阿里云
CDN

域名管理

文档版本：20190716

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **禁止:** 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **警告:** 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 **说明:** 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定。
`courier` 字体	命令。	执行 `cd /d C:/windows` 命令，进入Windows系统文件夹。
*## *	表示参数、变量。	`bae log list --instanceid` *`Instance_ID`*
[]或者[a|b]	表示可选项，至多选择一个。	`ipconfig` *`[-all|-t]`*
{}或者{a|b}	表示必选项，至多选择一个。	`swich` *`{stand | slave}`*

目录

1 控制台说明

本文主要介绍阿里云CDN控制台界面上展示的相关功能。阿里云CDN控制台不仅可以帮助您完成配置域名等基本操作，也提供了实时数据分析的资源监控服务。同时您还可以了解自己的计费情况，随时变更计费方式。

控制台指引

控制台的界面展示如下：

- 左侧导航栏：控制台左侧为域名管理操作菜单栏，详细功能介绍请参见CDN功能列表。
- 概览区：控制台中部为概览区，包括三个部分：您的昨日使用数据、CDN使用指南和其他加速产品。
  - 昨日使用数据：根据您的计费方式，系统会在这里展示您计费项中的使用数据。
  - CDN使用指南：您可以在这里查阅CDN相关的使用指南。如果您想了解更多，可以参考CDN学习路径。
  - 其他加速产品：您可以了解CDN的其他产品。如果您对安全有更高的需求，可以选择安全加速SCDN；如果您对动态加速有重点需求，可以选择阿里云全站加速。
- 右侧计费展示区：包括您的计费方式、资源包数量、域名数量和域名流量排名。

2 CDN功能列表

本文为您介绍了CDN的相关功能，具体功能信息请参见相关文档。

HTTPS安全加速

项目	说明	默认值
HTTPS安全加速	提供全链路HTTPS安全加速方案，仅需开启安全加速模式后上传加速域名证书/私钥，并支持对证书进行查看、停用、启用、编辑操作。	未开启
强制跳转	加速域名开启HTTPS安全加速的前提下，支持自定义设置，将您的原请求方式进行强制跳转。	未开启
HTTP/2设置	开启HTTP/2，您可以享受二进制协议带来的更多扩展性、内容安全性、多路复用、头部压缩等优势。	未开启
TLS	TLS协议版本开启后，您的加速域名也将开启TLS握手。	目前TLSv1.0、TLSv1.1和TLSv1.2版本默认开启。
HSTS	HSTS的作用是强制客户端（如浏览器）使用HTTPS与服务器创建连接。	未开启

回源设置

项目	说明	默认值
回源HOST	指定回源HOST域名，提供三种选项：加速域名、源站域名、自定义域名。	加速域名
协议跟随回源	开启该功能后，回源使用协议和客户端访问资源的协议保持一致。	未开启

项目	说明	默认值
私有Bucket回源	若加速域名想要回源至您账号下标记为私有的bucket时，需要首先进行授权，授权成功并开启授权配置后，您开启了私有bucket授权的域名有权限访问私有bucket。	未开启

缓存设置

项目	说明	默认值
缓存过期时间	自定义指定资源内容的缓存过期时间规则	未开启
设置HTTP头	可设置http请求头，目前提供10个http请求头参数可供自行定义取值。	未开启
自定义404页面	提供三种选项：默认404、公益404、自定义404。	默认404

访问控制

项目	说明	默认值
Refer防盗链	您可以通过配置访问的referer黑白名单来对访问者身份进行识别和过滤	未开启
鉴权配置	URL鉴权方式保护您源站资源	未开启
IP黑名单	您可以通过配置访问的 IP黑名单来对访问者身份进行识别和过滤	未开启

性能优化

项目	说明	默认值
页面优化	压缩与去除页面中无用的空行、回车等内容，有效缩减页面大小。	未开启
智能压缩	支持多种内容格式的智能压缩，有效减少您传输内容的大小。	未开启

项目	说明	默认值
过滤参数	勾选后，回源会去除url中？之后的参数。	未开启

视频相关设置

项目	说明	默认值
Range回源	指客户端通知源站服务器只返回指定范围的部分内容，对于较大文件的分发加速有很大帮助。	未开启
拖拽播放	开启即支持视音频点播的随机拖拽播放功能	未开启

高级配置

项目	说明	默认值
带宽封顶	当统计周期（5分钟）产生的平均带宽超出所设置的带宽最大值时，为了保护您的域名安全，此时域名会自动下线，所有的请求会回到源站。	未开启

刷新与预热

项目	说明	默认值
URL刷新和预热	· 通过提供文件URL的方式，强制CDN节点回源拉取最新的文件。 · 将指定的内容主动预热到CDN的L2节点上，您首次访问即可直接命中缓存，降低源站压力。	开启

数据监控与统计分析

项目	说明	默认值
数据监控	您可以选择想监控的域名、区域、运营商、时间粒度（1分钟、5分钟、1小时）以及想查询的时间段（今天、昨天、近7天、近30天或自定义）。	开启
统计分析	统计分析包含五个部分：PV和UV、地区和运营商、域名排名、热门Refer、热门URL。您可以导出原始详细数据，如网络带宽、流量，域名按流量占比排名以及访客区域、运营商分布等。	开启

用量查询

项目	说明	默认值
用量查询	查询并获取到某一段时间内的实际用量数据（流量、带宽或请求数），您可以使用用量查询功能。	开启
账单导出	导出按日计费，或者是按月计费的实际用量数据，以便于与费用中心的出账用量进行比对。	开启
明细导出	导出流量带宽及请求数的5分钟明细数据，便于您通过明细来核对或计算实际消费的计量数。	开启

日志管理

项目	说明	默认值
日志下载	您可以下载最近一个月的日志数据。	开启
实时日志	在借助CDN加速访问资源的过程中，CDN会产生大量的日志数据，这些日志数据CDN会进行实时的采集。	开启

项目	说明	默认值
日志转存	帮助您将日志存储更长的时间，目前CDN的离线日志服务，默认提供1个月的存储时间。如果您有更长时间的存储需求，可以将日志转存至OSS，方便您根据实际情况对日志进行保存和分析。	开启

其他设置

项目	说明	默认值
设置httpDNS	httpDNS是域名解析服务，通过HTTP协议直接访问阿里云CDN的服务器。	未开启

3 批量复制

通过批量复制域名配置功能，您可以将某一个加速域名的一个或多个配置，复制到另外一个或者多个域名上。

前提条件

您在进行批量复制前，请确保已经启用并配置了您想复制的域名，否则将无法批量复制。

背景信息

您在批量复制某个域名的配置时，请注意：

- 复制的内容会覆盖目标域名已经配置的内容，请您谨慎操作，以免造成服务不可用。
- 域名复制后，复制不可回退。请确认被复制的域名正在服务或已有配置，且流量带宽较大。请务必确认您的域名复制选择无误，谨慎操作。

操作步骤

1. 登录CDN控制台。
2. 在域名管理页，选择您想要复制配置的域名，单击复制配置。

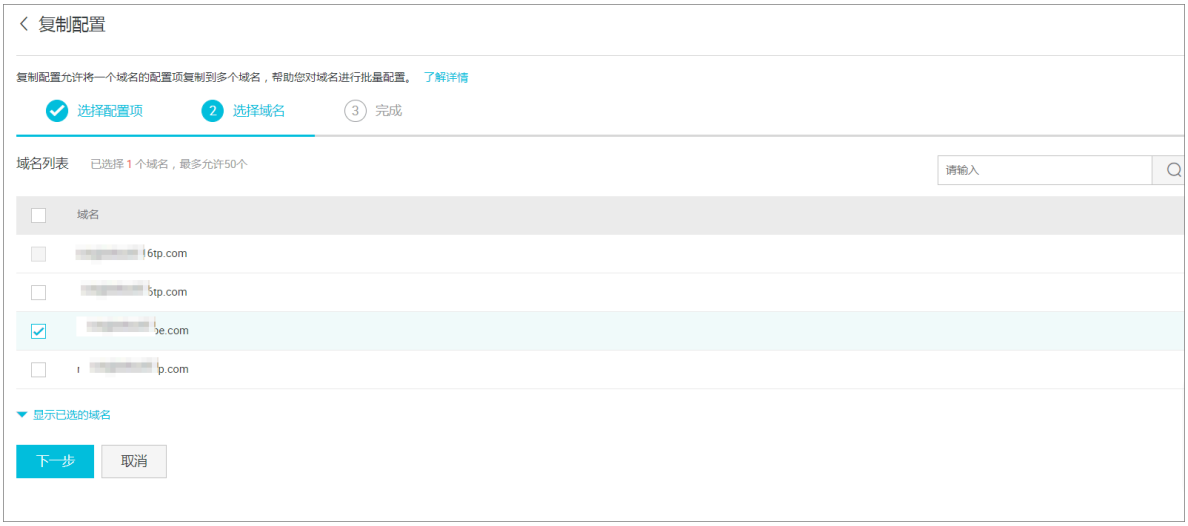
3. 勾选您想要复制的配置项，单击下一步。

> 说明：
>
> - 源站信息和非源站信息无法同时复制。
> - 您无法复制HTTPS证书到其他域名，请您单独配置。
> - 自定义回源头为增量复制。例如，假设您的A域名有2条回源头配置，您从B域名复制了5条内容，则你会有7条回源头配置内容。
> - HTTP头为非增量复制。假设您的A域名配置了cache_control为private，您的B域名配置为public，复制后，您的cache_control为public。
> - 开关类的配置复制，将会覆盖域名原有的配置。

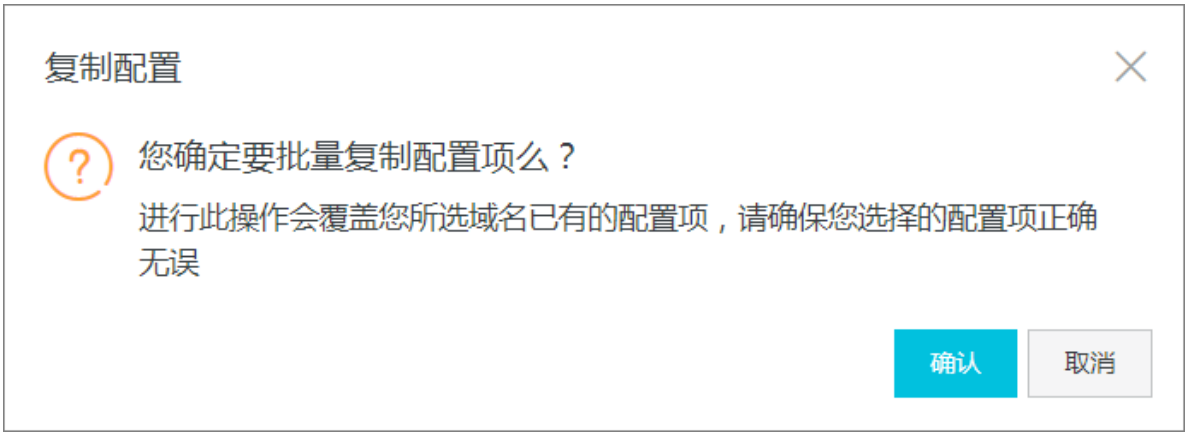
· **Refer黑白名单或IP黑白名单将会覆盖域名原有配置。**

4. **勾选您想要批量配置的目标域名，单击下一步。**

   您也可以输入关键词查找域名。



5. **在复制配置对话框中，单击确认，批量复制成功。**

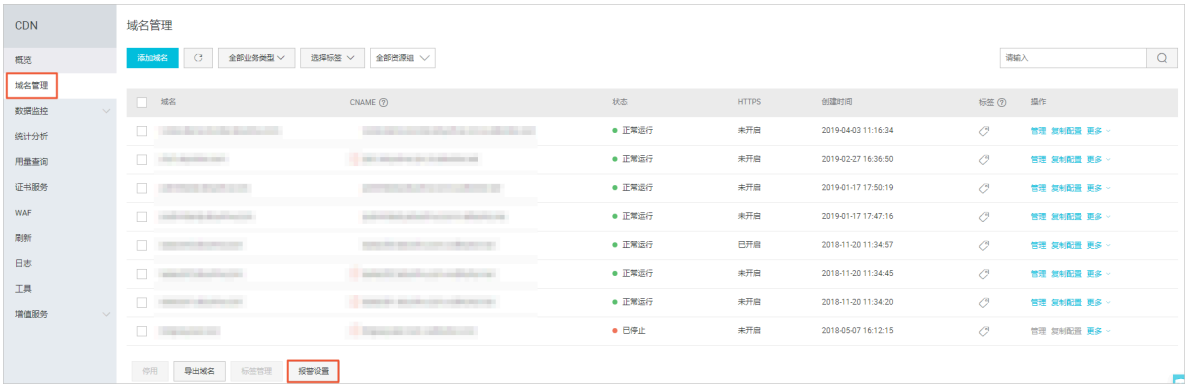


4 设置报警

当您需要监控CDN域名的带宽峰值、4xx5xx返回码占比、命中率、下行流量、QPS等监控项时，您可以直接在阿里云的云监控控制台设置报警规则。当报警规则被触发时，阿里云监控会根据您设置的短信、邮件等通知方式给您发送报警信息。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 单击报警设置，跳转到云监控控制台。



4. 选择云监控服务 > CDN，单击报警规则页签。
5. 单击创建报警规则。

6. 创建针对CDN的报警规则，详情请参见创建阈值报警规则。

5 标签管理

5.1 概述

阿里云CDN不对标签进行任何定义，仅严格按字符串对标签和域名进行匹配、筛选。您可以通过标签管理功能，对加速域名进行绑定标签、解绑标签、分组管理和筛选数据。

使用限制

- 每个标签都由一个键值对（Key:Value）组成。
- 每个域名最多绑定20个标签。
- 同一个域名的标签键（Key）不能重复。如果对一个域名设置2个同Key不同Value的标签，新值将覆盖旧值。例如对域名`test.example.com`先后设置了标签Key1:Value1和Key1:Value2，则最终`test.example.com`只会绑定标签Key1:Value2。
- 键（key）不支持aliyun、acs:开头，不允许包含http://和https://，不允许为空字符串。
- 值（value）不允许包含http://和https://，允许为空字符串。
- 最大键（key）长度：64个Unicode字符。
- 最大值（value）长度：128个Unicode字符。
- 区分大小写。

相关功能

您可以使用标签，对域名进行以下操作：

- 绑定标签，创建用于标记域名的用途或对域名进行分组管理的标签。
- 解绑标签，删除已经不再适用于您当前某个或多个域名用途的标签。
- 使用标签管理域名，域名绑定标签后，您可以使用标签，快速筛选对应的域名，进行分组管理。
- 使用标签筛选数据，域名绑定标签后，您可以使用标签，快速筛选对应的域名，查询域名数据。

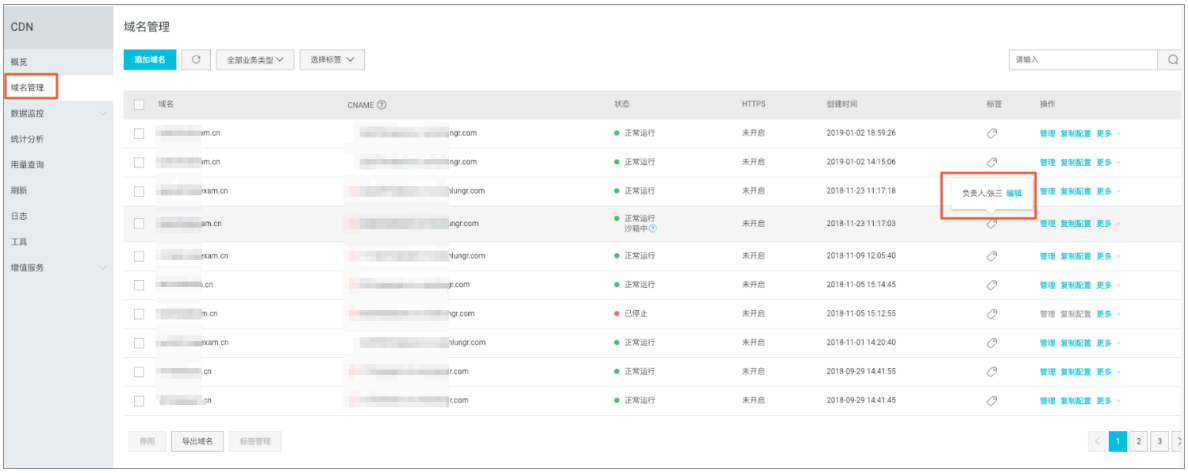
5.2 绑定标签

如果您需要标记域名或为域名分组，可以通过标签功能为该域名绑定标签。

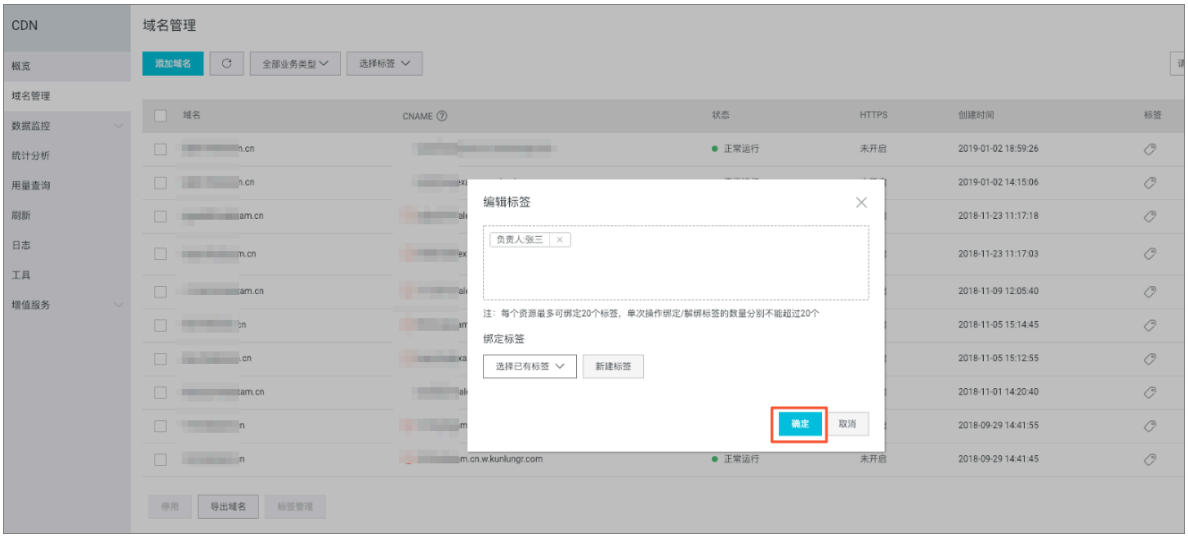
操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 选择您想要设置标签的域名，将鼠标移动到对应标签上。

4. 在浮窗内，单击编辑。



5. 在编辑标签对话框，您可以选择已有标签或新建标签进行绑定。



6. 编辑完成后，单击确定。

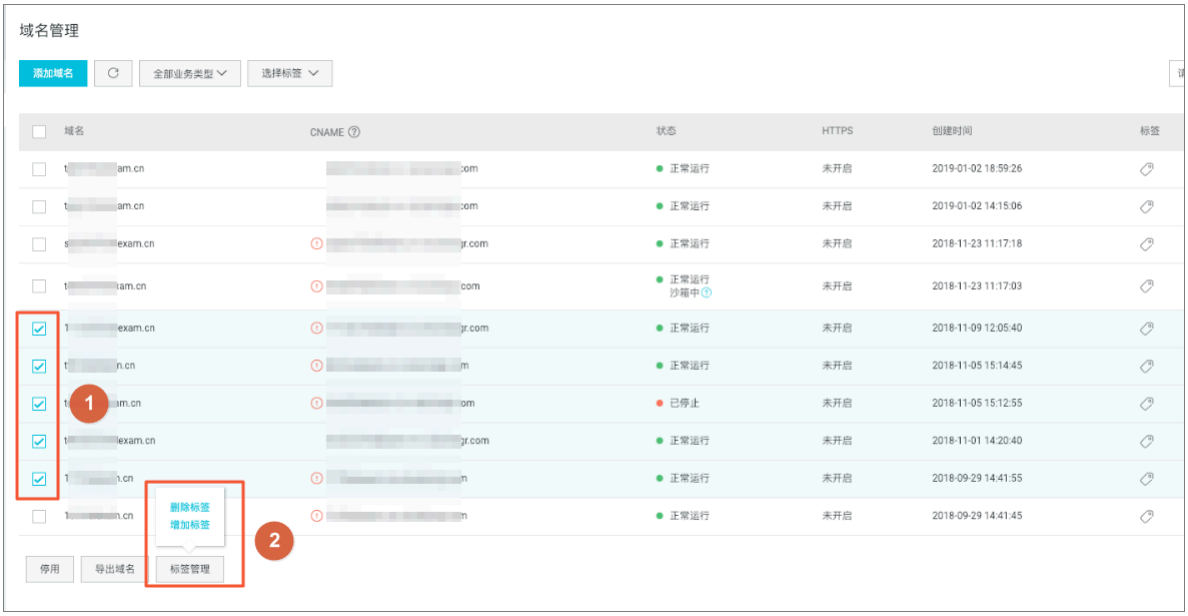
5.3 解绑标签

如果标签已经不再适用于您当前某个或多个域名的用途时，您可以解绑域名标签。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。

3. 勾选您想要处理的域名，选择标签管理 > 删除标签。



4. 在批量删除标签对话框，选择您需要删除的标签并单击确定，完成解绑。

5.4 使用标签管理域名

您可以在域名绑定标签后，使用标签，快速筛选对应的域名，进行分组管理。

操作步骤

1. 登录CDN控制台。

2. 单击域名管理。

3. 在域名管理页面，单击选择标签。

5.5 使用标签筛选数据

如果您需要查询部分域名的数据，您可以在域名绑定标签后，使用标签，快速筛选对应的域名，查询相关数据。

操作步骤

1. 登录CDN控制台。

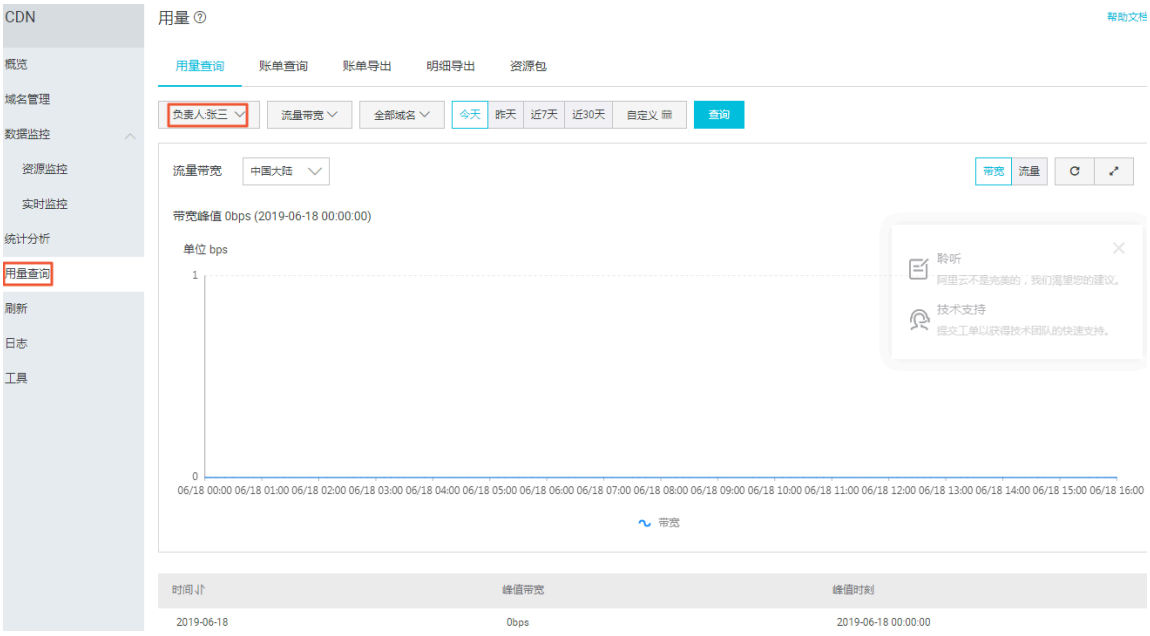
2. 您可以通过如下两种方式筛选并查询数据。

> **说明:**

如果您同时选择多个标签，则查询的结果是各个标签对应域名的交集。

- **选择数据监控 > 资源监控，选择对应的键（Key）和值（Value）标签，单击查询。**

- **单击用量查询，选择对应的键（Key）和值（Value）标签，单击查询。**



5.6 案例介绍

本文通过举例为您介绍如何使用标签进行域名的分组管理。

某公司在阿里云CDN拥有100个域名，分属电商、游戏、文娱三个部门，服务于营销活动、游戏 A、游戏 B、后期制作等业务。公司有三位运维负责人，分别是张三、李四、王五。

