

FACULDADE DE IMPERATRIZ

Eduardo Yuji Wada  
Everson Santos Araujo  
Matheus Santos Saraiva

## **Servidor de E-mail**

estudo de caso: Postfix + popa3d

Imperatriz (MA)

2004

Eduardo Yuji Wada  
Everson Santos Araujo  
Matheus Santos Saraiva

## **Servidor de E-mail**

estudo de caso: Postfix + popa3d

Trabalho apresentado à matéria de Redes de Computadores com finalidade de obtenção de nota de pesquisa em Serviços de Rede no Curso de Sistemas de Informação da Faculdade de Imperatriz.

Orientador: Prof. Ivaldo Monteiro Lobato

Imperatriz (MA)

2004

“Este trabalho é dedicado a todos aqueles que lutaram por uma vida melhor, por um mundo livre de LAG, NetSplit e pela criação da Banda Larga.”

“ A diferença entre arte e ciência é que a ciência é o que entendemos o suficiente para explicar ao computador. Arte é todo o resto.”

Donald Knuth

## RESUMO

Neste trabalho estaremos demonstrando os protocolos de troca de E-mail e tentaremos criar um servidor de e-mail completamente compatível com os padrões de protocolo existentes, mostraremos como funcionam os protocolos e os testaremos em alguns programas específicos escolhidos.

Estaremos buscando ferramentas para criação de um sistema padrão de troca de e-mails e tentaremos focalizar em alguns aspectos relevantes e não cobertos pelos padrões dos protocolos de cada uma das ferramentas utilizadas.

**PALAVRAS-CHAVE:** e-mail, servidor, smtp, pop, postfix, popa3d, redes de computadores.

## ABSTRACT

In this paper we will show our work on make a fully RFC compliant Mail Server, showing how Mail protocols should work and testing this protocols against programs we will ensure that we have a protocol safe Mail system for the overall production enviroment.

This is a research in potentialy using some tool for mail management in a standard enterprise, we will focus on some specific feature off the programs we shall use but only for reference, and only the ones that are not covered by the RFC.

**KEYWORDS:** e-mail, server, smtp, pop, postfix, popa3d, computer networks.

## LISTA DE FIGURAS

Figura 1 - Evolution, exemplo de cliente de e-mail.....	11
Figura 2 - Arquitetura SMTP.....	12
Figura 3 - Um exemplo de conexão SMTP.....	14
Figura 4 - Um exemplo de sessão POP3.....	17
Figura 5 - Postfix recebendo mensagem.....	18
Figura 6 - Postfix enviando de mensagem.....	19
Figura 7 - Exemplo de manipulação da conexão no popa3d.....	22

# SUMÁRIO

RESUMO.....	5
ABSTRACT.....	6
LISTA DE FIGURAS.....	7
1 E-mail.....	9
1.1 O que é e-mail?.....	9
1.2 Clientes de e-mail.....	10
2 Protocolos de e-mail.....	12
2.1 SMTP.....	12
2.2 POP3.....	15
3 Estudo de caso.....	18
3.1 Postfix.....	18
3.2 popa3d.....	21
3.3 Instalação e configuração dos serviços.....	22
CONCLUSÃO.....	26
REFERÊNCIAS.....	27



## 1. E-mail

### 1.1.O que é e-mail

Assim como cartas foram utilizadas a muitos anos para repor o serviço de um mensageiro, hoje em dia vemos o e-mail repondo o serviço de cartas.

Da forma mais simples possível, e-mail é uma mensagem, ou carta, eletrônica enviada de um computador à outro. É um serviço de mensagens utilizado para troca de informações entre pessoas e estações de trabalho de forma a dinamizar a comunicação, utilizando meios de comunicação eletrônicos.

Funciona do mesmo modo que uma mensagem de carta, porém utiliza como meio de disseminação um ambiente de rede de computadores. Seu corpo pode ser dividido em três partes: remetente, destinatário e mensagem. A mensagem pode conter textos e arquivos de qualquer tipo anexado ou embutido no texto.

O remetente é a pessoa que escreveu o e-mail, destinatário é a pessoa que deve receber o e-mail e a mensagem congrega todas as demais informações que fazem parte de um e-mail, tais como: data de envio, título, mensagem, arquivos adicionais.

