链滴

Nginx 是如何解决 epoll 惊群的

作者：zhaozhizheng
原文链接：https://ld246.com/article/1588731832846
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

•

accept_mutex 锁

如果开启了 accept_mutex锁，每个 worker 都会先去抢自旋锁，只有抢占成功了，才把 socket 加到 epoll 中，accept 请求，然后释放锁。accept_mutex锁也有负载均衡的作用。

accept_mutex效率低下，特别是在长连接的时候。因为长连接时，一个进程长时间占用 accept_mutex锁，使得其它进程得不到 accept 的机会。因此不建议使用，默认是关闭的。

EPOLLEXCLUSIVE 标识

EPOLLEXCLUSIVE是4.5+内核新添加的一个 epoll 的标识，Ngnix 在 1.11.3 之后添加了 NGX_EXCLSIVE_EVENT。

EPOLLEXCLUSIVE标识会保证一个事件发生时候只有一个线程会被唤醒，以避免多侦听下的“惊群”题。不过任一时候只能有一个工作线程调用 accept，限制了真正并行的吞吐量。

SO_REUSEPORT 选项

SO_REUSEPORT 是惊群最好的解决方法，Ngnix 在 1.9.1 中加入了这个选项，每个 worker 都有自的 socket，这些 socket 都 bind同一个端口。当新请求到来时，内核根据四元组信息进行负载均衡非常高效。

参考：

- http://xiaorui.cc/2015/12/02/使用socket-so_reuseport提高服务端性能/
- http://pureage.info/2015/12/22/thundering-herd.html
- http://www.cnblogs.com/sxhlinux/p/6254396.html
- http://m.blog.csdn.net/russell_tao/article/details/7204260
- https://www.zhihu.com/question/51618274?utm_medium=social&utm_source=wechat_session&from=groupmessage&isappinstalled=1
- http://www.sohu.com/a/148006569_470018