

## Instalação e configuração do serviço de e-mail Postfix.

### 1 – Histórico

O correio eletrônico [e-mail] foi uma das primeiras aplicações práticas utilizadas no início da internet e, até hoje, é uma das mais utilizadas. Qualquer pessoa que possua um endereço eletrônico pode enviar e receber mensagens através da internet. O endereço conhecido como e-mail address ou endereço de correio eletrônico possui a estrutura básica **user@host.domain** ou **user@domain**, onde *user* representa o identificador do usuário, *host* representa o nome do *host* e *domain* é o nome do domínio na internet.

O funcionamento do correio eletrônico é baseado no paradigma "store-and-forward" (estocar e redirecionar), onde os usuários envolvidos na transferência de uma mensagem não interagem diretamente entre si, mas usam aplicações encarregadas de executar e gerenciar essa transferência.

### 2 – Conceitos SMTP

SMTP vem de *Simple Mail Transfer Protocol*, que é o protocolo usado na transferência de e-mails.

Os componentes principais de um sistema de correio eletrônico são:

- **Mail User Agent (MUA)** - aplicação que interage com o usuário, responsável pela obtenção de mensagens a serem transmitidas e também retirada de mensagens recebidas. São exemplos de MUA: Alpine<sup>1</sup>, Evolution, Outlook, Eudora<sup>2</sup>.
- **Mail Transport Agent (MTA)** - aplicação responsável pelo transporte de mensagens entre os pontos envolvidos, sejam elas na rede local ou através da internet. São exemplos de MTA: sendmail<sup>3</sup>, postfix<sup>4</sup>, qmail<sup>5</sup>, exim, exchange.
- **Mail Delivery Agent (MDA)** – aplicação responsável pelo recebimento do e-mail entregue pelo MTA local, deposita o e-mail na caixa postal do usuário. O MDA tem a função básica

---

1 <http://www.washington.edu/alpine/>

2 <http://www.eudora.com/>

3 <http://www.sendmail.org/>

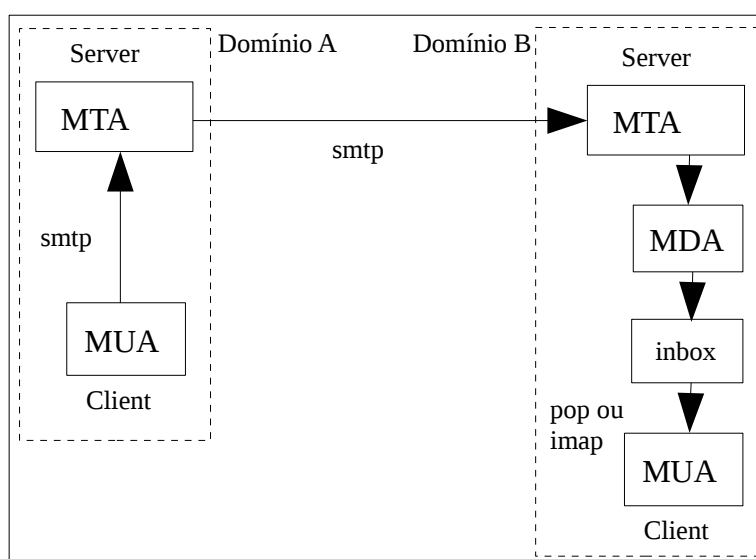
4 <http://www.postfix.org/>

5 <http://www.qmail.org/>

de filtrar o conteúdo da mensagem. São exemplos de MDA: procmail e aplicações de filtragem contra vírus e spam em geral.

- **Mail Submission Agent (MSA)** - aplicação com funcionalidade semelhante ao MTA, porém normalmente somente aceita transportar e-mails na intranet [e-mails locais] e não de outros domínios [internet]. Na prática, MSA é mais conceito que produto.
- **Mail Retrieval Agent/Mail Access Agent (MRA/MAA)** - aplicação usada pelo usuário para buscar seus e-mails depositados na Mail Box. Em muitos casos, aparece embutido [transparente] no MUA.
- **Mail Boxes** - caixas postais [inbox] onde são armazenadas as mensagens recebidas. O e-mail é depositado pelo MDA.