设置标签

为了方便管理，该公司使用标签来分类管理对应的域名，定义了下述标签键（Key）和值（Value）。

键（Key）	值（Value）
部门	电商、游戏、文娱
业务	营销活动、游戏 A、游戏 B、后期制作
负责人	张三、李四、王五

将这些标签的键和值绑定到域名上，域名与标签键值的关系如下表所示：

域名	Key为部门，Value为	Key为业务，Value为	Key为负责人，Value为
domain1	电商	营销活动	王五
domain2	电商	营销活动	王五
domain3	游戏	游戏 A	张三
domain3	游戏	游戏 B	张三
domain4	游戏	游戏 B	张三
domain5	游戏	游戏 B	李四
domain6	游戏	游戏 B	李四
domain7	游戏	游戏 B	李四
domain8	文娱	后期制作	王五
domain9	文娱	后期制作	王五
domain10	文娱	后期制作	王五

使用标签

- 如果您想筛选出王五负责的域名，则选择标签负责人：王五。
- 如果您想筛选出游戏部门中李四负责的域名，则选择标签部门：游戏和负责人：李四。

6 基本配置

6.1 概述

通过基本配置功能，您可以查看加速域名的基础信息和源站信息。此外，您还可以进行切换域名的加速区域和源站设置。

计费说明

- 如果您选择的源站类型为IP或源站域名，则仍然按照外网流量价格计费。
- 如果您选择的源站类型为OSS域名，即从CDN回源OSS，则按照内网的价格计费，具体价格请参见OSS价格详情。
- 如果选择域名类型为源站域名，并设置了一个OSS的域名，则仍然按照外网流量价格计费。

相关功能

您可以进行以下基本配置：

- 切换加速区域，变更您的CDN服务范围。
- 配置源站，修改源站类型、源站地址、端口等源站信息。

6.2 修改基础信息

当您需要变更您的CDN服务范围时，您可以通过切换加速区域功能实现。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 在基础信息区域框单击修改配置。

5. 在加速区域对话框选择您需要切换的加速区域，单击确定。

6.3 配置源站

当您需要变更源站类型时，通过本文档，您可以了解源站类型的修改方法，以及注意事项。

背景信息

如果您的业务类型为直播流媒体，则不支持源站配置。

阿里云CDN支持三种类型回源域名，包括OSS域名、IP和源站域名。其中，IP和源站域名支持多IP或多域名设置，并支持用在多源站场景下，进行回源优先级设置。

> 说明:
>
> 源站健康检查：实行主动四层健康检查机制，测试源站的80端口。每2.5秒检查一次，连续3次失败标记为不可用。

CDN主要支持主备方式切换源站场景。当多个源站回源时，优先回源优先级为主的源站。如果主站连续3次健康检查均失败，则回源优先级为备的源站。如果该源站的主站健康检查成功，则该源站将重新标记为可用，恢复其优先级。当所有源站的回源优先级相同时，CDN将自动轮询回源。

操作步骤

1. 登录CDN控制台。
2. 在左侧导航栏，单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 在左侧导航栏，单击基本配置。

5. 在源站信息区域框，单击修改配置。
6. 在源站配置对话框，设置源站类型、源站地址和端口。

源站类型说明如下：

源站类型	说明
OSS域名	您可以自定义OSS源站域名，OSS作为源站能够为您节省更多回源流量费用。OSS域名必须以aliyuncs.com或aliyun-inc.com结尾，例如：xxx.oss-cn-hangzhou.aliyuncs.com。
IP	您可以配置多源站IP地址，为源站设置主备优先级。
源站域名	您可以配置多源域名，为源站设置主备优先级。

端口说明如下：

端口	说明
80端口	资源以HTTP或HTTPS协议回源到80端口。
443端口	资源以HTTP或HTTPS协议回源到443端口。如果您的源站为单个IP地址提供多个域名服务，您需要配置回源SNI。
自定义端口	目前仅支持以HTTP协议回源到自定义端口。请先将静态协议跟随回源配置为HTTP协议，再配置自定义端口，操作方法请参见配置协议跟随回源。 · 如果您的静态协议配置为跟随，则无法配置自定义端口。 · 如果您通过OpenAPI将回源协议设置为跟随，请确保您的回源协议和自定义端口均能正常使用。 · 如果您通过端口设置了回源协议（HTTP或HTTPS）和自定义端口，则无论您在CDN控制台如何设置，都按照端口配置回源。

7. **单击确认**。

7 回源配置

7.1 配置回源HOST

如果您需要自定义CDN节点回源时需要访问的具体服务器域名，则需要配置回源HOST的域名类型。回源HOST可选域名类型包括：加速域名、源站域名和自定义域名。

背景信息

回源HOST指CDN节点在回源过程中，在源站访问的站点域名。

说明:
如果您的源站绑定了多个域名或站点时，您需要在自定义域名中，指定具体域名，否则回源会失败。

源站和回源HOST的区别：

- 源站：源站决定了回源时请求到的具体IP。
- 回源HOST：回源HOST决定了回源请求访问到该IP上的具体站点。

回源HOST的默认值为：

- 在您源站类型是IP的情况下，您的回源HOST类型默认为加速域名。
- 在您源站类型是OSS域名的情况下，您的回源HOST类型默认为源站域名。

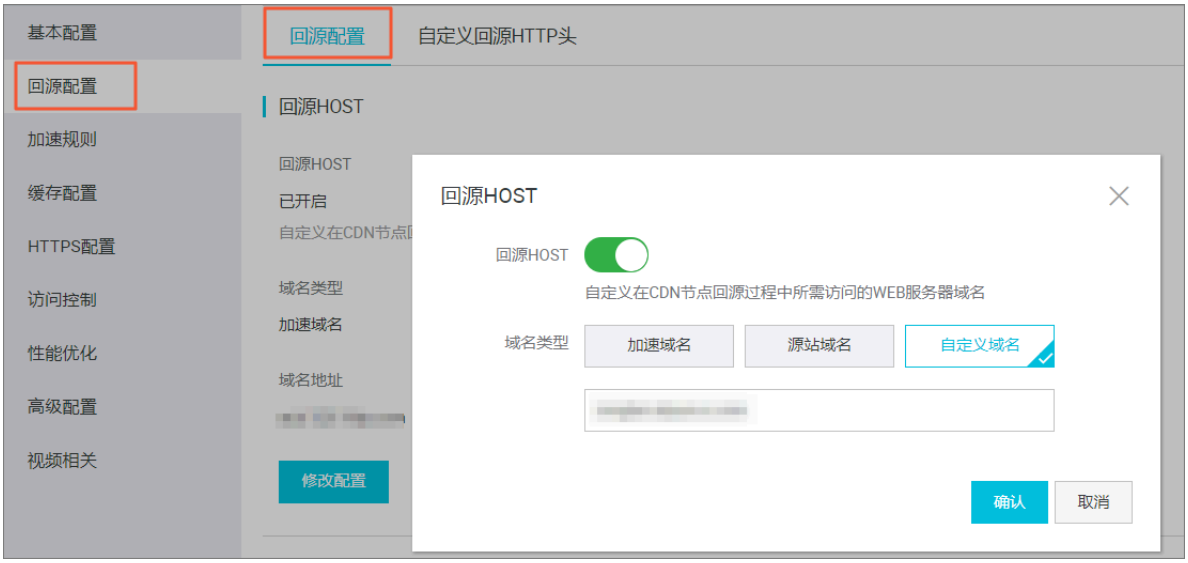
示例：

- 在您的源站是域名源站`www.a.com`的情况下，您选择将回源HOST设置为`www.b.com`，则实际回源的是`www.a.com`解析到的IP站点`www.b.com`。
- 在您的源站是IP源站`1.1.1.1`的情况下，您选择将回源HOST设置为`www.b.com`，则实际回源的是`1.1.1.1`对应的主机上的站点`www.b.com`。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 单击回源配置。
5. 在回源HOST区域框，单击修改配置。

6. 打开回源HOST开关，选择域名类型，单击确认，配置成功。



7.2 配置协议跟随回源

本文档介绍了什么是协议跟随回源，开启该功能后，您可以按照您设定的协议规则进行回源。

背景信息

协议跟随回源，指回源使用的协议和客户端访问资源的协议保持一致，即如果客户端使用HTTPS方式请求资源，当节点上未缓存该资源时，会使用相同的HTTPS方式回源获取资源。同理，如果客户端使用HTTP协议，CDN节点也将使用HTTP协议回源。

> **说明:**
> 源站需要同时支持80端口和443端口，否则有可能会造成回源失败。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 单击回源配置，在协议跟随回源区域框中，打开协议跟随回源开关。

5. 单击修改配置，您可以选择的回源协议类型为：跟随、HTTP或HTTPS。

- 跟随：客户端以HTTP或HTTPS协议请求CDN，CDN跟随客户端的协议请求源站。
- HTTP：CDN只以HTTP协议回源。
- HTTPS：CDN只以HTTPS协议回源。

6. 单击确认，配置成功。

7.3 开启私有Bucket回源授权

本文档介绍了如何开通加速域名访问私有bucket资源内容的权限。您可以通过RAM（Resource Access Management）控制台，取消对应角色名称的授权，关闭私有Bucket回源功能。

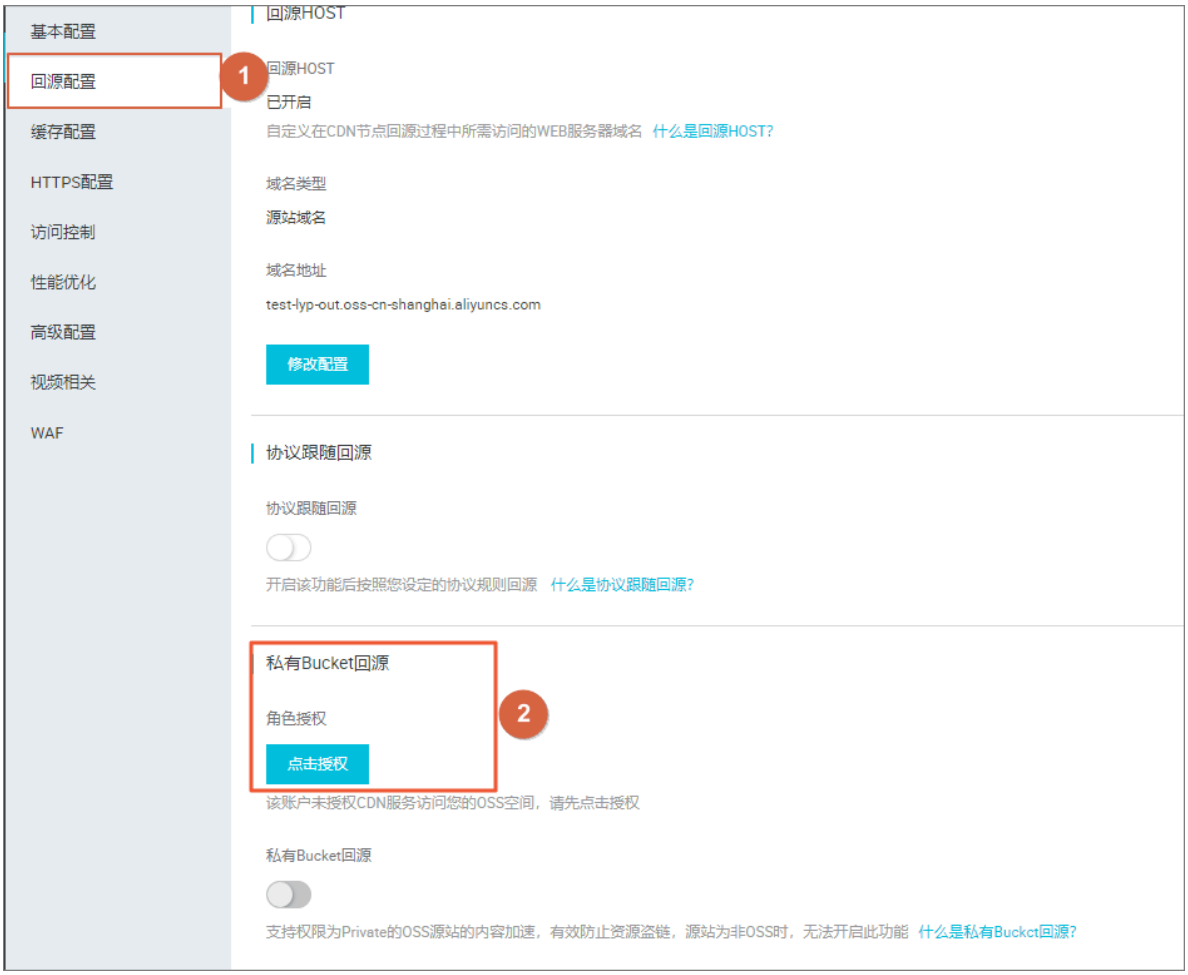
背景信息

您可以配合使用阿里云CDN提供的Refer防盗链功能、鉴权功能，有效保护您的资源安全。详细说明，请参见配置防盗链和配置URL鉴权。

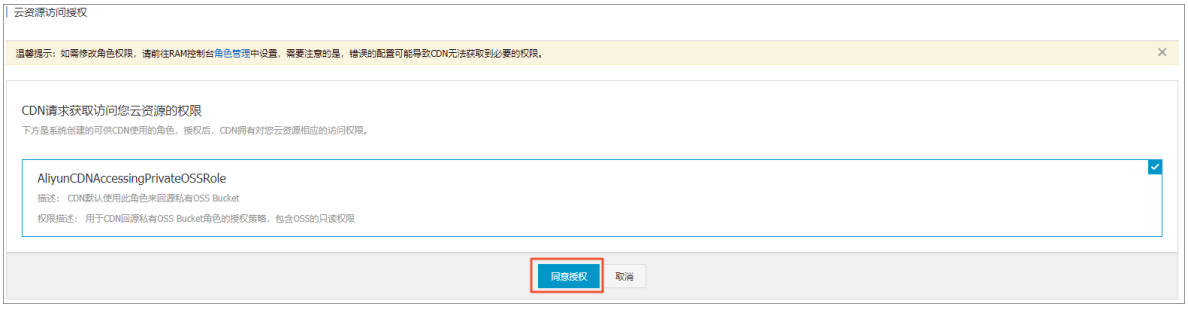
> 注意:
>
> - 仅支持源站类型为OSS域名的加速域名开启私有Bucket回源功能。
> - 进行一次回源授权，即授权CDN对您所有Bucket的只读权限，不只是对当前Bucket授权。
> - 授权成功并开启了对应域名的私有Bucket回源授权功能，该加速域名可以访问您的私有bucket内的资源内容。开启该功能前，请根据实际的业务情况，谨慎决策。如果您授权的私有Bucket内容并不适合作为CDN加速域名的回源内容，请勿授权或者开启此功能。
> - 如果您的网站有被攻击风险：
>   - 请购买高防服务。
>   - 请勿授权或开启私有Bucket回源授权功能。

操作步骤

1. 登录CDN控制台。
2. 在左侧导航栏，单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 在左侧导航栏，单击回源配置。
5. 在私有Bucket回源区域框中，单击点击授权。



6. 单击同意授权。



7. 在私有Bucket回源区域框中，打开私有Bucket回源开关。

   了解如何关闭私有Bucket，请参见关闭私有Bucket回源。

7.4 关闭私有Bucket回源授权

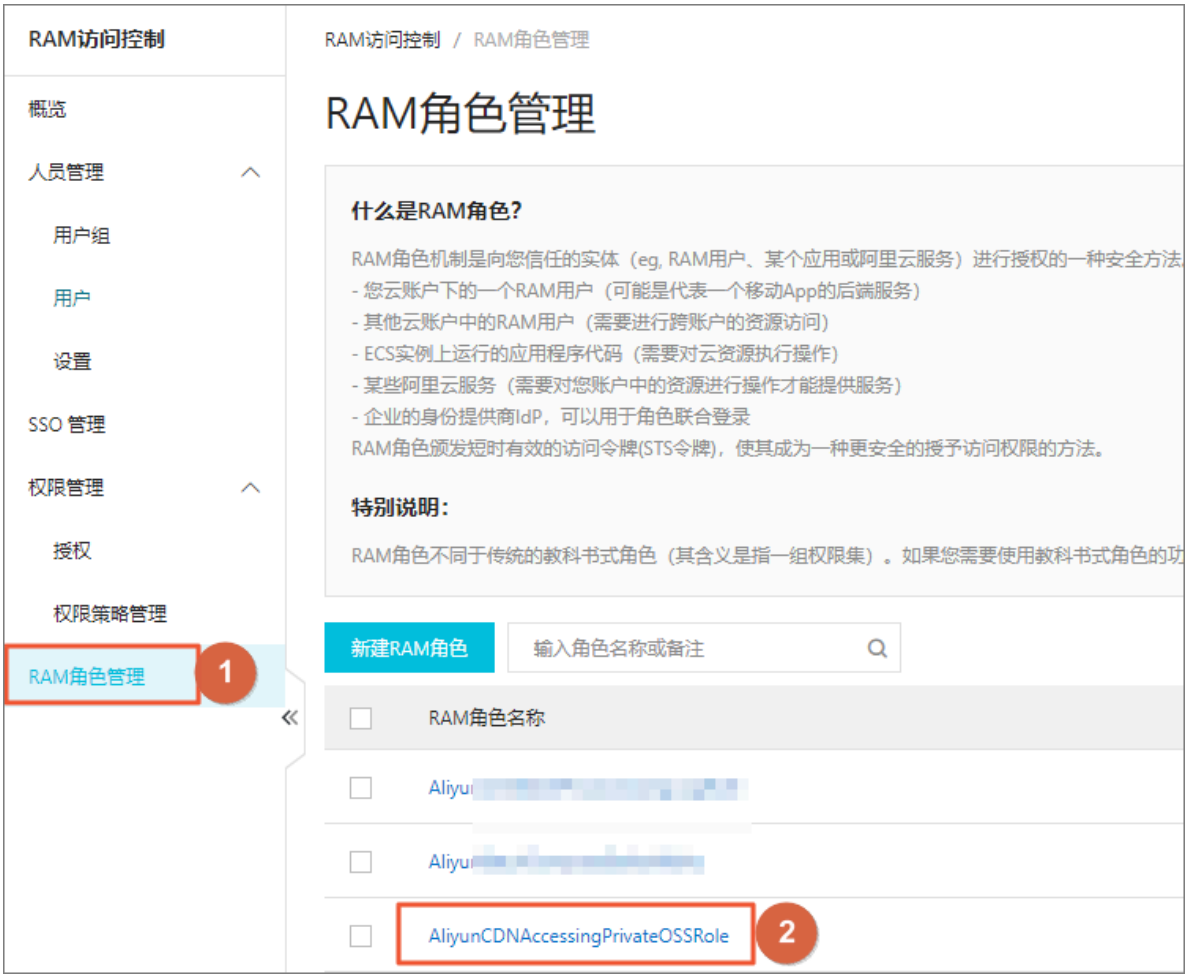
本文档介绍了如何移除加速域名能够访问您私有bucket内资源内容的权限。您可以通过RAM（Resource Access Management）控制台，取消对应角色名称的授权，关闭私有Bucket回源功能。

背景信息

> **说明:**
> 若您的加速域名正在使用私有bucket作为源站进行回源，请不要关闭或删除私有bucket授权。

操作步骤

1. 登录RAM控制台。
2. 在左侧导航栏，单击RAM角色管理。
3. 在RAM角色管理页面，单击RAM角色名称AliyunCDNAccessingPrivateOSSRole。



4. 单击待删除权限对应的移除权限。

   在移除权限确认对话框中，单击确认。

5. 返回RAM角色管理页面，单击待删除角色对应的删除。

   在删除RAM角色确认对话框中，单击确认。

7.5 配置回源SNI

如果您的源站IP绑定了多个域名，当CDN节点以HTTPS协议访问您的源站时，您可以设置回源SNI，指明具体访问域名。

背景信息

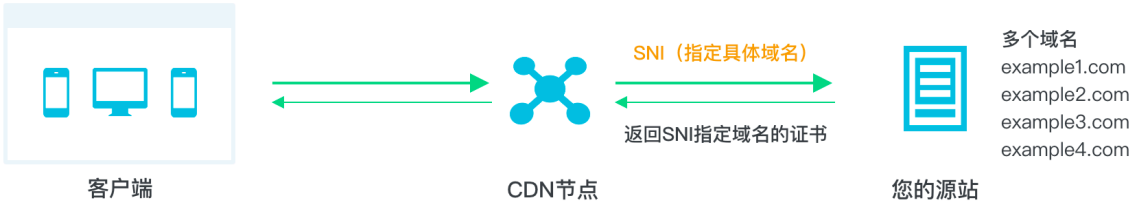
服务器名称指示 Server Name Indication（SNI）是一个扩展的传输层安全性协议 Transport Layer Security（TLS）。在该协议下，握手过程开始时，客户端会告诉它正在连接的那台服务器即将要连接的主机名称，以允许该服务器在相同的IP地址和TCP端口号上呈现多个证书，即一台服务器可以为多个域名提供服务。因此，同一个IP地址上提供的多个安全的HTTPS网站（或其他任何基于TLS的服务），不需要使用相同的证书。

如果您的源站服务器使用单个IP提供多个域名的HTTPS服务，且您已经为CDN设置了443端口回源（CDN节点以HTTPS协议访问您的服务器），您就需要设置回源SNI，指明所请求的具体域名。这样CDN节点以HTTPS协议回源访问您的服务器时，服务器才会正确地返回对应的证书。