Como qualquer sistema eletrônico, o e-mail utiliza protocolos específicos de troca de dados que descrevem como se deve realizar a comunicação entre os serviços e os clientes e de servidor de e-mail para servidor de e-mail.

Um e-mail se caracteriza pela atribuição de um usuário e um servidor como endereço de envio e recepção, no modelo usuario@servidor, que descreve usuário “em” (*at*) servidor, e serve para direcionar a mensagem para o servidor correto para que este possa direcionar ao usuário.

## 1.2. Clientes de e-mail

Cliente de e-mail é qualquer programa apto a conectar a um servidor de e-mail e entender seu protocolo para que possa realizar a troca de informações com o mesmo, devido a isso foram criadas as padronizações de protocolos de e-mail, para que se possam criar clientes e servidores compatíveis com os padrões especificados.

Um cliente de e-mail é conhecido como MUA (*Mail User Agent* – Agente Usuário de E-mail), existem inúmeros clientes de e-mail atualmente no mercado que oferecem desde a simples troca de e-mails até filtros de mensagem, filtros anti-spam, dentre inúmeras outras ferramentas de organização e indexação das mensagens e ajuda ao usuário.

Não sendo específico a nenhum cliente, podemos dizer que todos devem ter no mínimo as funcionalidades de enviar e receber mensagens.

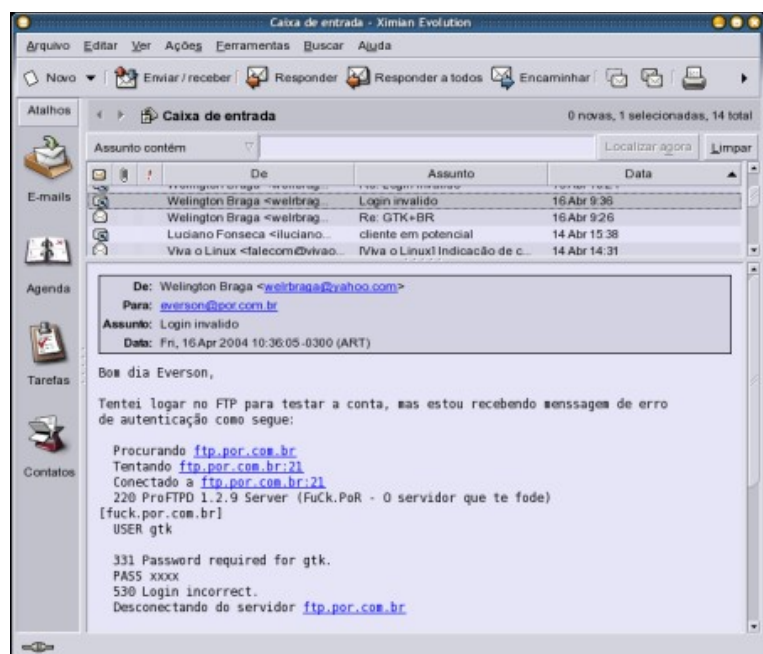


Figura 1 - Evolution, exemplo de cliente de e-mail

## 2. Protocolos de e-mail

### 2.1.SMTP

O protocolo SMTP padrão que prevê a troca de mensagens entre clientes e servidores e entre os servidores até o servidor do destinatário. Utiliza a porta 25 como padrão de troca de mensagens.

O protocolo consiste de um sistema simples de troca de caracteres, independente de um sistema particular de transmissão requerendo apenas um canal de dados que possam ser ordenados.

Foi descrito inicialmente em Agosto de 1982 na *RFC 821* sob o nome de *Simple Mail Transfer Protocol* (Protocolo de Transferência de E-mail Simples), atualmente utiliza a *RFC 2821* de 2001 sob o mesmo nome.

