

高等院校计算机精品教材系列

计算机网络原理 实验教程

(第2版)



提供客户端软件、
源代码和配置文件

徐明伟 崔勇 徐恪 等编著



机械工业出版社
CHINA MACHINE PRESS

013033587

TP393-33
25-2

高等院校计算机精品教材系列

计算机网络原理实验教程

第2版

徐明伟 崔勇 徐恪 等编著



机械工业出版社



北航 C1640342

TP393-33
25-2

013033287

本书在简要介绍主要网络协议原理的基础上,重点讲述了对这些协议进行观察思考、编程实现和动手实践的基本方法。网络协议数量繁多,本书从中挑选了除物理层以外的包含所有网络层次的 21 个主要协议进行观察分析,并进一步从中精选设置了 11 个编程开发实验。本书还安排了对 4 种常用网络设备、8 种典型网络服务器的配置管理实践。本书以 Windows 和 Linux 为操作系统平台,以 Wireshark 为网络协议分析工具,以 NetRiver 实验系统为协议编程开发和测试环境,以 Socket 为网络应用编程接口。

本书可作为计算机、电子工程、通信、自动化等信息类专业的高年级本科生和研究生的实验教材,也可作为有一定计算机网络原理基础的工程技术人员人员的参考书。

本书提供客户端软件、源代码和配置文件,需要的教师可登录 www.cmpedu.com 免费注册,审核通过后下载,或联系编辑索取(QQ: 2399929378, 电话: 010-88379753)。

图书在版编目(CIP)数据

计算机网络原理实验教程 / 徐明伟等编著. —2 版. —北京: 机械工业出版社, 2013.4

高等院校计算机精品教材系列

ISBN 978-7-111-41864-1

I. ①计… II. ①徐… III. ①计算机网络—高等学校—教材 IV. ①TP393

中国版本图书馆 CIP 数据核字(2013)第 053716 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 郝建伟 责任印制: 张楠

中国农业出版社印刷厂印刷

2013 年 4 月第 2 版·第 1 次印刷

184mm×260mm·16.75 印张·417 千字

0001—3000 册

标准书号: ISBN 978-7-111-41864-1

定价: 36.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心: (010) 88361066

教材网: <http://www.cmpedu.com>

销售一部: (010) 68326294

机工官网: <http://www.cmpbook.com>

销售二部: (010) 88379649

机工官博: <http://weibo.com/cmp1952>

读者购书热线: (010) 88379203

封面无防伪标均为盗版

前 言

以因特网为代表的计算机网络是 20 世纪人类最伟大的发明之一。计算机网络已经成为支撑现代经济发展、社会进步和科技创新的基础设施。掌握计算机网络原理及关键技术是对信息学科学生和专业人员的基本要求，这通常需要经历 4 个阶段：理论学习、观察思考、编程实现和动手实践。理论学习是通过课堂教学和资料阅读来了解基本原理；观察思考是通过观察实际网络协议的运行状态和交互过程，进一步思考和理解协议的特点；编程实现是通过编写程序来实现具体的协议功能，从而深入了解网络协议的实现机理；动手实践则通过亲自动手配置网络软硬件，运用综合技能来搭建网络并提供网络管理和服务。

本书在简要介绍主要网络协议原理的基础上，重点讲述了对这些协议进行观察思考、编程实现和动手实践的基本方法。网络协议数量繁多，RFC 文本个数已经超过 6000。本书从中挑选了除物理层以外的包含所有网络层次的 21 个主要协议进行观察分析（其中 4 个协议配置观察实验放在机械工业出版社和清华大学计算机网络实验室网站上），并进一步精选设置了 11 个编程开发实验。在网络协议实验的基础上，本书还安排了对 4 种常用网络设备、8 种典型网络服务器的配置管理实践。

本书选择的实验环境以 Windows 和 Linux 为操作系统平台，以 Wireshark 为网络协议分析工具，以 NetRiver 实验系统为协议编程开发和测试环境，以 Socket 为网络应用编程接口。网络实验往往需要购置一定数量的网络设备，搭建复杂的网络环境，建设成本高，却很难支持典型协议的开发实验。基于 Linux 等开源系统的编程实验虽然支持典型协议开发，但会使实验人员在实验细节上投入大量精力，效率很低。清华大学计算机系网络所在高性能 IPv4/IPv6 路由器等科研成果的基础上研制开发了 NetRiver 网络教学实验系统，提供了支持各层协议开发的编程接口和辅助函数，实验人员能够集中精力实现网络协议的核心机制，而无须关心烦琐的细节。NetRiver 实验系统提供了支持程序编辑、编译、调试、可视化执行、自动测试和用户管理的一体化实验环境。

