TurboMail® 邮件系统

Turbomail Enterprise Mail System

管理员基本培训材料

各版本通用

Windows/Linux/Aix/Solaris/FreeBSD 通用

目录

第一章 电子邮件系统基础知识 4

1 Email 系统基本原理 4

1.1 Internet 邮件系统 4

1.2 基本网络协议 14

2.1 RFC821(SMTP) 14

2.2 POP3 协议 17

2.3 IMAP4 协议 24

1.3 电子邮件系统传送过程 35

1.4 电子邮件技术名词解释 36

第二章 TurboMail 电子邮件系统介绍 39

2.1 传统电子邮件系统的局限 39

2.2 大容量电子邮件系统的特点 40

2.2.1 系统结构 40

2.2.2 系统效率 41

2.2.3 对公共协议的支持 42

2.2.4 对平台的支持 42

2.2.5 垃圾信件的处理功能 42

2.2.6 邮件列表功能 43

2.2.7 虚拟主机和虚拟域 43

2.2.8 Web Mail 43
2.2.9 系统的可扩展性 43
2.2.10 系统的可管理性 44
2.2.11 支持 Unified Messaging 44
2.2.12 系统安全性 44
2.2.13 若干流行电子邮件系统技术分析 44
2.3. 电子邮件系统评估准则 45
2.3.1 结构 45
2.3.2 管理 48
2.3.3 服务器管理 49
2.3.4 安全性和防止垃圾邮件 52
2.3.5 灵活性，延伸性和一体性 53
2.3.6 集成商的经验和支持 54

第一章 电子邮件系统基础知识

1 Email 系统基本原理

本课程面向的对象是TurboMail电子邮件系统管理员或技术支持工程师，培训目标是使学员在培训结束时掌握Email系统的若干基本概念、工作原理，让学员最终掌握Email系统的必要基础知识。

不管你想在Internet上做什么，你都必须阅读这一节。这是为什么呢？这是因为每一个计算机在Internet上都有自己唯一的地址，同样，每一个使用Internet的人也都有地址，因此，你必须学习和了解Internet的地址。 在日常生活中，我们需要记住各种类型的地址以便与人通信联络，如邮件地址、独立的街道地址、住宅电话号码、商业电话号码、传真号码等等。 在Internet上只有一种地址即电子地址。当我们知道了某人的Internet地址时，我们就可以向那个人发送电子邮件、传输文件、对话，甚至可以查找关于那个人的其他有关信息。反之，当你开始使用Internet时，为了取得与他人通讯联络，你须提供一个简单的地址。 当某人告诉你在Internet上可以查找一种信息或一种特殊的资源时，他只需向你提供计算机地址就行了，就象在本书的目录上列出的各种条目一样。 因为上述原因，了解Internet编址体系就成了使用Internet的关键。在这一章中，我们将对此作详细说明。

1.1 Internet 邮件系统

在Internet信息资源中，电子邮件是使用最多的。实际上对许多人来讲，一提起邮件就意味着使用Internet。

在本章，我们将详细介绍Internet电子邮件系统，并探讨在初学时必须掌握的常识。首先让我们从基本观念和概念开始：提供Internet邮件服务的程序是什么?有哪些技术词汇和术语?

在这里，通过一些使用电子邮件系统的索引和每个文件的专业资料说明，来把所需理解的问题确定下来。在后面，还将向你讲解怎样使用标准的Unix邮件程序。那时，将给你一个收发文件操作说明。

1.1.1 标准的Internet地址

在Internet上，地址这个词总是指电子地址，而不是通常的邮政地址。如果一个用计算机的人询问你的"地址"，他要的就是你的Internet地址。

Internet地址都按照下列形式：个人用户标识，后面是一个@印刷符号(即"at"符号)，再后面是计算机的名字(在Internet上，每一个计算机都有一个唯一的名字)。下面是一个典型的例子。

Harley@mailer.com.cn

在这种情况中，用户标识是harley，计算机的名字是mailer.com.cn。请注意，就象这个例子表示的那样，在一个地址中绝无任何空格。

你可能还记得，你的Internet入门指导老师已解释过每一个人都有一个被称作用户标识(userid)的用户名称，它被用作地址的首要部分。如果你在使用Unix系统，你的用户名将是登录时的名字。

在@印刷符号后的地址部分被称作domain(域)。在上面的例子中，域是mailer.com.cn。

这样，所有Internet的地址通用形式为：

userid@domain(域名)

象你可以想象的那样，用户标识肯定不是唯一的，例如在整个Internet上，也许就许多人碰巧都有一个绝妙的叫harley的用户名。

什么是唯一所具备的条件呢?那就是用户标识与域名的结合。因此，虽然在Internet上可以有多个harley，但是，在名为mailer.com.cn的计算机上这样的用户标识则只有一个。如果你大声地读一个Internet地址，所使用的@印刷符号就相当于"at"。例如，假如你想发送一个邮件给我们刚刚提到的那个地址的一个人，在Unix上的命令为：

mail harley@mailer.com.cn

当你敲入这个命令时，你就可以对自己说："我正在给harley发送邮件，他在名为mailer.com.cn的计算机上。

有时侯，这种形式的地址又被称作全限定域名或缩写为(FQDN)。

1.1.2 域名和子域

在上一部分中，我们使用harley@mailer.com.cn作为Internet地址的一个范例。在该例子中，我们说harley叫用户标识，而mailer.com.cn为域。域的每一个部分又被称作子域。你可以看见子域是用点分开的，共有3个子域；即mailer，com和cn。

了解域名的方法是从右向左看子域。域名的结构是为了使每一个子域都告诉你一些有关计算机的信息。最右边的子域叫做最高级域，它是最常用的，当你往左边读时，子域就变得愈加专门化。

在我们所使用的例子中，最高级域cn告诉我们计算机属于中国(我们将在下面解释各种最高级域的含义)，下一个子域com告诉我们这个机构是商业机构，最后一个最左边的子域告诉我们是名为mailer的专有计算机名称。这样，当你录入命令：

mail harley@mailer.com.cn

你就可以对你自己说"我正在给harley发送邮件，他在一个商业机构，位于mailer的计算机上。"

当你输入地址时，你可以输入大写字母，也可以输入小写字母，例如下面两个地址是等同的：

mail harley@mailer.com.cn

mail harley@MAILER.COM.CN

此外，你还将看见两种常见的情况。第一种是，有一些人只用大写字母表达最高级域。

mail harley@mailer.com.cn

另一种情况是，有一些人喜欢强调计算机的位置：

mail harley@mailer.com.cn

其实，在所有的情况中，都可以任意选用大写字母。

按照一般的原则，作为Internet的地址全部都使用小写字母，因实在不需要混用大写字母。

如果你看见一个地址，其字母有一些是大写的，你把它们变成小写字母是不会带来任何影响的。

如果你们一定要使用一些大写字母，也最好不要用来写用户标识，因为在一些计算机系统上可能有一些差别，虽然这种差别是不应该产生的。

1.1.3 标准的Internet地址格式的形式

所有的Internet地址格式都按照标准格式：

userid(用户标识)@domain(域名)

但是，你完全有可能遇到不是标准格式的几种形式，我们曾经使用过的例子有3个子域：

harley@mailer.com.cn

而你们常常会看见，为了更加有确定性，地址会具有多个子域。这里是一个例子：

scott@emmenthaler.cs.wisc.edu

在这种情况中，用户标识是scott，域涉及计算机emmenthaler，那是威斯康星大学(wisc)(属于教育机构)计算机科学系的一部分。

大多数Internet节点都使用某种含义的词给计算机取名。例如，你可以看见用卡通怪人、神话英雄、地区界标，或用诸如此类的东西给计算机取名的。在威斯康星大学里的大多数计算机都是以"奶酪"这个名字取名的。

学世界地理的学生都知道，美国每个州都有给自己取浑名的习惯。例如，加尼福尼亚是"金子州"、纽约是"帝国州"。在威斯康星有如此多 "奶酪"计算机的原因是，威斯康星是众所周知的"奶酪州"。

可能当某人冥思苦想为威斯康星取名的时侯，所有好听的浑名都已经被用光了。

有些Internet地址只有两个子域(最低限度)。这里有两个例子。

rick@tsi.com

randy@ucsd.edu

当你看见只有两个子域的地址时，可能是两种情况。第一种情况可能表示这是在Internet上只有一台计算机的小公司。上面第一个例子，就是这种情况。该例子中的最高级域com告诉我们，这是一个商业机构(关于更多的最高级域域名含义下面就会谈到)，另一个子域，tsi，是计算机的名称。这是一个名为"技术系统集成者"公司，这也是人们经常使用的时髦名称--它在Internet上只有一台计算机。

第2个例子是拥有多台计算机的大机构，在这样的机构中，有一台计算机专用作与外部世界之间电子邮件的收发。

在我们的例子中，这个计算机的名字是ucsd.edu。该机构的系统管理员为每个人都安排有简单化的邮件地址，以便能够在ucsd.edu地址上收发邮件。

一般，术语"gateway"(网关) 涉及的是两个不同系统之间的连接，假如我们有一个网关(gateway)，在我们所使用的例子中，网关起着内部网络与外部世界之间的连接作用。因网关有用户标识和本地地址表，当一个邮件到达时，网关就可以校查该表，并把该邮件发送给相应的本地计算机。

例如，某人在名为misty的计算机上有一个melissa的用户标识，一般他的地址应该是：

melissa@misty.ucsd.edu

为了使他的地址简化，他可以向网关登记，然后就可以以melissa@ucsd.edu地址收到邮件，当邮件到达时，网关将自动地将它发送给名为misty的计算机。

因此，象你所看见的那样，一种具有两个子域的地址，通常要么这个机构很小(象tsi.com)，要么很大(象ucsd.edu)。

我们这样详细地解释这种情况，是因为可能有你们需要某个人确切地址的时侯，如果采用talk命令联络他们，在这种情况中，过于简化了的邮件地址不可能工作，你必须请求一个较长一点更加专有的地址。

现在，我们将讨论最后一种Internet地址，有时侯你可能会看到下面形式的地址。在这种地址中，%(百分号)符号被用作地址的一部分，该%符号排在@印刷符号的左边。例如：

melissa%misty@ucsd.edu.

