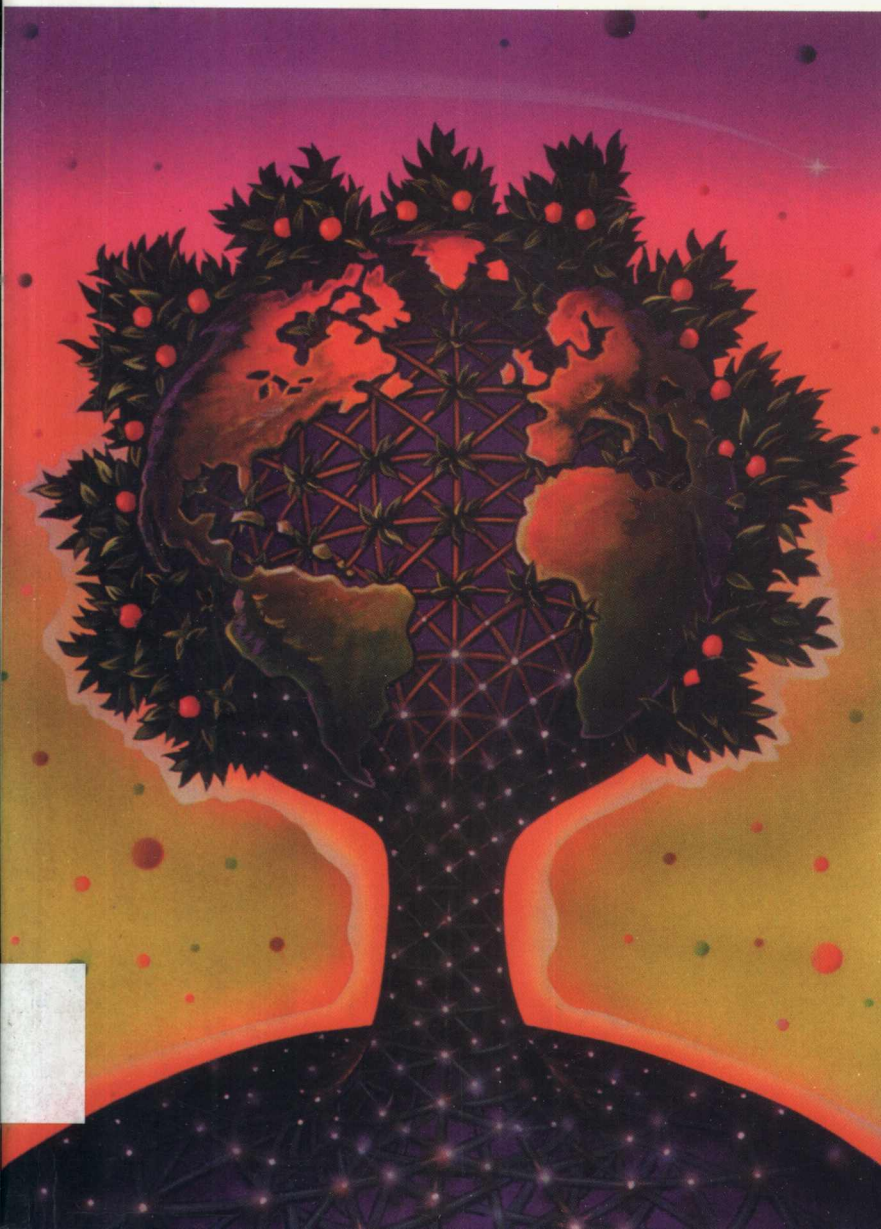


科海电脑技术丛书

Network
Programming with **Perl**

Perl 网络编程

[美] Lincoln D. Stein 著
王超 刘云 译



Pearson Education
培生教育出版集团
<http://www.pearsoned.com>



清华大学出版社

<http://www.tup.tsinghua.edu.cn>

Addison-Wesley

科海电脑技术丛书

Perl网络编程

Network programming with Perl

[美] Lincoln D. Stein 著

王超 刘云 译

清华大学出版社

(京)新登字158号

著作权合同登记号: 01-2001-5313

内 容 提 要

这是一本使用Perl编程语言创建基于网络的应用程序的指导书。

全书内容分为四个部分: 基础知识, 为公共服务开发客户, 开发TCP客户/服务器系统和高级主题。本书的目的是教会读者利用最新版Perl为TCP/IP提供的接口, 创建健壮、易维护且高效的客户/服务器应用程序, 核心内容是设计基于TCP的客户/服务器系统的方法和选择以及用于特定应用的高级技术。

全书层次分明, 内容由浅入深, 实例丰富, 并给出了具有实际指导意义的源代码。因此, 对于网络管理员和广大Perl初中级程序员来说, 本书是一种重要的资源。

Network programming with Perl

Copyright © 2001 by The Addison-Wesley.

