ASSP version 2.5.5(16366)

Table of Contents:

Configuration Synchronization and Sharing

☐ **Enable Configuration Sharing** ***(enableCFGShare)***

Read all positions in this section carefully (multiple times is recommended!!!)! A wrong configuration sequence or wrong configuration values can lead in to a destroyed ASSP configuration!

If set, the configuration value and option files synchronization will be enabled. This synchronization belong to the configuration values, to the file that is possibly defined in a value and to the include files that are possibly defined in the configured file. If you don't want a specific configuration file or include file to be synchronized (send and receive), write
assp-no-sync
as a comment anywhere in the file. A possible reason can be for example '**localDomains**' - if ASSP1 is hosting DOMAIN1 and DOMAIN2 but ASSP2 is hosting only DOMAIN2 - so the entry for DOMAIN2 could be put in a not synchronized include file on ASSP1 and the synchronized main config file contains the entry for DOMAIN1.
If the configuration of all values in this section is valid, the synchronization status will be shown in the GUI for each config value that is, or **can be shared**. There are several configuration values, that can not be shared. The list of all shareable values can be found in the distributed file assp_sync.cfg

For an initial synchronization setup set the following config values in this order: setup **syncServer**, **syncConfigFile**, **syncTestMode** and as last **syncCFGPass** (leave **isShareSlave** and **isShareMaster** off). Use the default (distributed **syncConfigFile** assp_sync.cfg) file and configure all values to your needs - do this on all peers by removing lines or setting the general sync flag to 0 or 1 (see the description of **syncConfigFile**).
If you have finished this initial setup, enable **isShareMaster** or **isShareSlave** - now assp will setup all entries in the configuration file for all sync peers to the configured default values (to 1 if **isShareMaster** or to 3 if **isShareSlave** is selected). Do this on all peers. Now you can configure the synchronization behavior for each single configuration value for each peer, if it should differ from the default setup.
For the initial synchronization, configure only one ASSP installation as master (all others as slave). If the initial synchronization has finished, which will take up to one hour, you can configure all or some assp as master and slave. On the initial master simply switch on **isShareSlave**.
On the inital slaves, switch on **isShareMaster** and change all values in the sync config file that should be bidirectional shared from 3 to 1. As last action enable **enableCFGShare** on the SyncSlaves first and then on the SyncMaster.
After such an initial setup, any changes of the peers (**syncServer**) will have no effect to the configuration file (**syncConfigFile**)! To add or remove a sync peer after an initial setup, you have to configure **syncServer** and you have to edit the sync config file manually.

This option can only be enabled, if **isShareMaster** and/or **isShareSlave** and **syncServer** and **syncConfigFile** and **syncCFGPass** are configured!
Because the synchronization is done using a special SMTP protocol (without "mail from" and "rcpt to"), this option requires an installed Net::SMTP module in PERL. If you want the sync feature to use a secured connection (using STARTTLS) , DoTLS has to be set to "do TLS". This special SMTP protocol is not usable to for any MTA for security reasons, so the "sync mails" could not be forwarded via any MTA.
For this reason all sync peers must have a direct or routed TCP connection to each other peer.

If you build a sync topology with more than two ASSP, please notice, that it is not allowed to build any ring-synchronization. Only a chain-, tree- or star- topology is supported. It is also not allowed to build a sync ring inside any of the three allowed topologies!
show sync status

☐ **This is a Share Master** ***(isShareMaster)***
If selected, ASSP will send configured configuration changes to sync peers.

☐ **This is a Share Slave** ***(isShareSlave)***
If selected, ASSP will receive configured configuration changes from sync peers. To accept a sync request, every sending peer has to be defined in **syncServer** - even if there are manually made entries in the sync config file for a peer.

Default Sync Peers ***(syncServer)***

Define all configuration sync peers here (to send changes to or to receive changes from). Separate multiple values by "|". Any value must be a pair of hostname or ip-address and :port, like 10.10.10.10:25 or mypeerhost:125 or mypeerhost.mydomain.com:225 or [2202::00FF]:25.
The :port must be defined!
The target port can be the **listenPort** , **listenPort2** , **relayPort** or if **syncUsesSSL** is enabled, it has to be the **listenPortSSL** of the peer.

☐ **SSL is used for the Sync SMTP Transport** ***(syncUsesSSL)***
If selected, SSL will be used for the transport of the synchronization requests. In this case the target ip:port of all peers must be its **listenPortSSL** ! The Perl modules Net::SMTP::SSL and IO::Socket::SSL must be installed and enabled if this option is selected, otherwise all synchronization requests will fail!

☐ **Test Mode for Config Sync** ***(syncTestMode)***
If selected, a master (**isShareMaster**) will process all steps to send configuration changes, but will not realy send the request to the peers. A slave (**isShareSlave**) will receive all sync requests, but it will not change the configuration values and possibly sent configuration files will be stored at the original location and will get an extension of ".synctest".

Configuration File for Config Sync* ***(syncConfigFile)***
file:assp_sync.cfg Edit file

Define the synchronization configuration file here (default is file:assp_sync.cfg).
This file holds the configuration and the current status of all synchronized assp configuration values.
The format of an initial value is: "varname:=syncflag" - where syncflag could be 0 -not shared and 1 -is shared - for example: HeaderMaxLength:=1 . The syncflag is a general sign, which means, a value of 0 disables the synchronization of the config value for all peers. A value of 1, enables the peer configuration that possibly follows.
The format after an initial setup is: "varname:=syncflag,syncServer1=status,syncServer2=status,.....". The "status" can be one of the following:

0 - no sync - changes of this value will not be sent to this **syncServer** - I will ignore all change requests for this value from there
1 - I am a SyncMaster, the value is still out of sync to this peer and should be synchronized as soon as possible
2 - I am a SyncMaster, the value is still in sync to this peer - I am also a SyncSlave to this peer (bidirectional sync) if **isShareSlave** is enabled
3 - I am not a SyncMaster but a SyncSlave - only this SyncMaster (peer) knows the current sync status to me
4 - I am a SyncMaster and a SyncSlave (bidirectional sync) - a change of this value was still received from this **syncServer** (peer) and should not be sent back to this **syncServer** - this flag will be automatically set back to 2 at the next synchronization check

Config Sync Password ***(syncCFGPass)***

The password that is used and required (additionally to the sending IP address) to identify a valid sync request. This password has to be set equal in all ASSP installations, from where and/or to where the configuration should be synchronized.
The password must be at least six characters long.
If you want or need to change this password, first disable **enableCFGShare** here and on all peers, change the password on all peers, enable **enableCFGShare** on SyncSlaves then enable **enableCFGShare** on SyncMasters.

