SMTP&POP3 实验

一 实验目的

1. 理解 SMTP 协议的工作原理
2. 掌握邮件消息的格式
3. 了解简单的邮件服务器的搭建流程

二 预备知识

1. 邮件传输协议

电子邮件需要在邮件客户端和邮件服务器之间，以及两个邮件服务器之间进行传递，就必须遵循一定的规则，这些规则就是邮件传输协议。SMTP 协议定了邮件客户端与 SMTP 服务之间，以及两台 SMTP 服务器之间发送邮件的通信规则；POP3/IMAP 协议定义了邮件客户端与 POP3 服务器之间收发邮件的通信规则。

2. SMTP 协议命令格式

SMTP 协议中一共定了 18 条命令，发送一封电子邮件的过程通常只需要其中的 6 条命令（如表 1 所示）即可完成发送邮件的功能。对于 SMTP 邮件发送程序发送的每一条命令，SMTP 邮件接收程序都将回应一条响应信息。

表 1

SMTP 命令格式	说明
ehlo<SP><domain><CRLF>	ehlo 命令是 SMTP 邮件发送程序与 SMTP 邮件接收程序建立连接后必须发送的第一条 SMTP 命令,<domain>表示 SMTP 邮件发送者的主机名。ehlo 命令用于替代传统 SMTP 协议中的 helo 命令。
auth<SP><para><CRLF>	如果 SMTP 邮件接收程序需要 SMTP 邮件发送程序进行认证时，它会向 SMTP 邮件发送程序提示它所采用的认证方式，SMTP 邮件发送程序接着应该使用这个命令回应 SMTP 邮件接收程序，参数<para>表示回应的认证方式，通常是 SMTP 邮件接收程序先前提示的认证方式。
mail<SP>From:<reverse-path><CRLF>	此命令用于指定邮件发送者的邮箱地址，<reverse-path>表示发件人的邮箱地址。
rcpt<SP>To:<forword-path><CRLF>	此命令用于指定邮件接收者的邮箱地址，<forward-path>表示接收者的邮箱地址。如果邮件要发送给多个接收者，那么应使用多条 Rcpt<SP>To 命令来分别指定每一个接收者的邮箱地址。
data<CRLF>	此命令用于表示 SMTP 邮件发送程序准备开始传送邮件内容，在这个命令后面发送的所有数据都将被当作邮件内容，直至遇到“<CRLF> . <CRLF>”标识符，则表示邮件内容结束。
quit<CRLF>	此命令表示要结束邮件发送过程，SMTP 邮件接收程

	序接收到此命令后，将关闭与 SMTP 邮件发送程序的网络连接。

3. POP3 协议命令格式

POP3命令格式	说明
user<SP>username<CRLF>	user 命令是POP3客户端程序与POP3邮件服务器建立连接后通常发送的第一条命令，参数 username 表示收件人的帐户名称。
pass<SP>password<CRLF>	pass 命令是在user命令成功通过后，POP3客户端程序接着发送的命令，它用于传递帐户的密码，参数 password 表示帐户的密码。
apop<SP>name,digest<CRLF>	apop 命令用于替代user和pass命令，它以MD5 数字摘要的形式向POP3邮件服务器提交帐户密码。
stat<CRLF>	stat 命令用于查询邮箱中的统计信息，例如：邮箱中的邮件数量和邮件占用的字节大小等。
uidl<SP>msg#<CRLF>	uidl 命令用于查询某封邮件的唯一标志符，参数msg#表示邮件的序号，是一个从1开始编号的数字。
list<SP>[MSG#]<CRLF>	list 命令用于列出邮箱中的邮件信息，参数 msg#是一个可选参数，表示邮件的序号。当不指定参数时，POP3服务器列出邮箱中所有的邮件信息；当指定参数msg#时，POP3服务器只返回序号对应的邮件信息。
retr<SP>msg#<CRLF>	retr 命令用于获取某封邮件的内容，参数 msg#表示邮件的序号。
dele<SP>msg#<CRLF>	dele 命令用于在某封邮件上设置删除标记，参数msg#表示邮件的序号。POP3服务器执行dele命令时，只是为邮件设置了删除标记，并没有真正把邮件删除掉，只有POP3客户端发出quit命令后，POP3服务器才会真正删除所有设置了删除标记的邮件。
rest<CRLF>	rest 命令用于清除所有邮件的删除标记。
top<SP>msg#<SP>n<CRLF>	top 命令用于获取某封邮件的邮件头和邮件体中的前n行内容，参数msg#表示邮件的序号，参数n表示要返回邮件的前几行内容。使用这条命令以提高 Web Mail系统（通过Web站点上收发邮件）中的邮件列表显示的处理效率，因为这种情况下不需要获取每封邮件的完整内容，而是仅仅需要获取每封邮件的邮件头信息。
noop<CRLF>	noop 命令用于检测POP3客户端与POP3服务器的连接情况。
quit<CRLF>	quit 命令表示要结束邮件接收过程，POP3服务器接收到此命令后，将删除所有设置了删除标记的邮件，并关闭与POP3客户端程序的网络连接。