> **说明:**
> 如果您的源站是阿里云OSS，则无需设置回源SNI。

工作原理

回源SNI的工作原理如下图所示：

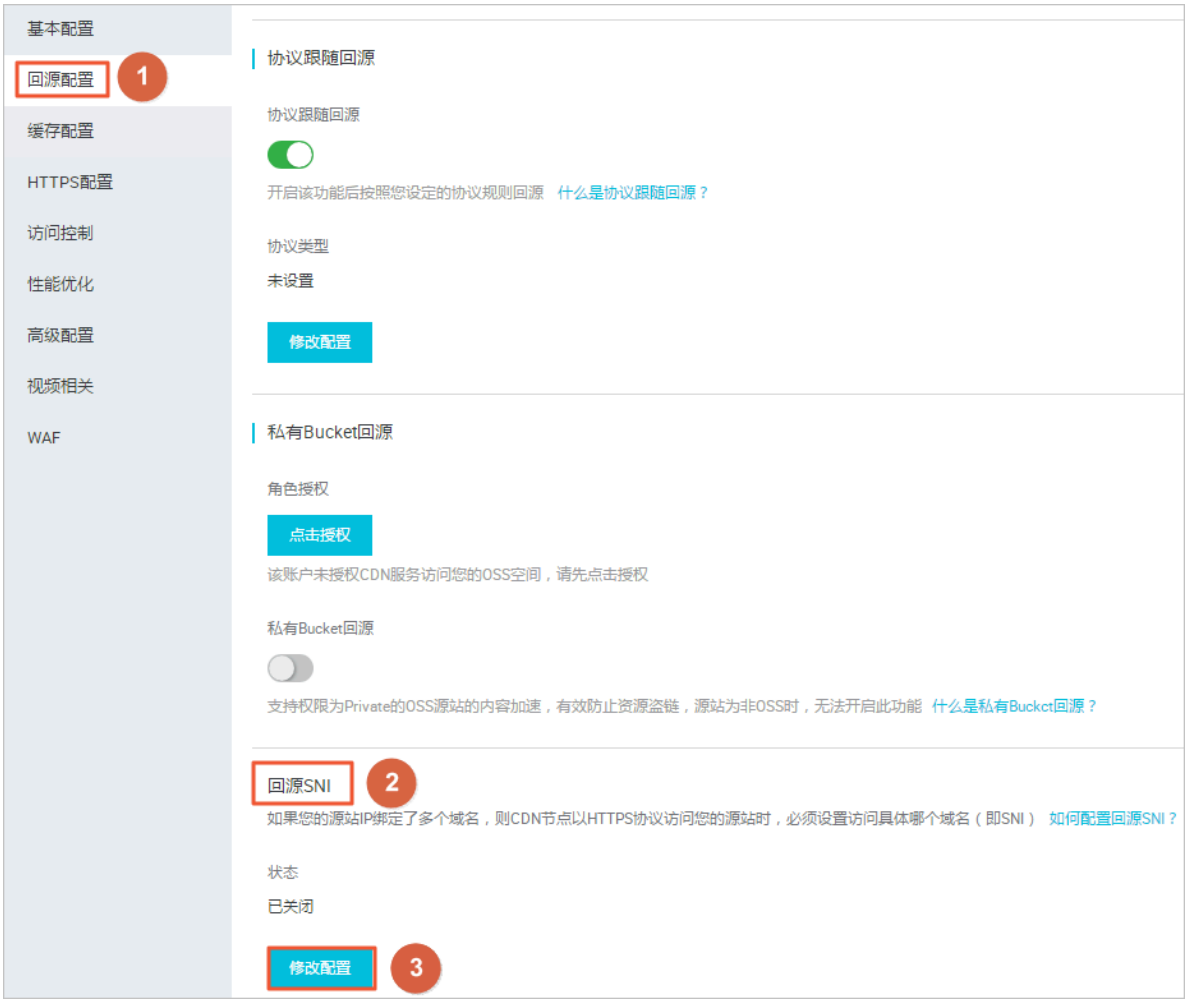


1. CDN节点以HTTPS协议访问源站时，在SNI中指定访问的域名。
2. 源站接收到请求后，根据SNI中记录的域名，返回对应域名的证书。
3. CDN节点收到证书，与服务器端建立安全连接。

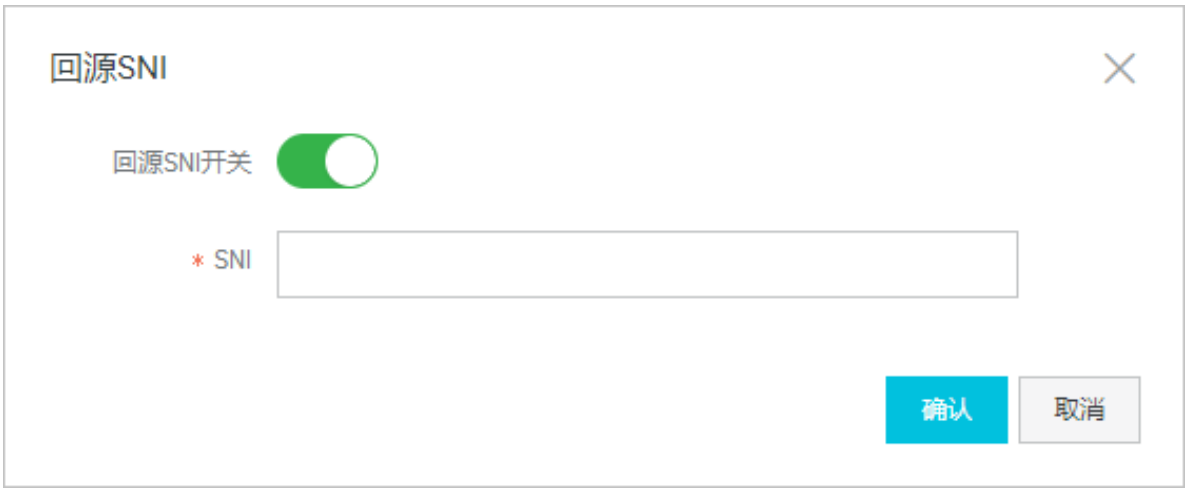
操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 单击回源配置。

5. 在回源SNI区域框，单击修改配置。



6. 打开回源SNI开关，填入您服务器源站提供服务的具体域名，单击确认，完成配置。



7.6 配置自定义回源HTTP头

HTTP请求回源时，您可以添加或删除回源HTTP头。

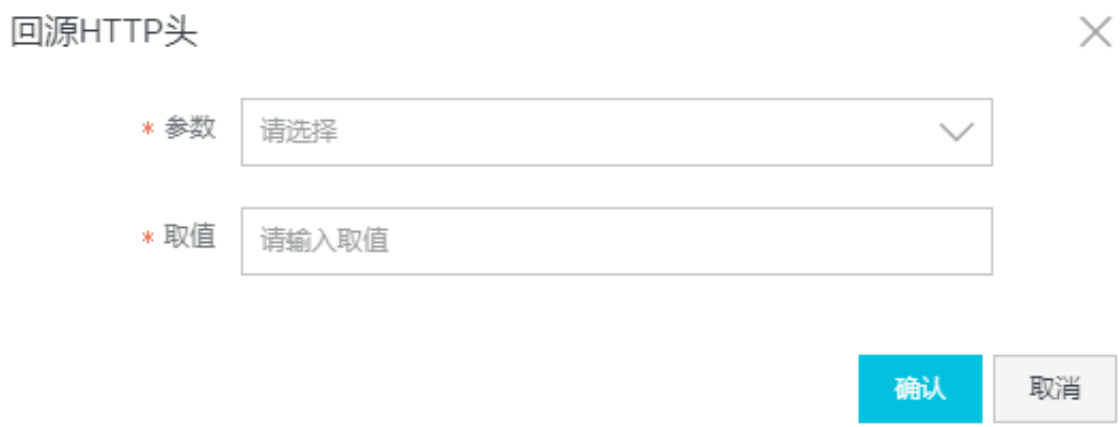
背景信息

HTTP消息头是指，在超文本传输协议（Hypertext Transfer Protocol，HTTP）的请求和响应消息中，协议头部的组件。HTTP消息头准确描述了正在获取的资源、服务器或客户端的行为，定义了HTTP事务中的具体操作参数。

在HTTP消息头中，按其出现的上下文环境，分为通用头、请求头、响应头等。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 单击回源配置。
5. 单击自定义回源HTTP头。
6. 单击添加。
7. 在回源HTTP头页面，选择回源参数，并设置取值，单击确认。



8 缓存配置

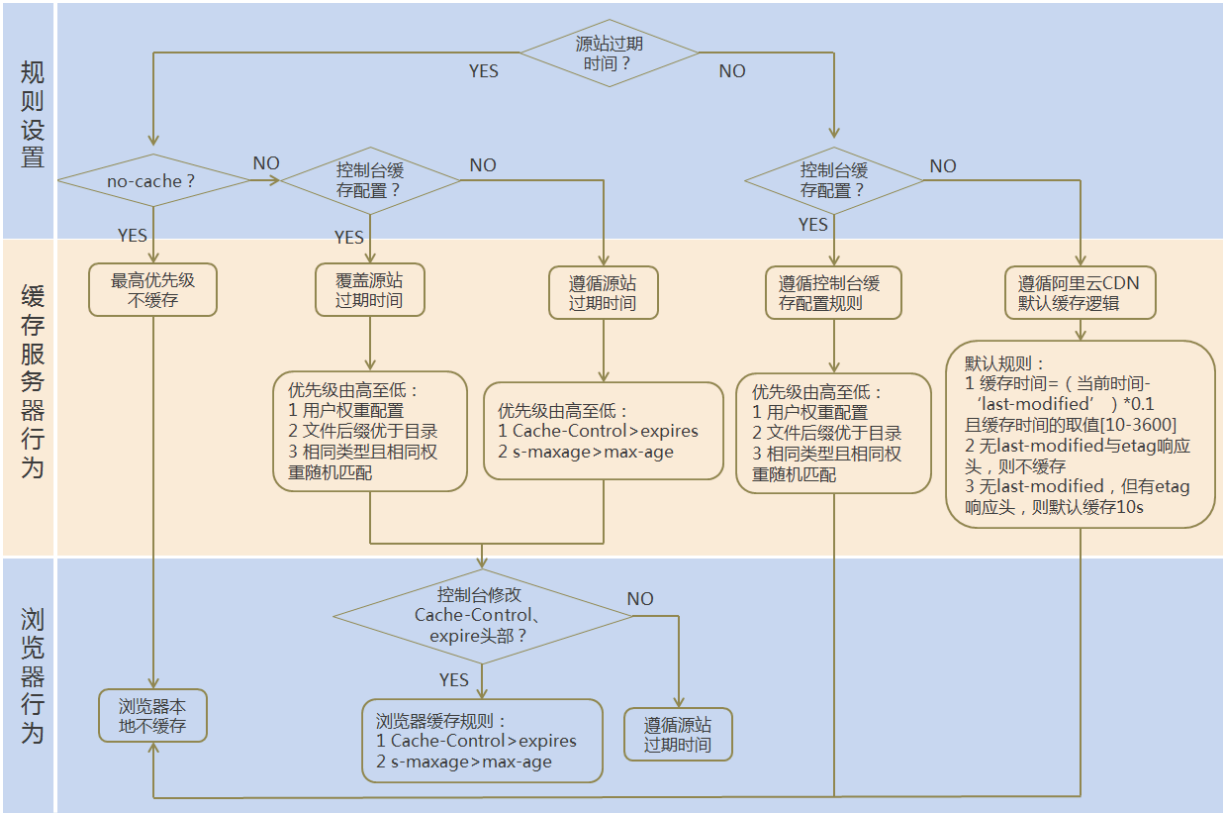
8.1 配置缓存过期时间

通过本文档，您可以针对静态资源配置指定目录和文件后缀名的缓存过期时间，以及优先级，使其在CDN上按照缓存规则进行缓存。

背景信息

配置静态资源的缓存过期时间之前，建议您源站的内容不使用同名更新，以版本号的方式同步，即采用*img-v1.0.jpg*、*img-v2.1.jpg*的命名方式。

CDN节点上缓存资源的缓存策略流程图如下。



说明:

- Cache的默认缓存策略用于配置文件过期时间，在此配置的优先级高于源站配置。如果源站未配置cache，则支持按目录、文件后缀两种方式设置（支持设置完整路径缓存策略）。
- CDN节点上缓存的资源，可能由于热度较低而被提前从节点删除。

操作步骤

1. 登录CDN控制台。

2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 单击缓存配置。
5. 在缓存过期时间页签，单击添加。
6. 配置缓存规则，您可以选择按目录或文件后缀名进行配置。

配置项	说明
类型	· 目录：指定路径下的缓存资源。 · 文件后缀名：指定文件类型的缓存资源。
地址	· 添加单条目录（支持完整路径）时，须以“/”开头，如/directory/aaa。 · 添加多个文件后缀名时，须以半角逗号分隔，如jpg,txt。
过期时间	资源对应的缓存时间。过期时间最多设置为3年，建议您参照以下规则进行配置： · 对于不经常更新的静态文件（如图片类型、应用下载类型等），建议您将缓存时间设置为1个月以上。 · 对于频繁更新的静态文件（如js、css等），您可以根据实际业务情况设置。 · 对于动态文件（如php、jsp、asp等），建议您将缓存时间设置为0s，即不缓存。

配置项	说明
权重	缓存规则的优先级。 **说明:** · 取值范围：1 ~ 99间的整数。数字越大，优先级越高，优先生效。 · 不推荐设置相同的权重，权重相同的两条缓存策略优先级随机。 示例：为加速域名`example.aliyun.com`配置三条缓存策略，缓存策略1优先生效。 · 缓存策略1：文件名后缀为jpg、png的所有资源过期时间设置为1月，权重设置为90。 · 缓存策略2：目录为*/www/dir/aaa*过期时间设置为1小时，权重设置为70。 · 缓存策略3：完整路径为*/www/dir/aaa/example.php*过期时间设置为0s，权重设置为80。

7. 单击确认。

   您也可以单击修改或删除，对当前配置的缓存策略进行相应操作。

8.2 配置状态码过期时间

通过本文档，您可以配置资源的指定目录或文件后缀名的状态码过期时间。

背景信息

您在设置状态码过期时间时，请注意：

- 对于状态码303、304、401、407、600和601，不进行缓存。
- 对于状态码204、305、400、403、404、405、414、500、501、502、503和504，如果源站响应了Cache-Control，则遵循源站的Cache-Control原则。如果未设置状态码，则缓存时间默认(negative_ttl)为一秒。
- 如果您同时设置了目录和文件后缀名这两种类型的状态码过期时间，那么先设置的类型生效。

操作步骤

1. 登录CDN控制台。

2. 在左侧导航栏，单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 在左侧导航栏，单击缓存配置。
5. 在状态码过期时间页签，单击添加，增加状态码的缓存策略。

6. 在状态码过期时间对话框，选择类型并进行相关设置。

类型	注意事项
目录	· 添加单条目录（支持完整路径）须以/开头，如/directory/aaa。 · 不支持配置状态码2xx和3xx。

类型	注意事项
文件后缀名	· 输入多个文件后缀名，须以半角逗号分隔，如txt,jpg。 · 不支持*匹配所有类型文件。 · 不支持配置状态码2xx和3xx。

7. 单击确认，配置成功。

您也可以单击修改或删除，对当前状态码过期时间的配置进行相应操作。

8.3 配置HTTP响应头

通过本文档，您可以配置资源缓存过期的HTTP消息头。

背景信息

HTTP消息头是指，在超文本传输协议（Hypertext Transfer Protocol，HTTP）的请求和响应消息中，协议头部的组件。HTTP消息头准确描述了正在获取的资源、服务器或客户端的行为，定义了HTTP事务中的具体操作参数。

在HTTP消息头中，按其出现的上下文环境，分为通用头、请求头、响应头等。目前阿里云提供10个HTTP响应头参数可供您自行定义取值，参数解释如下：

参数	描述
Content-Type	指定客户端程序响应对象的内容类型。
Cache-Control	指定客户端程序请求和响应遵循的缓存机制。
Content-Disposition	指定客户端程序把请求所得的内容存为一个文件时提供的默认的文件名。
Content-Language	指定客户端程序响应对象的语言。
Expires	指定客户端程序响应对象的过期时间。
Access-Control-Allow-Origin	指定允许的跨域请求的来源。
Access-Control-Allow-Headers	指定允许的跨域请求的字段。
Access-Control-Allow-Methods	指定允许的跨域请求方法。
Access-Control-Max-Age	指定客户端程序对特定资源的预取请求返回结果的缓存时间。
Access-Control-Expose-Headers	指定允许访问的自定义头信息。

配置HTTP响应头时，注意事项如下：

- HTTP响应头的设置会影响该加速域名下所有资源客户端程序（例如浏览器）的响应行为，但不会影响缓存服务器的行为。
- 目前仅支持上述HTTP头参数取值设置。如果您有其他HTTP头设置需求，请提交工单反馈。
- 关于参数Access-Control-Allow-Origin的取值，您可以填写`*`表示全部域名；也可以填写完整域名，例如`www.aliyun.com`。
- 目前不支持泛域名设置。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 单击目标域名后的管理。
4. 单击缓存配置 > HTTP头。
5. 单击添加，选择参数，并输入取值。

6. 单击确认，配置成功。

   您也可以单击修改或删除，对当前配置的缓存策略进行相应操作。

8.4 自定义错误页面

当客户端通过浏览器请求Web服务时，如果请求的URL不存在，则Web服务默认会返回404报错页面。Web服务器预设的报错页面通常不美观，为了提升访问者体验，您可以根据所需自定义HTTP或者HTTPS响应返回码跳转的完整URL地址。通过本文，您可以了解自定义错误页面的操作方法。

背景信息

阿里云提供两种状态码返回页面，分别是默认页面和自定义页面。以返回码404为例，介绍默认页面和自定义页面的差异。

- 默认值：http响应返回404时，服务器返回默认404 Not Found页面。
- 自定义404：http响应返回404时，将会跳转到自定义的404页面，需要自定义跳转页的完整URL地址。

> 说明:
>
> - 404页面属于阿里云公益资源，不会产生任何费用。
> - 自定义页面属于个人资源，按照正常分发计费。

操作步骤

1. 登录CDN控制台。
2. 在左侧导航栏，单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 在左侧导航栏，单击缓存配置。
5. 在自定义页面页签，单击添加，增加自定义返回码的页面内容。

本文以自定义错误码404为例，假设您需要将404页面资源`error404.html`，与其他静态文件一样存储到源站域名下，并通过加速域名`exp.aliyun.com`访问。那么，您只需选择404并填写完整的加速域名URL即可，URL为：`http://exp.aliyun.com/error404.html`。

6. 单击确认。

您也可以单击修改或删除，对当前配置进行相应操作。

8.5 配置重写

通过本文档，您可以对请求的URI进行修改和302重定向至目标URI。重新功能可以配置多条rewrite匹配规则。

背景信息

如果您需要对请求URI进行修改，请添加重写功能。例如：您的某些用户或者客户端仍然使用http协议访问http://example.com，您可以通过该功能配置，所有http://example.com请求都重定向到https://example.com。

执行规则说明：

- Redirect：若请求的URI匹配了当前规则，该请求将被302重定向跳转到目标URI。
- Break：若请求的URI匹配了当前规则，执行完当前规则后，将不再匹配剩余规则。

操作步骤

1. 登录CDN控制台。
2. 在左侧导航栏，单击域名管理。
3. 在您需要设置的域名，单击管理。
4. 在左侧导航栏，单击缓存配置。
5. 在重写区域框中，单击添加。
6. 根据您的需求进行配置，选择Redirect或Break，单击确定。

样例	待重写URI	目标URI	执行规则	结果说明
样例一	/hello	/index.html	Redirect	客户端请求`http://domain.com/hello`，CDN节点将返回302让客户端重新请求`http://domain.com/index.html`的内容。

样例	待重写URI	目标URI	执行规则	结果说明
样例二	^/hello$	/index.html	Break	客户端请求`http://domain.com/hello`，CDN节点将返回`http://domain.com/index.html`的内容。且该请求不再继续匹配其余的重写规则。
样例三	^/$	/index.html	Redirect	客户端请求`http://domain.com`，CDN节点将返回302让客户端重新请求`http://domain.com/index.html`的内容。

9 HTTPS安全加速

9.1 什么是HTTPS加速

本文档介绍了HTTPS安全加速的工作原理、优势和注意事项。您可以通过开启HTTPS安全加速，实现客户端和CDN节点之间请求的HTTPS加密，保障数据传输的安全性。

什么是HTTPS?

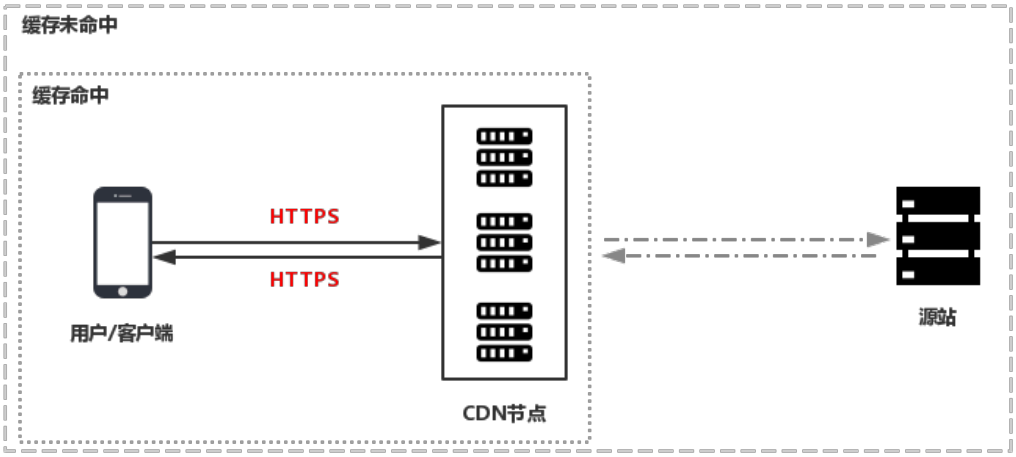
HTTP协议以明文方式发送内容，不提供任何方式的数据加密。HTTPS协议是以安全为目标的HTTP通道，简单来说，HTTPS是HTTP的安全版，即将HTTP用SSL/TLS协议进行封装，HTTPS的安全基础是SSL/TLS协议。HTTPS提供了身份验证与加密通讯方法，被广泛用于万维网上安全敏感的通讯，例如交易支付。

根据2017年EFF（Electronic Frontier Foundation）发布的报告，目前全球已有超过一半的网页端流量采用了加密的HTTPS进行传输。

工作原理

在阿里云CDN控制台开启的HTTPS协议，将实现客户端和阿里云CDN节点之间请求的HTTPS加密。CDN节点返回从源站获取的资源给客户端时，按照源站的配置方式进行。建议源站配置并开启HTTPS，实现全链路的HTTPS加密。

HTTPS加密流程如下：



1. 客户端发起HTTPS请求。
2. 服务端生成公钥和私钥（可以自己制作，也可以向专业组织申请）。

3. 服务端把相应的公钥证书传送给客户端。
4. 客户端解析证书的正确性。
   - 如果证书正确，则会生成一个随机数（密钥），并用公钥随机数进行加密，传输给服务端。
   - 如果证书不正确，则SSL握手失败。

   
   说明:
   正确性包括：证书未过期、发行服务器证书的CA可靠、发行者证书的公钥能够正确解开服务器证书的发行者的数字签名、服务器证书上的域名和服务器的实际域名相匹配。
5. 服务端用之前的私钥进行解密，得到随机数（密钥）。
6. 服务端用密钥对传输的数据进行加密。
7. 客户端用密钥对服务端的加密数据进行解密，拿到相应的数据。