☐ **Show Detail Sync Information in GUI** ***(syncShowGUIDetails)***

If selected, the detail synchronization status is shown at the top of each configuration parameter like:

nothing shown - there is no entry defined for this parameter in the **syncConfigFile** or it is an unsharable parameter
"(shareable)" - the parameter is shareable but the general sync sign in the **syncConfigFile** is zero
"(shared: ...)" - the detail sync status for each sync peer

If not selected, only different colored bulls are shown at the top of each configuration parameter like:

nothing shown - no entry in the **syncConfigFile** or it is an unsharable parameter
"black bull ●" - the parameter is shareable but the general sync sign in the **syncConfigFile** is zero
"green bull ●" - the parameter is shared and in sync to each peer
"red bull ●" - the parameter is shared but it is currently out of sync to at least one peer

If you move the mouse over the bull, a hint box will show the detail synchronization status. A click on the bull or link will open a sync config dialog box for the single configuration parameter.

Notes Config Sync
Notes

Network Setup for Incoming Mail and Authentication

☐ **Disable all new SMTP and Proxy Network Connections** ***(DisableSMTPNetworking)***

If selected, ASSP will not answer to new SMTP and Proxy connections on '**listenPort** , **listenPort2** , **listenPortSSL** , **relayPort** and **ProxyConf**'. Currently existing SMTP and Proxy connections are not affected! Web and Stat connection are also not affected.

☐ **Enable IPv6 support** ***(enableINET6)***

For IPv6 network support to be enabled, check this box. Default is disabled. IO::Socket::INET6 is able to handle both IPv4 and IPv6. NOTE: This option requires an installed **IO::Socket::INET6** module in PERL and your system should support IPv6 sockets to give enabling this option a sense!
It is recommended to leave this option OFF as long as you don't want to use IPv6 addresses for a listener or a destination (SMTP,DNS-server,LDAP-server etc.).
Before you enable or disable IPv6, please check every IP listener and destination definition in assp and correct the settings. **After changing this option a restart of assp is recommended.** IPv4 addresses are defined for example 192.168.0.1 or 192.168.0.1:25 - IPv6 addresses are defined like [FE80:1:0:0:0:0:0:1]:25 or [FE80:1::1]:25 ! If an IPv4 address is defined for a listener, assp will listen only on the IPv4 socket. If an IPv6 address is defined for a listener, assp will listen only on the IPv6 socket. If only a port is defined for a listener, assp will listen on both IPv4 and IPv6 sockets.
For the definition of destination IP's applies the same. You are free to define hostnames instead of IP addresses like myhost.mydomain.com:25 - how ever, because of the needed IP address resolving, this will possibly slow down assp.

SMTP Listen Port ***(listenPort)***

25

The port number on which ASSP will listen for incoming SMTP connections (normally 25). You can specify both an IP address and port number to limit connections to a specific interface. Separate multiple entries by "|".
Examples: 25, 127.0.0.1:25, 127.0.0.1:25|127.0.0.2:25|[FE80:1::1]:25

SMTP Destination ***(smtpDestination)***

125

The IP **number!** and port number of your primary SMTP **mail transfer agent** (MTA). If multiple servers are listed and the first listed MTA does not respond, each additional MTA will be tried. If only a port number is entered, or the dynamic keyword **INBOUND** is used with a port number, then the connection will be established to the local IP address on which the connection was received. This is useful when you have several IP addresses with different domains or profiles in your MTA. If INBOUND:PORT is used, ReportingReplies (Analyze,Help,etc and CopyMail will go to 127.0.0.1:PORT or [::1]:PORT. If your needs are different, use smtpReportServer (SMTP Reporting Destination) and **sendAllDestination** (Copy Spam SMTP Destination). Separate multiple entries by "|".
If you need to connect to the SMTP destination host using native SSL, write 'SSL:' in front of the IP/host definition. In this case the Perl module **IO::Socket::SSL** must be installed and enabled (**useIOSocketSSL**).

Examples: 125, 127.0.0.1:125, 127.0.0.1:125|127.0.0.5:125|SSL:127.0.0.1:465, INBOUND:125

SMTP Destination Routing Table* ***(smtpDestinationRT)***

If INBOUND is used in the SMTP Destination field, the rules specified here are used to route the inbound IP address to a different outbound IP address. You must specify a port number with the outbound IP address.
*Example:*141.120.110.1=>141.120.110.129:25|141.120.110.2=>141.120.110.130:125|141.120.110.3=>SSL:141.120.110.130:125

SMTP and Proxy - Destination to Local IP-address Mapping* ***(smtpLocalIPAddress)***

You need to use the "file: ..." option for this parameter!
On windows systems at least Vista/2008 is required!
On multihomed systems with multiple default gateways, it could be required to define the local IP address (source) used for outgoing SMTP and Transparent Proxy (**ProxyConf**) connections.
This parameter allows to define local IP addresses used for specific targets (IP's or hosts) - based on the local address, the system will use the right gateway/interface.
Define one entry per line, comments (#) are allowed. The syntax for an entry is 'target=>local-IP'.
target could be any of: IP(4/6) network, IP(4/6) address, hostname, domain-name with wildcard (*).

for example:
22.* => 192.168.1.1 # IP4 Network
2222:333:* => FE81::1 # IP6 Network
22.23.24.25 => 10.1.1.1, # host IP4
1:2:3:4:5:6:7:8 => FE94::5 # host IP6
*.domain.com => 10.1.1.1 # domain
host.domain.com => 192.168.1.1 # host
* => 172.16.1.1 # default - if not defined, the system default is used

NOTICE: assp will NOT check, that the local IP address is available and bound to a local interface! It will also NOT check the system routing table! YOU SHOULD KNOW WHAT YOU DO!

SMTP Secure Listen Port ***(listenPortSSL)***

The port number on which ASSP will listen for incoming secure (SSL only) SMTP connections (normally 465). You can specify both an IP address and port number to limit connections to a specific interface. Separate multiple entries by "|".
Examples: 465, 127.0.0.1:465, 127.0.0.1:465|127.0.0.2:465
. More configuration options are **smtpSSLRequireClientCert**, **SSLSMTPCertVerifyCB** and **SSLSMTPConfigure** .

