SCO UnixWare 7 网络教程

叶国庄　编著

- 详尽讲述拥有最新技术的操作系统平台 UnixWare 7 系统
- 从网络的应用和管理出发，指导用户配置 UnixWare 7 系统
- 针对具体问题，提出详细的解决方案，指导用户管理 UnixWare 网络

人民邮电出版社
www.pptph.com.cn

SCO UnixWare 7
网络教程

叶国庄　编著

人民邮电出版社

图书在版编目（CIP）数据

SCO UnixWare 7网络教程／叶国庄编著. —北京：人民邮电出版社，2001.11
ISBN 7-115-09714-3

Ⅰ.S... Ⅱ.叶... Ⅲ.UNIX操作系统—教材 Ⅳ.TP316.81

中国版本图书馆CIP数据核字(2001)第069121号

内 容 提 要

本书介绍了网络的基本概念、UnixWare网络的文件系统及其配置、UnixWare 的PPP协议、UnixWare路由配置、UnixWare的DNS及邮件服务、UnixWare的远程服务及UnixWare的系统备份及安全等相关方面的知识。

本书内容新颖，通俗易懂，并针对具体问题提出详细的解决方案，可操作性强。本书可作为使用SCO UnixWare的中、高级技术人员的指导用书，也可以作为UNIX用户自学的参考用书。

SCO UnixWare 7 网络教程 TP316.8/06

◆ 编　　著　叶国庄
责任编辑　张瑞喜　赵桂珍
执行编辑　牛　磊

◆ 人民邮电出版社出版发行　北京市崇文区夕照寺街14号
邮编　100061　　电子函件　315@ pptph.com.cn
网址　http://www.pptph.com.cn
读者热线:010－67129212　010－67129211(传真)
北京汉魂图文设计有限公司制作
北京顺义向阳胶印厂印刷
新华书店总店北京发行所经销

◆ 开本:787×1092　1/16
印张:15.5
字数:371千字　　2001年11月第1版
印数:1－5 000册　　2001年11月北京第1次印刷
ISBN 7-115-09714-3/TP·2512

定价:25.00元

前　言

UNIX系统从诞生到现在已经有30多年的历史了，经过这么长时间的风风雨雨，UNIX当之无愧地成为主流操作系统之一，在科研、教育和商业领域发挥了巨大的作用，然而长期以来，UNIX系统一直是计算机专业人员的“专利”，寻常百姓无从见到。但现在不同了，随着网络时代的到来和Internet的普及，人们发现Internet上的大部分主机都是UNIX主机，因此，掌握UNIX操作系统的使用方法，已成为大家目前的迫切需要。

从UNIX诞生以来，SCO UNIX系统以其稳定、可靠等特点，在大型数据库、应用软件等信息传输方面占据了比较明显的优势，1994年SCO UNIX进入中国，中国用户对SCO UNIX在网络计算领域的可靠性、稳定性等认识逐步加强，其SCO OpenServer产品在中国邮政业、中国银行业、信托投资公司等一批金融和非金融机构中得到广泛的应用。1997年SCO从Novell公司购买了UNIX系统源代码许可业务和UnixWare产品线，将SCO OpenServer的功能合并到UnixWare中，形成一个集两个操作系统的优点于一体，并具有最新技术的操作系统平台UnixWare 7系统。该产品的推出使SCO公司的市场占有率得到很大的提高，UnixWare产品得到广泛应用。

为了大家尽快学习UnixWare 7这种强大的操作系统，我们根据使用OpenServer和UnixWare 7开发系统的经验，并参考了UnixWare 7的英文手册，编著了这套UnixWare 7的教程，包括《SCO UnixWare 7实用教程》和《SCO UnixWare 7网络教程》两本。这是一套理想的指导用户配置和使用该产品的指导性丛书。

《SCO UnixWare 7网络教程》从网络的应用和管理出发，介绍了网络的基本概念、UnixWare网络的文件系统及其配置、UnixWare 的PPP协议、UnixWare路由配置、UnixWare的DNS及邮件服务、UnixWare的远程服务、UnixWare的系统备份及安全等相关方面的知识。