All rights reserved. No part of the book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

本书中文简体字版由美国培生教育出版集团授权清华大学出版社和北京科海培训中心合作出版。未经出版者书面允许不得以任何方式复制或抄袭本书内容。

版权所有, 盗版必究。

本书封面贴有 Pearson Education 培生教育出版集团激光防伪标签, 无标签者不得销售。

书 名: Perl 网络编程

作 者: Lincoln D. Stein

译 者: 王超 刘云

出版者: 清华大学出版社 (北京清华大学校内, 邮编 100084)

印刷者: 北京市耀华印刷有限公司

发行者: 新华书店总店北京科技发行所

开 本: 787×1092 1/16 印张: 37.875 字数: 921 千字

版 次: 2002 年 3 月第 1 版 2002 年 3 月第 1 次印刷

印 数: 0001~5000

书 号: ISBN 7-302-05289-1/TP·3109

定 价: 62.00 元

前 言

网络无处不在。在办公室中，机器由网线相连形成局域网，局域网再通过Internet互连。在家庭中，个人计算机间歇地连入Internet，或者是，越来越多的个人计算机通过“一直存在的”电缆和DSL调制解调器连入Internet。譬如“蓝牙”（Bluetooth）这样的新的无线技术承诺将极大地扩展网络的领域，包括从蜂窝电话到厨房用具的每一件东西。

这样的环境为革新创造了巨大的机会。现在可以预测的所有应用类型都依赖于高带宽的“一直存在的”连接的性能。交互式游戏允许来自全球的游戏者在虚拟的游戏场上比赛，并且，即时信息协议让他们能够将获胜的消息广播给他们的朋友。新的对等系统，例如Napster和Gnutella，允许人们直接交换MP3音频文件和其他类型的数字内容。SETI@Home工程利用分布在世界各地的成千上万台个人计算机的空闲时间，在巨大的宇宙噪声集合中寻找宇宙生命的信号。

利用无处不在的网络也可以开发一些比较实际的应用。使用正确的知识，你可以编写机器人(程序)，让它从竞争者的Web站点上搜集产品的价格并概述出来；你可以编写脚本，让它在特定的股票价格低于指定的水准时通知你；你可以编写程序，让它生成每日的管理报告并通过电子邮件发送出去；你还可以配置服务器，在单一的高性能机器上集中处理一些数字任务，或者将任务分配给一个计算机群集中的多个结点处理。

不论你是搜索蒲团（铺于地板的垫子）的最佳价格还是在遥远的星系中寻找生命，你都需要理解网络应用程序是如何工作的，只有这样才能充分利用网络环境所创造的机会。你需要理解TCP/IP协议，TCP/IP协议是所有基于Internet的通信的通用标准，也是用于局域网的最通用的协议。你还必须知道如何连接远程程序，如何和该程序交换数据，当发生错误时要做什么。为了和已存在的应用程序(例如Web服务器)共同工作，你还需理解应用层协议是如何建立在TCP/IP之上的，以及如何处理公用的数据交换格式(例如XML和MIME)。

本书使用Perl编程语言来阐明如何设计和实现实用的网络应用程序。Perl是一种理想的网络编程语言。首先，和该语言的其他部分一样，Perl的网络功能也被设计得使容易的事情更容易。对Internet上某个地方的服务器打开一个网络连接并向其发送一条消息只需两行代码。一个功能完全的Web服务器仅需写几十行代码。

其次，Perl的开放式体系结构鼓励了许多有才能的程序员，让他们为第三方模块的不断扩展的库做贡献。许多这样的模块都为通用的网络应用提供了功能强大的接口。例如，装载LWP::Simple模块之后，进行一次函数调用就可以让你取回远程Web页的内容并将其存储到一个变量中。其他的第三方模块提供了电子邮件、FTP、网络新闻和各种网络数据库的直观接口。

