负载均衡

运维指南

产品文档

腾讯云

文档目录

运维指南
客户端 timewait 过多解决方案

最近更新时间：2022-11-29 16:59:47

本文背景

客户压测 CLB 时，常会遇到一些客户端 timewait 过多，端口被快速占满，导致 connect 失败的问题，下面会说明原因和解决方案。

Linux 参数介绍

tcp_timestamps： 是否开启 tcp timestamps 选项，timestamps 是在 tcp 三次握手过程中协商的，任意一方不支持，该连接就不会使用 timestamps 选项。
tcp_tw_recycle： 是否开启 tcp time_wait 状态回收。
tcp_tw_reuse： 开启后，可直接回收超过1s的 time_wait 状态的连接。

原因分析

客户端timewait太多，是因为客户端主动断开连接，客户端每断开一个连接，该连接都会进入timewait状态，默认60s超时回收。一般情况下，遇到这种场景时，客户会选择打开 tcp_tw_recycle 和 tcp_tw_reuse 两个参数，便于回收timewait状态连接。
然而当前 CLB 没有打开 tcp_timestamps 选项，导致客户端打开的 tcp_tw_recycle 和 tcp_tw_reuse 都不会生效，不能快速回收 timewait 状态连接。下面会解释几个 Linux 参数的含义和 CLB 不能开启 tcp_timestamps 的原因。

1. tcp_tw_recycle 和 tcp_tw_reuse只有在 tcp_timestamps 打开时才会生效。
2. 在 FullNAT 场景下，tcp_timestamps和tcp_tw_recycle是不能同时打开的，因为公网客户端经过 NAT 网关访问服务器，会存在问题，原因如下：
   tcp_tw_recycle/tcp_timestamps 都开启的条件下，60s内同一源 IP 主机的 socket connect 请求中的 timestamp 必须是递增的。以2.6.32内核为例，具体实现如下：

```
if(tmp_opt.saw_tstamp && tcp_death_row.sysctl_tw_recycle &&
(dst = inet_csk_route_req(sk,req))!= NULL &&
(peer = rt_get_peer((struct rtable *)dst))!= NULL &&
peer->v4daddr == saddr){
if(get_seconds()< peer->tcp_ts_stamp + TCP_PAWS_MSL &&
(s32)(peer->tcp_ts - req->ts_recent) > TCP_PAWS_WINDOW){
```

```
NET_INC_STATS_BH(sock_net(sk),LINUX_MIB_PAWSPASSIVEREJECTED);
goto ↓drop_and_release;
}
}
```

> 说明：
> tmp_opt.saw_tstamp：该 socket 支持 tcp_timestamp。
> sysctl_tw_recycle：本机系统开启 tcp_tw_recycle 选项。
> TCP_PAWS_MSL：60s，该条件判断表示该源 IP 的上次 tcp 通讯发生在60s内。
> TCP_PAWS_WINDOW：1，该条件判断表示该源 IP 的上次 tcp 通讯的 timestamp 大于本次 tcp。

3. CLB（7层）关闭了 tcp_timestamps 原因，因为公网客户端经过 NAT 网关访问服务器，可能会存在问题，如下例：
a) 某五元组还是 time_wait 状态。NAT网关对端口的分配策略，2MSL内复用了同个五元组，发来syn包。
b) 在开启 tcp_timestamps 情况下，同时满足如下两个条件，会丢弃该 syn 包（因为开启了时间戳选项，认为是老包）。
i. 上次时间戳 > 本次时间戳。
ii. 24天内收过包（时间戳字段是32位，Linux 默认1ms更新一次时间戳，24天会发生时间戳回绕）。

> 说明：
> 在移动端该问题更为明显，因为客户端都是在运营商NAT网关下面共享有限的公网 IP，五元组还可能在2MSL内被复用，不同客户端传来的时间戳不能保证是递增的。

以2.6.32内核为例，具体实现如下：

```
static inline int tcp_paws_check(const struct tcp_options_received *rx_opt,int paws_win)
{
if((s32)(rx_opt->ts_recent - rx_opt->rcv_tsval)<= paws_win)
return 1;
if(unlikely(get_seconds()>=rx_opt->ts_recent_stamp + TCP_PAWS_24DAYS))
return 1;
return 0;
}
```

> 说明：
> rx_opt->ts_recent：上次的时间戳。
> rx_opt->rcv_tsval：本次收到的时间戳。
> get_seconds（）： 当前时间。
> rx_opt->ts_recent_stamp： 上次收到包的时间。