SSL Destination ***(smtpDestinationSSL)***

The IP **address!** and port number to connect to when mail is received on the SSL listen port. If the field is blank, the primary SMTP destination will be used.
If you need to connect to the SSL destination host using native SSL, write 'SSL:' in front of the IP/host definition. In this case the Perl module **IO::Socket::SSL** must be installed and enabled (**useIOSocketSSL**).
*Examples:*127.0.0.1:565, 565

Second SMTP Listen Port ***(listenPort2)***

A secondary port number on which ASSP can accept SMTP connections. This is useful as a dedicated port for VPN clients or for those who cannot directly send mail to a mail server outside of their ISP's network because the ISP is blocking port 25. You may also specify an IP address to limit connections to a specific interface. Separate multiple entries by "|".
Examples: 2525, 127.0.0.1:2525, 192.168.0.100:25000

Second SMTP Destination ***(smtpAuthServer)***

The IP address and port number to connect to when mail is received on the second SMTP listen port. If the field is blank, the primary SMTP destination will be used. The purpose of this setting is to allow remote users to make authenticated connections and transmit their email without encountering SPF failures. If you need to connect to the second SMTP destination host using native SSL, write 'SSL:' in front of the IP/host definition. In this case the Perl module **IO::Socket::SSL** must be installed and enabled (**useIOSocketSSL**).
Examples: 587, 127.0.0.1:587, SSL:127.0.0.1:465

Transparent TCP Proxy Table* ***(ProxyConf)***

Define any transparent TCP Port Proxy here. ASSP will proxy/forward (NOT route !) incoming TCP packets to a specific destination.
For example: if you want incoming TCP connections on port 465 (SMTP-SSL) to be forwarded to your email server.

*Example:*10.1.1.1:22=>172.16.22.33:22|0.0.0.0:465=>192.168.1.25:465<=12.1.1.3,34.5.6.0/16,67.23.2.1-67.23.2.5|10.1.1.1:1477=>192.168.1.23:1234<=1 [allow_proxy_1234]

Those connection are not especially SMTP related and they are not inspected by assp. Any application that uses the TCP layer, can use such a proxy (eg. SSH, RDP, VNC, POP3, HTTP, LDAP, Notes ...).
Proxy connections can be define in any direction: privat<->privat , privat<->public , public<->privat and public<->public

The syntax is: localIP:localPORT=>forwardIP:forwardPORT[<=AllowfromIP1,AllowfromIP2,...]|next Proxy configuration|....

The file-option (eg. file:files/proxy_conf.txt) is supported - if used, define one proxy configuration per line.
You have to configure the IP-address and IP-port for both - local and forward values! The optional AllowfromIP extension are comma separated values of IP-addresses (eg. 192.168.1.1), IP-networks (eg. 10.1.1.0/24) and IP-address ranges (172.16.1.3-172.16.1.10) from where connections are allowed. **Groups** definitions (eg. [allow_ssh_proxy]) may be used in AllowfromIP. If there is no allow value defined, all source IP addresses will be accepted!

Disable AUTH support on listenPorts ***(NoAUTHlistenPorts)***

This disables the SMTP AUTH command on the defined listenPorts independent from any other setting. This option works for **listenPort** , **listenPort2** and **listenPortSSL** . The listener definition here has to be the same like in the port definitions. Separate multiple entries by "|".
Examples: 25, 127.0.0.1:25, 127.0.0.1:25|127.0.0.2:25

☐ **Disable SMTP AUTH for External Clients** ***(DisableExtAUTH)***

If you do not want external clients (IP not in **acceptAllMail** or **relayPort** is not used) to use SMTP AUTH - for example to prevent address and password harvesting - check this option.
The "AUTH" offer in the EHLO and HELP reply will be stripped out, if set to on.
If this option is enabled and the AUTH command is used by an external client or server, **autValencePB** will be used to score the message and IP.
Notice: setting this option to ON could prevent roaming users (dynamic IP) from being able to authenticate!

Disable AUTH for these HELO's* ***(noAUTHHeloRe)***

If configured and a helo matches this regular expression, the AUTH offer will be removed from the EHLO reply and the AUTH command will be disallowed. For example: ^\w+\.noauthdomain\.com$,

Allow AUTH Only for these HELO's* ***(onlyAUTHHeloRe)***

If configured and a helo does not match this regular expression, the AUTH offer will be removed from the EHLO reply and the AUTH command will be disallowed. For example: ^\w+\.onlyauthdomain\.com$,

SMTP AUTH requires SSL/TLS ***(AUTHrequireTLS)***

NO

An SSL listener or STARTTLS is required before the SMTP AUTH command can be used.
This setting is ignored for all private IP addresses (localhost, RFC 1918, RFC 4193)!
In case of a mistake '538 5.7.11 transport layer encryption (SSL/TLS) required for requested authentication mechanism' is replied to the client.
'NO' is the default setting, but 'ALL' is recommended!

☐ **Force SMTP AUTH on Second SMTP Listen Port** ***(EnforceAuth)***

Force clients connecting to the second listen port to authenticate before transferring mail. To use this setting, both **listenPort2** (Second SMTP Listen Port) and **smtpAuthServer** (Second SMTP Destination) must be configured.

Notes On Network Setup
Notes

Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

Accept All Mail* ***(acceptAllMail)***

Relaying is allowed for these IPs. They contribute also to the whitelist. Before setting this option, please read the complete section - it is recommended to configure **relayPort** to send mails from your LAN to the Internet. This can take either a directly entered list of IP's separated by pipes or a file 'file:files/acceptall.txt'.
For example: 145.145.145.145|146.145.

☐ **Do Local Domain Check for Local Sender** ***(DoLocalSenderDomain)***

If activated, each local sender address must have a valid Local Domain. **acceptAllMail** and redlisted mails breaks this rule.

☐ **Do Local Address Check for Local Sender** ***(DoLocalSenderAddress)***

If activated, each local sender address must have a valid Local Address. **acceptAllMail** and redlisted mails breaks this rule.

☐ **Skip Local Domain Check** ***(nolocalDomains)***

Do not check relaying based on **localDomains**. Let the mailserver do it. **NOT RECOMMENDED**.

☐ **Do LDAP lookup for local domains** ***(ldLDAP)***

Check local domains against an LDAP database.
Note: Checking this requires filling in LDAP DomainFilter (**ldLDAPFilter**) in the LDAP section.
This requires an installed **NET::LDAP** module in Perl.

ISP/Secondary MX Servers* ***(ispip)***

Enter any addresses that are your ISP or backup MX servers, separated by pipes (|).
These addresses will (necessarily) bypass **Griplist**, IP Limiting, Delaying, Penalty Box, SPF, DNSBL & SRS checks unless the IP can be determined by (**ispHostnames**) ISP/Secondary Hostnames. For example: 127.0.0.1|172.16..

