防火墙

——网络安全解决方案

杨辉 吴昊 编著

国防工业出版社

火墙

网络安全解决方案

杨辉 吴昊 编著

国防工业出版社

·北京·

图书在版编目(CIP)数据

防火墙:网络安全解决方案/杨辉,吴昊编著.—北京:国防工业出版社,2001.9
ISBN 7-118-02539-9

Ⅰ.防… Ⅱ①杨…②吴… Ⅲ.计算机网络—安全技术 Ⅳ.TP393.08
中国版本图书馆 CIP 数据核字(2001)第 23189 号

国防工业出版社出版发行
(北京市海淀区紫竹院南路 23 号)
(邮政编码 100044)
北京奥隆印刷厂印刷
新华书店经售
*
开本 787×1092 1/16 印张 11¾ 259 千字
2001 年 9 月第 1 版 2001 年 9 月北京第 1 次印刷
印数:1—4000 册 定价:16.00 元

前言

互联网的爆炸式发展出乎人们的想象，新技术、新概念层出不穷，互联网实现了人们在信息时代的梦想，预示着新经济时代的到来。随着全球的网络化，网络的全球化，世界缩小了，人类的工作、生活变得更加快捷、方便。网络应用在经济、军事、科技、教育等各个领域。电子商务、远程教育、网上医疗渐渐步入千家万户，网络吸引了亿万用户，很多人已经离不开互联网，它已经成为人们生活的一部分。

伴随网络的发展，网络与信息的安全问题也越来越受到人们的重视。防火墙技术是目前各种网络安全解决方案中常用的技术。防火墙和网络认证设备同是主机系统和应用系统的安全屏障，它们能有效地将大多数入侵阻挡在内部网络之外。

本书分为两大部分，第一篇的第一章简要介绍了与互联网紧密相关的 TCP/IP 协议，第二章介绍了目前互联网上普及的各种应用和服务，描述了网络上存在的漏洞和危险。接下来的第三到第五章讲解防火墙技术以及防火墙的配置方法，包括防火墙体系结构、堡垒主机、数据包过滤、代理系统等。在第二篇中，我们通过收集整理的国内外市场上主要防火墙产品的技术资料，对七种防火墙产品进行了功能、性能、特色等方面的比较和分析。本书的最后列出了二十多种市场上常见防火墙产品的相关信息，以期对读者购买和使用防火墙产品有些实际的帮助。

周灵、赵洪彪、张佃、谢振明、李伟斌、鲁海军、李俊华、熊克奇、朱刚、邓廷勋等同志参与了书稿的编排和整理工作，作者对他们给予本书的贡献表示深深的感谢。作者也非常感谢所有在本书的编写过程中给予作者以鼓励和帮助的人们。

对于本书中存在的错误和不足之处，作者殷切希望广大读者批评指正。

作　者

2000 年 9 月于北京

第一篇　防火墙的理论和技术

第一章　TCP/IP 简介

在介绍防火墙之前，我们必须了解一下 TCP/IP 协议。它是了解防火墙原理以及后面一些问题的基础。

TCP(Transmission Control Protocol)/IP(Internet Protocol)协议是由美国国防部高级研究计划局 DARPA(Defense Advanced Research Project Agency)在 20 世纪 70 年代研究创立的。TCP/IP 不是一个单个的协议，它是一个包括 TELNET、FTP、ICMP 在内的一个协议组。

1.1　OSI 模型

协议分层模型包括两方面的内容，一个是层次结构，另一个是各层功能的描述。OSI 模型包括七层，如图 1-1 所示。

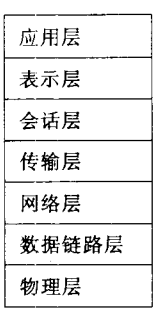

图 1-1　协议分层模型

应用层：提供最常用的且通用的应用程序，例如：电子邮件、文件传输等等。

表示层：提供信息转换操作，包括信息的压缩解压、加密解密、信息标准格式的转换等等。

会话层：主要针对远程终端访问。包括会话管理、传输同步以及活动管理等操作。

传输层：主要的目的在于弥补网络层服务与用户需求之间的差距。传输层通过向上提供一个标准的通用的界面，使上层与下三层的细节相隔离。传输层的主要任务是提供进程间通信机制和保证数据传输的可靠性。