解决方案

客户端 Timewait 过多问题，有如下解决方案：

- HTTP 使用短连接（Connection: close），这时由 CLB 主动关闭连接，客户端不会产生 timewait。
- 如果场景需要使用长连接，可以打开 socket 的 SO_LINGER 选项，使用 rst 关闭连接，避免进入 timewait 状态，达到快速回收端口的目的。

负载均衡 HTTPS 服务性能测试

最近更新时间：2022-11-29 15:18:00

1. CLB 负载均衡器 HTTPS 能力说明

腾讯云 CLB 负载均衡器通过对协议栈及服务端的深度优化，实现了 HTTPS 性能的巨大提升。同时，我们也通过证书的国际合作，极大降低了证书的成本。腾讯云 CLB 在如下几个方面能够为业务带来非常显著的收益：

1. 使用 HTTPS 并不会降低 Client 端的访问速度。
2. 集群内单台服务器 SSL 加解密性能，高达6.5W cps的完全握手。相比高性能 CPU 提升了至少3.5倍，节省了服务端成本，极大提升了业务运营及流量突涨时的服务能力，增强了计算型防攻击的能力。
3. 支持多种协议卸载及转换。减少业务适配客户端各种协议的压力，业务后端只需要支持 HTTP1.1 就能使用 HTTP2，SPDY，SSL3.0，TLS1.2 等各版本协议。
4. 一站式 SSL 证书申请、监控、替换。我们和国际证书厂商 comodo，symantec 展开对话，探讨合作，大幅缩减证书申请流程及成本。
5. 防 CC 及 WAF 功能。能够有效杜绝慢连接、高频定点攻击、SQL 注入、网页挂马等应用层攻击。

2. 测试目的

HTTPS 服务拥有身份验证，信息加密及完整性校验等优势，但通过新增 SSL 协议实现安全通信，必然会产生一定的性能损耗，主要包括延时的增加及加解密消耗 CPU 资源等方面。本文测试了腾讯云 HTTPS 服务在 SSL 加解密情况下的极限性能数据，供用户与 HTTPS 传统性能数据进行比对和参考。

3. 测试环境

- 压力工具：wrk 4.0.2
- 腾讯云底层服务环境：Nginx 1.1.6_1.9.9 + Openssl 1.0.2h
- 安装Nginx机器操作系统信息：Linux TENCENT64.site 3.10.94-1-tlinux2-0036.tl2 #1 SMP Thu Jan 21 03:40:59 CST 2016 x86_64 x86_64 x86_64 GNU/Linux
- 其他压力机器操作系统：Linux TENCENT64.site 2.6.32.43-tlinux-1.0.17-default #1 SMP Tue Nov 17 18:03:12 CST 2015 x86_64 x86_64 x86_64 GNU/Linux