Regular Expression to Identify Forwarded Messages* ***(contentOnlyRe)***

Put anything here to identify messages which should bypass the PenaltyBox, Sender Validation, **Griplist**, IP Limiting, Delaying, SPF, DNSBL & SRS checks. For example: email addresses of people who are forwarding from other accounts to their mailbox on your server.

Regular Expression to Identify ISP/Secondary Hostnames* ***(ispHostnames)***

Hostnames (regular expression) to lookup the IP that connected to the ISP/Secondary server.
If found, this address is used to perform IP-based checks on forwarded messages.
For example: mx1\.yourisp\.com or mx1\.yourisp\.net|mx2\.yoursecondary\.com . *This hostnames are found in the 'Received:' header, like 'Received: from ...123.123.123.123... by* ***mx1.yourisp.com****'*. Leave this blank to disable the feature.

☑ **Send 250 OK To ISP/Secondary MX Servers** ***(send250OKISP)***

Set this checkbox if you want ASSP to reply to IP's in ISPIP with '250 OK' instead of SMTP error code '554 5.7.1'.

ISP/Secondary MX Grip Value ***(ispgripvalue)***

0.5

It is recommended to set it to 0.5 (Completely GReyIP) for ISP and Secondary MX servers. If left blank the **Griplist** X value is used (percentage of spam messages in relation to total).
Note: value has to be greater than 0 and less than 1, where 0 = never spam and 1 = always spam

Bounce Senders* ***(BounceSenders)***

postmaster|mailer-daemon

Envelope sender addresses treated as bounce origins. Null sender (<>) is always included.
Accepts specific addresses (postmaster@domain.com), usernames (mailer-daemon), or entire domains (@bounces.domain.com)
Automatic whitelist addition is skipped for mails from all bounce senders, the same way like redlisted mails are skipped from automatic whitelist addition.
If the list of bounce sender addresses is changed, a repair operation for the **whitelistdb** will be started. This task removes all whitelist entries, which are related to any local bounce sender.
Separate entries with pipes: |. For example: postmaster|mailer-daemon

Pop Before SMTP DB File ***(PopB4SMTPFile)***

Enter the DB database filename of your POP before SMTP implementation with records stored for dotted-quad IP addresses.
For example: /etc/mail/popip.db

☐ **Pop Before SMTP Merak Style** ***(PopB4SMTPMerak)***

If set Merak 7.5.2 is supported.

Relay Host ***(relayHost)***

Your isp's mail relayhost (smarthost). For example: mail.isp.com:25
If you run Exchange/Notes and you want assp to update the nonspam database and the whitelist, then enter your isp's smtp relay host here.
Blank means no relayhost and the **smtpDestination** will be used. Separate multiple entries by "|".
If you need to connect to the relay host using native SSL, write 'SSL:' in front of the IP/host definition. In this case the Perl module **IO::Socket::SSL** must be installed and enabled (**useIOSocketSSL**).
Examples: your_ISP_Server:25, 149.1.1.1:25, SSL:149.1.1.2:465|any_other_host:25 !

User to Authenticate to Relay Host *(relayAuthUser)*

The username used for SMTP AUTH authentication to the relayhost - for example, if your ISP need authentication on the SMTP port! Supported authentication methods are PLAIN, LOGIN, CRAM-MD5 and DIGEST-MD5 . If the relayhost offers multiple methods, the one with highest security option will be used. The Perl module **Authen::SASL** must be installed to use this feature! The usage of this feature will be skipped, if the sending MTA uses the AUTH command. Leave this blank, if you do not want use this feature.

Password to Authenticate to Relay Host *(relayAuthPass)*

The password used for SMTP AUTH authentication to the relayhost ! Leave this blank, if you do not want use this feature.

Relay Port *(relayPort)*

Tell your mail server to connect to this IP/port as its smarthost / relayhost. For example: 225
Note that you'll want to keep the **relayPort** protected from external access by your firewall. To restrict access to the **relayPort** per IP address or network, use **allowRelayCon** .
You can supply an interface:port to limit connections. Separate multiple entries by "|".
Examples: 225, 127.0.0.1:225, 192.168.1.1:225|192.168.2.1:225 !

Allow Relay Connection from these IP's* *(allowRelayCon)*

Enter any addresses that are allowed to use the **relayPort** , separated by pipes (|). If empty, any ip address is allowed to connect to the **relayPort**. If this option is defined, keep in mind : Addresses defined in **acceptAllMail** are **NOT** automatically included and have to be also defined here, if them should allow to use the **relayPort**. For example: 127.0.0.1|172.16..
If you use MS Office 365, you should define the **EOP IP addresses** here and you should configure your firewall to redirect connection from the hosted Exchange server to the **relayPort** .

☐ **Allow Relaying Only for Local Sender *(RelayOnlyLocalSender)***

If set, the envelope sender (MAIL FROM:) is immediately checked after the DATA command is received (to be valid). If the sender address could not be validated, the connection is dropped.
This setting is ignored for **BounceSenders**, which can relay at any time.
The connection will be dropped regardless any other assp setting (except **EmailSenderOK**).
It is recommended to switch this to ON, if you use for example MS Office 365. At least, it is wise, to switch this (or **RelayOnlyLocalDomains**) to ON in every case

☐ **Allow Relaying Only for Local Domains *(RelayOnlyLocalDomains)***

If set, the envelope sender domain (MAIL FROM:) is immediately checked after the DATA command is received (to be a local domain). If the sender domain could not be validated, the connection is dropped.
This setting is ignored for **BounceSenders**, which can relay at any time.
The connection will be dropped regardless any other assp setting (except **EmailSenderOK**).
It is recommended to switch this to ON, if you use for example MS Office 365. At least, it is wise, to switch this (or **RelayOnlyLocalSender**) to ON in every case

No Relaying Error *(NoRelaying)*

530 Relaying not allowed
SMTP error message to deny relaying.

Default Local Host *(defaultLocalHost)*

assp.local
If you want to be able to send mail to local users without a domain name then put the default local domain here.
Blank disables this feature. For example: mydomain.com .