本书有以下几点特色。

- 理论实践结合。每一个实验前，都简要介绍了实验相关的协议、设备或应用的原理，使得本书成为一个自包含的系统，不需要读者再去查阅其他资料。
- 内容全面深入。本书内容覆盖了主流的 Windows 和 Linux 两种操作系统，IPv4 和 IPv6 两套因特网协议栈，以及除物理层以外的所有协议层次。作者从繁多的网络协议中精选了 21 个主要协议进行实验，并把协议核心机制作为实验要点。
- 层次清晰递进。实验安排从易到难，逐级递进。基础配置观察实验是初级实验，通过这些实验可以掌握协议的运行机制和交互过程。高级编程开发实验是高级实验，通过这些实验可以加深对协议的理解，培养编程开发能力。通过网络综合实践，可以综合运用多方面知识，锻炼解决实际问题的能力。
- 实验环境简单完备。基础配置观察实验只需要基本的网络环境。高级编程开发实验

主要基于 NetRiver 实验系统，不需要大量的网络设备和复杂的网络环境，就可以进行各层协议的编程开发实验。

全书分为 5 章。第 1 章介绍实验准备，包括 Windows 和 Linux 操作系统下的各种网络配置命令和工具、网络协议分析工具 Wireshark 的使用方法、Socket 编程基础，以及 NetRiver 实验系统。第 2 章按照计算机网络分层模型，自底向上安排了 17 个协议的配置观察实验。第 3 章安排了 11 个编程开发实验，其中 10 个实验基于 NetRiver 实验系统，1 个实验基于标准 Socket。NetRiver 实验系统还提供了图形界面交互实验，实验人员只需根据协议规范，通过交互式图形界面填写相关分组内容即可完成实验，降低了实验难度。而且在学习过程中，可根据情况选择交互实验或者编程实验。第 4、5 章进一步融合了计算机网络各方面的知识，以亲自动手配置网络、实现网络管理和服务为目标，进行综合实践。第 4 章介绍了交换机、路由器、NAT 网关和防火墙等常用网络设备的配置管理方法。第 5 章介绍了 DHCP、DNS、Web、FTP、E-mail、TELNET、SNMP 和 VPN 等典型网络服务器的配置方法。附录中列出了本书相关工具软件的下载网址，请读者自行下载使用。附录中的所有网址在本书出版时均已经过验证，可以正常登录。

本书的配套资料放在机械工业出版社网站 (<http://www.cmpedu.com>) 和清华大学计算机网络实验室网站 (<http://www.netlab.edu.cn>) 上，包含以下内容。

- NetRiver 实验系统客户端软件和使用手册。
- Windows 和 Linux 下 Socket 编程准备。
- Windows 和 Linux 下 Socket 编程实例的源代码和相关工程配置文件。
- ND、SNMP、TELNET、SMTP 与 POP3 配置观察实验。

通过 NetRiver 客户端可以访问 NetRiver 服务器，并进行一些基本的实验。如果想进行更全面的实验，建议配备全套 NetRiver 实验系统。NetRiver 实验系统的相关事宜可以咨询 <http://www.netlab.edu.cn>。

本书可作为计算机、电子工程、通信、自动化等信息类专业的高年级本科生和研究生的实验教材，也可作为有一定计算机网络原理基础的工程技术人员的参考书。

本书由徐明伟、崔勇和徐恪共同编写，由徐明伟完成全书的统稿。在编写过程中，周云涛、杨荒、王影新、陈蒙、张媛媛、季佳、刘峥等同学参与了实验设计和代码编写。本书是作者多年教学实践工作的总结，历届助教博士生都对网络实验的设计和完美做出了贡献。在此对以上人士表示谢意。

