

Postfix – SMTP Relay e Entrega Local com Fetchmail

By [admin](#) on 9 de setembro de 2010 in [Linux](#)

1 Flares Twitter 0 Facebook 1 Google+ 0 LinkedIn 0 Email -- Filament.io 1 Flares [×](#)

Muitas vezes temos a necessidade de montar um servidor de email local, seja para uma melhor administração, entrega local mais rápida, filtragem e controle de mensagens, entre muitas outras opções.

Mas o grande problema é que muitos dos provedores atuais , não permitem relay em seus servidores, ou ainda pior , não fazem redirecionamento do MX. Se não o bastante, nós precisamos de um endereçamento IP fixo e que não esteja em nenhuma black list (o que é difícil com os endereços de algumas operadoras).

Acaba por aumentando muito o custo em manter um servidor local.

Mas existe uma opção barata e simples para esse problema. O postfix poderá fazer relay em um provedor, mesmo que esse solicite autenticação. Podemos configurar o postfix para autenticar em um SMTP remoto, de forma que as mensagens saiam pelo próprio provedor, quando forem destinados a alguém fora do domínio. Mas se a mensagem for para um pessoa do mesmo domínio, ele fara uma entrega local, sem necessidade de enviar para fora da rede, evitando gargalos com emails grandes.

Desta forma as mensagens sairam com um IP que não estará em nenhuma blacklist, teremos o reverso do MX, e evitaremos envio de mensagens para fora do mesmo dominio, desperdiçando banda.

Para usarmos um SMTP local, será necessário configurarmos também um POP Local, de forma que os usuários possam buscar suas mensagens localmente. Mas ai vem a dúvida. E as mensagens externas , que foram recebidas pelo provedor ?

Neste caso poderemos usar o fetchmail para buscar mensagens no provedor e entregar localmente, para usuários cadastrados.

Podemos trabalhar de duas formas:

1 – Dominio Local e Dominio Externo : Neste caso temos um dominio local, por exemplo, @local.net e um dominio externo, como @dominio.com.br. Faremos um mapeamento dos usuários locais para os usuários externos. Esse caso é interessante , se temos pessoas com emails do dominio, que não estão fisicamente no mesmo local, por exemplo uma filial.

O postfix tratará o @local.net, como dominio local, e o dominio.com.br, como dominio externo.

2 – Dominio Local é igual a Dominio Externo: Neste segundo caso , usamos o mesmo dominio internamente e externamente. É muito interessante se não houver usuários alocados em outros locais fisicamente.

Vamos botar as mãos na massa.

Instalando os pacotes Necessários

O primeiro passo a ser feito é a instalação dos pacotes necessários para que consigamos realizar a tarefa de envio de mensagens.

No Debian , os pacotes são : `libsasl2-modules` , `libsasl2-2`, `sasl2-bin`, `postfix`.

Configuração do Postfix

Após instarmos os pacotes, iremos configurar o postfix. Para este exemplo usarei o dominio chamado *statointerno.net*.

O primeiro item a ser configurado é o próprio dominio de email, que poderá ser feito utilizando o arquivo */etc/mailname*. Neste arquivo deverá ser colocado o dominio interno, neste caso , *statointerno.net*.

```
# echo 'statointerno.net' >/etc/mailname
```

O arquivo */etc/postfix/main.cf*, também terá algumas alterações, veremos abaixo :

```
smtp_use_tls = yes
```

```
smtp_sasl_auth_enable = yes
```

```
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
```

```
smtp_sasl_security_options = noanonymous
```

```
smtp_sasl_tls_security_options = noanonymous
```

```
smtp_generic_maps = hash:/etc/postfix/generic
```

```
relay = [smtp.dominio.com.br]
```

Nas configurações acima , estamos habilitando o uso da autenticação através do smtp para o nosso servidor. Ou seja, ele fará uso do sasl, para autenticar no servidor configurado em relay. No item `smtp_sasl_password_maps` temos o usuario e senha que serão utilizados para autenticar em tal dominio, em `smtp_generic_maps` o mapeamento dos usuários interno para o dominio externo, que neste caso , utilizei o **dominio.com.br**, e ainda temos o próprio relay, que informa qual é o servidor externo de smtp, que será utilizado para fazer o relay.

