SLES10 SP1 DSPAM

Installatie

- **Language**
  - English US
- **Accept Licence Agreement**
  - Yes
- **Installation Type**
  - New
- **Clock and timezone**
  - Clock is set to Local time
  - Timezone Europe, Netherlands
- **Partitioning**
  - Create LVM Based Proposal
- **Software Selection**
  - Server Base System
  - Common Code Base
  - GNOME Desktop Enviroment
  - X Window System
  - Web and LAMP Server
  - C/C++ Compiler and tools
- **Accept packages**
  - Accept settings
- **Root Password**
  - novell
- **Hostname**
  - dspam
- **Domain name**
  - comsolve.nl
  - Change hostname via DHCP
    - Unchecked
- Write hostname to /etc/hosts
  - Checked
- Network Mode
  - Traditional
- Firewall
  - Enabled
  - SSH port Open
- Network interfaces
  - Eth0
    - IP adres 172.16.0.93
    - Netmask 255.255.224.0
    - Gateway 172.16.0.254
    - DNS 172.16.0.66
- VNC Remote Administration

        - Allow Remote Administration
        - Open port in firewall
            - Checked
- Test internet connection
    - Yes
- Customer center configuration
    - Configure Now
        - Hardware profile checked
        - Optional information checked
        - Registration Code checked
    - Regulary Synchronize with the Customer Center checked
- Product activation
    - M.honkoop@nsnl.nl
    - FD4B8481015B48
- Online update
    - Skip Update
- CA management
    - Accept defaults (check if country is correct)
- OpenLDAP server
    - Start : No
- Authentication method
    - Local (etc/passwd)
- New Local user
    - No new local user
        - Accept waring, continue with Yes.
- Hardware configuration
    - Accept defaults
- Install completion
    - Clone system for AutoYast checked

Basic Configuratie

- Update Server to latest patches/versions.

Preparations :

- create work dirs

```
# mkdir –p /var/work/source
# mkdir –p /var/work/compile/configure
```

- install locate (to find files quickly) and update it’s database

```
# yast –i findutils-locate
# updatedb
```

- Get DSPAM source

```
# cd /var/work/source
# wget http://dspam.nuclearelephant.com/sources/dspam-3.8.0.tar.gz
```

- Get postfix, MySQL development packages, ClamAV

```
# yast –i postfix mysql-devel clamav
```

- Update clamav's definitions

```
# freshclam
```

- Start clamav daemon & enable it and the updater on system startup

```
# /etc/init.d/clamd start
# chkconfig clamd on
# chkconfig freshclam on
```

- User & group creation

```
# groupadd -g 2000 dspam
# useradd -u 2001 -g 2000 -d /var/dspam -c "DSPAM Server" -s /sbin/false -G maildrop
dspam
```

- Start MySQL Server and enable service on startup

```
# /etc/init.d/mysql start
# chkconfig mysql on
```

- Set root password for MySQL

```
#mysqladmin -u root password novell
```

In order not to have to type 'mysql -u root -p' each times we want to login mysql;
here is the tip:

```
#vi ~/.my.cnf
```

```
[client]
password=novell
```

```
#chmod 400 ~/.my.cnf
```

Compiling DSPAM

- Unpack DSPAM source

```
# cd ../compile
# tar -zxvf ../source/dspam-3.8.0.tar.gz
```

- **Create the configuration file for DSPAM**

```
# cd  dspam-3.8.0
# vi ../configure/dspam
```

```
#!/bin/sh
./configure \
       --with-dspam-home=/var/dspam \
       --with-dspam-home-mode=770 \
       --with-dspam-home-owner=dspam \
       --with-dspam-home-group=dspam \
       --with-dspam-owner=dspam \
       --with-dspam-group=dspam \
       --with-delivery-agent=/usr/sbin/sendmail \
       --with-storage-driver=mysql_drv \
       --with-mysql_includes=/usr/include/mysql \
      --with-mysql-libraries=/usr/lib/mysql \
      --enable-preferences-extension \
      --enable-domain-scale \
      --enable-virtual-users \
      --enable-clamav \
      --enable-daemon \
      --enable-debug
```

```
# chmod 755 ../configure/dspam
# ../configure/dspam
# make && make install
# mkdir -p /usr/local/share/dspam/
```

MySQL Dspam user and Database creation

```
# cd src/tools.mysql_drv/
# mysql -e "create database dspam"
# mysql -e "grant all on dspam.* to dspam@localhost identified by 'spameater'"
# mysql dspam < mysql_objects-4.1.sql
# mysql dspam <virtual_users.sql
# cp purge-4.1.sql /usr/local/share/dspam/
```

- Creare a cronjob for cleaning up the DSPAM database

```
#crontab –e
```

```
      0 0 * * * /usr/bin/mysql -udspam -pspameater dspam <
usr/local/share/dspam/purge-4.1.sql 2>&1
```

Postfix Configuration

```
# cd /etc/postfix
```

```
# mkdir original
# cp master.cf original/
# vi master.cf
```

(add / remove what is needed)

```
#
# Postfix master process configuration file.  For details on the format
# of the file, see the Postfix master(5) manual page.
#
# ==========================================================================
# service type  private unpriv  chroot  wakeup  maxproc command + args
#               (yes)   (yes)   (yes)   (never) (100)
# ==========================================================================
smtp inet  n       -       n       -       -        smtpd
  -o content_filter=lmtp:[127.0.0.1]:10024
localhost:10026 inet  n -       n       -       -        smtpd
  -o content_filter=
  -o receive_override_options=no_unknown_recipient_checks,no_header_body_checks
  -o smtpd_helo_restrictions=
  -o smtpd_client_restrictions=
  -o smtpd_sender_restrictions=
  -o smtpd_recipient_restrictions=permit_mynetworks,reject
  -o mynetworks=127.0.0.0/8
  -o smtpd_authorized_xforward_hosts=127.0.0.0/8
#submission inet n       -       n       -       -       smtpd
#       -o smtpd_etrn_restrictions=reject
#       -o smtpd_client_restrictions=permit_sasl_authenticated,reject
#smtps     inet  n       -       n       -       -       smtpd
#  -o smtpd_tls_wrappermode=yes -o smtpd_sasl_auth_enable=yes
#submission   inet    n       -       n       -       -       smtpd
#  -o smtpd_etrn_restrictions=reject
#  -o smtpd_enforce_tls=yes -o smtpd_sasl_auth_enable=yes
#628       inet  n       -       n       -       -       qmqpd
pickup    fifo  n       -       n       60      1       pickup
cleanup   unix  n       -       n       -       0       cleanup
qmgr      fifo  n       -       n       300     1       qmgr
#qmgr     fifo  n       -       n       300     1       oqmgr
#tlsmgr    unix  -       -       n       1000?   1       tlsmgr
rewrite   unix  -       -       n       -       -       trivial-rewrite
bounce    unix  -       -       n       -       0       bounce
defer     unix  -       -       n       -       0       bounce
trace     unix  -       -       n       -       0       bounce
verify    unix  -       -       n       -       1       verify
flush     unix  n       -       n       1000?   0       flush
proxymap  unix  -       -       n       -       -       proxymap
smtp      unix  -       -       n       -       -       smtp
# When relaying mail as backup MX, disable fallback_relay to avoid MX loops
relay     unix  -       -       n       -       -       smtp
        -o fallback_relay=
#       -o smtp_helo_timeout=5 -o smtp_connect_timeout=5
showq     unix  n       -       n       -       -       showq
error     unix  -       -       n       -       -       error
discard   unix  -       -       n       -       -       discard
local     unix  -       n       n       -       -       local
virtual   unix  -       n       n       -       -       virtual
lmtp      unix  -       -       n       -       -       lmtp
anvil     unix  -       -       n       -       1       anvil
scache    unix  -       -       n       -       1       scache
#
# ====================================================================
# Interfaces to non-Postfix software. Be sure to examine the manual
# pages of the non-Postfix software to find out what options it wants.
#
```

```
# Many of the following services use the Postfix pipe(8) delivery
# agent.  See the pipe(8) man page for information about ${recipient}
# and other message envelope options.
# ====================================================================
#
# maildrop. See the Postfix MAILDROP_README file for details.
# Also specify in main.cf: maildrop_destination_recipient_limit=1
#
maildrop  unix  -       n       n       -       -       pipe
  flags=DRhu user=vmail argv=/usr/local/bin/maildrop -d ${recipient}
cyrus     unix  -       n       n       -       -       pipe
  user=cyrus argv=/usr/lib/cyrus/bin/deliver -e -r ${sender} -m ${extension} $
{user}
uucp      unix  -       n       n       -       -       pipe
  flags=Fqhu user=uucp argv=uux -r -n -z -a$sender - $nexthop!rmail ($recipient)
ifmail    unix  -       n       n       -       -       pipe
  flags=F user=ftn argv=/usr/lib/ifmail/ifmail -r $nexthop ($recipient)
bsmtp     unix  -       n       n       -       -       pipe
  flags=Fq. user=foo argv=/usr/local/sbin/bsmtp -f $sender $nexthop $recipient
procmail  unix  -       n       n       -       -       pipe
  flags=R user=nobody argv=/usr/bin/procmail -t -m /etc/procmailrc ${sender} $
{recipient}
dspam     unix  -       n       -       -       10      pipe
    flags=Rhqu user=dspam argv=/usr/local/bin/dspam --deliver=innocent --user $
{recipient} -i -f ${sender} -- ${recipient}
```