由于作者水平有限，书中难免存在不妥之处，欢迎读者提出宝贵意见。

编者

目 录

前言

第 1 章 实验准备	1
1.1 Windows 系统网络操作	2
1.1.1 图形界面网络配置	2
1.1.2 基本命令行配置 ipconfig	4
1.1.3 网络配置脚本 netsh	5
1.1.4 路由配置 route	8
1.1.5 IPv6 网络配置	12
1.1.6 基本网络诊断技术	14
1.1.7 高级网络诊断技术	16
1.2 Linux 系统网络操作	20
1.2.1 基本网络配置 ifconfig	21
1.2.2 路由配置 route	22
1.2.3 基本网络诊断技术	23
1.2.4 高级网络诊断技术	25
1.3 网络协议分析工具 Wireshark	28
1.3.1 简介	28
1.3.2 安装 Wireshark	30
1.3.3 捕捉数据分组	32
1.3.4 分组格式分析	35
1.3.5 高级话题	35
1.4 Socket 与网络编程	38
1.4.1 基本概念	38
1.4.2 网络字节序转换	38
1.4.3 Windows Sockets 编程接口	39
1.4.4 Windows 下的服务器端编程实例	41
1.4.5 Linux Sockets 编程接口	45
1.4.6 Linux 下的客户端编程实例	48
1.5 计算机网络实验系统 NetRiver	49
1.5.1 实验系统介绍	50
1.5.2 实验准备	51
1.5.3 实验程序编辑与编译	52
1.5.4 实验程序调试与报文分析	53
1.5.5 实验程序测试与提交	55

第 2 章 基础配置观察实验	56
2.1 IEEE 802.3 协议	56
2.2 PPP	58
2.3 IPv4	64
2.4 ICMPv4	69
2.5 ARP	72
2.6 DHCPv4	78
2.7 NAT/NAPT 协议	84
2.8 IPv6	88
2.9 TCP	92
2.10 UDP	97
2.11 RIP	99
2.12 OSPF 协议	102
2.13 BGP	107
2.14 DNS 协议	113
2.15 HTTP	118
2.16 FTP	123
2.17 BT 协议	128
第 3 章 高级编程开发实验	134
3.1 滑动窗口协议实验	134
3.2 IPv4 收发实验	139
3.3 IPv4 转发实验	144
3.4 IPv6 收发实验	148
3.5 IPv6 转发实验	153
3.6 RIP 实验	157
3.7 IPSec 实验	164
3.8 移动 IP 实验	177
3.9 简单 TCP 实验	187
3.10 BGP 状态机实验	196
3.11 FTP 实验	204
第 4 章 网络设备综合实践	206
4.1 网络综合实践环境	206
4.2 配置路由器	207
4.2.1 路由器简介	207
4.2.2 路由器配置需求	207
4.2.3 商用路由器配置	208
4.2.4 Linux 路由器配置	213
4.2.5 路由器配置验证	215
4.3 配置交换机	215

4.3.1	交换机介绍	215
4.3.2	交换机配置需求	216
4.3.3	交换机配置	217
4.3.4	交换机配置验证	218
4.4	配置 NAT 网关	219
4.4.1	配置 NAT 网关需求	219
4.4.2	NAT 网关配置	219
4.4.3	NAT 网关配置验证	221
4.5	配置网络防火墙	221
4.5.1	网络防火墙配置需求	221
4.5.2	网络防火墙启用	221
4.5.3	网络防火墙配置	221
4.5.4	网络防火墙验证	223
第 5 章	网络服务综合实践	225
5.1	配置 DHCP 服务器	225
5.1.1	DHCP 服务器配置需求	226
5.1.2	DHCP 服务器安装	226
5.1.3	DHCP 服务器配置	226
5.1.4	DHCP 服务验证	228
5.2	配置 DNS 服务器	228
5.2.1	DNS 服务器配置需求	229
5.2.2	DNS 服务器安装	229
5.2.3	DNS 服务器配置	229
5.2.4	DNS 服务验证	231
5.3	配置 Web 服务器	232
5.3.1	Web 服务器配置需求	232
5.3.2	Web 服务器安装	232
5.3.3	Web 服务器配置	234
5.3.4	Web 服务验证	236
5.4	配置 FTP 服务器	236
5.4.1	FTP 服务器配置需求	236
5.4.2	FTP 服务器安装	237
5.4.3	FTP 服务器配置	237
5.4.4	FTP 服务验证	239
5.5	配置电子邮件服务器	240
5.5.1	电子邮件服务器配置需求	240
5.5.2	电子邮件服务器安装	241
5.5.3	电子邮件服务器配置	241
5.5.4	电子邮件服务验证	243

5.6	配置 TELNET 服务器	244
5.6.1	TELNET 服务器配置需求	245
5.6.2	TELNET 服务器安装与配置	245
5.6.3	TELNET 服务验证	247
5.7	配置网管服务器	248
5.7.1	网管服务器配置需求	249
5.7.2	网管服务器安装	249
5.7.3	网管服务器配置	250
5.7.4	网管服务验证	252
5.8	配置 VPN 服务器	252
5.8.1	VPN 服务器配置需求	252
5.8.2	VPN 服务器安装	252
5.8.3	VPN 服务器配置	253
5.8.4	VPN 服务验证	256
附录 本书相关工具软件		257
参考文献		259