Alguns itens mais comuns devem ser configurados, como `mynetworks`, `interfaces`, etc...
Veja abaixo:

```
mynetworks = 127.0.0.1 192.168.0.0/24
```

```
mydestination = localhost, statointerno.net, localhost.statointerno.net,  
webmail.statointerno.net
```

```
inet_interfaces = all
```

Nestes itens estamos informando quais redes poderão fazer relay neste servidores. Em nosso caso somente redes internas e localhost. Em mydestination, informamos quais domínios serão considerados locais, ou seja, serão entregues localmente e por último em que interface o postfix estará ouvindo solicitações.

A parte de configuração do main.cf terminou, mas não ainda a configuração do postfix. Em seu arquivo foram colocados arquivos externos, como mapeamento de usuários, usuário utilizado para autenticar, etc. Esses arquivos deverão ser criados.

Habilitando sasl e postfix para autenticação

No arquivo main.cf, fizemos referência a um arquivo que conterá usuário e senha para o item *relay*. Em nosso caso o arquivo foi o /etc/postfix/sasl_passwd, informado em smtp_sasl_password_maps. Deveremos então editar esse arquivo e informar usuário e senha, que será utilizado para autenticar no smtp. É importante o uso da conta principal no domínio externo, afim de evitar problemas futuros. O formato do conteúdo do arquivo, será:

```
[servidor smtp] <tab>usuario:senha
```

Note que é inserido servidor, em seguida um TAB, para posteriormente colocarmos o usuário e senha. Neste exemplo usaremos o usuário *andre@dominio.com.br* com senha igual a *123456*. Vejamos a edição do arquivo.

```
#vim /etc/postfix/sasl_passwd
```

```
[smtp.dominio.com.br]         andre@dominio.com.br:123456
```

```
#chmod 600 /etc/postfix/sasl_passwd
```

Se dermos uma olhada no arquivo main.cf, poderemos ver que nesta entrada de arquivo, informamos o endereço da seguinte deste arquivo da seguinte forma : **hash:/etc/postfix/sasl_passwd**. O que quer dizer, que este arquivo é uma mapa binário, de forma que deveremos transforma-lo com o comando postmap. Mas não esqueça de alterar as permissões, para garantir um mínimo de segurança.

```
#postmap /etc/postfix/sasl_passwd
```

Após executar esse comando, no diretório /etc/postfix, será criado um arquivo chamado sasl_passwd.db, que será usado pelo postfix.

O próximo passo é criar o arquivo de mapeamento dos usuários internos para o externo. Através do item `smtp_generic_maps` informamos ao postfix para usar um hash (um mapa binário) do arquivo `/etc/postfix/generic`. Seu formato é :

usuario@dominiointerno.net usuario@dominioexterno.com.br

Onde no primeiro campo informaremos o nome de usuário criado no Linux e o domínio interno, e no segundo campo o usuário do provedor, junto ao seu domínio. Neste caso usarei um usuário interno chamado `tstato`, e usuário externo somente `stato`. Vejamos como ficará.

tstato@statointerno.net stato@dominio.com.br

É interessante notar que não utilizei o mesmo usuário usado para autenticar no smtp, o `andre@dominio.com.br`, mas poderia usá-lo sem nenhum problema. Deverá ser feito isso para cada usuário existente.

Após a criação destes, o próximo passo é compilar o hash novamente com o comando `postmap`.

#postmap /etc/postfix/generic

Quase pronto, bastando agora habilitar o serviço `sasl` para uso da autenticação. O arquivo que habilitará ele é o `/etc/default/saslauthd`. Deveremos editá-lo e habilitar para iniciar automaticamente.

