负载均衡

常见问题

产品文档

腾讯云

文档目录

常见问题
- 计费相关
- 负载均衡配置相关
- 健康检查异常排查
  - 健康检查异常排查
  - 健康检查异常排查v2
- HTTPS 相关
- WS/WSS 协议支持相关
- HTTP/2 协议支持相关
- 默认域名阻断提示

常见问题
计费相关

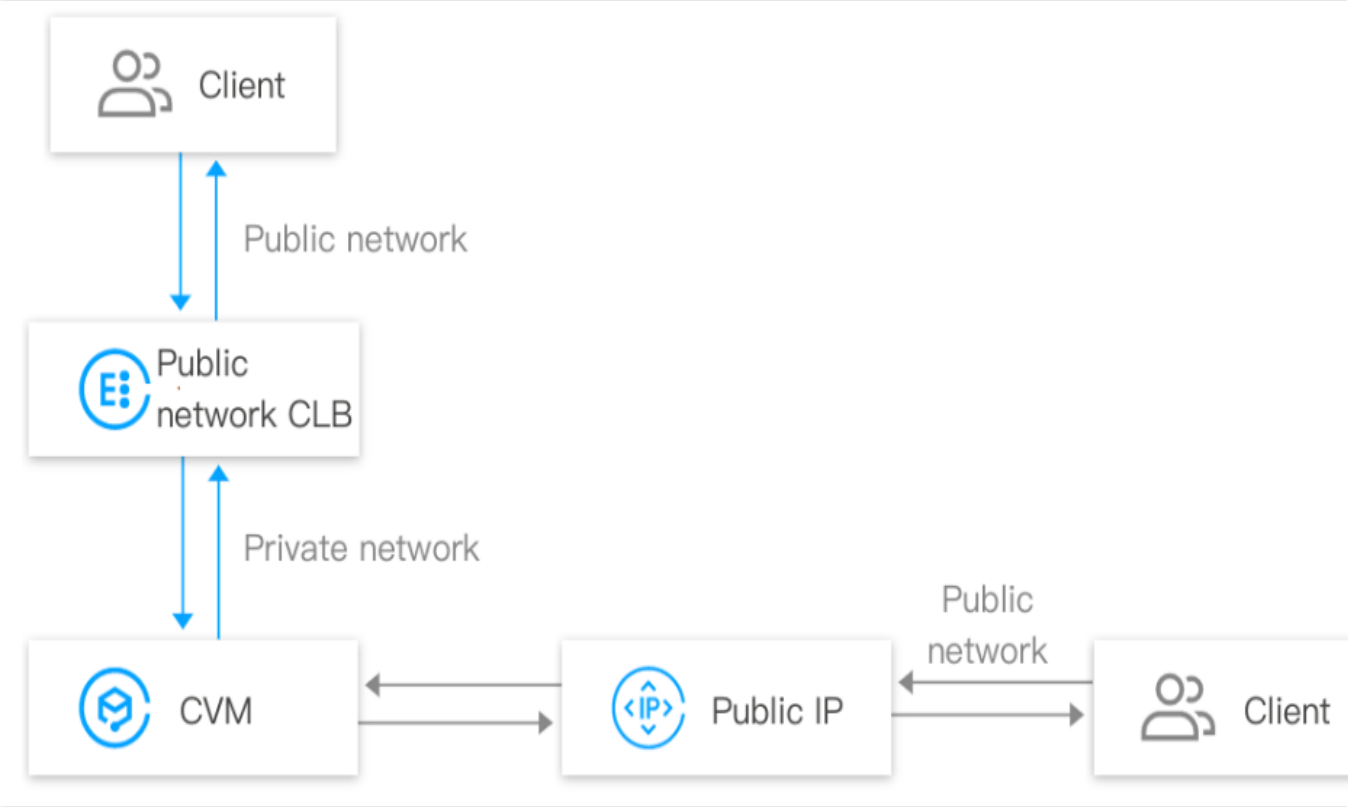
最近更新时间：2024-01-04 17:32:01

计费相关问题

CLB 和后端云服务器（CVM）之间是使用公网还是内网通信？
CLB 如何收费？
标准账户类型和传统账户类型可以互相切换吗？
跨地域绑定费用如何收取？
CLB 的带宽上限是多少？
CLB 计费相关的监控指标都有哪些呢？

CLB 和后端云服务器（CVM）之间是使用公网还是内网通信？

不论您的账户类型是标准账户类型还是传统账户类型，CLB 和 CVM 之间都是使用内网流量。两种账户类型的网络流量路径皆如下所示：
客户端使用公网访问 CLB，CLB 通过内网将流量转发给 CVM，CVM 的响应仍然通过内网返回给 CLB，CLB 再回复给客户端。
CVM 主动访问公网（或者直接通过公网 IP 被动访问）时，CVM 通过公网 IP（或弹性公网 IP）和客户端交互。



[回到顶部]

CLB 如何收费？

CLB 的费用根据不同类型的账户收费不同，详情请参见 计费概述。标准账户类型和传统账户类型的主要区别在于计费统计维度不同，其余业务访问和监控等都是一致的。不同账户类型的差异请参见 账户类型。
标准账户类型的公网网络费用：访问 CLB 的公网流量，其公网网络费仅在 CLB 上收取，不在 CVM 上收取；通过公网 IP 访问的流量，其公网网络费仅在公网 IP 上收取，不在 CVM 上收取。
传统账户类型的公网网络费用：访问 CLB 和公网 IP 的公网流量，其公网网络费均在 CVM 上收取，不在 CLB 和公网 IP 上收取。

[回到顶部]

标准账户类型和传统账户类型可以互相切换吗？

传统账户类型可以升级为标准账户类型。
标准账户类型不能回退为传统账户类型。

[回到顶部]

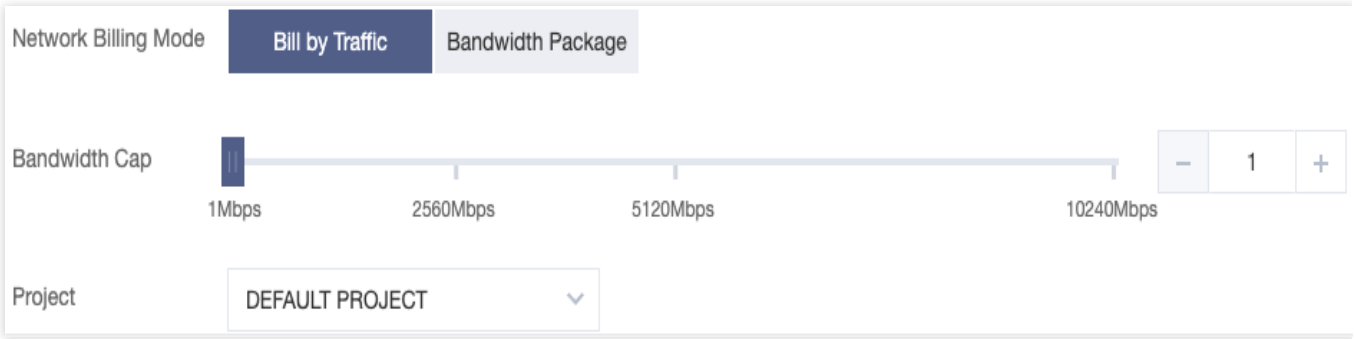
跨地域绑定费用如何收取？

仅在您的 CLB 使用了跨地域绑定功能时，才会收取跨域费用，否则就不收取。
跨地域1.0：在 CLB 上收取跨地域费用。
跨地域2.0：通过云联网来收取跨地域费用，CLB 上不再收取跨地域费用。