网络层：负责将数据从物理连接的一端传输到另一端，主要功能有寻径、相关的流量控制和拥塞控制等等。

数据链路层：包括介质访问控制(MAC)和逻辑链路控制(LLC)两个子层。MAC 子层解决广播型网络中多用户竞争信道使用权的问题，LLC 将有噪声的物理信道变为

无传输差错的通信信道，提供数据成帧、差错控制、流量控制和链路控制等功能。物理层：涉及在物理信道上的数据传输和处理与传输物理介质有关的机械的、电气的过程的接口。

1.2 无连接的协议和面向连接的协议

无连接的协议和面向连接的协议是根据它们的操作是无连接的还是面向连接的来区分的。无连接的操作和面向连接的操作的特征如下：

1. 无连接的操作

数据传输之前，在用户和网络之间没有逻辑上的连接。数据单元被当作独立单元传输。无连接的服务把用户 PDU(协议数据单元)作为独立的、不相关的实体来管理。在成功的数据传送之间不保持任何关系，并且几乎不保留执行用户到用户通信进程的记录。如果数据在传输中出现错误，无连接的协议也不会将数据重新传输。

2. 面向连接的操作

在数据传送之前，用户和网络要先建立一个逻辑上的连接，通常，在通过用户/网络连接传送的数据单元之间要保持某种关系。面向连接的服务要求在两个末端用户和服务提供者(例如网络)之间有一个三向协议，也允许通信部分议定合适的选项和服务质量功能，在连接建立期间，所有三部分互相存储信息。如果在传输过程中出现问题，面向连接的协议就为错误的单元的重新传输提供机制。

1.3 IP 协 议

IP 是无连接的协议，位于 OSI 模型的网络层。它允许两个主机计算机在没有任何预先调用设置的情况下交换信息。下面讲述一些 IP 的基本概念。

1.3.1 物理地址(MAC 地址)

通信链路或网络上的每台设备都是由一个物理地址来标识的，这个地址通常称为硬件地址。许多厂家把物理地址放置在设备内或直接与该设备相连的接口单元内的逻辑板上。在通信时要用到发送方和接收方两个物理地址。物理地址的长度是可变的。Internet 用的是 48 位地址，这种地址也叫做 MAC 地址。

1.3.2 IP 地址

就像每个人有不同的身份证号一样，Internet 上的每台计算机都必须有一个不同于其他主机的代号，以保证数据的正确发送和接收，这就是 IP 地址。虽然用物理地址(MAC 地址)同样可以标识每台主机，但是由于它没什么规律和难以记忆，加上有些代理和拨号上网的许多用户用的 MAC 地址都是 ISP 服务器的 MAC 地址，使用起来难以区别。相比之下 IP 地址更方便和简洁。

1.3.3 IP 地址的分类

TCP/IP 网络用 32 位地址来标识主机和与主机相连的网络。IP 地址的格式是：

IP 地址 = 网络地址 + 主机地址

网络地址标识了一个网络。主机地址则是主机在该网络中的一个编号。IP 地址不仅标识主机，还标识了主机和它的网络的连接。如果主机移到另一个网络中，那么它的地址空间必须要改变。

IP 地址按照它们的格式可以分为四类：A 类、B 类、C 类和 D 类，如图 1-2 所示。

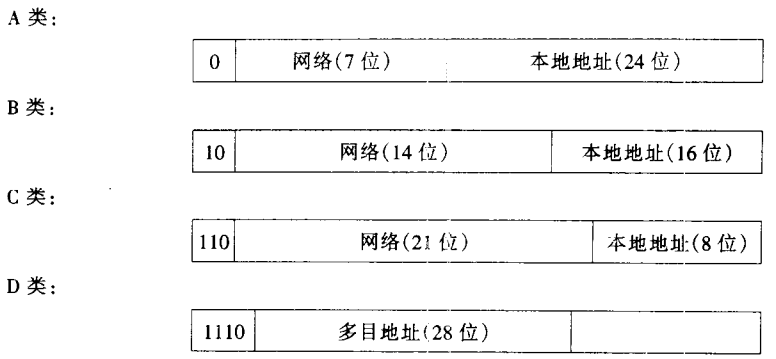

图 1-2　IP 地址的格式

1.3.4　IP 数据包格式

IP 数据包由包头和数据体两部分组成。包头由 20 字节的固定部分和变长的可选项组成。IP 数据包的格式如图 1-3 所示。

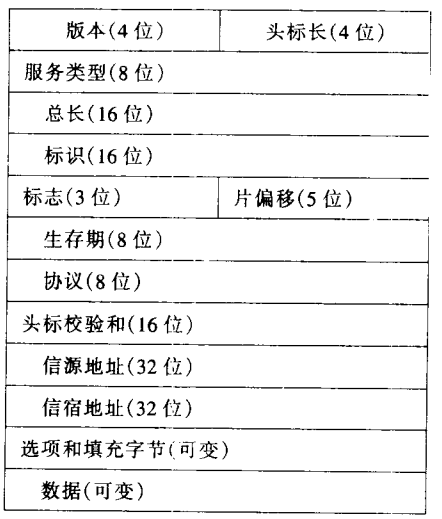

图 1-3　IP 数据包格式

各部分的含义如下：

版本：标识 IP 的版本。

头标长：标识 IP 头标的全部长度，包括 20 字节的固定部分和可变部分。

服务类型：指明需要协议栈的低层提供什么样的服务，不同的服务有着不同的可靠性和传输速率。

总长：包括头标和数据在内的总长度。

标识：和地址区一起用于唯一的标识数据单元。(为分片而设)

