邮件系统中的SMTP认证机制

陈韬伟，车文刚，张海洲
(昆明理工大学 信息工程与自动化学院，云南 昆明 650051)

摘要：SMTP认证是邮件客户端用户向服务器端验证身份的一种机制，通过利用基于客户端与服务器的认证协议形成了连接双方相互通信的安全层，有效的阻止了垃圾邮件的泛滥.本文在描述了电子邮件系统中SMTP协议特点的基础上，以Qmail作为邮件传输代理(MTA)，提出了采用嵌入式模块来实现SMTP认证机制的方法，并与其他几种防垃圾邮件的方法进行了比较.从而为今后邮件系统中防止垃圾邮件的传播提供了技术参考.

关键词：邮件传输代理；SMTP；Qmail
中图分类号：TP393 **文献标识码**：A **文章编号**：1007－855X(2003)02－0083－04

SMTP AUTH in E－mail System

CHEN Tao-wei, CHE Wen-gang, ZHANG Hai-zhou
(Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650051, China)

Abstract: Smtp Auth is an authentication mechanism that an SMTP client may indicate to the server. By using the protocol based on the authentication between client and server, a security layer of negotiation over connection is established to successfully prevent the spamers and unauthorized people from using SMTP server. The principle and implementation of realizing SMTP AUTH in E－mail system with Qmail are discussed. Compared with the other methods of Anti－spam, a builtin module is used to realize the authentication mechanism of SMTP., to offer a technological reference in ending up being open relay.

Key words: MTA; SMTP; Qmail

0 引言

电子邮件(Electronic mail)，亦称E－mail，是Internet服务中重要的应用之一.由于它简便、廉价、高速的通信方式而深受广大Internet用户的欢迎[1].然而，由于技术、历史的原因，邮件系统最初是允许公开转发的(open relay：接受针对任何接收方的消息，并对邮件消息进行转发).所以，无限制转发常常被发送垃圾邮件的人利用.为了防止垃圾邮件的传播，大部分邮件服务器已经通过配置文件的方式、限制IP地址方式来关闭公开转发的功能.此外，防止其转发功能被滥用的另外一个方法就是在发送邮件时要求用户认证(SMTP认证)，就象用户利用POP3/IMAP服务器收信是需要认证一样.

本文回顾了电子邮件系统相关协议，从用户级、系统级分析了几种防垃圾邮件的方法.最后，对SMTP认证在具体实现和应用中的问题提出了解决方法.对于在建立大容量的电子邮件系统中，如何从全方位、多层次的控制垃圾邮件的传播提供了理论和实践的参考.

1 电子邮件基本原理及其相关协议

1.1 电子邮件基本原理

电子邮件系统级的主要功能模块是邮件传输代理(MTA：Mail Transfer Agent)，它的任务是传输进入和发出的邮件消息.对于发出的邮件消息，MTA根据带有接受方地址的消息决定发往本地主机还是一个远

收稿日期：2002－09－25.
第一作者简介：陈韬伟(1972～)，男，硕士研究生；主要研究方向：计算机网络与多媒体.

程邮件服务器;对于进入的邮件消息,MTA 必须能够接受远程邮件服务器的连接请求,并能够为本地用户接受邮件消息.随着电子邮件系统需求的不断增长极其复杂性,IETF(Internet Engineering Task Force)制订了一组有关电子邮件系统的标准协议,如:SMTP(Simple Message Transfer Protocol)、POP3 (Post Office Protocol version 3)等.图 1 中描述了邮件系统之间传输的关系,发送方与接受方之间邮件消息的传送、接收过程.并给出了客户端通过 **POP3/IMAP** 服务器认证收取邮件的方式.

由于 MTA 的转发功能,SMTP 服务器并不区分邮件是来自一个客户端还是来自另外一个 SMTP 服务器,只要双方建立了 SMTP 协议的连接,那么 SMTP 服务器就进行转发.

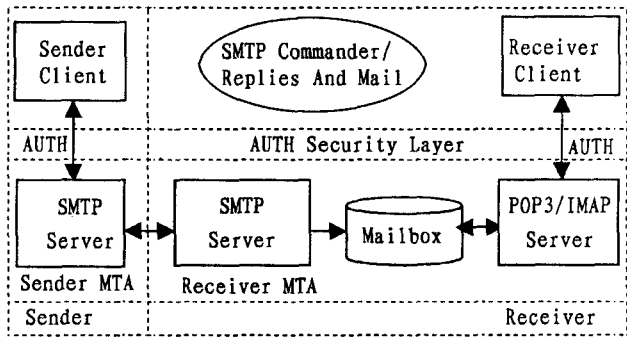

图 1 电子邮件传输框图

1.2 SMTP 协议

简单邮件传输协议(Simple Mail Transfer Protocol)是作为 Internet 上 MTA 服务器之间传输消息的基本方法而被开发的[2].它主要包括三个方面的内容:

1) SMTP 协议工作于 Internet 上.在端口 25 上使用一个 TCP/IP 连接.