Local Frequency Interval *(LocalFrequencyInt)*

0
The time interval in seconds in which the number of envelope recipients per sending address has not to exceed a specific number (**LocalFrequencyNumRcpt**).
Use this in combination with **LocalFrequencyNumRcpt** to limit the number of recipients in a given interval, to prevent local abuse - for example from hijacked local accounts. A value of 0 (default) will disable this feature and clean the cache within five minutes. It is recommended to enable **DoLocalSenderAddress** and/or **DoLocalSenderDomain**, if you want to use this feature. To give users the chance to inform an admin about such blocked mails, local mails to **EmailAdmins** are never blocked because of that feature.
edit local Frequency Cache

Local Frequency Recipient Number *(LocalFrequencyNumRcpt)*

0
The number of envelope recipients per sending address that has not to exceed in a specific time interval (**LocalFrequencyInt**).
Use this in combination with **LocalFrequencyInt** to limit the number of recipients in a given interval, to prevent local abuse - for example from hijacked local accounts. A value of 0 (default) will disable this feature and clean the cache within five minutes. It is recommended to enable **DoLocalSenderAddress** and/or **DoLocalSenderDomain**, if you want to use this feature. To give users the chance to inform an admin about such blocked mails, local mails to **EmailAdmins** are never blocked because of that feature.
edit local Frequency Cache

Check local Frequency for this Users only* *(LocalFrequencyOnly)*

A list of local addresses, for which the 'local frequency check' should be done. Leave this field blank (default), to do the check for every address. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).
For example: fribo*@thisdomain.com|jhanna|@sillyguys.org

Check local Frequency NOT for this Users* *(NoLocalFrequency)*

A list of local addresses, for which the 'local frequency check' should not be done. Noprocessing messages will skip this check. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).
For example: fribo*@thisdomain.com|jhanna|@sillyguys.org

Check local Frequency NOT for this IP's* ***(NoLocalFrequencyIP)***

A list of local IP-addresses, for which the 'local frequency check' should not be done.
For example: 145.145.145.145|145.146.

☐ **Generate and Add DKIM signatures to relayed messages** ***(genDKIM)***

If selected, ASSP will add DKIM signatures to relayed messages if it finds a valid DKIM configuration in **DKIMgenConfig** for the sending domain. This will also be done for noprocessing mails. This requires an installed **Mail::DKIM** module in PERL.

The File with the DKIM configurations* ***(DKIMgenConfig)***

file:dkim/dkimconfig.txt Edit file

The file that contains the DKIM configuration. A description how to configure DKIM could be found in the default file dkim/dkimconfig.txt.

Notes On Relaying
Notes

SMTP Session Limits

Maximum Errors Per Session *(MaxErrors)*

5

The maximum number of SMTP session errors encountered before the connection is dropped. A value of zero disables this feature. PB: **meValencePB**

Maximum Sessions *(maxSMTPSessions)*

64

The maximum number of simultaneous SMTP sessions. This can prevent server overloading and DoS attacks. 64 simultaneous sessions are typically enough. Zero means no limit. Connections on **relayPort** will be counted, but connections on **relayPort** will never be limited because of this value. If the value is reached, assp will wait until the number of simultaneous SMTP sessions is lower than (value - 20) or (value * 0.75).

No Maximum Sessions IP numbers* *(noMaxSMTPSessions)*

Mail from any of these IP numbers will pass through without checking maximum number of simultaneous SMTP sessions. For example: 145.145.145.145

Maximum Sessions Per IP Address *(maxSMTPipSessions)*

5

The maximum number of SMTP sessions allowed per IP address. Use this setting to prevent server overloading and DoS attacks. 5 sessions are typically enough. If set to 0 there is no limit imposed by ASSP. **ispip** (ISP/Secondary MX Servers) and **acceptAllMail** (Accept All Mail) matches are excluded from SMTP session limiting. PB: **ipValencePB**

Maximum Header Size *(HeaderMaxLength)*

50000

The maximum allowed header length, in bytes. At each mail hop header information is added by the mail server. A large mail header can indicate a mail loop. If the value is blank or 0 the header size will not be checked.

Detect Possible Mailloop *(detectMailLoop)*

3

If set to a value higher than 0, ASSP count it's own Received-header in the header of the mail. If this count exceeds the defined value, the transmission of the message will be canceled.

Maximum Equal X-Header Lines* *(MaxEqualXHeader)*

*=>20

The maximum allowed equal X-header lines - eg. "X-SubscriberID". If the value is set to empty the header will not be checked for equal X-header lines. This check will be skipped for noprocessing, whitelisted and outgoing mails.
The default is "*=>20", which means any X-header can occur 20 time maximum. You can define different values for different X-headers - wildcards like "*" and "?" are allowed to be used.
For example:
=>20|X-Notes-Item=>100|X-Subscriber=>10|X-AnyTag=>0
A value of zero disables the check for the defined X-header. The check is also skipped if no default like "*=>20" is defined and the X-header definition is not found.

Max Real Size of Local Message *(maxRealSize)*

If the value of (number of [rcpt to] * [message size]) exceeds **maxRealSize** in bytes the transmission of the local message will be canceled. No limit is imposed by ASSP if the field is left blank or set to 0. This option allows admins to limit useless bandwidth wasting based on the total transmit size.

Max Real Size of Local Message Addresses* *(MaxRealSizeAdr)*

file:files/MaxRealSize.txt Edit file

Use this parameter to set individual **maxRealSize** values for email addresses, domains, user names and IP addresses. A file must be specified if used.
Accepts specific addresses (user@domain.com), user parts (user), entire domains (@domain.com) and IP addresses (IP-ranges and CIDR notation like 123.1.101/32 and IPv6 shortening like FE80::1 is here **NOT** supported!) - group definitions could be used. Use one entry per line. Wildcards are supported (fribo*@domain.co?) except for IP addresses. A second parameter separated by "=>" specifies the size limit in byte.
For example:
fribo*@thisdomain.co?=>1000000
jhanna=>0
@sillyguys.org=>500000
101.1.2.16=>0
[admins]=>0
If multiple matches (values) are found in a mail for any IP address in the transport mail chain, any envelope recipient and the envelope sender, the highest value or 0 (no limit) will be used! If no match (value) is found in a mail, the definition in **maxRealSize** will take place. NoProcessing (except npsize) will skip this check.

Max Real Size of External Message *(maxRealSizeExternal)*

If the value of (number of [rcpt to] * [message size]) exceeds **maxRealSizeExternal** in bytes the transmission of the external message will be canceled. No limit is imposed by ASSP if the field is left blank or set to 0. This option allows admins to limit useless bandwidth wasting based on the total transmit size.

Max Real Size of External Message Addresses* *(MaxRealSizeExternalAdr)*

file:files/MaxRealSizeExt.txt Edit file

Use this parameter to set individual **maxRealSizeExternal** values for email addresses, domains, user names and IP addresses. A file must be specified if used.