标志：在分片操作中使用。

片偏移：描述该数据包位于原始 PDU 的何处。

生存期：用于决定数据包在网际中生存多久，IP 包每经过一个中间主机，都要减 1，

当减到0时，就丢弃这个包。

协议：用来确定最后信宿中要接收用户数据的高一层协议，如：TCP或UDP。

头标校验和：用于在头标中进行容错校验。

信源和信宿地址：标识信源和信宿主机以及和它们直接相连的网络。

选项和填充字节：用于给数据包32位的整数倍的字节。

数据：容纳用户数据。

1.3.5 子网和网关

Internet用网关或路由器来描述一台完成网络间信息转发的机器，如图1-4所示。

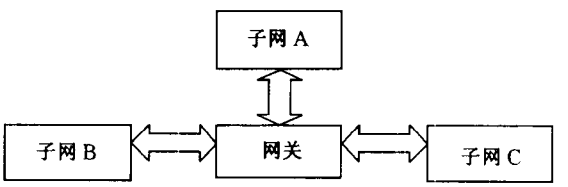

图1-4 子网和网关

子网是一个完备的逻辑网络，帮助实现了互联的整个操作。Internet实际上是由若干个子网构成的。网关用来协调和完成各个子网间的通信。对末端用户来说，互联网关是透明的。实际上末端用户应用程序一般是驻留在和网络相连的主机中而很少放在网关中的。网关的主要目的是接收包含正确寻址信息的PDU，使得它可以由该PDU到达下一个网关或最后的信宿，网关对应用层是透明的。

1.3.6 子网掩码

子网掩码的目的是确定IP地址的哪个部分与子网有关，哪个部分与主机有关。掩码有32个二进制位，它的内容设置如下：

1：标识IP地址的网络地址部分

0：标识IP地址的主机地址部分

例如：IP地址212.106.96.82，子网掩码255.255.255.224；它的网络地址部分是11010100 01101010 01100000 010，主机地址部分是10010。

1.4 TCP和UDP

TCP及传输控制协议对应于OSI模型的传输层。最初TCP是为了ARPANET和Internet而开发的，但如今已在世界各地广泛使用。UDP(用户数据包协议)在某些应用中，常用它来代替TCP。解释协议的最好方式就是看看它的报文结构了。下面讲述了关于这两种协议的报文结构。

1.4.1 TCP段

TCP和UDP都是位于IP之上的上层协议。在OSI模型中，数据经过每一层，由每一层相应的协议对数据进行封装和解读。TCP和UDP的协议和数据内容都在IP数据包的数据部分。

两个TCP模块中相互交换的PDU叫做段，段分为两部分：头标和数据。数据跟在

头标后面。如图 1-5 所示,说明了 TCP 段的格式。

<table>
<tr><td colspan="9">信源端口(16 位)</td><td colspan="3">信宿端口(16 位)</td></tr>
<tr><td colspan="12">序号(32 位)</td></tr>
<tr><td colspan="12">确认号(32 位)</td></tr>
<tr><td>数据
偏移
(4 位)</td><td colspan="2">保留区
(6 位)</td><td>U
R
G</td><td>A
C
K</td><td>P
S
H</td><td>R
S
T</td><td>S
Y
N</td><td>F
I
N</td><td colspan="3">窗口(16 位)</td></tr>
<tr><td colspan="9">校验和(16 位)</td><td colspan="3">紧急指针(16 位)</td></tr>
<tr><td colspan="9">数据(变量)</td><td colspan="3">填充</td></tr>
<tr><td colspan="12">选项(变量)</td></tr>
</table>