功能优势

- HTTP明文传输，存在各类安全风险：
  - 窃听风险：第三方可以获知通信内容。
  - 篡改风险：第三方可以修改通信内容。
  - 冒充风险：第三方可以冒充他人身份参与通信。
  - 劫持风险：包括流量劫持、链路劫持、DNS劫持等。
- HTTPS安全传输的优势：
  - 数据传输过程中对您的关键信息进行加密，防止类似Session ID或者Cookie内容被攻击者捕获造成的敏感信息泄露等安全隐患。
  - 数据传输过程中对数据进行完整性校验，防止DNS或内容遭第三方劫持、篡改等中间人攻击（MITM）隐患，详情请参见使用HTTPS防止流量劫持。
  - HTTPS是主流趋势：未来主流浏览器会将HTTP协议标识为不安全，谷歌浏览器Chrome 70以上版本以及Firefox已经在2018年将HTTP网站标识为不安全，若坚持使用HTTP协议，除了安全会埋下隐患外，终端客户在访问网站时出现的不安全标识，也将影响访问。
  - 百度与Google均对HTTPS网站进行搜索加权，主流浏览器均支持HTTP/2，而支持HTTP/2必须支持HTTPS。可以看出来，无论从安全，市场，还是用户体验来看，普及HTTPS是未来的一个方向，所以强烈建议您将访问协议升级到HTTPS。

应用场景

主要将应用场景分为以下五类：

- 企业应用：若网站内容包含crm、erp等信息，这些信息属于企业级的机密信息，若在访问过程中被劫持或拦截窃取，对企业是灾难级的影响。

- 政务信息：政务网站的信息具备权威性，正确性等特征，需预防钓鱼欺诈网站和信息劫持，避免出现信息劫持或泄露引起社会公共的信任危机。
- 支付体系：支付过程中，涉及到敏感信息如姓名，电话等，防止信息劫持和伪装欺诈，需启用HTTPS加密传输，避免出现下单后，下单客户会立即收到姓名、地址、下单内容，然后以卡单等理由要求客户按指示重新付款之类诈骗信息，造成客户和企业的双重损失。
- API接口：保护敏感信息或重要操作指令的传输，避免核心信息在传输过程中被劫持。
- 企业网站：激活绿色安全标识(DV/OV)或地址栏企业名称标识(EV)，为潜在客户带来更可信、更放心的访问体验。

注意事项

分类	注意事项
配置	· 支持开启HTTPS安全加速功能的业务类型包括： - 图片小文件，主要适用于各种门户网站、电子商务类网站、新闻资讯类站点或应用、政府或企业官网站点、娱乐游戏类站点或应用等。 - 大文件下载，主要适用于下载类站点和音视频的应用。 - 视音频点播，主要适用于各类视音频站点，如影视类视频网站、在线教育类视频网站、新闻类视频站点、短视频社交类网站以及音频类相关站点和应用。 - 直播流媒体，主要适用于交互性在线教育网站、游戏竞技类直播站点、个人秀场直播、事件类和垂直行业的直播平台等。 - 全站加速，主要适用于电商、社交、政企、游戏和金融平台。 · 支持泛域名HTTPS服务。 · 支持HTTPS安全加速的启用和停用。 - 启用：您可以修改证书，系统默认兼容HTTP和HTTPS请求。您也可以配置强制跳转，自定义原请求方式。 - 停用：停用后，系统不再支持HTTPS请求且不再保留证书或私钥信息。再次开启证书，需要重新上传证书或私钥。详细说明，请参见配置HTTPS证书。 · 您可以查看证书，但由于私钥信息敏感，不支持私钥查看。请妥善保管证书相关信息。 · 您可以更新证书，但请谨慎操作。更新HTTPS证书后1分钟内全网生效。
计费	HTTPS安全加速属于增值服务，开启后将产生HTTPS请求数计费，详细计费标准请参见静态HTTPS请求数。  说明: HTTPS根据请求数单独计费，费用不包含在CDN流量包内。请确保账户余额充足再开通HTTPS服务，以免因HTTPS服务欠费影响您的CDN服务。

分类	注意事项
证书	· 开启HTTPS安全加速功能的加速域名，您需要上传格式均为PEM的证书和私钥。 说明: 由于CDN采用的Tengine服务基于Nginx，因此只支持Nginx能读取的PEM格式的证书。详细说明，请参见证书格式说明。 · 上传的证书需要和私钥匹配，否则会校验出错。 · 不支持带密码的私钥。 · 只支持携带SNI信息的SSL/TLS握手。 其他证书相关的常见问题，请参见更多证书问题。

相关功能

为了数据传输的安全，您可以根据实际业务需求，配置以下功能：

- 配置HTTPS证书，实现HTTPS安全加速。
- 设置HTTP/2，HTTP/2是最新的HTTP协议，Chrome、 IE11、Safari以及Firefox等主流浏览器已经支持HTTP/2协议。
- 设置强制跳转，强制重定向终端用户的原请求方式。
- 设置TLS，保障您互联网通信的安全性和数据完整性。
- 设置HSTS，强制客户端（如浏览器）使用HTTPS与服务器创建连接，降低第一次访问被劫持的风险。

9.2 证书格式说明

您需要配置HTTPS证书，才能使用HTTPS方式访问资源，实现HTTPS安全加速。本文档介绍了阿里云CDN支持的证书格式和不同证书格式的转换方式。

Root CA机构颁发的证书

Root CA机构提供的证书是唯一的，一般包括Apache、IIS、Nginx和Tomcat。阿里云CDN使用的证书是Nginx，.crt为证书，.key为私钥。

证书规则为：

- 请将开头-----BEGIN CERTIFICATE-----和结尾-----END CERTIFICATE-----一并上传。
- 每行64字符，最后一行不超过64字符。

Linux环境下，PEM格式的证书示例如下：

```
-----BEGIN CERTIFICATE-----
MIIE+TCCA+GgAwIBAgIQU306HIX4KsioTW1s2A2krTANBgkqhkiG9w0BAQUFADCB
tTELMAkGA1UEBhMCVVMxFzAVBgNVBAoTDlZlcmlTaWduLCBJbmMuMR8wHQYDVQQL
ExZWZXJpU2lnbiBUcnVzdCBOZXR3b3JrMTswOQYDVQQLEzJUZXJtcyBvZiB1c2Ug
YXQgaHR0cHM6Ly93d3cudmVyaXNpZ24uY29tL3JwYSoAYykwOTEvMC0GA1UEAxMm
VmVyaVNpZ24gQ2xhc3MgMyBTZWN1cmUgU2VydmVyIENBIC0gRzIwHhcNMTAxMDA4
MDAwMDAwWhcNMTMxMDA3MjM1OTU5WjBqMQswCQYDVQQGEwJVUzETMBEGA1UECBMK
V2FzaGluZ3RvbjEQMA4GA1UEBxQHU2VhdHRsZTEYMBYGA1UEChQPQW1hem9uLmNv
bSBJbmMuMRowGAYDVQQDFBFpYW0uYW1hem9uYXdzLmNvbTCBnzANBgkqhkiG9w0B
AQEFAAOBjQAwgYkCgYEA3Xb0EGea2dB8QGEUwLcEpwvGawEkUdLZmGL1rQJZdeeN
3vaF+ZTm8Qw5Adk2Gr/RwYXtpx04xvQXmNm+9YmksHmCZdruCrW1eN/P9wBfqMMZ
X964CjVov3NrF5AuxU8jgtw0yu//C3hWn0uIVGdg76626gg0oJSaj48R2n0MnVcC
AwEAAaOCAdEwggHNMAkGA1UdEwQCMAAwCwYDVR0PBAQDAgWgMEUGA1UdHwQ+MDww
OqA4oDaGNGh0dHA6Ly9TVlJTZWN1cmUtRzItY3JsLnZlcmlzaWduLmNvbS9TVlJT
ZWN1cmVHMi5jcmwwRAYDVR0gBD0wOzA5BgtghkgBhvhFAQcXAzAqMCgGCCsGAQUF
BwIBFhxodHRwczovL3d3dy52ZXJpc2lnbi5jb20vcnBhMB0GA1UdJQQWMBQGCCsG
AQUFBwMBBggrBgEFBQcDAjAfBgNVHSMEGDAWgBSl7wsRzsBBA6NKZZBIshzgVy19
RzB2BggrBgEFBQcBAQRqMGgwJAYIKwYBBQUHMAGGGGh0dHA6Ly9vY3NwLnZlcmlz
aWduLmNvbTBABggrBgEFBQcwAoY0aHR0cDovL1NWUlNlY3VyZS1HMi1haWEudmVy
aXNpZ24uY29tL1NWUlNlY3VyZUcyLmNlcjBuBggrBgEFBQcBDARiMGChXqBcMFow
WDBWFglpbWFnZS9naWYwITAfMAcGBSsOAwIaBBRLa7kolgYMu9BSOJsprEsHiyEF
GDAmFiRodHRwOi8vbG9nby52ZXJpc2lnbi5jb20vdnNsb2dvMS5naWYwDQYJKoZI
hvcNAQEFBQADggEBALpFBXeG782QsTtGwEE9zBcVCuKjrsl3dWK1dFiq3OP4y/Bi
ZBYEywBt8zNuYFUE25Ub/zmvmpe7p0G76tmQ8bRp/4qkJoiSesHJvFgJ1mksr3IQ
3gaE1aN2BSUIHxGLn9N4F09hYwwbeEZaCxfgBiLdEIodNwzcvGJ+2LlDWGJOGrNI
NM856xjqhJCPxYzk9buuCl1B4Kzu0CTbexz/iEgYV+DiuTxcfA4uhwMDSe0nynbn
1qiwRk450mCOnqH4ly4P4lXo02t4A/DI1I8ZNct/Qfl69a2Lf6vc9rF7BELT0e5Y
R7CKx7fc5xRaeQdyGj/dJevm9BF/mSdnclS5vas=
-----END CERTIFICATE-----
```

中级机构颁发的证书

中级机构颁发的证书文件包含多份证书，您需要将服务器证书与中间证书拼接后，一起上传。

> **说明:**
> 拼接规则为：服务器证书放第一份，中间证书放第二份。一般情况下，机构在颁发证书的时候会有对应说明， 请注意规则说明。

中级机构颁发的证书链：

```
-----BEGIN CERTIFICATE-----

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

-----END CERTIFICATE-----
```

证书链规则：

- 证书之间不能有空行。

- 每一份证书遵守第一点关于证书的格式说明。

RSA私钥格式要求

RSA私钥规则：

- 本地生成私钥：`openssl genrsa -out privateKey.pem 2048`。其中，`privateKey.pem`为您的私钥文件。
- 以`-----BEGIN RSA PRIVATE KEY-----`开头，以`-----END RSA PRIVATE KEY-----`结尾，请将这些内容一并上传。
- 每行64字符，最后一行长度可以不足64字符。

如果您并未按照上述方案生成私钥，得到如`-----BEGIN PRIVATE KEY-----`或`-----END PRIVATE KEY-----`这种样式的私钥时，您可以按照如下方式转换：

```
openssl rsa -in old_server_key.pem -out new_server_key.pem
```

然后将`new_server_key.pem`的内容与证书一起上传。

证书格式转换方式

HTTPS配置只支持PEM格式的证书，其他格式的证书需要转换成PEM格式，建议通过openssl工具进行转换。下面是几种比较流行的证书格式转换为PEM格式的方法。

- DER转换为PEM

  DER格式一般出现在java平台中。

  - 证书转化：

    ```
    openssl x509 -inform der -in certificate.cer -out certificate.pem
    ```

  - 私钥转化：

    ```
    openssl rsa -inform DER -outform pem -in privatekey.der -out
    privatekey.pem
    ```

- P7B转换为PEM

  P7B格式一般出现在windows server和tomcat中。

  - 证书转化：

    ```
    openssl pkcs7 -print_certs -in incertificat.p7b -out outcertifi
    cate.cer
    ```

    获取outcertificat.cer里面-----BEGIN CERTIFICATE-----， -----END CERTIFICATE-----的内容作为证书上传。

  - 私钥转化：P7B证书无私钥，您只需在CDN控制台填写证书部分，私钥无需填写。

- PFX转换为PEM

  PFX格式一般出现在windows server中。

  - 证书转化：

    ```
    openssl pkcs12 -in certname.pfx -nokeys -out cert.pem
    ```

  - 私钥转化：

    ```
    openssl pkcs12 -in certname.pfx -nocerts -out key.pem -nodes
    ```

9.3 配置HTTPS证书

阿里云CDN仅支持PEM格式的证书和私钥，您需要上传HTTPS证书至CDN，开启HTTPS安全加速。本文档介绍了不同类型的HTTPS证书的认证方式和配置方法。

前提条件

配置HTTPS证书前，您需要先购买证书，您可以在云盾控制台快速申请免费的证书或购买高级证书。

背景信息

根据证书认证级别分类如下：

- DV是Domain Validation，仅认证域名所有权，通常是验证域名下指定文件内容，或者验证与域名相关TXT记录，显示明显的安全锁。
- OV是Organization Validation，验证企业组织真实性的标准型SSL证书，比DV SSL证书更安全可信，审核更严格，审核周期也更长。一般多用于电商，教育，游戏等领域。
- EV是Extended Validation，CA/browser forum指定的全球统一标准，通过证书object identifier（OID）来识别，显示完整企业名称，是目前全球最高等级的SSL证书，多用于金融支付，网上银行等领域。

> 说明:
> 目前CDN仅支持PEM格式的证书，如果您的证书不是PEM格式，请进行格式转换，操作方法请参见证书格式转换方式。

操作步骤

1. 登录CDN控制台。
2. 在左侧导航栏，单击域名管理。
3. 在域名管理界面，单击目标域名后的管理。
4. 在左侧导航栏，单击HTTPS配置。
5. 在HTTPS证书界面，单击修改配置。

6. 在HTTPS设置界面，打开HTTPS安全加速开关，配置证书相关参数。

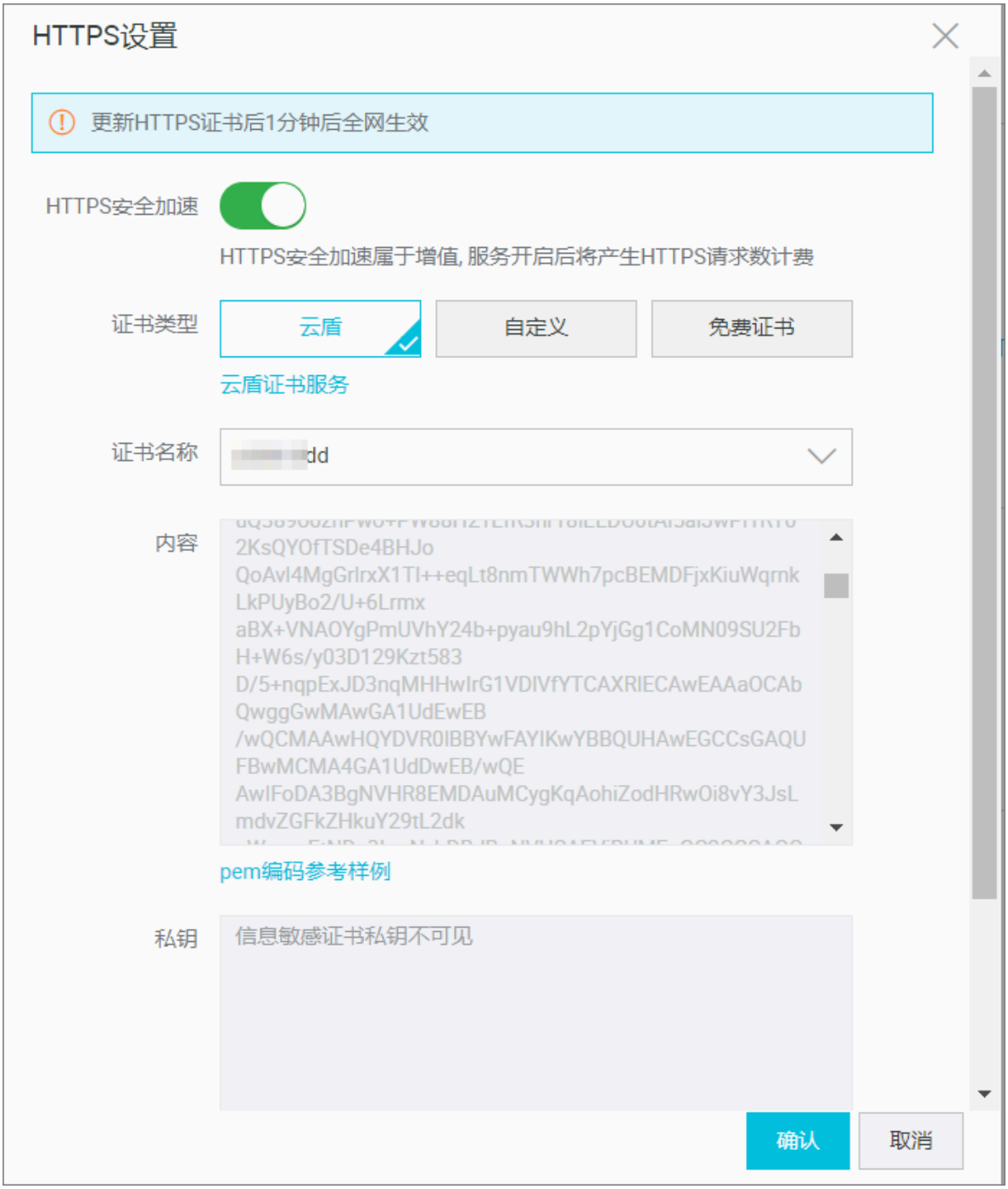
   当您打开HTTPS安全加速开关时，系统弹出确认开启HTTPS界面，该操作单独计费，您可以根据所需选择是否开启。HTTPPS计费标准请参考增值服务计费。

参数	说明
证书类型	· 云盾 您可以在云盾控制台快速申请免费的证书或购买高级证书。 · 自定义 如果证书列表中无

参数	说明
证书名称	当证书类型选择云盾或自定义时，需要配置证书名称。

参数	说明
内容	当证书类型选择自定义时，需要配置该参数。 配置方法请参考内容输入框下方的pem编码参考样例。

参数	说明
私钥	当证书类型选择自定义时，需要配置该参数。 配置方法请参考私钥输入框下方的pem编码参考样例。

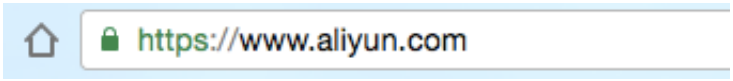


7. 单击确认。

   您可以停用、启用和修改证书。停用证书后，系统将不再保留证书信息。再次开启证书时，需要重新上传证书或私钥。

8. 验证证书是否生效。

更新HTTPS证书1分钟后全网生效，使用HTTPS方式访问资源，如果浏览器中出现绿色HTTPS标识，则HTTPS安全加速生效。



9.4 设置HTTP/2

通过本文档，您可以了解HTTP/2协议的概念、优势和设置方法。

前提条件

开启HTTP/2功能前，您需要确保已经成功配置HTTPS证书，操作方法请参见配置HTTPS证书。

> 说明:
>
> · 如果您是第一次配置HTTPS证书，则需要等证书配置完成且生效后，才能开启HTTP/2。
> · 如果您开启HTTP/2后，关闭了HTTPS证书功能，HTTP/2会自动失效。

背景信息

HTTP/2也被称为HTTP 2.0，是最新的HTTP协议。目前，Chrome、 IE11、Safari和Firefox等主流浏览器已经支持HTTP/2协议。HTTP/2优化了性能，兼容了HTTP/1.1的语义，与SPDY相似，与HTTP/1.1有巨大区别。

HTTP/2的优势：

· 二进制协议：相比于HTTP 1.x基于文本的解析，HTTP/2将所有的传输信息分割为更小的消息和帧，并对它们采用二进制格式编码。基于二进制可以使协议有更多的扩展性，例如，引入帧来传输数据和指令。
· 内容安全：HTTP/2基于HTTPS，具有安全特性。使用HTTP/2特性可以避免单纯使用HTTPS引起的性能下降问题。
· 多路复用（MultiPlexing）：通过该功能，在一条连接上，您的浏览器可以同时发起无数个请求，并且响应可以同时返回。另外，多路复用中支持了流的优先级（Stream dependencies）设置，允许客户端告知服务器最优资源，可以优先传输。
· Header压缩（Header compression）：HTTP请求头带有大量信息，而且每次都要重复发送。HTTP/2采用HPACK格式进行压缩传输，通讯双方各自缓存一份头域索引表，相同的消息头只发送索引号，从而提高效率和速度。

- 服务端推送（Server push）：同SPDY一样，HTTP/2也具有客户端推送功能。目前，大多数网站已经启用HTTP/2，如淘宝。使用浏览器Chrome登录控制台，您可以查看是否启用HTTP/2。

> 说明：
> SPDY是Google开发的基于TCP的应用层协议，用以最小化网络延迟，提升网络速度，优化用户的网络体验。SPDY并不是一种用于替代HTTP的协议，而是对HTTP协议的增强。新协议的功能包括：数据流的多路复用、请求优先级和HTTP报头压缩，与HTTP/2相似。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 单击HTTPS配置。
5. 在HTTP/2设置页签，打开HTTP/2开关，开启该功能。

9.5 配置强制跳转

您可以通过配置强制跳转功能，将客户端至L1的原请求方式强制重定向为HTTP或者HTTPS。

前提条件

配置强制跳转功能前，您需要确保已经成功配置HTTPS证书，操作方法请参见配置HTTPS证书。

操作步骤

1. 登录CDN控制台。
2. 在左侧导航栏，单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 在左侧导航栏，单击HTTPS配置。
5. 在强制跳转区域框，单击修改配置。

6. 在强制跳转对话框，选择跳转类型。

跳转类型	说明
默认	同时支持HTTP和HTTPS方式的请求。
HTTPS -> HTTP	客户端到L1的请求将强制重定向为HTTP方式。

跳转类型	说明
HTTP -> HTTPS	客户端到L1的请求将强制重定向为HTTPS方式，确保访问安全。

以跳转类型为HTTP -> HTTPS为例，介绍强制跳转功能。

当您设置了强制HTTPS跳转后，客户端发起一个HTTP请求，服务端返回301重定向响应，原HTTP请求强制重定向为HTTPS请求，如图所示：

```
$ curl http://[redacted]/' -i
HTTP/1.1 301 Moved Permanently
Server: Tengine
Date: Mon, 03 Jun 2019 13:26:01 GMT
Content-Type: text/html
Content-Length: 278
Connection: keep-alive
Location: https://[redacted]/
Via: cache2.cn201[,0]
Timing-Allow-Origin: *
EagleId: 2a786b0215595683612635433e

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html>
<head><title>301 Moved Permanently</title></head>
<body bgcolor="white">
<h1>301 Moved Permanently</h1>
<p>The requested resource has been assigned a new permanent URI.</p>
<hr/>Powered by Tengine</body>
</html>
```

7. 单击确认。

9.6 配置TLS

为了保障您互联网通信的安全性和数据完整性，阿里云CDN提供TLS版本控制功能。您可以根据不同域名的需求，灵活地配置TLS协议版本。

前提条件

开启TLS功能前，您需要确保已成功配置HTTPS证书。

背景信息

TLS（Transport Layer Security）即安全传输层协议，在两个通信应用程序之间提供保密性和数据完整性。最典型的应用就是HTTPS。HTTPS，即HTTP over TLS，就是安全的HTTP，运行在HTTP层之下，TCP层之上，为HTTP层提供数据加解密服务。

目前，TLS主要有4个版本：

- TLSv1.0：RFC2246，1999年发布，基于SSLv3.0，该版本易受各种攻击（如BEAST和POODLE），除此之外，支持较弱加密，对当今网络连接的安全已失去应有的保护效力。不符合PCI DSS合规判定标准。支持的主流浏览器： IE6+、Chrome 1+、Firefox 2+。
- TLSv1.1：RFC4346，2006年发布，修复TLSv1.0若干漏洞。支持的主流浏览器：IE11+、Chrome22+、Firefox24+、Safri7+。
- TLSv1.2：RFC5246，2008年发布，目前广泛使用的版本。支持的主流浏览器：IE11+、Chrome30+、Firefox27+、Safri7+。
- TLSv1.3：RFC8446，2018年发布，最新的TLS版本，支持0-RTT模式（更快），只支持完全前向安全性密钥交换算法（更安全）。支持的主流浏览器：Chrome 70+和Firefox 63+。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 单击目标域名后的管理。
4. 单击HTTPS配置。

5. 在TLS版本控制区域框，根据您的需要，开启或关闭对应的TLS版本。

说明:

目前TLSv1.0、TLSv1.1和TLSv1.2版本默认开启。

9.7 配置HSTS

通过开启HSTS（HTTP Strict Transport Security）功能，您可以强制客户端（如浏览器）使用HTTPS与服务器创建连接，降低第一次访问被劫持的风险。

前提条件

开启HSTS功能前，您需要确保已经成功配置HTTPS证书。

背景信息

当您网站全站使用HTTPS后，需要将所有HTTP请求301/302重定向到HTTPS。如果您在浏览器输入HTTP链接，或在其他地方单击了HTTP链接，则服务器会将该HTTP请求301/302重定向到 HTTPS。但是这个过程可能被劫持，导致重定向后的请求没有发送到服务器，该问题可以通过HSTS来解决。

HSTS是一个响应头：`Strict-Transport-Security: max-age=expireTime [; includeSubDomains] [; preload]`，各参数说明如下：

- max-age：单位是秒。
- Strict-Transport-Security： 在浏览器缓存的时间，浏览器处理域名的HTTP访问时，若该域名的Strict-Transport-Security没有过期，则在浏览器内部做一次307重定向到HTTPS，从而避免浏览器和服务器之间301/302重定向被劫持的风险。
- includeSubDomains：可选参数。如果指定这个参数，说明这个域名所有子域名也适用上面的规则。
- preload：可选参数，支持preload列表。

 说明:

- HSTS生效前仍然需要第一次301/302重定向到HTTPS。
- HSTS响应头在HTTPS访问的响应中有效，在HTTP访问的响应中无效。
- 仅对443端口有效，对其他端口无效。
- 仅对域名有效，对IP无效。
- 启用HSTS之后，一旦网站证书错误，在缓存时间。

操作步骤

1. 登录CDN控制台。
2. 单击域名管理。
3. 在域名管理页面，单击目标域名后的管理。
4. 单击HTTPS配置。

5. 在HSTS区域框，单击修改配置。

6. 在HSTS设置对话框，打开HSTS开关，配置过期时间和包含子域名。

7. 单击确定。

9.8 常见问题

- CDN开启HTTPS加速后，会有额外收费吗?
- 开启HTTPS加速后，会消耗更多服务资源或者降低访问速度吗?
- 我的站点只有登录才需要HTTPS，其他都不需要HTTPS了?
- 常见的HTTP攻击类型有哪些?