这个地址的含义是，收到邮件的计算机(如ucsd.edu)将查看@符号左边的每一个项目(在本地址就是melissa%misty这一部分)，并试图弄清楚它。

通常%符号是从地方计算机的名称中分离出来的用户标识。在这个例子中，用户标识melissa使用的是一个名为misty的本地计算机。在本地网络中，从网关到本地计算机，有几种不同的连接，当邮件到达时，网关将选择最快的路径传送邮件。

许多人即使他们不了解Internet，但是他们都知道自己的Internet地址。有的机构还有多种发送邮件的地址。有的系统的管理员常常会把工作得最好的地址告诉他的用户。因此，你大可不必为地址的变化过多而担心。当你发送邮件给某人时，只用那个人给你的地址。

1.1.4 最高级域

像早先提到的那样，理解地址的方式是从右向左阅读。最高级域，所代表的范围最宽，在我们下面看到的例子中：

mailharley@fuzzball.ucsd.edu

最高级域edu，告诉我们这个计算机属于教育机构。我们还可以看到另一个地址

rick@tsi.com

在这里面的最高级域com表明是一种商业机构。

一般有两种形式的最高级域：一般称之为机构域，正如这两个例子中所述的；还有一种称之为地区域。

机构域是按Internet建立之前的地址编制法则制订的，原打算主要在美国使用。

最高级域表示的是机构类型的范畴和属性。表4-1表示的是各种机构类别(范畴)，所有的这些范畴，除int 是近期才作为跨越国家边界的某一机构(如NATO)增加的外，其余的都是自Internet开通就已经有了的。

一旦Internet扩大成国际性网络，它会需要新的、更加专有的最高级域。为了满足这种需要，就编制了新的地区域系统，在该系统中有许多这样的以两个字母的缩写代表一个国家的高级域。表4-2显示了具有代表性的地区域。

作为例子，看一看下面的地址：

michael@music.tuwien.ac.at

该例子表明在奥地利(at最高级域)的维也纳技术大学的一台计算机的地址。

有一些国家使用最高级域左边的一个子域将地址进行分类。例如ac涉及的是一个学术机构，co涉及的是一个商业公司。在我们的例子里，可以看见奥地利使用的是ac。

表二、最高级域地区域名范例对照表

就绝大部分而言，地区域就是某个国家所用的标准字母缩写。但是这个规则有些例外，如大不列颠王国，它的国际代码缩写为gb，可是它也使用英国uk作域名。

虽然你能够从表4-2中看见，美国也有一个地区域名称(us)但是他们用得不多，不过，除美国外，几乎所有国家都唯一地使用地区域名称。不管你的机构使用什么类型的最高级域，你都能够与Internet上的任一地址通信，任何地方也都能辨认最高级域的两种类型。

在英国和新西兰中，域的顺序常常被颠倒。例如，你可能看见象下面那样的地址：

peirce@uk.ac.oxford.compsci

有的人在与外界通信时，将域的顺序颠倒了，这样偶尔出现的地址颠倒的例子，就象对外国消费者出口的劣质酒一样。如果在英国境外使用这种地址，必须将子域颠倒过来，仍改为：

peirce@compsci.oxford.ac.uk.

以使其成为标准地址。

表一、最高级域机构域名属性对照表

域　　类型

com　商业机构（全称：commercial organization)

edu　教育机构（全称：educational institution)

gov　政府部门（全称：government)

int 国际性机构（全称：international organization)

mil 军队（全称：military)

net 网络机构（全称：networking organization)

org 非盈利机构（全称：non-profit organization)

表二、最高级域地区域名范例对照表

域 国家

at 奥地利（全称：Austria)

au 澳大地亚（全称：Australia)

ca 加拿大（全称：Canada)

ch 瑞士（全称：Switzerland("ConfoederatioHlvetia")

cn 中国（全称：China)

de 德国（全称：Germany("Deutschland")

dk 丹麦（全称：Denmark)

es 西班牙（全称：Spain("Espana")

fr 法国（全称：France)

gr 希腊（全称：Greece)

ie 爱尔兰共和国（全称：Republic of Ireland)

jp 日本（全称：Japan)

nz 新西兰（全称：New Zealand)

uk 英国（全称：United Kingdom)

us 美国（全称：United States)

1.1.5 伪Internet地址

有许多机构希望与Internet相连接，但是没有时间或经费与Internet保持长期的连接。作为一个替代的方法，它们与附近的Internet点商定，让其作为它们的邮件代理。例如，一个小公司可能与一个Internet入口提供者作这种协商。

这类机构利用这种设置，可得到一个看起来是标准的Internet地址。但是这些机构实际上并不与Internet相连接。

例如，假定Marlinspike咨询公司想要一个Internet地址， 它与一个地区Internet入口提供者SnowyNET商定处理它的邮件。作为地址的部分域名，取名为marlin.com，并用Internet寻址系统记录下来。任何送到marlin.com的邮件都从自动路由送到SnowyNET网关计算机。在商定时间，marlin.com计算机通过电话线与SnowyNET计算机连接，取走它的邮件。同时，从marlin.com送出公司的任何邮件都传到SnowyNET计算机上。

在有这种域名的公司，用户使用的地址看起来很象标准的Internet地址。例如，某人可能告诉你，他的地址是：

tintin@marlin.com

当你看见这样一个地址时，无法知道它是否表示一个真实的Internet地址。换句话说，仅从这个地址，常常无法断定计算机是否真正与Internet相连接。如果你看见一个象下面这样的名字：

small-company-name.com

你可以产生怀疑，因为这个格式根本就不对，甚至一些大公司也使用邮件代理形式，但是并没有真正与Internet连接。

如果要确切知道一台计算机是否与Internet连接(这很重要)，可使用host命令，紧接着，我们就来叙述它。

1.1.6 IP地址与host命令

前面，我们已经谈论了每个子域都是有具体名字的Internet地址。这种地址形式已为社会所接受，实际上，这种地址的外表下面，是一个典型的计算机游戏：真正的Internet地址实际上是数字，而不是字母。例如，我们前面提到的计算机ucsd.edu，实际上是128.54.16.1。

当然，人们使用字母作名字更方便些，但是，每次使用域地址时，系统就自动将其转换成数字，只不过你并不知道这个细节。你如果学习过Internet的基本原理，就应该知道Internet将数据包从一个地方移到另一地方那个功能叫做IP(Internet Protocol)。因此，地址的数字版本叫做IP地址。例如，计算机ucsd.edu的IP地址为128.54.16.1。

IP地址看来很象我们已经讨论的域地址，其中由句点分成几个部分。但是，IP地址的各部分并不直接与子域名字一一对应，因此请不要机械地用这个模式去套用。

可以在使用正规地址的任何地方使用IP地址。例如，下面两个发电子邮件的命令是等效的。

mail randy@ucsd.edu

mail randy@[128.54.16.1]

Internet的地址记录系统叫域名系统或DNS。DNS是将域名地址与IP数字地址这两个版本来回转换的一种TCP／IP服务。 幸好所有这一切都是在你不知道的情况下进行，否则你整天会为这些细节而烦恼。

偶尔你的系统识别域地址可能出现故障。如果出现这种情况，使用IP地址一样可以正常工作(只要IP的地址是正确的)。

如果要测试DNS，可以使用host命令。有以下两种格式：

host standard-address

host IP-address

如果你从一个标准地址入手：

host ucsd.edu

DNS将显示IP地址：

host 128.54.16.1

反之如果你从一个IP地址入手，DNS将显示标准地址。

只有Internet主机才有IP地址(仿真终端是没有的)。因此，可以使用host命令检查计算机是否与Internet直接连接。如使用host能测试出IP地址，说明该计算机是与Internet连接的。

如果指定的地址是未与Internet直接连接的计算机的，将看到：

host not found.

如果出现这种情况，请再检查一下拼写是否正确，以得出最后的结论。

host命令并不适用于所有的计算机。如果你的系统没有host，请试一下nslookup或nstest。

1.2 基本网络协议

2.1 RFC821(SMTP)

SMTP标准是Internet协议栈中的一个广为使用的上层协议。SMTP定义了如何在两个用户间传输邮件。

SMTP使用了spooling的概念，允许将邮件从一个本地应用发送到一个SMTP应用，SMTP应用把邮件存储到一个设备或内存中，一旦邮件到达SMTP应用，该邮件被放入到一个队列中，一个服务器检查是否有邮件到达，然后试图投递到达的邮件。如果邮件的收方不存在，服务器然后还会再进行尝试。最终，如果邮件不能被投递，则将该邮件丢弃或返回给邮件的发送者。这一概念被称为端到端的投递，并且它将邮件保存在队列中，直到邮件被投递出去。可以从两个RFC中找到有关SMTP的讨论。RFC822描述了报文的结构，其中还包括了信封。RFC821规定了在两台机器间控制邮件交换的协议。

2.1.1 SMTP模型

下图显示了SMTP的基本模型。发送方建立起与接收方SMTP的通信。在传输邮件之前，两个SMTP实体可以交换口令或其他鉴别信号，然后，发送方发出一个特殊命令，称为MAIL，该命令给出了发送方的标识及一些有关邮件交换的其他信息。接收方必须返回一个对MAIL命令的确认。在SMTP中，该确认可写为250，在另一些文档中，该确认为250 OK。不管是哪一种格式，确认表示所要求的邮件操作完成了。



图一、SMTP 模型

邮件传输过程的下一步是传输RCPT命令以标识报文的目的端，接收方仍必须给出对该命令的一个确认。

第三步是发送DATA命令。发送方SMTP发出该命令提醒接收方一个报文就要到达了。在发送了该命令之后，发送方SMTP便一行接一行地发送数据，直到发送方发出了一个特殊的控制字符序列，表示报文结束了。这时，服务器使用QUIT命令终止该过程。

2.1.2 地址段格式

发送方SMTP使用了一个标准的发送地址和接收地址格式，它们的形式如下：

local-part@domain-name

因此，SMTP名字遵从了域名系统(DNS)的概念，一些系统使用同一个服务器，将该名字映射成IP地址。实际应用中，这种格式可能出现为：

Jones@beta.aus.edu

其中，本地用户的名字为Jones，beta.aus.edu是Jones的域标识符。local-part@domain-name还可能采取其他形式以表示如下含义：

一个直接的连接(user@host)