第1章 实验准备

在进行网络实验及实践之前，不仅要将与网络相关的软硬件环境配置好，还需学习基本的网络协议分析方法和编程技术。本章将介绍这些准备工作，包括 Windows 和 Linux 操作系统的网络配置方法、网络协议分析工具 Wireshark 的使用方法、网络编程基础知识，以及后续章节中网络实验需要用到的 NetRiver 系统的使用方法等。本章将从以下几个方面来介绍网络实验的准备。

1. Windows 系统网络操作

Microsoft Windows 是目前最常用的一种操作系统。本章将以 Windows XP 系统为例，介绍网络常见参数的配置方法，以及一些实用网络配置工具的使用方法。其中，网络配置包括使用图形界面和命令行工具配置系统的 IP 地址、DNS 和路由等，实用工具则主要是基本的诊断工具，如 arp、ping 等。

2. Linux 系统网络操作

Linux 操作系统与因特网的迅速发展有着密切的联系，Linux 系统所提供的多种网络服务和开源的网络编程接口也极大地推动了 Linux 操作系统的广泛应用。本章将介绍 Linux 操作系统的网络配置方法，包括 IP 地址、DNS 和路由配置等部分，还将介绍包括 arp、ping、traceroute、nslookup 和 netstat 等命令在内的各种实用工具及其使用方法，便于读者掌握基本的网络诊断技术。

3. 网络协议分析工具 Wireshark 的介绍

要想深入地学习网络知识，就必须对网络中运行的各种主要协议有一定的了解。而了解网络协议运行过程的最好方法就是进行实际观察，即在真实的网络环境中，使用一定的工具截获网络中传送的数据分组，对其内容进行观察和分析，从而了解协议的运行机制。本章将介绍目前最流行的开源协议分析工具 Wireshark，详细说明软件安装、基本捕捉分析功能，以及诸如 TCP 流的跟踪等高级功能。掌握 Wireshark 的这些功能后，有助于理解协议的基本原理，进而指导后续的网络配置和网络编程实验。

4. Socket 与网络编程知识介绍

真正的实践一定要亲自动手编写网络程序，只有这样才能够对系统的网络各层结构以及网络相关编程接口有更深入的了解。无论是 Windows 系统还是 Linux 系统，它们都有一套用于编程的 Socket 接口，提供了标准的网络操作规程。本章将在简单介绍 Socket 的基本概念后，具体讲解 Windows 系统和 Linux 系统下的 Socket 编程接口，并提供两个平台下的编程实例，为读者后续的网络编程奠定基础。

5. 网络实验系统 NetRiver 介绍

很多高校为了做网络实验，要么购买大量的路由器和交换机等网络设备，组建一定规模的局域网，然后让学生在此基础上进行配置观察实验；要么让学生直接在 Linux 上进行一定的编程开发。那些配置观察实验不仅需要大量设备费投资，还难以让学生真正理解网络协

议运行的深层机理；而基于 Linux 的编程开发实验则需要学生将大量精力耗费在程序代码与操作系统接口等细枝末节的调试上，难以把主要精力投入到掌握多种代表性网络协议的基本原理上。针对这个问题，清华大学开发了一套网络实验系统 NetRiver。该系统为用户提供了一个自动化的开发、测试平台，提供了两种基本的实验方法，即交互式实验和编程开发实验。交互式实验让用户根据协议规范，通过图形界面的方式填写消息分组中各字段的值；编程开发实验则让用户完成协议核心函数代码，以促进对协议运行过程的理解。本章将介绍网络实验系统 NetRiver，以便于后续实验的开展。

1.1 Windows 系统网络操作

Windows 是目前使用最广泛的操作系统之一。本节将以 Windows XP 为例，介绍在 Windows 系统下查看和配置网络的方法，以及 Windows 系统中常见的网络诊断实用工具的使用方法。

一般来说，在 Windows 下可以使用图形界面对网络进行配置，也可以使用命令行对网络进行配置。配置和查看的内容可以从基本的本机 IP 地址、子网掩码、网关 IP 地址等，到较为高级的本地转发路由等。下面将以最常见的 Windows XP 操作系统为例，首先介绍在图形界面下如何查看和配置 IP 地址及网关等信息，然后介绍在命令行下进行相关配置的命令及使用方法。

