

轻量级 TCP/IP 协议栈

在开源系统中，可能一些[协议栈](#)的使用会稍许不同，那么下面我们就讲解了几个常用的开源协议栈，来为大家详细讲解和对比一下。具体内容，还请大家从下文来了解吧，首先让我们看看 BSD TCP/IP 协议栈。

1、BSD TCP/IP 协议栈

BSD 栈历史上是其他商业栈的起点，大多数专业 TCP/IP 栈（VxWorks 内嵌的 TCP/IP 栈）是 BSD 栈派生的。这是因为 BSD 栈在 BSD 许可协议下提供了这些专业栈的雏形，BSD 许用证允许 BSD 栈以修改或未修改的形式结合这些专业栈的代码而无须向创建者付版税。同时，BSD 也是许多 TCP/IP 协议中的创新（如广域网中拥塞控制和避免）的开始点。

2、uC/IP

uC/IP 是由 Guy Lancaster 编写的一套基于 uC/OS 且 TCP/IP 开源协议栈，亦可移植到其它操作系统，是一套完全免费的、可供研究的 TCP/IP 协议栈，uC/IP 大部分源码是从公开源码 BSD 发布站点和 KA9Q（一个基于 DOS 单任务环境运行的 TCP/IP 协议栈）移植过来。uC/IP 具有如下一些特点：带身份验证和报头压缩支持的 PPP 协议，优化的单一请求/回复交互过程，支持 IP/TCP/UDP 协议，可实现的网络功能较为强大，并可裁减。uC/IP 协议栈被设计为一个带最小化用户接口及可应用串行链路网络模块。根据采用 CPU、编译器和系统所需实现协议的多少，协议栈需要的代码容量空间在 30-60KB 之间。

3、LwIP

LwIP 是瑞士计算机科学院（Swedish Institute of Computer Science）的 Adam Dunkels 等开发的一套用于嵌入式系统的开放源代码 TCP/IP 协议栈。LwIP 的含义是 Light Weight（轻型）IP 协议，相对于 uIP。LwIP 可以移植到操作系统上，也可以在无操作系统的情况下独立运行。LwIP TCP/IP 实现的重点是在保持 TCP 协议主要功能的基础上减少对 RAM 的占用，一般它只需要几十 K 的 RAM 和 40K 左右的 ROM 就可以运行，这使 LwIP 协议栈适合在低端嵌入式系统上使用。LwIP 的特性如下：支持多网络接口下的 IP 转发，支持 ICMP 协议，包括实验性扩展的 UDP（用户数据报协议），包括阻塞控制，RTT 估算和快速恢复和快速转发的 TCP（传输控制协议），提供专门的内部回调接口（Raw API）用于提高应用程序性能，并提供了可选择的 Berkeley 接口 API。

4、uIP

uIP 是专门为 8 位和 16 位控制器设计的一个非常小的 TCP/IP 栈。完全用 C 编写，因此可移植到各种不同的结构和操作系统上，一个编译过的栈可以在几 KB ROM 或几百字节 RAM 中运行。uIP 中还包括一个 HTTP 服务器作为服务内容。许可：BSD 许用证

5、TinyTcp

TinyTcp 栈是 TCP/IP 的一个非常小和简单的实现, 它包括一个 FTP 客户. TinyTcp 是为了烧入 ROM 设计的并且现在开始对大端结构似乎是有用的(初始目标是 68000 芯片). TinyTcp 也包括一个简单的以太网驱动器用于 3COM 多总线卡

选择一个开源协议栈可以从四个方面来考虑:

一个是是否提供易用的底层硬件 API, 即与硬件平台的无关性;

一个是与操作系统的内核 API. 协议栈需要调用的系统函数接口是否容易构造,

另一个对于应用支持程度.

最关键的是占用的系统资源是否在可接受范围内, 有裁减优化的空间否?

其中, BSD 栈可完整实现 TCP/IP 协议, 但代码庞大, 70KB-150KB 之间, 裁减优化有难度, uIP 和 TinyTcp 代码容量小巧, 实现功能精简, 限制了在一些较高要求场合下的应用, 如可靠性与大容量数据传输.

LwIP 和 uC/IP 是同量级别的两个开源协议栈, 两者代码容量和实现功能相似, LwIP 没有操作系统针对性, 它将协议栈与平台相关的代码抽象出来, 用户如果要移植到自己的系统, 需要完成该部分代码的封装, 并为网络应用支持提供了 API 接口的可选性. uC/IP 协议最初是针对 uC/OS 设计, 为方便用户移植实现, 同样也抽象了协议栈与平台相关代码, 但是协议栈所需调用的系统函数大多参照 uC/OS 内核函数原型设计, 并提供了协议栈的测试函数, 方便用户参考, 其不足在于该协议栈对网络应用支持不足.

根据以上分析, 从应用和开发的角度看, 似乎 LWIP 更得到了网上很多朋友使用的青睐; uC/IP 在文档支持与软件升级管理上有很多不足, 但是它最初是针对 UC/OS 而设计, 如果选用 UC/OS 作为软件基础的话, 在系统函数构造方面有优势. 当然你选择其他操作系统的话, 可参照 OS_NULL 文件夹下的文件修改. 以上的这些开源协议栈也并非免费, 拿来就可以用, 据我所知, UC/OS 的母公司推出 UC/OS-TCP/IP 花了 6 人*2 年的工作量, 国内某公司使用 LWIP 作为移植的参照, 花了 4-5 人*2 年的工作量来测试与优化协议, 使用商用 TCP/IP 栈的高费用就不足为奇了. 作为广大的爱好者学习而言, 如果只是跑跑原型, 实验一下效果, 以上的几种开源协议栈都提供了测试的例子, 应该是不错的选择.

个人的看法: LWIP 可优先考虑, 至少网上有很多的人一块研究, 参考的资料较多; UC/IP 其次, 如果你想深入学习 TCP/IP 的话, 移植 UC/IP 是一种挑战性的工作, 它尚需完善.