Perl也提供了强大的可移植性。本书中开发的大多数应用程序无需修改就可以运行在UNIX机器、Windows、Macintosh、VMS系统和OS/2上。

然而，选择Perl用于网络应用程序开发的最重要原因是，它允许你完全利用TCP/IP的能

力。Perl提供了对低层网络调用的完全访问，就如同C程序和其他编译语言所能利用的低层网络调用一样。你能够创建组播应用程序、实现多路复用服务器、设计对等系统。使用Perl，你能够快速创建新的网络应用程序的原型并对已存在的应用程序开发接口。如果你曾经使用C或Java来编写网络应用程序，你将惊喜地发现Perl API完成了许多这些语言中要做的事。

本书面向的读者

本书面向初学者和中级Perl程序员。我假定你知道Perl编程的基本知识，包括如何写循环语句、如何构造if-else语句、如何写规则表达式模式匹配、\$_自动变量的概念，以及数组和数组的基本知识。

你应当能够访问一个Perl解释器，并具有编写、运行和调试脚本的经历。最重要的，你应当能够访问既连接局域网又连接Internet的计算机。在第10章中设置基于Perl的网络服务器在机器启动时自动启动，尽管这种设置方法需要超级用户(管理员)访问，但是其他的例子都不需要对机器的特权访问。

本书用到了Perl版本5和更高版本的面向对象的功能，但是大多数章节没有假定你需具有面向对象系统的深入知识。第1章说明了作为Perl对象的临时用户所需的全部细节。

本书没有全面复习最低层TCP/IP协议，也不是安装和配置网络集线器、路由器和名字服务器的指导。许多TCP/IP协议机制和网络管理方面具有代表性的图书都列在本书的附录C中。

本书的内容

本书分为四个主要的部分：基础知识、为公共服务开发客户、开发TCP客户/服务器系统和高级主题。

第1部分：“基础知识”介绍TCP/IP网络通信的基本原理。

- 第1、2章，“输入/输出基础”和“进程、管道和信号”介绍了Perl的用于输入和输出的函数和变量、讨论了I/O操作中会发生的异常、并使用管道化文件句柄作为介绍套接字的基础。这两章也回顾了Perl的进程模型，包括信号和创建子进程，也介绍了Perl的面向对象的扩展。
- 第3章，“Berkeley套接字简介”讨论了Internet网络的基本知识，并且讨论了IP地址、网络端口，以及客户/服务器应用程序的原理。然后转向介绍Berkeley套接字API，Berkeley套接字API为程序员提供对TCP/IP的接口。
- 第4、5章，“TCP协议”和“IO::Socket API”介绍了TCP的基本知识，TCP是提供可靠的面向流的通信的网络协议。这两章阐述了如何创建客户和服务应用程序，然后介绍了几个例子，展示了技巧的力量和一些常见的问题。

第2部分：“为公共服务开发客户”查看最佳的第三方模块的一个集合，这些第三方模块是开发者贡献给CPAN (Comprehensive Perl Archive Network)的。

- 第6章，“FTP和Telnet”介绍了两个模块。一个模块用于访问FTP文件共享服务，另一个为灵活的Net::Telnet模块，该模块允许你创建访问所有类型网络服务的客户

程序。

- 电子邮件仍然是Internet上的重要应用程序。第7章“SMTP：发送邮件”介绍了邮件发送方和接收方中的发送方。本章说明了如何迅速地创建电子邮件消息(包括二进制附件)并发往目的地。
- 第8章，“POP，IMAP和NNTP：处理邮件和网络新闻”介绍了电子邮件发送方和接收方中的接收方，解释了使从邮件发送系统接收邮件成为可能的模块，这些模块还处理邮件的内容，包括二进制附件。
- 第9章，“Web客户”讨论了LWP模块，该模块提供了和Web服务器对话、下载并处理HTML文档以及解析XML所需的所有东西。

第3部分：“开发TCP客户/服务器系统”——本书最长的部分——讨论了设计基于TCP的客户/服务器系统的众多选择。这些章节中使用的主要例子是交互式心理疗法学者服务器，该例子基于Joseph Weizenbaum的经典的Eliza程序。