Accepts specific addresses (user@domain.com), user parts (user), entire domains (@domain.com) and IP addresses (IP-ranges and CIDR notation like 123.1.101/32 and IPv6 shortening like FE80::1 is here **NOT** supported!) - group definitions could be used. Use one entry per line. Wildcards are supported (fribo*@domain.co?) except for IP addresses. A second parameter separated by "=>" specifies the size limit in byte. For example:
fribo*@thisdomain.co?=>1000000
jhanna=>0
@sillyguys.org=>500000
101.1.2.16=>0
[admins]=>0
If multiple matches (values) are found in a mail for any IP address in the transport mail chain, any envelope recipient and the envelope sender, the highest value or 0 (no limit) will be used! If no match (value) is found in a mail, the definition in **maxRealSizeExternal** will take place. NoProcessing (except npsize) will skip this check.

max real message size Error *(maxRealSizeError)*

552 message exceeds MAXREALSIZE byte (size * rcpt)

SMTP error message to reject **maxRealSize** / **maxRealSizeExternal** exceeding mails. For example:552 message exceeds MAXREALSIZE byte (size * rcpt)! MAXREALSIZE will be replaced by the value of **maxRealSize** / **maxRealSizeExternal**.

Max Size of Local Message *(maxSize)*

If the value of ([message size]) exceeds **maxSize** in bytes the transmission of the local message will be canceled. No limit is imposed by ASSP if the field is left blank or set to 0. This option allows admins to limit useless bandwidth wasting based on the transmit size.

Max Size of Local Message Addresses* *(MaxSizeAdr)*

file:files/MaxSize.txt Edit file

Use this parameter to set individual **maxSize** values for email addresses, domains, user names and IP addresses. A file must be specified if used.
Accepts specific addresses (user@domain.com), user parts (user), entire domains (@domain.com) and IP addresses (IP-ranges and CIDR notation like 123.1.101/32 and IPv6 shortening like FE80::1 is here **NOT** supported!) - group definitions could be used. Use one entry per line. Wildcards are supported (fribo*@domain.co?) except for IP addresses. A second parameter separated by "=>" specifies the size limit in byte. For example:
fribo*@thisdomain.co?=>1000000
jhanna=>0
@sillyguys.org=>500000
101.1.2.16=>0
[admins]=>0
If multiple matches (values) are found in a mail for any IP address in the transport mail chain, any envelope recipient and the envelope sender, the highest value or 0 (no limit) will be used! If no match (value) is found in a mail, the definition in **maxSize** will take place. NoProcessing (except npsize) will skip this check.

Max Size of External Message *(maxSizeExternal)*

If the value of ([message size]) exceeds **maxSizeExternal** in bytes the transmission of the external message will be canceled. No limit is imposed by ASSP if the field is left blank or set to 0. This option allows admins to limit useless bandwidth wasting based on the transmit size.

Max Size of External Message Addresses* *(MaxSizeExternalAdr)*

file:files/MaxSizeExt.txt Edit file

Use this parameter to set individual **maxSizeExternal** values for email addresses, domains, user names and IP addresses. A file must be specified if used.
Accepts specific addresses (user@domain.com), user parts (user), entire domains (@domain.com) and IP addresses (IP-ranges and CIDR notation like 123.1.101/32 and IPv6 shortening like FE80::1 is here **NOT** supported!) - group definitions could be used. Use one entry per line. Wildcards are supported (fribo*@domain.co?) except for IP addresses. A second parameter separated by "=>" specifies the size limit in byte. For example:
fribo*@thisdomain.co?=>1000000
jhanna=>0
@sillyguys.org=>500000
101.1.2.16=>0
[admins]=>0
If multiple matches (values) are found in a mail for any IP address in the transport mail chain, any envelope recipient and the envelope sender, the highest value or 0 (no limit) will be used! If no match (value) is found in a mail, the definition in **maxSizeExternal** will take place. NoProcessing (except npsize) will skip this check.

max message size Error *(maxSizeError)*

552 message exceeds MAXSIZE byte (size)

SMTP error message to reject **maxSize** / **maxSizeExternal** exceeding mails. For example:552 message exceeds MAXSIZE byte (size)! MAXSIZE will be replaced by the value of **maxSize** / **maxSizeExternal**.

Max Number of AUTHentication Errors *(MaxAUTHErrors)*

If an IP (/24 network is used for incoming mails) exceeds this number of authentication errors (535 or 530) the transmission of the current message will be canceled and any new connection from that IP will be blocked for 5-10 minutes.
Every 5 Minutes the 'AUTHError' -counter of the IP will be decreased by one. **autValencePB** is used for the penalty box.
No limit is imposed by ASSP, if the field is left blank or set to zero (zero cleans the related cache 'AUTHError'). This option allows admins to prevent external bruteforce or dictionary attacks via AUTH command. Whitelisted, **noBlockingIPs** , **noMaxAUTHErrorIPs** and NoProcessing IP's are ignored like any relayed connection.

Reset the MaxAUTHErrors Counter for these IP's* *(ResetMaxAUTHErrorIPs)*

List of IP's for which **MaxAUTHErrors** counter should be cleared immediatly after a successful login. For example: 145.145.145.145|145.146. It is not recommended to use this option for security reasons, but it may required for client networks behind a NAT.

Do not check MaxAUTHErrors for these IP's* ***(noMaxAUTHErrorIPs)***

List of IP's which should not be checked for **MaxAUTHErrors** . For example: 145.145.145.145|145.146.

Max IP Changes for AUTHentication per User ***(AUTHUserIPfrequency)***

If the authentication methodes PLAIN, LOGIN or CRAM-MD5 are used by clients, two space separated values specify the number of different IP's and a timeframe in seconds, which should not be exeeded by a user.
For example "2 600" - notice these are the minimum values for IP-number and seconds.
The example disallows a user to authenticate (using PLAIN or LOGIN) from two or more different IP-addresses within 600 seconds. In other words - an user is allowed to authenticate from another IP-address, 601 seconds after the last authentication.
Each attempt to authenticate is counted by this feature.
MaxAUTHErrors is counted, if a user breakes this rule.
Leave this blank to disable this feature.
AUTHIP Cache

Check Same Subjects ***(DoSameSubject)***

disabled
If activated, assp will check the mail subjects for equality using the config parameters below. Scoring is done with '**isValencePB**'.