[回到顶部]

CLB 的带宽上限是多少？

标准账户类型：您在购买 CLB 时，需为 CLB 选择带宽上限，当业务带宽超过该上限时，CLB 会自动做丢包处理。



传统账户类型：您在购买 CLB 时，无需为 CLB 选择带宽上限，CLB 的带宽受限于后端 CVM 的公网总带宽。

[回到顶部]

CLB 计费相关的监控指标都有哪些呢？

和 CLB 计费（包括网络费用和 LCU 费用等）相关的计费指标可参考以下监控指标：

费用类型	指标英文名	指标中文名
网络费用	AccOuttraffic	LB 到后端的出流量
	OutTraffic	LB 到后端的出带宽
	InTraffic	LB 到后端的入带宽
LCU 费用	ClientConnum	客户端到 LB 的连接数
	ClientNewConn	客户端到 LB 的新建连接数

	TotalReq	每秒请求数
	ClientAccOuttraffic	客户端到 LB 的出流量
	ClientAccIntraffic	客户端到 LB 的入流量

和 CLB 限速（包括带宽限速和 LCU 限速等）相关的限速指标可参考以下监控指标：

限速类型	指标英文名	指标中文名
带宽限速	ClientOuttraffic	客户端到 LB 的出带宽
	ClientIntraffic	客户端到 LB 的入带宽
LCU 限速	ClientConcurConn	客户端到 LB 的并发连接数
	ClientNewConn	客户端到 LB 的新建连接数
	TotalReq	每秒请求数
	ClientOuttraffic	客户端到 LB 的出带宽
	ClientIntraffic	客户端到 LB 的入带宽

CLB 监控指标详情请参见 监控指标说明，网络计费详情请参见 网络费用，LCU 计费详情请参见 性能容量单位 LCU 费用。

[回到顶部]

负载均衡配置相关

最近更新时间：2024-01-04 17:32:00

概念相关问题
四层负载均衡和七层负载均衡有什么区别？
UDP 协议与 TCP 协议有什么区别?
负载均衡 Cookies 会话保持方式的原理是什么？
什么是后端服务器权重？
重置为0与解绑 RS 有什么区别？
健康检查相关问题
健康检查提示异常该如何处理
为什么健康检查探测频率过高？
访问相关问题
负载转发中的 HTTP 重定向问题
可以为哪些 TCP 端口执行负载均衡？
发送 843 的 policy 请求（即 flash server 请求）时，没有返回策略文件，连接直接断掉，该如何处理？
负载均衡是否可以直接获取 Client 端 IP？
CVM 可通过配置内网型负载均衡，将流量从端口A转发回同一台服务器的其他端口吗？
后端 CVM 需要公网带宽吗？是否会影响负载均衡的服务？
客户端、服务器端 HTTP 版本不一致时，兼容版本说明
支持 Gzip 兼容性
负载均衡后端服务器的安全组应该怎么设置？怎样设置访问黑名单？
设置访问黑名单
负载均衡与后端服务器之间的通讯是走的内网还是外网？
关于 Ping 负载均衡的 VIP 说明
关于 Telnet 负载均衡监听端口的说明
关于内网回环问题的说明
关于同一个客户端通过不同的中间节点访问同一个后端服务器的同一个端口时串流问题的说明
为什么后端 CVM 配置安全组禁止公网访问，仅允许负载均衡访问，但却未生效？
为什么负载均衡已配置监听器并绑定后端 CVM，当配置域名解析至后端 CVM 的 IP 并访问域名时，负载均衡的监控没有具体的信息？
为什么没有创建843监听器，却可以 Telnet 通？
CLB 的访问日志里记录的9/11网段的地址是腾讯云内网网段吗？

四层负载均衡和七层负载均衡有什么区别？

四层均衡能力，是基于 IP + 端口的负载均衡。

七层是基于应用层信息（如 HTTP 头部、URL 等）的负载均衡。
四到七层负载均衡，就是在对后台的服务器进行负载均衡时，依据四层的信息或七层的信息来决定怎么样转发流量。
例如，四层的负载均衡，就是通过发布三层的IP地址（VIP），然后加四层的端口号，来决定哪些流量需要做负载均衡，对需要处理的流量进行 NAT 处理，转发至后台服务器，并记录下这个 TCP 或者 UDP 的流量是由哪台服务器处理的，后续这个连接的所有流量都同样转发到同一台服务器处理。
七层的负载均衡，就是在四层的基础上，再考虑应用层的特征。
例如，同一个 Web 服务器的负载均衡，除了根据 VIP 和80端口辨别是否需要处理的流量，还可根据七层的 URL、浏览器类别、语言来决定是否要进行负载均衡。
七层负载均衡，也称为“内容交换”，也就是主要通过报文中的真正有意义的应用层内容，再加上负载均衡设备设置的服务器选择方式，决定最终选择的内部服务器。
七层负载均衡要根据真正的应用层内容选择服务器，只能先代理最终的服务器和客户端建立连接(三次握手)后，才可能接受到客户端发送的真正应用层内容的报文，然后再根据该报文中的特定字段，以及负载均衡设备设置的服务器选择方式，决定最终选择的内部服务器。负载均衡设备在这种情况下，更类似于一个代理服务器。负载均衡和前端的客户端以及后端的服务器会分别建立 TCP 连接。

[回到顶部]

UDP 协议与 TCP 协议有什么区别？

TCP 是面向连接的协议，在正式收发数据前，必须和对方建立可靠的连接。UDP 是面向非连接的协议，它在数据发送前不与对方先进行三次握手，而是直接进行数据包发送传送。UDP 协议主要适用于关注实时性而相对不注重可靠性的场景，如视频聊天、金融实时行情推送、DNS、物联网等。

[回到顶部]

负载均衡 Cookies 会话保持方式的原理是什么？

在 Cookie 插入模式下，CLB 将负责插入 Cookie，后端服务器无需作出任何修改。当客户进行第一次请求时，客户 HTTP 请求（不带 Cookie）进入 CLB， CLB 根据负载平衡算法策略选择后端一台服务器，并将请求发送至该服务器，后端服务器进行 HTTP 回复（不带 Cookie）被发回 CLB，然后 CLB 插入 Cookie，将 HTTP 回复（带 Cookie）返回到客户端。
当客户请求再次发生时，客户 HTTP 请求（带有上次 CLB 插入的 Cookie）进入 CLB，然后 CLB 读出 Cookie 里的会话保持数值，将HTTP请求（带有与上面同样的 Cookie）发到指定的服务器，然后后端服务器进行请求回复，由于服务器并不写入 Cookie，HTTP 回复将不带有 Cookie，回复流量再次经过进入 CLB 时，CLB 再次写入更新后的会话保持 Cookie。

[回到顶部]

什么是后端服务器权重？

用户可以指定后端服务器池内各 CVM 的转发权重，权重比越高的 CVM 将被分配到更多的访问请求，用户可以根据后端 CVM 的对外服务能力和情况来区别设定。
如果您同时开启了会话保持功能，那可能会造成对后端应用服务器的访问并不是完全相同的，建议您可以暂时关闭会话保持功能观察一下是否依然存在这种情况。