- 第10章，“创建子进程的服务器和inetd守护进程”介绍了通用的TCP服务器类型，该服务器创建新进程，处理每个进入的连接。本章也介绍UNIX和Windows的inetd守护进程，inetd守护进程允许不是专为网络而设计的程序用作服务器。
- 第11章，“多线程应用程序”解释了Perl的试验性的多线程API，并展示了它是如何大大简化TCP客户和服务器的设计。
- 第12、13章，“多路复用应用程序”和“无阻塞I/O”讨论了select()调用，该调用使得应用程序能够在不使用多进程或多线程的情况下同时处理多个I/O流。
- 第14章，“安全的服务器”讨论了用于加强网络服务器的可靠性和可维护性的技巧。讨论的主题有日志、信号处理、异常、以及重要的网络安全。
- 第15章，“预创建子进程和预创建线程”介绍了前面各章中讨论的创建子进程和创建线程的模式。这些增强的功能提高了服务器在重负荷下更好执行的能力。
- 第16章，“IO::Poll”讨论了UNIX平台上可用来替代select()的一个选择。该模块允许应用程序使用一个API来多路复用多个I/O流，一些人发现这种方法比select()更自然好用。

第4部分：“高级主题”用于说明一些对特定应用程序有用的技术。

- 第17章，“TCP紧急数据”介绍TCP紧急数据（或称为TCP“带外”数据）。这项技术经常用于高度交互式应用程序，在这样的应用程序中用户需要紧急地向远程服务器传送信号。
- 第18、19章，“UDP协议”和“UDP服务器”介绍用户数据报协议(UDP, User Datagram Protocol)，UDP提供轻便的面向消息的通信服务。第18章介绍UDP协议，第19章展示如何设计UDP服务器。这两章和下面两章的主要例子包含一个完全用Perl编写的实时在线聊天和消息传送系统。
- 第20、21章，“广播”和“组播”展示了如何构建一对全和一对多消息广播系统，由此扩展了上面讨论的UDP。在这两章中，我们利用自动的服务器发现和组播功能扩展了聊天系统。

- 第22章, “UNIX域套接字”说明如何在同一台机器的进程之间创建轻便的通信通道。UNIX域套接字对于例如记录器这样的特定应用很有用。

Perl的众多版本

所有好的事物都在不断演进以适应不断改变的条件, Perl在其短暂的发展历程中已经经历了几次主要的改变。本书针对Perl的5.X系列版本而编写(5.003版本, 建议使用更高的版本)。在我写作前言的时候(2000年8月), Perl最新的5.6、5.7版本也已出版。我料想Perl的5.8版本和5.9版本(假定出现这样的版本)将兼容本书所给出的代码例子。

然而, Perl的版本6也已经冒出了水平线。预计在2001年夏季出现其早期的 α 版本, 版本6将修正许多Perl早期版本的特性和不当的功能。然而, 这样做, 势必会影响多数已存在的脚本。幸运的是, Perl语言开发者负责开发工具用于将已存在的脚本自动移植到版本6。考虑到这一点, 我尽力使本书的例子通用, 避免使用过于晦涩难懂的Perl结构。

跨平台兼容性

更严重的问题是Perl在各种操作系统上实现之间的差别。Perl开始于UNIX(和Linux)系统, 但已经被移植到许多不同的操作系统上, 包括Microsoft Windows, Macintosh, VMS, OS/2, Plan9和其他的操作系统。为Windows平台所写的脚本无需修改就能在UNIX或Macintosh上运行。

问题是, 对不同的操作系统来说, I/O子系统(管理输入和输出操作的系统部分)是有最大区别的部分。这个原因就限制了Perl的I/O系统完全可移植的能力。虽然Perl的基本I/O功能对不同的平台是相同的, 但是在非UNIX平台上, 一些更复杂的操作要么没有要么就是对不同的平台很不同。当然, 这就影响了网络编程, 因为网络基本上是关于输入和输出的。

本书中, 第1章到第9章使用一般的网络调用, 将运行在所有的平台上, 一个例外就是第5章的最后一个例子, 该例子调用了fork()函数, fork()函数没有在Macintosh上实现。还有第2章的进程管理的介绍性讨论是针对UNIX系统的。这些章节所讨论的技术都是绝大多数客户程序所需的, 并且足够用于简单服务器的启动和运行。第10章到第22章涉及服务器设计的更高级的主题。下表显示了各章中所描述的Perl功能是否被UNIX, Windows和Macintosh所支持。

章	主题	UNIX/Linux	Windows	Macintosh
1~9	基本网络编程	+	+	+
10	创建子进程的服务器	+	P	—
11	多线程服务器	+	+	—
12	多路复用	+	+	+
13	无阻塞I/O	+	+	+
14	安全的服务器	+	P	—
15	预创建子进程和预创建线程	+	—	—
16	IO::Poll	+	—	—
17	TCP紧急数据	+	—	—

