Mail Server com Postfix

Este artigo descreve bem detalhadamente a configuração de um servidor de email no CentOS. Precisei fazer isso e não obtive muita informação referente a uma instalação de um servidor de email mais robusto no CentOS, visto que é uma distribuição mais recente. Então depois de ralar bastante, segue minha contribuição ao pessoal do VOL e a alguém que venha precisar! Configurações iniciais

Preparando o sistema

Primeiramente instale o CentOS com uma configuração bem enxuta, sem instalar banco MySQL, Apache, DNS, nada.

Todos os comandos abaixo foram feitos como root; os que não foram feitos utilizando o root irei informar.

Se você usa uma conexão que precisa passar por um proxy, configure-o em **/etc/profile** colocando as seguintes linhas no final do arquivo:

```
export http_proxy=http://user:senha@IPSERVIDOR:PORTA
export ftp_proxy=http://user:senha@IPSERVIDOR:PORTA
```

Adicione repositório DAG

```
# vi /etc/yum.repos.d/dag.repo
```

Com este conteúdo

```
[dag]
name=Dag RPM Repository for Red Hat Enterprise Linux
baseurl=http://apt.sw.be/redhat/el$releasever/en/$basearch/dag
gpgcheck=1
enabled=0
```

E o repositório rpmforge:

```
# wget http://apt.sw.be/packages/rpmforge-release/rpmforge-release-
0.3.6-1.el5.rf.i386.rpm
# rpm -qlip rpmforge-release-0.3.6-1.el5.rf.i386.rpm
# vi /etc/yum.repos.d/rpmforge.repo
```

E altere a linha:

```
enabled=1
```

para:

```
enabled=0
```

Se achar conveniente, atualize todos os pacotes já instalados.

Instalando os pacotes

Instale os seguintes pacotes utilizando o yum:

```
# yum install mysql-devel httpd mysql-server pkgconfig rpm-build php
gcc cyrus-sasl-devel php-mysql php-mbstring openssl-devel zlib-devel
pcre-devel openldap-devel libtool postgresql-devel gdbm-devel pam-
devel expect gcc-c++ gamin-devel openldap-servers unrar
```

Baixar o courier e maildrop:

```
# mkdir $HOME/downloads
# cd $HOME/downloads
# wget http://surfnet.dl.sourceforge.net/sourceforge/courier/courier-
authlib-0.58.tar.bz2
# wget http://surfnet.dl.sourceforge.net/sourceforge/courier/maildrop-
2.0.2.tar.bz2
```

Criar os diretórios para poder compilar os arquivos baixados no item anterior:

```
# mkdir $HOME/rpm
# mkdir $HOME/rpm/SOURCES
# mkdir $HOME/rpm/SPECS
# mkdir $HOME/rpm/BUILD
# mkdir $HOME/rpm/SRPMS
# mkdir $HOME/rpm/RPMS
# mkdir $HOME/rpm/RPMS/i386
# echo "%_topdir $HOME/rpm" >> $HOME/.rpmmacros
```

Compile o courier-authlib e instale-o:

```
# rpmbuild -ta courier-authlib-0.58.tar.bz2
# cd $HOME/rpm/RPMS/i386
# rpm -ivh courier-authlib-mysql-0.58-1.i386.rpm
# rpm -ivh courier-authlib-0.58-1.i386.rpm
# rpm -ivh courier-authlib-devel-0.58-1.i386.rpm
```

Compile o maildrop e instale-o:

```
# cd $HOME/downloads
# rpmbuild -ta maildrop-2.0.2.tar.bz2
# cd $HOME/rpm/RPMS/i386
# rpm -ivh maildrop-2.0.2-1.i386.rpm
```

Para compilar o courier-imap é necessário estar logado com um usuário que não seja o root. Então, caso não exista, crie um outro usuário, logue com este e execute o passo 5 novamente. Em seguida rode:

```
# su - usuário
# mkdir $HOME/downloads
# cd $HOME/downloads
# wget http://surfnet.dl.sourceforge.net/sourceforge/courier/courier-
imap-4.1.1.tar.bz2
# rpmbuild -ta courier-imap-4.1.1.tar.bz2
# su -
# cd /home/usuário/rpm/RPMS/i386
# rpm -ivh courier-imap-4.1.1-1.i386.rpm
```

