

# fastsocket

Monday, March 28, 2016 10:55 AM

随着网络的快速发展，单机的CPU核心数目越来越多。伴随着移动互联网的飞速发展，网络中心的应用中需要处理的小包和短连接也越来越多。因此服务器的多核拓展性变得越来越重要。在Linux Kernel中有以下几个方面特别需要注意，一个是TCB的管理，另一个是VFS。这两个层次的问题主要有两个，一个是TCB是全局的。另一个是VFS中对inode和dentry的操作也会导致同步操作。

另一方面要考虑的是生产环境的需求，TCP/IP协议栈要能够和以往的应用程序兼容。与此同时要能够保证安全性。以及资源的隔离。

对Linux Kernel中的TCP/IP协议栈做一个分析，发现主要有以下几个问题：

1. TCB 的管理，包括全局监听表和全局建立表
2. 缺乏链接的本地性
3. 其他的如VFS的损耗

fastsocket设计的架构

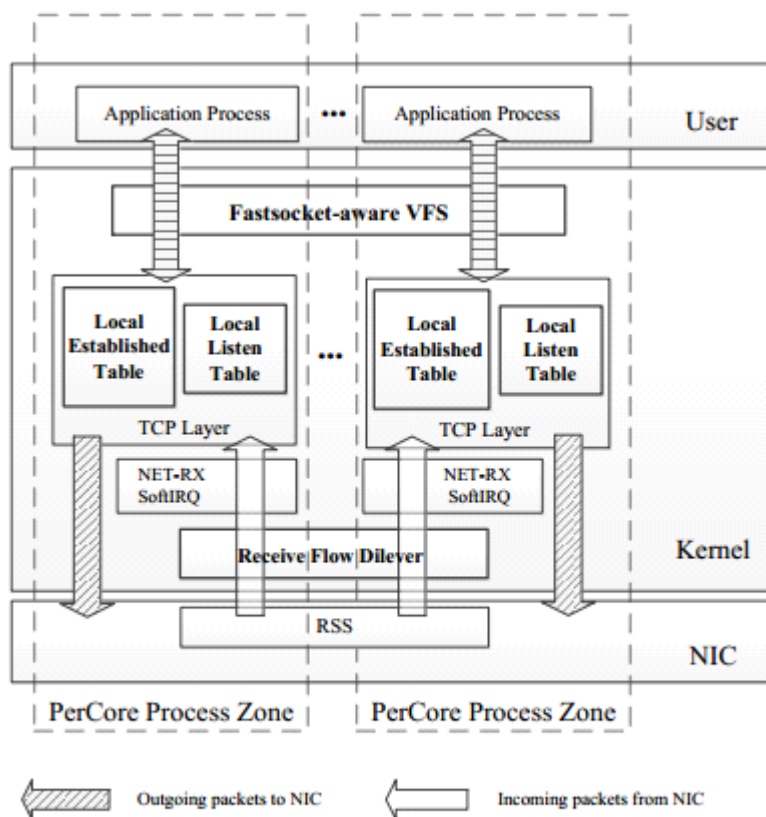


Figure 1: Fastsocket Architecture

fastsocket包括一个被分开的TCB数据结构（分成本地的表）RFD和一个经过修改的VFS层。

为了构建CPU的私有数据，修改共享数据结构。

1. Fastsocket完整的将数据结构分开。
2. 在数据包进入网络链路之前，Fastsocket使用RFD暗示包的合适的CPU。
3. VFS层是一个Socket API抽象的关键。修改后的VFS减少或跳过了一些不必要的锁操作。

除了这三个提高，值得关注的还有Fast Path和Slow Path。根据过来的包的类型和序号，可以判断它是否是快速处理的。