

23 更快、更高、更强的 Nginx

更新时间：2020-02-28 10:36:28



“世界上最宽阔的是海洋，比海洋更宽阔的是天空，比天空更宽阔的是人的胸怀。”——雨果

更多资源请+q:511661754
+v: Andvaclu

前言

我们在上一篇文章中介绍了如何通过 `gzip` 命令来减小传输体积，减小数据传输时间，提升用户体验。

`Nginx` 还有很多其他的参数可以提高响应速度，改善用户体验，我们在本篇文章中简单的介绍几个最常用优化指令。

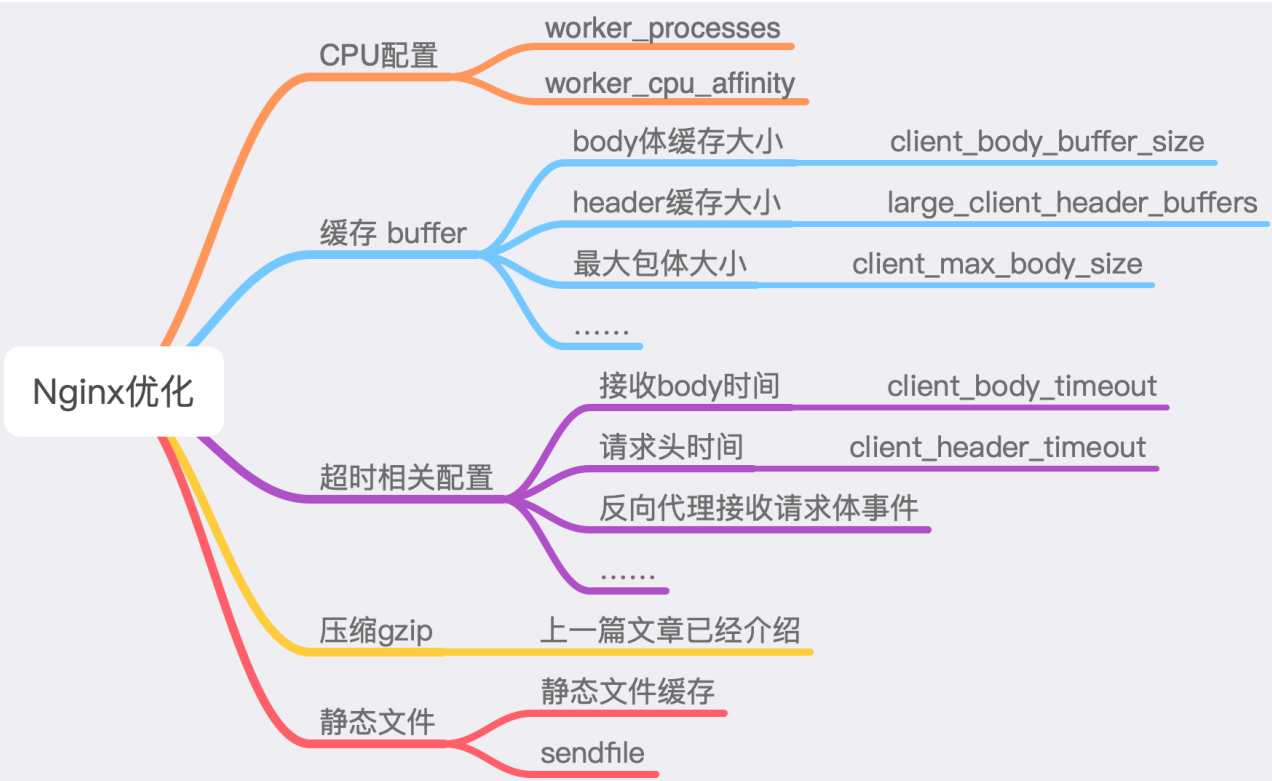
优化措施

一位 `伟人` 曾经说过：

没有任何性能问题是增加一台机器解决不了的，如果还没有结局，那就再增加一台。

但是增加一台机器会带来很大的财务压力以及运维压力，我们更希望从软件本身方便进行优化，最大力度的榨干服务器性能，降低硬件成本和运维成本。

我们从以下几个方面来介绍常用的优化措施：



CPU 优化

更多资源请+q:311861754

我们可以通过修改 Nginx 中 CPU 相关指令来提升服务器的性能。

这里主要涉及到两个指令，分别是 worker_processes 和 worker_cpu_affinity 指令。

worker_processes 指令

这个指令用来设置 Nginx 的 worker 进程数量。我们在本系列文章的开始时候就介绍过，Nginx 是一种 Master-Worker 机制，真正干活的是 worker 进程，所以适当的提高 worker 进程的数量，可以提高服务器处理请求的速度。但是 worker 进程的数量也不是越多越好，worker 进程数量太多的话会增加 CPU 切换进程的负担，一般建议设置为和当前服务器核心数量相同。

worker_cpu_affinity 指令

affinity

英 [ə'fɪnəti] 美 [ə'fɪnəti]

n. 密切关系；吸引力；姻亲关系；类同

[复数 affinities]

网络释义

英英释义

亲和性

我们先看一下 **affinity** 的意思是 **亲和性**。那么 **cpu_affinity** 就是 **cpu亲和性**。那么什么是 **cpu亲和性** 呢？

这个概念是对于进程来说的。我们知道，在现代的多核处理器上面，**OS** 在多次调度同一个进程的时候，可能将进程分配到不同的处理器上面进行处理，这样就可能导致进程的缓存不可用，会增加进程执行的时间。所以就产生了 **cpu亲和性** 这个东西，这个功能可以把一个进程和一个处理核绑定在一起，这样就可以充分利用 **CPU** 的缓存，提高缓存命中率，加快进程的执行速度。