Subject Frequency Interval ***(subjectFrequencyInt)***

300
The time interval in seconds in which the number of equal subjects has not to exceed a specific number (**subjectFrequencyNumSubj**).
Use this in combination with **subjectFrequencyNumSubj** to limit the number of equal subjects in a given interval. A value of 0 (default) will disable this feature and clean the cache within five minutes.
edit Subject Frequency Cache

Subject Frequency Number of Subjects ***(subjectFrequencyNumSubj)***

5
The number of equal subjects that has not to exceed in a specific time interval (**subjectFrequencyInt**).
Use this in combination with **subjectFrequencyInt** to limit the number of equal subjects in a given interval. A value of 0 (default) will disable this feature and clean the cache within five minutes.
edit Subject Frequency Cache

Check Equal Subject Frequency for this Users only* ***(subjectFrequencyOnly)***

A list of local addresses, for which the 'subject frequency check' should be done. Leave this field blank (default), to do the check for every address.
Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).
For example: fribo*@thisdomain.com|jhanna|@sillyguys.org

Check Equal Subject Frequency NOT for this Users* ***(NoSubjectFrequency)***

A list of local addresses, for which the 'subject frequency check' should not be done.
Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).
For example: fribo*@thisdomain.com|jhanna|@sillyguys.org

Check Equal Subject Frequency NOT for this IP's* ***(NoSubjectFrequencyIP)***

Mail from any of these IP numbers will pass through without checking the equality of subjects. For example: 145.145.145.145

SMTP Idle Timeout ***(smtpIdleTimeout)***

180
The number of seconds a session is allowed to be idle before being forcibly disconnected. The default is 180 seconds. No limit is imposed by ASSP if the field is left blank or set to 0. If you have not defined an IdleTimeout on your MTA, this value should not be set to 0, because then a connection will never be timed out!

SMTP Idle Timeout for Whitelisted and Noprocessing ***(NpWlTimeOut)***

1200
The number of seconds a whitelisted or noprocessing session is allowed to be idle before being forcibly disconnected. The default is 1200 seconds. No limit is imposed by ASSP if the field is left blank or set to 0. If you have not defined an IdleTimeout on your MTA, this value should not be set to 0, because then a connection will never be timed out!

SMTP Idle Timeout after NOOP ***(smtpNOOPIdleTimeout)***

0
The number of seconds a session is allowed to be idle after a "NOOP" command is received, before being forcibly disconnected. The default is 0 seconds. No limit is imposed by ASSP if the field is left blank or set to 0.
This should prevent hackers to hold and block connections by sending "NOOP" commands short before the "**smtpIdleTimeout**" is reached.

SMTP Idle Timeout after NOOP Count ***(smtpNOOPIdleTimeoutCount)***

0
The number of counts a session is allowed send "NOOP" commands following on each other, before being forcibly disconnected. The default is 0. No limit is imposed by ASSP if the field is left blank or set to 0.
This in cooperation with "**smtpNOOPIdleTimeout**" should prevent hackers to hold and block connections by sending repeatedly "NOOP" commands short before the "**smtpNOOPIdleTimeout**" is reached. If "**smtpNOOPIdleTimeout**" is not defined or 0, this value will be ignored!

Notes

Group Definition for IP's , Users and Domains

Address and Domain Groups* ***(Groups)***

file:files/groups.txt [Edit file]

If you don't want to use group definitions, leave this field blank otherwise a file definition like 'file:files/groups.txt' is required.
Group definitions could be used in any other configuration value where multiple user names, email addresses or domain names or IP addresses could be defined.
Groups are defined and used using the same syntax [group-name] (including the brackets) in a single line. In the configuration parameters, the line [group-name] will be replaced by the content of the group definition, that is done here.
All group definitions are case sensitive. Group names can only contain the following characters: A-Z, a-z, 0-9, - , _ and @ !
The structure of this file has to be as follows:

```
[super_spamlovers]
myBoss
ldap:{host=>my_LDAP_server:389,base=>(sep)DC=domain,DC=tld(sep),user=>(sep)CN=admin,DC=domain(sep),password=>(sep)pass
(sep),timeout=>2,scheme=>ldap,STARTTLS=>1,version=>3},{(CN=management)}{member},{(CN=%USERID%)}{mailaddress}
entry
exec:/usr/bin/list_postfix_users --domain mydomain --group postoffice
entry
...

[admins]
ldap:{host=>domino1.mydomain.com:389,base=>(sep)DC=domain,DC=tld(sep),user=>(sep)Administrator(sep),password=>(sep)pass
(sep),timeout=>2,scheme=>ldap,STARTTLS=>1,version=>3},{(CN=LocalDomainAdmins)}{member},{(CN=%USERID%)}{mailaddress}
entry
# include files/other.file.txt entry
...

[specialIPList]
1.2.3.4
123.234.0.0/16
::1
```

Lines starting with a # OR ; are consider a comment. Empty lines will be ignored. A group definition stops, if a new group definition starts or at the end of the file. Comments are not allowed inside a definition line.

There are two possible methods to import entries from an external source in to a group - the execution of a system command or an LDAP query.
To import entries via a system command like (eg. cat|grep or find or your self made shell script), write a single line that begins with exec: followed by the command to be executed - like:
exec:cat /etc/anydir/*.txt|grep '@'
The executed system command has to write a comma(,) or pipe(|) or linefeed(LF,CRLF) separated list of entries to STDOUT, that should become part of that group, where this line is used. There could be multiple and any combination of entry types in one group definition.
Be carefull! The external script should never BLOCK, DIE or RUN longer than some seconds. It is may be better, to schedule the script by a system cron job, write the output of the script to a file and to include this file here.

If you are familar with the usage of LDAP, you can define LDAP queries to import entries from one or more LDAP server. This is done, defining one query per line. The syntax of such a line is:

ldap:{host_and_protocol},{LDAP_group_query_filter}{LDAP_group_query_attribut_to_return},{LDAP_entry_query_filter}
{LDAP_entry_query_attribut_to_return}

If the 'host_and_protocol' part is empty {}, the default LDAP configuration will be used. A 'host_and_protocol' part should contain the following entries in the following structure:
{host=>127.0.0.1:389,base=>(sep)DC=domain,DC=tld(sep),user=>(sep)...(sep),password=>(sep)pass
(sep),timeout=>..,scheme=>ldap/ldaps,STARTTLS=>0/1,version=>2/3}
The 'host' has to be set, if you want to define any other LDAP parameter. If any other parameter is not defined, the default LDAP configuration value will be used, except user and password. The port definition (:xxx) in the host setting is optional - if not defined, the default LDAP ports 389(LDAP) and 636(LDAPS) will be used. It is possible to define a comma(,) separated list of hosts for failover functionality like 'host=>"localhost:389,192.168.1.1:389,...."' - notice the quotes as terminator which are required in this case!
The value of the **base**, password and user parameter has to start and end with a single character (sep) as terminator, that is not part of the value and is not used in the value. The parameter "**base**" defines the LDAP search root like **LDAPRoot** .