图 1-5　TCP 段的格式

信源端口和信宿端口:用于标识使用 TCP 连接的上层应用程序。

序号:用于其后用户数据编号的初试发送序列。

确认号:置为确认以前收到数据的值。该区中的值包含从传输机发出的下一待接收字节的序号值。

数据偏移:规定构成 TCP 头标的以 32 位为单位的字数。用于确定数据区从哪里开始。

保留区:保留以备将来使用。

URG:该位表示紧急指针是否重要。

ACK:该位表示确认区是否重要。

PSH:该位表示本模块将实行推挽功能。

RST:该位指示连接将被复位。

SYN:该位指示序号将被同步;在连接建立段中用它作为标志,指出将要发生握手操作。

FIN:该位指示发送机器再也没有数据要发送了。

窗口:指示接收机希望接收多少字节。

校验和:用于在头标和数据中进行容错校验。

紧急指针:仅在 URG 被置位时才使用,指示那些包含有紧急数据的数据字节在数据流的何处开始,何处结束。

数据:TCP 协议中包含的用户数据。

填充:用于保证 TCP 头标是由 32 位的偶数倍填充的。

选项:用于适应 TCP 未来的发展,每个选项由一个包含选项号的单字节,一个包含选项长度的区及其本身的选项值组成。

1.4.2　UDP 段

UDP 是面向连接的,但它并不使用面向连接协议中常用的扩展状态管理操作。在不需要 TCP 全部服务的情况下,有时用 UDP 来代替 TCP。如图 1-6 所示,显示了 UDP 数据包的格式。

<table>
<tr><td>信源端口(16 位)</td><td>信宿端口(16 位)</td></tr>
<tr><td>长度(16 位)</td><td>校验和(16 位)</td></tr>
<tr><td colspan="2">数据(可变)</td></tr>
</table>

图 1-6　UDP 数据包的格式

第二章　Internet 上的常见服务

Internet 即全球计算机互联网络，是全球信息高速公路的雏形。Internet 使用世界上先进的网络技术，蕴含极丰富的信息资源，向用户提供了广泛的信息服务，是目前世界上分布最广的计算机网络。Internet 网络提供了许多非常有用的服务，这里只简略的介绍几种常见的服务。

2.1　域名系统(DNS)

为了讲解后面内容的方便，让我们先了解一下域名系统(DNS)。

Internet 协议(IP)地址结构记忆起来还是不够方便。实际上，绝大多数组织是采用缩写词或有意义的名字来标识数字地址的。那么采用缩写词的网络用户是怎么与使用 IP 地址的 Internet 实现互联的呢?

最早的名字的组织和管理是由 SRI 网络信息中心(NIC)完成的。它维护一个称为 HOST.TXT 的文件，该文件列出了网络、网关和主机名字以及它们对应的 IP 地址。最先采用的是无层次名字(flat name)空间，无层次名字的描述形式仅由标识对象的字符组成，没有进一步的含义或结构。这在 Internet 的早期工作得很好，但是随着 Internet 的增长，HOST.TXT 文件的管理日益成为一项庞大的工作。因此出现了一个称为域名系统的解决方案。

2.1.1　域名的树型层次结构

DNS 采用分层次的树状方案创建名字。层次结构和公司里的组织机构差不多，最高层位于树型层次结构的顶部，其下属位于下面一层的分枝上，依此类推。相同层次的各个节点标号必须是完全清楚和互不相同的。也就是说，标号必须是相互可以区别的名字，在同一层次上彼此可以区分开。图 2-1 显示了域名的这种树型层次结构。

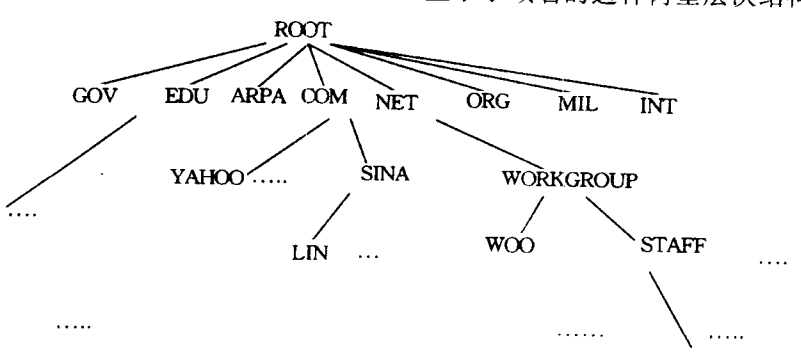

图 2-1　域名的树型层次结构

2.1.2 域名

沿着树型结构下行，直到主机，就可以得到主机的域名。例如：在图 2-1 中，主机 WOO 的域名就是 WOO.WORKGROUP.NET。

DNS 提供两种不同的最高层域名分级方式：地理的和组织的。按地理划分就是把全世界的机器按国家（或地区）来划分。例如：美国的所有机器的最高一级的域为 US，中国所有机器的最高一级的域为 CN。按组织划分就是全世界的机器按照所属组织的类型来划分。例如：163.NET，Yahoo.COM。表 2-1 列出了现在使用的几个顶级域名。