1.1.1 图形界面网络配置

Microsoft Windows 系统为用户提供了便捷的图形界面工具，以使用户方便地配置本机 IP 地址、子网掩码和网关 IP 地址等。下面将以 Windows XP 为例，介绍在 Windows 环境下使用系统提供的图形界面工具对 IP 地址等相关网络信息进行查看和配置的方法。

1. 基本图形界面

网络配置的图形界面工具可在控制面板中找到。具体操作是，首先打开“控制面板”，进入“网络连接”，双击“本地连接”，在“本地连接状态”对话框中单击“属性”按钮，即可打开“本地连接属性”对话框，如图 1-1 所示。

在“本地连接属性”对话框中，用户可对网络的主要基本信息进行查看和配置。本书仅讨论 TCP/IP 协议栈，因此双击列表中的“Internet 协议 (TCP/IP)”，打开如图 1-2 所示的“Internet 协议 (TCP/IP) 属性”对话框。在该对话框中，可对网络信息进行相应的配置。

2. 基本配置

计算机在因特网上通常需要配置和使用 IP 地址等信息以便与其他计算机通信，这些信息包括如图 1-2 所示对话框中的主机 IP 地址、子网掩码、网关 IP 地址以及域名系统 (DNS) 等信息。在图 1-2 所示的设置中，IP 地址和 DNS 服务器地址都使用自动配置的方式，即当主机所在的网络能够提供动态主机配置协议 (Dynamic Host Configuration Protocol, DHCP) 的支持时，系统会在启动时自动获取并配置 IP 地址、DNS 服务器地址等信息。

如果网络不提供 DHCP 的支持，或者出于某些需求要静态配置本机地址信息，可使用如图 1-3 所示的静态配置方式。

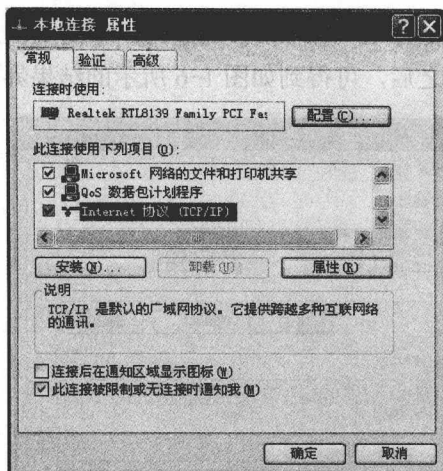


图 1-1 “本地连接属性”对话框

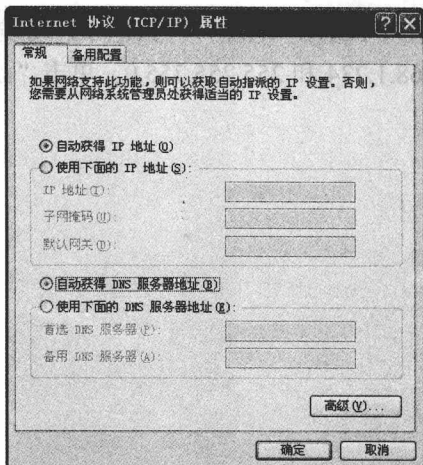


图 1-2 “Internet 协议 (TCP/IP) 属性”对话框

在图 1-3 所示的配置中，主机所使用的 IP 地址是 192.168.1.69，子网掩码是 255.255.255.0，默认的网关地址是 192.168.1.1，所使用的两个 DNS 服务器地址分别是 166.111.8.28 和 166.111.8.29。

3. 多 IP 配置

在图 1-2 和图 1-3 所示的配置界面中，可以完成对系统的基本地址配置。当用户使用计算机上网时，有时需要为每个网络接口（即网卡）配置多个 IP 地址。例如，一些服务器需要多个不同的 IP 地址，以提供多个 Web 服务。下面介绍如何为系统配置多个 IP 地址。

单击图 1-2 或图 1-3 所示对话框中的“高级”按钮，打开如图 1-4 所示的“高级 TCP/IP 设置”对话框。

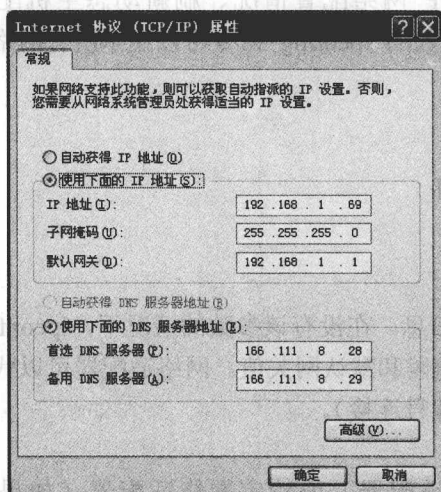


图 1-3 “Internet 协议 (TCP/IP) 属性”对话框 (静态配置)