- Create a new main.cf

```
# mv main.cf original/
#vi main.cf
```

```
# Disable biff service
biff = no
# Set the welcome banner on connect
smtpd_banner = $myhostname ESMTP $mail_name (SuSe Linux Enterprise Server)
# postfix related settings
mail_spool_directory = /var/mail
queue_directory = /var/spool/postfix
command_directory = /usr/sbin
daemon_directory = /usr/lib/postfix
debug_peer_level = 2
debugger_command =
        PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin
        xxgdb $daemon_directory/$process_name $process_id & sleep 5
mail_owner = postfix
sendmail_path = /usr/sbin/sendmail
newaliases_path = /usr/bin/newaliases
mailq_path = /usr/bin/mailq
setgid_group = maildrop
inet_interfaces = all

# appending .domain is the MUA's job.
append_dot_mydomain = no
# LDAP config to eDirectory
alias_maps = ldap:/etc/postfix/ldap-aliases.cf
alias_database = $alias_maps


mydomain= comsolve.nl
myorigin = $mydomain
myhostname = dspam.$mydomain
relay_domains = $mydomain
mynetworks = 172.16.0.0/19, 127.0.0.0/8
message_size_limit = 10485760
local_transport = error:no local mail delivery
```

```
mydestination = $myhostname, $mydomain
unknown_address_reject_code = 550

virtual_alias_maps = hash:/etc/postfix/virtual
transport_maps = hash:/etc/postfix/transport

recipient_delimiter = +

smtpd_helo_required = yes

smtpd_sender_restrictions =
        check_sender_access hash:/etc/postfix/access,
        reject_non_fqdn_sender,
        reject_unauth_destination,
        reject_unknown_sender_domain,
        reject_rbl_client list.dsbl.org,
        reject_rbl_client bl.spamcop.net,
        reject_rbl_client sbl-xbl.spamhaus.org,
        reject_rbl_client sbl.spamhaus.org,
        reject_rbl_client cbl.abuseat.org
maps_rbl_domains =
        relays.visi.com,
        relays.mail-abuse.org,
        dialups.mail-abuse.org,
        blackholes.mail-abuse.org

smtpd_data_restrictions =
        reject_unauth_pipelining

header_checks =
        regexp:/etc/postfix/header_checks

smtpd_recipient_restrictions =
        check_sender_access hash:/etc/postfix/not_our_domain_as_sender,
        permit_mynetworks,
        reject_unauth_destination,
        reject_invalid_hostname,
        reject_unknown_recipient_domain,
        check_recipient_access hash:/etc/postfix/protect_subdomains,
        check_helo_access pcre:/etc/postfix/helo_checks

smtpd_helo_restrictions =
        reject_invalid_hostname,
        reject_unknown_hostname
```

- Create a ldap-aliases.cf

```
# vi ldap-aliases.cf
```

```
# LDAP configuration
server_host     = 172.16.0.72
server_port     = 389
timeout         = 90
search_base     = o=COMSOLVE
scope           = sub
query_filter    = mail=%s
result_attribute= mail
result_filter   = %s
```

- edit transport

```
# mv transport original/
# vi transport
```

```
comsolve.nl smtp:[172.16.0.63]
```

```
# postmap transport
```

- Edit virtual

```
# mv virtual original/
# vi virtual
```

```
root            postmaster@comsolve.nl
postmaster      postmaster@comsolve.nl
abuse           postmaster@comsolve.nl
```

```
# postmap virtual
```

- Create a relay_recipient

```
# vi relay_recipients
```

```
@comsolve.nl OK
```

```
# postmap relay_recipients
```

- Create a not_our_domain_as_sender

```
# vi not_our_domain_as_sender
```

```
comsolve.nl      554 Do not use my domain in your envelope sender
```

```
# postmap not_our_domain_as_sender
```

- Create a protect_subdomains

```
# vi protect_subdomains
```

```
dspam.comsolve.nl       554 Domain not available
```

```
# postmap protect_subdomains
```

- Create a helo_checks

```
# vi helo_checks
```

```
/^dspam\.comsolve\.nl$/          550 Don't use my hostname
/^213\.84\.172\.75$/             550 Don't use my IP address
/^\[213\.84\.172\.75\]$/         550 Don't use my IP address
/^[0-9.]+$/                      550 Your client is not RFC 2821 compliant
```

```
# postmap helo_checks
```

- Create a header_checks and body_checks

```
# touch header_checks
# touch body_checks
```

- Reload postfix

```
# postfix reload
```

Dspam webinterface

- Create appropriate directory structure

```
# cd /srv/www/
# mkdir dspam
# chmod 555 dspam
# chown dspam.dspam /srv/www/dspam
```

- Copy appropriate files to directories

```
# cd dspam
# cp -r /var/work/compile/dspam-3.8.0/webui/cgi-bin/* .
# cp –r /var/work/compile/dspam-3.8.0/webui/htdocs/* .
# rm -f Makefile*
# chown -R dspam.dspam *
# chmod 444 *.*
# chmod 554 *.cgi
# chmod 555 templates
# chmod 444 templates/*
```

- Create a vhost for dspam

```
# cd /etc/apache2/vhosts.d
# vi dspam.conf
```

```
Documentroot "/srv/www/dspam"
ServerName dspam.comsolve.nl
ServerAdmin webmaster@example.com
ErrorLog /var/log/apache2/dspam-error.log
TransferLog /var/log/apache2/dspam-access.log
SuexecUserGroup dspam dspam
HostnameLookups Off
UseCanonicalName On
ServerSignature On
RewriteEngine On
RewriteRule ^/$ /dspam.cgi [R]
 <Directory /srv/www/dspam>
      Order allow,deny
      Allow from all
      AuthName "Dspam Quarantine Area"
```

```
      AuthType Basic
      AuthBasicProvider ldap
      AuthLDAPUrl ldap://172.16.0.50/o=COMSOLVE?mail?sub?
      AuthzLDAPAuthoritative Off
       Require valid-user
       Options FollowSymLinks ExecCGI
       AddHandler cgi-script .cgi .pl
       Satisfy all
</Directory>
```

- Edit Apache settings for loading correct modules, either manual or via Yast*

Modules that are needed :

- Suexec module
- Rewrite module
- Ldap module
- Authz_ldap module

Note* I was faster with Yast in the Gui then searching where to put the loadmodule statements.

For complete documentation below the /etc/apache2/sysconfig.d/loadmodule.conf file

```
#
# Files in this directory are created at apache start time by /usr/sbin/rcapache2
# Do not edit them!
#

# as listed in APACHE_MODULES (/etc/sysconfig/apache2)

LoadModule suexec_module                /usr/lib/apache2-prefork/mod_suexec.so
LoadModule authz_host_module            /usr/lib/apache2-
prefork/mod_authz_host.so
LoadModule actions_module               /usr/lib/apache2-prefork/mod_actions.so
LoadModule alias_module                 /usr/lib/apache2-prefork/mod_alias.so
LoadModule auth_basic_module            /usr/lib/apache2-
prefork/mod_auth_basic.so
LoadModule authz_groupfile_module       /usr/lib/apache2-
prefork/mod_authz_groupfile.so
LoadModule authn_file_module            /usr/lib/apache2-
prefork/mod_authn_file.so
LoadModule authz_user_module            /usr/lib/apache2-
prefork/mod_authz_user.so
LoadModule authn_dbm_module             /usr/lib/apache2-
prefork/mod_authn_dbm.so
LoadModule autoindex_module             /usr/lib/apache2-
prefork/mod_autoindex.so
LoadModule cgi_module                   /usr/lib/apache2-prefork/mod_cgi.so
LoadModule dir_module                   /usr/lib/apache2-prefork/mod_dir.so
LoadModule env_module                   /usr/lib/apache2-prefork/mod_env.so
LoadModule expires_module               /usr/lib/apache2-prefork/mod_expires.so
LoadModule include_module               /usr/lib/apache2-prefork/mod_include.so
LoadModule log_config_module            /usr/lib/apache2-
prefork/mod_log_config.so
LoadModule mime_module                  /usr/lib/apache2-prefork/mod_mime.so
LoadModule negotiation_module           /usr/lib/apache2-
prefork/mod_negotiation.so
LoadModule setenvif_module              /usr/lib/apache2-prefork/mod_setenvif.so
LoadModule status_module                /usr/lib/apache2-prefork/mod_status.so
LoadModule userdir_module               /usr/lib/apache2-prefork/mod_userdir.so
LoadModule asis_module                  /usr/lib/apache2-prefork/mod_asis.so
LoadModule imagemap_module              /usr/lib/apache2-prefork/mod_imagemap.so
LoadModule ldap_module                  /usr/lib/apache2-prefork/mod_ldap.so
LoadModule authnz_ldap_module           /usr/lib/apache2-
prefork/mod_authnz_ldap.so
LoadModule proxy_module                 /usr/lib/apache2-prefork/mod_proxy.so
LoadModule rewrite_module               /usr/lib/apache2-prefork/mod_rewrite.so
LoadModule ssl_module                   /usr/lib/apache2-prefork/mod_ssl.so
LoadModule php5_module                  /usr/lib/apache2/mod_php5.so
LoadModule auth_imap_module             /usr/lib/apache2-
prefork/mod_auth_imap.so
```

```
LoadModule authz_default_module           /usr/lib/apache2-
prefork/mod_authz_default.so
```