O princípio do funcionamento do correio eletrônico segue os seguintes passos: o remetente compõe a mensagem no seu aplicativo cliente de e-mail (agente de mensagens - MUA) e envia para o endereço eletrônico do destinatário. A mensagem é direcionada ao agente de transporte (MTA), cuja função é a entrega ou encaminhamento da mensagem à caixa postal correta do destinatário. Essa transferência é feita através do protocolo SMTP (Simple Mail Transfer Protocol). Na máquina destino, o MTA recebe a mensagem e a deixa na caixa postal correspondente do usuário.



Numa situação básica, o processo de envio/recebimento de e-mail pode ter três protocolos envolvidos:

- o **SMTP** (Simple Mail Transfer Protocol), responsável pelo envio de mensagens entre dois serviços de e-mail, que envolve pelo menos uma caixa postal de usuário [destinatário];
- o **POP** (Post Office Protocol) que gerencia o acesso às mensagens que chegaram à caixa postal do usuário de e-mail;
- o **IMAP** (Internet Message Access Protocol), que também gerencia o acesso às mensagens que chegaram à caixa postal do usuário de e-mail.

### 3 – Conceitos pop, imap, relay, spam e webmail

Existem três principais protocolos de estruturação de sistemas de e-mail: POP (Post Office Protocol), DMSP (protocolo distribuído do sistema do correio) e IMAP (protocolo de acesso de mensagem de internet).

Dos três, o POP é o mais antigo e conseqüentemente o mais conhecido. DMSP é limitado a uma única aplicação, PCMAIL, para sustentação da operação desconectada. O IMAP combina as potencialidades do POP e do DMSP, e fornece a melhor sustentação para todas as três modalidades do acesso remoto da caixa postal: offline, online e desconectado.

Outra vantagem do IMAP é a possibilidade de criar pastas [diretórios] no servidor de e-mail para poder guardar mensagens já lidas, e com isso desocupar espaço da caixa de entrada [inbox], além de permitir uma melhor organização dessas mensagens.

O RELAY é uma permissão que o serviço de e-mail concede ao usuário para poder enviar [forward] seus e-mails através desse MTA. Usualmente numa intranet, todos os usuários estão autorizados a usarem o MTA local, mas não é permitido o RELAY para usuários externos. A razão dessa limitação no uso é que, se for concedido acesso irrestrito para envio de e-mails, toda a internet irá pôr esse serviço MTA para transportar os e-mails deles, e muito dos quais passarão a enviar extensas listas de e-mail não solicitados com propaganda, que é o conceito de spam.

O Webmail permite a integração do servidor de e-mail com o servidor web, que é uma forma prática, simples, eficiente e segura de enviar/receber e-mails. Nesse caso é o servidor web que invoca o serviço IMAP para acessar a caixa postal do usuário para ler e-mail, e é também o servidor web que faz relay no MTA para envio de mensagens.

A porta TCP padrão do serviço POP é 110, do serviço IMAP é 143, e do SMTP é 25. Se forem usados os protocolos seguros, pop3s é 995, imaps é 993 e smtp seguro é 587.

### 4 – Instalação do serviço de e-mail Postfix

Essa instalação também será *standalone*.

Para descobrir se o servidor Postfix está instalado, procurar pelo seu script de inicialização em **/etc/init.d**:

```
shell# ls /etc/init.d | grep postfix
/etc/init.d/postfix
```

Se não houver saída no comando acima é indicativo de que o serviço **postfix** não está instalado. Nesse caso, instalar com o comando **yum**:

```
shell# yum install postfix
```

NOTA:

Se fosse um Ubuntu, o comando seria "**apt-get install postfix**". E nesta instalação, seria necessário usar a tecla <TAB> para navegar entre os menus para personalizar a instalação. Para o nosso caso, é melhor manter o *default* do instalador, que é "Internet".

Após instalado, verificar se existem os seguintes arquivos:

```
shell# file /etc/init.d/postfix
/etc/init.d/postfix: Bourne shell script text executable
shell# file /usr/sbin/postfix
/usr/sbin/postfix: ELF 32-bit LSB shared object, Intel 80386, version 1 (SYSV), dynamically linked (uses
shared libs), for GNU/Linux 2.6.18, stripped
shell# file /etc/postfix
/etc/postfix: directory
```

onde:

- **/etc/init.d/postfix** é o script de inicialização do serviço postfix;
- **/usr/sbin/postfix** é o executável que dará origem ao processo daemon;
- **/etc/postfix** é o diretório de configuração do serviço de e-mail postfix.