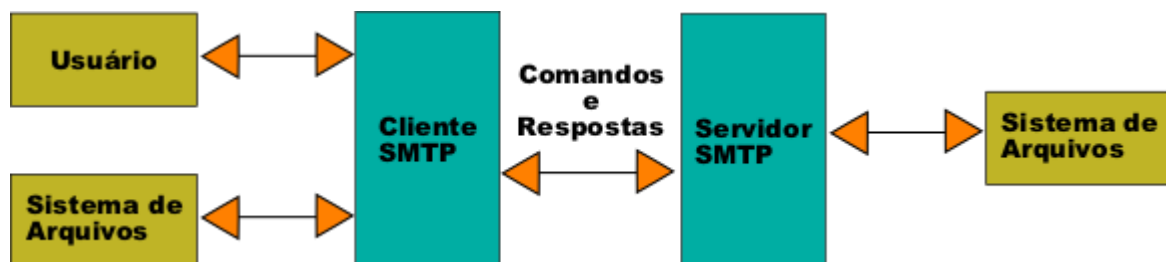


Figura 2 - Arquitetura SMTP

Quando um cliente tem uma mensagem a ser transmitida, ele estabelece um canal de comunicação de duas-vias com o servidor SMTP, a responsabilidade de um cliente SMTP é de transmitir a mensagem a um ou mais servidores ou reportar uma falha caso não seja possível ser entregue a mensagem.

Após realizar a conexão, deve-se iniciar a transação de e-mail, esta transação consiste de uma série de comandos para especificar origem e destino da mensagem, bem como todo seu conteúdo. A cada comando enviado, o servidor deve responder com comando aceito, comandos adicionais são requeridos ou com erro temporário ou permanente.

Alguns dos controles descritos no protocolo são:

**EHLO** - Início da transação e requisição dos módulos suportados pelo host.

**MAIL FROM:<origem>** - Indica de onde vem o e-mail, endereço de origem.

**RCPT TO:<destino>** - Indica o endereço final da mensagem.

**DATA** – Inicia o corpo da mensagem.

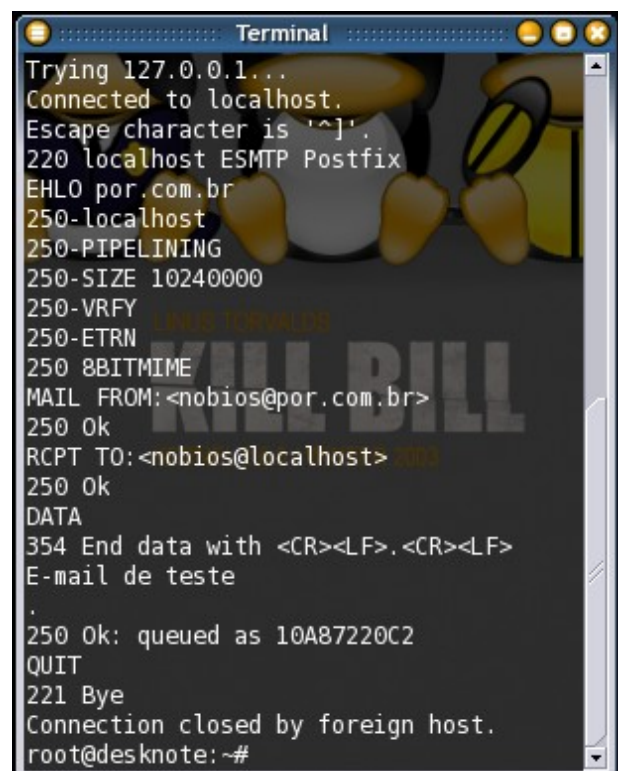
**RSET** – Descarta a mensagem atual.

**QUIT** – Fecha a conexão.

Extensões:

**VERFY usuario@servidor** - Realiza a verificação da existência do usuário.

**EXPN lista** – Expande uma lista de e-mails indicando todos que participam desta.



```

Terminal
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 localhost ESMTP Postfix
EHLO por.com.br
250-localhost
250-PIPELINING
250-SIZE 10240000
250-VERFY
250-ETRN
250 8BITMIME
MAIL FROM:<nobios@por.com.br>
250 Ok
RCPT TO:<nobios@localhost>
250 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
E-mail de teste
.
250 Ok: queued as 10A87220C2
QUIT
221 Bye
Connection closed by foreign host.
root@desknote:~#

```

Figura 3 - Um exemplo de conexão SMTP

## 2.2.POP3

Este protocolo prevê a recuperação das mensagens de um usuário que estão salvas em um servidor remoto para sua estação de trabalho. Utiliza a porta 110 para realizar a comunicação.