- Editting dspam.conf

```
# cd /usr/local/etc/
# mkdir original
#  mv dspam.conf original
# vi dspam.conf
```

```
## $Id: dspam.conf.in,v 1.82 2006/06/23 03:11:31 jonz Exp $
## dspam.conf -- DSPAM configuration file
##

#
# DSPAM Home: Specifies the base directory to be used for DSPAM storage
#
Home /var/dspam

#
# StorageDriver: Specifies the storage driver backend (library) to use.
# You'll only need to set this if you are using dynamic storage driver plugins
# from a binary distribution. The default build statically links the storage
# driver (when only one is specified at configure time), overriding this
# setting, which only comes into play if multiple storage drivers are specified
# at configure time. When using dynamic linking, be sure to include the path
# to the library if necessary, and some systems may use an extension other
# than .so (e.g. OSX uses .dylib).
#
# Options include:
#
#   libmysql_drv.so     libpgsql_drv.so   libsqlite_drv.so
#   libsqlite3_drv.so   libhash_drv.so
#
# IMPORTANT: Switching storage drivers requires more than merely changing
# this option. If you do not wish to lose all of your data, you will need to
# migrate it to the new backend before making this change.
#
StorageDriver /usr/local/lib/libmysql_drv.so

#
# Trusted Delivery Agent: Specifies the local delivery agent DSPAM should call
# when delivering mail as a trusted user. Use %u to specify the user DSPAM is
# processing mail for. It is generally a good idea to allow the MTA to specify
# the pass-through arguments at run-time, but they may also be specified here.
#
# Most operating system defaults:
#TrustedDeliveryAgent "/usr/bin/procmail"       # Linux
#TrustedDeliveryAgent "/usr/bin/mail"           # Solaris
#TrustedDeliveryAgent "/usr/libexec/mail.local" # FreeBSD
#TrustedDeliveryAgent "/usr/bin/procmail"       # Cygwin
#
# Other popular configurations:
#TrustedDeliveryAgent "/usr/cyrus/bin/deliver"  # Cyrus
#TrustedDeliveryAgent "/bin/maildrop"           # Maildrop
#TrustedDeliveryAgent "/usr/local/sbin/exim -oMr spam-scanned" # Exim
#
TrustedDeliveryAgent "/usr/bin/procmail"

#
# Untrusted Delivery Agent: Specifies the local delivery agent and arguments
# DSPAM should use when delivering mail and running in untrusted user mode.
# Because DSPAM will not allow pass-through arguments to be specified to
# untrusted users, all arguments should be specified here. Use %u to specify
# the user DSPAM is processing mail for. This configuration parameter is only
# necessary if you plan on allowing untrusted processing.
#
#UntrustedDeliveryAgent "/usr/bin/procmail -d %u"
```

```
#
# SMTP or LMTP Delivery: Alternatively, you may wish to use SMTP or LMTP
# delivery to deliver your message to the mail server instead of using a
# delivery agent. You will need to configure with --enable-daemon to use host
# delivery, however you do not need to operate in daemon mode. Specify an IP
# address or UNIX path to a domain socket below as a host.
#
# If you would like to set up DeliveryHost's on a per-domain basis, use
# the syntax: DeliveryHost.domain.com 1.2.3.4
DeliveryHost        127.0.0.1
DeliveryPort        10026
DeliveryIdent       localhost
DeliveryProto       SMTP

#
# FallbackDomains: If you want to specify certain domains as fallback domains,
# enable this option. For example, you could create a user @domain.com, and
# if bob@domain.com does not resolve to a known user on the system, the user
# could default to your @domain.com user. NOTE: This also requires designating
# fallbackDomain for the domain name;
# e.g. dspam_admin ch pref domain.com fallbackDomain on
#
#FallbackDomains on

#
# Quarantine Agent: DSPAM's default behavior is to quarantine all mail it
# thinks is spam. If you wish to override this behavior, you may specify
# a quarantine agent which will be called with all messages DSPAM thinks is
# spam. Use %u to specify the user DSPAM is processing mail for.
#
#QuarantineAgent        "/usr/bin/procmail -d spam"

#
# DSPAM can optionally process "plused users" (addresses in the user+detail
# form) by truncating the username just before the "+", so all internal
# processing occurs for "user", but delivery will be performed for
# "user+detail". This is only useful if the LDA can handle "plused users"
# (for example Cyrus IMAP) and when configured for LMTP delivery above
#
# NOTE: Plused detail presently only works when usernames are provided and
#       not fully qualified email address (@domain).
#
#EnablePlusedDetail     on

#
# Quarantine Mailbox: DSPAM's LMTP code can send spam mail using LMTP to a
# "plused" mailbox (such as user+quarantine) leaving quarantine processing
# for retraining or deletion to be performed by the LDA and the mail client.
# "plused" mailboxes are supported by Cyrus IMAP and possibly other LDAs.
# The mailbox name must have the +
#
#QuarantineMailbox      +quarantine

#
# OnFail: What to do if local delivery or quarantine should fail. If set
# to "unlearn", DSPAM will unlearn the message prior to exiting with an
# un successful return code. The default option, "error" will not unlearn
# the message but return the appropriate error code. The unlearn option
# is use-ful on some systems where local delivery failures will cause the
# message to be requeued for delivery, and could result in the message
# being processed multiple times. During a very large failure, however,
# this could cause a significant load increase.
#
OnFail error

#
# Trusted Users: Only the users specified below will be allowed to perform
```

```
# administrative functions in DSPAM such as setting the active user and
# accessing tools. All other users attempting to run DSPAM will be restricted;
# their uids will be forced to match the active username and they will not be
# able to specify delivery agent privileges or use tools.
#
Trust root
Trust mail
Trust mailnull
Trust smmsp
Trust daemon
Trust postfix
Trust dspam
#Trust nobody
#Trust majordomo

#
# Debugging: Enables debugging for some or all users. IMPORTANT: DSPAM must
# be compiled with debug support in order to use this option. DSPAM should
# never be running in production with debug active unless you are
# troubleshooting problems.
#
# DebugOpt: One or more of: process, classify, spam, fp, inoculation, corpus
#   process     standard message processing
#   classify    message classification using --classify
#   spam        error correction of missed spam
#   fp          error correction of false positives
#   inoculation message inoculations (source=inoculation)
#   corpus      corpusfed messages (source=corpus)
#
#Debug *
#Debug bob bill
#
#DebugOpt process spam fp

#
# ClassAlias: Alias a particular class to spam/nonspam. This is useful if
# classifying things other than spam.
#
#ClassAliasSpam badstuff
#ClassAliasNonspam goodstuff

#
# Training Mode: The default training mode to use for all operations, when
# one has not been specified on the commandline or in the user's preferences.
# Acceptable values are:
#     toe     Train on Error (Only)
#     teft    Train Everything (Trains on every message)
#     tum     Train Until Mature (Train only tokens without enough data)
#     notrain Do not train or store signatures (large ISP systems, post-train)
#
TrainingMode teft

#
# TestConditionalTraining: By default, dspam will retrain certain errors
# until the condition is no longer met. This usually accelerates learning.
# Some people argue that this can increase the risk of errors, however.
#
TestConditionalTraining on

#
# Features: Specify features to activate by default; can also be specified
# on the commandline. See the documentation for a list of available features.
# If _any_ features are specified on the commandline, these are ignored.
#
Feature noise
Feature whitelist

# Training Buffer: The training buffer waters down statistics during training.
```

```
# It is designed to prevent false positives, but can also dramatically reduce
# dspam's catch rate during initial training. This can be a number from 0
# (no buffering) to 10 (maximum buffering). If you are paranoid about false
# positives, you should probably enable this option.
#
#Feature tb=5

#
# Algorithms: Specify the statistical algorithms to use, overriding any
# defaults configured in the build. The options are:
#    naive       Naive-Bayesian (All Tokens)
#    graham      Graham-Bayesian ("A Plan for Spam")
#    burton      Burton-Bayesian (SpamProbe)
#    robinson    Robinson's Geometric Mean Test (Obsolete)
#    chi-square  Fisher-Robinson's Chi-Square Algorithm
#
# You may have multiple algorithms active simultaneously, but it is strongly
# recommended that you group Bayesian algorithms with other Bayesian
# algorithms, and any use of Chi-Square remain exclusive.
#
# NOTE: For standard "CRM114" Markovian weighting, use 'naive', or consider
#       using 'burton' for slightly better accuracy
#
# Don't mess with this unless you know what you're doing
#
#Algorithm chi-square
#Algorithm naive
Algorithm graham burton

#
# Tokenizer: Specify the tokenizer to use. The tokenizer is the piece
# responsible for parsing the message into individual tokens. Depending on
# how many resources you are willing to trade off vs. accuracy, you may
# choose to use a less or more detailed tokenizer:
#   word    uniGram (single word) tokenizer
#           Tokenizes message into single individual words/tokens
#           example: "free" and "viagra"
#   chain   biGram (chained tokens) tokenizer (default)
#           Single words + chains adjacent tokens together
#           example: "free" and "viagra" and "free viagra"
#   sbph    Sparse Binary Polynomial Hashing tokenizer
#           Creates sparse token patterns across sliding window of 5-tokens
#           example: "the quick * fox jumped" and "the * * fox jumped"
#   osb     Orthogonal Sparse biGram
#           Similar to SBPH, but only uses the biGrams
#           example: "the * * fox" and "the * * * jumped"
#
Tokenizer chain

#
# PValue: Specify the technique used for calculating Probability Values,
# overriding any defaults configured in the build. These options are:
#    bcr         Bayesian Chain Rule (Graham's Technique - "A Plan for Spam")
#    robinson    Robinson's Technique (used in Chi-Square)
#    markov      Markovian Weighted Technique (for Markovian discrimination)
#
# Unlike the "Algorithms" property, you may only have one of these defined.
# Use of the chi-square algorithm automatically changes this to robinson.
#
# Don't mess with this unless you know what you're doing.
#
#PValue robinson