Baixar e instalar o Postfix:

```
# wget
ftp://ftp.pbone.net/mirror/ftp.centos.org/4.4/centosplus/i386/RPMS/pos
tfix-2.2.10-1.RHEL4.2.mysql_pgsql.c4.i386.rpm
# rpm -ivh postfix-2.2.10-1.RHEL4.2.mysql_pgsql.c4.i386.rpm
```

Para verificar se o Postfix tem suporte ao mysql, dê o comando:

```
# postconf -m
```

Instalar MySQL

Configurar MySQL para inicializar no boot:

```
# chkconfig --levels 235 mysqld on
```

Iniciar o MySQL:

```
# /etc/init.d/mysqld start
```

Configurar a senha do MySQL:

```
# mysqladmin -u root password suasenha
```

Agora vamos configurar o phpMyAdmin:

```
# wget http://prdownloads.sourceforge.net/phpmyadmin/phpMyAdmin-
2.10.0.2-all-languages.tar.gz?download
# tar -zxvf phpMyAdmin-2.10.0.2-all-languages.tar.gz
# mv phpMyAdmin-2.10.0.2-all-languages /var/www/html/phpmyadmin
# cd /var/www/html/phpmyadmin
# cp config.sample.inc.php config.inc.php
# vi config.inc.php
```

E altere as linhas para o usuário cadastrado no item 3:

```
$cfg['Servers'][$i]['controluser'] = 'root';
$cfg['Servers'][$i]['controlpass'] = 'suasenha';
```

Nesta linha, entre com a frase secreta... pode ser qualquer coisa:

```
$cfg['blowfish_secret'] = '';
```

E altere esta linha para http:

```
$cfg['Servers'][$i]['auth_type'] = 'http';
```

Configurar apache para inicializar no boot:

```
# chkconfig --levels 235 httpd on
```

Iniciar o Apache:

```
# /etc/init.d/httpd start
```

Para acessar o phpMyAdmin, digite no navegador:

http://localhost/phpmyadmin

Será pedido usuário (root) e senha (suasenha do item 3).

Criando o banco do Postfix no MySQL

Criando o banco mail:

```
# mysqladmin -u root -p create mail
```

e entre com a senha do root do mysql - (suasenha)

Agora criaremos o usuário mail_admin, que será o usuário que administrará o banco mail. Execute as queries abaixo no prompt do MySQL ou no phpmyadmin.

PS: Caso use o phpmyadmin, não o utilize no konqueror, utilize outro navegador.

```
GRANT SELECT, INSERT, UPDATE, DELETE ON mail.* TO
'mail_admin'@'localhost' IDENTIFIED BY 'sua_senha';
GRANT SELECT, INSERT, UPDATE, DELETE ON mail.* TO
'mail_admin'@'localhost.localdomain' IDENTIFIED BY 'sua_senha';
FLUSH PRIVILEGES;

USE mail;

CREATE TABLE domains ( domain varchar(50) NOT NULL, PRIMARY KEY
(domain) ) TYPE=MyISAM;

CREATE TABLE forwardings ( source varchar(80) NOT NULL, destination
TEXT NOT NULL, PRIMARY KEY (source) ) TYPE=MyISAM;

CREATE TABLE users ( email varchar(80) NOT NULL, password varchar(20)
NOT NULL, quota INT(10) DEFAULT '10485760', PRIMARY KEY (email) )
TYPE=MyISAM;

CREATE TABLE transport ( domain varchar(128) NOT NULL default '',
transport varchar(128) NOT NULL default '', UNIQUE KEY domain (domain)
) TYPE=MyISAM;
```