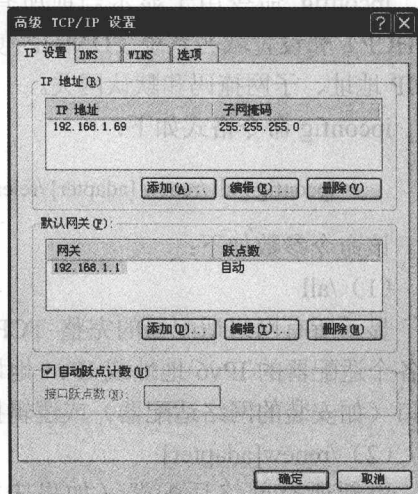


图 1-4 “高级 TCP/IP 设置”对话框

在图 1-4 所示的对话框中，可以添加、编辑或删除主机的 IP 地址。例如，单击该对话框中的“添加”按钮，打开如图 1-5 所示的“TCP/IP 地址”对话框。

在“TCP/IP 地址”对话框中，可输入想要添加的 IP 地址和子网掩码，如图中所示的 192.168.1.234 和 255.255.255.0。单击“添加”按钮之后，可得到如图 1-6 所示的结果界面。

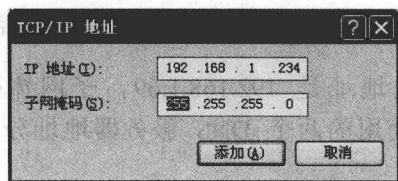


图 1-5 “TCP/IP 地址”对话框

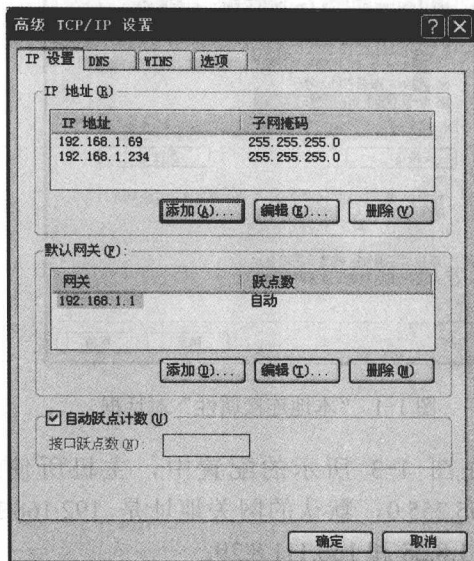


图 1-6 “高级 TCP/IP 设置”对话框（添加地址之后）

在添加地址之后的“高级 TCP/IP 设置”对话框中，可以看到“IP 地址”列表中已经有刚才添加的 IP 地址和相应的子网掩码了。

1.1.2 基本命令行配置 ipconfig

ipconfig 命令用于显示当前所有的 TCP/IP 网络配置情况、刷新动态主机配置协议 (DHCP) 和设置域名系统 (DNS)。使用不带参数的 ipconfig 命令可以显示所有网络适配器的 IP 地址、子网掩码和默认网关。

ipconfig 命令格式如下：

```
ipconfig [/all /renew [adapter] /release [adapter]]
```

该命令参数如下：

(1) /all

显示所有网络适配器的完整 TCP/IP 配置信息。在没有该参数的情况下，ipconfig 只显示各个适配器的 IPv6 地址或 IPv4 地址、子网掩码和默认网关值。网络适配器可以代表物理接口（如安装的网络适配器）或逻辑接口（如拨号连接）。

(2) /renew[adapter]

更新所有网络适配器（如果未指定网络适配器）或特定网络适配器（如果包含了 adapter 参数）的 DHCP 配置。该参数仅在网络适配器配置为自动获取 IP 地址的计算机上可用。若要指定网络适配器名称，请输入 ipconfig/all 命令所显示的网络适配器名称。

(3) /release[adapter]