#PValue markov
PValue bcr

#
# WebStats: Enable this if you are using the CGI, which writes .stats files
WebStats on
```

```
#
# ImprobabilityDrive: Calculate odds-ratios for ham/spam, and add to
# X-DSPAM-Improbability headers
#
ImprobabilityDrive on

#
# Preferences: Specify any preferences to set by default, unless otherwise
# overridden by the user (see next section) or a default.prefs file.
# If user or default.prefs are found, the user's preferences will override any
# defaults.
#
Preference "spamAction=quarantine"
Preference "signatureLocation=headers"  # 'message' or 'headers'
Preference "showFactors=on"
#Preference "spamAction=tag"
#Preference "spamSubject=SPAM"

#
# Overrides: Specifies the user preferences which may override configuration
# and commandline defaults. Any other preferences supplied by an untrusted user
# will be ignored.
#
AllowOverride trainingMode
AllowOverride spamAction spamSubject
AllowOverride statisticalSedation
AllowOverride enableBNR
AllowOverride enableWhitelist
AllowOverride signatureLocation
AllowOverride showFactors
AllowOverride optIn optOut
AllowOverride whitelistThreshold

# --- MySQL ---

#
# Storage driver settings: Specific to a particular storage driver. Uncomment
# the configuration specific to your installation, if applicable.
#
MySQLServer     /var/lib/mysql/mysql.sock
#MySQLPort
MySQLUser               dspam
MySQLPass               spameater
MySQLDb                 dspam
MySQLCompress           true

# If you are using replication for clustering, you can also specify a separate
# server to perform all writes to.
#
#MySQLWriteServer       /var/lib/mysql/mysql.sock
#MySQLWritePort
#MySQLWriteUser         dspam
#MySQLWritePass         changeme
#MySQLWriteDb           dspam_write
#MySQLCompress          true

# If your replication isn't close to real-time, your retraining might fail if
# the  signature isn't found. One workaround for this is to use the write
# database for all signature reads:
#
#MySQLReadSignaturesFromWriteDb on

# Use this if you have the 4.1 quote bug (see doc/mysql.txt)
#MySQLSupressQuote      on

# If you're running DSPAM in client/server (daemon) mode, uncomment the
# setting below to override the default connection cache size (the number
```

```
# of connections the server pools between all clients). The connection cache
# represents the maximum number of database connections *available* and should
# be set based on the maximum number of concurrent connections you're likely
# to have. Each connection may be used by only one thread at a time, so all
# other threads _will block_ until another connection becomes available.
#
MySQLConnectionCache    10

# If you're using vpopmail or some other type of virtual setup and wish to
# change the table dspam uses to perform username/uid lookups, you can over-
# ride it below

MySQLVirtualTable          dspam_virtual_uids
MySQLVirtualUIDField       uid
MySQLVirtualUsernameField  username

# UIDInSignature: MySQL supports the insertion of the user id into the DSPAM
# signature. This allows you to create one single spam or fp alias
# (pointing to some arbitrary user), and the uid in the signature will
# switch to the correct user. Result: you need only one spam alias

MySQLUIDInSignature    on

# --- PostgreSQL ---

#PgSQLServer            127.0.0.1
#PgSQLPort              5432
#PgSQLUser              dspam
#PgSQLPass              changeme
#PgSQLDb                dspam

# If you're running DSPAM in client/server (daemon) mode, uncomment the
# setting below to override the default connection cache size (the number
# of connections the server pools between all clients).
#
#PgSQLConnectionCache   3

# UIDInSignature: PgSQL supports the insertion of the user id into the DSPAM
# signature. This allows you to create one single spam or fp alias
# (pointing to some arbitrary user), and the uid in the signature will
# switch to the correct user. Result: you need only one spam alias

#PgSQLUIDInSignature    on

# If you're using vpopmail or some other type of virtual setup and wish to
# change the table dspam uses to perform username/uid lookups, you can over-
# ride it below

#PgSQLVirtualTable          dspam_virtual_uids
#PgSQLVirtualUIDField       uid
#PgSQLVirtualUsernameField  username

# --- SQLite ---

#SQLitePragma   "synchronous = OFF"

# --- Hash ---

#
# HashRecMax: Default number of records to create in the initial segment when
# building hash files. 100,000 yields files 1.6MB in size, but can fill up
# fast, so be sure to increase this (to a million or more) if you're not using
# autoextend.
#
# NOTE: If you're using a heavy-weight tokenizer, such as SBPH, you should be
#       looking for settings in the 'millions' of records.
#
# Primes List:
```

```
#  53, 97, 193, 389, 769, 1543, 3079, 6151, 12289, 24593, 49157, 98317, 196613,
#  393241, 786433, 1572869, 3145739, 6291469, 12582917, 25165843, 50331653,
#  100663319, 201326611, 402653189, 805306457, 1610612741, 3221225473,
#  4294967291
#
HashRecMax              98317

#
# HashAutoExtend: Autoextend hash databases when they fill up. This allows
# them to continue to train by adding extents (extensions) to the file. There
# will be a small delay during the growth process, as everything needs to be
# closed and remapped.
#
HashAutoExtend          on

#
# HashMaxExtents: The maximum number of extents that may be created in a single
# hash file. Set this to zero for unlimited
#
HashMaxExtents          0

#
# HashExtentSize: The initial record size for newly created extents. Creating
# this too small could result in many extents being created. Creating this too
# large could result in excessive disk space usage. Typically, a value close
# to half of the HashRecMax size is good.
#
HashExtentSize          49157

#
# HashPctIncrease: Increase the next extent size by n% from the size of the
# last extent. This is useful in accommodating systems where the default
# HashExtentSize can be too small for certain high-volume users, and can also
# help keep seeks nice and speedy and/or prevent too many unnecessary extents
# from being created when using a low HashMaxSeek. The default behavior, when
# HashPctIncrease is not used, is to always use # HashExtentSize with no
# increase.
#
HashPctIncrease 10

#
# HashMaxSeek: The maximum number of record seeks when inserting a new record
# before failing or adding a new extent. This ultimately translates into the
# max # of acceptable seeks per segment. Setting this too high will exhaustively
# scan each segment and hurt performance. Typically, a low value is acceptable
# as even older extents will continue to fill as training progresses.
#
HashMaxSeek             10

#
# HashConcurrentUser: If you are using a single, stateful hash database in
# daemon mode, specifying a concurrent user below will cause the user to be
# permanently mapped into memory and shared via rwlocks. This is very fast and
# very cool if you are running a "userless" relay appliance.
#
#HashConcurrentUser     user

#
# HashConnectionCache: If running in daemon mode, this is the max # of
# concurrent connections that will be supported. NOTE: If you are using
# HashConcurrentUser, this option is ignored, as all connections are read-
# write locked instead of mutex locked.
#
HashConnectionCache     10

# -- LDAP -

#
```

```
# LDAP: Perform various LDAP functions depending on LDAPMode variable.
# Presently, the only mode supported is 'verify', which will verify the
# existence of an unknown user in LDAP prior to creating them as a new user in
# the system.  This is useful on some systems acting as gateway machines.
#
#LDAPMode       verify
#LDAPHost       ldaphost.mydomain.com
#LDAPFilter     "(mail=%u)"
#LDAPBase       ou=people,dc=domain,dc=com

# -- Profiles --

#
# You can specify multiple storage profiles, and specify the server to
# use on the commandline with --profile. For example:
#
#Profile DECAlpha
#MySQLServer.DECAlpha   10.0.0.1
#MySQLPort.DECAlpha     3306
#MySQLUser.DECAlpha     dspam
#MySQLPass.DECAlpha     changeme
#MySQLDb.DECAlpha       dspam
#MySQLCompress.DECAlpha true
#
#Profile Sun420R
#MySQLServer.Sun420R    10.0.0.2
#MySQLPort.Sun420R      3306
#MySQLUser.Sun420R      dspam
#MySQLPass.Sun420R      changeme
#MySQLDb.Sun420R        dspam
#MySQLCompress.Sun420R  false
#
#DefaultProfile DECAlpha

#
# If you're using storage profiles, you can set failovers for each profile.
# Of course, if you'll be failing over to another database, that database
# must have the same information as the first. If you're using a global
# database with no training, this should be relatively simple. If you're
# configuring per-user data, however, you'll need to set up some type of
# replication between databases.
#
#Failover.DECAlpha      SUN420R
#Failover.Sun420R       DECAlpha

# If the storage fails, the agent will follow each profile's failover up to
# a maximum number of failover attempts. This should be set to a maximum of
# the number of profiles you have, otherwise the agent could loop and try
# the same profile multiple times (unless this is your desired behavior).
#
#FailoverAttempts       1

#
# Ignored headers: If DSPAM is behind other tools which may add a header to
# incoming emails, it may be beneficial to ignore these headers - especially
# if they are coming from another spam filter. If you are _not_ using one of
# these tools, however, leaving the appropriate headers commented out will
# allow DSPAM to use them as telltale signs of forged email.
#
#IgnoreHeader X-Spam-Status
#IgnoreHeader X-Spam-Scanned
#IgnoreHeader X-Virus-Scanner-Result

#
# Lookup: Perform lookups on streamlined blackhole list servers (see
# http://www.nuclearelephant.com/projects/sbl/). The streamlined blacklist
# server is machine-automated, unsupervised blacklisting system designed to
# provide real-time and highly accurate blacklisting based on network spread.
```

```