4. Postfix、Dovecot

Postfix 是实现了 SMTP 协议的邮件发送软件。Dovecot 是实现了 POP 和 IMOP 协议的邮件接收软件。本文采用 Postfix、Dovecot 在 centos 7 操作系统上搭建简单的邮件服务器，邮件客户端使用 Foxmail 软件。

三 实验环境

右上角选择“SMTP”，网络拓扑图如图 1 所示，三台主机与一台交换机相连，所在子网网段为“**192.168.3.0/24**“，其中一台主机为 Centos 7 系统（IP：**192.168.3.10**），用于搭建一个本地 DNS 服务器；另需两台操作系统为 WinXp 系统的主机，其中 Client1 的 IP 为 **192.168.3.4**，Client2 的 IP 为 **192.168.3.11**，用于测试邮件服务器和分析相关协议。（注：子网网段、IP 地址等信息会因具体的实验环境而不同。）

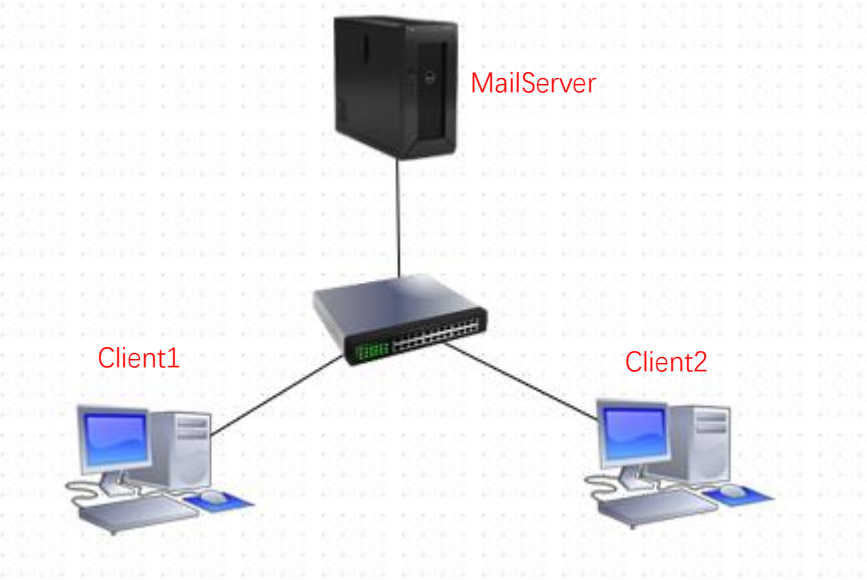


图 1

四 实验步骤

1.配置 DNS 服务器

为了使客户端 Client1 和 Client2 可以解析邮件服务器域名，实际生活中搭建邮件服务器时首先要为邮件服务器进行域名注册。考虑到我们搭建邮件服务器仅用于实验目的，本文选择在 MailServer 上配置 BIND 作为两个客户端的 DNS 服务器。请结合“DNS 实验”，完成 DNS 服务器的配置。

1）在 MailServer 上添加域名“**csuvnetwork.com**”并创建”**csuvnetwork.com.zone**”文件,在 **csuvnetwork.com.zone** 域配置文件中插入相关 MX 和 A 记录。

```
$TTL 1D
$ORIGIN csuvnetwork.com.
@       IN SOA  csuvnetwork.com. root (
                                        0       ; serial
                                        1H      ; refresh
                                        1H      ; retry
                                        1W      ; expire
                                        3H )    ; minimum
        IN      NS      dnsserver
@       IN      MX      10      mail
mail    IN      A       192.168.3.10
dnsserver       IN      A       192.168.3.10
```