邮件接收方位于一台非SMTP主机，通过一个邮件网关接收邮件(user%remote-host@gateway-host)

多台主机间的应答(@host-b@host-c@hostd)。

2.1.3 SMTP操作举例

发送方 接收方

建立连接 =========> <========= 220 OK

HELO Jones@alpha.com =========> <========= 250 Smith@beta.com

MAIL FROM :Jones@alpha.com =========> <========= 250 OK

RCPT TO : smith@beta.com =========> <========= 250 OK

DATA =========> <========= 354

<Email Data Line by Line> =========>

<CRLF>.<CRLF> =========>

<========= 250 OK

QUIT =========> <========= 221

上图显示了两个SMTP用户交换邮件的简单操作，图的左侧显示了发送方建立连接。接收方用220 OK作为响应。Hello命令用作在两台机器间交换的标识符。MAIL FROM命令告诉smith@Beta.com开始一个新的邮件事务。接收方使用该命令清除自己的缓冲区，复位状态表并为接收报文作好准备。RCPT命令给出了接收方的转发路径地址，在本例中为smith@Beta.com，应答为250 OK。DATA命令通知接收方接下来将发送报文的内容，应答为354，表示开始邮件输入，CRLF.CRLF表示邮件的结束。

数据被一行一行地发送，用CRLF.CRLF表示传输结束，接收方用250 OK作为响应，然后使用QUIT断开连接，服务器返回一个221表示服务器正在关闭连接。

下图显示了其他类型的SMTP邮件的例子。图的顶部显示了验证命令，它的目的在于确认接收方的名字。在本例中，Jsmith应当返回一个全名和一个完全指定的邮箱。图中第二个操作描述了SMTP如何在邮件列表中确认一个身份。通信双方应当返回全名和完全指定的邮箱。

发送方 接收方

验证一个用户名

VRFY smith =========> <========= 220 OK smith@beta.com

扩展一个邮件列表

EXPN List A =========>

<========= 250 Jsmith@beta.com

<========= 250 Brown@charlie.com

<========= 250 Mblack@delta.com

上面演示了SMTP众多功能中的少数例子，讲解了SMTP协议中标准命令。另外，TurboMail邮件系统还支持部分可选命令。

RSET

退出当前SMTP邮件交互过程，后退至HELO命令执行后的状态。

HELP

显示帮助信息。

NOOP

实际上不做任何事情，只要求接收方返回一个OK应答。

2.2 POP3 协议

2.2.1 POP3协议操作概述

首先， POP3 服务器启动 POP3 服务，并监听 110 端口。当客户机要使用 POP3 服务时， 它先与 POP3 服务器建立 一条 TCP 连接。若连接成功建立， POP3 服务器向客户端发送一条欢迎消息，然后客户机与 POP3 服务器之间就可以进入命令交互和响应状态，直到连接关闭或异常中断。

POP3 协议命令由一个大小写无关的关键字和零个、一个或多个参数组成。所有的命令都以一对回车换行（CRLF）结束。命令关键字和参数都由可打印的 ASCII 字符组成，由一个空格字符隔开。关键字只有三到四个字符，而每个参数最多可达40 个字符长。

POP3 协议的命令响应信息由一个状态标志和一个响应关键字组成，后面可以带响应的信息。所有的响应也以一对回车换行符（CRLF）结束。每条响应信息最多可达 512 个字符长（包括回车换行符 CRLF ）。 目前，POP3 协议响应标志有两个：成功标志"+OK"和失败标志"-ERR"。POP3服务器必须以大写字母的形式发送这两个响应标志。

对于某些 POP3 协议命令的响应信息可以由多行组成。对于这种响应， POP3 服务器首先要发送带有"+OK"或"-ERR"标志的响应行和一对回车换行符， 然后才发送随后的其它响应信息，它们的每一行都以一对回车换行符。当所有的响应都发送完后，服务器再发送一个结束行，该结束行由一个终结字?quot;." 和一对回车换行符结束。 若有以 "." 作为第一个字符开头的响应行，则该开头的"."必须被转义，方法是在该行开头在添加一个"."字符，于是该响应行就变为以".."开头的字符串。这样每次多行响应总是以五个字符 "CRLF.CRLF" 结束。 当客户端接收多行响应信息时，它必须检查每行的开头是否为"." 字符。 若是且后继字符不是"CRLF"， 则将该行的首字符"."去掉，而后续字符串不变。若首字符"."的后继字符为"CRLF"，则表示多行响应结束，并且该行 ".CRLF"不属于响应信息，要被去掉。

每个 POP3 会话总是处于一系列的状态变化中。一旦客户机与 POP3 服务器之间的 TCP 连接建立起来并且 POP3 服务器的欢迎信息传到客户端。 POP3 会话过程就进入认证状态。在该状态下，客户机必须进行身份验证。一旦验证通过，并且 POP3 服务器获得足够的资源打开用户的邮箱。这时 POP3会话过程进入"命令交互状态"。 在"命令交互状态"下，客户端可以向 POP3 服务器提出各种服务请求， 由 POP3 服务器完成该服务并返回成功或失败的响应信息。当客户端发出"QUIT"命令后， POP3 会话过程进入"更新状态"。 在"更新状态"下， POP3 服务器释放在"命令交互状态"下获得的资源， 更新用户邮箱， 然后关闭 TCP 连接。

POP3 服务器必须负责处理不可识别的、未实现的或语法错误的命令，并返回带错误标志"-ERR"响应信息。 若 POP3 服务器收到客户端传来的不属于当前所处状态的命令时，必须返回带错误响应"-ERR"的响应信息。

POP3 服务器应该具有在收不到命令请求一段时间后， 自动关闭 TCP 连接的功能。 这段间隔时间必须在 10 分钟以上。 在此期间，若收到任何命令请求，POP3 服务

器都应重新计时。 当连接超时，而收不到任何命令请求时， POP3 服务器不会进入"更新状态"，不会删除任何该用户的邮件。

2.2.2 POP3协议标准命令简介（RFC1939）

USER

这个命令以用户的帐号名称作为参数。

PASS

这个命令以用户的口令作为参数，缺省情况下用户的口令将以明文在网络上传输。

STAT

该命令无参数， 仅用于"命令交互状态"。

获取当前用户邮件总数和邮件总大小。POP3 服务器返回一个成功的响应，以 "+OK"开头，后跟一个空格字符，然后是当前用户邮箱内邮件的总数，之后又是一个空格字符，最后是用户邮件的总大小（BYTES）。

C：STAT

S：+OK 2320

LIST

该命令可以没有参数也可以有一个参数，该参数代表邮件的序号，但该序号不能是已删除邮件。

该命令仅用于"命令交互状态"。

C：LIST

S：+OK 2 messages(320 octets)

S:1 120

S:2 200

S:.

. . . . . .

C：LIST 2

S：+OK 2 200

. . . . . .

C：LIST 3

S：-ERR no such message， only 2 messages in maildrop.

RETR

该命令必须带有一个邮件序号， 该邮件序号不能是已删除邮件。

该命令仅用于"命令交互状态"。

若 POP3 服务器返回一个成功响应， 则随后的响应属于多行响应。 在首行 "+OK"响应之后， POP3 服务器把对应于该邮件序号的邮件发送到客户端。

C：RETR 1

S：+OK 120 octets

S：< The POP3 server sends the entire message here >

C:RETR 5

S:-ERR no such message.

DELE

该命令必须带一个邮件序号参数。仅用于"命令交互状态"

POP3 服务器把该邮件标志为"已删除"， 之后，任何对该邮件的操作都将返回错误。
POP3 服务器并不实际删除该邮件，除非 POP3 会话进入"更新状态"。

C：DELE 1

S：+OK message 1 deleted

. . . . . .

C：DELE 1

S：-ERR message 2 already deleted

NOOP

该命令不带参数，实际上也不做任何事情。

仅用于"命令交互状态"。

C： NOOP

S： +OK

RSET

该命令不带参数。仅用于"命令交互状态"。

POP3 服务器将所有已被标志为删除的邮件的删除标志取消，并返回成功响应。

C：RSET

S：+OK maildrop has 2 messages (320 octets)

QUIT

该命令无参数，仅用于"命令交互状态"。

POP3 服务器把所有已标志为删除的邮件删除，并释放在"命令交互状态"中获得的资源，然后关闭 TCP 连接。 POP3 会话结束。

此外，AIMC 除支持上述POP3 协议的标准命令之外，还支持其它可选命令。

TOP

该命令有两个参数，第一个参数是邮件序号，第二个参数是要显示的邮件信体信息前n行的行数。该命令会返回邮件头信息和邮件信体指定的前n行信息。本命令仅用于"命令交互状态"。

C: TOP 1 10

S: +OK

S: < the POP3 server sends the headers of the

message， a blank line， and the first 10 lines

of the body of the message >

UIDL

该命令可以没有参数也可以有一个参数，该参数代表邮件的序号，但该序号不能是已删除邮件。该命令会返回唯一标识指定邮件的ID值，本命令仅用于"命令交互状态"。

C: UIDL 2

S: +OK 2 QhdPYR:00WBw1Ph7x7

LAST

该命令无参数，获取已读的且未删除的编号最大的信件编号，本命令仅用于"命令交互状态"。

2.2.3 Example:

以下是一个完整的 POP3 会话例子。

S: < wait for connection on TCP port 110 >

C: < open connection >

S: +OK POP3 server ready < aimc.com >

C: USER mrose

S: +OK mrose is a real hoopy frood

C: PASS secret

S: +OK mrose's maildrop has 2 messages (320 octets)

C: STAT

S: +OK 2 320

C: LIST

S: +OK 2 messages (320 octets)

S: 1 120

S: 2 200

S: .

C: RETR 1

S: +OK 120 octets

S: < the POP3 server sends message 1 >

S: .

C: DELE 1

S: +OK message 1 deleted

C: RETR 2

S: +OK 200 octets

S: < the POP3 server sends message 2 >

S: .