## 5 – Configuração do serviço

Primeiro vamos apenas olhar as configurações de instalação do serviço. Para isso, entrar no diretório **/etc/postfix**:

```
shell# cd /etc/postfix
shell# ls
```

Existem dois arquivos de configuração, **master.cf** e **main.cf**. Vamos dar uma olhada no arquivo de configuração "**main.cf**" com o comando **more**:

```
shell# more main.cf
```

Nesse arquivo, uma configuração importante é **inet\_interfaces**, que pelo padrão [default] de instalação está para **inet\_interfaces = localhost**. Isso significa que esse MTA somente aceita relay da interface de rede local [localhost], e desse modo não atende a nenhum cliente pela rede.

Vamos mudar essa configuração para "**inet\_interfaces = all**" para passar a aceitar relay de e-mail de qualquer origem na rede.

NOTA:

Ao invés de **all**, o correto seria definir apenas a interface de rede pela qual os clientes acessam o serviço para o envio de e-mail.

Ainda, se fosse um sistema Ubuntu o default de instalação estaria como "loopback-only", e também precisaria ser reconfigurado para **all**.

Além disso, precisa também incluir o domínio **aluno.br** para poder enviar e-mail para "aluno@aluno.br". Precisa também declarar a rede de onde o MTA aceita e-mails, que no nosso caso é 192.168.1.0/24.

Todas estas configurações são mostradas abaixo:

```
==== main.cf =====  
myorigin = aluno.br  
inet_interfaces = all  
mydestination = $myhostname, localhost.$mydomain, localhost, aluno.br  
mynetworks = 192.168.1.0/24, 127.0.0.0/8  
=====
```

Em negrito estão as configurações que foram feitas no arquivo **main.cf**.

## 6 – Inicialização e teste do serviço

Para verificar se o *daemon* está rodando, usar o comando **ps**:

```
shell# ps -ef | grep postfix
```

Se não houver saída no comando acima, indica que o processo não está rodando. Para iniciar, usar o script de inicialização:

```
shell# /etc/init.d/postfix start
Iniciando o postfix: [ OK ]
```

Agora, o comando **ps -ef | grep postfix** deverá mostrar que o serviço Postfix está rodando.

Depois disso, o scan de portas com **nmap** vai mostrar que a porta 25 também está aberta:

```
shell# nmap localhost

Starting Nmap 4.76 ( http://nmap.org ) at 2009-11-15 19:38 BRST
Interesting ports on localhost (127.0.0.1):
Not shown: 998 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
631/tcp   open  ipp

Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
```

Nesse ponto, já pode ser testado o envio de e-mail para o serviço de e-mail em localhost. O e-mail será enviado para o usuário local aluno, com e-mail **aluno@aluno.br**. Ou seja, o usuário aluno em localhost vai enviar um e-mail para a caixa postal dele mesmo em localhost.

Como já sabemos, um MTA é um dispositivo que transporta tráfego SMTP. Nesse processo, o cliente de envio MTA conecta no MTA que vai receber a mensagem, que assim é transportada de um MTA para o outro.

No nosso caso, só temos um MTA, que é o que vai receber a mensagem, então precisamos também de um agente que conecte nesse MTA para enviar a mensagem. A função de agente de envio pode ser feita usando o comando **telnet**, para conectar na porta 25 em localhost e enviar a mensagem:

```
shell# telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 localhost ESMTP Postfix
helo Postfix
250 localhost
Mail from: aluno@aluno.br
250 2.1.0 aluno@aluno.br... Sender ok
Rcpt to: aluno@aluno.br
250 2.1.5 aluno@aluno.br... Recipient ok
data
354 Enter mail, end with "." on a line by itself
Subject: teste
Apenas um teste de envio de e-mail.
.
250 2.0.0 q7QHCaPV004001 Message accepted for delivery
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

Repare que a aplicação cliente **telnet** conectou na porta 25, que é onde está o serviço de e-mail. Na cor azul aparecem as respostas do MTA Postfix em localhost. Uma vez conectado, tem que especificar quem é o destinatário [**Rcpt to: aluno@aluno.br**], quem está enviando o e-mail [**Mail from: aluno@aluno.br**] e, só depois disso, usar a palavra *data* para iniciar o conteúdo da mensagem. Para finalizar a mensagem, digitar um ponto “.” numa linha nova, dar <enter> e depois escrever *quit*.

Essa mensagem foi para a caixa postal [INBOX] do usuário aluno, em **/var/spool/mail/aluno**. Para visualizar o conteúdo da mensagem recebida, basta usar o comando **tail**:

```
shell# tail /var/spool/mail/aluno
From aluno@aluno.br Sun Aug 26 14:14:03 2012
Return-Path: <aluno@aluno.br>
Received: from localhost [127.0.0.1]
    by localhost (8.14.5/8.14.5) with SMTP id q7QHCaPV004001
    for aluno@aluno.br; Sun, 26 Aug 2012 14:13:33 -0300
Subject: teste
Message-Id: <201208261713.q7QHCaPV004001@localhost>
Date: Sun, 26 Aug 2012 14:12:36 -0300
From: aluno@aluno.br
Apenas um teste de envio de e-mail.
```

O arquivo de log do serviço de e-mail, **/var/log/maillog**, vai acusar que a mensagem foi enviada. Para visualizar esse log usar o comando **tail**:

```
shell# tail /var/log/maillog
Aug 26 14:14:03 127.0.0.1 postfix[4001]: q7QHCaPV004001: from=aluno@aluno.br, size=32, class=0, nrcpts=1, msgid=<201208261713.q7QHCaPV004001@localhost>, proto=SMTP, daemon=MTA, relay=localhost [127.0.0.1]
Aug 26 14:14:03 127.0.0.1 postfix[4015]: q7QHCaPV004001: to=aluno@aluno.br, ctladdr=aluno@aluno.br (1000/1000), delay=00:00:30, xdelay=00:00:00, mailer=local, pri=30313, dsn=2.0.0, stat=Sent
```

NOTA:

Num Ubuntu, o arquivo de log é **/var/log/mail.log**.

## 7 – Envio de e-mail de uma máquina para outra

O teste agora consiste em enviar um e-mail de uma máquina para outra. neste exercício, no endereço IP 192.168.1.11 está o MTA cliente, e no endereço 192.168.1.10 está o MTA que vai receber a mensagem.

Mas para haver transporte de mensagens de uma máquina para a outra, precisaria um registro MX no serviço de nomes, pois o endereço e-mail é do tipo **user@domain** ou **user@host.domain**.

Então, para esse envio, seria necessário primeiro configurar um serviço de nomes e declarar num registro MX que o host em 192.168.1.10 hospeda o serviço de e-mail. Porém, isso daria algum trabalho adicional e tiraria o foco, que é apenas testar o envio de e-mail.

Uma alternativa ao serviço de e-mail com registro MX é declarar no arquivo **/etc/hosts** das duas máquinas os seus respectivos nomes [hostnames] e endereços IP. Desse modo, mesmo sem serviço de nomes entre essas duas máquinas poderá ser usado nomes de hosts, pois os arquivos **/etc/hosts** traduzem esses nomes para endereços IP.

Vamos agora declarar os hostnames e endereços IP dessas duas máquinas em seus respectivos arquivos **/etc/hosts**.

O formato no arquivo **/etc/hosts** consiste de entradas para IPs e hostnames, desse modo basta declarar nesse arquivo o endereço IP e o hostname da máquina que hospeda o serviço MTA e também o hostname e endereço IP da máquina cliente.

E para saber qual o nome do host, basta usar o comando **hostname**:



```
shell# hostname  
host10
```

Na máquina cliente, no endereço 192.168.1.11, o comando **hostname** retorna que seu nome é **host11**.

Então, nas máquinas 192.168.1.10 e 192.168.1.11, precisa editar o arquivo **/etc/hosts** e incluir as duas seguintes linhas de configuração:

```
===== /etc/hosts =====  
192.168.1.10      host10  
192.168.1.11      host11  
=====
```

Após, na máquina 192.168.10, testar com o comando **ping** para verificar se a máquina que hospeda o serviço MTA responde pelo nome **host10**, e também se o cliente responde pelo nome **host11**:

```
shell# ping host10  
PING host10 (192.168.1.10) 56(84) bytes of data.  
64 bytes from host10 (192.168.1.10): icmp_seq=1 ttl=58 time=11.5 ms  
64 bytes from host10 (192.168.1.10): icmp_seq=2 ttl=58 time=16.4 ms  
^C  
shell# ping host11  
PING host11 (192.168.1.11) 56(84) bytes of data.  
64 bytes from host11 (192.168.1.11): icmp_seq=1 ttl=58 time=11.5 ms  
64 bytes from host11 (192.168.1.11): icmp_seq=2 ttl=58 time=16.4 ms  
^C
```

NOTA:

Para cessar o ping, pressionar simultaneamente as teclas control "<ctrl>" e "c" [<ctrl>c].

Na máquina cliente, 192.168.1.11, também deve ser testado se **host10** e **host11** respondem ao **ping**:

```
shell_cliente$ ping host10
PING host10 (192.168.1.10) 56(84) bytes of data.
64 bytes from host10 (192.168.1.10): icmp_seq=1 ttl=58 time=12.3 ms
64 bytes from host10 (192.168.1.10): icmp_seq=2 ttl=58 time=15.7 ms
^C
shell_cliente$ ping host11
PING host11 (192.168.1.11) 56(84) bytes of data.
64 bytes from host11 (192.168.1.11): icmp_seq=1 ttl=58 time=10.9 ms
64 bytes from host11 (192.168.1.11): icmp_seq=2 ttl=58 time=16.1 ms
^C
```

**NOTA:**

Se não responder ao **ping** pelo hostname, ou a máquina está inacessível ou a entrada no **/etc/hosts** está errada.

Conforme fizemos no exemplo anterior, para enviar o e-mail são será usada uma aplicação MUA, mas sim conectar direto na máquina que hospeda o serviço MTA com **telnet** na porta 25. A diferença é que agora esse acesso e envio do e-mail será feito pelo cliente **host11** no IP 192.168.1.11.

O e-mail será enviado pelo usuário aluno em **host11** para aluno no MTA em **host10**. No e-mail [Mail from:] é aluno@aluno.br e no [Rcpt to:] também é aluno@aluno.br.

```
shell_cliente$ telnet host10 25
Trying 192.168.1.10...
Connected to host10.
Escape character is '^'.
220 host10 ESMTP Postfix
helo Postfix
250 host10
Mail from: aluno@aluno.br
250 2.1.0 Ok
Rcpt to: aluno@aluno.br
250 2.1.5 Ok
data
354 End data with <CR><LF>.<CR><LF>
Subject: Outro teste
Outro teste, agora o e-mail é enviado de 192.168.1.11 para 192.168.1.10
.
250 2.0.0 Ok: queued as 533F98C09BE
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

Na cor **azul** aparecem as respostas do MTA Postfix em **host10** [192.168.1.10].

Essa mensagem foi enviada para a caixa postal [INBOX] do usuário aluno, em **/var/spool/mail/aluno** na máquina MTA [**host10**, 192.168.1.10]. Nessa máquina, para visualizar o conteúdo da mensagem recebida, basta usar o comando **tail**:

```
shell# tail /var/spool/mail/aluno
From aluno@aluno.br Sun Aug 26 23:00:27 2012
Return-Path: <aluno@aluno.br>
X-Original-To: aluno@aluno.br
Delivered-To: aluno@aluno.br
Received: from host11 (host11 [192.168.122.175])
    by pinguim.localdomain (Postfix) with SMTP id 533F98C09BE
    for <aluno@aluno.br>; Sun, 26 Aug 2012 22:57:59 -0300 (BRT)
Subject: Outro teste

Outro teste, agora o e-mail é enviado de 192.168.1.11 para 192.168.1.10
```