[回到顶部]

权重置为0与解绑 RS 有什么区别？

权重置为0：TCP 监听器存量连接继续转发，UDP 监听器相同五元组的继续转发，HTTP/HTTPS 监听器存量连接继续转发。TCP、UDP、HTTP/HTTPS 监听器新增连接不会再转发到权重为 0 的 RS 上。
解绑 RS：TCP/UDP 监听器存量连接立即停止转发，HTTP/HTTPS 监听器存量连接继续转发，存量连接转发完毕之后断开与 RS 的连接。

[回到顶部]

健康检查提示异常该如何处理？

请按如下步骤进行排查：
确保您直接通过后端服务器访问到您的应用服务。
确保后端服务器已开启了相应的端口。
检查后端服务器内部是否有防火墙之类的防护软件，可能导致负载均衡系统无法与后端服务器通讯。
检查负载均衡检查参数设置是否正确。
建议使用静态页面来健康检查。
检查后端的云服务器是否有高负载导致云服务器对外响应慢。
确保云服务器子机没有做 iptables 限制。

[回到顶部]

为什么健康检查探测频率过高？

健康检查探测包频率过高，控制台设置接受探测包5秒1次，实际后端服务器发现1秒内收到1次甚至多次健康检查请求，原因如下：
当前，健康检查频率过高的问题，主要跟负载均衡后端健康探测实现机制有关。假设100万的 client 端请求，会分散在4台 CLB 后端物理机上，再转给后端服务器。 健康检查探测是在 CLB 的后端物理机上各自探测的。因此，CLB 实

例设置5秒1次的探测请求，实际上 CLB 后端的每台物理机都会每5s发送一次探测。因此在后端服务器上，会收到多次探测请求。假设 CLB 实例所在集群有8台物理机，那么每台机器5s发送一次请求，后端主机可能会在5s中收到8次探测。
该实现方案的优势是：效率高，探测精准，避免误剔除。例如，CLB 实例集群的8台物理机中，其中1台判断失败，仅那1台机器不再转发流量，另外7台的流量是正常的。
因此，如果您后端服务器的探测频率过高，可以通过设置更长的探测间隔时间来解决（如设置为15s探测一次）。

[回到顶部]

负载转发中的 HTTP 重定向问题

当浏览器访问网站 `http://example.com` 时，对服务器而言需要进行一次重定向，判断需要定向至根目录。而当浏览器访问网站 `http://example.com/` 时服务器会直接返回网站设置的根目录默认页面。同样的，假设 `http://cloud.tencent.com/movie` 被 URL 重写跳转到 `http://cloud.tencent.com/movie/` 上的话，则输入 `http://cloud.tencent.com/movie` 就会多一次 URL 重写的过程，在性能和时间上都有微小的损耗，但在结果上没有差别。若 `http://cloud.tencent.com/product` 被 URL 重写转跳到非 `http://cloud.tencent.com/product/` 同一页面上，则需要考虑是否在二级页面后添加 `/` 。
在腾讯云负载均衡中，如果前后端端口号不一致时，为了避免 HTTP 重定向后导致端口号更改，访问二级页面需要加 `/` 保证页面的正常访问。
假设七层转发下，负载均衡实例监听80端口，后端服务器监听 8081 端口。此时客户端访问 `http://www.example.com/movie` ，经由负载均衡转发至后端服务器，服务器收到发往 `http://www.example.com/movie` 的请求并会重定向到 `http://www.example.com:8081/movie/` （监听端口为8081），此时客户端访问失败（端口错误）。
因此，建议用户将访问请求改写为带 `/` 的二级页面如 `http://www.example.com/movie/` ，这样可以避免 HTTP 重定向，减少一次不必要的判断，降低不必要的负载。如果必须使用 HTTP 重定向时，请保证负载均衡的监听端口和后端服务器的监听端口相同。

[回到顶部]

可以为哪些 TCP 端口执行负载均衡？

您可以为如下 TCP 端口执行负载均衡：21（FTP）、25（SMTP）、80（HTTP）、443（HTTPS），以及1024 - 65535等端口。

[回到顶部]

发送 843 的 policy 请求（即 flash server 请求）时，没有返回策略文件，连接直接断掉，该如何处理？

负载均衡收到 843 的 policy 请求，会主动回复通用的 crossdomain 策略配置文件，如果出现没有返回策略文件，连接直接断掉的情况，可能是 flash server 请求不正确。
请确认发送正确的 flash server 的请求：\\0 。
注意：
这里需要以\\0结尾，一共23个字节。\\0是指一个 ASCII 码为 0 的符号，只占用一个字节。
正常的 843 返回结果如下图所示：

```
VM_02_sles10_64:/ # perl -e 'printf "<policy-file-request/>%c",0' | netcat -i 1 101.226.62.63 8
<?xml version="1.0"?>
<!DOCTYPE cross-domain-policy SYSTEM "/xml/dtds/cross-domain-policy.dtd">

<!-- Policy file for xmlsocket://socks.example.com -->
<cross-domain-policy>

   <!-- This is a master socket policy file -->
   <!-- No other socket policies on the host will be permitted -->
   <site-control permitted-cross-domain-policies="master-only"/>

   <!-- Instead of setting to-ports="*", administrator's can use ranges and commas -->
   <!-- This will allow access to ports 123, 456, 457 and 458 -->
   <allow-access-from domain="*" to-ports="*" />

</cross-domain-policy>
```

[回到顶部]

负载均衡是否可以直接获取 Client 端 IP？

IPv6 NAT64 负载均衡不支持获取 Client IP。
公网七层 IPv4 和 IPv6 负载均衡提供 X-Forwarded-For 的方式获取访问者真实 IP，负载均衡侧默认开启，需要后端服务做相应配置来获取 Client IP。详情请见 如何获取客户端真实 IP。
公网四层 IPv4 和 IPv6 负载均衡（TCP 协议）服务可以直接在后端 CVM 上获取来访者真实 IP 地址，无需进行额外的配置；内网四层负载均衡自从2016年10月24日起，新购的实例不再进行 SNAT 处理，支持直接从 server 端获取真实的 client IP，无需额外配置。

[回到顶部]

CVM 可通过配置内网型负载均衡，将流量从端口A转发回同一台服务器的其他端口吗？

不可以。对服务器 A（10.66.*.101*）*端口 a 的访问可通过内网型负载均衡将请求转发至服务器 B（10.66.*.102）的端口 b。但无法将流量转发至同一台服务器 A（10.66.* .101）的另一端口 b。

[回到顶部]

后端 CVM 需要公网带宽吗？是否会影响负载均衡的服务？

标准账户类型的负载均衡绑定的后端 CVM 无需配置公网带宽。
传统账户类型的负载均衡不收取任何的流量或带宽费用。负载均衡服务产生的公网流量费用，由绑定的后端的 CVM 收取，建议购买后端 CVM 时，公网带宽选择按使用流量计费，并设定合理的最高的带宽峰值上限，这样就无需关注 CLB 出口的总流量的涨跌。互联网 Web 业务的流量起伏较大，无法准确预测。若按带宽计费，带宽买多了不划算，买得太少，业务高峰期会出现丢包的情况。

[回到顶部]