表 2-1 域名的划分

域　名	含　义
COM	商业组织
EDU	教育机构
GOV	政府机构
MIL	军队
NET	主要网络支持中心
ORG	与上面不同的其他组织
ARPA	临时的 ARPANET 域（已陈旧）
INT	国际组织
国家（地区）代码	每个国家或地区（按地理划分）
说明：域名系统的比较是与大小写无关的，上述域名的小写也具有同样含义。	

名字的语法不能决定它命名对象的类型或协议族的种类。也就是说，名字中的标号的数目并不能判断名字是指向一个单个对象（机器）还是一个域。例如前面的树型结构中 WOO.WORKGROUP.NET 是台主机，而 STAFF.WORKGROUP.NET 却是一个子域的名字。因此，一个给定的名字可能映射到域名系统的多个项目。在解析名字的时候必须指定所需对象的类型，并由服务器返回那种类型的对象。

2.1.3 名字服务器

名字服务器是提供名字到地址转换的服务器程序。通常，服务器程序在专用处理器上运行，并把机器本身称为名字服务器。

根据域名的树型层次结构，我们很容易想象名字服务器与它们命名等级对应的树结构。树的根是识别顶层域（对应前面列出的几个顶级域名）的服务器，它知道解析每个域的服务器。给定一个要解析的名字后，根可为该名字选择一个正确的服务器。下一级的服务器都可以为上一层的域提供回答。然而，实际上服务器树没有几层，因为一个物理服务器可能含有大部分命名等级的所有信息。

2.1.4 域名解析

用户在解析域名的时候所要做的事情是相当简单的。用户只需要给称为名字解析器的本地机构（也就是我们常说的域名服务器）提供一系列询问和如何操作的请求，由本地机构根据域名取回信息或把请求发送给名字服务器。

域名服务器承担解析名字和为名字/地址解析把名字发送到独立的合作系统中的

任务。图 2-2 显示了域名解析的操作。

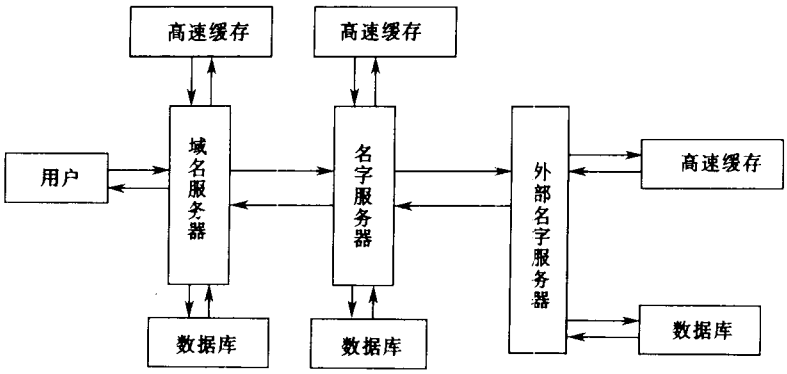

图 2-2 域名解析

用户向域名服务器提出域名解析请求。服务器先检查高速缓存中是否有相应的信息,如果没有,它就查询它本地的数据库。如果还找不到,它就到上层名字服务器中查询。如果上层名字服务器在它的高速缓存和数据库中都找不到,它就向外部名字服务器查询。找到后,服务器把信息返回给提出查询请求的服务器或主机,更新高速缓存。如果找不到它就返回失败的信息。

域名服务器和名字服务器都可以修改它的数据库信息来更新 Internet 上不断增加和变化的主机域名。高速缓存通常是不完整的,它只能提供询问最频繁的信息,加速域名解析的过程。

有两种类型的 DNS 网络活动:查询和区域传输。查询用于一个 DNS 客户机向一个 DNS 服务器查询信息,如:某个主机的 IP 地址;区域传输是在一个 DNS 服务器(第二个服务器)向另一个 DNS 服务器(主服务器)查询时使用,可查询主服务器所知的有关某个 DNS 树的任何信息。区域传输只在服务器之间使用,并假定它们提供同样的信息。在实施时常用 UDP 来执行 DNS 查询,用 TCP 来实现区域传输。DNS 服务器使用众所周知的端口 53 来做 UDP 的所有工作,同时该端口也作为它的 TCP 服务器端口,TCP 请求使用大于 1023 的随机端口。DNS 客户机使用大于 1023 的随机端口做 UDP 和 TCP 的端口。

2.1.5 DNS 数据

DNS 是一个树形结构的数据库,各种各样的子树服务器分布在 Internet 上。在该树上有许多定义好的记录类型,见表 2-2。

表 2-2 DNS 中的记录类型

记录类型	用 途
A	把主机名翻译成 IP 地址
PTR	把 IP 地址翻译成主机名
CNAME	把主机别名翻译成主机名
HINFO	提供主机的软/硬件信息
NS	把 DNS 树授权给其他某个服务器
SOA	指示 DNS 树授权开始
TXT	非结构化文本记录