E o arquivo de log **/var/log/maillog** na máquina que hospeda o serviço de e-mail [192.168.1.10] vai acusar que a mensagem foi recebida. Para visualizar esse log, usar o comando **tail**:

```
shell# tail /var/log/maillog
Aug 26 22:59:39 pinguim postfix/smtpd[9277]: 533F98C09BE:
client=host11[192.168.1.11]
Aug 26 23:00:27 pinguim postfix/cleanup[9294]: 533F98C09BE: message-id=<>
Aug 26 23:00:27 pinguim postfix/qmgr[9059]: 533F98C09BE: from=<aluno@aluno.br>,
size=269, nrcpt=1 (queue active)
Aug 26 23:00:27 pinguim postfix/local[9304]: 533F98C09BE: to=<aluno@aluno.br>,
relay=local, delay=148, delays=148/0.01/0/0.05, dsn=2.0.0, status=sent (delivered to
mailbox)
Aug 26 23:00:27 pinguim postfix/qmgr[9059]: 533F98C09BE: removed
Aug 26 23:00:29 pinguim postfix/smtpd[9277]: disconnect from host11[192.168.1.11]
```

NOTA:

Num Ubuntu, o arquivo de log é **/var/log/mail.log**.

## 8 – Leitura de e-mail com pop3 e imap

Uma vez o e-mail depositado na caixa de entrada do usuário, o modo usual de acessar estas mensagens é através do serviço pop3 ou imap.

Estes dois serviços costumam ser instalados juntos, pela instalação do pacote **dovecot**.

Para descobrir se o pacote **dovecot** está instalado, procurar pelo seu script de inicialização em **/etc/init.d**:

```
shell# ls /etc/init.d | grep dovecot
/etc/init.d/dovecot
```

Se não houver saída no comando acima é indicativo de que o serviço **dovecot** não está instalado. Nesse caso, instalar com o comando **apt-get**:

```
shell# apt-get install dovecot-core dovecot-pop3d dovecot-imapd
```

Num CentOS, o comando para instalar é "**yum install dovecot**".

Depois de instalado, precisa configurar os arquivos **/etc/dovecot/dovecot.conf**, **/etc/dovecot/conf.d/10-auth.conf** e **/etc/dovecot/conf.d/10-mail.conf**.

Em **dovecot.conf** precisa incluir a seguinte linha:

```
=== /etc/dovecot/dovecot.conf =====
protocols = pop3 imap # nao precisa incluir pop3s nem imaps, pois jah sao default
=====
```

Em **10-auth.conf**, precisa **descomentar e corrigir a seguinte linha**:

```
=== /etc/dovecot/conf.d/10-auth.conf =====
disable_plaintext_auth = no # para poder trabalhar com protocolos inseguros
=====
```

Em **10-mail.conf**, precisa **descomentar e corrigir a seguinte linha**:

```
=== /etc/dovecot/conf.d/10-mail.conf =====
mail_location = mbox:~/mail:INBOX=/var/spool/mail/%u # INBOX em /var/spool/mail
=====
```

Depois disso é necessário reiniciar o serviço **dovecot**:

```
shell# /etc/init.d/dovecot restart
Reiniciando o dovecot: [ OK ]
```

Agora, os serviços **pop3**, **pop3s**, **imap** e **imaps** devem estar rodando e com as suas respectivas portas de serviço abertas. Um scan com **nmap** vai mostrar:

```
shell# nmap localhost
```

```
Starting Nmap 4.76 ( http://nmap.org ) at 2009-11-15 21:45 BRST
```

```
Interesting ports on localhost (127.0.0.1):
```

```
Not shown: 998 closed ports
```

PORT	STATE	SERVICE
25/tcp	open	smtp
110/tcp	open	pop3
143/tcp	open	imap
631/tcp	open	ipp
993/tcp	open	imaps
995/tcp	open	pop3s

```
Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds
```

Para testar o acesso para os serviços inseguros, usar o telnet na respectiva porta (110 e 143).

```
shell_cliente$ telnet localhost 110
```

```
Trying 127.0.0.1...
```

```
Connected to localhost.
```

```
Escape character is '^['.
```

```
+OK Dovecot ready.
```

```
user aluno
```

```
+OK
```

```
pass uninove
```

```
+OK Logged in.
```

```
list
```

```
+OK 1 messages:
```

```
1 488
```

```
.
```

```
quit
```

```
+OK Logging out.
```

```
Connection closed by foreign host.
```