客户端、服务器端 HTTP 版本不一致时，兼容版本说明

转发兼容性

前端（client 端），当前支持 HTTP1.0/1.1，向下兼容。
后端（server 端），当前腾讯云使用 HTTP1.0 协议，支持 HTTP1.0/1.1，向下兼容。
注意：
HTTP/2 只在 HTTPS 中支持，且 client 及 server 端可以向下兼容。当前不支持 HTTP 协议。

支持 Gzip 兼容性

前端（client 端），当前支持 HTTP1.0/1.1 向下兼容。（用户无需配置，主流浏览器都支持 Gzip）
后端（server 端），在云服务器端，由于腾讯云内部全网支持 HTTP/1.1 协议，因此用户也无需配置，使用 Nginx 默认配置（HTTP/1.1）即可兼容。
注意：
HTTP/2 只在 HTTPS 中支持，但 Gzip 可以用在腾讯云所支持的任意 HTTP 版本中。

[回到顶部]

负载均衡后端服务器的安全组应该怎么设置？怎样设置访问黑名单？

负载均衡安全组配置

若后端服务器设置了安全组规则，可能会出现负载均衡实例无法与其通信的状况。因此，在四层转发和七层转发下，建议后端服务器安全组均设置为全放通。若打开了安全组，并默认允许全协议全ip段的地址访问时，需要配置所有客户端 IP 到本机 IP 的安全组规则。
对于某些恶意 IP，可以设置把恶意IP加在安全组前排规则，禁止其访问后端服务器；再放通所有 IP（0.0.0.0）到本机服务端口，让正常客户端可以访问。（安全组规则是有顺序的，自顶而下进行匹配）。

私有网络内的七层负载转发若设置了健康检查，必须把负载均衡 VIP 加入到后端服务器的安全组放通规则，否则健康检查可能失效。

设置访问黑名单

如用户需要给某些 Client IP 设置黑名单，拒绝其访问，可以通过配置云服务关联的安全组实现。安全组的规则需要按照如下步骤进行配置：

注意

如下配置步骤有顺序要求，顺序相反会导致黑名单配置失效。

将需要拒绝访问的 client IP + 端口添加至安全组中，并在策略栏中选取拒绝该 IP 的访问。

设置完毕后，再添加一条安全组规则，默认开放该端口全部 IP 的访问。

配置完成后，安全组规则如下：

```
clientA ip+port drop
clientB ip+port drop
0.0.0.0/0+port accept
```

关于安全组的更多说明，请参见 后端云服务器安全组配置说明。

[回到顶部]

负载均衡与后端服务器之间的通讯是走的内网还是外网？

负载均衡与后端服务器的通讯始终走内网，绑定的 CVM 有外网 IP 的情况下也一样。

[回到顶部]

关于 Ping 负载均衡的 VIP 说明

Ping 负载均衡的 VIP ：由负载均衡集群响应，不会转发到后端的服务器。

公网负载均衡的 VIP 支持 Ping。

内网负载均衡的 VIP 仅支持来自本 VPC 的客户端 Ping，来自其他 VPC、本地 IDC 的客户端 Ping 由于是代答且不能反应真实链路，因此无法保障 Ping 通。（如使用云联网、对等连接等打通 VPC 的场景，建议您使用 Telnet 来探测）。

[回到顶部]

关于 Telnet 负载均衡监听端口的说明

创建四层（TCP、UDP、TCP SSL） 监听器后，如果不绑定后端服务器，则无法 Telnet 通监听端口；绑定后端服务器后，可以 Telnet 通监听端口。
创建七层（HTTP、HTTPS）监听器后，即使不绑定后端服务器，也可以 Telnet 通监听端口，由 CLB 代答。

[回到顶部]

关于内网回环问题的说明

内网负载均衡不支持同一个 CVM 既作为客户端又作为服务器，此时 CLB 看到的 Client IP 和 Server IP 是一样的，会导致访问不通。
当您的客户端需要同时作为服务器时，请至少绑定两个后端服务器。CLB 有自动避免回环的策略，当 Client A 访问负载均衡时，负载均衡会自动调度到非 Client A 的后端服务器上。

[回到顶部]

关于同一个客户端通过不同的中间节点访问同一个后端服务器的同一个端口时串流问题的说明

问题现象

同一个客户端在同一时刻，通过不同的中间节点访问同一个后端服务器的同一个端口会出现串流现象。具体场景如下：
同一个客户端，同时通过同一个 CLB 的四层、七层监听器，访问同一个后端服务器的同一个端口。
同一个客户端，同时通过不同 CLB 的不同监听器，访问同一个后端服务器的同一个端口。
访问内网 CLB 的客户端比较集中，且后端服务相同时，有较大概率会出现串流。（访问公网 CLB 的客户端来源较广，很少出现串流。）

问题原因

当前 CLB 会透传客户端 IP 到后端服务器，因此会导致 `client_ip:client_port -> vip:vport -> rs_ip:rs_port` 最终变为 `client_ip:client_port --> rs_ip:rs_port` 。

解决方案

分散客户端：使用多个客户端发起访问。
收敛 CLB：在满足业务功能和容灾需求的前提下，减少 CLB 的实例、监听器个数。
分散后端服务端口：后端服务使用多个端口提供服务，避免后端端口集中。
分散部署：不同 CLB 绑定不同的后端服务端口，如 CLB1 绑定一组 CVM，CLB2 绑定另一组 CVM，两个 CLB 同时提供访问。

[回到顶部]

为什么后端 CVM 配置安全组禁止公网访问，仅允许负载均衡访问，但却未生效？

若需要通过 CLB 访问后端 CVM，需要后端 CVM 和 CLB 两个安全组均放通公网访问，建议先把后端 CVM 的安全组仅放通 CLB 的 VIP 公网访问，CLB 的安全组按需放通公网访问 IP。

[回到顶部]

为什么负载均衡已配置监听器并绑定后端 CVM，当配置域名解析至后端 CVM 的 IP 并访问域名时，负载均衡的监控没有具体的信息？

需通过负载均衡进行访问，才有具体的监控信息，可配置域名解析至负载均衡的 VIP 并访问域名，即可在负载均衡的监控中查看具体的信息。

[回到顶部]

为什么没有创建843监听器，却可以 Telnet 通？

为满足部分用户重置访问 Flash 的需求，CLB 默认放通了843端口。若您想关闭该端口，请配置 TCP:843 监听器后，不绑定后端服务器即可。

[回到顶部]

CLB 的访问日志里记录的9/11网段的地址是腾讯云内网网段吗？

是的。腾讯云负载均衡 CLB 产品使用9/11网段的地址作为内网网段地址。

[回到顶部]

健康检查异常排查

健康检查异常排查

最近更新时间：2024-01-04 17:32:01

负载均衡（CLB）通过健康检查来判断后端服务的可用性。若您遇到健康检查异常，可参考以下方式进行排查。

说明：

当健康检查探测到异常时，CLB 将不再向异常后端服务转发流量。

当健康检查探测到所有后端服务都有异常时，请求将会被转发给所有后端服务。

健康检查原理可参考 健康检查。

检查后端服务器的公网带宽

传统账户类型，负载均衡绑定的后端 CVM 需要配置公网带宽，否则会导致健康检查异常。因为该账户的带宽属性在 CVM 上，而非 CLB 上。

标准账户类型，负载均衡绑定的后端 CVM 无需配置公网带宽，且不会影响负载均衡服务。

说明

若您无法确定账户类型，请参见 判断账户类型。

传统账户类型的负载均衡不收取任何流量或带宽费用。负载均衡服务产生的公网流量费用，由绑定的后端 CVM 收取。

您可在不分配公网 IP 的情况下，为 CVM 购买公网带宽。

检查安全组配置

检查负载均衡实例是否开启安全组默认放通功能。如果未开启，则需在 CVM 的安全组上放通来源 IP。如果您的 CLB 服务支持任意 IP 的访问，则在安全组的入站规则中配置来源 IP 为0.0.0.0/0。详情请参考 配置负载均衡安全组。