#vim /etc/default/saslauthd

...

START=yes

...

Após alterado esse item, o nosso servidor estará pronto para fazer já autenticação e relay no servidor externo.

#/etc/init.d/saslauthd restart

#/etc/init.d/postfix restart

Configurando o POP com o fetchmail

O `fetchmail` será utilizado por baixar as mensagens dos usuários diretamente no provedor e entregar localmente para o pop local. Desta forma já sabemos então que devemos ter um POP local e instalar o `fetchmail`. O pacote de POP há ser utilizado, fica a critério do administrador, já que temos muitos softwares que fazem isso. Neste exemplo usarei o `solid-pop3d`, que é muito fácil configurar, tão fácil, que depois de instalado , só precisamos reiniciar o serviço de `inet.d`, ou `xinet.d` dependendo da distribuição.

#apt-get install fetchmail solid-pop3d

A configuração do fetchmail consiste em configurar um arquivo, informando o servidor POP que será usado, seguido dos nomes de usuarios e senha deste servidor, e o nome do usuário local onde serão entregues tais mensagens. As configurações do fetchmail poderão ficar no diretório pessoal do usuário no arquivo .fetchmailrc, ou poderá ser criado um único arquivo em /etc/fetchmailrc, para todos os usuários. Caso utiliza individualmente, deverá executar várias instancias no fetchmail manualmente ou via crontab. Já no segundo caso basta utilizar o serviço do fetchmail, habilitando-o em /etc/default.

Vamos ver primeiramente o formato deste arquivo. Neste exemplo usarei o fetchmailrc no /etc/.

```
#vim /etc/fetchmailrc
```

```
set no syslog
```

```
set logfile "/var/log/fetchmaillog"
```

```
set postmaster "root"
```

```
set daemon 30
```

```
set bouncemail
```

```
poll pop.dominio.com.br proto POP3
```

```
user stato@dominio.com.br password "abc123" is tstato here
```

```
user andre@dominio.com.br password "123456" is tstato here
```

```
user maria@dominio.com.br password "abc123" is mariap here
```

```
keep
```

Devemos alterar as permissões do mesmo, para garantir também a segurança.

```
#chmod 600 /etc/fetchmailrc
```

O primeiro item set logfile, já é auto explicativa. Informa onde ficarão os logs do fetchmail. Este log é muito interessante para verificação de problemas, pois nele vão constar todas solicitações e respostas do servidor POP.

No item "set daemon 30" é especificado o tempo em segundo para que o servidor interno (Fetchmail) vá buscar mensagens no POP. Neste caso 30 segundos.

A opção `set postmaster`, especifica o usuário local, que receberá mensagens, caso quando da entrega não seja encontrado o usuário pré-configurado. Isto ocorrerá em caso de deleção de usuário local, e não removido do arquivo do `fetchmail`.

O item `boucemail`, é utilizado justamente para mensagens de erros serem encaminhadas para o `postmaster`, em nosso caso o `root`.

O item `pool` informa o endereço do servidor POP seguido do protocolo. Caso não informado o protocolo e tentará auto configurar.

Em seguida colocamos as configurações para cada usuário. O item `user` especifica o usuário do provedor, ou seja, o domínio externo seguido de sua senha. O próximo item na linha é `"is usuariointerno here"`, que informa em que caixa postal será entregue, ou seja, o nome do usuário local.

Não há problema algum em usar várias vezes o mesmo usuário local, como mostrado no exemplo.

No exemplo as mensagens de `stato` e `andre` serão entregues localmente para `tstato`. Enquanto as mensagens de `maria@dominio.com.br` serão entregues localmente para `mariap`.

Para habilitarmos o `fetchmail` na inicialização devemos editar o arquivo `/etc/default/fetchmail`.