2）在客户端测试 DNS 服务器是否配置成功。

```
C:\Documents and Settings\admin>ping mail.csuvnetwork.com

Pinging mail.csuvnetwork.com [192.168.3.10] with 32 bytes of data:

Reply from 192.168.3.10: bytes=32 time<1ms TTL=64
Reply from 192.168.3.10: bytes=32 time<1ms TTL=64
Reply from 192.168.3.10: bytes=32 time<1ms TTL=64
Reply from 192.168.3.10: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.3.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

2.搭建邮件服务器

1）配置 postfix

修改配置文件“**/etc/postfix/main.cf**“，修改相关参数。

```
myhostname=mail.csuvnetwork.com   # 换成邮箱服务器
mydomain=csuvnetwork.com          # 换成你的域名

#该参数含义为申请邮箱为 @$mydomain 结尾，如：user1@mail.csuvnetwork.com
myorigin = $mydomain

inet_interfaces = all                    # 设置可以接收所有域名的邮件
inet_protocols = ipv4                    # 设置 ip 协议
mydestination=$myhostname,localhost.$mydomain,localhost,$mydomain,mail.$mydomain
# 服务器地址为 192.168.3.10 这里写成 192.168.3.0
mynetworks = 192.168.3.0/24, 127.0.0.0/8
```

配置完成后启动 postfix：**systemctl start postfix**

2）配置 dovecot

Dovecot 相关配置文件有：**/etc/dovecot/dovecot.conf**、**/etc/dovecot/conf.d/10-auth.conf**、**/etc/dovecot/conf.d/10-mail.conf**、**/etc/dovecot/conf.d/10-ssl.conf**。已完成相关配置，感兴趣的同学可以打开各个配置文件查看相关配置。

启动 dovecot：**systemctl start dovecot**

3）配置 Sasl

Sasl 相关配置文件有：**/etc/sysconfig/saslauthd**、**/user/lib64/sasl2/smtpd.conf**。