# When performing a lookup, DSPAM will automatically learn the inbound message
# as spam if the source IP is listed. Until an official public RABL server is
# available, this feature is only useful if you are running your own
# streamlined blackhole list server for internal reporting among multiple mail
# servers. Provide the name of the lookup zone below to use.
#
# This function performs standard reverse-octet.domain lookups, and while it
# will function with many RBLs, it's strongly discouraged to use those
# maintained by humans as they're often inaccurate and could hurt filter
# learning and accuracy.
#
#Lookup "sbl.yourdomain.com"

#
# RBLInoculate: If you want to inoculate the user from RBL'd messages it would
# have otherwise missed, set this to on.
#
#RBLInoculate off

#
# Notifications: Enable the sending of notification emails to users (first
# message, quarantine full, etc.)
#
Notifications   off

#
# Purge configuration: Set dspam_clean purge default options, if not otherwise
# specified on the commandline
#
PurgeSignatures 14          # Stale signatures
PurgeNeutral    90          # Tokens with neutralish probabilities
PurgeUnused     90          # Unused tokens
PurgeHapaxes    30          # Tokens with less than 5 hits (hapaxes)
PurgeHits1S     15          # Tokens with only 1 spam hit
PurgeHits1I     15          # Tokens with only 1 innocent hit

#
# Purge configuration for SQL-based installations using purge.sql
#
#PurgeSignature off # Specified in purge.sql
#PurgeNeutral   90
#PurgeUnused    off # Specified in purge.sql
#PurgeHapaxes   off # Specified in purge.sql
#PurgeHits1S    off # Specified in purge.sql
#PurgeHits1I    off # Specified in purge.sql

#
# Local Mail Exchangers: Used for source address tracking, tells DSPAM which
# mail exchangers are local and therefore should be ignored in the Received:
# header when tracking the source of an email. Note: you should use the address
# of the host as appears between brackets [ ] in the Received header.
#
LocalMX 127.0.0.1

#
# Logging: Disabling logging for users will make usage graphs unavailable to
# them. Disabling system logging will make admin graphs unavailable.
#
SystemLog on
UserLog   on

#
# TrainPristine: for systems where the original message remains server side
# and can therefore be presented in pristine format for retraining. This option
# will cause DSPAM to cease all writing of signatures and DSPAM headers to the
# message, and deliver the message in as pristine format as possible. This mode
# REQUIRES that the original message in its pristine format (as of delivery)
# be presented for retraining, as in the case of webmail, imap, or other
```

```
# applications where the message is actually kept server-side during reading,
# and is preserved. DO NOT use this switch unless the original message can be
# presented for retraining with the ORIGINAL HEADERS and NO MODIFICATIONS.
#
# NOTE: You can't use this setting with dspam_trian; if you're going to use it,
#       wait until after you train any corpora.
#
#TrainPristine on

#
# Opt: in or out; determines DSPAM's default filtering behavior. If this value
# is set to in, users must opt-in to filtering by dropping a .dspam file in
# /var/dspam/opt-in/user.dspam (or if you have homedirs configured, a .dspam
# folder in their home directory).  The default is opt-out, which means all
# users will be filtered unless a .nodspam file is dropped in
# /var/dspam/opt-out/user.nodspam
#
#
Opt out

#
# TrackSources: specify which (if any) source addresses to track and report
# them to syslog (mail.info). This is useful if you're running a firewall or
# blacklist and would like to use this information. Spam reporting also drops
# RABL blacklist files (see http://www.nuclearelephant.com/projects/rabl/).
#
TrackSources spam nonspam

#
# ParseToHeaders: In lieu of setting up individual aliases for each user,
# DSPAM can be configured to automatically parse the To: address for spam and
# false positive forwards. From there, it can be configured to either set the
# DSPAM user based on the username specified in the header and/or change the
# training class and source accordingly. The options below can be used to
# customize most common types of header parsing behavior to avoid the need for
# multiple aliases, or if using LMTP, aliases entirely..
#
# ParseToHeader: Parse the To: headers of an incoming message. This must be
#                set to 'on' to use either of the following features.
#
#                set to 'on' to use either of the following features.
#
# ChangeModeOnParse: Automatically change the class (to spam or innocent)
#   depending on whether spam- or notspam- was specified, and change the source
#   to 'error'. This is convenient if you're not using aliases at all, but
#   are delivering via LMTP.
#
# ChangeUserOnParse: Automatically change the username to match that specified
#   in the To: header. For example, spam-bob@domain.tld will set the username
#   to bob, ignoring any --user passed in. This may not always be desirable if
#   you are using virtual email addresses as usernames. Options:
#     on or user        take the portion before the @ sign only
#     full              take everything after the initial {spam,notspam}-.
#
ParseToHeaders on
ChangeModeOnParse on
ChangeUserOnParse off

#
# Broken MTA Options: Some MTAs don't support the proper functionality
# necessary. In these cases you can activate certain features in DSPAM to
# compensate. 'returnCodes' causes DSPAM to return an exit code of 99 if
# the message is spam, 0 if not, or a negative code if an error has occured.
# Specifying 'case' causes DSPAM to force the input usernames to lowercase.
# Spceifying 'lineStripping' causes DSPAM to strip ^M's from messages passed
# in.
#
#Broken returnCodes
```

```
Broken case
#Broken lineStripping

#
# MaxMessageSize: You may specify a maximum message size for DSPAM to process.
# If the message is larger than the maximum size, it will be delivered
# without processing. Value is in bytes.
#
#MaxMessageSize 4194304

#
# Virus Checking: If you are running clamd, DSPAM can perform stream-based
# virus checking using TCP. Uncomment the values below to enable virus
# checking.
#
# ClamAVResponse: reject (reject or drop the message with a permanent failure)
#                 accept (accept the message and quietly drop the message)
#                 spam   (treat as spam and quarantine/tag/whatever)
#
ClamAVPort      3310
ClamAVHost      127.0.0.1
ClamAVResponse  reject

# -- CLIENT / SERVER --

#
# Daemonized Server: If you are running DSPAM as a daemonized server using
# --daemon, the following parameters will override the default. Use the
# ServerPass option to set up accounts for each client machine. The DSPAM
# server will process and deliver the message based on the parameters
# specified. If you want the client machine to perform delivery, use
# the --stdout option in conjunction with a local setup.
#
ServerPort              10024
ServerQueueSize         32
ServerPID               /var/dspam/dspam.pid

#
# ServerMode specifies the type of LMTP server to start. This can be one of:
#     dspam: DSPAM-proprietary DLMTP server, for communicating with dspamc
#  standard: Standard LMTP server, for communicating with Postfix or other MTA
#      auto: Speak both DLMTP and LMTP; auto-detect by ServerPass.IDENT
#
ServerMode auto

# If supporting DLMTP (dspam) mode, dspam clients will require authentication
# as they will be passing in parameters. The idents below will be used to
# determine which clients will be speaking DLMTP, so if you will be using
# both LMTP and DLMTP from the same host, be sure to use something other
# than the server's hostname below (which will be sent by the MTA during a
# standard LMTP LHLO).
#
#ServerPass.Relay1      "secret"
#ServerPass.Relay2      "password"

# If supporting standard LMTP mode, server parameters will need to be specified
# here, as they will not be passed in by the mail server. The ServerIdent
# specifies the 250 response code ident sent back to connecting clients and
# should be set to the hostname of your server, or an alias.
#
# NOTE: If you specify --user in ServerParameters, the RCPT TO will be
#       used only for delivery, and not set as the active user for processing.
#
ServerParameters        "--deliver=innocent -d %u"
ServerIdent             "localhost.localdomain"

# If you wish to use a local domain socket instead of a TCP socket, uncomment
# the following. It is strongly recommended you use local domain sockets if
```

```
# you are running the client and server on the same machine, as it eliminates
# much of the bandwidth overhead.
#
#ServerDomainSocketPath  "/tmp/dspam.sock"

#
# Client Mode: If you are running DSPAM in client/server mode, uncomment and
# set these variables. A ClientHost beginning with a / will be treated as
# a domain socket.
#
#ClientHost     /tmp/dspam.sock
#ClientIdent    "secret@Relay1"
#
#ClientHost     127.0.0.1
#ClientPort     24
#ClientIdent    "secret@Relay1"

# RABLQueue: Touch files in the RABL queue
# If you are a reporting streamlined blackhole list participant, you can
# touch ip addresses within the directory the rabl_client process is watching.
#
#RABLQueue      /var/spool/rabl

# DataSource: If you are using any type of data source that does not include
# email-like headers (such as documents), uncomment the line below. This
# will cause the entire input to be treated like a message "body"
#
#DataSource      document

# ProcessorWordFrequency: By default, words are only counted once per message.
# If you are classifying large documents, however, you may wish to count once
# per occurrence instead.
#
#ProcessorWordFrequency  occurrence

# ProcessorURLContext: By default, a URL context is generated for URLs, which
# records their tokens as separate from words found in documents. To use
# URL tokens in the same context as words, turn this feature off.
#
ProcessorURLContext on

# ProcessorBias: Bias causes the filter to lean more toward 'innocent', and
# usually greatly reduces false positives. It is the default behavior of
# most Bayesian filters (including dspam).
#
# NOTE: You probably DONT want this if you're using Markovian Weighting, unless
# you are paranoid about false positives.
#
ProcessorBias off

## EOF
```