续表

章	主题	UNIX/Linux	Windows	Macintosh
18	UDP	+	+	+
19	UDP服务器	+	+	+
20	广播	+	+	+
21	组播	+	—	—
22	UNIX域套接字	+	—	—

关键字：+支持的功能；—不支持的功能；P部分支持的功能

庆幸的是，Perl的非UNIX版本正在快速地改进，很可能在你读本书时Perl又有了新的功能。

取得示例的代码

本书讨论的所有例子脚本和模块都在Web上以ZIP和TAR/GZIP格式提供。下载这些资源的URL是<http://www.modperl.com/perl-networking>。该页也包括解包并安装源代码的指令。

安装模块

Perl的许多网络模块都预安装在标准的发布版本中。其他的就是第三方模块，必须从Web上下载并安装。多数第三方模块用纯Perl编写，但是有一些(包括本书提到的几个)模块的一部分是用C编写，所以在使用之前必须编译它们。

CPAN是一个巨大的基于Web的由他人所贡献的Perl模块的集合。你可以通过Web或FTP浏览器访问CPAN，或者使用Perl本身内置的命令行应用程序来访问它。

从 Web 安装

将你的Web浏览器指向<http://www.cpan.org>，找出一个离你最近的CPAN站点。该站点将显示一个页面，你可以在此查找特定的模块，或者浏览以多种方式排序的别人所贡献的模块的完全列表。当你发现想要的模块时，下载到磁盘上来。

Perl模块以压缩过的tar文档形式发布，你可以按如下方式将其解包：

```
% gunzip -c Digest-MD5-2.00.tar.gz | tar xvf -
Digest-MD5-2.00 /
Digest-MD5-2.00 / typemap
Digest-MD5-2.00 / MD2 /
Digest-MD5-2.00/MD2 / MD2.pm
...
```

一旦文档解包后，你就进入新创建的目录并执行perl Makefile.PL，make，make test和make install命令。这些命令将构建、测试并安装下载模块。

```
% cd Digest-MD5-2.00
% perl Makefile.PL
```



```

Testing alignment requirements for U32...
Checking if your kit is complete...
Looks good
Writing Makefile for Digest::MD2
Writing Makefile for Digest::MD5
% make
mkdir ./blib
mkdir ./blib/lib
mkdir ./blib/lib/Digest
...
% make test
make [1]: Entering directory '/home/lstein/Digest-MD5-2.00/MD2'
make [1]: Leaving directory '/home/lstein/Digest-MD5-2.00/MD2'
PERL_DL_NONLAZY=1 /usr/local/bin/perl -I./blib/arch -I.
/blib/lib...
t/digest..... ok
t/files..... ok
t/md5-aaa..... ok
t/md5..... ok
t/rfc2202..... ok
t/sha1..... skipping test on this platform
All tests successful.
Files=6, Tests=291, 1 secs ( 1.37 cusr 0.08 csys = 1.45 cpu)
% make install
make [1] Entering directory '/home/lstein/Digest-MD5-2.00/MD2'
make [1] Leaving directory '/home/lstein/Digest-MD5-2.00/MD2'
Installing /usr/local/lib/perl5/site_perl/i586-
linux/./auto/Digest/MD5/MD5.so
Installing /usr/local/lib/perl5/site_perl/i586-
linux/./auto/Digest/MD5/MD5.bs

```

在UNIX系统上，可能需要超级用户特权来执行最后一步。如果你没有这样的特权，你可以在你自己的主目录安装这些模块。在perl Makefile.PL步骤中，以你自己的主目录的路径提供PREFIX=参数。例如，假如你的主目录可以在/home/jdoe下找到，你将键入：

```
%perl Makefile.PL PREFIX=/home/jdoe
```

其他的安装过程和上述的过程相同。

如果你使用一个定制的安装目录，你必须告诉Perl到该目录下去找被安装的模块。一个方法就是将该目录的名字加入环境变量PERL5LIB中。例如：

```
setenv PERL5LIB /home/jdoe          #C shell
PERL5LIB=/home/jdoe; export PERL5LIB #bourne shell
```