Assim como o SMTP se baseia na troca de simples caracteres de comandos que não dependem de um sistema particular de transmissão.

Este já se encontra na versão 3 que foi descrita na *RFC 1939* em Maio de 1996, sob o nome de *Post Office Protocol* – Protocolo de Escritório do Correio.

Este protocolo tem a intenção de prover acesso dinâmico a e-mails de um usuário que um servidor de e-mails está mantendo para este usuário. Não tem a intenção de prover manipulação remota destes e-mails; normalmente o e-mail é baixado e logo após deletado do servidor.

O protocolo consiste em quatro estados bem distintos, que são estado de espera, autorização, transação e atualização. Toda conexão bem sucedida deve passar por todos esses estados.

Um servidor POP3 deve aguardar uma conexão. Quando um cliente realizar uma conexão TCP, o servidor deve enviar uma saudação e passar para o estado de autenticação. Após a autenticação ter sido feita, a sessão entra no modo de transação onde o cliente tem acesso a todos os comandos cobertos pelo protocolo POP3. Toda resposta de um servidor POP3 deve ser +OK ou -ERR com ou sem texto adicional, indicando uma resposta positiva ou negativa ao último comando. Ao final da conexão, o servidor deve então atualizar os e-mails do usuário realizando os passos que forem necessários para isso.

Alguns dos controles descrito nesse protocolo são:

**STAT** – Estatus atual das mensagens, retorna o número de mensagens e o tamanho total da caixa de correio em octetos.

**LIST** – Retorna uma listagem de todas as mensagens com o tamanho de cada uma em octetos.

**RETR n** – Retorna toda a mensagem. Onde n deve se referir a uma mensagem na caixa de correio que não pode estar marcada para remoção.

**DELE n** – Marca uma mensagem para remoção, a mensagem só é realmente removida no estado de atualização antes do servidor fechar a conexão.

**RSET** – Retorna a caixa de correio para a situação anterior à sessão atual.

**QUIT** – Fecha a sessão.



```

Terminal
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^'.
+OK
USER nobios
+OK
PASS XXXXXXXX
+OK
STAT
+OK 1 454
LIST
+OK
1 454
RETR 1
+OK
Return-Path: <nobios@por.com.br>
X-Original-To: nobios@localhost
Delivered-To: nobios@localhost.desknote.nobios.net
Received: from por.com.br (localhost [127.0.0.1])
    by localhost (Postfix) with ESMTP id 10A87220C2
    for <nobios@localhost>; Mon, 14 Jun 2004 16:35:38 -0400 (AMT)
Message-Id: <20040614203538.10A87220C2@localhost>
Date: Mon, 14 Jun 2004 16:35:38 -0400 (AMT)
From: nobios@por.com.br
To: undisclosed-recipients:;
E-mail de teste
DELE 1
+OK
RSET
+OK
QUIT
+OK
Connection closed by foreign host.
root@desknote:~#

```

Figura 4 - Um exemplo de sessão POP3

Extensões:

**UIDL n** – Retorna o identificador único de cada mensagem.

**TOP msg n** – Requisita as n primeiras linhas da mensagem especificada.

### 3. Estudo de caso

#### 3.1. Postfix

Postfix é um servidor de troca de e-mails, que foi criado para ser uma alternativa ao *sendmail*, pelo Wietse Venema, que é um especialista em segurança e pesquisador na IBM.

Postfix é rápido, fácil de administrar e seguro, enquanto ao mesmo tempo se mantém compatível com o *sendmail*. Apesar de seu exterior ser parecido com o *sendmail* seu interior é completamente diferente.



Figura 5 - Postfix recebendo mensagem

Quando uma mensagem chega ao Postfix, inicialmente ela fica guardada em um agendamento de mensagens recebidas. O e-mail pode chegar através da rede ou através de algum processo local de envio de e-mail.