CDN开启HTTPS加速后，会有额外收费吗?

会额外收费。CDN开启HTTPS加速，开启的是客户端到CDN边缘节点这段链路的HTTPS。因为SSL协议的握手、内容解密都需要计算，所以会增加CDN服务器的CPU资源损耗。但是不会增加客户源站的服务器资源损耗，因为CDN边缘节点到客户源站这段链路使用的仍然是HTTP协议，对客户源站没有额外增加损耗。

- 若您购买不同类型的证书，则需要额外付费。

> 说明:
> 您可以直接在CDN控制台申请免费证书，免费证书等级为DV，每个加速域名可以申请一个免费证书，证书有效期为一年，到期后可以免费自动续签。

- 设置好HTTPS证书后，该域名的所有在CDN上的HTTPS请求数会收费，静态HTTPS请求数收费标准为每万次0.05元。

开启HTTPS加速后，会消耗更多服务资源或者降低访问速度吗？

不会消耗更多服务资源，也不会降低访问速度。

您首次访问HTTPS站点比HTTP要慢，因为建立SSL连接需要的时间更长，首次页面加载速度慢了约10%。但是浏览器建立了活跃的keep-alive HTTPS连接后，后续的页面刷新性能和HTTP几乎无差别。

我的站点只有登录才需要HTTPS，其他都不需要HTTPS了？

不是。

- 从安全看，一些页面为HTTP，一些页面为HTTPS，当通过HTTP或不安全的CDN服务加载其他资源(例如JS或CSS文件)时网站也存在用户信息暴露的风险，而全站HTTPS是防止这种风险最简单的方法。
- 从性能看，当网站存在HTTPS和HTTP两种协议时，跳转需对服务器进行了大量的重定向，当这些重定向被触发时会减慢页面加载速度。
- 从全网来看，浏览器对HTTPS的支持会更友好，搜索引擎也对HTTPS的收录有更好的支持。

常见的HTTP攻击类型有哪些？

HTTPS只是安全访问的其中一环，若想要全面的保证网络安全，还需要接入WAF，DDOS等防御能力全面来保证网站安全，以下为常见的HTTP攻击类型：

- SQL注入：它是利用现有应用程序，将（恶意）的SQL命令注入到后台数据库引擎执行的能力，它可以通过在Web表单中输入（恶意）SQL语句得到一个存在安全漏洞的网站上的数据库，而不是按照设计者意图去执行SQL语句。
- 跨站脚本攻击：跨站脚本攻击XSS（Cross-site scripting）是最常见和基本的攻击WEB网站的方法。攻击者在网页上发布包含攻击性代码的数据。当浏览者看到此网页时，特定的脚本就会以浏览者用 户的身份和权限来执行。通过XSS可以比较容易地修改用户数据、窃取用户信息。
- 跨站请求伪造攻击：跨站请求伪造CSRF（Cross-site request forgery）是另一种常见的攻击。攻击者通过各种方法伪造一个请求，模仿用户提交表单的行为，从而达到修改用户的数据，或者执行特定任务的目的。为了 假冒用户的身份，CSRF攻击常常和XSS攻击配合起来做，但也可以通过其它手段，例如诱使用户点击一个包含攻击的链接。
- Http Heads攻击：凡是用浏览器查看任何WEB网站，无论你的WEB网站采用何种技术和框架，都用到了HTTP协议。HTTP协议在Response header和content之间，有一个空行，即两组CRLF（0x0D 0A）字符。这个空行标志着headers的结束和content的开始。“聪明”的

攻击者可以利用这一点。只要攻击者有办法将任意字符“注入”到 headers中，这种攻击就可以发生。

- 重定向攻击：一种常用的攻击手段是“钓鱼”。钓鱼攻击者，通常会发送给受害者一个合法链接，当链接被点击时，用户被导向一个似是而非的非法网站，从而达到骗取用户信任、窃取用户资料的目的。为防止这种行为，我们必须对所有的重定向操作进行审核,以避免重定向到一个危险的地方。常见解决方案是白名单，将合法的要重定向 的url加到白名单中，非白名单上的域名重定向时拒绝。第二种解决方案是重定向token，在合法的url上加上token，重定向时进行验证。

10 配置访问控制

10.1 配置Refer防盗链

您可以通过配置访问的Referer黑名单和白名单来实现对访客身份的识别和过滤，从而限制访问CDN资源的用户，提升CDN的安全性。通过本文您可以了解Refer防盗链的配置方法。

背景信息

- 防盗链功能基于HTTP协议支持的Referer机制，通过Referer跟踪来源，对来源进行识别和判断。
- 目前防盗链功能支持黑名单或白名单机制，您对资源发起请求后，请求到达CDN节点，CDN节点会根据您预设的防盗链黑名单或白名单，对访客的身份进行过滤。符合规则的用户可以顺利请求到资源，不符合规则的用户，请求会返回403响应码。

> 注意:
>
> - 防盗链是可选配置，默认不启用。
> - 黑白名单互斥，同一时间您只能选择一种方式。
> - 配置防盗链后，CDN自动添加泛域名支持。例如，如果您填写a.com，则最终配置生效的是*.a.com，所有子级域名都会生效。
> - 您可以设置是否允许空Referer字段访问CDN资源，即允许在浏览器地址栏输入地址直接访问资源URL。

操作步骤

1. 登录CDN控制台。
2. 在域名管理页面，选择您想设置的域名，单击管理。
3. 在左侧导航树，单击访问控制。

4. 在Refer防盗链区域框中，单击修改配置。

5. 根据界面提示，设置黑名单或白名单。

参数	说明
Refer类型	Refer防盗链类型如下： · 黑名单 黑名单内的域名均无法访问当前的资源。 · 白名单 只有白名单内的域名能访问当前资源，白名单以外的域名均无法访问当前的资源。 黑名单和白名单互斥，同一时间只支持其中一种方式生效。
规则	使用回车符分隔多个Refer黑名单或白名单，支持通配符如a.*b.com，可以匹配到a.aliyun.b.com或a.img.b.com等。

6. 单击确认。

10.2 配置URL鉴权

10.2.1 配置URL鉴权

URL鉴权功能主要用于保护用户站点的资源不被非法站点下载盗用。通过防盗链方法添加Referer黑名单和白名单的方式可以解决一部分盗链问题，由于Referer内容可以伪造，所

以Referer防盗链方式无法彻底保护站点资源。因此，您可以采用URL鉴权方式保护源站资源更为安全有效。

背景信息

URL鉴权功能通过阿里云CDN加速节点与客户资源站点配合，实现了一种更为安全可靠的源站资源防盗方法。

- CDN客户站点提供加密URL，URL中包含权限验证信息。
- 用户使用加密后的URL向加速节点发起请求。
- 加速节点对加密URL中的权限信息进行验证，判断请求的合法性。正常响应合法请求，拒绝非法请求。

如果您想了解Python鉴权代码示例，请参见 鉴权代码示例。

操作步骤

1. 登录CDN控制台。
2. 在域名管理页面，选择您想设置的域名，单击管理。
3. 在左侧导航树，单击访问控制。
4. 在右侧域名管理区域，单击URL鉴权。
5. 在鉴权URL设置区域框中，单击修改配置。

6. 根据界面提示，打开URL鉴权，配置URL鉴权信息。

参数	说明
鉴权类型	阿里云CDN兼容并支持三种鉴权方式。您可以根据自己的业务情况，选择合适的鉴权方式，来实现对源站资源的有效保护。URL鉴权类型如下： · 配置鉴权方式A · 配置鉴权方式B · 配置鉴权方式C
主KEY	输入鉴权方式对应的主用密码。
备KEY	输入鉴权方式对应的备用密码。

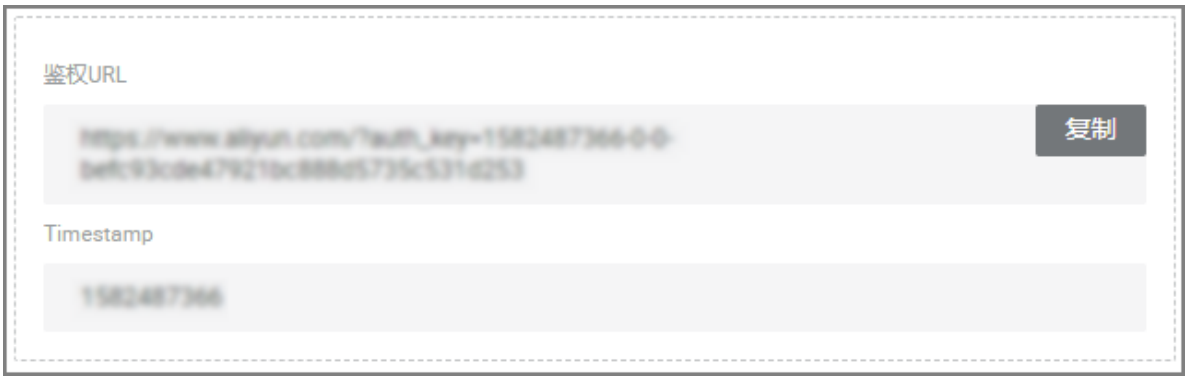
7. 单击确认。

8. 在生成鉴权URL区域，配置原始URL和鉴权信息。

参数	说明
原始URL	您可以输入完整的原始URL地址。例如：https://www.aliyun.com
鉴权类型	您可以根据所需，选择合适的URL鉴权类型： · 配置鉴权方式A · 配置鉴权方式B · 配置鉴权方式C
鉴权KEY	您可以根据所需，设置鉴权密码。鉴权KEY是鉴权URL设置中配置的主KEY备KEY。
有效时间	您可以根据所需，设置URL鉴权的有效时长。单位为：秒，例如：1800。

9. 开始生成。

获得鉴权URL和Timestamp。



10.2.2 配置鉴权方式A

URL鉴权功能主要用于保护用户站点资源不被非法站点下载盗用。阿里云CDN为您提供了三种鉴权方式，本文为您详细介绍鉴权方式A的原理，并用示例说明。

原理说明

访问加密URL构成：

```
http://DomainName/Filename?auth_key=timestamp-rand-uid-md5hash
```

鉴权字段描述：

字段	描述
DomainName	CDN站点的域名。
Filename	实际回源访问的URL，鉴权时Filename需以/开头。
auth_key	您设定的鉴权密钥。
timestamp	失效时间，整形正数，固定长度10，值为1970年1月1日以来的当前时间秒数+过期时间秒数。用来控制失效时间，过期时间由客户端设置，若设置为1800s，您访问CDN的时间超过1800s后，该鉴权失效。 例如，您设置访问时间为2020-08-15 15:00:00，则链接的真正失效时间为2020-08-15 15:30:00。
rand	随机数。建议使用UUID，不能包含中划线-，例如：477b3bbc253f467b8def6711128c7bec。
uid	用户ID，暂未使用（设置成0即可）。
md5hash	通过md5算法计算出的字符串，由数字0-9和小写英文字母a-z混合组成，固定长度32。

CDN服务器接到资源访问请求后，首先判断请求中的`timestamp`是否小于当前时间。

- 如果小于当前时间，服务器判定过期失效，并返回HTTP 403错误。
- 如果大于当前时间，构造出一个同样的字符串，参考下方`sstring`字符串，然后使用MD5算法算出`HashValue`，再与请求中`md5hash`进行比对。
  - 结果一致，鉴权通过，返回资源请求。
  - 结果不一致，鉴权失败，返回HTTP 403错误。

  `HashValue`是通过以下字符串计算出来的：

  ```
  sstring = "URI-Timestamp-rand-uid-PrivateKey" (URI是用户的请求对象相对地址，不包含参数，如/Filename)
  ```

```
HashValue = md5sum(sstring)
```

示例说明

通过以下示例说明，您可以准确理解鉴权方式A的实现方式。

1. **通过**`req_auth`**请求对象。**

```
http:// cdn.example.com/video/standard/1K.html
```

2. **设置密钥为：aliyuncdnexp1234。**
3. **设置鉴权配置文件有效时间为：2015年10月10日00:00:00，计算出秒数为：1444435200。**
4. **CDN服务器会构造一个用于计算**`Hashvalue`**的签名字符串。**

```
/video/standard/1K.html-1444435200-0-0-aliyuncdnexp1234
```

5. **根据该签名字符串，CDN服务器会计算出**`Hashvalue`**。**

```
HashValue = md5sum("/video/standard/1K.html-1444435200-0-0-
aliyuncdnexp1234") = 80cd3862d699b7118eed99103f2a3a4f
```

6. **加密URL请求。**

```
http://cdn.example.com/video/standard/1K.html?auth_key=1444435200-0-
0-80cd3862d699b7118eed99103f2a3a4f
```

如果计算出来的`HashValue`**值与请求中带的**`md5hash`**值相同，都**

为*80cd3862d699b7118eed99103f2a3a4f*，**则鉴权通过；反之鉴权失败。**

10.2.3 鉴权方式B

阿里云CDN鉴权功能为您提供了三种方式，本文档为您介绍了鉴权方式B的原理并用示例说明。鉴权功能主要用于保护用户站点的内容资源不被非法站点下载盗用。

原理说明

访问加密URL格式：

```
http://DomainName/timestamp/md5hash/FileName
```

当鉴权通过时，实际回源的URL是：

```
http://DomainName/FileName
```

鉴权字段描述

字段	描述
DomainName	CDN站点的域名。

字段	描述
timestamp	资源失效时间，作为URL的一部分，同时作为计算md5hash的一个因子，格式为：YYYYMMDDHHMM，有效时间1800s。 例如您设置访问时间为2020-08-15 15:00:00，则链接的真正失效时间为2020-08-15 15:30:00。
md5hash	通过md5算法计算出的验证串，由数字0-9和小写英文字母a-z混合组成，固定长度32。
PrivateKey	您设定的鉴权密钥。
Filename	实际回源访问的URL，鉴权时Filename需以/开头。

示例说明

您可以通过以下示例说明更好地理解鉴权方式B的实现。

1. 回源请求对象：

```
http://cdn.example.com/4/44/44c0909bcfc20a01afaf256ca99a8b8b.mp3
```

2. 密钥设为：aliyuncdnexp1234 (您自行设置)。
3. 访问源服务器时间为 201508150800（格式为： YYYYMMDDHHMM）。
4. CDN服务器会构造一个用于计算Hashvalue的签名字符串。

```
aliyuncdnexp1234201508150800/4/44/44c0909bcfc20a01afaf256ca99a8b8b.
mp3
```

5. 服务器根据该签名字符串计算md5hash。

```
md5hash = md5sum("aliyuncdnexp1234201508150800/4/44/44c0909bcf
c20a01afaf256ca99a8b8b.mp3") = 9044548ef1527deadafa49a890a377f0
```

6. 请求url为：

```
http://cdn.example.com/201508150800/9044548ef1527deadafa49a890a377f0
/4/44/44c0909bcfc20a01afaf256ca99a8b8b.mp3
```

如果计算出来的md5hash与您请求中带的md5hash

值（9044548ef1527deadafa49a890a377f0 ）一致，鉴权通过。

鉴权方式A和鉴权方式C具体原理和示例，请参见鉴权方式A、鉴权方式C。

10.2.4 鉴权方式C

本文为您介绍了鉴权方式C的原理并用示例说明。

原理说明

访问加密URL格式有如下两种格式。

- 格式1

```
http://DomainName/{<md5hash>/<timestamp>}/FileName
```

- 格式2

```
http://DomainName/FileName{&KEY1=<md5hash>&KEY2=<timestamp>}
```

说明:

{}中的内容表示在标准URL基础上添加的加密信息。

鉴权字段描述

字段	描述
PrivateKey	您设定的鉴权密钥。
FileName	实际回源访问的URL，鉴权时Filename需以/开头。
timestamp	访问源服务器时间，取UNIX时间。未加密的字符串，以明文表示。固定长度10，1970年1月1日以来的秒数，表示为十六进制。
DomainName	CDN站点的域名。

示例说明

- PrivateKey取值：`aliyuncdnexp1234`。
- FileName取值：`/test.flv`。
- timestamp取值：`55CE8100`。

- md5hash计算值为：

```
md5hash = md5sum(aliyuncdnexp1234/test.flv55CE8100) = a37fa50a5f
b8f71214b1e7c95ec7a1bd
```

- 生成加密URL:
  - 格式一：

```
http://cdn.example.com/a37fa50a5fb8f71214b1e7c95ec7a1bd/55CE8100/
test.flv
```

  - 格式二：

```
http://cdn.example.com/test.flv?KEY1=a37fa50a5fb8f71214b1
e7c95ec7a1bd&KEY2=55CE8100
```

当您使用加密URL访问加速节点，CDN服务器先把加密串1提取出来，并得到原始的URL的`FileName`和访问时间，然后按照定义的业务逻辑进行验证，验证步骤如下：

1. 使用原始的URL中的Filename、请求时间及PrivateKey进行md5加密得到一个加密串2。
2. 比较加密串2与加密串1是否一致，如果不一致则拒绝。
3. 取加速节点服务器当前时间，并与从访问URL中所带的明文时间相减，判断是否超过设置的时限t(时间域值t默认为1800s)。
   - 时间差小于设置时限，视作合法请求，CDN加速节点正常响应。
   - 时间差大于设置时限，拒绝该请求并返回HTTP 403。

> 说明:
> 有效时间1800s是指，当您访问源服务器时间超过自定义时间的1800s后，该鉴权失效。例如您设置了访问时间2020-08-15 15:00:00，链接真正失效时间是2020-08-15 15:30:00。

10.2.5 鉴权示例

通过本文档中的Demo，结合您的业务需要，您可以方便地对URL进行鉴权处理。URL鉴权规则请查阅鉴权配置。

Python版本

以下Python Demo包含三种鉴权方式：鉴权方式A、鉴权方式B、鉴权方式C，它们分别描述了三种不同鉴权方式的请求URL构成和哈希字符串构成。

> 说明:
> 如果URL中包含中文，请进行urlencod编码。

```
import re
import time
```

```
import hashlib
import datetime
def md5sum(src):
    m = hashlib.md5()
    m.update(src)
    return m.hexdigest()
#鉴权方式A
def a_auth(uri, key, exp):
    p = re.compile("^(http://|https://)?([^/?]+)(/[^?]*)?(\\?.*)?$")
    if not p:
        return None
    m = p.match(uri)
    scheme, host, path, args = m.groups()
    if not scheme: scheme = "http://"
    if not path: path = "/"
    if not args: args = ""
    rand = "0"      # "0" by default, other value is ok
    uid = "0"       # "0" by default, other value is ok
    sstring = "%s-%s-%s-%s-%s" %(path, exp, rand, uid, key)
    hashvalue = md5sum(sstring)
    auth_key = "%s-%s-%s-%s" %(exp, rand, uid, hashvalue)
    if args:
        return "%s%s%s%s&auth_key=%s" %(scheme, host, path, args,
auth_key)
    else:
        return "%s%s%s%s?auth_key=%s" %(scheme, host, path, args,
auth_key)
#鉴权方式B
def b_auth(uri, key, exp):
    p = re.compile("^(http://|https://)?([^/?]+)(/[^?]*)?(\\?.*)?$")
    if not p:
        return None
    m = p.match(uri)
    scheme, host, path, args = m.groups()
    if not scheme: scheme = "http://"
    if not path: path = "/"
    if not args: args = ""
    # convert unix timestamp to "YYmmDDHHMM" format
    nexp = datetime.datetime.fromtimestamp(exp).strftime('%Y%m%d%H%M')
    sstring = key + nexp + path
    hashvalue = md5sum(sstring)
    return "%s%s/%s/%s%s%s" %(scheme, host, nexp, hashvalue, path,
args)
#鉴权方式C
def c_auth(uri, key, exp):
    p = re.compile("^(http://|https://)?([^/?]+)(/[^?]*)?(\\?.*)?$")
    if not p:
        return None
    m = p.match(uri)
    scheme, host, path, args = m.groups()
    if not scheme: scheme = "http://"
    if not path: path = "/"
    if not args: args = ""
    hexexp = "%x" %exp
    sstring = key + path + hexexp
    hashvalue = md5sum(sstring)
    return "%s%s/%s/%s%s%s" %(scheme, host, hashvalue, hexexp, path,
args)
def main():
    uri = "http://xc.cdnpe.com/ping?foo=bar"            # original uri
    key = "<input private key>"                         # private key
of authorization
    exp = int(time.time()) + 1 * 3600                   # expiration
time: 1 hour after current itme
```

```
    authuri = a_auth(uri, key, exp)                    # auth type:
a_auth / b_auth / c_auth
    print("URL : %s\nAUTH: %s" %(uri, authuri))
if __name__ == "__main__":
    main()
```