Configurar Postfix para se conectar no MySQL

Criaremos 6 arquivos para a conexão com as tabelas do banco mail. No lugar de "sua_senha", coloque a senha criada na página 2 item 3.

```
vi /etc/postfix/mysql-virtual_domains.cf

user = mail_admin
password = sua_senha
dbname = mail
query = SELECT domain AS virtual FROM domains WHERE domain='%s'
hosts = 127.0.0.1
vi /etc/postfix/mysql-virtual_forwardings.cf
```

```
user = mail_admin
password = sua_senha
dbname = mail
query = SELECT destination FROM forwardings WHERE source='%s'
hosts = 127.0.0.1
vi /etc/postfix/mysql-virtual_mailboxes.cf

user = mail_admin
password = sua_senha
dbname = mail
query = SELECT CONCAT(SUBSTRING_INDEX(email,'@',-
1),'/',SUBSTRING_INDEX(email,'@',1),'/') FROM users WHERE email='%s'
hosts = 127.0.0.1
vi /etc/postfix/mysql-virtual_email2email.cf

user = mail_admin
password = sua_senha
dbname = mail
query = SELECT email FROM users WHERE email='%s'
hosts = 127.0.0.1
vi /etc/postfix/mysql-virtual_transports.cf

user = mail_admin
password = sua_senha
dbname = mail
query = SELECT transport FROM transport WHERE domain='%s'
hosts = 127.0.0.1
vi /etc/postfix/mysql-virtual_mailbox_limit_maps.cf

user = mail_admin
password = sua_senha
dbname = mail
query = SELECT quota FROM users WHERE email='%s'
hosts = 127.0.0.1
```

Mudando as permissões dos arquivos criados:

```
# chmod o= /etc/postfix/mysql-virtual_*.cf
# chgrp postfix /etc/postfix/mysql-virtual_*.cf
```

Agora criaremos o grupo e usuário v_mail:

```
# groupadd -g 5000 vmail
# useradd -g vmail -u 5000 vmail -d /home/vmail -m
```

Configurando o postfix -> execute estas linhas de comando:

```
# postconf -e 'myhostname = server1.example.com'
# postconf -e 'mydestination = server1.example.com, localhost,
localhost.localdomain'
# postconf -e 'mynetworks = 127.0.0.0/8'
# postconf -e 'transport_maps = proxy:mysql:/etc/postfix/mysql-
virtual_transports.cf'
```

Contas virtuais:

```
# postconf -e 'virtual_alias_domains ='
```

```
# postconf -e 'virtual_alias_maps = proxy:mysql:/etc/postfix/mysql-
virtual_forwardings.cf, mysql:/etc/postfix/mysql-
virtual_email2email.cf'
# postconf -e 'virtual_mailbox_domains =
proxy:mysql:/etc/postfix/mysql-virtual_domains.cf'
# postconf -e 'virtual_mailbox_maps = proxy:mysql:/etc/postfix/mysql-
virtual_mailboxes.cf'
# postconf -e 'virtual_mailbox_base = /home/vmail'
# postconf -e 'virtual_uid_maps = static:5000'
# postconf -e 'virtual_gid_maps = static:5000'
# postconf -e 'virtual_create_maildirsize = yes'
# postconf -e 'virtual_mailbox_extended = yes'
# postconf -e 'virtual_mailbox_limit_maps =
proxy:mysql:/etc/postfix/mysql-virtual_mailbox_limit_maps.cf'
# postconf -e 'virtual_mailbox_limit_override = yes'
# postconf -e 'virtual_maildir_limit_message = "The user you are
trying to reach is over quota."'
# postconf -e 'virtual_overquota_bounce = yes'
# postconf -e 'proxy_read_maps = $local_recipient_maps $mydestination
$virtual_alias_maps $virtual_alias_domains $virtual_mailbox_maps
$virtual_mailbox_domains $relay_recipient_maps $relay_domains
$canonical_maps $sender_canonical_maps $recipient_canonical_maps
$relocated_maps $transport_maps $mynetworks
$virtual_mailbox_limit_maps'
# postconf -e 'inet_interfaces = all'
```

SASL:

```
# postconf -e 'smtp_sasl_auth_enable = yes'
# postconf -e 'smtpd_sasl_auth_enable = yes'
# postconf -e 'smtpd_sasl_security_options = noanonymous'
# postconf -e 'broken_sasl_auth_clients = yes'
# postconf -e 'smtpd_recipient_restrictions =
permit_sasl_authenticated, reject'
```