E muito cuidado com o `log`, já que o mesmo será salvo pelo usuário `fetchmail` em `/var/log`, onde o mesmo não tem permissão. Talvez seja interessante alterar esse local.

```
#vim /etc/default/fetchmail
```

```
START_DAEMON=yes
```

Depois disso basta iniciarmos:

```
#!/etc/init.d/fetchmail restart
```

Se for o caso, é possível fazer essa configuração em um determinado arquivo de usuário, por exemplo no diretório `/root/.fetchmailrc`. Ao fazermos isso, devemos garantir que o `fetchmail` seja iniciado na inicialização. Para isso, podemos fazer uso do arquivo `/etc/rc.local`:

```
#vim /etc/rc.local
```

```
...
```

```
fetchmail -f /root/.fetchmailrc
```

```
exit
```

Observações

Todas mensagens baixadas do POP serão deletadas do servidor, caso queira deixar uma cópia da mesma, pode-se usar a opção **keep fetchmail**, da seguinte forma.

```
user andre@dominio.com.br password "123456" keep fetchmail is stato here
```

Desta forma o fetchmail deixará uma cópia da mensagem no servidor.

Um outro item interessante a usar é um WEBMAIL. Costumo usar o openwebmail, que é bem fácil de configurar.

Fazendo uso dele, podemos facilmente inserir um novo endereço no DNS do provedor, algo como **webmailinterno.dominio.com.br**, apontando para o endereço do servidor, ou firewall da empresa.

Se for um servidor totalmente interno, é necessário fazer DNAT para o mesmo. Imagine que o servidor interno seja 192.168.0.100, e a interface de internet (a que está configurado o IP Público) seja eth0. A regra no iptables seria algo como :

```
iptables -t nat -A PREROUTING -i eth0 -p tcp -dport 80 -j DNAT -to 192.168.0.100
```

Caso não possua um ip fixo, ainda sim é possível utilizar DDNS, como no-ip, e outros. E no DNS do provedor usar o endereço do DDNS. Por exemplo, vamos supor que não tenho endereço fixo, uso speedy PPPOE. Desta forma faço um cadastro no **www.no-ip.com**, e me registro como smtpstato.sytes.net. Faço download do programa de atualização , configuro-o e pronto. O DDNS está pronto. Sempre que tentar resolver smtpstato.sytes.net ele indicará o endereço IP do meu speedy.

Agora vou ao provedor e cadastro um novo endereço para webmailinterno.dominio.com.br apontando para smtpstato.sytes.net.

É bem provável que seja necessário usar um apelido. No caso de DNS um CNAME. Uma entrada parecida com a abaixo :

```
webmailinterno.dominio.com.br.      IN CNAME smtpstato.sytes.net.
```

E um outro item que pode ser interessante é o uso, do mesmo domínio , tanto interno como externo, com o mesmo nome de domínio. Desta forma as configurações dos clientes usariam o endereço de email real. Esta solução, somente funcionaria se não houver usuários externos, ou seja, filiais ou outros locais físicos. O arquivo generic teria um formato similar a este.

```
user stato@dominio.com.br password "abc123" is stato here
```

E obviamente o /etc/mailname , teria o mesmo nome de domínio, em nosso caso o dominio.com.br. E não esqueça de configurar também o mydestination como dominiolocal.

Utilizando esse formato , somente o que não é destinado a dominio.com.br, seria feito relay externo no provedor.

E por fim, para configurar o cliente, basta apontar o POP e SMTP do cliente para o servidor postfix, via endereço IP.

CONCLUSÃO

Com o uso do postfix (diga-se de passagem um ótimo servidor SMTP), sasl, pop, fetchmail, temos um servidor local, totalmente capaz. O mesmo será responsável por entregar mensagens no provedor, evitando que os email internos sejam considerados spam, e ainda busca as mensagens no provedor para entrega local, evitando tráfego entre usuários internos.

Uma solução open, simples, e pode ser implementada até em máquinas virtuais , devido a seu baixo consumo.

Com essa solução as vezes evitamos até o gasto adicional de aumento de banda, quando emails entre usuários internos tomam muita banda.

Espero que aproveitem.