- Set ownership of the configuration file

```
# chown dspam.dspam dspam.conf
```

Runlevel config

- Create file for runlevel and set it to executable.

```
# cd /etc/init.d/
# vi dspam
```

```
#!/bin/sh
#
#     Template SUSE system startup script for example service/daemon FOO
#     Copyright (C) 1995--2005  Kurt Garloff, SUSE / Novell Inc.
#
#     This library is free software; you can redistribute it and/or modify it
#     under the terms of the GNU Lesser General Public License as published by
#     the Free Software Foundation; either version 2.1 of the License, or (at
#     your option) any later version.
#
#     This library is distributed in the hope that it will be useful, but
#     WITHOUT ANY WARRANTY; without even the implied warranty of
#     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
#     Lesser General Public License for more details.
#
#     You should have received a copy of the GNU Lesser General Public
#     License along with this library; if not, write to the Free Software
#     Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307,
#     USA.
#
# /etc/init.d/dspam
#   and its symbolic link
# /usr/sbin/rcdspam
#
# Template system startup script for some example service/daemon FOO
#
# LSB compatible service control script; see http://www.linuxbase.org/spec/
#
# Note: This template uses functions rc_XXX defined in /etc/rc.status on
# UnitedLinux/SUSE/Novell based Linux distributions. If you want to base your
# script on this template and ensure that it works on non UL based LSB
# compliant Linux distributions, you either have to provide the rc.status
# functions from UL or change the script to work without them.
# See skeleton.compat for a template that works with other distros as well.
#
### BEGIN INIT INFO
#Provides:         DSPAM
#Required-Start:    $syslog $remote_fs
#Should-Start: $time ypbind sendmail
#Required-Stop:     $syslog $remote_fs
#Should-Stop: $time ypbind sendmail
#Default-Start:     3 5
#Default-Stop:      0 1 2 6
#Short-Description: DSPAM Antispam Engine
#Description:       Start the DSPAM antispam Engine
### END INIT INFO
#
# Any extensions to the keywords given above should be preceeded by
# X-VendorTag- (X-UnitedLinux- X-SuSE- for us) according to LSB.
#
# Notes on Required-Start/Should-Start:
# * There are two different issues that are solved by Required-Start
#    and Should-Start
# (a) Hard dependencies: This is used by the runlevel editor to determine
#     which services absolutely need to be started to make the start of
#     this service make sense. Example: nfsserver should have
#     Required-Start: $portmap
#     Also, required services are started before the dependent ones.
#     The runlevel editor will warn about such missing hard dependencies
#     and suggest enabling. During system startup, you may expect an error,
#     if the dependency is not fulfilled.
# (b) Specifying the init script ordering, not real (hard) dependencies.
#     This is needed by insserv to determine which service should be
#     started first (and at a later stage what services can be started
#     in parallel). The tag Should-Start: is used for this.
#     It tells, that if a service is available, it should be started
#     before. If not, never mind.
# * When specifying hard dependencies or ordering requirements, you can
```

```
#   use names of services (contents of their Provides: section)
#   or pseudo names starting with a $. The following ones are available
#   according to LSB (1.1):
#     $local_fs           all local file systems are mounted
#                         (most services should need this!)
#     $remote_fs          all remote file systems are mounted
#                         (note that /usr may be remote, so
#                          many services should Require this!)
#     $syslog                   system logging facility up
#     $network            low level networking (eth card, ...)
#     $named              hostname resolution available
#     $netdaemons         all network daemons are running
#   The $netdaemons pseudo service has been removed in LSB 1.2.
#   For now, we still offer it for backward compatibility.
#   These are new (LSB 1.2):
#     $time               the system time has been set correctly
#     $portmap            SunRPC portmapping service available
#   UnitedLinux extensions:
#     $ALL                indicates that a script should be inserted
#                         at the end
# * The services specified in the stop tags
#   (Required-Stop/Should-Stop)
#   specify which services need to be still running when this service
#   is shut down. Often the entries there are just copies or a subset
#   from the respective start tag.
# * Should-Start/Stop are now part of LSB as of 2.0,
#   formerly SUSE/Unitedlinux used X-UnitedLinux-Should-Start/-Stop.
#   insserv does support both variants.
# * X-UnitedLinux-Default-Enabled: yes/no is used at installation time
#   (%fillup_and_insserv macro in %post of many RPMs) to specify whether
#   a startup script should default to be enabled after installation.
#   It's not used by insserv.
#
# Note on runlevels:
# 0 - halt/poweroff                   6 - reboot
# 1 - single user               2 - multiuser without network exported
# 3 - multiuser w/ network (text mode)  5 - multiuser w/ network and X11 (xdm)
#
# Note on script names:
# http://www.linuxbase.org/spec/refspecs/LSB_1.3.0/gLSB/gLSB/scrptnames.html
# A registry has been set up to manage the init script namespace.
# http://www.lanana.org/
# Please use the names already registered or register one or use a
# vendor prefix.


# Check for missing binaries (stale symlinks should not happen)
# Note: Special treatment of stop for LSB conformance
DSPAM_BIN=/usr/local/bin/dspam
test -x $DSPAM_BIN || { echo "$DSPAM_BIN not installed";
      if [ "$1" = "stop" ]; then exit 0;
      else exit 5; fi; }

# Check for existence of needed config file and read it
DSPAM_CONFIG=/usr/local/etc/dspam.conf
test -r $DSPAM_CONFIG || { echo "$DSPAM_CONFIG not existing";
      if [ "$1" = "stop" ]; then exit 0;
      else exit 6; fi; }

# Read config
# $DSPAM_CONFIG

# Source LSB init functions
# providing start_daemon, killproc, pidofproc,
# log_success_msg, log_failure_msg and log_warning_msg.
# This is currently not used by UnitedLinux based distributions and
# not needed for init scripts for UnitedLinux only. If it is used,
# the functions from rc.status should not be sourced or used.
```

```
#. /lib/lsb/init-functions

# Shell functions sourced from /etc/rc.status:
#      rc_check         check and set local and overall rc status
#      rc_status        check and set local and overall rc status
#      rc_status -v     be verbose in local rc status and clear it afterwards
#      rc_status -v -r  ditto and clear both the local and overall rc status
#      rc_status -s     display "skipped" and exit with status 3
#      rc_status -u     display "unused" and exit with status 3
#      rc_failed        set local and overall rc status to failed
#      rc_failed <num>  set local and overall rc status to <num>
#      rc_reset         clear both the local and overall rc status
#      rc_exit          exit appropriate to overall rc status
#      rc_active        checks whether a service is activated by symlinks
. /etc/rc.status

# Reset status of this service
rc_reset

# Return values acc. to LSB for all commands but status:
# 0     - success
# 1       - generic or unspecified error
# 2       - invalid or excess argument(s)
# 3       - unimplemented feature (e.g. "reload")
# 4       - user had insufficient privileges
# 5       - program is not installed
# 6       - program is not configured
# 7       - program is not running
# 8--199  - reserved (8--99 LSB, 100--149 distrib, 150--199 appl)
#
# Note that starting an already running service, stopping
# or restarting a not-running service as well as the restart
# with force-reload (in case signaling is not supported) are
# considered a success.

case "$1" in
    start)
      echo -n "Starting DSPAM Antispam Engine "
      ## Start daemon with startproc(8). If this fails
      ## the return value is set appropriately by startproc.
      /sbin/startproc -u dspam $DSPAM_BIN --daemon

      # Remember status and be verbose
      rc_status -v
      ;;
    stop)
      echo -n "Shutting down DSPAM Antispam Engine "
      ## Stop daemon with killproc(8) and if this fails
      ## killproc sets the return value according to LSB.

      /sbin/killproc -TERM $DSPAM_BIN

      # Remember status and be verbose
      rc_status -v
      ;;
    try-restart|condrestart)
      ## Do a restart only if the service was active before.
      ## Note: try-restart is now part of LSB (as of 1.9).
      ## RH has a similar command named condrestart.
      if test "$1" = "condrestart"; then
            echo "${attn} Use try-restart ${done}(LSB)${attn} rather than
condrestart ${warn}(RH)${norm}"
      fi
      $0 status
      if test $? = 0; then
            $0 restart
      else
            rc_reset     # Not running is not a failure.
```

```
    fi
    # Remember status and be quiet
    rc_status
    ;;
  restart)
    ## Stop the service and regardless of whether it was
    ## running or not, start it again.
    $0 stop
    $0 start

    # Remember status and be quiet
    rc_status
    ;;
  force-reload)
    ## Signal the daemon to reload its config. Most daemons
    ## do this on signal 1 (SIGHUP).
    ## If it does not support it, restart the service if it
    ## is running.

    echo -n "Reload service DSPAM "
    ## if it supports it:
    /sbin/killproc -HUP $DSPAM_BIN
    touch /var/dspam/dspam.pid
    rc_status -v

    ## Otherwise:
    #$0 try-restart
    #rc_status
    ;;
  reload)
    ## Like force-reload, but if daemon does not support
    ## signaling, do nothing (!)

    # If it supports signaling:
    echo -n "Reload service DSPAM "
    /sbin/killproc -HUP $DSPAM_BIN
    touch /var/dspam/dspam.pid
    rc_status -v

    ## Otherwise if it does not support reload:
    #rc_failed 3
    #rc_status -v
    ;;
  status)
    echo -n "Checking for service DSPAM "
    ## Check status with checkproc(8), if process is running
    ## checkproc will return with exit status 0.

    # Return value is slightly different for the status command:
    # 0 - service up and running
    # 1 - service dead, but /var/run/  pid  file exists
    # 2 - service dead, but /var/lock/ lock file exists
    # 3 - service not running (unused)
    # 4 - service status unknown :-(
    # 5--199 reserved (5--99 LSB, 100--149 distro, 150--199 appl.)

    # NOTE: checkproc returns LSB compliant status values.
    /sbin/checkproc $DSPAM_BIN
    # NOTE: rc_status knows that we called this init script with
    # "status" option and adapts its messages accordingly.
    rc_status -v
    ;;
  probe)
    ## Optional: Probe for the necessity of a reload, print out the
    ## argument to this init script which is required for a reload.
    ## Note: probe is not (yet) part of LSB (as of 1.9)

    test /usr/local/etc/dspam.conf -nt /var/dspam/dspam.pid && echo reload
```

```
      ;;
    *)
      echo "Usage: $0 {start|stop|status|try-restart|restart|force-reload|reload|
probe}"
      exit 1
      ;;
esac
rc_exit
```

```
# chmod +x dspam
```

- Create a symlink in /usr/sbin

```
# cd /usr/sbin
# ln -s /etc/init.d/dspam rcdspam
```

WebUI Graphics (GD graphics)

- Install gd-devel (dependant packages will also be installed)

```
# yast –i gd-devel
```

- Load cpan

```
# cpan
```

Run through the 1st time setup :

- Are you ready for manual configuration? [yes] * just hit Enter
- CPAN build and cache directory? [/root/.cpan] **/var/work/cpan**
- Cache size for build directory (in MB)? [10] * just hit Enter
- Perform cache scanning (atstart or never)? [atstart] * just hit Enter
- Cache metadata (yes/no)? [yes] * just hit Enter
- Your terminal expects ISO-8859-1 (yes/no)? [yes] * just hit Enter
- File to save your history? [/var/work/cpan/histfile] * just hit Enter
- Number of lines to save? [100] * just hit Enter
- Policy on building prerequisites (follow, ask or ignore)? [ask] * just hit Enter
- Where is your gzip program? [/usr/bin/gzip] * just hit Enter
- Where is your tar program? [/bin/tar] * just hit Enter
- Where is your unzip program? [/usr/bin/unzip] * just hit Enter
- Where is your make program? [/usr/bin/make] * just hit Enter
- Where is your lynx program? []* just hit Enter (ignore warning)
- Where is your wget program? [/usr/bin/wget] * just hit Enter
- Where is your ncftpget program? []* just hit Enter (ignore warning)
- Where is your ncftp program? []* just hit Enter (ignore warning)
- Where is your ftp program? [/usr/bin/ftp] * just hit Enter
- Where is your gpg program? [/usr/bin/gpg] * just hit Enter
- What is your favorite pager program? [less] * just hit Enter
- What is your favorite shell? [/bin/bash] * just hit Enter

- Parameters for the 'perl Makefile.PL' command? [] * just hit Enter
- Parameters for the 'make' command? [] * just hit Enter
- Parameters for the 'make install' command? [] * just hit Enter
- Timeout for inactivity during Makefile.PL? [0] * just hit Enter
- Your ftp_proxy? * just hit Enter
- Your http_proxy? * just hit Enter
- Your no_proxy? * just hit Enter
- Select your continent (or several nearby continents) [4] (Europe for me)
- Select your country (or several nearby countries) [] (enter Space RETURN , I need Netherlands, and it's on the next page)
- Select your country (or several nearby countries) [] **19**
- Select the mirrors you want , I chose xs4all.nl [**3**]
- Enter another URL or RETURN to quit: []* just hit Enter

Now that you have your cpan> prompt, start installing GD modules

```
# install GD
# install GD::Text
# install GD::Graph
# install GD::Graph3d
# quit
```

Training dspam

To train dspam at first i use a public corpus of ham/spam.

- Create a workingdir, download the files and unpack

```
# cd /var/work/source
# mkdir ../publiccorpus
# wget http://spamassassin.apache.org/publiccorpus/20021010_easy_ham.tar.bz2
# wget http://spamassassin.apache.org/publiccorpus/20021010_hard_ham.tar.bz2
# wget http://spamassassin.apache.org/publiccorpus/20021010_spam.tar.bz2
# wget http://spamassassin.apache.org/publiccorpus/20030228_easy_ham.tar.bz2
# wget http://spamassassin.apache.org/publiccorpus/20030228_easy_ham_2.tar.bz2
# wget http://spamassassin.apache.org/publiccorpus/20030228_hard_ham.tar.bz2
# wget http://spamassassin.apache.org/publiccorpus/20030228_spam_2.tar.bz2
# wget http://spamassassin.apache.org/publiccorpus/20050311_spam_2.tar.bz2
#  cd ../publiccorpus
# for i in ../source/*.bz2; do tar -xjvf $i; done
```

In the past i used a perl-script called 'publiccorpus.pl' that was available/downloadable from dspam's site, but not anymore.
I still have it, so below is the code to create it.

```
# vi publiccorpus.pl
```

```
#!/usr/bin/perl

use strict;
use vars qw { @archives $user };

@archives = ( "easy_ham_2", "easy_ham", "hard_ham", "spam", "spam_2" );

print "SpamAssassin Public Corpus Trainer v0.1.0\n\n";
$user = shift;
if ($user eq "") {
  print "Syntax: $0 [username]\n";
  exit(-1);
}

foreach(@archives) {
  print   "Searching for corpus $_ ...\n";
  if (-d $_) {
    print "...found it!\n";
    print "Training with corpus $_ ...";
    &Train($user, $_);
    print "...done!\n";
  } else {
    print "...not found.\n";
  }
}

print "Training complete.\n";
print "Now run \"dspam_clean -p0 $user\" to purge uninteresting data\n";

sub Train {
  my($user, $corpus) = @_;
  my(@files, $file, $cmd, $class);

  opendir(DIR, "$corpus");
  @files = grep(!/^\.\.?$/, readdir(DIR));
  closedir(DIR);

  if ($corpus =~ /ham/) {
    $class = "innocent";
  } elsif ($corpus =~ /spam/) {
    $class = "spam";
  } else {
    print "Unable to determine whether $corpus is ham or spam. Skipping.\n";
    return;
  }

  foreach $file (@files) {
    my($ret);
    next if ($file eq "cmds");
    $cmd = "dspam --user $user --class=$class --source=corpus < $corpus/$file";
    $ret = system($cmd);
    print "Command returned error $ret: $cmd\n" if ($ret);
  }

  return;
}
```

```
# chmod +x publiccorpus.pl
```