Os e-mails que chegam pela rede são pré-processados pelo servidor smtpd ou qmqpd, estes servidores removem o encapsulamento de seus protocolos, realizam algumas checagens nos conteúdos das mensagens e entregam todas as partes do e-mail separadamente ao servidor de limpeza (cleanup).

Os e-mails locais são recebidos com um programa compatível com o *sendmail* e então enviados ao agendamento de e-mails (maildrop), então o servidor de coleta (pickup) recolhe estes e-mail realiza algumas checagens nos conteúdos da mensagem e entrega ao servidor de limpeza todas as partes do e-mail separadamente, que são o remetente, o destinatário e a mensagem.

O servidor de limpeza implementa os processamentos finais antes da entrega do e-mail ao agendamento de e-mails recebidos. Este adiciona e transforma cabeçalhos quando necessário, transforma endereços através de reescrita (trivial-rewrite) e faz checagens no conteúdo da mensagem. Então o servidor de limpeza adiciona o resultado em um único arquivo no agendamento de mensagens recebidas (incoming) e notifica o gerenciador de agendamento.

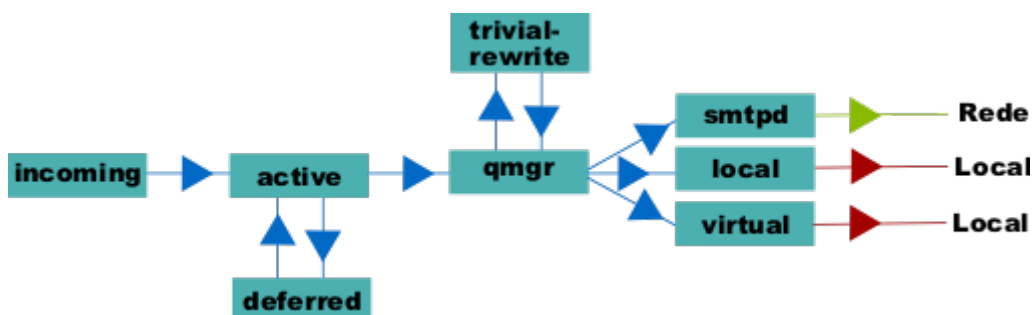


Figura 6 - Postfix enviando de mensagem

Assim que uma mensagem chega ao agendamento de mensagens recebidas o próximo passo é a entrega da mensagem, então o postfix realiza novamente checagens e agendamentos de envio.

O gerenciador de agendamento (qmgr) é o coração do sistema de entrega de e-mails, ele realiza a conexão com o agente de entrega adequado para cada mensagem que ele retira do agendamento para a realização da entrega.

O gerenciador de agendamento mantém uma pequena pequena lista de agendamento ativo (active) com as mensagens que ele retirou do agendamento de entrega. Este agendamento ativo funciona como uma janela de tamanho fixo em potencialmente grandes agendamentos de entrega, este



tamanho limitado o impede de ficar sem memória disponível durante momentos de intenso movimento.

O gerenciador de agendamento mantém um agendamento separado de mensagens que ele não pode enviar, são mensagens que ficam marcadas para envio posterior (deferred). Esta estratégia é criada para evitar um agendamento de mensagens muito grande, que dificultaria a entrega de mensagens que já podem ser entregues instantaneamente.

O serviço de reescrita trivial (trivial-rewrite) resolve cada o endereço de cada um dos endereços da mensagem de acordo com sua classe local ou remota de endereço. Além disso, pode receber informações adicionais de roteamento e verifica a mudança de e-mails de usuários.

O cliente smtp verifica uma lista de servidores de troca de mensagens para o endereço do destinatário, organiza a lista por preferência e tenta cada servidor até encontrar algum que o responda. Então realiza o encapsulamento da mensagem contendo endereço de origem, de destino e a mensagem propriamente dita como requerido pelo protocolo SMTP.

O agente de entrega local entende vários estilos de caixa de mensagem e realiza suas operações paralelamente, exceto para um mesmo usuário, e ainda pode simplesmente delegar o envio da mensagem a outro programa como o *procmail*.