另一个方法就是将下面一行置于使用该安装模块的每个脚本的顶部。

```
use lib'/home /jdoe';
```

从命令行安装

一种更简单的方法就是用Andreas Koenig的奇妙的CPAN shell。使用它，你能够通过一个简单的命令shell查找、下载、构建并安装Perl模块。install命令做所有这些事：

```
% perl -MCPAN -e shell
cpan shell -- CPAN exploration and modules installation (v1.40)
ReadLine support enabled
cpan> install MD5
Running make for GAAS/Digest-MD5-2.00.tar.gz
Fetching with LWP:
  ftp://ftp.cis.ufl.edu/pub/perl/CPAN/authors/id/GAAS/Digest-
  MD5-2.00.tar.gz
CPAN: MD5 loaded ok
Fetching with LWP:
  ftp://ftp.cis.ufl.edu/pub/perl/CPAN/authors/id/GAAS/CHECKSUMS
...
Checksum for /home/lstein/.cpan/sources/authors/id/GAAS/Digest-MD5-
2.00.tar.gz ok
Digest-MD5-2.00 /
Digest-MD5-2.00 / typemap
...
Installing /usr/local/lib/perl5/site_perl/i586-
linux/./auto/Digest/MD5/MD5.so
Installing /usr/local/lib/perl5/site_perl/i586-
linux/./auto/Digest/MD5/MD5.bs
Installing /usr/local/lib/perl5/site_perl/i586-linux/./auto/MD5 /
MD5.so
...
Writing /usr/local/lib/perl5/site_perl/i586-linux/./auto/MD5 /
.packlist
Appending installation info to /usr/local/lib/perl5/i586-linux /
5.00101/perllocal.pod
cpan> exit
```

使用 PPM 安装模块

这些例子都假定你拥有gzip, tar和make命令的UNIX兼容版本。纯粹的Windows系统没有这些实用程序。Cygwin软件包(可从<http://www.cygwin.com/cygwin/>上下载)提供了这些实用程序，作为UNIX兼容工具的完全集合的一部分。

然而，使用ActiveState Perl Package Manager (PPM) 会更容易。默认地，Perl脚本安装在Perl的ActiveState发布版本中，可通过<http://www.activestate.com>下载取得。其接口和前

面一段中所述的CPAN命令行接口相似，除此之外，它不仅可以安装纯Perl脚本也可以安装预编译的二进制文件。例如：

```
C:\WINDOWS>ppm
PPM interactive shell (1.1.3) -type 'help' for available commands .
PPM> install MD5
Install package 'MD5?' (Y/N) : Y
Retrieving package 'MD5' ...
Installing C:\Perl\site\lib\auto\MD5\MD5.bs
Installing C:\Perl\site\lib\auto\MD5\MD5.dll
Installing C:\Perl\site\lib\auto\MD5\MD5.exp
Installing C:\Perl\site\lib\auto\MD5\MD5.lib
Installing C:\Perl\site\lib\MD5.pm
Installing C:\Perl\site\lib\auto\MD5\autosplit.ix
Writing C:\Perl\site\lib\auto\MD5\packlist
PPM> exit
Quit!
C:\WINDOWS>
```

从 MacPerl 安装模块

MMP(MacPerl Module Porter)站点，<http://pudge.net/cgi-bin/mmp.plx>，包含一系列模块，这些模块已经移植可以用于MacPerl。在Macintosh上开发了各种有帮助的程序让模块安装更容易。这些包在<http://pudge.net/macperl/macperlmodinstall.html>上有介绍，该站点也给出了下载和安装这些包的指令。

联机文档

除了图书和Web站点，本书还参考了两种主要的联机信息资源，分别为Internet RFC和Perl POD文档。

Internet RFC

Internet的所有基本协议的规格说明都在提交给IETF(Internet Engineering Task Force, 因特网工程任务组)的一系列RFC(Requests for Comment, 请求评论)中描述。这些文档是顺序编号的。例如，RFC 1927——“Suggested Additional MIME Types for Associating Documents”——是提交的第1927个RFC。这些RFC中的一些最终将成为Internet标准，这样它们就被给定一个顺序编号的STD名字。然而，大多数RFC仍然是RFC。虽然RFC是非正式的，但是人们仍然以它作为参考，用于学习网络协议的细节并验证他们的特定实现是否正确。

RFC文档在Internet的许多地方都有镜像，并由几个组织以可搜索的形式维护。最佳的文档之一在<http://www.faqs.org/rfcs/>。为了从该站点检索一个RFC，进入指定的页面并在标为“Display the document by number”的文本字段中键入所需RFC的编号。该文档将以最小化的HTML格式交付。该页面也允许你搜索标准文档，并可以通过关键字和短语搜索文档。如果你更喜欢纯文本格式，www.faqs.org站点包含一个到它们的FTP站点的链接，在FTP站