Now train up an administrative (existing) email, ie postmaster@comsolve.nl and get yourself a cup of coffee (this takes some time)

```
# ./publiccorpus.pl postmaster@comsolve.nl
```

After the training is done run a cleanup to purge uninteresting data

```
# dspam_clean -p0 postmaster@comsolve.nl
```

As the training is done as root user, it will break some ownership settings, to correct it :

```
# cd /var
# chown –R dspam.dspam dspam
```

Finalizing setup

To create an administrative user in the webinterface able to set preferences and switch to all quarantine boxes edit /srv/www/dspam/admins

```
# cd /srv/www/dspam
# vi admins
```

```
        postmaster@comsolve.nl
```

To create a reference for all new users in recognition untill they have enough mail passed by their own create a globalgroup.

```
# cd /var/dspam
# vi group
```

```
      globalgroup:classification:*postmaster@comsolve.nl
```

Set the correct ownership

```
# chown dspam.dspam group
```

To have the DSPAM box send Notifications to users about a quarantine getting full, etc.

Edit /usr/local/etc/dspam.conf and replace Notifications from off to on

```
# cd /usr/local/etc
# vi dspam.conf
```

```
#
# Notifications: Enable the sending of notification emails to users (first
# message, quarantine full, etc.)
#
Notifications       on
```

- Create appropriate directory in /var/dspam

```
# cd /var/dspam
```

```
# mkdir txt
# cp /var/work/compile/dspam-3.8.0/txt/* ./txt
# rm –f txt/Makefile*
# chown –R dspam.dspam txt
```

- Edit the txtfiles to match your organisation / dspam WebUI url.

- Start up the DSPAM Daemon & set its runlevel

```
# rcdspam start
# chkconfig dspam on
```

- Changing logrotation for the maillog, so we will have daily log (good for reporting)

```
# cd /etc/logrotate.d/
# vi syslog
```

```
/var/log/mail /var/log/mail.info /var/log/mail.warn /var/log/mail.err {
    compress
    delaycompress
    daily
#    dateext
    maxage 365
    rotate 99
    missingok
    notifempty
#    size +4096k
    create 640 root root
    sharedscripts
    postrotate
        /etc/init.d/syslog reload
    endscript
}
```

Enhancing it even further

Reporting postfix stats with pflogsumm.

Pflogsumm is a perlscript that reads postfix logs and creates a statistics mail when run.

- Get the source, unpack & place files in appropriate places

```
# cd /var/work/source
# wget http://jimsun.linxnet.com/downloads/pflogsumm-1.1.0.tar.gz
# cd ../compile
# tar -zxvf ../source/pflogsumm-1.1.0.tar.gz
# cp pflogsumm-1.1.0/pflogsumm.pl /usr/local/bin/pflogsumm
# chown 755 /usr/local/bin/pflogsumm/pflogsumm.pl
# cp pflogsumm-1.1.0/pflogsumm.1 /usr/local/man/man1/pflogsumm.1
# chmod 644 /usr/local/man/man1/pflogsumm.1
```

- Get the Date::Calc module from cpan if not present

```
# cpan
#       install Date::Calc
```

- Add a cronjob to run it's daily stats

```
# crontab –e
```

```
0 1 * * * /usr/local/bin/pflogsumm -d yesterday /var/log/mail.1 2>&1 | mail -s
"`uname -n` Daily postfix stats" postmaster
```

Adding greylisting with Postgrey

Greylisting is a nice feature for stopping off those pesky email bots.

- Get the source

```
# cd /var/work/source
# wget http://postgrey.schweikert.ch/pub/postgrey-1.31.tar.gz
# cd ../compile
# tar -zxvf ../source/postgrey-1.31.tar.gz
#  cd postgrey-1.31
# mkdir /etc/postgrey
# cp -r . /etc/postgrey
# cp postgrey_whitelist_* /etc/postfix/
```

- Get needed modules from cpan

```
# cpan
        install Net::Server
        install IO::Multiplex
        install BerkeleyDB
# exit
```

- Add a group and user for postgrey, and create appropriate dirs

```
# groupadd postgrey
# useradd -u 2002 -g postgrey -d /etc/postgrey -c "Postgrey service" postgrey
# mkdir /var/spool/postfix/postgrey
# chown postgrey /var/spool/postfix/postgrey
```

- Create a runscript for the postgrey services

cd /etc/init.d
vi postgrey

```
#!/bin/sh
#
#     Template SUSE system startup script for example service/daemon FOO
#     Copyright (C) 1995--2005  Kurt Garloff, SUSE / Novell Inc.
```

```
#
#     This library is free software; you can redistribute it and/or modify it
#     under the terms of the GNU Lesser General Public License as published by
#     the Free Software Foundation; either version 2.1 of the License, or (at
#     your option) any later version.
#
#     This library is distributed in the hope that it will be useful, but
#     WITHOUT ANY WARRANTY; without even the implied warranty of
#     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
#     Lesser General Public License for more details.
#
#     You should have received a copy of the GNU Lesser General Public
#     License along with this library; if not, write to the Free Software
#     Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307,
#     USA.
#
# /etc/init.d/postgrey
#   and its symbolic link
# /(usr/)sbin/rcpostgrey
#
# Template system startup script for some example service/daemon Postgrey
#
# LSB compatible service control script; see http://www.linuxbase.org/spec/
#
# Note: This template uses functions rc_XXX defined in /etc/rc.status on
# UnitedLinux/SUSE/Novell based Linux distributions. If you want to base your
# script on this template and ensure that it works on non UL based LSB
# compliant Linux distributions, you either have to provide the rc.status
# functions from UL or change the script to work without them.
# See skeleton.compat for a template that works with other distros as well.
#
### BEGIN INIT INFO
# Provides:          Postgrey
# Required-Start:    $syslog $remote_fs
# Should-Start: $time ypbind sendmail
# Required-Stop:     $syslog $remote_fs
# Should-Stop: $time ypbind sendmail
# Default-Start:     3 5
# Default-Stop:      0 1 2 6
# Short-Description: Postfix Greylisting Policy Server
# Description:       Postfix Greylisting Policy Server
### END INIT INFO
#
# Any extensions to the keywords given above should be preceeded by
# X-VendorTag- (X-UnitedLinux- X-SuSE- for us) according to LSB.
#
# Notes on Required-Start/Should-Start:
# * There are two different issues that are solved by Required-Start
#    and Should-Start
# (a) Hard dependencies: This is used by the runlevel editor to determine
#     which services absolutely need to be started to make the start of
#     this service make sense. Example: nfsserver should have
#     Required-Start: $portmap
#     Also, required services are started before the dependent ones.
#     The runlevel editor will warn about such missing hard dependencies
#     and suggest enabling. During system startup, you may expect an error,
#     if the dependency is not fulfilled.
# (b) Specifying the init script ordering, not real (hard) dependencies.
#     This is needed by insserv to determine which service should be
#     started first (and at a later stage what services can be started
#     in parallel). The tag Should-Start: is used for this.
#     It tells, that if a service is available, it should be started
#     before. If not, never mind.
# * When specifying hard dependencies or ordering requirements, you can
#   use names of services (contents of their Provides: section)
#   or pseudo names starting with a $. The following ones are available
#   according to LSB (1.1):
#     $local_fs           all local file systems are mounted
```

```
#                           (most services should need this!)
#     $remote_fs            all remote file systems are mounted
#                           (note that /usr may be remote, so
#                            many services should Require this!)
#     $syslog                     system logging facility up
#     $network              low level networking (eth card, ...)
#     $named                hostname resolution available
#     $netdaemons           all network daemons are running
#   The $netdaemons pseudo service has been removed in LSB 1.2.
#   For now, we still offer it for backward compatibility.
#   These are new (LSB 1.2):
#     $time                 the system time has been set correctly
#     $portmap              SunRPC portmapping service available
#   UnitedLinux extensions:
#     $ALL                  indicates that a script should be inserted
#                           at the end
# * The services specified in the stop tags
#   (Required-Stop/Should-Stop)
#   specify which services need to be still running when this service
#   is shut down. Often the entries there are just copies or a subset
#   from the respective start tag.
# * Should-Start/Stop are now part of LSB as of 2.0,
#   formerly SUSE/Unitedlinux used X-UnitedLinux-Should-Start/-Stop.
#   insserv does support both variants.
# * X-UnitedLinux-Default-Enabled: yes/no is used at installation time
#   (%fillup_and_insserv macro in %post of many RPMs) to specify whether
#   a startup script should default to be enabled after installation.
#   It's not used by insserv.
#
# Note on runlevels:
# 0 - halt/poweroff                   6 - reboot
# 1 - single user               2 - multiuser without network exported
# 3 - multiuser w/ network (text mode)  5 - multiuser w/ network and X11 (xdm)
#
# Note on script names:
# http://www.linuxbase.org/spec/refspecs/LSB_1.3.0/gLSB/gLSB/scrptnames.html
# A registry has been set up to manage the init script namespace.
# http://www.lanana.org/
# Please use the names already registered or register one or use a
# vendor prefix.


# Check for missing binaries (stale symlinks should not happen)
# Note: Special treatment of stop for LSB conformance
POSTGREY_BIN=/etc/postgrey/postgrey
test -x $POSTGREY_BIN || { echo "$POSTGREY_BIN not installed";
      if [ "$1" = "stop" ]; then exit 0;
      else exit 5; fi; }

# Source LSB init functions
# providing start_daemon, killproc, pidofproc,
# log_success_msg, log_failure_msg and log_warning_msg.
# This is currently not used by UnitedLinux based distributions and
# not needed for init scripts for UnitedLinux only. If it is used,
# the functions from rc.status should not be sourced or used.
#. /lib/lsb/init-functions

# Shell functions sourced from /etc/rc.status:
#      rc_check         check and set local and overall rc status
#      rc_status        check and set local and overall rc status
#      rc_status -v     be verbose in local rc status and clear it afterwards
#      rc_status -v -r  ditto and clear both the local and overall rc status
#      rc_status -s     display "skipped" and exit with status 3
#      rc_status -u     display "unused" and exit with status 3
#      rc_failed        set local and overall rc status to failed
#      rc_failed <num>  set local and overall rc status to <num>
#      rc_reset         clear both the local and overall rc status
#      rc_exit          exit appropriate to overall rc status
```

```
#      rc_active        checks whether a service is activated by symlinks
. /etc/rc.status

# Reset status of this service
rc_reset

# Return values acc. to LSB for all commands but status:
# 0     - success
# 1       - generic or unspecified error
# 2       - invalid or excess argument(s)
# 3       - unimplemented feature (e.g. "reload")
# 4       - user had insufficient privileges
# 5       - program is not installed
# 6       - program is not configured
# 7       - program is not running
# 8--199  - reserved (8--99 LSB, 100--149 distrib, 150--199 appl)
#
# Note that starting an already running service, stopping
# or restarting a not-running service as well as the restart
# with force-reload (in case signaling is not supported) are
# considered a success.

case "$1" in
    start)
      echo -n "Starting Postgrey service "
      ## Start daemon with startproc(8). If this fails
      ## the return value is set appropriately by startproc.
      /sbin/startproc -u postgrey $POSTGREY_BIN --user=postgrey --group=postgrey --
inet=10023 --daemonize

      # Remember status and be verbose
      rc_status -v
      ;;
    stop)
      echo -n "Shutting down Postgrey service "
      ## Stop daemon with killproc(8) and if this fails
      ## killproc sets the return value according to LSB.

      /sbin/killproc -TERM $POSTGREY_BIN

      # Remember status and be verbose
      rc_status -v
      ;;
    try-restart|condrestart)
      ## Do a restart only if the service was active before.
      ## Note: try-restart is now part of LSB (as of 1.9).
      ## RH has a similar command named condrestart.
      if test "$1" = "condrestart"; then
            echo "${attn} Use try-restart ${done}(LSB)${attn} rather than
condrestart ${warn}(RH)${norm}"
      fi
      $0 status
      if test $? = 0; then
            $0 restart
      else
            rc_reset     # Not running is not a failure.
      fi
      # Remember status and be quiet
      rc_status
      ;;
    restart)
      ## Stop the service and regardless of whether it was
      ## running or not, start it again.
      $0 stop
      $0 start

      # Remember status and be quiet
      rc_status
```

```
      ;;
    force-reload)
      ## Signal the daemon to reload its config. Most daemons
      ## do this on signal 1 (SIGHUP).
      ## If it does not support it, restart the service if it
      ## is running.

      echo -n "Reload service Postgrey "
      ## if it supports it:
      /sbin/killproc -HUP $POSTGREY_BIN
      touch /var/run/postgrey.pid
      rc_status -v

      ## Otherwise:
      #$0 try-restart
      #rc_status
      ;;
    reload)
      ## Like force-reload, but if daemon does not support
      ## signaling, do nothing (!)

      # If it supports signaling:
      echo -n "Reload service Postgrey "
      /sbin/killproc -HUP $POSTGREY_BIN
      touch /var/run/postgrey.pid
      rc_status -v

      ## Otherwise if it does not support reload:
      #rc_failed 3
      #rc_status -v
      ;;
    status)
      echo -n "Checking for service Postgrey "
      ## Check status with checkproc(8), if process is running
      ## checkproc will return with exit status 0.

      # Return value is slightly different for the status command:
      # 0 - service up and running
      # 1 - service dead, but /var/run/  pid  file exists
      # 2 - service dead, but /var/lock/ lock file exists
      # 3 - service not running (unused)
      # 4 - service status unknown :-(
      # 5--199 reserved (5--99 LSB, 100--149 distro, 150--199 appl.)

      # NOTE: checkproc returns LSB compliant status values.
      /sbin/checkproc $POSTGREY_BIN
      # NOTE: rc_status knows that we called this init script with
      # "status" option and adapts its messages accordingly.
      rc_status -v
      ;;
    probe)
      ## Optional: Probe for the necessity of a reload, print out the
      ## argument to this init script which is required for a reload.
      ## Note: probe is not (yet) part of LSB (as of 1.9)

      ;;
    *)
      echo "Usage: $0 {start|stop|status|try-restart|restart|force-reload|reload|
probe}"
      exit 1
      ;;
esac
rc_exit
```