O agente de entrega virtual pode realizar a entrega de mensagens para diferentes domínios em um mesmo servidor, o que o torna especialmente utilizado em sistemas de máquinas com endereços compartilhados.

### 3.2.popa3d

O servidor de acesso à caixa de mensagens popa3d tem como principal objetivo a segurança, logicamente não adicionando extensões ao protocolo POP3, sendo então limitado por ele.

Uma das limitações básicas do protocolo POP3 é a autenticação de usuário realizada através de texto-puro, ou seja, a senha do usuário é transmitida como texto e pode ser interceptada no meio do caminho entre o usuário e o servidor sem que este perceba essa interceptação. Para evitar este problema, foram criadas algumas extensões no protocolo POP3 para o envio criptografado da senha, no entanto, este sistema requer que a senha seja mantida no servidor em forma de texto-puro, o que não o torna mais seguro que o envio de texto-puro já que os usuários permanecem vulneráveis caso se tenha acesso a este arquivo de senhas. O sistema de segurança se baseia em um servidor de e-mail que tenha capacidade de ser seguro e realizar seu trabalho, então ele não torna a comunicação segura, ele é um servidor seguro em sua forma de realizar seu trabalho no servidor.

O funcionamento interno do popa3d define que ele deve iniciar como root e realizar o tratamento das informações como um usuário sem privilégios tendo como diretório base um local sem informações. O acesso ao sistema de autenticação é feito por dois processos divididos que rodam em paralelo para realizar a troca de informações e comunicarem-se entre si.

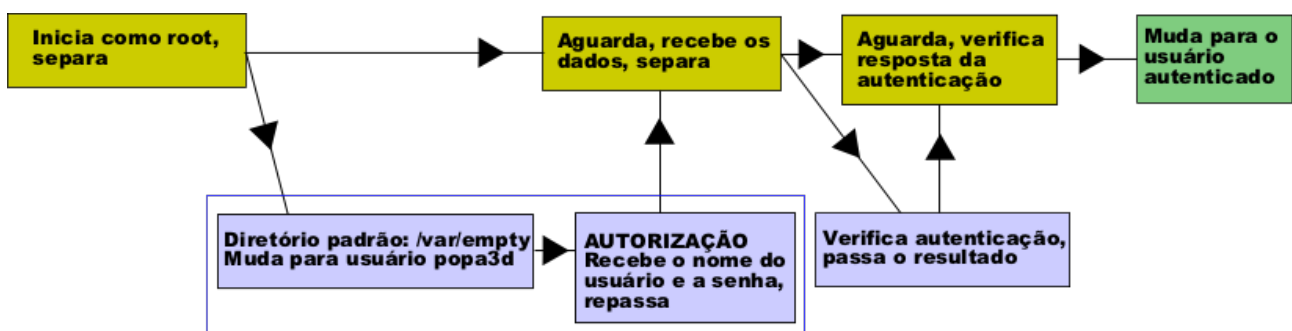


Figura 7 - Exemplo de manipulação da conexão no popa3d

### 3.3. Instalação e configuração dos serviços

Para realizar a instalação e configurações dos serviços aqui mencionados, utilizamos a distribuição Slackware Linux, escolhida devido a sua arquitetura e reconhecido sistema simples de configuração e uso, demonstrando grande preocupação com a segurança e sendo bastante estável. Utilizamos um sistema atualizado, comumente chamado de *current* no pacote desta distribuição, baseado na árvore de 22 de maio de 2004, kernel do Linux utilizado foi o 2.6.5.

Utilizamos uma rede com três computadores em um mini-laboratório, utilizamos como servidor um Athlon XP 2000+, com 256 mb de DDR RAM, disco Maxtor ATA 133 120 Gb, placa de rede Realtek RTL8139D, e dois clientes: um rodando Slackware Linux 9.1 e outro com Kurumin Linux 3.2. Já como programas cliente de e-mail utilizamos o K-mail, Ximian Evolution e Mozilla Mail.

Para realizar a configuração do servidor, primeiro verificamos a disponibilidade dos pacotes dos softwares escolhidos pré-compilados para o Slackware, encontramos ambos os servidores o que facilitou a realização da instalação através do gerenciamento de pacotes do próprio Slackware.