检查四层监听器

说明

TCP 协议下，负载均衡使用 SYN 包进行探测。

UDP 协议下，负载均衡使用 `ping` 命令进行探测。

在页面查看 CLB 后端服务器端口的健康状态，状态为异常时的排查方法如下：

确定 CLB 后端服务器是否配置了安全组导致影响服务。后端服务器可通过安全组进行访问控制从而保证服务正常运行，详情请参考 后端云服务器安全组配置说明。
使用 `netstat` 命令，检查后端服务器的端口是否有进程在监听。若未发现进程，则重新启动服务。

检查七层协议

针对七层（HTTP 协议）服务，当某一监听出现健康检查“异常”时，可以通过以下方面进行排查：
由于负载均衡的七层健康检查服务与后端 CVM 之间通过内网通信，您需要登录服务器检查应用服务器端口是否正常监听在内网地址上，如果没有监听在内网地址，请将应用服务器端口监听到内网上，从而确保负载均衡系统和后端 CVM 之间的正常通信。
假设负载均衡前端端口是80，CVM 后端端口也是80，CVM 内网 IP 是：`1.1.1.10` 。
Windows 系统服务器使用如下命令：

```
netstat -ano | findstr :80
```

Linux 系统服务器使用如下命令：

```
netstat -anp | grep :80
```

如果可以看到 `1.1.1.10:80` 的监听或 `0.0.0.0:80` 的监听则说明此配置正常。
请确保后端服务器开启了您在负载均衡监听器中配置的后端端口。
如果是四层负载均衡，只要后端端口 `telnet` 有响应即可，可以使用 `telnet 1.1.1.10 80` 来测试。
如果是七层负载均衡，需要 HTTP 状态码是200 等代表正常的状态码，检验方法如下：
Windows 系统可以直接在 CVM 内的浏览器输入内网 IP 测试是否正常，本例为：`http://1.1.1.10` 。
Linux 系统可以通过 `curl -I` 命令查看状态是否为 HTTP/1.1 200 OK，本例为：`curl -I 1.1.1.10` 。
检查后端 CVM 内部是否有防火墙或其他安全类防护软件，这类软件很容易将负载均衡系统的本地 IP 地址屏蔽，从而导致负载均衡系统无法跟后端服务器进行通信。
检查服务器内网防火墙是否放行80端口，可以暂时关闭防火墙进行测试。
Windows 系统可以在运行输入 `firewall.cpl` 命令关闭。
Linux 系统可以输入 `/etc/init.d/iptables stop` 命令关闭（CenOS 7.x 系统请运行 `systemctl stop firewalld` 命令）。
检查负载均衡健康检查参数设置是否正确，建议参考 健康检查 提供的健康检查参数默认值进行设置。
健康检查指定的检测文件，建议是以 HTML 形式的简单页面，只用于检查返回结果。不建议使用 PHP 等动态脚本语言。
检查后端是否有较高负载导致 CVM 对外提供服务响应慢。
检查 HTTP 请求方式。
如果使用 HEAD 方法，则后端服务一定要支持 HEAD。
如果是 GET 方法，则后端服务一定要支持 GET。

如果同时开启了 TCP 的快速回收（tcp_tw_recycle）和时间戳（tcp_timestamps）可能导致健康检查异常，建议关闭 tcp_tw_recycle，详见 原因分析。

健康检查探测频率过高

控制台设置5s接收1次探测包，实际后端服务器发现1s内收到1次甚至多次健康检查请求，导致健康检查探测频率过高的原因主要是和负载均衡的后端健康探测实现机制有关：
假设100万的 Client 端请求，会分散在4台 CLB 后端物理机上，再转发给后端服务器。 健康检查探测是在 CLB 的各个后端物理机上分别进行探测，因此，CLB 实例设置5s1次的探测请求时，实际上 CLB 后端的每台物理机都会每5s发送一次探测。此时后端服务器上可能会在5s中收到4次探测请求。

该方案的优势是效率高，探测精准，避免误剔除。例如，CLB 实例集群的8台物理机中，其中1台判断失败，那么仅此台机器不再转发流量，另外7台仍然正常转发流量。

如果您的业务对负载敏感性高，高频率的健康检查探测可能会对正常业务访问造成影响，您可以通过增大探测时间间隔的方式来降低对业务的影响（例如设置为15s探测一次）。

一个后端服务器绑定在多个 CLB 实例上时，每个 CLB 实例都会发送健康探测报文用于探测该服务器是否健康，从而导致健康探测的频率较高。

健康检查异常排查v2

最近更新时间：2024-12-20 12:14:13

负载均衡 CLB 通过健康检查来判断后端服务的可用性。若您遇到健康检查异常，可参考以下方式进行排查。

说明：

当健康检查探测到异常时，CLB 将不再向异常后端服务转发流量。

当健康检查探测到所有后端服务都有异常时，请求将会被转发给所有后端服务。

健康检查原理，可参考 健康检查概述。

一、排查子机安全组与 ACL 拦截

注意：

若已配置安全组默认放通则可忽略。

步骤1：查看实例健康探测源 IP

1. 登录 负载均衡控制台，单击需要查看健康探测源 IP 的**实例 ID**。
2. 在实例详情页，单击**监听器管理**页签，单击**监听器**，再右侧**展开**监听器详情。

3. 在**监听器详情**页，即可查看到当前健康检查源 IP，如上示例健康检查源 IP 为100.64.0.0/10网段。

步骤2：确认安全组放通健康探测源 IP

1. 登录 负载均衡控制台，单击负载均衡**实例 ID**。
2. 在负载均衡示例详情页，单击**安全组**页签 > 已绑定的**安全组 ID**，进入安全组规则页面。

3. 在**入站规则**页签下，单击**添加规则**。
4. 在添加入站规则弹窗中，在来源中输入 查看实例健康探测源 IP 中的100.64.0.0/10网段（若第一步中确认的健康探测源 IP 为负载均衡 VIP，则将该 VIP 填写在**来源**处），协议端口填写后端服务器使用的协议端口，策略选择**允许**，单击**确定**，完成添加。

步骤3：确认子网的网络 ACL 放通健康探测源 IP

1. 登录 云服务器控制台，单击**云服务器实例**，进入基本信息页。
2. 在**基本信息**页，单击网络信息模块中的**所属子网**，跳转至子网信息页。

3. 单击 **ACL 规则**页签，在该页面，单击已绑定的 ACL，并在**入站规则**和**出站规则**中，放通健康探测源 IP。
4. 若第一步中，确认的健康探测源 IP 为100.64.0.0/10网段（若确认的健康探测源 IP 为**负载均衡 VIP**），则将其填写在源 IP 处，协议类型填写健康检查方式中选择的协议，端口填写 **ALL**，策略选择**允许**，单击**保存**完成添加。