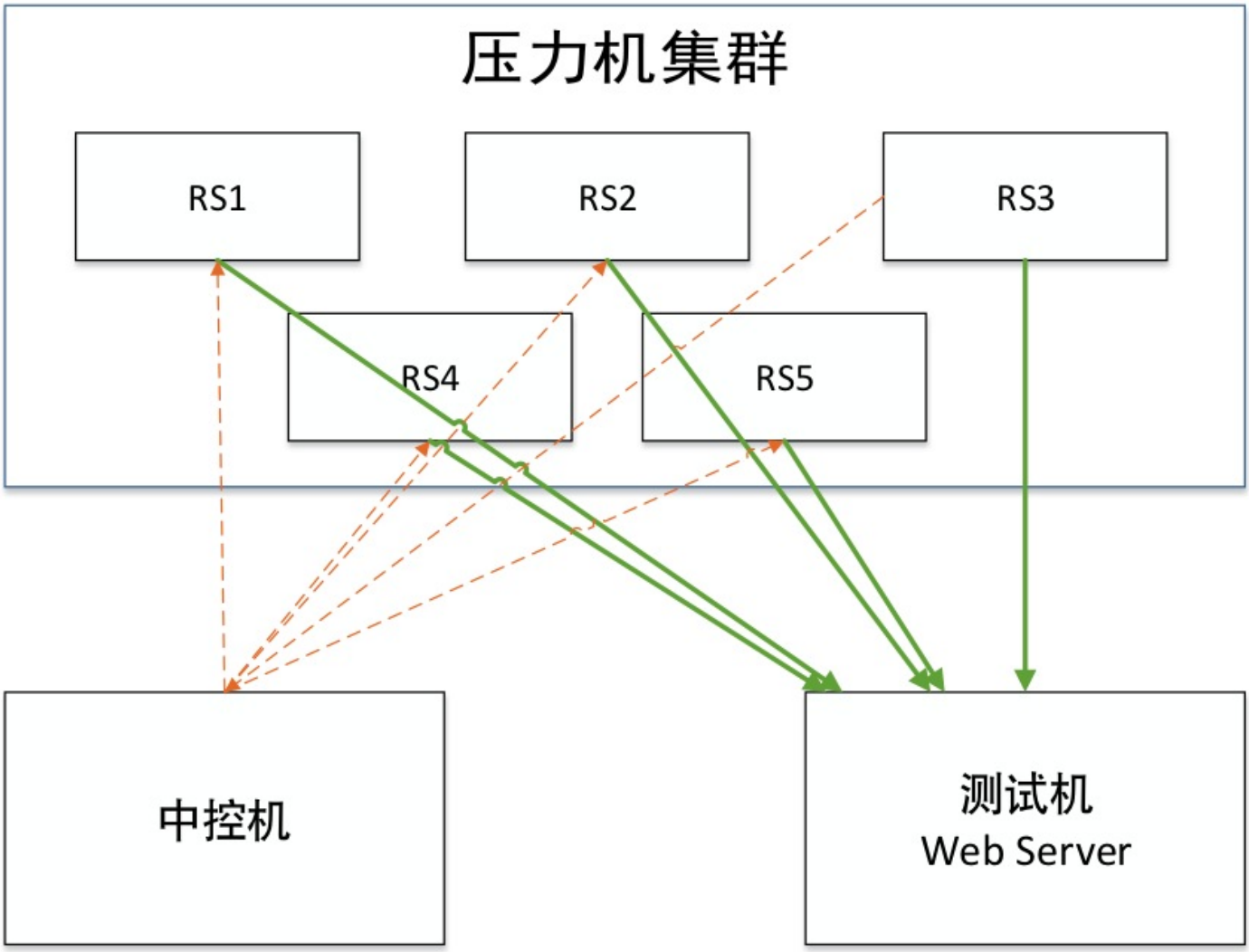
4. WebServer 集群测试方案

由于单台压力机无法发送足够大的压力测试腾讯云https服务的极限性能，需要采用多台压力机来发送压力，整个测试包含三部分：

1. 压力机集群。用来发送 http/https 压力，并输出单台机器的压力测试结果 。

2. 中控机，同步控制压力机集群的启动和结束，获取各台机器的压力数据并汇总输出。
3. 测试机，即承载腾讯云 HTTPS 服务的云机器，测试 WebServer 性能，直接返回页面，不需要连接 upstream。

连接关系如下表示：



5. HTTPS WebServer 测试性能数据

连接类型	Session cache	包体 (bytes)	加密套件	性能 (qps)
长	打开	230	ECDHE-RSA-AES128-GCM-SHA256	296241
短	关闭	230	ECDHE-RSA-AES128-GCM-SHA256	65630

6. CLB HTTPS 能力测试结论

由上表可知，腾讯云 HTTPS 支持 SSL 加解密，其后端拥有多个服务器集群，单集群的单台云服务器完全握手性能可达到65000 qps，长连接时性能可以达到约300000qps。

普通情况下，HTTPS 协议由于使用 SSL 协议，增加了至少一次完整握手的过程，因此增加延时为2 × RTT。此外，SSL 对称/非对称加密将消耗大量 CPU 资源，RSA 的解密能力是困扰 HTTPS 接入的主要难题。

使用腾讯云负载均衡的 HTTPS 服务，用户无需为 SSL 加解密单独部署服务，且腾讯云不收取任何额外费用，让用户轻松拥有极强的业务承载能力和防攻击能力。

压力测试常见问题

最近更新时间：2022-11-29 14:56:59

根据客户压测经验，本文总结常见的压测性能问题，为用户提供排查方案，并提供压测时的建议。

压力测试常见问题

后端主机未开启公网流量

购买云服务器时，如果不开启公网流量，则该主机挂载公网负载均衡时会导致转发不通的情况。

后端主机带宽设置不够

如果后端主机设置带宽过低，则带宽超过设定阈值后，后端服务器不会回包给 CLB，这样 CLB 处理时会返回504、502给客户端。

客户端端口不足

客户端个数过少，或客户端的端口范围设置过小时，客户端端口不足，会导致建立连接失败。此外，长连接建立时如果 keep_alive 字段大于0，此时连接会一直占用端口，导致客户端端口不足。