本书内容新颖，通俗易懂，并针对具体问题，提出详细的解决方案，可操作性强，本书可作为使用SCO UnixWare的中、高级技术人员的指导用书，同时也可以作为UNIX用户自学和参考用书。

由于编者水平有限，对于本书存在的缺点和不足，敬请广大读者指正。

编　者

目　录

第 1 章　TCP/IP 网络导论

UNIX系统和其他操作系统相比，最重要的特点就是有卓越的联网功能，同时支持多种网络协议。网络协议中最常用的是TCP/IP协议。本章从UNIX系统的硬件连接开始，介绍在UNIX系统下如何联网和TCP/IP协议是如何工作的。此部分的内容适用于所有的UNIX系统，如果学习过其他UNIX系统的网络知识，可以略过本章内容，直接跳到下一章继续学习。

1.1　UNIX系统连接

1.1.1　系统连接的基本概念

1. 主机与终端

UNIX系统在同一时刻能够支持多个用户，也就是说UNIX系统是一个多用户的分时系统。在UNIX系统中，可以把设备简单地分为主机和终端。带有处理功能，完成绝大部分工作的计算机设备叫主机（Host）；而只能进行输入或显示，没有进程处理能力的设备都可以称为终端（Terminal）。UNIX操作系统的任务之一就是调控多个用户（即多个终端）对主机资源的共享。因为一台UNIX计算机可以与任何其他的UNIX计算机相连接，所以，在网络中UNIX系统的作用非常重要。众多的UNIX计算机的连接是使用主机/终端（或仿真终端）这种方式进行的。

通过终端来使用UNIX系统是一个好的办法。一般地说，一个普通的UNIX终端由显示器、键盘等几个部分组成（也有部分终端更复杂一些，比如X-Window终端，它允许使用鼠标对图像进行操作）。很多机型都可以作为终端使用，通常选用PC机或Macintosh机作终端。这样，所选用的计算机就既可以作为一台独立的计算机单独使用，又可以作为终端连接到主机使用。

当终端和主机连接后，它们就形成一个整体。当终端上的用户按一下键或移动鼠标，相应的信号就传入主机，运行在主机上的程序就会对所接收到的信号进行处理，并把处理结果返回给终端，控制终端完成相应的动作。主机和终端之间的这种关系可以用图1.1表示。

随着对UNIX系统了解的深入，人们发现，即使主机距离用户较远（如几公里之外），用户也能将自己的终端连接到它上面去。在这种情况下，从键入字符或移动鼠标到在终端显示器上看到响应就需要一定的时间，3秒、5秒或更长，这一时间的长短取决于计算机的处理速度和传输介质的传输速率。

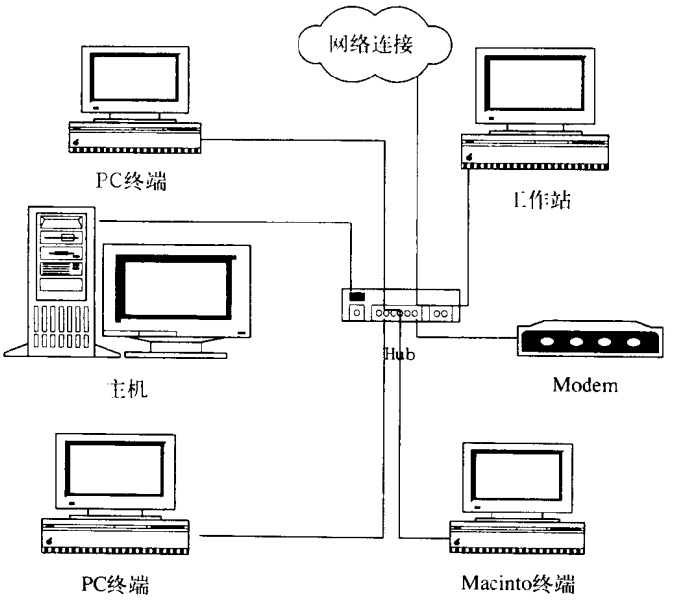

图1.1　主机和终端结构

2. 主机和子网

所谓计算机网络，是指相互连接在一起，能够完成一定功能的相对独立的计算机集合。计算机的连接要通过软件和硬件两方面来实现，“互连”意味着相互连接的两台计算机能够互相交换信息。信息的交换在底层由硬件实现，而在高层则由软件实现。