它的值是一个二进制，对应物理核心数量，置 **1** 的那一位表示绑定到该物理核。

worker_cpu_affinity 00000001 00000010 00000100 00001000 00010000 00100000 01000000 10000000

缓存控制

对于任何一个优秀的软件来说，都离不开缓存 **buffer**。在 **Nginx** 中对 **buffer** 处理的方法都比较一致，一般情况下我们设置单个 **buffer** 的大小以及使用 **buffer** 的数量。如果单个 **buffer** 不够，那么继续分配 **buffer**，直到达到指定数量。如果还不够，还可能使用临时文件来保存。这里就要尽量地避免开辟临时文件，这样可以保证所有的内容都保存在内存中，可以提高响应速度。

我们要合理的评估相关内容的大小，分配适应的缓存。

和 **buffer** 相关的几个常用配置指令如下：

- **client_max_body_size** : 客户端请求服务器最大的 **body** 大小
- **client_body_buffer_size** : **Nginx** 分配给请求数据的 **Buffer** 大小，如果请求的数据小于 **client_body_buffer_size** 直接将数据先在内存中存储。如果请求的值大于 **client_body_buffer_size** 小于 **client_max_body_size**，就会将数据先存储到临时文件中
- **client_header_buffer_size** : 指定用于请求 **header** 的缓存区大小

此外还有很多类似的 **buffer** 大小，功能都是大同小异。

超时时间

这个时间用于控制一些时间参数，比如从客户端接收请求头的时间，接收请求体的时间，反向代理服务器接收请求内容的时间等等。通过对他们设置一些合理的值，可以让一些请求直接结束。

常用的超时时间有下面几个：

client_body_timeout : 客户端和服务端建立连接之后接收一个完整 **body** 体的时间。

client_header_timeout : 服务器接收一个完整的请求 **header** 的时间

总结

Nginx 还有很多优化的参数，但是总体上可以分为上面几类，大家在使用的过程中要多多总结，多多领会。

}

更多资源请+q:311861754
+v: Andvac1u