10.3 配置IP黑/白名单

您可以通过配置IP黑名单和白名单来实现对访客身份的识别和过滤，从而限制访问CDN资源的用户，提升CDN的安全性。通过本文您可以了解IP黑/白名单的配置方法。

背景信息

- IP黑名单：黑名单内的IP均无法访问当前资源。

  如果您的IP被加入黑名单，该IP的请求仍可访问到CDN节点，但是会被CDN节点拒绝并返回403，CDN日志中仍会记录这些黑名单中的IP请求记录。
- IP白名单：只有白名单内的IP能访问当前资源，白名单以外的IP均无法访问当前资源。

> 说明：
>
> - IP黑名单和白名单均支持IPv6地址。
> - IP黑名单和白名单均支持IP网段添加。例如：192.168.0.0/24，24表示采用子网掩码中的前24位有效位，即用32-24=8bit来表示主机号，该子网可以容纳2^8-2=254台主机。故192.168.0.0/24表示IP网段范围是：192.168.0.1~192.168.0.254。

操作步骤

1. 登录CDN控制台。
2. 进入域名管理页面，选择需要设置的域名，单击管理。
3. 在左侧导航树，单击访问控制。
4. 在右侧域名管理区域，单击IP黑/白名单。

5. 单击修改配置。

6. 根据界面提示，配置IP的黑名单或白名单。

参数	说明
名单类型	IP名单类型如下： · 黑名单 黑名单内的IP均无法访问当前资源。 · 白名单 只有白名单内的IP能访问当前资源，白名单以外的IP均无法访问当前资源。 黑名单和白名单互斥，同一时间只支持其中一种方式生效。
规则	最多配置100个IP地址，使用回车符分隔，不可配置重复网段，例如：192.168.0.1/24。

7. 单击确认。

10.4 UsageAgent黑/白名单

本文为您介绍了UsageAgent黑/白名单原理、使用场景和控制台操作步骤。您可以配置UsageAgent黑/白名单功能，CDN节点服务器会根据您请求的Usage-Agent字段进行黑白名单的管理。

背景信息

当您需要根据请求的Usage-Agent字段进行访问控制，请配置UsageAgent黑/白名单功能，实现对请求过滤。

> **说明:**
>
> - User-Agent规则不区分大小写，且支持 `*`通配符。例如：`*curl*|*IE*|*chrome*|*firefox*` ，多个值用|分割。
> - 黑白名单互斥，只支持同时启用其中一个名单。

操作步骤

1. 登录CDN控制台。
2. 在左侧导航栏，单击域名管理。
3. 在您需要设置的域名，单击管理。
4. 在左侧导航栏，单击访问控制。
5. 在UserAgent黑/白名单区域框中，单击修改配置。
6. 根据您的需求配置黑白名单的规则，单击确定。

11 性能优化设置

11.1 页面优化

当您开启页面优化功能时，CDN自动清除HTML页面冗余的注释和重复的空白符，缩小文件体积，提升页面可阅读性。本文为您详细介绍开启页面优化功能的方法。

背景信息

开启页面优化功能后，CDN自动删除当前域名下所有HTML页面中冗余的注释和重复的空白符，这样可以有效地去除页面的冗余信息，减小文件体积，提高加速分发效率。

如果源站文件配置了MD5校验机制，则请勿开启该功能。当CDN进行页面优化时，该文件的MD5值会被更改，导致优化后文件的MD5值和源站文件的MD5值不一致。

操作步骤

1. 登录CDN控制台。
2. 进入域名管理页面，选择您需要设置的域名，单击管理。
3. 在左侧导航树，单击性能优化。
4. 在页面优化区域框中，打开页面优化开关。

11.2 智能压缩

当您开启智能压缩功能时，CDN自动对静态文件进行Gzip压缩。通过智能Gzip压缩方式，可以有效减小传输文件大小，提升加速效率。本文为您详细介绍开启智能压缩功能的方法。

背景信息

- 目前智能压缩支持的内容格式：text/html、text/xml、text/plain、text/css、application/javascript、application/x-javascript、application/rss+xml、text/javascript、image/tiff、image/svg+xml、application/json、application/xmltext。
- 客户端请求携带请求头`Accept-Encoding: gzip`：客户端希望获取对应资源的gzip压缩响应。
- 服务端响应携带响应头`Content-Encoding: gzip`：服务端响应的内容为gzip压缩的资源 。

> 注意：
>
> - 如果源站文件配置了MD5校验机制，则请勿开启该功能。当CDN对静态文件进行压缩优化时，该文件的MD5值会被更改，导致压缩优化后文件的MD5值和源站文件的MD5值不一致。
> - 只有当源站文件大小超过1024B时，CDN才会进行Gzip压缩。
> - Internet Explorer 6对Gzip的兼容性较差，如果有Internet Explorer 6的访问需求，不建议开启智能压缩功能。

操作步骤

1. 登录控制台。
2. 进入域名管理页面，选择您需要设置的域名，单击管理。
3. 在左侧导航树中，单击性能优化。
4. 在智能压缩区域框中，打开智能压缩开关。

11.3 Brotli压缩

当您需要对静态文本文件进行压缩时，可以开启此功能，有效减小传输内容大小，加速分发效果。本文为您详细介绍开启Brotli压缩功能的方法。

背景信息

Brotli是开源的一种新型压缩算法。开启Brotli压缩功能后，CDN节点返回请求资源时，会对html、js、css等文本文件进行Brotli压缩。压缩文本文件时，Brotli压缩比智能压缩性能提升约15~25%。

- 当客户端的请求携带请求头`Accept-Encoding: br`时，表示客户端希望获取对应资源时进行Brotli压缩。
- 当服务端响应携带响应头`Content-Encoding: br`时，表示服务端响应的内容是Brotli压缩的资源。

> **注意：**
> 当Brotli压缩和Gzip压缩同时开启时，客户端请求头`Accept-Encoding`同时带`br`和`gzip`时，CDN节点将优先选择Brotli压缩。

操作步骤

1. 登录CDN控制台。
2. 进入域名管理页面，选择您需要设置的域名，单击管理。
3. 在左侧导航树中，单击性能优化。
4. 在Brotli压缩区域框中，打开Brotli压缩开关。

11.4 过滤参数

如果您的URL请求中携带?和##，例如：`http://alibaba.com/content?a=10`，则CDN节点在收到URL请求后，判断是否需要携带参数的URL返回源站。 本文为您详细介绍配置过滤参数的方法。

背景信息

- 开启过滤参数。

  请求URL到CDN节点后，会截取到没有参数的请求URL，且CDN节点仅保留一份副本。

  - 虽然大部分HTTP请求中包含参数，但是参数内容优先级不高，可以忽略参数浏览文件，开启后可以有效提高文件缓存命中率，提升分发效率。
  - 如果参数有重要含义，例如，包含文件版本信息等，则推荐您设置为保留过滤参数。您最多可以设置10个保留参数，如果请求URL中包含您设置的保留参数，则会携带该参数回源。

- 关闭过滤参数。

  每个不同的URL都缓存不同的副本在CDN的节点上。

过滤参数包括保留过滤参数和忽略参数这两个功能。

- 保留过滤参数：保留指定参数，多个参数逗号隔开，未指定的参数将不会被保留。
- 忽略参数：删除指定的参数，多个参数之间用空格隔开，剩余参数将不会被忽略。

> **说明:**
> URL鉴权功能的优先级高于过滤参数。由于鉴权方式A中的鉴权信息包含http请求的参数部分，所以CDN优先进行鉴权判断，鉴权通过后在CDN节点缓存一份副本。配置URL鉴权的操作方法，请参见配置URL鉴权。

操作步骤

1. 登录CDN控制台。
2. 进入域名管理页面，选择需要设置的域名，单击管理。
3. 在左侧导航树中，单击性能优化。
4. 在过滤参数区域框中，单击保留过滤参数或忽略参数区域的修改配置。

5. 您可以根据所需配置保留过滤参数或忽略参数。

   · 保留过滤参数

参数	说明
过滤参数	保留过滤参数开关。打开过滤参数开关后，资源回源时会去除URL中? 之后的参数，提升文件缓存命中率。
保留参数	配置需要保留的参数。最多可以配置10个保留参数,，用空格作分隔符。
保留回源参数	保留回源参数开关。打开保留回源参数开关后，资源回源时，保留所有参数。

   · 忽略参数

参数	说明
过滤参数	忽略过滤参数开关。打开过滤参数开关后，资源回源时会删除指定参数，剩余参数将不会被删除。
忽略参数	配置需要忽略的参数。最多可以配置10个忽略参数,，用空格作分隔符。
保留回源参数	保留回源参数开关。打开保留回源参数开关后，资源回源时，保留所有参数。

   示例说明：

   `http://www.abc.com/a.jpg?x=1`请求URL到CDN节点。

   · 开启保留过滤参数功能：

     a. CDN节点向源站发起请求`http://www.abc.com/a.jpg`，忽略参数x=1。

     b. 源站响应该请求内容后，响应到达CDN节点。

     c. CDN节点会保留一份副本，然后继续向终端响应 `http://www.abc.com/a.jpg` 的内容。

     d. 所有类似的请求`http://www.abc.com/a.jpg?##`均响应CDN副本`http://www.abc.com/a.jpg` 的内容。

   · 关闭保留过滤参数功能：`http://www.abc.com/a.jpg?x=1`和`http://www.abc.com/a.jpg?x=2`会响应不同参数源站的响应内容。

6. 单击确认。

12 高级配置

12.1 配置带宽封顶

带宽封顶功能是指当统计周期（5分钟）产生的平均带宽超出您设置的带宽最大值时，为了保护您的域名安全，此时域名会自动下线，所有的请求会回到源站，CDN将停止加速服务，避免异常流量给您带来的异常消费。域名下线后，您可以在控制台重新启用该域名。通过本文，您可以快速了解开通带宽封顶功能的方法和注意事项。

背景信息

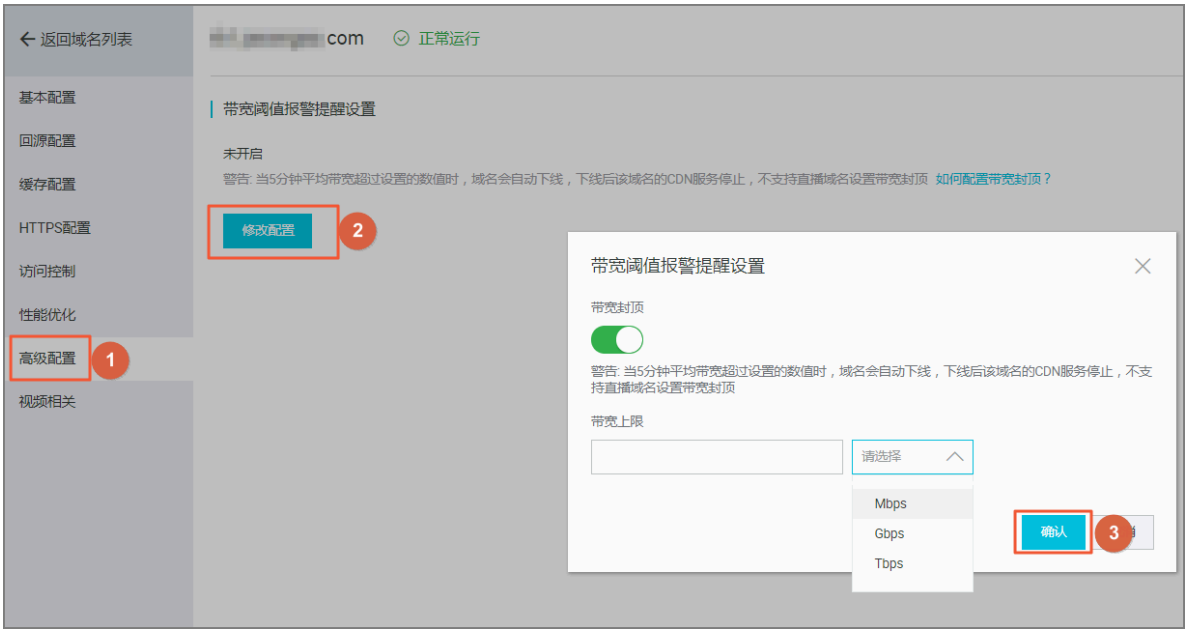
配置带宽封顶功能的注意事项如下：

- 如果RAM子账号需要开通带宽封顶功能，则您需要登录RAM控制台，新增管理CDN的权限`AliyunCDNFullAccess`。
- 泛域名暂不支持带宽封顶功能，设置后不会生效。
- 开启带宽封顶功能后，您的业务会受到带宽封顶的限制而下线，为了不影响您的域名业务，建议您合理评估，谨慎设置您的带宽峰值。
- 如果您的CDN加速服务因带宽封顶而下线，则可以在CDN控制台的域名管理页面，选中该域名对应的复选框，单击启用，重新启用该域名。

操作步骤

1. 登录CDN控制台。
2. 在域名管理页面，选择需要设置的域名，单击管理。
3. 在左侧导航树中，单击高级配置。
4. 在带宽封顶区域框，单击修改配置。

5. 打开带宽封顶开关，配置带宽上线值。



说明:

- 各个单位之间进制为1000。例如：1Tbps=1000Gbps，1Gbps=1000Mbps。
- 您可以根据域名的实际使用情况，选择开启或者关闭带宽封顶功能。
- 如果带宽封顶功能处于升级中，则无法开启该功能。

6. 单击确认。

13 视频相关

13.1 Range回源

Range回源是指客户端通知源站服务器只返回指定范围内的部分内容，有利于较大文件的分发加速。开启Range回源功能，可以减少回源流量消耗，并且提升资源响应时间。通过本文您可以了解开启Range回源的方法和注意事项。

背景信息

配置Range回源的注意事项如下：

- 配置Range回源之前，需要源站支持Range请求，即http请求头中包含Range字段，源站能够响应正确的206文件分片。
- Range回源是可选配置项，默认不开启。

操作步骤

1. 登录CDN控制台。
2. 在域名管理页面，选择需要设置的域名，单击管理。
3. 在左侧导航树中，单击视频相关。
4. 在Range回源区域框，单击修改配置。
5. 选择Range回源为开启、关闭强制。

Range回源	具体描述	示例
开启	参数可以请求回源站。源站需要依据Range的参数，响应文件的字节范围，同时CDN节点也会向客户端响应相应字节范围的内容。	客户端向CDN请求中含有`range:0-100`，则源站端收到的请求中也会含有`range:0-100`，并且源站响应给CDN节点，然后CDN节点响应给客户端字节内容是0-100这个范围，一共101个字节。
关闭	CDN上层节点会向源站请求全部的文件，由于客户端会在收到Range定义的字节后自动断开http链接，请求的文件没有缓存到CDN节点上，最终导致缓存的命中率较低，并且回源流量较大。	客户端向CDN请求中含有`range: 0-100`，则源站端收到的请求中没有range这个参数。源站响应给CDN节点完整文件，但是CDN节点响应给客户端的就是101个字节，由于连接断开了，会导致该文件没有缓存到CDN节点上。

Range回源	具体描述	示例
强制	参数请求强制回源站。	当选择Range回源为强制，请确保源站支持Range参数。

说明:

您指定Range回源为强制后，任何分片请求都会强制分片回源。

您还可以通过调用API使用该功能，详情请参见SetRangeConfig。

6. 单击确认。

13.2 拖拽播放

通过配置拖拽播放功能，您可以在播放视音频时，随意拖拽播放进度，而不影响视音频的播放效果。通过本文您可以了解开启拖拽播放功能的方法和背景信息。

背景信息

拖拽播放功能是指在视音频点播场景中，如果您拖拽播放进度时，客户端会向服务器端发送类似`http://www.aliyun.com/test.flv?start=10`的URL请求，服务器端会向客户端响应从第10字节的前一个关键帧（如果start=10不是关键帧所在位置）的数据内容。

- 配置拖拽播放功能之前，需要确认源站支持Range请求，即如果http请求头中包含Range字段，源站需要能够响应正确的206文件分片。

- 拖拽播放功能支持的文件格式和URL格式如下表所示。

文件格式	meta信息	start参数	举例
MP4	源站视频的meta信息必须在文件头部，不支持meta信息在尾部的视频。	start参数表示的是时间，单位是s，支持小数以表示ms（如start=1.01，表示开始时间是1.01s），CDN会定位到start所表示时间的前一个关键帧（如果当前start不是关键帧）。	请求http: //domain/video.mp4?start=10就是从第10秒开始播放视频。
FLV	源站视频必须带有meta信息。	start参数表示字节，CDN会自动定位到start参数所表示的字节的前一个关键帧（如果start当前不是关键帧）。	对于http: //domain/video.flv，请求http:// domain/video.flv?start=10就是从第10字节的前一个关键帧（如果start=10不是关键帧所在位置）开始播放视频。

操作步骤

1. 登录CDN控制台。
2. 在域名管理页面，选择需要设置的域名，单击管理。
3. 在左侧导航树中，单击视频相关。
4. 在拖拽播放区域框，开启拖拽播放功能。

14 CDN WAF防护功能

本文档介绍了阿里云CDN的WAF防护功能、使用场景和费用说明。目前CDN的WAF功能正在公测，您暂时无法自动在控制台开启WAF功能。

功能介绍

阿里云CDN的WAF功能，是指CDN融合了云盾Web应用防火墙（Web Application Firewall，简称 WAF）能力，在CDN节点上，提供WAF防护功能。WAF防护具体功能，请参见什么是Web应用防火墙。

适用场景

CDN的WAF服务主要适用于金融、电商、O2O、互联网+、游戏、政府、保险等行业，保护您的网站在使用CDN加速的同时，免受因外部恶意攻击而导致的意外损失。

使用CDN WAF功能后，可以帮助您解决以下问题：

- 防数据泄密，避免因黑客的注入入侵攻击，导致网站核心数据被拖库泄露。
- 阻止木马上传网页篡改，保障网站的公信力。
- 提供虚拟补丁，针对网站被曝光的最新漏洞，最大可能地提供快速修复规则。

操作步骤

目前阿里云CDN的WAF功能属于公测阶段，您暂时无法自动在控制台开启WAF功能。（仅支持开启后自动关闭WAF功能）

您可以扫下图二维码加入CDN WAF钉钉讨论群或提工单，我们在了解您的需求后，会为您进行WAF配置。

费用说明

当您开启WAF功能后，CDN WAF会对此域名的所有请求进行检测，并按照账户维度，对域名开启WAF功能的请求次数汇总，然后收费。

公测期间，请求数按每小时计算。WAF的计费规则如下：

每小时请求数	费用
1-20000	0.4元（固定付费）
20001-500000	0.2元/万次请求
500001-5000000	0.18元/万次请求
大于5000000	0.15元/万次请求

计费案例：以用户A和用户B分别在2019年2月28日10:20开通了2个域名的CDN WAF功能为例，他们产生的费用如下表所示：

用户	10：20-11：20 产生请求次数	11：21分收到账单金额（元）
A	15000	0.4
B	350000	7 （350000/10000*0.2）

15 域名管理FAQ

如何解决在使用CDN+OSS组合过程中，静态文件强制下载的问题?

原因：由于OSS的某个策略，当访问默认三级域名时，会给文件加上attachment属性，从而使文件强制下载。

解决办法：将您的回源HOST类型修改为加速域名。具体配置步骤，请参见设置回源HOST。

如问题还未解决，请提交工单。

CDN节点默认缓存策略是什么?

- 缓存时间计算

  缓存时间为t，单位为秒。

  - t =（savetime - last_modified）*0.1
  - t = max(10, t)
  - t = min(t, 3600)

- 默认缓存规则
  - 当对象Last-Modified为20140801 00:00:00，当前时间为20140801 00:01:00，（curtime-Last_modified)*0.1=6s，那么缓存时间为10s，因为最小值为10s。
  - 当对象Last-Modified为20140801 00:00:00，当前时间为20140802 00:00:00，(curtime-Last_modified)*0.1=8640s，那么缓存时间为3600s。
  - 当对象Last-Modified为20140801 00:00:00，当前时间为20140801 00:10:00，(curtime-Last_modified)*0.1=60s，那么缓存时间为60s。
  - 如果源站没有Last-Modified响应头，但有ETag，则该对象极有可能是静态资源，将其默认缓存时间设置为 dft_expires指令配置的最小值。
  - 如果源站没有Last-Modified，也没有ETag，则认为该对象为动态内容，将其默认缓存时间设置为0，每次都回源。

  说明:

  - 因为网站开发及其相关技术人员更清楚自身网站的业务逻辑、静态和动态因素，所以建议您通过控制台根据文件类型和文件所在目录，设置缓存时间。详细说明，请参见设置缓存过期时间。
  - 如果您已经配置了缓存策略，那么Cache的默认缓存策略不生效。

使用CDN后，文件与源文件不一致，如何刷新缓存?

使用CDN产品后，如果遇到源站内容更新，并且使用旧URL发布给用户使用。需要在更新源站内容后，同时刷新CDN节点的缓存，这样才能保证源站内容与CDN的缓存内容保持一致。

如何刷新，请参见配置刷新预热。

如何解决CDN下载文件不一致的问题?

访问CDN上未过期的缓存，将直接返回该缓存资源。如果您对源站进行了同名更新，则访问时会发现请求到的资源仍然是旧的资源，导致网站内容错乱。建议您从如下几个方面解决该问题：

1. 建议您源站的内容不使用同名更新，而是以版本号的方式同步，即给URL增加版本参数，使CDN回源请求新资源。

   
   说明:
   如果开启了过滤参数功能，则该方式无效。

2. 同名更新后，您可以通过CDN控制台或OpenAPI手动刷新对应资源的URL。如何刷新，请参见配置刷新预热。

   
   说明:

   - URL刷新主要适合于单个资源，刷新速度较快。
   - 目录刷新会刷新该目录下的所有文件，刷新速度较慢，且由于该目录下所有资源下次请求都会回源，因此可能会增加源站带宽压力。

3. 如果CDN的源站是OSS源站，您可以通过OSS控制台，开启CDN缓存刷新。这样就可以在OSS源站出现Object的同名更新时，调用CDN的刷新接口刷新对应的URL。详细说明，请参见开启 CDN 缓存自动刷新。

关于CDN缓存方面的知识点有哪些?

- CDN读取数据过程为：客户端 -> CDN L1层 -> CDN L2层 -> 源站。当客户端访问您的源站资源时，客户端将先查看CDN的1级节点，再查找CDN的2级节点，如果2级节点没有，再查找源站。源站中的数据同步到2级节点，2级节点同步到1级节点，再从1级节点返回给客户端需要访问的数据。
- CDN的刷新缓存，请参见配置刷新预热。
- CDN缓存规则的配置，请参见设置缓存过期时间。
- 为了最优使用CDN服务，建议您将动静态页面进行域名分离，静态页面的域名使用CDN加速。
- 如果源站的cachecontrol、expires、lastmodify和etag都没设置，且CDN也没设置缓存规则，则CDN不进行缓存。
- 如果源站设置了no-cache、private、max-age = 0都遵循源站，则CDN不进行缓存。

导致CDN缓存命中率下降的因素有哪些?

导致CDN缓存命中率下降的因素包括：

- 客户是否刷新过缓存?

  CDN的URL或目录刷新会清除CDN缓存，如果您刷新缓存，有可能造成短时间命中率下降。
- 带宽是否突增？并且访问的都是新的URL?

  带宽突增或者访问的新URL较多，会导致CDN节点回源较多，命中率会表现有下降趋势。
- 源站是否有新内容发布?

  CDN节点访问新内容，导致CDN节点回源较多，命中率会表现有下降趋势。
- 源站是否出现过异常?

  源站出现过异常，导致5XX和4XX增加，由于5XX和4XX不缓存，会表现命中率下降。
- 源站访问URL的header参数，或者在CDN控制管理后台的缓存配置规则是否改变过?