Agora vamos configurar o certificado SSL, caso você deseje usar. Eu particularmente preferi não utilizar, pois comercialmente desgasta muito o cliente, mas pra quem quiser utilizar, segue como configurar:

```
# cd /etc/postfix
# openssl req -new -outform PEM -out smtpd.cert -newkey rsa:2048 -
nodes -keyout smtpd.key -keyform PEM -days 365 -x509
Seu Pais --> Country Name (e.g., "BR")
Seu Estado --> State or Province Name
Sua Cidade --> Enter your City.
Sua Empresa --> Enter your Organization Name (e.g., the name of your
company).
Seu Departamento --> Enter your Organizational Unit Name (e.g. "CPD").
Seu Dominio --> Enter the Fully Qualified Domain Name of the system
(e.g. "teste.net").
Seu Email --> Enter your Email Address
```

Com isso foi criado o arquivo smtpd.key. Vamos mudar as permissões dele:

```
# chmod o= /etc/postfix/smtpd.key
```

e adicione estas linhas em /etc/postfix/main.cf:

```
smtpd_use_tls = yes
smtpd_tls_cert_file = /etc/postfix/smtpd.cert
smtpd_tls_key_file = /etc/postfix/smtpd.key
smtpd_tls_auth_only = yes
smtp_use_tls = yes
smtp_tls_note_starttls_offer = yes
smtpd_tls_loglevel = 3
smtpd_tls_received_header = yes
```

Configurar SASLauthd

```
vi /usr/lib/sasl2/smtpd.conf

pwcheck_method: authdaemond
log_level: 3
mech_list: PLAIN LOGIN
authdaemond_path:/var/spool/authdaemon/socket
```

Agora vamos parar o serviço sendmail e startar o postfix, courier-auth e saslauth:

```
# chmod 755 /var/spool/authdaemon
# chkconfig --levels 235 courier-authlib on
# /etc/init.d/courier-authlib start
# chkconfig --levels 235 sendmail off
# chkconfig --levels 235 postfix on
# chkconfig --levels 235 saslauthd on
# /etc/init.d/sendmail stop
# /etc/init.d/postfix start
# /etc/init.d/saslauthd start
```

Agora vamos configurar o courier para autenticar via MySQL:

```
# vi /etc/authlib/authdaemonrc

authmodulelist="authmysql"
```

Atentar para os campos de username e senha:

```
# vi /etc/authlib/authmysqlrc

MYSQL_SERVER localhost
MYSQL_USERNAME mail_admin
MYSQL_PASSWORD mail_admin_password
MYSQL_PORT 0
MYSQL_DATABASE mail
MYSQL_USER_TABLE users
MYSQL_CRYPT_PWFIELD password
#MYSQL_CLEAR_PWFIELD password
MYSQL_UID_FIELD 5000
MYSQL_GID_FIELD 5000
MYSQL_LOGIN_FIELD email
MYSQL_HOME_FIELD "/home/vmail"
MYSQL_MAILDIR_FIELD CONCAT(SUBSTRING_INDEX(email,'@',-
1),'/',SUBSTRING_INDEX(email,'@',1),'/')
#MYSQL_NAME_FIELD
MYSQL_QUOTA_FIELD quota
```

e restartarmos o courier:

```
# chkconfig --levels 235 courier-imap on
# /etc/init.d/courier-authlib restart
# /etc/init.d/courier-imap restart
```

Para testar o serviço POP rode:

```
# telnet localhost pop3
```

Deverá retornar "+OK Hello there". Digite "quit" para sair.

Para testar o postfix e sasl, rode:

```
# telnet localhost 25

ehlo localhost
```

Deverá ser mostrado o seguinte:

```
250-mailserver
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-AUTH PLAIN LOGIN
250-AUTH=PLAIN LOGIN
250 8BITMIME
OK - quit para sair
```

Utilizando o Postfix

O arquivo de log do Postfix fica em /var/log/maillog. Para visualizar o fim do arquivo permanentemente use o comando:

```
# tail -f -n 20 /var/log/maillog
```

Por este arquivo você acompanha tudo o que acontece no postfix. É muito útil para detecção de problemas ou melhoramento de configurações.