C: DELE 2

S: +OK message 2 deleted

C: QUIT

S: +OK dewey POP3 server signing off (maildrop empty)

C: ＜close connection＞

S: ＜wait for next connection＞

2.3 IMAP4 协议

2.3.1 IMAP4协议简介

IMAP是斯坦福大学在1986年开发的，它是一个开放的标准，被设计成POP的超集，而POP是个地位牢固的开放邮件标准。IMAP标准的最新版是IMAP第4版修订版1(RFC 2060)。IMAP第4版修订版1与IMAP4的实际差别仅仅在于前者提高了拨号链路的性能。

有关IMAP4标准的详细信息请参见华盛顿大学的IMAP主页(www.washington.edu/imap/)。尽管在日常使用中，IMAP的特性及对它的支持还没有压倒POP3，但是它的发展趋势已经显而易见了。主要的群件产品厂商都计划支持IMAP4。

根据定义，POP3是个存储转发的信息交换系统。POP3邮件服务器把消息发送给客户，自己并不在服务器上保存副本。与之不同，IMAP4是个客户机/服务器模式的应用。无论是POP3还是IMAP4，进来的邮件都存储在中央的邮件服务器上。用户使用遵循协议的邮件客户软件来连接到邮件服务器上，先进行身份验证，鉴定登录名和口令，然后用户才获得访问邮箱的权利。但是，二者的共性仅限于此。

与POP3不同，IMAP4能以三种模式或者说消息传送范式来与客户进行交互：离线、在线和断连方式。此外，IMAP4可以让用户访问多个私用和共享邮箱。

在离线方式中，客户软件把邮箱存储在本地硬盘上以进行读取和撰写信息的工作。当需要发送和接受消息时，用户才连接服务器。对于那些长期奔波、很少停留在某个固定处所的人，他们通常使用离线方式。POP3典型地以离线方式工作。

相比之下，在线用户访问的邮箱是在邮件服务器上，但是邮箱仍然由客户软件处理。邮件始终存储在服务器上。在线方式主要是由位置固定的用户使用，典型地是在快速LAN连接下进行。但是从远程拨进的功能较弱的计算机在这种模式下也可以工作得很好。有一些POP3服务器也提供了在线功能，但是，它们没有达到IMAP4的功能级别。

断连方式提供了最大的灵活性。客户软件把用户选定的消息和附件复制或缓存到本地磁盘上，并把原始副本留存在邮件服务器上。缓存中的邮件可以被用户处理，以后用户重新连接邮件服务器时，这些邮件可以与服务器进行再同步。当前，该特性主要由邮件服务器实现，很少有客户软件支持断连方式。在我们评测的产品中，Sun的Internet Mail Client 是唯一提供这一功能的客户软件。

对于IMAP4而言，无论是在哪一种操作系统上，使用IMAP4邮件客户软件的用户都可以读取和回复邮件，远程地把邮件存储在层次式的文件夹中，并且可通过IMAP4邮件主机同步客户文件。客户软件和服务器相互配合，允许过滤来自特定地址的邮件，还能够防止通过低速拨号链路下载大文件。用户可以查看主机上的邮件报文，如果需要，可以决定把其中哪些内容下载到远程连接的客户机上。用户还可以有选择地下载邮件，而不必非要下载邮件的附件。

2.3.2 IMAP与POP3及Web Mail的比较

TurboMail电子邮件系统是目前国内众多电子邮件系统中率先提供IMAP4服务的系统，完整支持最新版本的IMAP4rev1(RFC2060)协议。IMAP是Internet Message Access Protocol的缩写，顾名思义，主要提供的是通过Internet获取信息的一种协议。IMAP4是IMAP协议的第4个版本，正如POP3是POP协议的第3个版本一样。在AIMC邮件系统内，IMAP是一种强有力的邮箱访问方式。

和POP3的区别

由于很多用户都对POP3非常熟悉，我们就从POP3说起。POP3提供了快捷的邮件下载服务，用户可以利用POP3把邮箱里的信下载到PC上进行离线阅读。一旦邮件进入PC的本地硬盘，就可以选择把邮件从服务器上删除，然后脱离与Internet的连接并选择在任何时候阅读已经下载的邮件。

IMAP同样提供了方便的邮件下载服务，让用户能进行离线阅读，但IMAP能完成的却远远不只这些。首先，IMAP提供的摘要浏览功能可以让你在阅读完所有的邮件到达时间、主题、发件人、大小等信息后才作出是否下载的决定。也就是说，你不必等所有的邮件都下载完毕后才知道究竟邮件里都有些什么。如果你根据摘要信息就可以决定某些邮件

对你毫无用处，你就可以直接在服务器上把这些邮件删除掉，而不必浪费你宝贵的上网时间。如果你的IMAP客户端软件完整支持IMAP4rev1的话(如Netscape 4.5)，则你还可以享受选择性下载附件的服务。举例来说，假如一封邮件里含有大大小小共 5 个附件，而其中只有2个附件是你需要的，你就可以只下载那两个附件，节省了下载其余3个的时间。

和WebMail的比较

也有很多用户喜欢通过Web来联机收发邮件，其中一个很重要的原因是这些用户希望把他的邮件都留在服务器上，并且通过WebMail服务建立多个文件夹，然后分类归档地管理自己的邮件。这样，WebMail的用户就可以不分时间地点，只要有一个浏览器就可以马上从服务器上获得自己的邮件，不管是刚收到的还是已经存放了很久，也不必担心客户端的PC重新安装了操作系统或换了一台电脑以后邮件全部丢失了的问题。

IMAP同样满足了WebMail用户的需要。IMAP与POP3不同的地方关键是在支持离线阅读的同时也鼓励用户把邮件存储和组织在服务器上。和WebMail一样，通过IMAP，允许用户在服务器上建立任意层次结构的文件夹，并且可以灵活地在文件夹之间移动邮件，随心所欲地组织你的邮箱(这些显然是通过POP3做不到的)。只要你的邮件存储在服务器上，任何时候通过一个IMAP的客户端软件都可以立即联机获得你的邮件，这一点与WebMail保持一致。

但是，IMAP的以下优点足以把WebMail用户吸引过来：凡是WebMail的用户都必需无奈地阅读页面上的广告，都必需花费宝贵的时间和带宽来下载页面上的图面，修饰字符等等；IMAP则忠实地只为你的Email服务，不让你的资源有丝毫的浪费。

IMAP的其它好处

IMAP协议还允许你方便地利用你的邮箱作为信息存储工具，一般的IMAP4客户软件都支持邮件在本地文件夹间和服务器文件夹间的随意拖动，让你得心应手地把本地硬盘上的文件存放到服务器上，然后在你需要的时候同样方便地取回来，所有的功能仅需要一次鼠标拖放的操作来实现，是不是很好?

总的来说，POP3和WebMail的优点在IMAP里都有，而IMAP却能很好地克服POP3和WebMail各自的缺点。喜欢新鲜的你是不是也想试一试?

2.3.3 IMAP4协议处理状态及工作流程

任何时候，每个IMAP4服务处理线程都处于4种处理状态的其中一种。大部分的IMAP4命令都只会在某种处理状态下才有效。如果IMAP4客户端软件企图在不恰当的状态下发送命令，则服务器将返回协议错误的失败信息，如BAD或NO等等。

非认证状态

在这个状态下，客户软件必须发出认证请求命令。在IMAP4连接建立时，服务器处理线程自动进入这个状态。

认证状态

在认证状态下，客户软件必须选择一个邮箱。这个状态在认证请求命令得到确认答复后进入，或在预认证连接建立后直接进入。

已选择状态

这个状态表示IMAP4客户软件已经选择了某一Folder。在这个状态下可以发送所有检索邮件内容的命令。

离线状态

在这个状态，连接已经终止，服务器将关闭这个连接。客户端软件可以发出命令或由服务器强制进入这个状态。

2.3.4 IMAP命令概要

IMAP协议中的命令与POP3中最大的不同是每条命令前都一有一个由Client指定的标签，同一个会话中，Client发出的每条命令都会有不同的标签；而Server对每个Client发出的命令都必须以相同的标签作应答。这就使得IMAP会话中Client可以同时送出多个命令，而Server则完全可以并发处理这些命令，不必等待上一个命令执行完毕才处理下一个。

下面是若干常用的IMAP命令

＜TAG＞ LOGIN ＜user name＞ ＜password＞

LOGIN命令用于登录IMAP服务器，与POP3显著不同的是用户名和口令同时作为参数提交给Server，而POP3完成同样的操作需要USER和PASS两条命令。

C: a001 LOGIN SMITH SESAME

S: a001 OK LOGIN completed

＜TAG＞ CREATE ＜folder＞

CREATE可以创建指定名字的文件夹。文件夹名字通常是带路径的文件夹全名。

C: A003 CREATE owatagusiam/

S: A003 OK CREATE completed

C: A004 CREATE owatagusiam/blurdybloop

S: A004 OK CREATE completed

＜TAG＞ DELETE ＜folder＞

DELETE命令删除指定名字的文件夹。文件夹名字通常是带路径的文件夹全名。

C: A683 DELETE blurdybloop

S: A683 OK DELETE completed

C: A684 DELETE foo

S: A684 NO Name "foo" has inferior hierarchical names

C: A685 DELETE foo/bar

S: A685 OK DELETE Completed

＜TAG＞ RENAME ＜old folder＞ ＜new folder＞

RENAME命令可以修改文件夹的名称。

C: A683 RENAME blurdybloop sarasoop

S: A683 OK RENAME completed

C: A684 RENAME foo zowie

S: A684 OK RENAME Completed

＜TAG＞ LIST ＜BASE＞ ＜template＞

LIST命令用于列出邮箱中已有的文件夹，有点像操作系统的列目录命令。这个命令可以包含起始的路径位置和需要列出的文件夹所符合的特征，可以使用通配符"*"。

C: A101 LIST "" ""

S: * LIST (\Noselect) "/" ""

S: A101 OK LIST Completed

C: A102 LIST #news.comp.mail.misc ""

S: * LIST (\Noselect) "." #news.

