负载均衡

运维指南

产品文档

腾讯云

文档目录

运维指南
健康检查异常排查思路

最近更新时间：2017-12-15 16:24:30

1. 四层排查

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LB后端服务器端口的健康状态，若不健康，排查思路如下：

- 确定CLB后端服务器是否有配置有防火墙影响了服务，如果有请关闭
- 使用netstat命令，确定后端服务器的端口是否有进程在监听，若未启动，则重新启动服务

2. 七层排查

针对7层（HTTP协议）服务，当某一监听出现健康检查"异常"时，可以通过如下方面进行排查：

- 由于负载均衡的七层健康检查服务与后端CVM之间的通讯是走内网的，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之间的通讯正常。

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
```

如果能看到1.1.1.10:80的监听或0.0.0.0:80的监听则说明这部分正常。

- 请确保后端服务器开启了相应的端口，该端口必须与您在负载均衡监听配置中配置的后端端口保持一致。

如果是4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7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` 命令看看状态是否为HTTP/1.1 200 OK，本例是： `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讯。

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系统可以运行输入 `firewall.cpl` 操作关闭
Linux系统可以输入 `/etc/init.d/iptables stop` 关闭

- 检查负载均衡健康检查参数设置是否正确，建议参照这里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。

客户端timewait过多解决方案

最近更新时间：2018-06-01 17:32:27

本文背景：

客户压测LB时，常会遇到了一些客户端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_resuse ： 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_resuse 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_resuse 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LB不能开启 tcp_timestamps 的原因。

1.tcp_tw_recycle 和 tcp_tw_resuse只有在tcp_timestamps打开时才会生效

2.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;
        }
}
```

> 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.LB（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）
备注：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;
}
```

> 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：

1. HTTP使用短连接（Connection: close），这时由LB主动关闭连接，客户端不会产生timewait

2. 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：2017-12-04 11:40:50

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。我们和国际顶级的证书厂商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

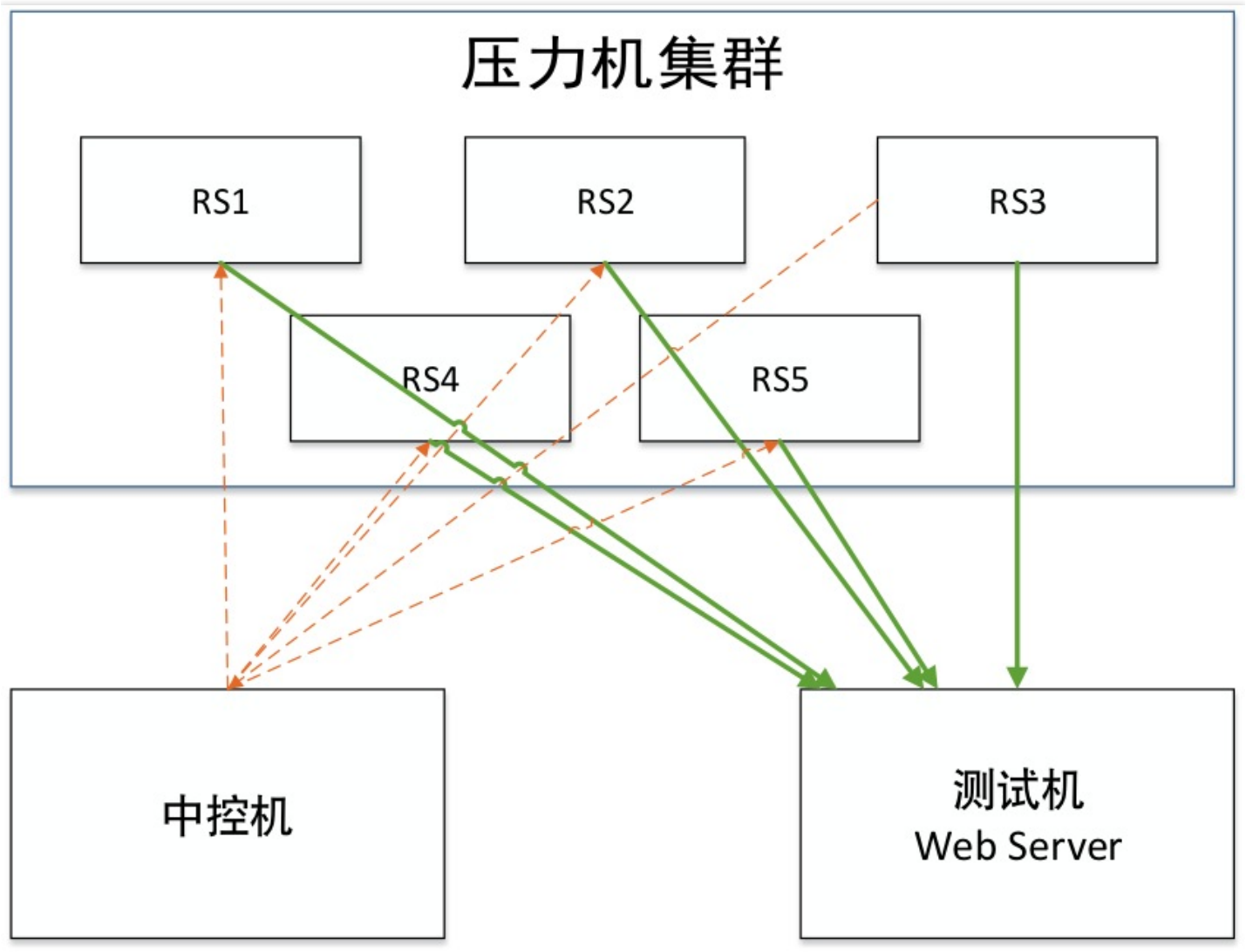
4. WebServer集群测试方案

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. CLBhttps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：2018-06-01 17:31:46

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

1. 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。

2. 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LB，这样LB处理时会返回504、502给客户端。

3. 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。

4. 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。

5. 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。

6. 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，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LB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