发送 DHCP release 消息到 DHCP 服务器，以释放所有网络适配器（如果未指定网络适

配器)或特定网络适配器(如果包含了 adapter 参数)的当前 DHCP 配置,并丢弃 IP 地址配置。该参数可以禁用配置为自动获取 IP 地址的网络适配器。

例如,如果需要查看所有网络适配器的 TCP/IP 配置信息,可以在命令提示符下输入: ipconfig/all, 如图 1-7 所示。从图中可以看出,当前主机的本地连接的介质访问控制(Media Access Control, MAC)地址为 00-16-D4-68-5D-85,未使用 DHCP,IP 地址为 192.168.1.168,子网掩码为 255.255.255.0,默认网关为 192.168.1.1,DNS 服务器为 166.111.8.28,然后是无线网络连接的相关地址信息。

```
C:\Documents and Settings\xerry>ipconfig /all

Windows IP Configuration

Host Name . . . . . : xiaodong
Primary Dns Suffix . . . . . :
Node Type . . . . . : Unknown
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No

Ethernet adapter 本地连接:

Connection-specific DNS Suffix . :
Description . . . . . : Broadcom 440x 10/100 Integrated Cont
roller
Physical Address. . . . . : 00-16-D4-68-5D-85
Dhcp Enabled. . . . . : No
IP Address. . . . . : 192.168.1.168
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1
DNS Servers . . . . . : 166.111.8.28

Ethernet adapter 无线网络连接:

Media State . . . . . : Media disconnected
Description . . . . . : Intel(R) PRO/Wireless 3945ABG Networ
k Connection #2
Physical Address. . . . . : 00-13-02-6B-4D-04
```

图 1-7 使用 ipconfig 命令查看 TCP/IP 配置信息

1.1.3 网络配置脚本 netsh

netsh 是命令行使用脚本进行网络配置的实用工具。它可以在本地或远程使用,允许显示或修改当前正在运行的计算机的网络配置。netsh 所提供的脚本功能,可以以批处理模式运行一组命令。为了将配置存档或将来进一步配置其他服务器,netsh 也可以将配置脚本保存在文本文件中。

netsh 使用动态链接库(DLL)文件来与其他操作系统组件进行交互。每个 netsh 帮助程序都提供一组称为上下文的扩展功能,这是网络组件专用的一组命令。这些上下文扩展了 netsh 的功能,可以对一个或多个服务、实用工具或协议进行配置和监视。例如,Dhcpmon.dll 为 netsh 提供了配置和管理 DHCP 服务器所必须的上下文和命令组。

1. 运行 netsh 命令

要运行 netsh 命令,需要先运行 cmd.exe 进入命令行界面,然后在命令行界面中输入 netsh,从而进入 netsh 命令包含的上下文。具体步骤如下。

- 1) 在 Windows “开始”菜单的“运行”对话框中输入 cmd.exe,打开“命令提示符”。

2) 在命令提示符下输入 netsh。

3) 输入表 1-1 右边一列中的某项命令，从而进入相应的网络组件配置的上下文。

表 1-1 netsh 上下文及其相关命令

netsh 上下文	netsh 下的命令	netsh 上下文	netsh 下的命令
AAAA	aaaa	Internet 协议安全性	ipsec
DHCP	dhcp	网桥	bridge
网络诊断 (diag)	diag	远程访问	ras
接口	interface	路由	routing
接口 IP	interface ip	RPC 助手	rpc
接口 IPv6	interface ipv6	WINS	wins
接口 Portproxy	interface portproxy		

例如，如果在 netsh 命令提示符下输入 dhcp，则进入 DHCP 上下文，但是如果没有安装 DHCP，则将出现消息“下列命令没有找到：dhcp”。

2. IP 地址配置

netsh 命令中，可以使用 netsh Interface IP 上下文中的命令配置 TCP/IP，包括 IP 地址、默认网关、DNS 服务器和 WINS 服务器，并显示配置和统计信息。下面将介绍如何使用 netsh 上下文来进行 TCP/IP 配置。首先按照前面的方法进入网络适配器的 IP 配置上下文，在 IP 配置上下文的基础上，再使用 set 命令配置指定接口上的 IP 地址和默认网关。

set 命令格式如下：

```
set address [name=]InterfaceName [source=]{dhcp | static [addr=]IPAddress [mask=]SubnetMask [gateway=]{none | DefaultGateway [[gwmetric=] GatewayMetric]}}
```

主要参数如下：

(1) [name =] InterfaceName

指定要配置其地址和网关信息的接口的名称，为必填参数。InterfaceName 参数必须与“网络连接”中指定的接口名称匹配。如果 InterfaceName 包含空格，则请将文本置于引号之中（例如，“Interface Name”）。