S: A102 OK LIST Completed

C: A103 LIST /usr/staff/jones ""

S: * LIST (\Noselect) "/" /

S: A103 OK LIST Completed

C: A202 LIST ~/Mail/ %

S: * LIST (\Noselect) "/" ~/Mail/foo

S: * LIST () "/" ~/Mail/meetings

S: A202 OK LIST completed

＜TAG＞ APPEND ＜folder＞ ＜attributes＞ ＜size＞ ＜mail data＞

APPEND命令允许Client上载一个邮件到指定的Folder中。命令中包含了新邮件的属性、大小，随后是邮件数据。

C: A003 APPEND saved-messages (\Seen) {310}

C: Date: Mon， 7 Feb 1994 21:52:25 -0800 (PST)

C: From: Fred Foobar ＜foobar@Blurdybloop.COM＞

C: Subject: afternoon meeting

C: To: mooch@owatagu.siam.edu

C: Message-Id: ＜B27397-0100000@Blurdybloop.COM＞

C: MIME-Version: 1.0

C: Content-Type: TEXT/PLAIN; CHARSET=US-ASCII

C:

C: Hello Joe， do you think we can meet at 3:30 tomorrow?

C:

S: A003 OK APPEND completed

＜TAG＞ SELECT ＜folder＞

SELECT命令让Client选定某个Folder，表示即将对该Folder内的邮件作操作。

C: A142 SELECT INBOX

S: * 172 EXISTS

S: * 1 RECENT

S: * OK [UNSEEN 12] Message 12 is first unseen

S: * OK [UIDVALIDITY 3857529045] UIDs valid

S: * FLAGS (\Answered \Flagged \Deleted \Seen \Draft)

S: * OK [PERMANENTFLAGS (\Deleted \Seen *)] Limited

S: A142 OK [READ-WRITE] SELECT completed

＜TAG＞ FETCH …

FETCH命令是IMAP协议里最复杂的命令。FETCH的命令参数很多、很复杂，难以一一列举，但基本的特征是允许将邮件按照MIME结构拆解为零碎的部件来提取。例如，可以利用FETCH命令提取邮件头、某一个附件、或某一邮件附件头部的某一字段，等等。

＜TAG＞ STORE ＜mail id＞ ＜new attributes＞

STORE命令用于修改指定邮件的属性，包括给邮件打上已读标记、删除标记，等等。

C: A003 STORE 2:4 +FLAGS (\Deleted)

S: * 2 FETCH FLAGS (\Deleted \Seen)

S: * 3 FETCH FLAGS (\Deleted)

S: * 4 FETCH FLAGS (\Deleted \Flagged \Seen)

S: A003 OK STORE completed

＜TAG＞ CLOSE

CLOSE命令表示Client结束对当前Folder的访问，随后可以SELECT另一Folder。

C: A341 CLOSE

S: A341 OK CLOSE completed

＜TAG＞ EXPUNGE

EXPUNGE将邮箱中所有打了删除标记的邮件彻底删除。EXPUNGE删除的邮件将不可以恢复。

C: A202 EXPUNGE

S: * 3 EXPUNGE

S: * 3 EXPUNGE

S: * 5 EXPUNGE

S: * 8 EXPUNGE

S: A202 OK EXPUNGE completed

＜TAG＞ LOGOUT

LOGOUT命令结束本次IMAP会话。

C: A023 LOGOUT

S: * BYE IMAP4rev1 Server logging out

S: A023 OK LOGOUT completed

(Server and client then close the connection)

2.3.5 IMAP4会话实例

S: * OK IMAP4rev1 Service Ready

C: a001 login mrc secret

S: a001 OK LOGIN completed

C: a002 select inbox

S: * 18 EXISTS

S: * FLAGS (\Answered \Flagged \Deleted \Seen \Draft)

S: * 2 RECENT

S: * OK [UNSEEN 17] Message 17 is the first unseen message

S: * OK [UIDVALIDITY 3857529045] UIDs valid

S: a002 OK [READ-WRITE] SELECT completed

C: a003 fetch 12 full

S: * 12 FETCH (FLAGS (\Seen) INTERNALDATE "17-Jul-1996 02:44:25 -0700"

RFC822.SIZE 4286 ENVELOPE ("Wed, 17 Jul 1996 02:23:25 -0700 (PDT)" "IMAP4rev1 WG mtg summary and minutes"

(("Terry Gray" NIL "gray" "cac.washington.edu"))

(("Terry Gray" NIL "gray" "cac.washington.edu"))

(("Terry Gray" NIL "gray" "cac.washington.edu"))

((NIL NIL "imap" "cac.washington.edu"))

((NIL NIL "minutes" "CNRI.Reston.VA.US")

("John Klensin" NIL "KLENSIN" "INFOODS.MIT.EDU")) NIL NIL "< B27397-0100000@cac.washington.edu >")

BODY ("TEXT" "PLAIN" ("CHARSET" "US-ASCII") NIL NIL "7BIT" 3028 92))

S: a003 OK FETCH completed

C: a004 fetch 12 body[header]

S: * 12 FETCH (BODY[HEADER] {350}

S: Date: Wed, 17 Jul 1996 02:23:25 -0700 (PDT)

S: From: Terry Gray < gray@cac.washington.edu >

S: Subject: IMAP4rev1 WG mtg summary and minutes

S: To: imap@cac.washington.edu

S: cc: minutes@CNRI.Reston.VA.US, John Klensin < KLENSIN@INFOODS.MIT.EDU >

S: Message-Id: < B27397-0100000@cac.washington.edu >

S: MIME-Version: 1.0

S: Content-Type: TEXT/PLAIN; CHARSET=US-ASCII

S:

S:)

S: a004 OK FETCH completed

C: a005 store 12 +flags \deleted

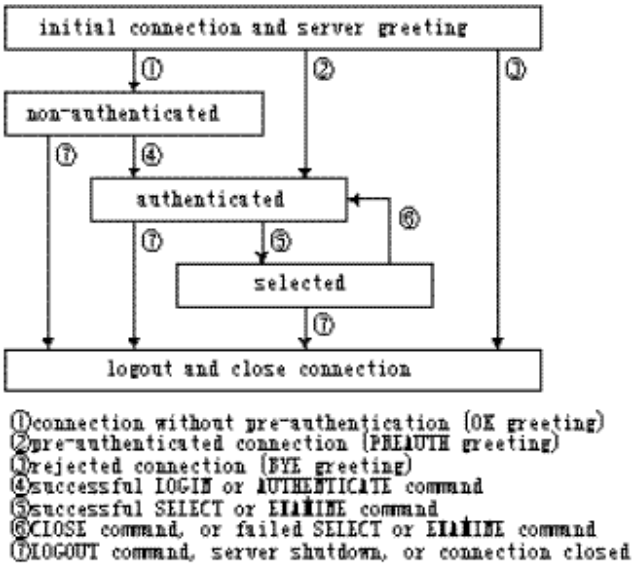
S: * 12 FETCH (FLAGS (\Seen \Deleted))

S: a005 OK +FLAGS completed

C: a006 logout

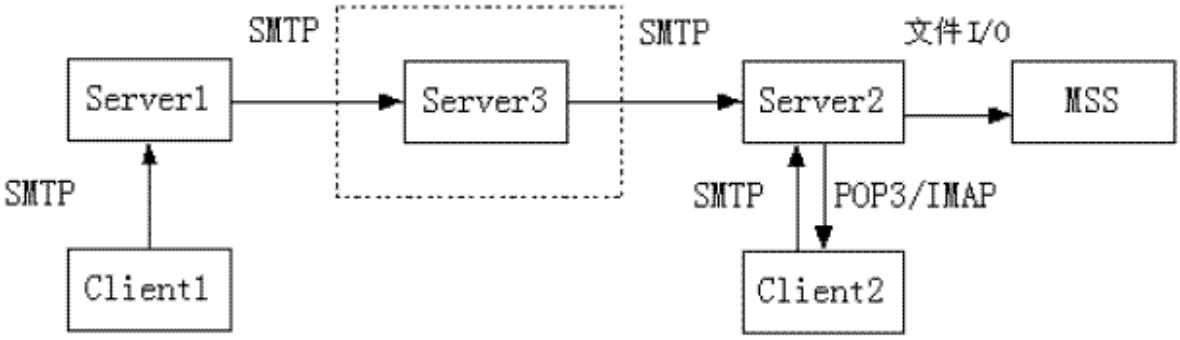
S: * BYE IMAP4rev1 server terminating connection

S: a006 OK LOGOUT completed



图二

1.3 电子邮件系统传送过程



图三、电子邮件传送过程示意图

如上图，描述了一封邮件从邮箱one@server1.com发往邮箱two@server2.com的传送流程，其中域名为server1.com的邮件系统安装在Server1上，域名为server2.com的邮件系统安装在Server2上。途中邮件经过若干Mail Transfer Agent作为中转，最终送达接收方Mail Server，传送流程的各个阶段采用了不同的邮件协议。

1. 发件人采用Mail Client端软件Client1撰写一封邮件发给收件人，首先将该邮件从Client1发送到他的邮箱one@server1.com所在的服务器Server1上。这步由Client发往Server的过程采用的是SMTP协议。

2. 一般情况下，邮件从发件人的邮件服务器Server1直接传送到收件人邮箱two@server2.com所在的服务器Server2 上。也可能需要经过第三方的邮件服务器Server3做中转后再送达接收方，这个过程称为Relay。图中虚线方框表示的是0到若干台Relay Server。这步在Mail Server之间进行邮件传送的过程采用的是SMTP协议。

3. 邮件发送到接收方的服务器Server2上后，由Server2负责将该邮件投递到收件人的邮箱two@server2.com中，保存在Server2的磁盘阵列中。此时，邮件传送过程结束。

4. 收件人访问他在Server2上的邮箱two@server2.com时， 采用Mail Client端软件Client2将Server2邮箱中的邮件下载到本地硬盘上阅读。这步将邮件从Server下载到Client的过程采用的是POP3或者IMAP协议。

5. 如果收件人回复该邮件，会把回复信件从Client2发送至Server2，这步过程与第1步相同，采用的是SMTP协议。

1.4 电子邮件技术名词解释

4.1 SMTP

是"简单邮件传输协议"(Simple Mail Transfer Protocol)的缩写。当你在互联网上收发电子邮件时，是SMTP在维护传输秩序，它规定邮件服务器之间进行哪些工作。

4.2 POP

是Post Office Protocol的缩写，是互联网电子邮件协议标准。我们可以通过有POP服务功能的主机传送及接收电子邮件。该协议的缺陷是，当你接收电子邮件时，所有的信件都从服务器上清除，下载到你的本地硬盘。当然也有一些客户端程序可以将电子邮件留在服务器上，或设置成超过一定大小的文件不可下载。随着邮件采用多媒体格式，邮件会越来越大，我们希望能够灵活掌握下载什么文件、何时下载，这就需要IMAP协议。目前POP的版本为POP3。