后端服务器依赖的应用成为性能瓶颈

请求经过负载均衡达到后端服务器后，后端服务器本身负载正常，但由于所有的后端服务器上的应用又依赖数据库等其他应用，此时如数据库出现性能瓶颈，也会影响压测性能。

后端服务器的健康状态异常

压测时容易忽略后端服务器的健康状态，如果有后端服务器健康检查失败或者健康检查状态反复（时好时坏，反复变化）时，也会导致压测性能低的现象。

负载均衡开启会话保持，后端主机流量分配不均

负载均衡开启会话保持后，容易造成请求落在固定的几台后端服务器上，导致流量分配不均衡，压测性能受到影响。建议压测时关闭会话保持。

压测建议

> **注意：**
> 以下设置仅用于压测负载均衡能力，并不表示用户生产环境也需要如此设置。

- 压测负载均衡转发能力时，建议使用短连接。
  一般除了验证会话保持等功能外，压测主要是希望验证负载均衡的转发能力，因此可以使用短连接来测试 CLB 和后端服务器的处理能力。
- 压测负载均衡吞吐量时建议使用长连接，用来测试带宽上限、或长连接业务等。
  此时建议将压测工具的超时时间调整为较小的阈值，超时时间过长时，会导致平均响应时间加长，从而不利于快速判断是否到达压测水位。
- 建议后端服务器提供一个静态网页用于压测，避免应用本身逻辑带来的损耗，如 I/O、DB 等。
- 监听不开启会话保持功能，否则压力会集中在个别的后端服务器，此外，压力性能不达标时，可以通过查看负载均衡下后端主机的监控数据判断是否流量分配均匀。
- 监听关闭健康检查功能，减少健康检查请求对后端服务器的访问请求。
- 使用多个 client（> 5）进行压测，源 IP 分散，能够更好的模拟线上实际情况。

CLB 证书操作权限问题

最近更新时间：2022-11-29 11:08:09

操作场景

由于2020年3月23日后，CLB 的证书相关操作均接入访问管理（CAM）鉴权。因此当使用子用户账号进行 CLB 的证书相关操作时，若提示“该操作需要授权，请联系您的开发商为您添加权限”，可根据如下操作，给子用户账号授予证书相关权限。

前提条件

登录的账号需为主账号，或有 CAM 相关权限的子用户账号：即已关联 QcloudCamFullAccess（用户与权限（CAM）全读写访问权限）策略的子用户账户。

> 说明：
>
> - 子用户账户 CAM 相关权限查看方式：您可在访问管理控制台的 用户列表 中，进入对应子用户的详情页，在“权限”中查看是否已关联 QcloudCamFullAccess 策略。
> - 若已关联 QcloudCamFullAccess 策略，但进行授予证书相关权限操作中仍出现“暂无API权限(message:GetReceiversOnAllType)，请联系开发商授权”的提示，请忽略并继续操作。

操作步骤

请选择如下任意一种方式，进行授予证书相关权限操作。

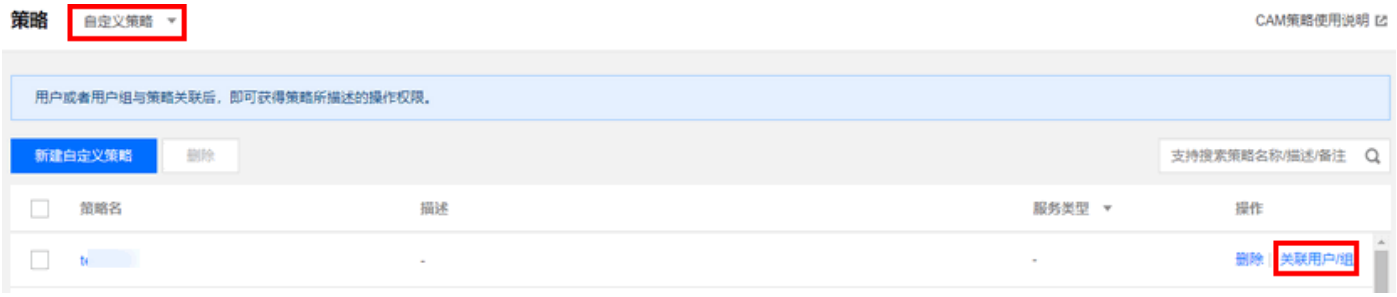
方式一：关联自定义策略

1. 登录 访问管理控制台。
2. 在左侧导航栏，单击**策略**，进行“策略”列表页面。
3. 单击**新增自定义策略**，在弹出框中，选择**按策略语法创建**。
4. 在“选择模板策略”页面中，选择**空白模板**，单击**下一步**。
5. 在“编辑策略”页面中，输入策略名称，在“编辑策略内容”的输入框中，输入如下策略内容。