(2) [source=]{dhcp | static [addr=]IPAddress [mask=]SubnetMask [gateway=]{none | Default Gateway [[gwmetric=]GatewayMetric]}}

指定是通过动态主机配置协议（DHCP）服务器配置 IP 地址还是使用静态 IP 地址，为必填参数。如果使用静态地址，那么 IPAddress 将指定要配置的 IP 地址，而 SubnetMask 将指定所配置 IP 地址的子网掩码。如果使用静态地址，还必须指定是保留当前默认的网关（如果有）还是为该地址配置一个网关。如果配置网关，则 DefaultGateway 将指定要配置的网关的 IP 地址，而 GatewayMetric 指定要配置的网关的跳数（即网络距离）。

例如，要为某台计算机的“本地连接”指定一个静态的 IP 地址为 192.168.1.25，并设置其子网掩码为 255.255.255.0，默认网关为 192.168.1.1。可以使用前面所述的图形界面配置方法，也可以在命令提示符下进行如下操作。

输入：

```
netsh
```

出现:

```
netsh>
```

输入:

```
interface
```

出现:

```
netsh interface>
```

输入:

```
ip
```

这样就进入了 IP 地址配置的上下文。

输入:

```
set address name = "本地连接" source = static addr = 192.168.1.25 mask = 255.255.255.0 gateway = 192.168.1.1 gwmetric = 1
```

这样就完成了对本地连接的静态 IP 地址的配置，然后还可以使用 `ipconfig/all` 命令来查看配置结果。

3. DNS 配置

通过 `netsh` 上下文来进行 DNS 配置，仍然需要首先进入网络适配器的 IP 配置上下文，然后再使用 `set dns` 命令配置指定接口上的 DNS。

`set dns` 命令格式如下:

```
set dns [name=]InterfaceName [source=]{dhcp | static} [addr=]{IPAddress | none} [register=]{none | primary | both}
```

主要参数如下:

(1) `[name=]InterfaceName`

指定要设置 DNS 信息的接口的名称。`InterfaceName` 参数必须与“网络连接”中指定的接口名称匹配。如果 `InterfaceName` 包含空格，则请将文本置于引号中（例如“Interface Name”）。

(2) `[source=]{dhcp|static}`

指定 DNS 服务器的 IP 地址是通过 DHCP 配置的还是为静态地址。

(3) `[addr=]{IPAddress|none}`

如果是静态 IP 地址，则 `IPAddress` 指定要配置的 DNS 服务器的 IP 地址，而 `none` 指定应删除的 DNS 配置。

(4) `[register=]{none|primary|both}`

`none` 指定是否禁用动态 DNS 解析，`primary` 指定仅在主 DNS 后缀下注册计算机域名，`both` 指定在主 DNS 后缀和特定连接的后缀下注册计算机域名。

值得一提的是，如果接口已配置为静态地址，则 `static` 参数将使用命令中指定的值替换

现有的 DNS 服务器地址列表。只有在[source]是 static 时, [addr]选项才可以用来为指定的接口配置 DNS 服务器 IP 地址的静态列表。

例如, 给某台计算机的“本地连接”指定一个 DNS 服务器的 IP 地址为 166.111.8.28, 则可以在命令提示符下进行如下操作。

输入:

```
netsh
```

出现:

```
netsh>
```

输入:

```
interface
```

出现:

```
netsh interface>
```

输入:

```
ip
```

出现

```
netsh interface ip>
```

输入

```
set dns name = "本地连接" source = static addr = 166.111.8.28
```

这样就完成了为“本地连接”指定 DNS 服务器的 IP 地址 166.111.8.28 的配置, 然后可以使用 ipconfig /all 命令来查看配置结果。

1.1.4 路由配置 route

路由是网络中最重要的概念, 任何分组在网络中的传递都要通过路径选择并逐条经过路由转发才能够到达目的地。对于主机来说, 路由的目的在于将从本机发出的分组送到网关, 再由网关转发至正确的目的地。

1. 命令格式与参数

Windows 系统中的路由配置命令 route, 能够显示和设置本地 IP 路由表。使用不带参数的 route 命令可以显示帮助信息。

route 命令格式如下:

```
route [-f] [-p] [Command [Destination] [mask Netmask] [Gateway] [metric Metric]] [if Interface]
```

主要参数如下:

(1) -f

清除所有不是主机路由 (网络掩码为 255.255.255.255 的路由)、环回网络路由 (目的地址为 127.0.0.0, 网络掩码为 255.255.255.0 的路由) 或组播路由 (目标为 224.0.0.0, 网络掩