事实上有两个不同的 DNS 数据树，一个是通过主机名获得信息（如：IP 地址、CNAME 记录、HINFO 记录或 TXT 记录，这些记录对应某个给定的主机）。下面的例子对此作了说明。假设域名为 workgroup.net，“IN”表示在某个记录中的内容：

```
workgroup.net IN SOA woo.workgroup.net. root.woo.workgroup.net.
    (
      2001      ;序列号
      36000     ;刷新频率(10 小时)
      1800      ;重试(半小时)
      3600000   ;过期(1000 小时)
      36000     ;默认 TTL(10 小时)
      )
        IN NS       woo.workgroup.net.
        IN NS       yang.workgroup.net.
woo     IN A        202.196.168.32
        IN MX       5 woo.workgroup.net..
        IN MX       8 yang.workgroup.net.
yang    IN A        202.196.168.33
        IN MX       5 woo.workgroup.net..
        IN MX       8 yang.workgroup.net.
ftp     IN CNAME    woo.workgroup.net.
www     IN CNAME    yang.workgroup.net.
cjh     IN A        202.196.168.35
```

另一个就是 PTR 记录集，用来把 IP 地址映射到主机名。要把 IP 地址翻译成主机名，需要把 IP 地址组成部分倒转过来，然后附加上 .IN-ADDR.ARPA，并查找该名字的 DNS 记录。例如：要翻译 IP 地址 2.3.4.5，就应该查找 5.4.3.2.IN-ADDR.APRA 的 PTR 记录。

```
168.196.202.IN-ADDR. ARPA IN SOA woo. workgroup.net. root. woo. work-
group. net.
    (
      2001      ;序列号
      36000     ;刷新频率(10 小时)
      1800      ;重试(半小时)
      3600000   ;过期(1000 小时)
      36000     ;默认 TTL(10 小时)
      )
          IN NS     woo.workgroup.net.
          IN NS     yang.workgroup.net.
    32    IN PTR    woo.workgroup.net.
    33    IN PTR    yang.workgroup.net
    35    IN PTR    cjh.workgroup.net.
```

2.2 电子邮件

电子邮件提供了快速、方便的信息传送方式，是最流行和最基本的网络服务之一。它是以可读方式为传输较小的文件而设计的。

2.2.1 电子邮件的地址

邮件发送者通过电子邮件地址将邮件准确的发给邮件接收者。通常电子邮件地址由用户的登录名和目的邮件服务器的机器的域名组成，中间用@分开。例如：邮件地址：ABC@MailServer.edu.cn，用户登录名是 ABC，目的邮件服务器的域名是 MailServer.edu.cn。

电子邮件的地址中还可以使用别名。由于域名系统包含一个单独的对邮件目的地址的查询类型，邮件的目的地址名字可以与分配给机器的通常域名没有联系。例如上面的地址：ABC@MailServer.edu.cn，发给 MailServer.edu.cn 的邮件可能送到了其他机器上，比如：Workgroup.com。

2.2.2 电子邮件的格式

电子邮件分为两部分：信头和正文，中间用一个空行隔开。邮件报文的 TCP/IP 标准指明了邮件信头的准确格式以及每个信头字段的语义解释，但它将正文格式的定义留给了发送者。标准指明了报文信头包含可读文本，可读文本被分为若干行，每行由一个关键字后跟冒号和一个值组成。一些关键字是必须的，一些是可选的，其余的是不可解释的。

例如：一封信，信头包含一行文本指明接收者的邮件地址（TO：接收者的邮件地址）一行包含发送者的邮件地址（FROM：发送者的邮件地址）。

表 2-3 列出了一些常用的关键字。

表 2-3 电子邮件中的常用关键字

关 键 字	说 明
To	指明接收者的邮件地址
From	指明发送者的邮件地址
Message-Id	邮件消息的标识号
Date	接收邮件的时间
Status	标识邮件的状态（已读，未读）
Content-Length	邮件消息的长度
Reply-to	发送者指明的回复地址

选择邮件报文格式是为了使异构机器之间处理和传输变得容易。保持邮件信头的直观使得它可用于大范围的系统。限制报文为可读文本避免了选择一种标准二进制表示以及在标准表示和本地表示间转换的问题。

为了能通过电子邮件发送非 ASCII 数据，IETF 定义了多用途邮件扩充 MIME（Multipurpose Internet Mail Extension）。MIME 允许用 ASCII 码对任意数据编码，然后在标准邮件报文中传送。这样我们就可以在电子邮件中发送图像、声音等多种数据。