从概念上，无论是局域网（LAN，Local Area Network）、广域网（WAN，Wide Area Network）还是其他形式的计算机网络，我们总可以将它分为两部分：主机和子网。如图1.2所示。

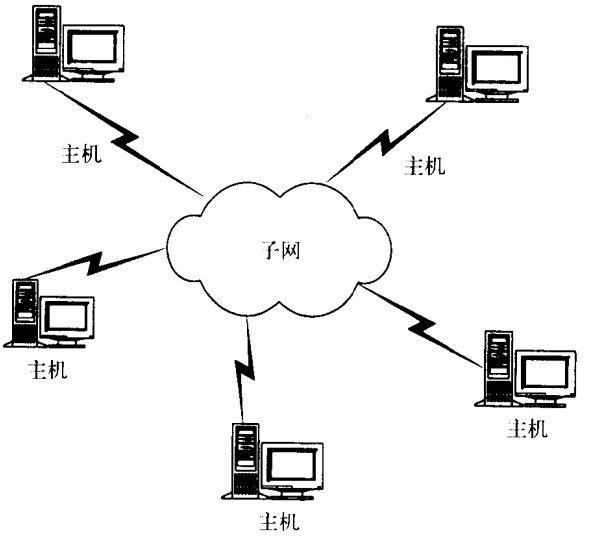

图1.2　UNIX计算机网络的结构

主机是组成计算机网络的独立计算机，用于运行用户程序（即运用程序）。子网也称为通信子网（Communication Subnet）是将入网主机连接起来的实体。

3. 拓扑结构

计算机网络的拓扑结构其实就是信道分布的结构。常见的拓扑结构有5种：总线型、星型、环型、树型和网型。不同的拓扑结构的信道访问技术、性能、设备开销等各不相同。

尽管不同的信道拓扑结构差别很明显，但总的来说可以分为两类：点到点（Point to Point）信道和广播（Broadcasting）信道。所谓点到点信道指网络中每两台主机、两台终端之间或主机和终端之间都存在一条物理信道，计算机（包括主机和终端）沿某信道发送的数据无疑只有信道另一端的唯一一台计算机收到。在这种点到点的拓扑结构中，没有信道竞争，几乎不存在信道访问控制问题。绝大多数广域网都采用点到点拓扑结构，尤其是网状结构。网状结构是典型的点到点拓扑，除此之外，星型结构也是点到点的。

在广播型拓扑结构中，所有主机共享一条信道，某主机发出的数据，所有其他主机都能收到。在广播信道中，由于信道共享会引起信道访问冲突，因此信道访问控制是必须解决的问题。广播型网的典型代表是总线网。

4. 数据交换方式

交换（Switch）的概念最早来自于电话系统。电话系统的交换方式叫作线路交换（Circuit Switching）。线路交换的外部表现是通信的两端一旦接通，便拥有一条实际的物理线路，双方独占此线路。线路交换的实质是在交换设备内部，硬件开关将输入线与输出线直接接通。

线路交换技术有两大优点：第一个优点得益于一旦建立连接便不再需要交换开销；第二个优点来自于独享物理线路。线路交换也有其缺点，首先它建立线路的时间很长，其次由于独享信道而造成了信道的浪费。

另一种交换技术叫做报文交换（Message Switching）。报文交换不事先建立线路，当发送方有数据块要发时，它把数据块作为一个整体（报文，Message）交给交换设备，交换设备选择一条合适的空闲输出线，将数据块通过该数据线传送出去。在这个过程中，可能经过若干交换设备，在每个交换设备处，报文首先被存储起来，在适当的时候再被转发出去，所以报文交换技术是一种存储转发（Store-and-Forward）技术。