2) SMTP 协议采用了一组简单的命令来建立连接并在主机之间传输消息和数据.命令通常由单个词及其后面的附加信息构成,如:"MAIL FROM: < cctw@ygd.edu.yn > "明确了消息的来源.每一个命令行通过简单的 ASCII 码进行传输.待收到命令后,远程主机以同样方式回应,告知命令是否接受成功.

3) SMTP 的一个重要特性是它的转发机制,通常发送方的邮件服务器 A 能够直接连接到目标服务器 B,则 A 直接将消息发送给 B.否则,就需要路由经过一个或多个 SMTP 服务器转发由 A 所发出的消息,最终达到 B.

综上所述:按照 SMTP 协议[4],发送方与目标邮件服务器之间的通信过程如下:

C:Telnet localhost 25 (在端口 25 上建立连接)

S:220 ygd.edu.yn ESMTP (邮件服务器响应)

C:MAIL FROM: < cctw@ygd.edu.yn > (消息的来源)

S:250 ok

C:RCPT: < zhz@ygd.edu.yn > (发送邮件的目标地址)

S:250 ok

C:DATA (发送邮件的内容)

S:354 go ahead

C:....etc.....etc...

C: < CRLF > . < CRLF >

S:250 ok

C:QUIT (客户端发送命令请求中断链接)

S:221 Connection closed by foreign host (邮件服务器终止链接)

2 SMTP 认证及其实现

2.1 SMTP 认证的提出

认证技术是信息安全理论与技术的一个重要方面.SMTP 属于身份认证,是安全系统中的第一道关卡.如图 1 所示,SMTP 认证使用了和 POP3 相似的认证机制(用户通过客户端从邮件系统上收取邮件的时候,用户必须进行身份认证),它的提出是基于以下两点:

1) 为了防止垃圾邮件的泛滥,几乎所有的邮件系统都对转发功能进行了限制,即:只接收不转发.不

幸的是，大多数邮件系统的用户均习惯使用他的邮件用户代理MUA(Mail User Agent，例如OUTLOOK，FOXMAIL等)向远程主机发送邮件.而MUA发送消息都是使用SMTP协议进行转发的.所以，邮件系统还是要提供一定的转发的功能.

2) 为漫游用户提供转发.虽然邮件系统已经作了有选择性转发的IP配置，但是合法用户使用不同的ISP，导致了用户的域或IP地址可能并不在你的邮件服务器所允许的转发范围内[8].

这样，SMTP认证就在客户端与服务器之间建立了一个认证机制，从而形成了基于通信协议和相互连接的简单安全层[7].从而解决了邮件系统的转发的问题，同时，允许合法的用户无论身在何处都可以发送电子邮件，阻止了非法用户和垃圾邮件的制造者利用SMTP服务器转发的功能.

2.2 SMTP认证的实现

SMTP认证的提出使得SMTP协议需要进一步扩展[5]，在原有命令基础上加入了一些认证的命令来验证用户[6].认证方式有LOGIN，CRAM－MD5，PLAIN等几种.不过目前国内仅仅支持LOGIN方式.认证过程如下：

C：AUTH LOGIN(客户端以LOGIN为认证的方式)

S：334 dXNlcm5hbWU6(编码字符串解码后为"username:"，说明要求客户端发送用户名.)

C：dXNlcm5hbWU6(以BASE64码的方式应答用户名，这里以用户名为username为例)

S：334 cGFzc3dvcmQ6(编码字符串解码后为"password:"，说明要求客户端发送用户口令.)

C：cGFzc3dvcmQ6(以BASE64码的方式应答用户密码，这里以密码为password为例)

S：235 Authentication successful.(认证成功)

针对LOGIN为认证的方式、现代流行的Qmail作为SMTP服务器为例，我们采用嵌入模块的方法来实现SMTP的认证机制，具体方案如下：

1) 认证模块的嵌入.由于Qmail是一个由Dan Bernstein开发的、免费使用且提供源代码的包.所以，可以根据应用需要开发符合自身邮件系统的包嵌入到源程序中.为此，主要在对客户端与服务器进行"请求——应答"的过程代码中，根据上述协议加入AUTH认证的命令代码.

2) 认证协议的要求.客户端与服务器端在认证的过程中要求以BASE64码的方式进行"请求、应答"，所以，为把传递给服务器的BASE64码转化为原来的用户名及密码进行验证，要在源程序中加入解码的模块.

3) 实现的具体方法.代码的认证部分主要采用了管道技术，在socket编程的基础上取回用户名和密码后，创建子进程.父进程利用管道将用户名和密码发给子进程，子进程收到后进行用户验证.子进程是利用原POP3的验证程序.