  调整缓存过期时间，有可能会带来命中率的变化。

CDN命中率低的原因是什么?

- CDN数据流向：客户端 -> CDN L1层 -> CDN L2层 -> 源站。阿里云CDN控制台统计的命中率仅仅是CDN L1层的命中率，实际上L2层的命中率数据也是从CDN节点获取的，并不会回源，所以真实的CDN命中率略高于控制台上显示的数据。
- 在您加速域名流量不高的情况下，即使未命中的URL不多，但是对命中率的统计计算影响很大。例如某CDN加速域名对外提供了10个可以访问的URL，其中有一个URL源站上设置了no-cache导致不缓存，即使其他URL访问都命中，命中率也仅有90%。
- 如果您源站上缓存header设置不当，或者缺少必要的header，根据CDN的缓存规则，这种情况下的缓存规则是不生效的，所以资源不缓存。

  说明:

  - 缓存header设置不当主要指您在配置缓存规则中，将cache-control设置为no-cache/no-store/max-age=0/private或者将Pragma设置为no-cache，此时CDN执行不缓存。
  - 缺少必要的header指源站的response头中不包含etag和last-modified，这种情况也会导致不缓存。
  - 导致不缓存的详细信息，请参考判断CDN不缓存某文件的方法。
- 控制台设置了不缓存的规则。例如：某目录或者某种后缀的文件设置缓存时间为0秒。
- 源站动态内容多。目前CDN主要是加速静态资源（css、js、html、图片、txt、视频等），针对动态资源（php、jsp、包含内部逻辑处理甚至cookie等），基本都会回源。
- CDN的访问URL中带有可变参数。例如：*http://dccdn.pier39.cn/1.txt?timestamp=14378923*中的timestamp表示时间戳，每次访问都会不同。CDN针对第一次访问的URL（之前未预热），无论该URL是否符合CDN的缓存规则，第一次访问都是未命中的，因为

此时CDN节点还未缓存该文件。由于URL后面的参数可变，所以每次访问都是一个全新的URL，每次都会未命中，从而影响命中率。

- 刷新操作频繁，有定时刷新的操作。由于每次刷新都会导致所有已经在CDN上缓存的URL失效，那么下次访问同样的URL会未命中，从而影响命中率。
- 文件热度不够，不经常被客户访问到，导致被提前从CDN节点删除。CDN节点上缓存的文件，可以理解为按照热度属性采取末尾淘汰制，热度指该文件在该节点上被访问的频率，文件热度不够，一定程度上跟这个域名本身的流量不高有关系。

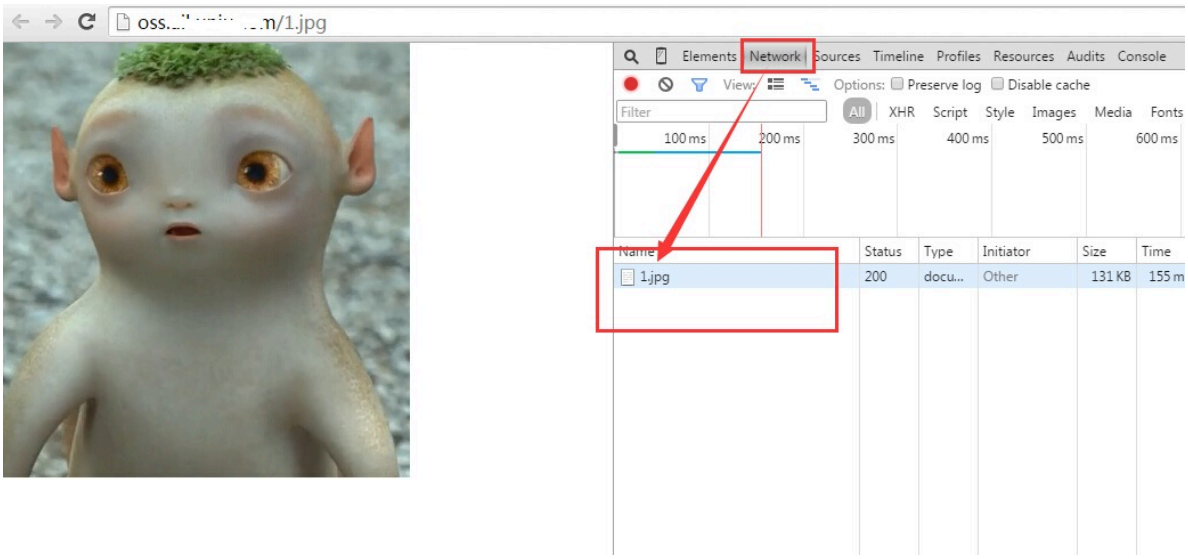
CDN如何设置某个目录或文件不缓存?

将目录或者文件的缓存时间设置为0即可，操作方法请参见设置缓存过期时间。

如何通过浏览器审查元素判断CDN缓存是否成功?

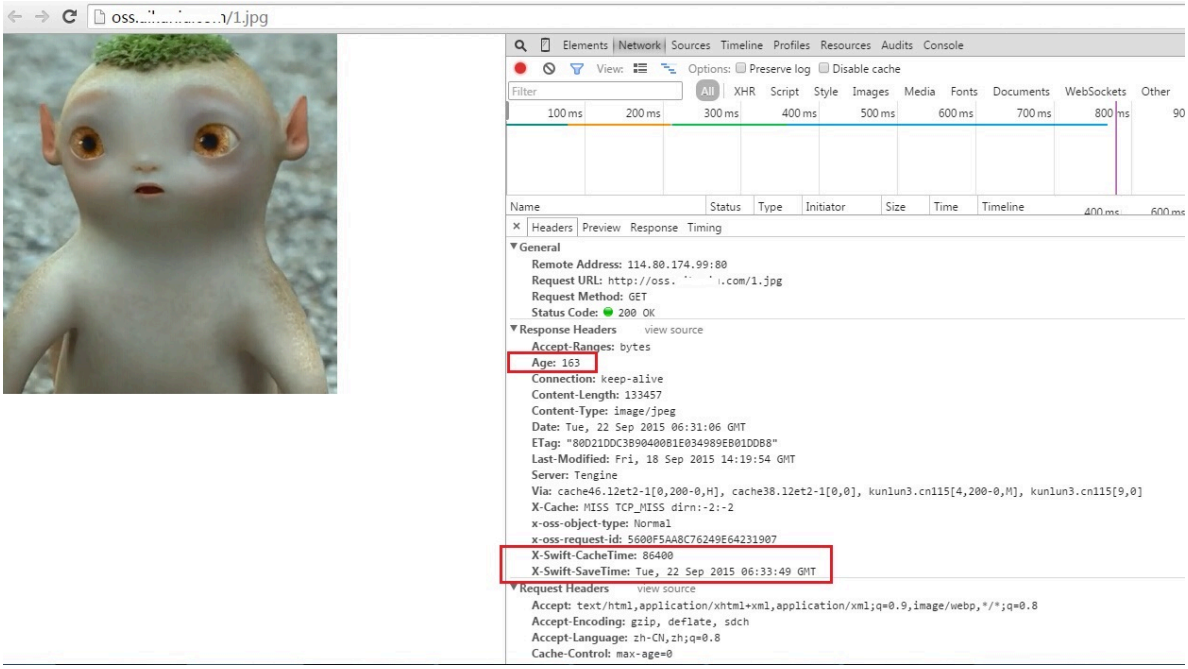
开通CDN服务并配置完成后，如果您需要查看您的内容是否缓存到CDN上，请参考如下步骤：

1. 打开浏览器（以Google Chrome浏览器为例），打开开发者工具，在地址栏输入要查看的URL。



2. 单击Network。

3. 单击访问的图片（要加速的内容），您可以查看请求（request）和返回（response）的详细报文信息。如下图：



您需要注意如下三个字段：

- X-Swift-SaveTime：内容开始在CDN上缓存的时间。如上图，即2015-09-22 06:33:49开始在CDN缓存。由于系统时间是GMT时间，所以需要折算成北京时间，也就是2015-09-22 14:33:49开始缓存。
- X-Swift-CacheTime：CDN的默认缓存时间，以秒为单位。如上图中的86400，即缓存24小时。
- Age：内容在CDN上已经缓存的时间。如上图中的163s，即该内容已经在CDN缓存了163秒。根据时间，从2015-09-22 14:33:49开始缓存的，当前时间则为2015-09-22 14:36:32。您可以和电脑当前时间进行对比。

如问题还未解决，请提交工单。

如何设置Apache缓存策略?

您可以通过apache的mod_expires和mod_headers两个模块设置Apache的缓存策略。

- mod_expires模块

  mod_expires模块允许通过配置文件控制HTTP的Expires和Cache-Control头内容，它的主要作用是自动生成页面头部信息中的Expires标签和Cache-Control标签，从而降低客户端的访问频率和次数，达到减少不必要流量和增加访问速度的目的。

  mod_expires是apache众多模块中配置比较简单的一个，它一共包括三条指令：

  - ExpiresActive指令：打开或关闭产生Expires:和Cache-Control:头的功能。
  - ExpiresByType指令：指定MIME类型的文档(如text、html)的过期时间。
  - ExpiresDefault指令：默认所有文档的过期时间。

  过期时间的写法：

```
“access plus 1 month”
“access plus 4 weeks”
“now plus 30 days”
“modification plus 5 hours 3 minutes”
A2592000
M604800
```

  说明：

  - access、now和A这三种写法的意义相同，都是指过期时间从访问时开始计算。
  - modification和M的意义相同，指过期时间从被访问文件的最后修改时间开始计算。

  本种写法只对静态文件起作用，对由脚本生成的动态页面无效。配置实例：

```
 ExpiresActive On(开启mod_expires功能)
 ExpiresDefault "access plus 6 months"(默认的过期时间是6个月)
 ExpiresByType image/* "access plus 10 years"(图片的文件类型缓存时间为10年)
 ExpiresByType text/* "access plus 10 years"(文本类型缓存时间为10年)
 ExpiresByType application/* "access plus 30 minutes"(application文件类型缓存30分钟)
```

  验证： image/jpeg的缓存时间为315360000s（10年）。

```
HTTP/1.1 200 OK
Date: Fri, 27 Apr 2012 08:12:30 GMT
Server: Apache
Last-Modified: Sat, 20 Aug 2011 05:38:30 GMT
Content-Length: 35706
Cache-Control: max-age=315360000
Expires: Mon, 25 Apr 2022 08:12:30 GMT
Content-Type: image/jpeg
```

  如果将image/jpeg设置为不缓存（将max-age设置为0s）：

```
#ExpiresByType image/* "access plus 10 years"
```

```
ExpiresByType image/*  A0
```

```
HTTP/1.1 200 OK
Date: Fri, 27 Apr 2012 08:04:34 GMT
Server: Apache
Last-Modified: Sat, 20 Aug 2011 05:38:30 GMT
Content-Length: 35706
Cache-Control: max-age=0
Expires: Fri, 27 Apr 2012 08:04:34 GMT
Content-Type: image/jpeg
```

- mod_headers模块

```
# YEAR
   Header set Cache-Control “max-age=2592000"
   # WEEK
   Header set Cache-Control “max-age=604800"
   # NEVER CACHE
   Header set Expires “Thu, 01 Dec 2003 16:00:00 GMT”
   Header set Cache-Control “no-store, no-cache, must-revalidate”
   Header set Pragma “no-cache”
```

如问题还未解决，请提交工单。

如何设置服务器端的过期时间?

过期时间控制支持三个维度，优先级依次为控制台设置 -> 源站header设置 -> cache的默认策略设置。

> 说明:
>
> - 关于控制台设置的详细说明，请参见设置缓存过期时间。
> - 关于cache的默认策略设置的详细说明，请参见CDN节点默认缓存策略是什么？。

Webserver缓存策略设置（源站设置）

1. 设置IIS缓存策略。详细说明，请参见如何设置IIS缓存策略？。
2. 设置Nginx缓存策略。详细说明，请参见如何设置Nginx缓存策略？。
3. 设置Apache缓存策略。详细说明，请参见如何设置Apache缓存策略？。

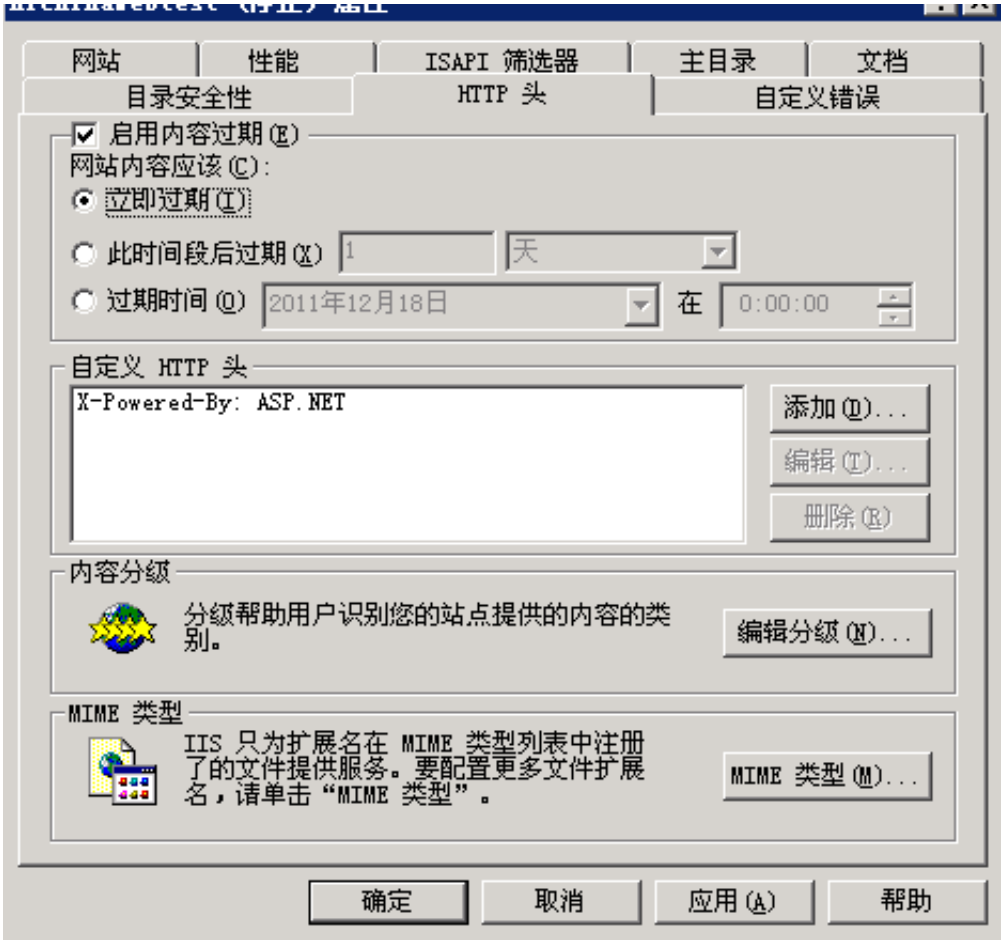
如何设置IIS缓存策略?

设置IIS缓存策略的操作步骤如下：

1. 因为整体的站点只对.html、.jpg、.png、.gif、.apk等文件进行缓存，首先将整个站点设置成不缓存，具体操作如下：

   a. 打开IIS信息管理器，右键单击服务网站a.cc.com的属性。

   b. 单击HTTP头，勾选启用内容过期，选择立即过期，单击确定。



2. 上述设置后，整个网站的内容都不会被CDN缓存，接着设置.html、.jpg、.png、.gif、.apk等文件类型的缓存策略。

   · 不同扩展名的文件都单独放在一个特定的目录下面，且该目录没有其他扩展名的文件。

   针对这个扩展名所在的整个目录设置缓存的时间。具体操作如下：

   a. 打开IIS信息管理器。

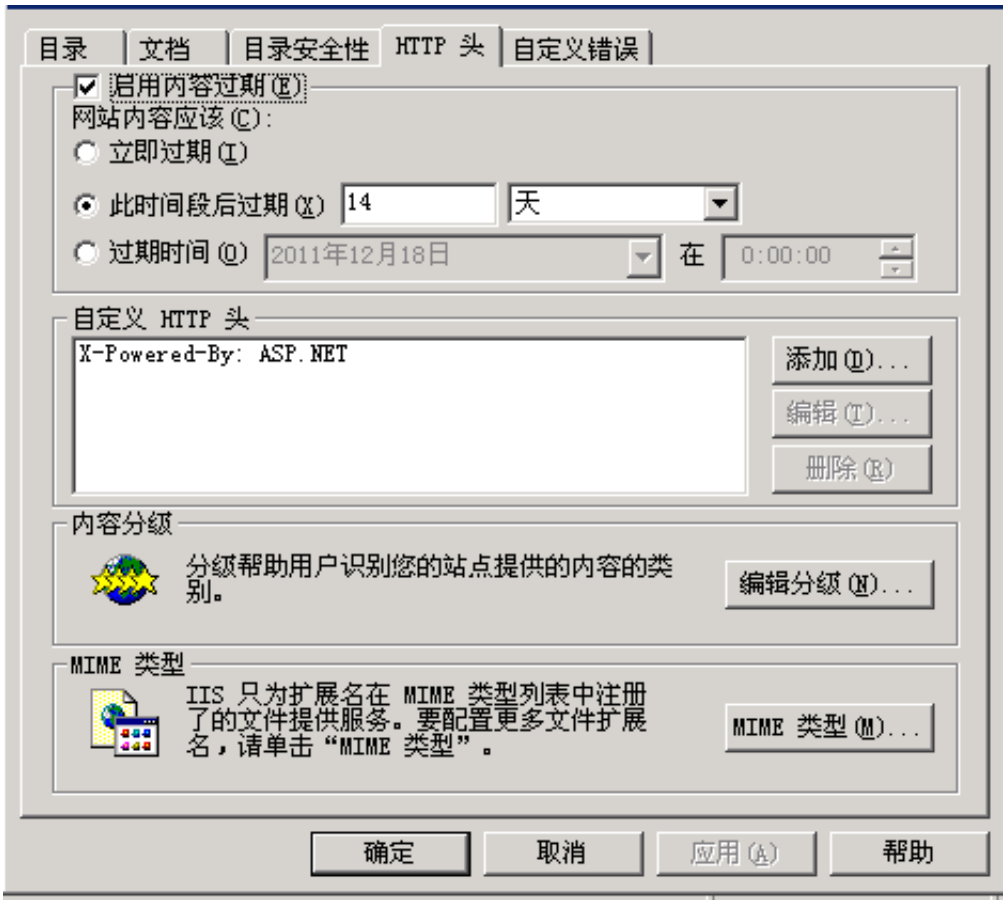
   b. 展开网站a.cc.com的目录，选中需要设置缓存时间的目录，如所有jpg文件都存储在*img*这个目录下，右键单击该目录并选择属性。

   c. 单击HTTP头。

   说明:

由于步骤1已经设置整个网站不缓存，所以此时HTTP头选项下的缓存设置和步骤1中的相同。

d. 选择此时间段后过期，设置具体的过期时间，单击确定。



· 特定扩展名的文件不是统一放在唯一的目录，而是和其他扩展名文件混合放在一个目录下。

说明:

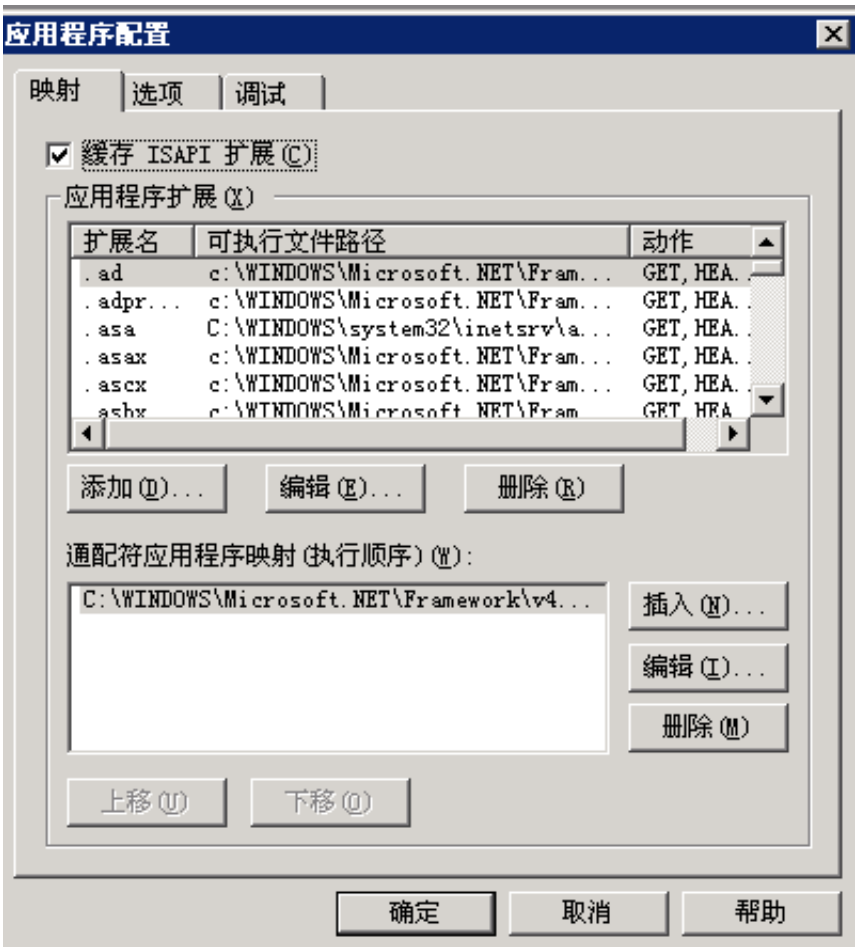
为了避免针对特定扩展名的文件进行逐个的配置，需要设置IIS支持通配符。

以*bin*目录下的*test.jpg*为例，介绍缓存设置的操作步骤。

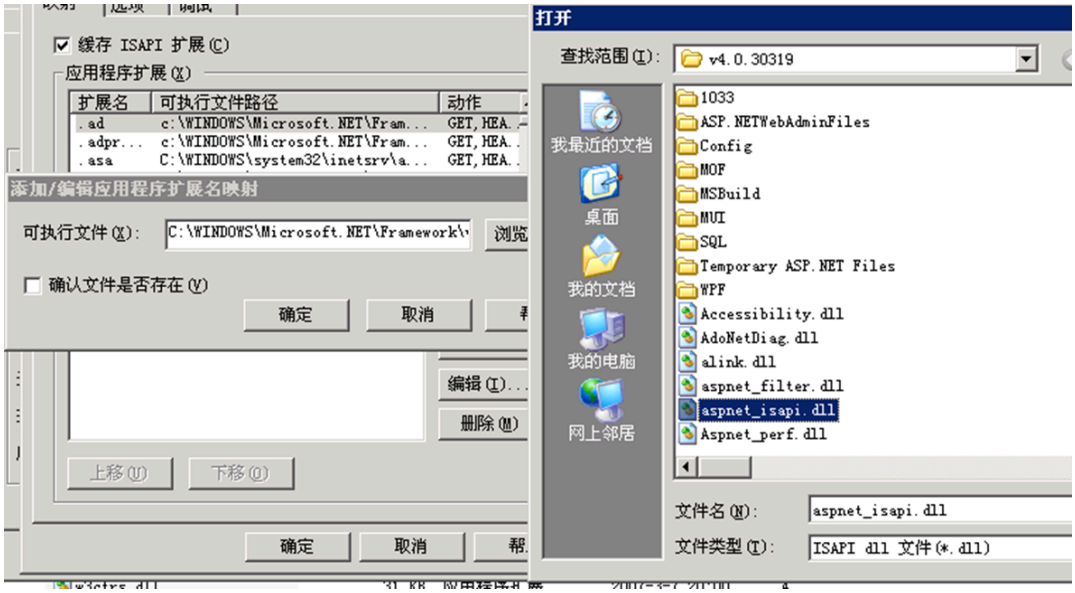
a. 设置IIS支持通配符。

A. 打开IIS信息管理器，右键单击服务网站a.cc.com的属性，选择主目录，单击配置。

B. 在弹出的对话框中，单击映射，单击插入。



C. 在弹出的对话框中，选择*C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\aspnet_isapi.dll*文件，单击确定。



说明:

不勾选确认文件是否存在。

D. 分别单击两个对话框中的确定，完成IIS通配符的支持配置。

b. 在*bin*目录下，选择*test.jpg*并右键单击，选择属性。

c. 单击HTTP头。

d. 选择此时间段后过期，设置具体的过期时间，单击确定。

e. 设置*bin*目录下其他相同扩展名文件的缓存时间，此时需要修改IIS的配置文件。具体操作如下：

A. 用记事本程序打开IIS的配置文件，配置文件在*C:\WINDOWS\system32\inetsrv\MetaBase.xml*（IIS6的设置）目录下。

B. 查找*/bin/test.jpg*，找到*bin*目录下*test.jpg*文件的缓存设置。

C. 将*test.jpg*改为**.jpg*并保存。

说明:

修改上述文件，需要在服务中关闭IIS admin Service。

f. 其他扩展名的文件缓存设置操作同上。

如问题还未解决，请提交工单。

如何设置Nginx缓存策略?