Tabela Domains Não tem mistério, apenas cadastre o domínio. Ex: dominio.net.

Tabela Users Cadastre os usuários: Ex.:

email: farley@dominio.net password: tipo(encrypt) quota: por padrão 10M, mas pode ser mudado.

Tabela Transport domain - transport dominio.net - Utiliza o servidor de email local.

dominio.net - smtp:outroservidor.com.br O domínio dominio.net utilizará o servidor de smtp de outroservidor.com.br.

dominio.net - smtp:outroservidor.com.br:2025 O domínio dominio.net utilizará o servidor smtp outroservidor.com.br na porta 2025.

- - smtp:outroservidor.com.br

Todos os domínios utilizarão o servidor outroservidor.com.br.

joe@examplo.com - smtp:outroservidor.com.br Emails para joe@exemplo.com serão entregues a outroservidor.com.br.

Tabela forwarding source - destination

email1@exemplo.net - email2@exemplo.net Redireciona os emails de email1@exemplo.net para email2@exemplo.net.

@exemplo.net - admin@exemplo.net Todos os emails enviados para @exemplo.net serão repassados para admin@exemplo.net, a não ser que o usuário esteja cadastrado neste domínio. Por exemplo, um email enviado para teste@exemplo.net e o usuário teste@exemplo.net estiver cadastrado na tabela de usuários, o email chegara para o usuário teste@exemplo.net. Se ele não estiver cadastrado, o email chegará para admin@exemplo.net.

@exemplo.net - @outroexemplo.net Redireciona todos os emails do domínio exemplo.net para outroexemplo.net. Por exemplo, o email passado para email1@exemplo.net será repassado para email1@outroexemplo.net.

email1@exemplo.net - email2@exemplo.net,teste@outroex.net Encaminha todos os emails de email1@exemplo.net para email2@exemplo.net e teste@outroex.net.

Instalando MailScanner e SpamAssassin

Instalando Mailscanner

Alterando local onde os RPMS irão ficar:

```
# vi /root/.rpmmacros

%_topdir /usr/src/redhat
```

Instalando o Mailscanner

```
# wget http://www.mailscanner.info/files/4/rpm/MailScanner-4.58.9-
1.rpm.tar.gz
# tar -xvzf MailScanner-4.58.9-1.rpm.tar.gz
# cd MailScanner-4.58.9-1
# export LANG=C
# vi
# ./install.sh
```

Configurando:

```
# vi /etc/MailScanner/MailScanner.conf
```

```
%org-name% = mycompany.hosting
%org-long-name% = MyCompany Hosting
%web-site% = http://www.mycompany.com
%report-dir% = /etc/MailScanner/reports/pt_br

Run As User = postfix
Run As Group = postfix
MTA = postfix

Incoming Queue Dir = /var/spool/postfix/hold
Outgoing Queue Dir = /var/spool/postfix/incoming

File Timeout = 120
Maximum Archive Depth = 20
Virus Scanners = clamavmodule
Monitors for ClamAV Updates = /var/lib/clamav/*.cvd
Use SpamAssassin = yes
SpamAssassin User State Dir = /var/spool/MailScanner/spamassassin

Spam List = ORDB-RBL SBL+XBL SORBS-DNSBL CBL RSL DSBL spamcop
Allow IFrame Tags = yes
Allow Script Tags = yes
Allow Object Codebase Tags = yes
Convert Dangerous HTML To Text = no
Minimum Stars If On Spam List = 3
Spam Lists To Reach High Score = 3
Sign Clean Messages = yes
Spam Actions = deliver
High Scoring Spam Actions = deliver
# vi /etc/postfix/main.cf

header_checks = regexp:/etc/postfix/header_checks
# vi /etc/postfix/header_check
```

e adicione esta linha no final do arquivo:

```
/^Received:/ HOLD
# mkdir /var/spool/MailScanner/spamassassin
# chown postfix.postfix /var/spool/MailScanner/spamassassin
# chown postfix.postfix /var/spool/MailScanner/incoming
# chown postfix.postfix /var/spool/MailScanner/quarantine
# vi /etc/MailScanner/virus.scanners.conf
```