点上可以查找并下载原始形式的RFC。

POD 文档

许多Perl的内部文档以POD(Plain Old Documentation)格式出现。这些文档多数是纯文本，也插入了一些标记元素以指示标题、子标题和详细列举的列表。

当你安装Perl时，也安装了POD文档。POD文件位于Perl库目录的pod子目录。你可以直接阅读POD文档，或者使用perldoc脚本来格式化POD文档，并在例如more这样的文本页面调度器中显示。

为了使用perldoc，键入命令和要查看的POD文件的名字。开始的最佳位置是Perl的内容表(Perltoc)：

```
% perldoc perltoc
```

这条命令将给出能够显示的其他POD页面的列表。

为了取得一个特定Perl函数的快速概述，perldoc接受-f标记。例如，为了看socket()函数的概述，键入：

```
%perldoc -f socket
```

对Macintosh用户，MacPerl发行版本有一个称为shuck的“helper”应用程序。shuck将POD查看功能加入到MacPerl Help菜单中。

目 录

第1部分 基础知识

第1章 输入/输出基础	1
1.1 Perl和网络	1
1.1.1 为进程间通信建立的语言	1
1.1.2 为文本处理建立的语言	2
1.1.3 开放源代码项目	2
1.1.4 面向对象网络扩展	2
1.1.5 安全	2
1.1.6 性能	2
1.2 轻松上网	3
1.3 文件句柄	5
1.3.1 标准文件句柄	5
1.3.2 输入和输出操作	6
1.3.3 检测文件尾	8
1.3.4 行尾之处的混乱	9
1.3.5 打开和关闭文件	11
1.3.6 缓冲和阻塞	15
1.3.7 标准I/O缓冲	17
1.3.8 传递和存储文件句柄	18
1.3.9 错误检测	19
1.4 使用IO::Handle模块和IO::File模块的面向对象语法	20
1.4.1 对象和引用	21
1.4.2 IO::Handle模块和IO::File模块	22
1.5 小结	26
第2章 进程、管道和信号	28
2.1 进程	28
2.1.1 fork()函数	28
2.1.2 system()函数和exec()函数	30
2.2 管道	31
2.2.1 打开管道	31
2.2.2 使用管道	32
2.2.3 易用的管道：反引号运算符	34

2.2.4	强大的管道: pipe()函数.....	34
2.2.5	双向管道	37
2.2.6	从一般的文件句柄中区分管道	38
2.2.7	可怕的PIPE错误	38
2.3	信号.....	40
2.3.1	通用信号	40
2.3.2	截取信号	42
2.3.3	处理PIPE异常	43
2.3.4	发送信号	45
2.3.5	对信号处理程序的忠告	46
2.3.6	让慢的系统调用超时	46
2.4	小结.....	48
第3章	Berkeley套接字简介	49
3.1	客户、服务器和协议.....	49
3.1.1	协议	49
3.1.2	面向二进制和面向文本的协议	50
3.2	Berkeley套接字	51
3.2.1	套接字分析	52
3.2.2	数据报套接字	54
3.2.3	流套接字	55
3.2.4	数据报套接字与流套接字	55
3.3	套接字寻址.....	56
3.3.1	IP地址.....	56
3.3.2	保留IP地址、子网和网络掩码.....	57
3.3.3	IPv6.....	59
3.3.4	网络端口	59
3.3.5	sockaddr_in结构.....	60
3.4	一个简单的网络客户	60
3.5	网络名和服务.....	62
3.5.1	将主机名转换为IP地址.....	62
3.5.2	主机名转换示例	63
3.5.3	获取协议和服务的相关信息	64
3.5.4	再访daytime客户	65
3.5.5	网络信息的其他资源	66
3.6	网络分析工具.....	67
3.6.1	ping.....	67
3.6.2	nslookup	67
3.6.3	traceroute	68