#### a) Instalando o postfix

Baixamos o pacote do postfix do Linux Packages (<http://linuxpackages.net>) que é um site de pacotes de software destinados ao slackware, instalamos com o root utilizando o comando:

```
# installpkg [nome-do pacote]
```

Logo após fomos ao arquivo de configuração do postfix (/etc/postfix/main.cf) e modificamos os valores mínimos para que ele possa funcionar, que são:

```
myhostname = endereço local completo  
mydomain = domínio para receber e-mails
```

Com essa simples configuração já foi possível fazer com que o postfix executasse e permite-se com que trocássemos e-mails entre os usuários, logicamente funcionando com todos os outros valores de configuração em formato padrão.

Para podermos conferir os valores de configuração do postfix, digitamos o comando `postconf` que nos retornou todos os valores de todas as configurações, porém como estávamos apenas testando em um laboratório bem específico, somente realizamos a verificação dos valores e não precisamos realizar nenhuma modificação.

#### b) Instalando o popa3d

A instalação do popa3d foi ainda mais simples, o popa3d tem um pacote incluso na própria distribuição Slackware, então fizemos o download do mesmo no mirror oficial ([ftp.slackware.com](http://ftp.slackware.com)) e o instalamos com o comando de instalação de pacotes, anteriormente apresentado.

Preferimos utiliza-lo como um serviço rodando através do *inetd* (internet super-server) que é o sistema que aguarda conexões vindas da rede para determinadas portas, decide qual o serviço que deve ser chamado para responder à requisição e o invoca, seu arquivo de configuração se encontra em `/etc/inetd.conf`.

Como o popa3d já faz parte do Slackware Linux, ele já vem pré-configurado para aceita-lo,

apenas precisamos remover o comentário da linha correta nesse arquivo de configuração e está tudo pronto:

```
pop3 stream tcp nowait root /usr/sbin/tcpd /usr/sbin/popa3d
```

Para realizarmos as configurações nos clientes, primeiro criamos uma pequena rede entre os mesmos e a demos o nome de testefacimp.grp, utilizamos a classe 192.168.0.x e configuramos o servidor como 192.168.0.1 (server.testefacimp.grp) e demos para cada um dos clientes um ip qualquer nessa faixa e um nome único.

Configuramos os clientes com servidor de envio e recebimento com o ip/nome escolhido para o servidor e então criamos 3 usuários na máquina servidor: local, maquina1 e maquina2. Em cada uma das máquinas clientes utilizamos um usuário diferente e deixamos o usuário local para receber e enviar mensagens através do próprio servidor.

Todos os testes ocorreram como o esperado, podendo realizar troca de e-mails entre todas as máquinas/usuários sem o menor problema, podemos também enviar e-mails para endereços reais na internet, já que o servidor estava conectado à internet, apenas não podemos receber e-mails da internet já que não tínhamos um domínio completamente válido para nos identificar na rede.

## CONCLUSÃO

O trabalho de pesquisa foi de grande importância para verificarmos as teorias que aprendemos na aula sobre protocolos e redes em um ambiente prático, além de nos capacitar a utilizar ferramentas produtivas de implementação de serviços de e-mail.

Aprendemos muito sobre como um protocolo se estrutura, e como se verifica a troca de informações entre diversas máquinas analisando os protocolos e os encapsulamentos realizados entre os serviços, nos tornando mais aptos a entender o funcionamento de redes de computadores em especial o acesso a recursos e serviços.

## REFERÊNCIAS

KLENSIN, J. RFC 2821 - Simple Mail Transfer Protocol, Abril 2001.

MYERS, J; ROSE, M. RFC 1939 - Post Office Protocol - Version 3.

TANNENBAUM, Andrew S. Redes de computadores: tradução [ds 3. ed. Original], Rio de Janeiro, Editora Campus, 3ª edição, 1997.

GASPARY, Luciano Paschoal. Redes de Computadores, Porto Alegre, 1999.

The Postfix Home Page, <http://www.postfix.org> acessado em 29/05/2004

popa3d a tiny POP3 daemon, <http://www.openwall.com/popa3d>. acessado em 29/05/2004