4.3 IMAP

是Internet Message Access Protocol的缩写。IMAP提供了一个在远程服务器上管理邮件的手段，它与POP协议相似，但功能比POP要多，功能包括：只下载邮件的标题、建立多个邮箱和在服务器上建立保存邮件的文件夹。

4.4 MIME

是Multipurpose Internet Mail Extensions的缩写。它是一种协议，可使电子邮件除包含一般纯文本以外，还可加上彩色图片、视频、声音或二进位格式的文件。它要求邮件的发送端和接收端必须有解读MIME协议的电子邮件程序。

4.5 UUencode

像MIME一样，UUencode可以让你在互联网上发送二进制文件。UUencode即从Unix到Unix之间编码，将二进制数据转换成ASCII文本格式，以便加入电子邮件的正文。收件人必须用UUdecode应用程序将文本转回二进制格式。

4.6 Attachment

附件，即附在电子邮件正文上的文件。许多邮件包都采用MIME编码来附加文件。

4.7 Filters

过滤器，也称"规则"，过滤器可用来管理收到的文件和保存的文件。具有过滤功能的软件可根据一定的规则自动采取行动，如按照发信人地址自动将邮件放到有关的文件夹内，或对某类信件予以事先写好的回复，或删除指定"事由"内容的信件。

4.8 News Group

新闻组是个人向新闻服务器所投递邮件的集合。新闻服务器由公司、群组或个人负责维护，并可管理成千上万个新闻组。

4.9 Directory Server

是用于在全球范围内查找用户和商业伙伴的强大的搜索工具，通常支持LDAP协议。

4.10 Address Book

具有存储联系人信息的功能，可以方便地检索联系人的信息。另外，它还能访问用于在Internet上查找用户和商业伙伴的Internet目录服务。

4.11 Spam

这个动词/名词是表示向很多人发送商业性质的消息，它也覆盖了其它不时发生的通过电子邮件发出的诸如破坏性的，或者令人讨厌的消息。

4.12 TCP

Transmission Control Protocol，传输控制协议，它在发送方将信息分割成若干个包，每个包标有序列号和接收方地址。此外包中还插有差错控制信号，当接收方出错时，将申请重发；所有包均正确接收后，就根据序列号构造原来的信息。

4.13 IP

Internet Protocol，Internet协议，它支持将数据从一个地方传送到另一个地方，可以处理各个节点间或各个网络间的数据包传送，而与网络类型无关。

练 习 题

1) 通过Outlook Express撰写一封邮件，"发件人"、"收件人"、"抄送"和"主题"等项填写齐全，保存为*.eml类型的文件。用文本编辑器打开该邮件，分析邮件头信息编写格式。

2) 接上一题，采用HTML格式书写邮件正文，并为该邮件插入多个不同格式的附件，保存为*.eml类型的文件。用文本编辑器打开该邮件，分析邮件中的MIME编码信息。

3) 采用telnet方式创建SMTP连接，模拟SMTP 会话过程。

a. 发件人向收件人发送一封邮件，要求按照收发双方的真实email地址填写该邮件头的From和To字段，保证收件人收到邮件后，能从邮件中正确得到收发双方的真实信息。

b. 发件人同时向多个收件人发送同一封邮件，邮件撰写规则同上题。

c. 发件人向收件人发送一封邮件，要求随意假造该邮件头的From字段，使得收件人收到邮件后，无法从中正确得到发件人。

d. 验证在当前邮件系统中某一用户是否存在。

4) 采用telnet方式创建POP3连接，模拟POP3会话过程。

a. 登录指定邮箱，查看该邮箱状态信息。

b. 查看指定邮件内容，然后从邮件服务器上删除该邮件。

c. 使用TOP命令查看某邮件的信头内容，使用UIDL命令查看Unique-id Listing，使用LAST指令查看最近阅读邮件的序号。

5) 采用telnet方式创建IMAP连接，模拟IMAP会话过程。

a. 登录指定邮箱，创建、改名或删除folder。

b. 查看某邮箱inbox文件夹内的邮件，并删除某封邮件

6) 通过比较POP3、IMAP和WebMail获取邮件之异同，深入分析IMAP的优势。

7) 分析邮件传送过程，并指明在传送各个阶段采用何种协议。

第二章 TurboMail 电子邮件系统介绍

近年来Internet在全球范围内得到了迅猛的发展，据美国因特网理事会2000年9月预测，到2005年全球用户将达到10亿！在中国1997年底互联网用户数大概是70万左右，到目前位置已经是2000多万，增长了足足20多倍，而且还按照每年翻一番的速度在增长！在这种情况下，互联网上的各种服务都将承受前所未见的压力，而作为Internet最为重要的服务之一的E-mail服务因其具有方便快捷地进行信息交流的特点，所以有着传统的邮政服务所无法比拟的优势。作为Internet信息服务提供者，Email系统能否满足大容量的要求，将是决定整个服务成败的关键。

2.1 传统电子邮件系统的局限

传统的电子邮件系统由于用户不多，其系统构架存在一定的缺陷，难以承受大用户量的访问。比如应用非常广泛的Sendmail系统：

首先是由于其用户信息存放于/etc/passwd文件中，在大用户量的情况下将出现认证效率大大降低的问题，而且由于Unix本身的限制，通常用户数无法突破6万。

其次是它的邮件存放位置，通常位于/var/mail下面，每个人的邮件是一个文件，用户越多邮件文件数量就越多，在Unix系统中同一级目录中的文件超过一定数量后文件系统运转效率将会大大降低；

另外，因为每一个用户的多封邮件位于同一个文件中，用户在取信过程中如果出现网络异常就很容易出现邮箱的死锁。

传统的电子邮件技术的扩展性较差，不支持邮箱的分布式存储，所有邮箱必须存储在同一共享磁盘阵列中。这样，一方面容易造成磁盘阵列单点故障，另一方面亦不利于在保证处理效率的基础上大幅度提高用户数量。

传统的电子邮件技术在邮件过滤、安全防卫等方面提供的手段较少，配置十分复杂，并且对系统管理员的技术和经验的依赖程度很高。

上述问题大大限制了sendmail邮件系统在大用户量情况下的应用。因此如果要适应大容量的需要，必须从体系结构上进行改进，使之在大用户量的情况下仍旧能够保持良好的性能。

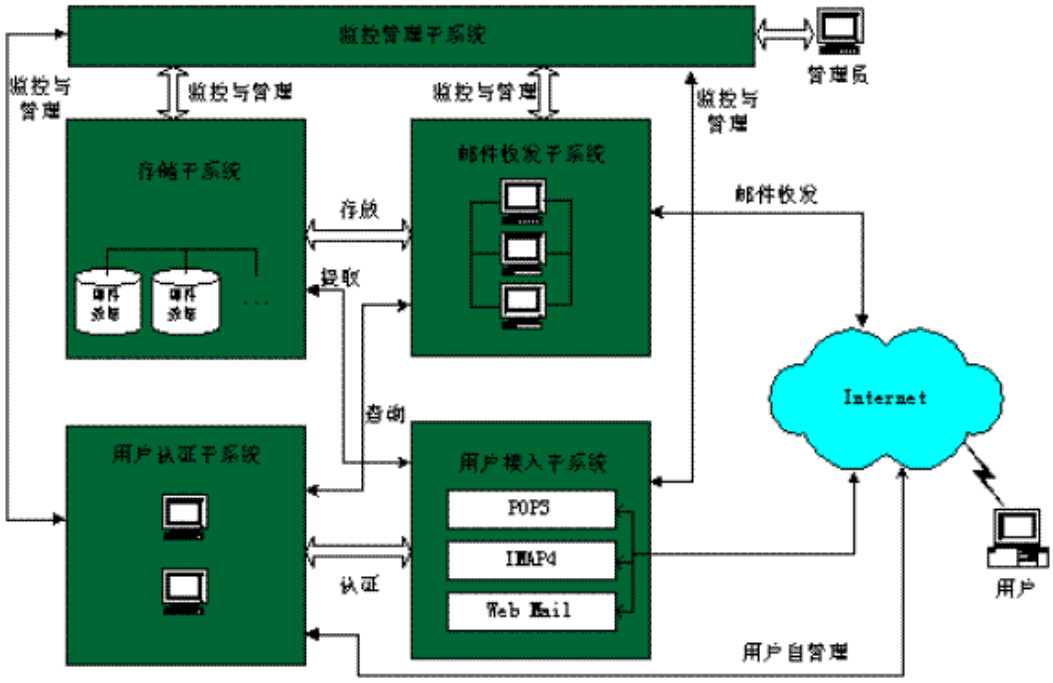
2.2 大容量电子邮件系统的特点

所谓大容量邮件系统，通常是指所支持的用户量在5万以上的系统。目前在市场上专为大容量应用所开发的Email系统有很多，其中包括Sun公司的SIMS、Software.com公司的Post.Office和InterMail、Netscape公司的Messaging Server、亚信公司的AI Mail Dispatcher、AI Mail Center、广州网易公司的MailServer产品等。这些产品各自具有不同的特点，但从技术上分析和评估大容量电子邮件系统时可以从以下几个方面进行考察。

2.2.1 系统结构

通常电子邮件系统分为两种结构，一种是集中式，另一种是分布式。集中式电子邮件系统的特点是邮件系统的所有部件都运行于同一台服务器上，不能够将不同的功能模块放在不同的机器上进行负载分担。这种结构所能够支持的用户量受到硬件本身功能的限制，只能单纯地通过提升服务器的硬件性能来增加所支持的用户量，在达到一定的限额之后，再升级系统的硬件所支持的用户量不能成比例地增长，因此这类系统通常适用于支持15万以下的用户的情况。

分布式邮件系统的特点是可以将软件的不同模块如收发模块、用户认证模块、邮件存储模块、用户接入模块等分别运行在不同的机器上共同来完成整个电子邮件系统的功能。如果需要的话每一种模块都可以拆分在不同的服务器上运行实现负载分担，因此系统可以根据需要和用户的使用模式进行定制。这种结构所支持的用户量有比较大的灵活性。目前采用这种结构的系统如InterMail、AIMC都可以支持到几百万以上的用户。如今最为著名的站点Hotmail所采用的也是多台机器分布式并行操作的方式，目前其用户已经达到近三千万。分布式邮件系统的体系结构大致如图所示。