另一种存储交换技术是分组交换（Packet Switching）。上面谈到的报文交换的特点是对传输数据块的大小不加限制，这不利于交互式通信。为了解决这个问题，分组交换技术严格限制数据块大小，保证任何用户不独占线路过长的时间，因此非常适合于交互式通信。分组交换比起报文交换的另一优点是吞吐率高：在具有多个分组的报文中，第二个分组尚未接收完之前，已接收到的第一个分组就可以往前传送，减小了时间延迟，提高了吞吐率。因此，交换方式其实质是在交换设备内部将数据从输入线切换到输出线的方式，两类交换方式——线路交换和存储交换的关键区别在于：前者静态分配线路，后者动态分配线路。

绝大多数通信子网采用分组交换技术，根据内部机制的不同，可以把分组交换子网分为两类：一类采用连接（即面向连接，Connect-Circuit），一类采用无连接（Connectless）。在有连接子网中，连接被称为虚电路（Virtual Circuit），类似于电话系统中的物理电路；无连接子网中的独立分组被称为数据报（Datagram），类似于邮政系统中的电报。

虚电路和数据报这两个概念不仅用于描述通信子网内部结构，实际上也是分组交换的两种形式，面向连接的分组交换技术通常称为虚电路，无连接分组交换技术称为数据报。

5. 服务类型

在计算机网络协议层次结构中，层与层之间是完全单向依赖的，相邻层之间通过一组服务原语（Service Primative）建立相互作用；下层是服务提供者，提供服务原语操作，上层是服务调用者，调用服务原语操作。下层向上层提供的服务分为两大类：面向连接的服务（Connect-Circuit Service）和无连接服务（Connectless Service）。

面向连接服务是电话系统服务模式的抽象，每一次完整的数据传输都必须经过建立连接、使用连接、终止连接 3 个过程。其特点是收发数据不但顺序一致而且内容相同。

无连接服务是邮政系统服务的抽象，其中每个分组都携带完整的信宿地址，各分组在系统中独立传送。

一般地讲，服务可分为 6 种不同的类型，如表1.1所示。

表1.1 6 种服务类型

服务类型	服务对象
可靠的字节流	远程登录
可靠的报文流	文件传输
不可靠连接	数字话音
不可靠的数据报	电子邮件
有确认的数据报	挂号信
请求-应答	网络数据库查询

1.1.2 UNIX系统网络连接过程

上面讲述了网络连接的一些基本概念，下面将就UNIX系统的具体网络连接过程加以论述。

1. 网络选择

为使网络应用系统能够移植到不同的环境中，在任何环境中其应用程序必须给可用的各种网络提供一个标准界面。网络服务提供了一个简单而又协调的界面，允许用户选择网络以使应用程序具有协议和媒体的独立性。UNIX联网服务系统允许用户利用这个标准界面参与网络的选择。

实际应用中，用户可以用菜单系统或键入的Shell命令来执行与网络服务有关的任务。在菜单系统中，当用户选择某个选项时，相应的子菜单和指令将引导用户进入相应的处理过程。用户也可以不用菜单系统而直接在Shell解释程序下键入命令，这些键入的命令对敏感的系统文件进行编辑，在编辑前必须注意为每个要编辑的文件作一个备份保存下来，当用户完成了对这个文件的编辑时，可以用命令确认一下已经完成了用户所希望的改变。

网络选择的内容可以是：

① 名为Netconfig的数据库文件，这个数据库包含了若干项，这些项用来登记UNIX系统可以使用的各种网络，这个文件存放在/etc目录下。

② 任选的NETPATH环境变量，这个变量是由用户或者系统管理员建立的含有网络标识符的有序表，网络标识符要和Netconfig文件中的Netword Id字段匹配，并用来连接Netconfig文件中的纪录。

具体的网络选择应用编程界面由一组Netconfig数据库访问程序组成。这些库程序中，一部分只访问由NETPATH环境变量标识的Netconfig文件中的项，另外一部分则用于直接访问

Netconfig文件。

任何应用系统都用这些库程序来访问NETPATH，它们允许用户改变所用的传输选择。如果一个应用程序不需要用户进行传输选择，那么用户可以使用直接访问Netconfig数据库的库程序。

网络选择库程序依赖的Netconfig文件是由系统管理员进行维护的。NETPATH环境变量通常是程序员和用户根据自己的需要进行相应的设置和修改的，也可以根据应用的需要由系统管理员来建立。