- Set the script to executable

```
# chmod +x postgrey
```

- Create a symbolic link rcpostgrey

```
# ln -s /etc/init.d/postgrey /usr/sbin/rcpostgrey
```

- Start postgrey and set it to run at boot

```
# rcpostgrey start
# chkconfig postgrey on
```

- Add config-line to postfix main.cf

```
# vi /etc/postfix/main.cf
```

```
smtpd_recipient_restrictions =
       check_sender_access hash:/etc/postfix/not_our_domain_as_sender,
       permit_mynetworks,
       reject_unauth_destination,
       reject_invalid_hostname,
       reject_unknown_recipient_domain,
       check_recipient_access hash:/etc/postfix/protect_subdomain,
       check_helo_access pcre:/etc/postfix/helo_checks,
      check_policy_service inet:127.0.0.1:10023
```

Postgrey reporting

- Add cpan modules needed

```
# cpan
       install Net::IP
       install Digest::HMAC_MD5
       install Net::DNS
# exit
```

- Create a crontab entry

```
# crontab –e
```

```
0 2 * * * /etc/postgrey/contrib/postgreyreport --nosingle_line --check_sender=mx,a
--separate_by_subnet=":==================\n"< /var/log/mail.1 2>&1 | mail -s
"`uname -n` Daily Postgrey stats" postmaster
```

Updating DSPAM

As all documents are made in a point in time, and all packages undergo changes while writing this document already some modifications were made to the WebUI and a preference setting.

Therefore I’ve included the way to update from CVS.

```
# cd /var/work/source
# cvs -z3 -d :pserver:cvs@cvs.nuclearelephant.com:/usr/local/cvsroot co dspam
# cp -r dspam/webui/cgi-bin/*.cgi /srv/www/dspam/
# cp -r dspam/webui/cgi-bin/templates/* /srv/www/dspam/templates/
# rm -rf /srv/www/dspam/templates/CVS/
# rm -f /srv/www/dspam/templates/Makefile*
# cp -r dspam/webui/htdocs/dspam* /srv/www/dspam/
# cp -r dspam/webui/htdocs/base.css /srv/www/dspam/
# vi /usr/local/etc/dspam.conf
```

Add the following directive to the appropriate section :

```
AllowOverride dailyQuarantineSummary
```

Restart/reload dspam

```
# rcdspam reload
```

phpMyAdmin

As we also want to be able to administer MySQL databases directly we'll also setup phpMyAdmin.

- Get needed modules for phpMyAdmin

```
# yast -i php5-mbstring php5-mcrypt
```

- Get the source and start configuration

```
# cd /var/work/source
# wget http://prdownloads.sourceforge.net/phpmyadmin/phpMyAdmin-2.11.5.1-all-languages.tar.gz?download
# cd /srv/wwwhtdocs
# tar -zxvf /var/work/source/phpMyAdmin-2.11.5.1-all-languages.tar.gz
# mv phpMyAdmin-2.11.5.1-all-languages phpMyAdmin
# cd phpMyAdmin
# cp config.sample.inc.php config.inc.php
```

In the config file a 'blowfish' secret is used, I used 'pwgen' to generate a random password, and use that as the blowfish secret.

```
# cd /var/work/source
# wget http://downloads.sourceforge.net/pwgen/pwgen-2.06.tar.gz?modtime=1183592957&big_mirror=0
# cd ../compile
# tar -zxvf ../source/pwgen-2.06.tar.gz
# cd pwgen-2.06
```

```
# ./configure
# make
# cp pwgen /usr/local/bin
```

Now just run pwgen, and grab yourself one of the generated passwords to use in the config.inc.php file of phpMyAdmin.

```
# pwgen
# vi /srv/www/phpmyadmin/config.inc.php
```

```
<?php
/* vim: set expandtab sw=4 ts=4 sts=4: */
/**
 * phpMyAdmin sample configuration, you can use it as base for
 * manual configuration. For easier setup you can use scripts/setup.php
 *
 * All directives are explained in Documentation.html and on phpMyAdmin
 * wiki <http://wiki.cihar.com>.
 *
 * @version $Id: config.sample.inc.php 10142 2007-03-20 10:32:13Z cybot_tm $
 */

/*
 * This is needed for cookie based authentication to encrypt password in
 * cookie
 */
$cfg['blowfish_secret'] = 'Ushah2fi'; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */

/*
 * Servers configuration
 */
$i = 0;

/*
 * First server
 */
$i++;
/* Authentication type */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
/* Server parameters */
$cfg['Servers'][$i]['host'] = 'localhost';
$cfg['Servers'][$i]['connect_type'] = 'tcp';
$cfg['Servers'][$i]['compress'] = false;
/* Select mysqli if your server has it */
$cfg['Servers'][$i]['extension'] = 'mysql';
/* User for advanced features */
// $cfg['Servers'][$i]['controluser'] = 'pma';
// $cfg['Servers'][$i]['controlpass'] = 'pmapass';
/* Advanced phpMyAdmin features */
// $cfg['Servers'][$i]['pmadb'] = 'phpmyadmin';
// $cfg['Servers'][$i]['bookmarktable'] = 'pma_bookmark';
// $cfg['Servers'][$i]['relation'] = 'pma_relation';
// $cfg['Servers'][$i]['table_info'] = 'pma_table_info';
// $cfg['Servers'][$i]['table_coords'] = 'pma_table_coords';
// $cfg['Servers'][$i]['pdf_pages'] = 'pma_pdf_pages';
// $cfg['Servers'][$i]['column_info'] = 'pma_column_info';
// $cfg['Servers'][$i]['history'] = 'pma_history';
// $cfg['Servers'][$i]['designer_coords'] = 'pma_designer_coords';

/*
 * End of servers configuration
 */
```

```
/*
 * Directories for saving/loading files from server
 */
$cfg['UploadDir'] = '';
$cfg['SaveDir'] = '';

?>
```

Now you should be able to login to phpMyAdmin at http://dspam.comsolve.nl/phpMyAdmin