2.3 SMTP认证实现的优化

在上述SMTP认证的实现中，基于以下的原因需要对模块原有的方法进行进一步的改进和优化：

1) 子进程采用了POP3的验证程序.为了获取数据库中的用户名和密码，子进程以root的用户权限运行；而父进程则是以qmaild的用户权限来执行.因此需要使用setuid系统命令允许父进程访问运行本来无权访问的进程.这给系统本身安全性带来了隐患.

2) 利用fork()函数、管道技术.在大用户量频繁的身份验证情况下这种技术会造成系统瓶颈.

基于上述原因，结合自身邮件系统的要求，在源程序中直接嵌入认证的模块.程序中加入认证函数int chkpwd()以内联的方式获取数据库中的用户名和密码来实现认证.这样把原来利用子进程认证方法变为直接的静态认证方式就可以解决了上述的安全性和系统开销的问题.

3 SMTP认证在防垃圾邮件中的作用

对于大容量电子邮件系统，SMTP认证方式解决了邮件转发服务和它所引起的垃圾邮件之间的矛盾，但要从多个层次，多个方面防止垃圾邮件的产生，就需要其他防垃圾邮件的方法.这关系到系统整体性能的关键之一.以基于UNIX下的Qmail邮件系统为例，介绍其他几种防垃圾邮件方法的特点，进一步说明

SMTP 认证在防垃圾邮件中的作用:

1) RBL(Real Time Black List)实时黑域名列表是在 Internet 中根据用户的举报和系统检测而设立的发送垃圾邮件域名的实时更新的列表.通过相应的 Rlbsmtpd 应用程序可定期访问该列表来更新本地列表,从而拒绝从该列表中发出的电子邮件.通过 RBL,能够控制大多数垃圾邮件,但由于它是按照域名进行控制,所以如果域名不小心在列表中,那么,从该域发出正常的邮件就会被拒收.

2) TCPSERVER 网络守护进程.由于 inetd 程序存在难以处理大量的突发性的连接请求、内存消耗大、安全性不好的问题,故开发了 inetd 程序的替代程序 Tcpserver[2,3].在完成 inetd 程序原有功能的同时,一个最通用的特点是它可以创建与应用程序连接的规则,既邮件系统管理员可以设立规则数据库来拒绝某个 IP 主机的访问.利用这种方式能够手工的控制垃圾邮件的发送,但缺点是只有受到垃圾邮件的攻击后才可以把其 IP 地址放入规则数据库中.

3) 系统级和用户级的过滤.

• 系统级过滤:可以由系统管理员指定发件人、接收人、主题、甚至内容的关键字进行系统级过滤,如果消息符合过滤规则,SMTP 服务器就拒绝接收.这样,可以防止垃圾邮件的中转.

• 用户级过滤:系统在接收邮件后发往邮箱时,由用户本人设置相应的过滤规则,对邮件进行扫描,凡符合条件的邮件内容均被拒绝.

综上所述,可以看出每一种方法都是事后人为处理,而 SMTP 认证是在客户发送邮件的时候就对用户有选择性了,但是无论是事前预防还是事后处理都不能最终解决垃圾邮件的产生,因此邮件系统要从多层次,多方面的利用多种方法来全方位的限制垃圾邮件.

4 结束语

SMTP 认证解决了客户端的合法漫游用户利用 SMTP 服务器的转发问题,同时又防止了垃圾邮件的产生.作为大容量电子邮件系统,SMTP 认证无疑是对其功能和安全性方面的完善.虽然垃圾邮件不可能通过某种技术彻底的解决,但是可以在建立邮件系统防垃圾邮件的过程中,综合采用多种技术,达到客户端与服务器端、系统级和用户级的多安全层次;达到系统功能和安全的一致性.在一定程度上可以节省用户上网的时间开销、节省邮箱有限的资源以及节省系统本身由于垃圾邮件的频频请求造成的负载过重.

参考文献:

[1] Andrew S. Tanebaum. Computer Networks(Third Edition)[M]. Prentice Hall PTR 1996.

[2] Richard Blum. Running qmail[M]. SAMS, 2000.

[3] 于中江,车文刚,等.利用 Fast CGI 应用程序提高 Apache Web Server 性能[J].昆明工大学学报(理工版),2001,26(6):35~38.

[4] RFC 821[OL].http://www.ietf.org/rfc/rfc0821.txt?.number=821.

[5] RFC 1869[OL].http://www.ietf.org/rfc/rfc1869.txt? number=1869.

[6] RFC 2554[OL].http://www.ietf.org/rfc/rfc2554.txt? number=2554.

[7] RFC 2222[OL].http://www.ietf.org/rfc/rfc2222.txt? number=2222.

[8] RFC 2505[OL].http://www.ietf.org/rfc/rfc2505.txt? number=2505.