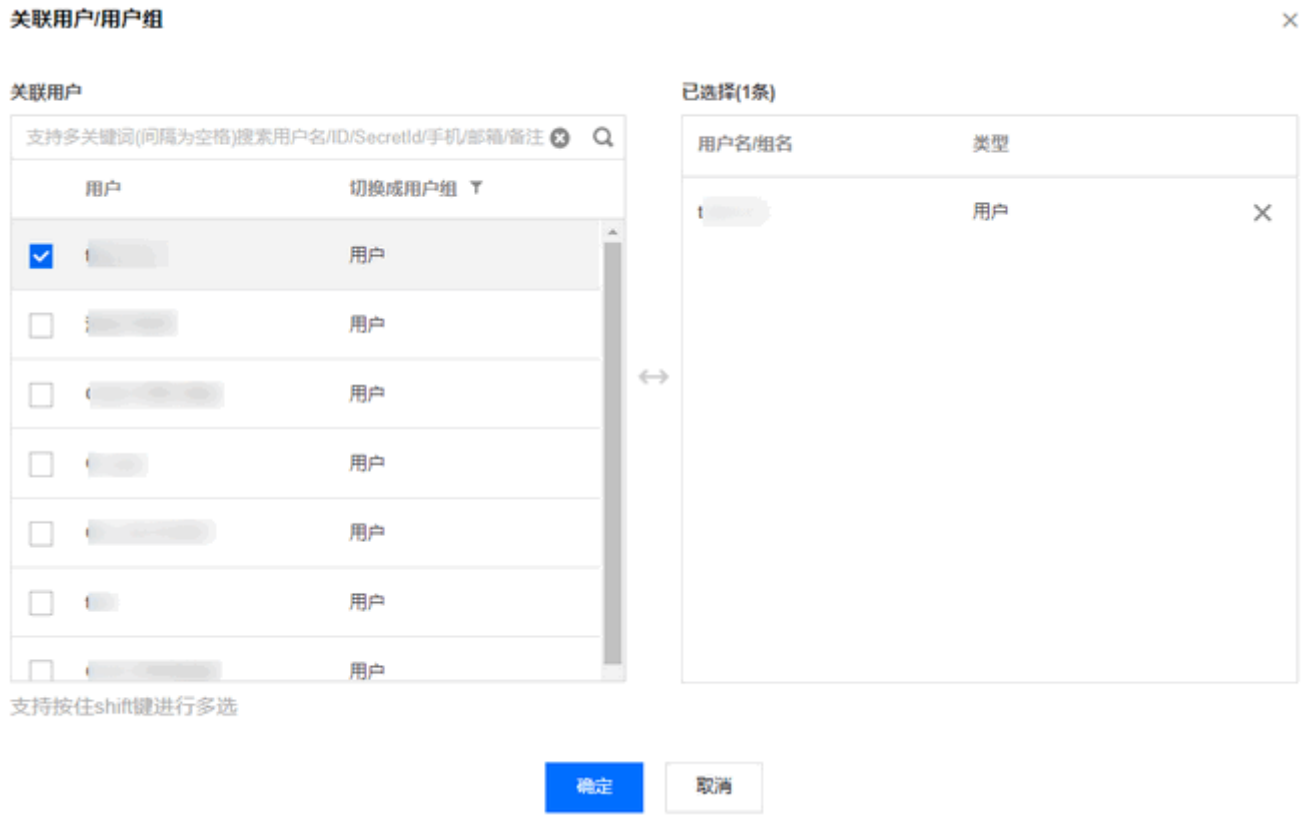
```
{
"version": "2.0",
"statement": [
{
"action": "name/ssl:*",
"resource": "qcs::ssl:::*",
```

```
"effect": "allow"
}
]
}
```

6. 完成后，单击**创建策略**，返回“策略”列表页面。
7. 在“策略”列表页面上方，选择**自定义策略**，在列表中找到刚创建的策略所在行，单击操作栏下的**关联用户/组**。



8. 在弹出框中，选择需授权的用户，单击**确定**即可。



方式二：关联预设策略

1. 登录 访问管理控制台。
2. 在左侧导航栏，选择**用户>用户列表**，进行“用户列表”页面。
3. 在需授权的子用户的所在行，单击操作栏下的**授权**。

4. 在弹出框中，选择 QcloudSSLFullAccess（SSL证书（SSL）全读写访问权限）或者 QcloudSSLReadOnlyAccess（SSL证书（SSL）只读访问权限），单击**确定**即可。