HTTP头处理模块(HTTP Headers)允许设置任意的HTTP头，您可以使用add_header和expires命令设置Nginx缓存策略。

指令	语法	默认值	使用字段
add_header	add_header name value	none	· http · server · location
expires 说明: 这个指令控制是否在应答中标记一个过期时间以及如何标记。	expires [time|epoch|max|off] 说明: · Time：控制Cache-Control的值，负数表示no-cache。 · epoch：将Expires头设置为1 January, 1970 00:00:01 GMT。 · max：将Expires头设置为31 December 2037 23:59:59 GMT，将Cache-Control最大化到10年。 · off：将禁止修改头部中的Expires和Cache-Control字段。	expires off	· http · server · location

通过expires设置，示例如下：

- 设置php的文件类型过期时间设置为1小时。

```
server {
    listen                  80;
    server_name             [redacted]
    root                    /[redacted]
    location ~* ^(.+\.php)(.*)$ {
    expires                 1h;
    fastcgi_pass            127.0.0.1:9090;
    fastcgi_index           index.php;
    fastcgi_hide_header X-Powered-By;
    fastcgi_intercept_errors on;
    fastcgi_buffers     64 16k;
    fastcgi_buffer_size 16k;
    fastcgi_busy_buffers_size 32k;
}
```

```
[redacted]# curl -D - -o /dev/null 2>/dev/null [redacted]download.php -x localhost:80
HTTP/1.1 200 OK
Server: Tengine
Date: Fri, 27 Apr 2012 14:32:37 GMT
Content-Type: text/html; charset=GB2312
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Accept-Encoding
Expires: Fri, 27 Apr 2012 15:32:37 GMT
Cache-Control: max-age=3600
```

- 设置php的文件类型为no-cache，cache服务器不缓存。

```
server {
    listen                  80;
    server_name             [redacted]
    root                    [redacted]
    location ~* ^(.+\.php)(.*)$ {
    expires                 -1;
    fastcgi_pass            127.0.0.1:9090;
    fastcgi_index           index.php;
    fastcgi_hide_header X-Powered-By;
    fastcgi_intercept_errors on;
    fastcgi_buffers     64 16k;
    fastcgi_buffer_size 16k;
    fastcgi_busy_buffers_size 32k;
}
```

```
[redacted]# curl -D - -o /dev/null 2>/dev/null [redacted]download.php -x localhost:80
HTTP/1.1 200 OK
Server: Tengine
Date: Fri, 27 Apr 2012 14:34:04 GMT
Content-Type: text/html; charset=GB2312
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Accept-Encoding
Expires: Fri, 27 Apr 2012 14:34:03 GMT
Cache-Control: no-cache
```

通过add_header设置，以动态的php文件设置为不缓存为例：

```
location ~ .*\.php$ {
    if ($request_uri !~ ^/dynamicimg/) {
        add_header              Cache-Control "no-cache";
        add_header              Pragma no-cache;
    }
  }
```

如问题还未解决，请提交工单。

CDN如何处理源站的302跳转?

根据客户终端的设备，决定对应网站的前端界面样式是常见的网站设计需求。该需求的常见设计思路是，源站根据用户的请求，在源站对用户的请求做302跳转到对应的页面上进行服务。

在对网站部署CDN后，由于CDN的产品性质，CDN会对用户的访问资源缓存到CDN的节点上，以便后续可以加快用户的访问。这种情况下，可能会出现第一个用户访问后，对相应的302请求进行缓存。而其他不同终端设备的用户通过该URL进行访问时，就会出现访问到的仍然是第一个用户缓存的302请求到的页面。这就造成用户源站对不同终端设置的适配功能失效。

解决这种问题的思路是：

- 设置对第一个请求的URL不缓存，而对302跳转后的页面进行缓存。这样设置能够保证用户源站的终端配置功能生效的同时，也可以实现CDN对于页面的加速。CDN对于缓存的配置暂时支持目录和文件后缀名，用户可以结合这两种的缓存配置以及其优先级来根据自己的站点目录结构定义初始URL不缓存。
- 在源站对于初始页面设置不缓存，因为源站的不缓存策略对于CDN是具有最高优先级的。只要该页面的response中带有下述头信息，就能保证该页面不缓存。
  - Cache-control:no-cache,no-store,private
  - Cache-control:s-maxage=0,max-age=0
  - pragma:no-cache

如问题还未解决，请提交工单。

CDN+OSS跨域访问失败的原因及处理方法是什么?

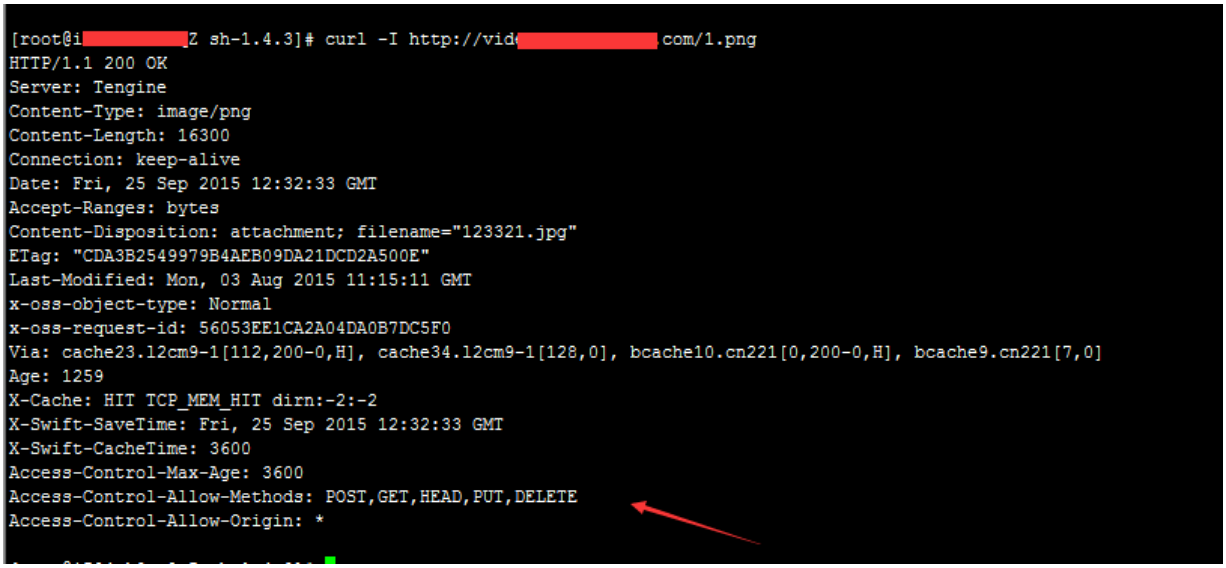
主要原因：

当您首次通过客户端浏览器访问资源时，如果CDN检测出该资源不存在，则回源进行访问。源站对比将数据，通过CDN返回给客户端浏览器。浏览器比对Access-Control-Allow-Origin后，允许正确，所以跨域正常。当您第二次访问资源时，CDN检测出存在该缓存资源，所以CDN将直接返回客户端缓存页面。由于CDN缓存了OSS未配置cors之前的文件及其头部，造成客户端浏览器判断失败，不允许访问，所以出现了跨域失败。

解决办法：

您可以通过设置Access-Control-Allow-Origin、Access-Control-Allow-Methods和Access-Control-Max-Age这三种HTTP头的参数的方式。如何设置，请参见设置HTTP响应头。

设置完成后，只要您在CDN节点访问，就会包含上述3个头部信息，不会影响正常访问。验证结果如下：



如问题还未解决，请提交工单。

为什么CDN加速导致CORS配置失效?

原因：

由于CDN加速是通过将文件缓存在节点，由节点直接返回给您，达到加速效果的，如果缓存文件未过期，即使在源站对该文件进行了变更，您访问到的依旧是之前缓存在节点的内容，而不是更新后的内容。

因此，当开启了CDN加速功能或开启了图片处理功能（默认开启CDN加速功能）后，在CDN节点上已经被访问过的文件都将被缓存，此时若您配置或变更了cors配置，则CDN已缓存的内容不会自动同步该配置更新，从而导致cors不生效。

解决方案：

建议您在变更了cors配置后，进行相关URL的缓存刷新操作，以便cors配置能够及时生效。

- 通过控制台方式进行刷新的具体操作，请参见配置刷新预热。
- 通过OpenAPI方式进行刷新的具体操作，请参见RefreshObjectCaches。

如何配置CDN支持CORS（跨域）？

配置步骤：关于CORS的具体配置步骤，请参见设置HTTP响应头。

注意事项：

- 目前不支持泛域名添加，如`*.12345.com`，仅支持域名精确匹配。
- 目前仅支持配置一条白名单域名。
- 若使用OSS产品作为源站，OSS与CDN平台同时配置CORS，CDN的配置将覆盖OSS。
- 若源站为您的服务器或ECS产品，建议先进行动静分离，静态文件使用CDN加速，CDN控制台配置的CORS功能，仅对静态文件生效。

使用CDN加速的网站如何设置CORS访问?

网站使用CDN加速后，如果某个CDN节点先发生了非跨域的访问，CDN会缓存一个没有CORS头部的文件内容，在过期之前发生的跨域访问，会因为没有CORS头部信息而导致访问报错。您可以通过自定义Header的方式设置CORS的头部信息，避免这种情况的发生。

自定义Header的具体操作步骤，请参见设置HTTP响应头。

如问题还未解决，请提交工单。

如何处理CDN回源流量较大的问题?

原因：

- 缓存命中率差，那么回源流量较大（一般缓存命中率建议在90%及以上）。
- 缓存命中率高，CDN总流量基数大，那么回源流量相对来说也较大。

对于缓存命中率差的情况，解决方法如下：

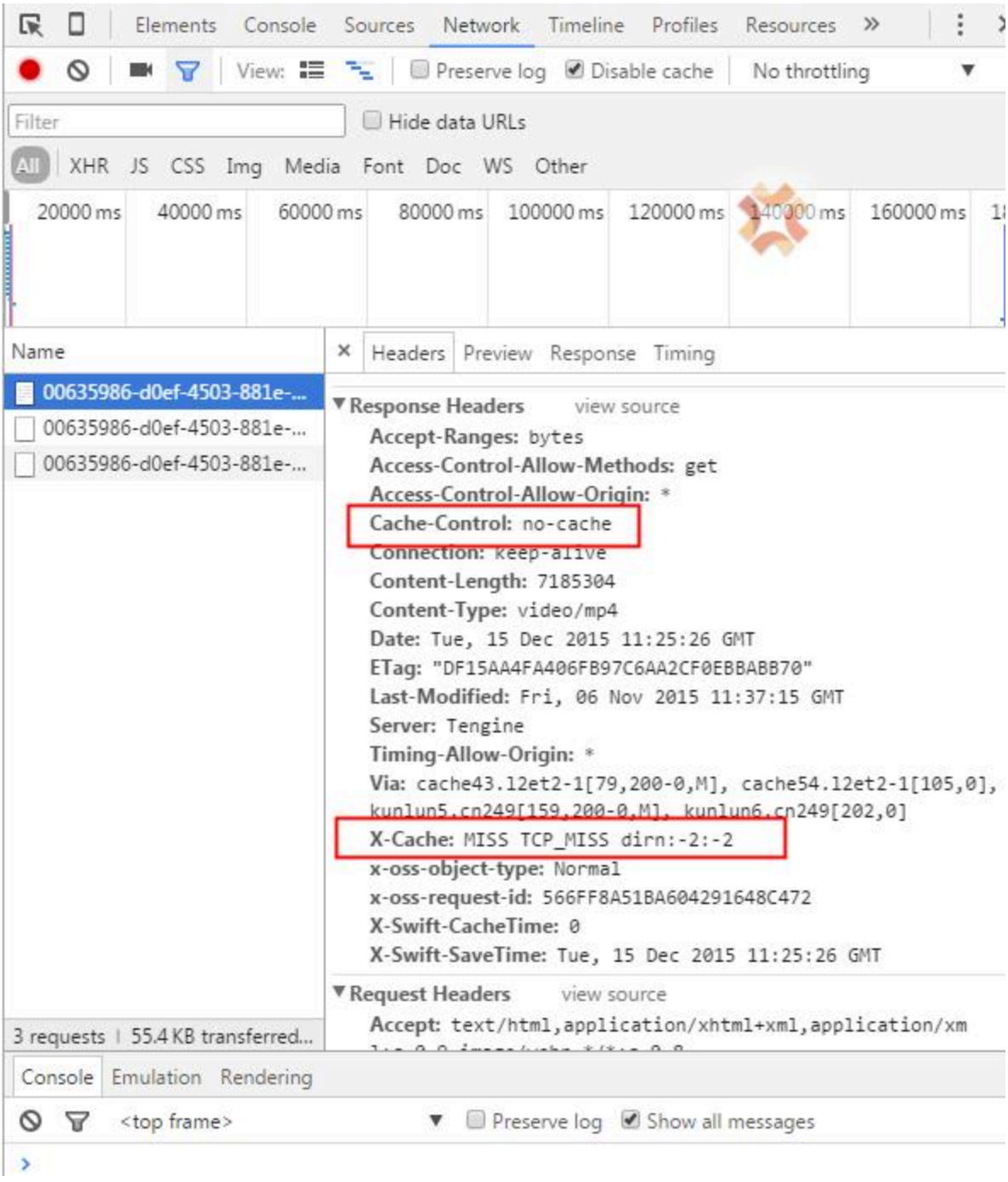
- 增加目录缓存。详细说明，请参见设置缓存过期时间。

  说明：

  - 为了确保其他未设置缓存规则的文件都能缓存命中，建议该条缓存规则设置在最下方。
  - 对于不需要缓存的，建议源站设置nocache，但不建议过多的文件设置nocache，过多的文件回源会影响加速效果。
- 排查CDN日志定位缓存总是不命中的文件。详细说明，请参见日志下载。

- 通过Google Chrome浏览器的开发者工具，定位缓存不命中的元素，排查每个元素的response头。



说明:

- X-cache：表示缓存是否命中。其中，MISS表示不命中。Hit表示命中。
- X-Swift-CacheTime：表示会在CDN一级节点中缓存多长时间。

- X-Swift-SaveTime:Tue, 15 Dec 2015 11:25:26 GMT：表示该资源缓存的具体时间点。

从图中可以看出，由于Cache-Control的值为no-cache，所以该资源缓存不命中。如果Cache-Control的值为private，也不能缓存命中。您可以定位该资源是否可以缓存，如果可以，请取消nocache。

- CDN只对get请求进行缓存，对于非get请求的资源，建议进行域名分离，只对静态资源进行CDN加速。

如问题还未解决，请提交工单。

如何处理使用CDN后网站访问速度变慢的问题?

CDN服务的主要功能是进行网站访问加速，出现使用CDN后的访问速度反而变慢的常见场景有两种场景。

- 缓存命中率不高

  影响缓存命中率的常见原因如下：

  - 缓存配置的问题。
  - 频繁的刷新UR或者目录缓存。
  - Http Header导致无法缓存。
  - 刚添加，缓存的文件还不多。
  - 网站访问量低，过期时间短，命中的文件少。
- 局部地区访问速度较慢，个别区域动态文件回源较慢。

  资源文件缓存到CDN后，CDN访问就会比源站访问快些的。对于这种情况，您可以按如下排查步骤进行定位。

  1. 测试域名解析是否正确，确保您的应用已经正常解析到CDN上。
  2. 测试域名进行访问，通过浏览器的开发者工具，测试静态页面是否已经缓存，主要看x-catch是否已经hit。hit说明已经命中，miss说明没有被缓存。
  3. 查看已经缓存的静态文件的加载时间，并通过截图标注。
  4. 将域名绑定到本地的hosts文件后，重复步骤3进行验证，对比两次的访问时间。

如问题还未解决，请提交工单。

CDN中GZIP压缩功能支持哪些格式?

目前阿里云CDN支持GZIP压缩的内容格式如下：

- content-type：text/xml
- content-type：text/plain

- content-type：text/css
- content-type：application/javascript
- content-type：application/x-javascript
- content-type：application/rss+xml
- content-type：text/javascript
- content-type：image/tiff
- content-type：image/svg+xml
- content-type：application/json

> 说明：
>
> - 源站上的资源大小需要超过1024B，才能进行GZIP压缩。
> - Internet Explorer6对GZIP的兼容性较差，如果有Internet Explorer6的访问需求，不建议您开启GZIP压缩功能。

如问题还未解决，请提交工单。

为什么CDN加速后，源站本来有的ETag字段消失了？

由于Nginx的默认行为：如果同时开启gzip和ETag，则直接不用ETag。

当您遇到这种情况时，建议排查是否开启了gzip功能。

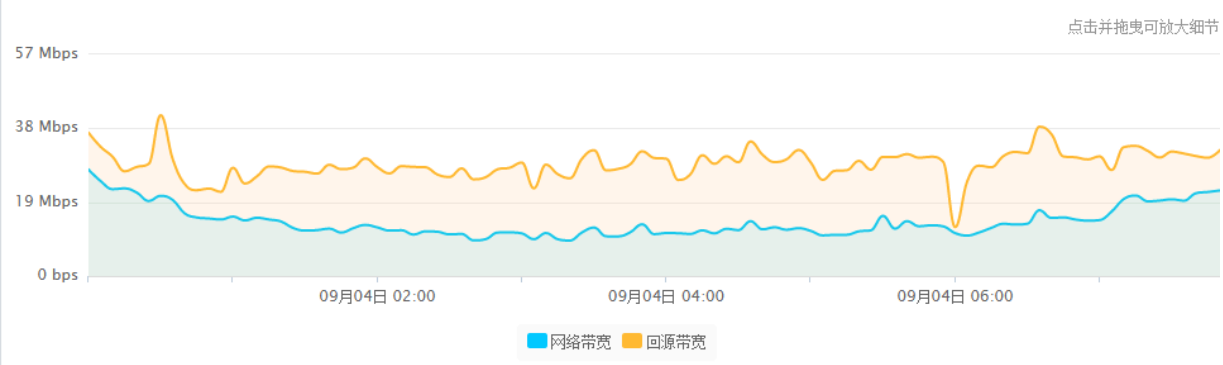
站点接入到CDN以后，为什么元素加载不了？

原因：可能CDN对应的加速域名开启了过滤参数功能。

解决办法：关闭过滤参数功能。详细说明，请参见过滤参数。

为什么CDN服务的回源流量大于访问流量？

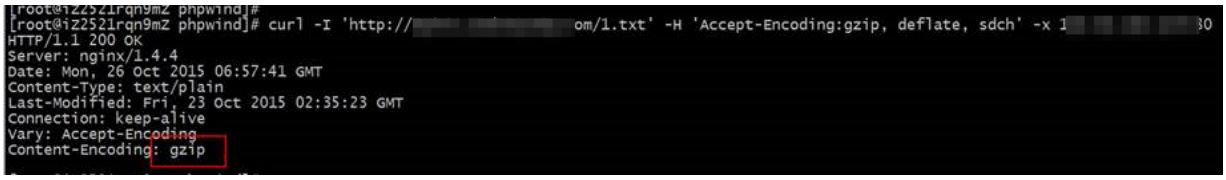
如果在使用CDN服务时，出现了回源流量大于访问流的情况，如下图：



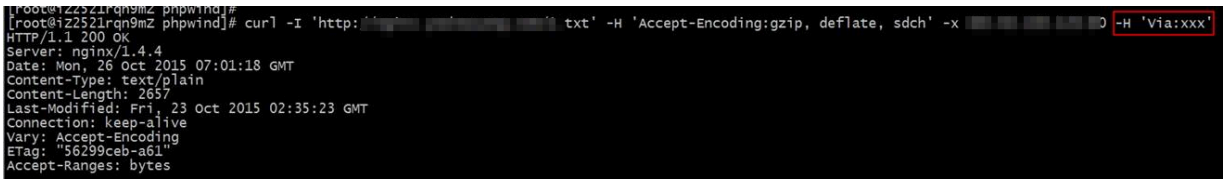
这类问题一般是由于源站未开启Gzip压缩，CDN开启了Gzip压缩导致的。

对于文本等类型的内容，Gzip的压缩比较高，所以如果源站没有开启Gzip压缩，但是客户访问CDN时，CDN进行了Gzip压缩，就可能会出现CDN回源带宽比访问带宽还高的情况，所以建议在源站上开启Gzip功能。

此外，由于使用CDN时，会添加一些CDN必须使用的header信息，如Via的header，即使源站已经设置了Gzip，如下图：



但是当在请求中带有Via的header时，源站可能会无法正确的响应Gzip，如下图：



针对这种情况，建议在源站设置全部开启Gzip的规则。以nginx为例，可以进行如下设置：

```
gzip_proxied any;
```

其它的web服务，请参考进行类似的修改。

如问题还未解决，请提交工单。

CDN基础配置中的过滤参数有什么作用?

开启过滤参数的作用：忽略URL请求中?后面的参数，提高CDN缓存的命中率。

- 开启过滤参数功能后，访问URL无需匹配?后面的参数，就可命中CDN的缓存，提高CDN的命中率。

  示例：第一次访问http://www.****.com/1.jpg时，CDN没有缓存，直接回源访问数据。第二次访问http://www.****.com/1.jpg?test1，由于开启了过滤参数，所以?后面的参数无需匹配，即可命中CDN缓存http://www.****.com/1.jpg。后续访问时，无论?后面带的是什么参数，都命中缓存http://www.****.com/1.jpg。
- 关闭过滤参数功能后，访问URL需精确匹配?后面的参数，提高请求的精确性。

  示例：第一次访问http://www.****.com/1.jpg时，CDN没有缓存，直接回源访问数据。第二次访问http://www.****.com/1.jpg?test1，由于关闭了过滤参数，所以?后面的参数需精确匹配，即无法响应CDN缓存内容http://www.****.com/1.jpg需要重新回源获取

`http://www.****.com/1.jpg?test1`。后续访问时，需要精确匹配`?`后面的参数，才能响应CDN缓存内容。

如问题还未解决，请提交工单。

16 类型6：移动加速