

负载均衡压测过程中的端口复用问题案例分享

四层服务器负载均衡

孔凡安

2023-08-30 发表

组网及说明

组网简化模型如下：

压测系统---LB---Nginx服务器

备注：LB设备（做了SNAT）基于四层负载，将前端的HTTP请求根据负载算法均匀分担到后端的多个Nginx服务器上。压测系统发起新建连接请求，收到HTTP响应即认为交易成功。

告警信息

不涉及

问题描述

压测系统新建连接较小（新建500）的情况下，交易成功率稳定在100%。但是新建连接速率提升以后，随着时间的推移交易成功率急速下降，最终成功率60%左右。

该场景为压测场景，模拟客户系统上线遭遇突发流量。因此问题亟需解决。

过程分析

首先还是分析压测过程中设备的性能参数（CPU、内存、接口利用率等），发现均没有超过设备性能。设备作用主要是四层负载，基于数据流（网络层和传输层信息）进行负载分担，性能压力不是很大。

那么这个时候能做的就是抓个包来看下，抓包位置位于LB设备（ip.addr==172.34.84.72）和后端Nginx服务器之间。通过抓包发现：

1. LB转发给Nginx的TCP请求，发起了SYN请求，Nginx也返回了ACK。
2. SYN和ACK的Seq无法匹配，认为连接失败。等待3秒后再次发起请求，再次收到错误回应。再等待2秒后发起RST，认为连接失败。

The screenshot shows a Wireshark capture of a failed TCP connection. The packet list shows a SYN request from the client (172.34.84.72) to the server (172.34.82.36) with Seq=2366768198. The server responds with an ACK (Seq=1710004877) but the Seq values do not match. The packet details show the SYN flag is set and the ACK flag is not. The packet bytes show the raw TCP and IP headers.

那么为什么会出现这种情况呢？根据报文推测后端Nginx服务器上还存在原先的连接，并没有老化。LB上发起的新的连接无法与之匹配，导致两端交互异常。如下报文可以证实这一点，通过五元组以及Seq以及Ack等参数可以进一步明确Nginx上还存在原来的连接。

The screenshot shows a Wireshark capture of a successful TCP connection. The packet list shows a SYN request from the client (172.34.84.72) to the server (172.34.82.36) with Seq=3224055689. The server responds with an ACK (Seq=1710004171) and the Seq values match. The packet details show the SYN flag is set and the ACK flag is not. The packet bytes show the raw TCP and IP headers.

下面进行TCP/IP协议关于四次挥手过程的科普环节，下图为TCP拆链过程中状态机的跳转过程，主动断开连接的一方接收到最后一个ACK确认报文后要等待2 MSL时间才能关闭连接。RFC 793 中有指出TCP 连接需要在 TIME_WAIT 中等待 2 倍的 MSL，RFC 793 文档将 MSL 的时间设置为 120 秒，即两分钟，然而这并不是一个经过严密推断的数值；实际上，Linux 开始就将 TIME_WAIT 的等待时间 TCP_TIMEWAIT_LEN 设置成 60 秒，以便更快地复用 TCP 连接资源。

简单进行一个算数题，在 Linux 上，客户端的可以使用端口号 32,768 ~ 61,000，总共 28,232 个端口号与远程服务器建立连接，应用程序可以在将近 3 万的端口号中任意选择一个，但是如果主机在过去一分钟时间内与目标主机的特定端口创建的 TCP 连接数超过 28,232，那么再创建新的 TCP 连接就会发生错误，也就是说如果我们不调整主机的配置，那么每秒能够建立的最大 TCP 连接数约为 470 (28232/60)。

那么基于以上分析，似乎优化Nginx服务器的2 MSL时间是一个最优解。有没有一种可能在B设备上做文章呢？

答案是可以，基于前面的分析我们可以得知，新建的连接对于LB设备来说是新连接，因为会话处于tcp-time-wait以及tcp-close状态时老化时间只有2S。那么网络设备去配合应用侧，在LB设备上把老化时间调大，Time-Wait时间比后端服务器更长，不会出现连接复用，则不会出现Seq出错，成功率得到维持

解决方法

LB设备调整老化时间长于后端Nginx服务器2 MSL时间+LB设备增加地址池地址组合拳。
以上的参数调整可以完美的解决连接复用的问题，然后事情都是一环扣一环环环相扣。老化时间调整后由此衍生出一个新的问题，那就是SNAT端口又不够用了。想象一下设备上有十几万会话维持在TCP-CLOSE或者TCP-TIME-WAIT状态，会导致LB设备端口不足，这会直接导致业务异常。
这个问题解决起来要简单多了，直接SNAT地址池加地址即可。