说明：

如果负载均衡绑定了 COS、CDB、Redis、Ckafka 等公共服务，需要检查服务绑定的安全组及所在子网的网络 ACL 是否有放通负载均衡健康检查源 IP，可以参考上述三个步骤排查。

步骤4：确认 IDC 放通 SNAT IP

如果用户通过云联网 CCN 或者专线产品绑定 IDC 内的机器作为负载均衡实例的后端服务器，需要确认 IDC 放通 SNAT IP。

1. 登录 负载均衡控制台，单击负载均衡**实例 ID**。
2. 在实例基本信息页面，在后端服务模块中，查看 SNAT IP。
3. 用户需检查 IDC 内防火墙设备或者机器 iptables 是否有放通该 SNAT IP。

二、排查云服务器 CVM

若后端服务器为 CVM，可参考以下步骤进行排查。

步骤1：机器内部自查

1. 登录 云服务器控制台，进入机器内部，检查服务端进程和端口。

检查 CLB 配置对应后端服务器端口，示例为 80 端口检查命令。

```
netstat -anltu | grep -w 80
```

2. 若返回80端口处于监听状态，则可以排除机器内部异常。

注意：

监听地址只能是0.0.0.0或者 CVM 的内网 IP，若监听地址仅有127.0.0.1，则不能排除机器内部异常。

```
tcp        0      0 0.0.0.0:80              0.0.0.0:*
LISTEN      9/nginx: master pro
tcp6       0      0 :::80                   :::*
LISTEN      9/nginx: master pro
```

步骤2：检查 CVM 能否正常返回

1. 用同 VPC 的其他机器，检查目标 CLB 后端 CVM 的 HTTP/HTTPS 端口是否正常返回。

例如 CLB 控制台配置的 location 目录是“/”，HTTP 端口检查后端 CVM 的内网 IP，以 IP 10.0.0.16 ，端口80为例。

```
curl -I http://10.0.0.16:80/
```

2. 实际判断响应结果是否正常，以控制台配置的响应状态码为准。例如配置的响应状态码是“200”或“404”，该返回结果为正常情况，可以排除本异常点。

```
HTTP/1.1 200 OK
Server: nginx/1.20.1
Date: Sat, 14 Sep 2024 07:07:01 GMT
Content-Type: text/html
```

```
HTTP/1.1 404 Not Found
Server: nginx/1.20.1
Date: Sat, 14 Sep 2024 07:08:51 GMT
Content-Type: text/html
```

步骤3：检查 iptables 是否放通

1. 检查方法请参考：防火墙问题，检查命令如下：

```
iptables -nvL
```

2. 若确认被拦截，则需要添加命令放通健康探测源 IP 和 CLB 监听器配置的后端服务器端口。以健康探测源 IP 100.64.0.0/10，后端服务器端口为80和443端口为例。

```
iptables -A INPUT -p tcp -s 100.64.0.0/10 --dport 80 -j ACCEPT
iptables -A INPUT -p tcp -s 100.64.0.0/10 --dport 443 -j ACCEPT
iptables -A INPUT -p icmp -s 100.64.0.0/10 -j ACCEPT
```

基于不同 Linux 执行以下命令：

```
#Centos/RHEL：
sudo systemctl enable iptables
sudo service iptables save
```

```
#Ubuntu/Debian：
sudo systemctl enable netfilter-persistent
sudo netfilter-persistent save
```

3. 放通后可再次执行检查命令排查。

说明：

后端协议为 HTTPS 的场景，健康检查异常建议修改为 HTTP。

若业务后端要求使用 HTTPS 协议，则请参考 Nginx 服务器 SSL 证书安装部署（Linux）进行 SSL 配置和检查 。若仍存在问题，请 提交工单处理。

仅 CLB 上配置 HTTPS 监听器，且后端协议为 HTTPS，才需在后端服务配置证书。

三、排查容器

若后端服务器为容器，可参考以下步骤进行排查，以绑定 TKE 集群为例。

在 TKE 容器场景下，CLB 的后端服务器可以分为直连 pod 和非直连（即 CLB 绑定 nodeport）两个场景。判断是否直连的方式如下，详情请参见：使用 LoadBalancer 直连 Pod 模式 Service-TKE 标准集群指南。

对于 service 有配置 service.cloud.tencent.com/direct-access: "true"注解，则为直连。

对于 ingress 有配置 ingress.cloud.tencent.com/direct-access: "true"注解，则为直连。

步骤1：CLB 直连 pod 场景

CLB 直连 pod 场景下，CLB 流量直接转发到后端 pod。

排查路径如下：

1. 检查容器内监听端口

登录容器后，参考 机器内部自查 进行检查。

登录容器的方式可参考：远程终端基本操作。

2. 检查容器本地访问自己正常

登录容器后，参考 检查 CVM 能否正常返回 进行检查。

登录容器的方式可参考：远程终端基本操作。

3. 检查从 pod 所在 node 访问 pod 正常

如果 pod 不是运行在超级节点上，可登录 node 后，参考 手动测试。

普通节点登录可参考：使用标准登录方式登录 Linux 实例（推荐）。

原生节点登录可参考：原生节点开启 SSH 密钥登录。

4. 检查 node 内部配置

4.1 检查 ip_forward

输入检查命令（若为 ipv6，则需把命令中 ipv4 换成 ipv6）：

```
sysctl net.ipv4.ip_forward
```

正常结果：

```
net.ipv4.ip_forward = 1
```

异常结果：

```
net.ipv4.ip_forward = 0
```

异常结果解决命令：

```
sysctl -w "net.ipv4.ip_forward=1" && echo 'net.ipv4.ip_forward=1' >>/etc/sysctl.con
```

4.2 检查网卡 forward

输入检查命令：

```
sysctl -a 2>/dev/null | grep ipv4 | grep -w forwarding
```

正常结果全部参数值皆为 1，如：

```
net.ipv4.conf.all.forwarding = 1
```

正常结果完整示例：

```
# sysctl -a 2>/dev/null | grep ipv4 | grep -w forwarding
net.ipv4.conf.all.forwarding = 1
net.ipv4.conf.cbr0.forwarding = 1
net.ipv4.conf.default.forwarding = 1
net.ipv4.conf.docker0.forwarding = 1
net.ipv4.conf.eth0.forwarding = 1
net.ipv4.conf.eth1.forwarding = 1
net.ipv4.conf.lo.forwarding = 1
net.ipv4.conf.veth44256889.forwarding = 1
net.ipv4.conf.vethb678241b.forwarding = 1
```

异常结果存在参数值为 0，如：

```
net.ipv4.conf.all.forwarding = 0
```

异常结果处理命令，如：（根据实际异常 net.xxx.forwarding 条目执行以下命令）

```
sysctl -w net.ipv4.conf.all.forwarding=1
```

4.3 检查 node 的 iptables 是否拦截 forward

输入检查命令：

```
iptables -nvL FORWARD
```

输出结果：

policy 后的策略应为 ACCEPT。如果为 DROP，则可能导致 forward 拦截。

只有 KUBE-FORWARD、KUBE-SERVICES、KUBE-EXTERNAL-SERVICES和DOCKER-USER 这四条规则。如果有其他规则，则可能导致 forward 拦截。