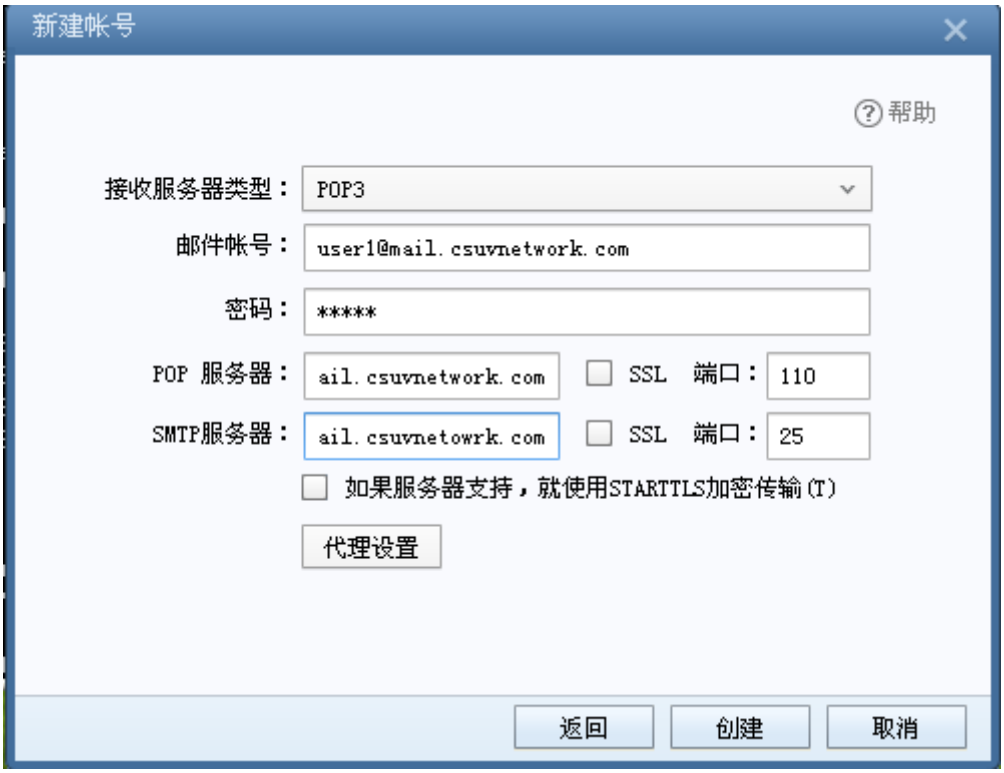
启动 sasl：**systemctl start saslauthd**

4）在 MailServer 上新增用户，用户就是你的邮箱账号，如新增用户 user，完整邮箱就是 **user@mail.csuvnetwork.com**，邮箱密码就是用户密码。此处请新增两个用户 user1 和 user2。

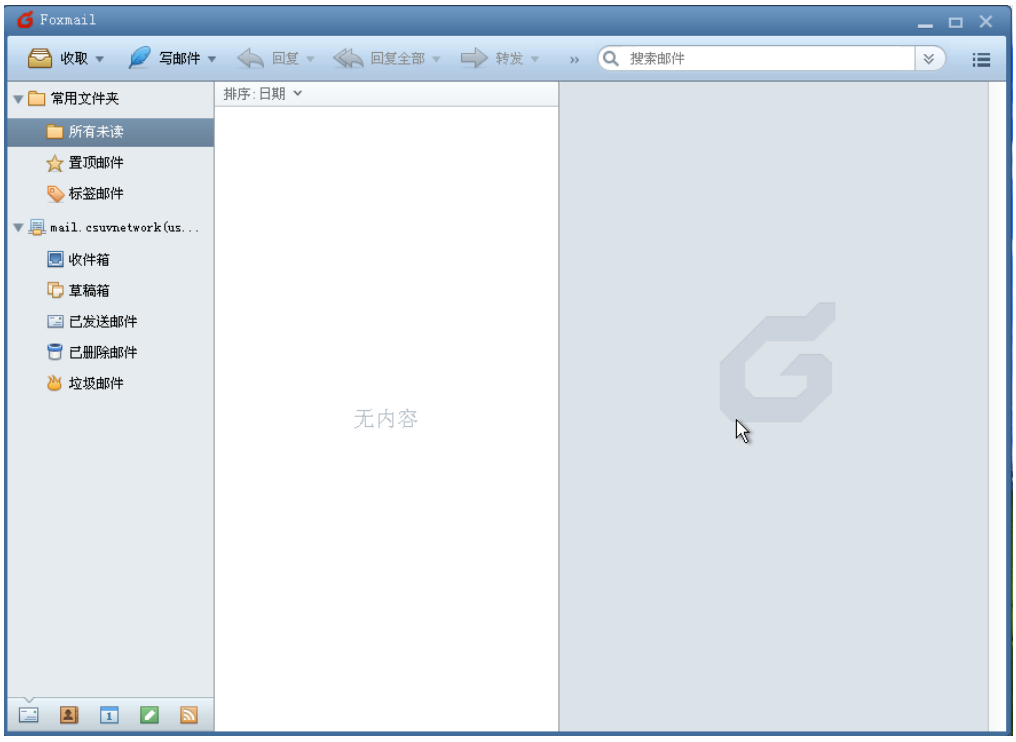
```
useradd user1    #新增用户
passwd user1    #设置密码
useradd user2    #新增用户
passwd user2    #设置密码
```

3.配置客户端

1）分别在 Client1 和 Client2 上使用 Foxmail 选择“**手动设置**“进行邮箱服务器的配置。下图以用户 user1 为例：



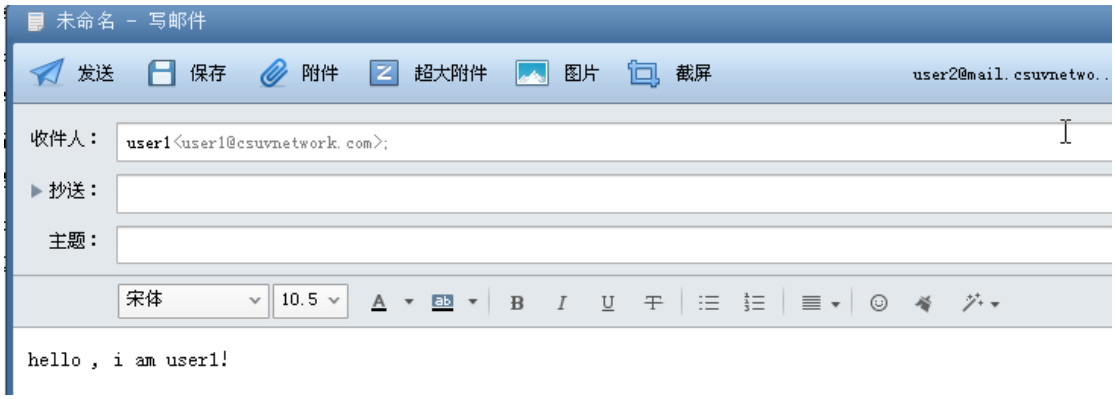
2）登录成功进入主界面。



5. 分析 SMTP

1）Client2 上打开 Wireshak 进行抓包，使用 Foxmail 向 user1@mail.csuvnetwork.com 发送

邮件。



2）分析 wirekshark 所抓取的数据包。可以看到客户端发送邮件时使用的相关命令（EHLO、AUTH、MAIL、DATA、QUIT）,请结合数据包掌握客户端发送邮件的流程并尝试分析分析各个命令的作用。

27	49.053533000	192.168.3.11	192.168.3.10	SMTP	64	C: Pass: dXNlcjI=
28	49.093113000	192.168.3.10	192.168.3.11	TCP	54	smtp > instl-boots [ACK] Seq=250 Ack=83 Win=282
29	53.018362000	192.168.3.10	192.168.3.11	SMTP	108	S: 535 5.7.8 Error: authentication failed: UGFz
30	53.018751000	192.168.3.11	192.168.3.10	SMTP	66	C: AUTH LOGIN
31	53.019249000	192.168.3.10	192.168.3.11	TCP	54	smtp > instl-boots [ACK] Seq=304 Ack=95 Win=282
32	57.023520000	192.168.3.10	192.168.3.11	SMTP	72	S: 334 VXNlcm5hbWU6
33	57.023790000	192.168.3.11	192.168.3.10	SMTP	64	C: User: dXNlcjI=