2.2.2 系统效率

如今用户通过网络阅读自己的邮件所遇到的延时通常由两部分组成，一个是网络本身的时延，这个可以通过提高网络层带宽来实现；另一个是电子邮件系统本身所带来的时延。用户检取自己的一封邮件通常要通过目录服务器搜索、目录服务器认证、邮件存储服务器定位和邮件服务器读取等多步操作，每一步都会带来或多或少的时延。因此应当充分采用如千兆以太网、SAN高速存储等硬件技术以及LDAP、多线程等高效的软件技术，对每一部分的效率都进行提高，尽量消除瓶颈。

在用户接入部分，分布式结构的邮件系统通常采用前端代理、DNS轮循或第四层交换的方法实现负载的分担。其中采用第四层交换机的方式效率较高，特别是采用千兆以太网技术的第四层交换机可以大大提高系统的效率。

在用户数据库管理和检索部分现在通常使用LDAP协议，它是一种公共标准，而且效率较高。

邮件存储部分通常容易成为大容量邮件系统的瓶颈，特别是分布式的结构，多台服务器访问用户数据库必须提高效率。通常比较好的解决方案有两种，即NFS方式和SAN方

式。NFS方式的优点是使用比较方便，而且互操作性较好。缺点是数据在局域网上传送，对局域网造成很大的压力，另外所有数据都多经过一层NFS封装，增加了额外的开销。SAN是一种新兴的存储解决方案，各个服务器通过光纤通道交换机和集线器与所有的存储设备建立物理联接，充分利用光纤通道的高速度以及这种连接方式的灵活性消除系统的瓶颈。另外，在这种结构下远端灾难备份的效率也大大提高。

2.2.3 对公共协议的支持

在Internet的世界里支持公共标准已经成为一种共识，电子邮件是一种互通的工具，因此支持公共标准就更加重要。在电子邮件领域通常的标准包括SMTP、ESMTP、POP3、IMAP、LDAP、MIME，DNS，UUCP，SSL等，支持的公共标准越多，就说明该系统的兼容性和互通性越好，其用户在使用电子邮件的时候就不需要考虑其信件在Internet上的通行问题。如今大多数的电子邮件系统都在逐步地支持尽可能多的公共标准，从而争取在更广泛的地方应用，获得更多的市场份额。

2.2.4 对平台的支持

所支持的平台数也是一种邮件系统的重要特性之一，现在除NT以外，仅Unix的平台就非常多，如Sun Solaris， HP-UX， IBM-AIX， SGI IRIX， Digital Unix等，应用都非常广泛，如果能够支持更多的平台，将极大地方便用户。有的邮件系统出自硬件厂家之手，通常绑定在自己的平台上面，如Sun公司的SIMS，也有的厂家的产品有不同的版本适用于不同的平台，如Software.com公司的Post.Office。

2.2.5 垃圾信件的处理功能

经常使用电子邮件的用户都对经常收到的一些垃圾信感到非常厌烦，因为它们不仅浪费用户的网络带宽，还占用了用户的很多宝贵时间。另外对于邮件系统管理员来说，许多借用自己的服务器进行邮件中继的信件也使服务器的工作效率大大降低。一套好的电子邮件服务器软件应当具有灵活而强大的垃圾信过滤和防中继功能。在系统级上管理员可以设置邮件过滤、重定向和防中继策略，普通邮件用户可以设置系统为其提供过滤功能，拒收不希望的信件。现在多数邮件系统都具有一定的垃圾邮件处理功能，但功能的实现机理和灵活程度各不相同，将来的发展趋势应当是使用更加简便灵活，过滤功能更强并且对系统的资源消耗更小。

2.2.6 邮件列表功能

邮件列表是电子邮件系统所能够提供的一种有效的增值服务功能。它类似于传统系统中的Alias功能，但又与它有很多不同。邮件列表可以允许任意的用户进行订阅和退定，对用户的订阅和退定请求的处理都是自动进行的，不需要管理员的人为干预。每个邮件列表有一个所有者，它可以给所有订户发送信息邮件，在定户达到一定数量后就可以有偿地搭载一定的广告获得收益。ISP也可以将邮件列表外包从而坐收利益。这一点已经在许多实际的案例中得到体现，电子邮件系统是否支持灵活的邮件列表功能是评估邮件程序的一个重要准则。

2.2.7 虚拟主机和虚拟域

虚拟主机和虚拟域功能是大容量电子邮件系统的另一个重要增值服务特性。拥有大容量电子邮件系统的ISP可以为小型企业提供电子邮件外包服务，在同一套服务器上提供多个域名，并且为每一个域指定一名管理员，这样小型企业的用户就不必为自己专门建立电子邮件服务器，也就省掉了其管理和维护费用。这对于小型的公司是比较经济的。ISP提供这种服务就可以获得增值收入。支持虚拟主机和虚拟域功能也是电子邮件系统的一种发展趋势。如今一些邮件系统可以支持虚拟主机，但不同的域的用户并没有真正分开可以进行二级管理，即系统管理员一级和域管理员一级。

2.2.8 Web Mail

Web Mail对于大型的ISP特别是提供免费电子邮件的ISP来说是非常重要的特性。首先一些用户没有自己固定的计算机，习惯于使用Web界面实现邮件收发和邮箱管理。另一个方面，提供Web Mail功能后可以在用户访问该站点的时候提供动态广告，增进ISP的额外收入。特别是一些提供门户站点的ISP，免费Web邮件作为其服务的重要组成部分将大大提高站点的访问量。

2.2.9 系统的可扩展性

互联网络技术发展日新月异，用户量也以极高的速度增长，作为ISP来说，希望能够保护自己在网络设备上的投资，也就是说不希望投资的设备在短时间内全部更换。因此这就要求电子邮件服务器具有较好的可扩展性。前面所提到的集中式结构的邮件服务器只能通过提高系统邮件配置来进行扩展，受到的限制很大；分布式结构的系统在开始可以是一台服务器，随着用户量的增加，增加服务器就可以使其处理能力成倍增加，因此具有较好的可扩展性。

2.2.10 系统的可管理性

一套支持十万甚至更多用户的邮件系统本身是非常复杂的，如果不能够为管理员提供简单易用的管理界面，每天的管理工作将会非常复杂。特别是分布式的邮件服务器，多台服务器并行工作，如果管理工作必须分别在不同的机器上进行，管理工作量将很多，在出现问题的时候处理起来也非常麻烦。一套好的邮件服务器管理应当是集中式的，管理员通过任意一台PC机或工作站，使用浏览器对整个系统进行管理和配置修改。

另外从用户的角度来讲，系统应当提供一个Web界面允许用户在经过管理员授权的情况下对自己的个人信息，自动回复信息，过滤器等进行配置。

2.2.11 支持 Unified Messaging

Unified Messaging简言之就是让用户可以在世界上的任何地方，采用电话、电子邮件系统、Web浏览器、传真机或传呼机等设备中的任意一种获知并访问自己的电子邮件、Voice Mail、传真等各种Message，以便随时随地实时的与外界保持联系，并且能够节省大量的时间和费用。在这个系统中位于后台的实际上就是一台邮件服务器，所有格式的邮件都存在用户的个人邮箱中。到现在为止，完全实现Unified Messaging所定义的内容还存在一定距离，但它的部分功能如Email to Pager、Email to 手机短消息等已经在一些地方使用而且反映很好，这体现了传统电子邮件向多元化发展的一个趋势。作为一套好的电子邮件系统应当在设计的时候就为将来提供Unified Messaging做好铺垫，在系统中预留到电信设备的接口，以便在需要的时候与电信设备互联。

2.2.12 系统安全性

随着互联网的普及，通过网络进行破坏性活动的黑客也越来越多，电子邮件系统保存了敏感的邮件信息，如果为黑客所破坏将对用户造成很大的损失。好的邮件系统应当具有良好的安全机制，使得用户的信息得到良好的保护。如今较为常用的安全机制包括多级口令保护、低权限运行、TLS/SSL等。

2.2.13 若干流行电子邮件系统技术分析

现代大规模电子邮件系统通常有四个组成部分：

存储用户资料并提供验证服务的用户验证服务器UAS(User Authentication Server)；

接收来自用户和系统外部邮件的邮件传递服务器MTA(Mail Transfer Agent)；

存储用户邮件的邮件存储服务器MSS(Mail Storage Server)；

提供邮件提取或检索服务的邮件提取服务器MRA(Mail Retrieve Agent)；

通过分析一个电子邮件系统利用何种技术实现这四个基本模块、采用何种系统结构将这四个部件有机组合并协同工作，就可以基本了解这个电子邮件系统的扩展能力、容错能力和处理能力。

2.3. 电子邮件系统评估准则

对于现在的网络服务商来说，能否为用户提供可靠且富有特色的电子邮件服务是其主要挑战之一。由于三分之一以上的用户上网主要是为了收发邮件，衡量电信公司和网络服务商的标准就通常是他们能否提供可靠和方便的邮件服务。一次广为人知的错失就足以使长期积累下的好名声毁掉。结果就是许多IP服务商纷纷转向使用商业化的邮件解决方案来满足这种极具挑战性的需求。

通过列出一系列服务商在考虑购买邮件系统时应向软件制造商方提出的问题，本节将为您提供一个评估此类软件的框架。电子邮件是一项主要的业务，而且基于客户的经验，我们认为一个服务商的技术及经营的成功还决定于其邮件服务系统的能力。这里描述了成功的邮件服务系统所需的功能和产品成熟程度。

2.3.1 结构

3.1.1 规模

1. 系统容量可否扩展以满足将来的要求？如何满足？到什么程度？

2. 纵向扩展包括对现有服务器增容，例如：RAM，CPU及存储设备；

3. 横向扩展是由添加服务器完成增容。

4. 服务系统扩容是否影响用户？

5. 当增加硬件以提高处理能力时，现有用户能否受益？还是只有新用户才能受益？

6. 邮件系统是不是真正全分布式设计？系统中的各部件能否放置到不同的地理位置？

7. 系统可否透明地在多服务器之间均衡负载？

8. 系统有没有较弱的部件?