e altere a seguinte linha pra ficar assim:

```
clamav /usr/lib/MailScanner/clamav-wrapper /usr
# vi /etc/MailScanner/filename.rules.conf

allow .[a-z][a-z0-9]{2,3}s*.[a-z0-9]{3}$ Found possible filename
hiding
allow s{10,0} Filename contains lots of white space
allow {[a-hA-H0-9-]{25,}} Filename trying to hide its real type
allow .exe$ Windows/DOS Executable
allow .bmp$ Windows bitmap file security vulnerability
# vi /etc/MailScanner/filetype.rules.conf

allow self-extract - -
allow ELF - -
allow executable - -
chkconfig --levels 235 postfix off
```

```
chkconfig --levels 2345 MailScanner on
```

Algumas configurações do MailScanner

- Max Children = 1 -> quantas instâncias do MailScanner estaremos executando por vez, útil para acelerar o escaneamento de emails, mas acarreta uma maior carga no servidor.

- Filename Rules = %etc-dir%/filename.rules.conf -> lista com as regras de processamento de anexos, edite este arquivo e configure conforme sua necessidade.

- Quarantine Infections = no -> armazena uma cópia da mensagem infectada no servidor, aconselho a colocar "no".

- Deleted Bad Content Message Report = %report-dir%/deleted.content.message.txt -> mensagem enviada ao usuário caso a mensagem apresente conteúdo perigoso, personalize este arquivo conforme sua necessidade.

- Deleted Bad Filename Message Report = %report-dir%/deleted.filename.message.txt -> mensagem enviada para o usuário notificando-o sobre o anexo infectado, personalize a mensagem conforme sua necessidade.

- Deleted Virus Message Report = %report-dir%/deleted.virus.message.txt -> mensagem enviada notificando sobre vírus, personalize a mensagem conforme sua necessidade.

- Stored Bad Content Message Report = %report-dir%/stored.content.message.txt -> mensagem enviada para o usuário notificando que o email está em quarentena, personalize conforme sua necessidade.

- Stored Bad Filename Message Report = %report-dir%/stored.filename.message.txt -> mensagem enviada ao usuário notificando que o anexo está em quarentena, personalize conforme sua necessidade.

- Stored Virus Message Report = %report-dir%/stored.virus.message.txt -> email enviado ao usuário notificado-sobre a mensagem que está em quarentena, personalize conforme sua necessidade.

- Disinfected Report = %report-dir%/disinfected.report.txt -> mensagem enviada ao usuário notificando que o email foi desinfectado, personalize conforme sua necessidade.

- Inline HTML Warning = %report-dir%/inline.warning.html -> mensagem adicionada ao corpo do email dizendo que este foi escaneado.

- Inline Text Warning = %report-dir%/inline.warning.txt -> mensagem adicionada ao corpo do email dizendo que este foi escaneado.

- Sender Content Report = %report-dir%/sender.content.report.txt -> mensagem de resposta notificando o usuário que ele está enviando um vírus.

- Sender Error Report = %report-dir%/sender.error.report.txt-> mensagem de resposta notificando o usuário que ele está enviando um vírus.

- Sender Bad Filename Report = %report-dir%/sender.filename.report.txt-> mensagem de resposta notificando o usuário que ele está enviando um vírus.

- Sender Virus Report = %report-dir%/sender.virus.report.txt-> mensagem de resposta notificando o usuário que ele está enviando um vírus.

- Local Postmaster = email@dominio.com.br -> Local postmaster que irá receber as notificações de tudo o que está acontecendo em seu MailScanner.

Travar por email

No arquivo MailScanner.conf existe uma linha assim:

```
Filename Rules = %etc-dir%/filename.rules.conf
Filetype Rules = %etc-dir%/filetype.rules.conf
```

A propriedade "Filename Rules" aponta para um arquivo que contém um conjunto de regras visando aceitar ou negar um nome de arquivo, por exemplo, arquivos com extensão .exe, com muitos espaços entre uma letra e outra e com extensões repetidas são fortes candidatos a serem negados.