系统管理员负责维护网络配置数据库文件/etc/netconfig。Netconfig文件由表1.2中的各个字段组成。

表1.2 Netconfig文件中的字段

Network ID	Semantics	Flag	Protocol Family	Protocol Name	Network Device	Directory Library

在文件/usr/include/netconfig.h中定义了和netconfig项有关的全部符号名、结构定义、网络服务特色的常数值，但并非每一个netconfig.h文件都那么全面。

netconfig.h文件是UNIX中关于网络应用的一个重要的数据库文件，其中的各项用于登记UNIX可以使用的各种网络就行了。因为这部分内容技术性比较强，用户在使用过程中可以查阅相关手册。

下面是一个netconfig文件的例子:

```
tcp tpi_cots_ord v inet tcp /dev/tcp /usr/, lib/
tcpip.so,/, usr/ lib/resolv.so
udp tpi_clts v inet udp /dev/udp /usr/lib/ tcpip.so,/ usr/ lib/
resolv.so
ticlts tpi_clts v loopback - /dev/ticlts /usr/lib/,
straddr.so
ticots tpi_cots v loopback - /dev/ticots /usr/lib/,
straddr.so
ticotsord tpi_cots_ord v loopback - /dev/ticotsord /usr/lib/,
straddr.so
icmp tpi_raw - inet icmp /dev/icmp /usr/ lib/ tcpip. so
,/, usr/, lib/ resolv .so
igmp tpi_raw - inet igmp /dev/igmp /usr/lib/tcpip.so,/,
usr/ lib/resolv.so
rawip tpi_raw - inet - /dev/rawip /usr/lib/tcpip.so,/,
usr/, lib/resolv.so
tcp6 tpi_cots_ord v inet6 tcp /dev/tcpipv6 /usr/lib/ tcpip.
so, / usr/lib/resolv.so
udp6 tpi_clts v inet6 udp /dev/udpipv6 /usr/lib/, tcpip.
so,/ usr/ lib/resolv.so
icmpv6 tpi_raw - inet6 icmpv6 /dev/icmpv6 /usr/lib/, tcpip.so,/,
usr/, lib/resolv.so
icmpv6 tpi_raw - inet6 icmpv6 /dev/icmpv6 /usr/lib/, tcpip.so,/,
usr/lib/resolv.so
rawipv6 tpi_raw - inet6 - /dev/rawipv6 /usr/lib/tcpip., so,/,
```

```
usr/, lib/resolv.so
#
#       The Network Configuration File.
#
# Each entry is of the form:
#
# network_id semantics flags protofamily protoname device nametoaddr_libs
#
nbclts  tpi_clts        -       nbcots  -       /dev/nbclts     /usr/lib/, straddr.so
nbcots  tpi_cots        -       nbcots  -       /dev/nbcots     /usr/lib/, straddr.so
```

网络选择是不醒目的。在大多数情况下，用户对通过哪一个网络来完成这项网络操作不感兴趣，系统管理员在Netconfig文件中建立了缺省的网络搜索路径，用于确定可连接的网络。但是，如果用户和系统管理员要修改由应用系统本身作出的选择，那么就要用一个新的标准Shell变量NETPATH来修改搜索路径。就这一点来说，NETPATH变量和PATH变量是类似的。

NETPATH 是由一些冒号分隔开的网络ID表组成。每个网络ID和Netconfig数据库文件中记录的Netword ID相对应。其中的冒号要用反斜杠“\”加入，如果原文件中已包含有反斜杠，用“\\”加入。由于网络ID不允许有空字符串，所以在NETPATH中用一个开始冒号、结束冒号或者两个相连的冒号表示空字符串都不是有效的项。

NETPATH环境变量不是在/etc/profile中设置的，它可以在用户的/home/profile中设置。

这和直接访问Netconfig文件的库程序不同，也不同于通过NETPATH环境变量访问Netconfig文件的库程序。对于直接访问文件的库程序来说，缺省网络的设置是整个Netconfig文件。