以下为正常的示例：

```
# iptables -nvL FORWARD
# Warning: iptables-legacy tables present, use iptables-legacy to see them
Chain FORWARD (policy ACCEPT 8 packets, 480 bytes)
 pkts bytes target     prot opt in     out     source               destination
  444  169K KUBE-FORWARD  all  --  *      *       0.0.0.0/0            0.0.0.0/0            /* kubernetes forwarding rules */
    8   480 KUBE-SERVICES  all  --  *      *       0.0.0.0/0            0.0.0.0/0            ctstate NEW /* kubernetes service portals */
    8   480 KUBE-EXTERNAL-SERVICES  all  --  *      *       0.0.0.0/0            0.0.0.0/0            ctstate NEW /* kubernetes externally-visible service portal
    8   480 DOCKER-USER  all  --  *      *       0.0.0.0/0            0.0.0.0/0
```

4.4 检查安全组是否放通

如果 pod 是 vpc-cni 模式，需检查 node 的弹性网卡安全组是否放通，否则需要检查 node 本身的安全组是否放通。可通过开启 CLB 默认放通或参考 确认安全组放通健康探测源 IP 进行放通。

步骤2：CLB 非直连场景

CLB 非直连场景下，CLB 流量先转发给集群内 node 的 nodeport 端口，再经过 iptables/ipvs转发，将进入到 nodeport 的流量转发给真正的后端 pod，链路较长。

排查路径：

1. **检查 CLB 直连 pod 场景相关内容**

检查 CLB 直连 pod 场景相关内容，在此基础上继续完成后续检查步骤。

2. **检查 node 的安全组放通情况**

检查节点的安全组和 VPC-CNI 模式 pod 的安全组是否参考以下文档放通：容器服务安全组设置。

普通节点的安全组设置可参考：配置安全组。

原生节点的安全组设置可参考：修改原生节点。

超级节点的安全组设置可参考：新建超级节点 和 超级节点可调度 Pod 说明。

VPC-CNI 模式 pod 的安全组设置可参考：VPC-CNI 模式安全组使用说明。

3. **检查健康异常节点上 kube-proxy 组件是否正常运行**

kube-proxy 组件用于 iptables/ipvs 规则下发。检查方法如下：

```
# 获取节点上 kube-proxy pod，并确实是否 Ready
kubectl get pod -n kube-system -l k8s-app=kube-proxy -owide | grep <节点名>
# 查看 kube-proxy 运行日志是否有明显报错
kubectl logs -n kue-system <kube-proxy-xxxxx名>
```

若存在异常，可参考 集群 Kube-Proxy 异常排障处理 处理。

4. **登录健康检查异常的 CLB 后端节点上，逐一访问后端 pod**

TCP 监听可用 telnet 测试连通性，HTTP/HTTPS 监听器可用 curl 测试访问结果。详情请参见 检查 TCP 服务连通性、检查 HTTP/HTTPS 服务返回。

手动测试的补充说明

步骤1：检查端口监听状态

可以通过 netstat、ss 等命令确认端口的监听状态。若返回监听地址仅有 127.0.0.1，则不能排除异常。

1. 通过 netstat 命令检查端口是否监听状态，以 80 端口为例：

```
netstat -tulnp | grep 80
```

输出存在以下内容即可视为监听状态：

```
tcp        0      0 0.0.0.0:80              0.0.0.0:*
LISTEN      9/nginx: master pro
```

```
tcp6       0      0 :::80                   :::*
LISTEN      9/nginx: master pro
```

2. 通过 ss 命令检查端口是否监听状态，以 80 端口为例：

```
ss -tulnp | grep 80
```

输出存在以下内容即可视为监听状态：

```
tcp    LISTEN     0      511        *:80                    *:*
users:(("nginx",pid=9,fd=6))
tcp    LISTEN     0      511     [::]:80                 [::]:*
users:(("nginx",pid=9,fd=8))
```

步骤2：检查 TCP 服务连通性

可以通过 telnet 命令检查 TCP 服务的连通性。

注意：

请勿使用低版本 busybox 的 telnet 测试，因为无论连接是否正常，都不会回显。

以查看 IP 172.16.1.29 的 80 端口为例：

```
echo "" |telnet 172.16.1.29 80
```

输出 Connected 为正常连通；停留在 Trying 为网络不通，需检查安全组等（具体请参见 确认子网的网络 ACL 放通健康探测源 IP、检查 iptables 是否放通 章节）；返回 Connection refused 为端口未监听。

```
Trying 172.16.1.29...
Connected to 172.16.1.29.
Escape character is '^]'.
Connection closed by foreign host.
```

步骤3：检查 HTTP/HTTPS 服务返回

可以通过 curl 命令检查服务返回的 HTTP 状态码。

以请求协议 HTTP、方法 GET、域名是 mydomain.com、路径 /health、端口8080、IP 172.16.1.29 为例。

```
curl -X GET -H "Host: mydomain.com" http://172.16.1.29:8080/health -s -o /dev/null
```

响应结果：

```
httpcode: 404
```

若在健康检查配置正常状态码中选择了预期返回 1xx-4xx，以上响应结果返回 404 符合预期。若返回结果不符合健康检查配置预期，但实际为正常情况，建议调整预期配置。

以上异常排查若仍未解决您的问题，请提交 工单处理。

HTTPS 相关

最近更新时间：2024-01-04 17:32:00

HTTP 相关问题

HTTPS 支持的加密套件有哪些？
HTTPS 支持哪些版本的SSL/TLS安全协议？
HTTPS 监听使用什么端口？
为什么需要 HTTPS 双向认证？
为什么 HTTPS 协议实际产生的流量会比账单流量多一些？
添加 HTTPS 监听器后，负载均衡到后端服务器间的请求是否依然通过 HTTP 协议传输？

证书问题

CLB 目前支持哪些类型的证书？
一个监听器可以绑定多少个 HTTPS 证书？
一个证书可以应用于多少个负载均衡器，多少个监听器？
如何上传证书？
证书区分地域吗？
证书需要上传到后端服务器吗？
证书过期后如何处理？
添加证书报错如何处理？

HTTPS 支持的加密套件有哪些？

ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-CHACHA20-POLY1305:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:AES128:AES256:AES:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!PSK

[回到顶部]

HTTPS 支持哪些版本的SSL/TLS安全协议？

负载均衡 HTTPS 目前支持的 ssl_protocols：TLSv1、TLSv1.1、TLSv1.2、TLSv1.3。

[回到顶部]

HTTPS 监听使用什么端口？

不强制，建议使用443端口。

[回到顶部]

为什么需要 HTTPS 双向认证？

有些客户对数据安全要求较高，如涉及到金融服务的客户等。他们不仅需要在服务端进行 HTTPS 认证，在客户端也需要进行 HTTPS 认证，为了满足这些客户的需求，我们推出 HTTPS 双向认证功能。

[回到顶部]

为什么 HTTPS 协议实际产生的流量会比账单流量多一些？

如果用户使用 HTTPS 协议，将会使用一些流量用于协议握手，因此其实际产生的流量会比账单流量更多一些。

[回到顶部]

添加 HTTPS 监听器后，负载均衡到后端服务器间的请求是否依然通过 HTTP 协议传输？

是的。添加 HTTPS 监听器后，客户端到负载均衡之间的请求将经过 HTTPS 协议加密，而负载均衡到后端服务器依然通过 HTTP 协议传输，因此后端服务器无需做 SSL 配置。