34	57.024483000	192.168.3.10	192.168.3.11	TCP	54	smtp > instl-boots [ACK] Seq=322 Ack=105 Win=28
35	57.024533000	192.168.3.10	192.168.3.11	SMTP	72	S: 334 UGFzc3dvcmQ6
36	57.024713000	192.168.3.11	192.168.3.10	SMTP	64	C: Pass: dXNlcjI=
37	57.043518000	192.168.3.10	192.168.3.11	SMTP	91	S: 235 2.7.0 Authentication successful
38	57.054784000	192.168.3.11	192.168.3.10	SMTP	105	C: MAIL FROM: <user2@mail.csuvnetwork.com> SIZE
39	57.055184000	192.168.3.10	192.168.3.11	SMTP	68	S: 250 2.1.0 Ok
40	57.055416000	192.168.3.11	192.168.3.10	SMTP	88	C: RCPT TO: <user1@csuvnetwork.com>
41	57.056626000	192.168.3.10	192.168.3.11	SMTP	68	S: 250 2.1.5 Ok
42	57.056804000	192.168.3.11	192.168.3.10	SMTP	60	C: DATA
43	57.057460000	192.168.3.10	192.168.3.11	SMTP	91	S: 354 End data with <CR><LF>.<CR><LF>
44	57.057550000	192.168.3.11	192.168.3.10	SMTP	441	C: DATA fragment, 387 bytes
45	57.098049000	192.168.3.10	192.168.3.11	TCP	54	smtp > instl-boots [ACK] Seq=442 Ack=593 Win=28
46	57.098090000	192.168.3.11	192.168.3.10	IMF	739	from: "user2@mail.csuvnetwork.com" <user2@mail.
47	57.098412000	192.168.3.10	192.168.3.11	TCP	54	smtp > instl-boots [ACK] Seq=442 Ack=1278 Win=3
48	57.109150000	192.168.3.10	192.168.3.11	SMTP	92	S: 250 2.0.0 Ok: queued as 6CC0910773DF
49	57.110472000	192.168.3.11	192.168.3.10	SMTP	60	C: QUIT

五 思考

掌握 SMTP 邮件发送协议的工作原理后，尝试在本实验中分析邮件接收协议的工作流程。