9. 基于负载管理原因，存储系统，邮件交换系统，POP/IMAP服务器应分开并能在各自单独的平台上运转。但是，这些组成部分也应能够在同一服务器上进行操作。

10. 某个部件的多个进程是否可以同时运行于一台服务器上(例如MSS)?

11. 多个邮件访问服务器(SMTP/POP/IMAP)必须能同时访问一个单独的邮件存储服务器。

12. 邮件访问服务器能否从一个较弱的目录服务器取得身份确认和信箱位置资料?

13. 系统中的各个组成部分可否根据信息量，用户数量或消息接入的频繁度和种类单独扩展?

3.1.2 效率

系统是否有效利用资源? 边际容量成本会增加，减少还是不变?

系统是否多线程? 用户级多线程还是系统级多线程?

是否采用单副本存储技术? 同时送到多个收件人的信是否只在系统里存储一个拷贝?

如果一条信息需要再发一次(由于错误)，反复步骤是否能避免收件人收到重复信息?

3.1.3 性能

邮件系统是否能有效利用硬件资源以达到（现在或将来）所需的要求?

系统性能是否会随负载增加而剧减，系统是否会在达到一定负载后就崩溃?

3.1.4 可靠性

如果系统崩溃，邮件是否会丢失?

邮件是否在被处理前已完全写入硬盘?

如果系统崩溃时邮件还未完成处理，结果会怎样?

服务器间通讯协议是否基于消息交换?

系统能否处理Internet上的意外情况?

预定要转发到外地的邮件是混合存储还是按目标域分类?

系统是否可处理非标准邮件格式?

3.1.5 可持续性

磁盘或服务器的错误是否导致邮件服务出错?

邮件服务器组件是否支持"高可靠性"?

邮件服务器组件是否包括"系统监控模块"?

支持哪些"高可靠性"产品?

可持续性如何体现?

服务器冗余

服务器failover

上述两者兼而有之

哪些组件设计为高可靠性的?

MTA（邮件交换系统）

MS（邮件存储系统）

POP

IMAP

目录服务

有无其它非高可靠性的组件?

3.1.6 网络标准

系统是否本身在其核心支持SMTP，POP3，IMAP4及MIME？是否要求网关改变邮件原有的协议或格式？

2.3.2 管理

3.2.1 备份和恢复

能否不关掉需备份的服务器在线备份？

假如一个邮件存储器的文件系统失败，邮箱能否恢复？如果能，则恢复到上次备份还是失败时的状况？

个人信箱能否被有选择性地恢复？

3.2.2 灵活性和伸缩性

怎样对系统进行自定义或修改以提供额外的功能？

管理员能否设定基于服务器的处理规则？

何种API可用来扩展或控制邮件系统？

3.2.3 邮箱转移

邮箱和用户资料能否从您现在的系统顺利转移到新的系统中？

提供了什么样的转移工具？

信箱转移可否分段进行？还是必须要一次性规模转移？

有无防范措施以备错误发生时需要恢复转移？

3.2.4 用户自行管理

用户可否自己进行适当管理而不向服务支持部门寻求帮助？

用户能否方便地改变其POP/IMAP密码？

用户能否更新其公共地址本资料？

用户能否选择使管理者设定的垃圾邮件处理功能生效或无效?

用户能否定义基于服务器的邮件过滤功能?

用户能否设置外出（自动回复）邮件?

2.3.3 服务器管理

3.3.1 配置

MTA

MTA是否会在高负载情况下停止服务？(例如，对并发的SMTP处理规定最高数量)

能否按IP地址或子网设定?

MTA是否会限制用于邮件数据的内存?

MTA能否输送邮件到邮件存储器而不将其先存入磁盘?

MTA能否要求进行SMTP认证?

能否支持多域名及其独立管理?

SMTP协议通信中的各步的空闲退出时间能否调整?

MTA所有的SMTP错误或消息性文字邮件能否设置?

待发邮件每次发送之间的间隔可否调整?

待发邮件的时效能否调整(在其被打回之前)?

SMTP端口号(预设为25)能否改变?

MTA能否完成地址补全？(比如，将To: john变为To: john@yourdomain.com)

邮件存储

"欢迎新用户"的邮件能否自动发给新用户?

邮箱是否能只在收到mail时才真实地创建?

信箱大小限制是否能按用户或组别进行设定?

超过邮箱大小限制的结果是否可调? (例如，警告，邮件打回或不打回)

POP及IMAP是否共享一个通用邮件存储器?

是否提供独立的存储区域提供给特定的用户(组)?

有无用来在信息存储服务器上透明转移（平衡负载）信箱的工具?

POP3

POP服务器能否在高负载状态下停止服务? (例如，对同时进行的POP任务规定最高数量)

POP端口号(预设为110)能否改变?

POP任务能否作为PROXY访问其他POP服务器?

IMAP4

由于IMAP4通常是一个锦上添花的服务，因此应该可以在目录服务中按用户或组设定是否允许使用。

目录服务

是不是所有的用户/信箱属性都存储在中央目录?

目录信息是否存在分布式的高速缓冲存储器中以避免目录服务的瓶颈?

目录缓冲存储器是否定期更新（按设定的时间）?

如果根目录缓冲存储器无法获取，有无一个替换机制以便需要目录信息的服务器可从另一个缓冲存储中读取?

当根目录缓冲存储器恢复之后有无一个自我恢复机制而不需要人工操作?

LDAP

核心LDAP资料是存在一个可靠的关系型数据库中还是一般数据库中?

LDAP数据能否在线备份?

如果您的LDAP数据损坏，您有什么恢复措施?

LDAP服务是否纵向可扩展(通过增加服务器，不影响服务地扩容)?

日常操作

是否提供图形管理界面?

是否提供命令行工具? （对开发用户特定的自动处理过程非常重要）

命令行界面（CLI）提供的功能是不是图形界面（GUI）功能的扩展集?

GUI提供何种CLI不能提供的功能?

事件记录是否详细? 其细节程度能否设置?

系统是否有一个运行监视系统，可以在发现特殊情况时警告操作员?

系统有无错误管理功能来辅助解决错误?

定期维护

日志文件能否自动截断更新?

有没有数据库存储优化工具?

能否作热备份?

扩容

是否提供了容量分析工具来跟踪和预告容量需要?

服务器升级

系统能否进行滚动式的升级，以便服务只需暂停更新一个独立部件的时间?

不同版本的部件能否相互协作?

2.3.4 安全性和防止垃圾邮件

3.4.1 安全性和身份验证

邮件系统的核心是否使用了公开代码？如果是这样，黑客就可以仔细地找出系统核心的每一个漏洞。

是否每个邮箱都要求一个UNIX帐号？如果是这样，那就方便了黑客们取得系统权限。

有无进程以超级权限运行？如果是这样，黑客们就易运行需要系统管理员特权的UNIX管理命令。

系统是否借助于外部确认机制?

/etc/passwd

NIS，NIS+

密码是否存储于用户可访问的服务器？将密码存在公众无法直接访问的服务器上较为安全。

密码是用明码还是加密存储?

系统有无任何巩固密码安全度的措施?

SMTP/POP/IMAP服务器是否支持SSL或TLS?

3.4.2 预防垃圾邮件，伪装(spoofing)或使服务停止的攻击

系统管理员（及最终用户）有什么措施来防止垃圾邮件，诈骗和拒绝服务的攻击?

从黑名单所列的来源的连接能否按以下规则拒绝：

域名

IP

子网

自收录在黑名单上的域名或用户发来的信件是否会被拒收?

如果有包含空白回复地址的邮件发给多个收件人，是否会被阻挡?

同一IP地址或子网的并发连接数是否有限制?

是否支持SMTP身份确认？这要求在用户通过您的服务发送邮件前进行身份确认。

邮件转发（relay）是否仅限于FROM栏为特定的用户地址发来的邮件?

用户可否选择忽略管理者设定的垃圾过滤规则?

2.3.5 灵活性，延伸性和一体性

3.5.1 杠杆型的基础结构

邮件系统的目录服务能否被其它如LDAP，RADIUS，DNS，DHCP等的服务使用?

商业型的服务能否以同样结构为客户提供服务?

3.5.2 与现有系统融为一体

什么API可用来得到系统帐号资料?

命令行

SQL

Perl

C

系统记录输出能否直接定向到用户的记录过滤器或事件监控器上?

日常管理的事务能否通过Script完成?

您的客户支持部门能否容易地知道使用者的密码?

3.5.3 集群组别管理

ISP们需要提供一种方式让用户们购买并独立管理一部分帐号。这些集群用户会由群组管理员来处理这些帐号的日常事务。

系统是否提供多级管理权限，并通过密码保护?

能否收集群内计数和资源报告?

3.5.4 不同级别的服务

不同类别、不同需求的用户可否被分别设定? 例如，用户可以有下列选择:

基本邮箱：POP，5MB份额

加强邮箱：POP，10MB份额，转发邮件，垃圾邮件过滤

特级邮箱：POP或IMAP，100MB份额，转发邮件，别名，垃圾邮件过滤，SSL

2.3.6 集成商的经验和支持

3.6.1 专业服务

是否能提供在下述领域的优良专业服务?

目标管理

站点审计

计划

安装

转移

部署

设置

一体化

自定义

测试

专业培训

3.6.2 支持

是否有昼夜不间断的技术支持服务?

3.6.3 文档和培训

文档是否全面，包括下述方面:

结构概览

命令行功能

图形界面功能

日志管理

编程操作

最终用户操作

每一个可能出现的错误信息是否记录以下要素并给出解释:

名称

参数

原因

效果

程度

建议方案

是否长期提供管理和操作培训?

3.6.4 适用性，安装基准和参考

邮件系统是否适用于多种硬件平台，还是限定您只能使用一种品牌?

这种系统是不是成熟、经得起考验的产品?

其它有名的ISP是否成功采用了该系统?

使用这种系统最大的运行环境是什么?

其它ISP是否有倾向试用或放弃这种系统?

未来方向和市场重心

练 习 题

1) 分布式邮件系统分哪四大部件? 分别负责哪些工作? 它们之间是如何协同工作的?

2) 评估一个邮件系统的优劣应该从哪些方面考虑?