为了适应于任意数据类型和表示，每个 MIME 报文包含告知接收者数据类型和使用编码的信息。MIME 的信息位于邮件信头中，MIME 的各行指明使用 MIME 的版本、发送数据的类型以及将数据转换为 ASCII 所使用的编码。下图显示了一个 MIME 报文，它包含标准 GIF 表示的一幅图片。GIF 图像被 base64 编码转换为 7 位 ASCII 码表示。

```
From: Yang@chinacollege.edu.cn
To: woo@sina.com.cn
MIME-Version: 1.0
Content-Type: image/gif
Content-Transfer-Encoding: base64
..... ....
```

信头中的“MIME-Version:”说明了该邮件使用 MIME 协议版本 1.0 编制报文；“Content-Type:”指明数据是一个 GIF 图像；“Content-Transfer-Encoding:”说明使用 base64 编码将图像转换为 ASCII 码。因此，为查看图像，接收者必须先将 base64 编码转换成二进制，然后用应用程序在用户屏幕上显示 GIF 图像。

MIME 标准规定 Content-Type 说明必须含有两个标识符：内容类型和子类型，中间用“/”分开。在本例中，image 是内容类型，gif 是子类型。

标准定义了七个基本内容类型，每个类型的子类型和传递编码。例如，image 类型的子类型必须为 jpeg 或 gif，text 类型却不能用这种子类型。表 2-4 列出了七种基本的内容类型。

另外，除了标准类型和子类型，MIME 还允许发送者和接收者定义专用的内容类型。专用的内容类型的名字要以字符串 X-开始。

表 2-4　MIME 中的基本内容类型

内容类型	对应的数据
文本	文本的(比如：文档)
图像	一幅静止图片
声音	一段声音
视像	视像记录(包括动画)
应用程序	程序的原始数据
多部分(包括：混合、替代、并行、文摘四种子类型)	多种报文，每个都有一个单独的内容类型和编码
报文	一个完整的电子邮件报文，或对一个报文的外部引用

2.2.3　电子邮件的系统组成

电子邮件系统通常由三部分组成：一个是用来向外部主机发送邮件和从外部主机接收邮件的服务器部分，例如一般的邮件服务器；一个是将邮件发送到服务器的发信代理部分；一个是用来阅读邮件和编辑发出邮件的用户代理部分，这三个部分可以由同一程序或不同程序组合来实现。例如：OUTLOOK EXPRESS 就是发信代理和用户代理的结合。

大多数系统提供邮件转发软件，它包括一个邮件别名扩展机制。邮件转发者允许本地服务器将邮件地址中使用的标识符映射为一个或多个新的邮件地址。通常，在用

户写完一个报文，并给接收者命名以后，邮件接口程序将查询本地别名，用映射的版本代替接收者的标识，然后再由传递系统将报文发出。

2.2.4 电子邮件的工作原理，SMTP，POP

Internet 上的邮件交换是通过 SMTP(简单邮件传输协议)来处理的。

SMTP 服务器接收邮件并检查邮件的目的地址，决定将邮件在本地范围发送(给同一服务器上的用户)还是发送到外部服务器上(给不同服务器上的用户)。如果是本地范围发送，它就为本地发送程序重新以合适的方式对信头和地址进行编码，然后将邮件交给本地发送程序；如果是发送到外部服务器，它就修改信头，并与外部服务器联系，或通过其他机器转发，然后发出邮件。

为处理延迟的传输，邮件系统使用缓冲池技术。假设要给一个外部邮件服务器的用户发送邮件。在本地服务器已连接的情况下，发送者在继续工作之前不必等待外部服务器可用，也不会仅仅由于暂时中断与外部服务器的连接就放弃传输。当用户发送一个邮件消息时，本地服务器将邮件副本与发送者、目的机器的标识以及存放时间一起放入私有存储区，再以后台方式启动本地发送程序。本地发送程序先通过域名解析将外部服务器或执行转发的主机映射成 IP 地址，随后建立到外部服务器或转发主机的 TCP 连接。如果成功，则将一份报文副本传递给外部服务器或转发主机。然后外部服务器或转发主机将此副本保存在自己的缓存区内。在邮件传送进程和接受的主机都认可已收到和存储副本之后，邮件传送进程就删除本地副本。如果由于不能建立 TCP 连接或连接失败，邮件传送进程记录下尝试传递和传递终止的时间。后台传送进程定期对整个缓存区扫描，检查是否有未传递的邮件，如果找到未传递的邮件或新的待发邮件，后台传送进程将尝试发送它们。如果邮件传送进程发现某个邮件消息过期还没传递到目的地，它就将此消息返回给发送者并报告传送失败。