Na cor **azul** aparecem as respostas do serviço **pop3 Dovecot**. Além dos comandos acima, o serviço **pop3** suporta também **retr**, **top** e **dele**. Por exemplo:

**retr 1** => mostra a mensagem de número 1;

**top 1** => mostra o topo (cabeçalho apenas) da mensagem de número 1;

**dele 1** => exclui a mensagem de número 1.

Para testar o **imap**, acessar a porta 143 e seguir a sintaxe de comando deste protocolo:

```

shell_cliente$ telnet localhost 143
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
* OK [CAPABILITY IMAP4rev1 LITERAL+ SASL-IR LOGIN-REFERRALS ID ENABLE IDLE
STARTTLS AUTH=PLAIN] Dovecot ready.
a1 login aluno uninove
a1 OK [CAPABILITY IMAP4rev1 LITERAL+ SASL-IR LOGIN-REFERRALS ID ENABLE IDLE
SORT SORT=DISPLAY THREAD=REFERENCES THREAD=REFS THREAD=ORDEREDSUBJECT
MULTIAPPEND URL-PARTIAL CATENATE UNSELECT CHILDREN NAMESPACE UIDPLUS
LIST-EXTENDED I18NLEVEL=1 CONDSTORE QRESYNC ESEARCH ESORT SEARCHRES
WITHIN CONTEXT=SEARCH LIST-STATUS SPECIAL-USE BINARY MOVE] Logged in
a2 list "" "*"
* LIST (\NoInferiors \UnMarked) "/" saved-messages
* LIST (\NoInferiors \UnMarked) "/" sent-mail
* LIST (\HasNoChildren) "/" INBOX
a2 OK List completed.
a3 examine INBOX
* FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
* OK [PERMANENTFLAGS ()] Read-only mailbox.
* 1 EXISTS
* 0 RECENT
* OK [UIDVALIDITY 1487080252] UIDs valid
* OK [UIDNEXT 2] Predicted next UID
a3 OK [READ-ONLY] Examine completed (0.000 secs).
a4 logout
* BYE Logging out
a4 OK Logout completed.
Connection closed by foreign host.

```

Na cor **azul** aparecem as respostas do serviço **imap Dovecot**. Além dos comandos acima, o serviço **imap** suporta também **fetch**, **create**, **copy** e **rename**, entre outros.

Para testar o acesso usando protocolo seguro, usar o comando **openssl**. Por exemplo, para acessar o serviço pop3s na porta 995:

```

shell_cliente$ openssl s_client -connect localhost:995
CONNECTED(00000003)
depth=0 OU = IMAP server, CN = imap.example.com, emailAddress = postmaster@example.com
verify error:num=18:self signed certificate
verify return:1
...
+OK Dovecot ready.
user aluno
+OK
pass uninove10
+OK Logged in.
quit
+OK Logging out.
closed

```

Depois de conectado, é só seguir a sintaxe do pop3.

Mas para ler as mensagens de e-mail direto no servidor SMTP, uma ferramenta bem útil é o **alpine** (ou, simplesmente, pine). Para saber se está instalado, comandar:

```
shell# rpm -aq | grep -i alpine
alpine-2.20-2.el7.x86_64
```

Num Ubuntu, para checar se está instalado usar o comando "**dpkg -L alpine**".

Caso não estivesse instalado, bastaria comandar:

```
shell# yum install alpine
```

Num Ubuntu, o comando para instalar é "**apt-get install alpine**".

Para acessar as mensagens, comandar **alpine** como aluno:

```
shell_cliente$ alpine
```

Isso vai abrir a caixa de mensagens do usuário no shell:

```
ALPINE 2.20  MAIN MENU  Folder: INBOX  1 Message

?  HELP          - Get help using Alpine
C  COMPOSE MESSAGE - Compose and send a message
I  MESSAGE INDEX  - View messages in current folder
L  FOLDER LIST   - Select a folder to view
A  ADDRESS BOOK  - Update address book
S  SETUP         - Configure Alpine Options
Q  QUIT          - Leave the Alpine program

For Copyright information press "?"

? Help          OTHER CMDS [ListFldrs] PrevCmd  RelNotes
0 [ListFldrs]  NextCmd      KBlock
```

*Tela de texto do alpine.*