A propriedade "Filetype Rules" é parecida com Filename, mas trata de tipos de arquivos baseados em sua extensão, .zip, .exe, .mp3.

Substitua estas linhas por:

```
Filename Rules = %etc-dir%/filename.rules
Filetype Rules = %etc-dir%/filetype.rules
```

O arquivo configurado poderá ter qualquer nome, desde que o mesmo termine com .rules. É assim que o mailscanner identifica se um arquivo é uma regra ou um conjunto de regras.

Crie os arquivos filename.rules e filetype.rules.

É necessário criar os arquivos para que o MailScanner identifique qual arquivo de regras utilizar.

A sintaxe do arquivo é:

From ou To ou FromOrTO:

Um exemplo de arquivo filename.rules:

From: wberbert@sermap.com.br %etc-dir%/filename.allow.exe.rules.conf FromOrTo: default %etc-dir%/filename.rules.conf

Isto significa que se o email vier de wberbert@sermap.com.br, o MailScanner utilizará o arquivo filename.allow.exe.rules.conf para validar os anexos, caso venha de outro email, utilizará filename.rules.conf.

A mesma sintaxe para filetype.rules:

From: wberbert@sermap.com.br %etc-dir%/filetype.allow.exe.rules.conf FromOrTo: default %etc-dir%/filetype.rules.conf

Não esqueça de mudar o nome do arquivo que será apontado filename é diferente de filetype.

Instalando MailWatch

Fonte: http://mailwatch.sourceforge.net/doku.php?id=mailwatch:documentation

Baixando o mailwatch:

```
# cd /root
# wget http://ufpr.dl.sourceforge.net/sourceforge/mailwatch/mailwatch-
1.0.3.tar.gz
```

Instalando dependências -> é necessário ter instalado: PHP com suporte a MySQL e GD, Perl, DBD, DBD-MySQL. Se for necessário instalar alguma dependência, logo em seguida reinicie o Apache:

```
# service httpd restart
```

Acertando a configuração do php:

```
# vi /etc/php.ini

short_open_tag = On
safe_mode = Off
register_globals = Off
magic_quotes_gpc = On
magic_quotes_runtime = Off
session.auto_start = 0
```

Executando algumas instruções SQL:

```
# tar -zxvf mailwatch-1.0.3.tar.gz
# cd mailwatch
# mysql -p < create.sql
# mysql -u root -p
entre com 'suasenha'

GRANT ALL ON mailscanner.* TO mailwatch@localhost IDENTIFIED BY
'password';
quit;
# vi MailWatch.pm
```

```
E acerte os campos $db_user e $db_pass para root e 'suasenha'.
# mv MailWatch.pm /usr/lib/MailScanner/MailScanner/CustomFunctions/
# mysql mailscanner -u mailwatch -p
entre com 'suasenha'

INSERT INTO users VALUES
('username',md5('password'),'name','A','0','0','0','0','0');
quit;
```

Instalando e configurando o MailWatch:

```
# mv mailscanner /var/www/html/
# cd /var/www/html/mailscanner
# chown root:apache images
# chmod ug+rwx images
# chown root:apache images/cache
# chmod ug+rwx images/cache
# cp conf.php.example conf.php
# vi conf.php

Entre com usuário e senha do MySQL nos campos DB_USER e DB_PASS.
QUARANTINE_USE_FLAG, true.
QUARANTINE_DAYS_TO_KEEP -> passe o número de dias que será limpo os
arquivos em quarentena e rode:

# /root/mailwatch/tools/quarantine_maint.php --clean
# echo "/root/mailwatch/tools/quarantine_maint.php --clean" >
/etc/cron.daily/mailwatch_quarantine_maint.sh
# chmod +x /etc/cron.daily/mailwatch_quarantine_maint.sh

# service MailScanner stop
# vi /etc/MailScanner/MailScanner.conf

Quarantine User = root
Quarantine Group = apache
Quarantine Permissions = 0660
Quarantine Whole Message = yes
Quarantine Whole Message As Queue Files = no
Detailed Spam Report = yes
Include Scores In SpamAssassin Report = yes
Always Looked Up Last = &MailWatchLogging

# service MailScanner start
```