3.6.4	netstat	68
3.6.5	tcpdump	69
3.6.6	MacTCP Watcher	69
3.6.7	scanner.exe	69
3.6.8	net_toolbox.exe.....	69
3.7	小结.....	70
第4章	TCP协议	71
4.1	一个TCP echo客户	71
4.2	外出连接相关的套接字函数	73
4.3	一个TCP echo 服务器.....	74
4.3.1	进入连接相关的套接字函数	78
4.3.2	tcp_echo_serv1.pl的局限性	79
4.4	调整套接字选项.....	80
4.4.1	通用的套接字选项	80
4.4.2	SO_REUSEADDR套接字选项	82
4.4.3	fcntl()函数和ioctl()函数	82
4.5	其他套接字相关的函数.....	82
4.6	TCP通信中的异常情况	84
4.6.1	调用connect()过程中的异常	84
4.6.2	读操作和写操作中的异常	85
4.7	小结.....	85
第5章	IO::Socket API	87
5.1	使用IO::Socket	87
5.1.1	一个daytime客户	87
5.1.2	TCP echo客户	88
5.2	IO::Socket方法	89
5.2.1	IO::Handle类层次	89
5.2.2	创建IO::Socket::INET对象.....	90
5.2.3	IO::Socket对象方法	93
5.3	更实用的例子.....	95
5.3.1	再访逆转的echo服务器.....	95
5.3.2	一个Web客户	97
5.4	性能和风格.....	100
5.5	并发客户.....	101
5.5.1	一个唠叨的客户, 第一次尝试	101
5.5.2	一个唠叨的客户, 第二次尝试	103
5.6	小结.....	107

第2部分 为公共服务开发客户

第6章	FTP和Telnet	108
6.1	Net::FTP	108
6.1.1	一个Net::FTP例子	108
6.1.2	FTP和基于命令的协议	109
6.1.3	Net::FTP API	111
6.1.4	一个目录镜像脚本	114
6.2	Net::Telnet	120
6.2.1	一个简单的Net::Telnet例子	121
6.2.2	Net::Telnet API	122
6.2.3	一个远程改变密码的程序	126
6.2.4	对非Telnet协议使用Net::Telnet	130
6.2.5	Expect模块	134
6.3	小结	134
第7章	SMTP: 发送邮件	135
7.1	邮件模块简介	135
7.2	Net::SMTP	135
7.2.1	SMTP协议	135
7.2.2	Net::SMTP API	137
7.2.3	使用Net::SMTP	139
7.3	MailTools	141
7.3.1	使用MailTools	141
7.3.2	Mail::Header	142
7.3.3	Mail::Internet	145
7.3.4	一个邮件自动响应程序	146
7.3.5	Mail::Mailer	149
7.4	MIME-Tools	151
7.4.1	MIME简介	151
7.4.2	MIME::*模块的组织	154
7.4.3	MIME::Entity	156
7.4.4	MIME::Head	161
7.4.5	MIME::Body	162
7.4.6	MIME::Parser	164
7.4.7	MIME例子: 邮寄新近的CPAN项	168
7.5	小结	171

第8章 POP, IMAP和NNTP: 处理邮件和网络新闻	172
8.1 POP	172
8.1.1 POP3邮箱概述	172
8.1.2 Net::POP3 API	174
8.1.3 通过POP检索并处理MIME消息	176
8.1.4 pop_fetch.pl脚本	178
8.1.5 PopParser模块	184
8.2 IMAP协议	186
8.2.1 IMAP邮箱概述	187
8.2.2 Net::IMAP::Simple API	188
8.3 Internet新闻客户端	190
8.3.1 Net::NNTP	192
8.3.2 Net::NNTP API	194
8.4 一个新闻到邮件的网关	201
8.5 小结	207
第9章 Web客户	208
9.1 安装LWP	208
9.2 LWP基础知识	209
9.2.1 HTTP::Request	211
9.2.2 HTTP::Response	214
9.2.3 LWP::UserAgent	218
9.3 LWP实例	222
9.3.1 抓取RFC列表	222
9.3.2 镜像RFC列表	223
9.3.3 模拟填充表单	224
9.3.4 使用HTTP::Request::Common发送填充表单	229
9.3.5 使用multipart/form-data的文件上传	231
9.3.6 抓取有密码保护的页面	235
9.4 解析HTML和XML	237
9.4.1 格式化HTML	237
9.4.2 HTML::Formatter API	239
9.4.3 HTML::TreeBuilder API	239
9.4.4 从get_url.pl脚本返回格式化的HTML	241
9.4.5 HTML::Parser模块	242
9.4.6 使用HTML::Parser	244
9.4.7 HTML::Parser API	245
9.4.8 使用HTML::Parser的search_rfc.pl	247
9.4.9 从远程URL中提取图像	250