[回到顶部]

CLB 目前支持哪些类型的证书？

目前支持服务器证书和 CA 证书的上传，服务器证书需要上传证书内容和私钥，CA 证书只需要上传证书内容；这两种类型的证书都只支持 PEM 编码格式的上传。

[回到顶部]

一个监听器可以绑定多少个 HTTPS 证书？

如果用户使用 HTTPS 单向认证，则一个监听只能绑定一个服务器证书；若用户使用 HTTPS 双向认证，则一个监听需要绑定一个服务器证书＋一个 CA 证书。

[回到顶部]

一个证书可以应用于多少个负载均衡器，多少个监听器？

一个证书可以应用于一个或多个负载均衡器，或多个监听器。

[回到顶部]

如何上传证书？

可以通过 API 或负载均衡控制台两种方式上传。

[回到顶部]

证书区分地域吗？

不区分。证书购买和颁发后，安装部署不受地域的限制。

[回到顶部]

证书需要上传到后端服务器吗？

不需要，负载均衡 HTTPS 提供证书管理系统管理和存储用户证书，证书不需要上传到后端 CVM，用户上传到证书管理系统的私钥都会加密存储。

[回到顶部]

证书过期后如何处理？

当前证书过期后，需要用户手动更新证书。

[回到顶部]

添加证书报错如何处理？

可能是私钥内容错误，需要用户替换为新的满足需求的证书。

[回到顶部]

WS/WSS 协议支持相关

最近更新时间：2024-01-04 17:32:00

产

品内容

什么是 WS/WSS?
为什么要使用 WS/WSS?
产品购买
WS/WSS 如何收费?
产品实施
如何在 CLB 上开启 WS/WSS?
支持 WS/WSS 的地域有哪些?

什么是 WS/WSS？

WebSocket 是一种在单个 TCP 连接上进行全双工通讯的协议。
WebSocket 使得客户端和服务器之间的数据交换变得更加简单，允许服务端主动向客户端推送数据。在 WebSocket API 中，浏览器和服务器只需要完成一次握手，两者之间就直接可以创建持久性的连接，并进行双向数据传输。

[回到顶部]

为什么要使用 WS/WSS？

在 WebSocket 出现之前，客户端获取服务器数据只能通过轮询的方式从服务器拉取(Pull)数据。
这样的数据交换方式存在两个最突出的问题：
1. 效率低。当客户端需要实时数据时，需要频繁的发起 Ajax 请求拉取数据。
2. 服务器无法主动推(Push)数据。
为解决这些问题，WebSocket 诞生了。WebSocket 是伴随 HTML5 发布的一种新协议。它实现了浏览器与服务器全双工通信(full-duplex)，可以传输基于消息的文本和二进制数据。从协议层面上解决了 HTTP 的上述难题。

WebSocket 的主要优点包括：
1. 更小的控制开销。连接建立后，用于控制的包头较小。相对于 HTTP 请求每次都要携带完整的头部，此项开销大大降低。
2. 更强的实时性。WebSocket 是全双工协议，服务器可实时推动数据给客户端。
3. 保持连接状态。

[回到顶部]

WS/WSS 如何收费？

CLB 默认支持 WS/WSS，不收取额外费用。

[回到顶部]

如何在 CLB 上开启 WS/WSS？

CLB 默认开启 WS/WSS。如果连接空闲超过60s时，则需要进行个性化配置，配置 `proxy_read_timeout` 参数，建议参数值小于900s，配置方式请参见 七层个性化配置。
监听器监听在 HTTP 或 TCP SSL，则默认支持 WS；监听器监听 HTTPS，则默认支持 WSS。
使用 WSS 时，CLB 会进行 SSL 卸载。
[回到顶部]

支持 WS/WSS 的地域有哪些？

目前**所有地域**均已支持 WS/WSS 协议。

[回到顶部]

HTTP/2 协议支持相关

最近更新时间：2024-01-04 17:32:00

产品内容

什么是 HTTP/2？
为什么要使用 HTTP/2？

产品购买

如何收费？

产品实施

如何在 CLB 上开启 HTTP/2？
支持的 HTTP/2 地域有哪些？

什么是 HTTP/2？

HTTP/2（超文本传输协议第2版），是 HTTP 协议的第二个主要版本，应用于 Web 服务。
HTTP/2 的设计目标是，解决 HTTP1.X 中的性能问题，更有效的利用网络资源，减少网络应用的延迟。
HTTP/2 向下兼容 HTTP1.X。
[回到顶部]

为什么要使用 HTTP/2？

相比于 HTTP1.X，HTTP/2 响应更快，效率更高，具备如下优势：
多路复用：并行处理，响应更快。
服务端推送：服务端主动推动客户端所需资源，减少请求次数。
更多功能包括：流量控制、请求优先级、头部压缩、二进制分帧等。
[回到顶部]

如何收费？

CLB 支持 HTTP/2 不收取额外费用。
[回到顶部]

如何在 CLB 上开启 HTTP/2？

注意：

HTTP 监听器不支持 HTTP/2。主流浏览器和 WebServer 仅支持基于 TLS 的 HTTP/2 协议。

CLB 与后端服务器之间仍使用 HTTP1.X 协议。

1. 在 HTTPS 监听器开启 HTTP/2

负载均衡型实例：您可以选择开启或关闭 CLB 对 HTTP/2 的支持，详情请参考 配置 HTTPS 监听器 。

传统型负载均衡型实例：2018年4月之前创建的 HTTPS 监听器无法启用 HTTP/2，2018年4月后创建的 HTTPS 监听器可以启用 HTTP/2。传统型负载均衡不支持修改 HTTP/2 的开关。

2. 客户端访问时共识协议

当客户端访问已启动 HTTP/2 的监听器时，在HTTPS 的握手过程中，会进行协议版本的协商。客户端使用 ALPN（应用层协议协商）通知服务端自身可支持的协议列表，服务端根据协议列表选择 HTTP/2 或 HTTP1.X，若客户端不支持 HTTP/2 ，则自动向下兼容，无需额外配置。

[回到顶部]

支持的 HTTP/2 地域有哪些？

全部地域均已支持 HTTP/2。

[回到顶部]

默认域名阻断提示

最近更新时间：2024-04-16 14:44:44

根据国家相关法律法规要求，腾讯云平台将不允许用户直接使用平台侧提供的默认域名进行访问，以广州为例，如果检测到 `xxxxxxx-gz-tencentclb.com` 将拒绝访问。
平台侧推荐您在DNS平台将自定义域名通过 CNAME 的方式解析到平台侧提供的默认域名上，操作详情可参见 配置负载均衡的转发域名。

配置自定义域名的优势
增强品牌形象：将自定义域名配置到您的服务中，自定义域名是您的个性化域名，增强品牌形象和专业性，增加用户信任度。
预防域名拦截：一些应用或平台可能会对负载均衡的默认域名进行拦截。而您绑定了自定义域名，可以确保您的服务始终能够被正常访问。
提升访问体验：使用自定义域名访问服务，让您的用户方便记忆，相比使用默认域名，更简洁友好，轻松访问和分享。
保障服务平滑：绑定自定义域名到服务后，即使后续服务有变化，用户仍可使用相同域名访问您的服务，确保链接的持久性，长期可访问。