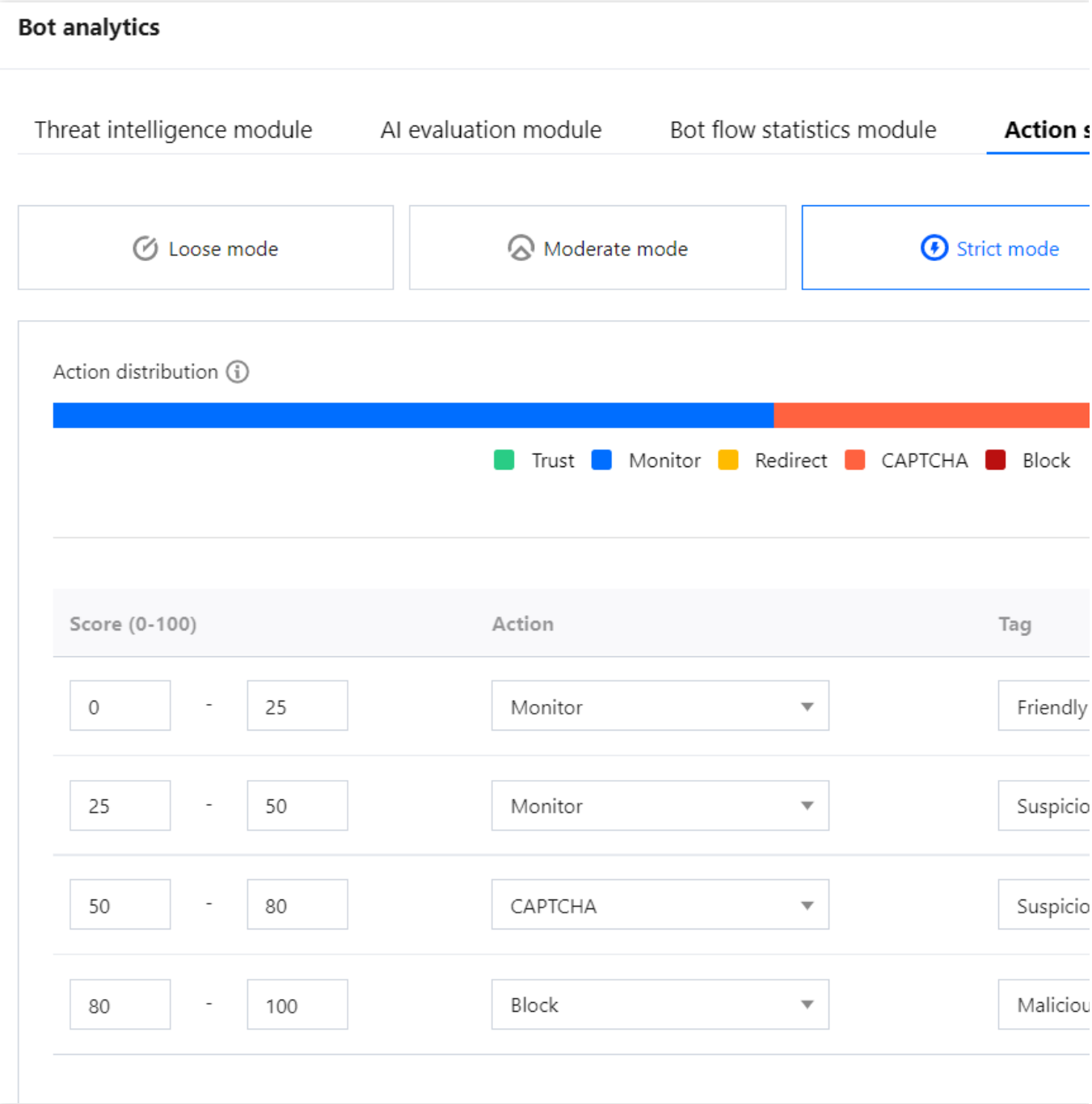


Setting action score

1. In the **Action setting** section on the **Bot management** page, click **Action score**.



2. On the **Action setting** tab, you can configure the score and action to precisely block risky access requests.



Use instructions

Mode: By default, there are loose, moderate, strict, and custom modes. The first three modes are preset, representing different recommended categories and handling policies for bots at different malicious levels in bot traffic management. Once modified, they become the custom mode.

Score range: A score ranges from 0 to 100. Ten score entries can be added to each range, which is left-closed and right-open and cannot be overlapped. You can set a range to null, and then no action will be processed in it.

Action: You can set an action to **Trust**, **Monitor**, **Redirect** (to a certain website URL), **CAPTCHA** (verification code), or **Block**.

Tag: You can set the tag to **Friendly bots**, **Malicious bots**, **Normal traffic**, or **Suspicious bots**.

Friendly bots: The bot is friendly and legal for the website by default.

Suspicious bots: The system finds the access source traffic suspicious but cannot determine if it is malicious to the website.

Normal traffic: The access traffic is regarded as from a real user.

Malicious bots: The bot has malicious traffic and is unfriendly to the website.

3. After completing the configuration, click **Publish** in the bottom-left corner of the page.