假设一个 163.net 邮件服务器上的邮件用户 Woo，要给 sina.com.cn 上的用户 Lin 发送一封邮件。用户 Woo 先将邮件发送到服务器 163.net，邮件的目的地址是 Lin@sina.com.cn，服务器 163.net 发现 sina.com.cn 不是本地服务器，也不是它的别名。于是它通过域名解析，得到 sina.com.cn 的 IP 地址。然后它与 sina.com.cn 建立 TCP 连接，如果成功，则 163.net 将用户 Woo 的邮件发送给邮件服务器 sina.com.cn；如果失败，163.net 记录下失败的时间，在下一个发送周期，再试图发送这封邮件。如果邮件消息过期了，它就给用户 Woo 发送一封邮件，告诉他邮件未到达目的地。

那么用户是怎么和本地服务器联系的呢？

下面描述了一次成功的通信。

首先，用户从一个大于 1023 的端口建立和服务器的 SMTP 端口(通常是 25)的 TCP 连接，并等待服务器发送一个 220 READY FOR MAIL 的报文。收到 220 报文后，用户向服务器发送一个 HELO 命令。

一行的结束标志着一个命令的结束。服务器通过标识自己作响应。一旦建立通信，发送者可传送一个或多个邮件报文、终止连接，或请求服务器交换发送者的身份以便报文能反向流动。服务器必须确认每个报文。它也可以异常终止整个连接或当前的报文传送。用户通过向服务器发送 MAIL 命令、发送者的标识符和一个包含接受差错报告的地址的 FROM:字段开始邮件事务。服务器准备接收新邮件的数据结构，并通过发送响应 250 回答 MAIL 命令。

成功执行 MAIL 命令后,用户发出标识邮件报文接收者的一系列 RCPT 命令。服务器通过发送 250 OK 确认每个 RCPT 命令。确认所有 RCPT 命令后,用户发出一个 DATA 命令,服务器用报文 354 回应表示可以开始传送邮件的内容了。然后用户向服务器发送邮件的内容。完毕后,用户向服务器发送"<CR><LF>.<CR><LF>"表示邮件内容的结束。服务器回应一个 250 OK 的报文表示完成。最后用户发出 QUIT 命令,结束传输。

同样,以上面的邮件传输来说明。

Woo:通过域名解析或直接使用 IP 地址和服务器端口 25 建立 TCP 连接

163.net: 220 bjsmtp2.163.net ESMTP

Woo: HELO 163.net <CR> <LF>

163.net: 250-bjsmtp2.163.net 250-PIPELINING 250-SIZE 10240000 250-ETRN 250 8BIT MIME

Woo: MAIL FROM: Woo@163.net <CR> <LF>

163.net: 250 OK

Woo: RCPT TO Lin@sina.com.cn <CR> <LF>

163.net: 250 OK <CR> <LF>

Woo: DATA <CR> <LF>

163.net: 354 End data with <CR> <LF> . <CR> <LF>

Woo: ……发送邮件的内容

Woo: <CR> <LF> . <CR> <LF>

163.net: 250 OK

Woo: QUIT <CR> <LF>

用户 Woo 断开和服务器的连接

163.net: 221 Bye

之后服务器断开和用户的 TCP 连接

说明:上述的传输中只对客户端的命令中的<CR> <LF>加以描述,而对服务器回应信息中的<CR> <LF>没有描述。

那么用户是怎么收取邮件的呢?

通常,用户通过 POP(Post Office Protocol)协议来收取邮件。POP 是处理用户电子邮件信箱的客户/服务器协议。借助 POP,用户信箱保存在服务器上,而不是在用户的个人计算机上。由于服务器是一直开机的,所以用户一般不用担心邮件会丢失。当用户想看邮件的时候,通过自己的客户程序(如 OUTLOOK EXPRESS,或 NETSCAPE MESSAGE)、通过 POP 协议到服务器上收取。

POP 是基于 TCP 的服务。目前使用的版本 POP3 的服务器使用端口 110,旧版本 POP2 的服务器使用的端口是 109。用户的客户端使用大于 1023 的端口。

下面描述一次成功的收取邮件的过程。

用户首先和 POP3 服务器端口 110 建立 TCP 连接。之后服务器返回 OK,表示连接已成功建立。之后用户向服务器发送用户名和口令等认证信息,得到服务器的确认后,通过状态、列表和收取等命令从服务器获得未读的邮件和作删除邮件的操作。

POP3 根据服务器的不同类型有不同的认证指令,对此本文不作详细解释。如果读者有兴趣可查看相关的 RFC 文档。