2. 建立名称到地址的映射

名称到地址的映射用来允许用户程序以和传输无关的方式在特定的计算机上得到服务的地址。

名称到地址映射是由应用程序使用的库程序组成的。在给定的计算机上使用这些库程序获得服务的地址。所有的库程序组合为一个库，这个库对应一个传输提供器。特定传输提供器使用的库应该在相应的网络中的Netconfig文件中定义。

Netdir-getbyname库程序对/etc/netconfig文件中Directory Library字段指定的库进行动态连接。

这些库程序有:

```
netdir-getbyname
netdir-getbyaddr
netdir-free
netdir-mergeaddr
netdir-options
taddr2uaddr
uaddr3taddr
```

下面的两个库提供了上述所有的库程序:

*tcpip.so包含处理TCP/IP协议的名称到地址映射库程序。

*straddr.so包含处理接收字符协议的名称到地址的映射库程序。

建立和维护每个库文件是系统管理员的任务。在上面这两个库中，tcpip.so这个动态库中的库程序可以从TCP/IP软件包得到/etc/hosts和/etc/service文件来建立地址。在/etc/hosts文件中包含两个地址：计算机的IP地址和计算机名称，下面是/etc/hosts的一个例子：

```
#ident  "@(#)hosts      1.2"
#ident "$Header: /sms/sinixV5.4es/rcs/s19-full/usr/src/cmd/cmd-inet/etc/hosts,v
1.1 91/02/28 16:30:32 ccs Exp $"
#
# Internet host table
#
127.0.0.1       localhost
203.0.1.2       rfs
203.0.1.8       rfs01
203.0.1.9       rfs02
203.0.1.138     yeguozhuang
~
```

在/etc/services文件中含有两个字段，分别表示服务名和端口数。端口数带有TCP或者UDP两个协议之一的说明。下面是一个/etc/services文件的例子：

```
#ident  "@(#)services   1.9"
#ident "$Header$"

#
#       assigned numbers from rfc1060
#
tcpmux          1/tcp
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
systat          11/udp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
netstat         15/udp
qotd            17/tcp          quote
qotd            17/udp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
ftp             21/tcp
telnet23/tcp
smtp            25/tcp          mail
time            37/tcp          timserver
time            37/udp          timserver
name            42/tcp          nameserver
name            42/udp          nameserver
whois           43/tcp          nicname         # usually to sri-nic
```

```
whois           43/udp          nicname         # usually to sri-nic
nameserver      53/udp          domain
nameserver      53/tcp          domain
apts            57/tcp                          #any private terminal service
apfs            59/tcp                          #any private file service
bootps          67/udp          bootp
bootpc          68/udp
tftp            69/udp
rje             77/tcp          netrjs          #any private rje
finger          79/tcp
http            80/tcp
link            87/tcp          ttylink
supdup          95/tcp
hostnames       101/tcp         hostname        # usually to sri-nic
iso-tsap        102/tcp
iso-tsap        102/tcp
x400            103/tcp                         # ISO Mail
x400-snd        104/tcp
csnet-ns        105/tcp                         #CSNET Name Service
pop-2           109/tcp                         # Post Office
pop-3           110/tcp                         # Post Office
sunrpc          111/udp         rpcbind
sunrpc          111/tcp         rpcbind
auth            113/tcp         authentication
sftp            115/tcp
uucp-path       117/tcp
nntp            119/tcp         usenet readnews untp # Network News Transfer
erpc            121/udp
ntp             123/tcp                         # Network Time Protocol
ntp             123/udp                         # Network Time Protocol
nb-ns           137/udp         nbns netbios-nameservice
nb-ns           137/tcp         nbns netbios-nameservice
nb-dgm          138/udp         nbdgm netbios-datagram
nb-dgm          138/tcp         nbdgm netbios-datagram
nb-ssn          139/tcp         nbssn netbios-session
imap-4          143/tcp
NeWS            144/tcp         news            # Window System
iso-tp0         146/tcp
iso-ip          147/tcp
bftp            152/tcp
snmp            161/udp
snmp-trap       162/udp
cmip-manage     163/tcp
cmip-agent      164/tcp
print-srv       170/tcp
bgp             179/tcp
smux            199/tcp
#
# UNIX specific services
#
```