The 'LDAP_group_query_filter' and 'LDAP_group_query_attribut_to_return' are used to query an LDAP group for it's members (users). The resulting list will contain the requested attributes of all group members. The definition of these two parameters could look as follows:
{(&(objectclass=dominoGroup)(CN=LocalDomainAdmins))}{member}

It is possible to modify each returned value with a callback-code. This is for example useful for MS-AD queries on the attribute 'proxyaddresses', which returns a list of all available mail addresses (SMTP,smtp,X400...).
example: ldap:{},{(&(CN=firstname lastname)(proxyaddresses=smtp:*))<=s/^\s*smtp:\s*(.+)\s*$/$1/i}{proxyaddresses},{}{}
<= is the required separator, s/^\s*smtp:\s*(.+)\s*$/$1/i is the callback code.
The callback code has to return a value of not zero or undef on success. The code gets the LDAP result in the variable $_ and has to modify this variable in place on success.
It is not allowed to use any of the following characters in the callback definition of an ldap line: {}|

The 'LDAP_entry_query_filter' and 'LDAP_entry_query_attribut_to_return' are used to query each member from the first query, for it's email address. The literal '%USERID%' in the 'LDAP_entry_query_filter' will be replaced by each LDAP-attribute result of the first query. The definition of these two parameters could look as follows:
{(&(objecttype=person)(CN=%USERID%)(o=%USERID%))}{mailaddress}
or more simple
{(&(objecttype=person)(CN=%USERID%))}{mailaddress}

A callback code could be used the same way like for 'LDAP_group_query_filter' - {(&(objecttype=person)(CN=%USERID%))<=callback-code}{mailaddress}.
To break long lines in to multiple, terminate a continued line with a slash "/"

If you are able to get all results (eg. email addresses or domain names) with the 'LDAP_group_query' query, leave the definition of 'LDAP_entry_query_filter' and 'LDAP_entry_query_attribut_to_return' empty {}{}.

The result of each group definition will be stored in a file in files/group_export/GROUPNAME.txt.
The groups are build at every start of assp and if the defined file or an include file is stored (changed file time). To force a reload of all groups,

open the file and click 'Save changes' or change the file time with an external shell script. It is also possible to use **GroupsReloadEvery**, to reload the **Groups** definition in time intervals, if the exec: or ldap: option are used.

Reload the Groups definitions every this minutes [s] ***(GroupsReloadEvery)***

60

ASSP will reload the **Groups** definition every this minutes, if the exec: or ldap: option is used in **Groups**.
A value of zero disables the scheduled reload. Defaults to 60 minutes.

Notes On Group Definitions

Notes

SPAM Control

Regular Expression to Identify Redlisted Mail* ***(redRe)***

file:files/redre.txt | Edit file

If an email matches this Perl regular expression it will be considered redlisted.
redRe detects tags to process a mail like the recipient were redlisted - nothing else (no redlist addition/removal).
The Redlist serves two purposes:
1) the Redlist is a list of addresses that cannot contribute to the whitelist and which are not considered local even if their mail is from a local computer. For example, if someone goes on a vacation and turns on their autoresponder, put them on the redlist until they return. Then as they reply to every spam they receive they won't corrupt your non-spam collection or whitelist: \[autoreply\]
2) Redlisted addresses will not be added to the Whitelist when your local user sends mail to that address, thereby preventing accidental pollution of the Whitelist by, say, inadvertent replies by your users to mails from the spammer.
Redlisted messages will not be stored in the SPAM/NOTSPAM-collection. As all fields marked by * this field accepts a list separated by | or a specified file 'file:files/redre.txt'.

☐ **Add Whitelist Removals To Redlist** ***(EmailWhiteRemovalToRed)***

If set addresses which are removed from Whitelist via email-interface will automatically be added to the Redlist. The address can only be added again to the Whitelist after it is removed from the Redlist.

Spam Error ***(SpamError)***

554 5.7.1 Mail appears to be unsolicited -- send error reports to postmaster@LOCAL

SMTP error message to reject spam. The literal LOCALDOMAIN will be replaced by the recipient domain. The literal LOCALUSER will be replaced by the recipient user part. For example:554 5.7.1 Mail appears to be unsolicited -- send error reports to postmaster@LOCALDOMAIN.

Ham Password SALT ***(NotSpamTag)***

If an incoming email subject contains the TAG generated based on this value, it will be considered as defined in **NotSpamTagProc** . The literal 'NOTSPAMTAG' (will be replaced by a 10 digit not-spam-tag) can be used in any 5xx error Reply of:

SpamError
SenderInvalidError
PenaltyError
SPFError
RBLError
URIBLError
UuencodedError
bombError
scriptError

to ask the sender for resending the mail with the TAG in the subject.
For example: **SpamError** may be set to:
554 5.7.1 ERROR mail appears to be unsolicited - send the mail again and append 'NOTSPAMTAG' to the mail subject - or send error reports to postmaster@LOCALDOMAIN

Randomly picked up bit sequences of the text defined here, are used as "SALT" to calculate a 10 digit not-spam-tag. This value must be at least 12 characters long. Leave this value empty to disable this feature.
Every generated TAG can be used by the sender exactly one time. Every additional usage of a TAG will be ignored, and the sender may get a new generated TAG.
To define your own static TAGs, use **whiteRe** and/or **npRe** and change the error reply definitions accordingly.
To generate a random 80 character string, run 'perl -e "print chr(int(rand(94))+33)for(0...79);"' from command line and copy and paste the result to here.
All assp (eg. backup MX), that are processing mails for the same domains, have to used the same value for this parameter!

If a mail fails on some specific checks (for example SPF, all HELO checks, local sender, spoofing, **ForceRBLCache**), NOTSPAMTAG is not provided (empty).
An sender who makes these mistakes, should never get the chance to bypass using the NOTSPAMTAG.

Not-Spam-Tag will consider the mail as ***(NotSpamTagProc)***

whitelisted

If a sender uses the Not-Spam-Tag , how should the mail be processed. Regardless of this setting, the IP address of the sender will not be penalized, if a **NotSpamTag** is found.

☐ **Don't Upload Griplist Stats** ***(noGriplistUpload)***

Check this to disable the **Griplist** upload. The **Griplist** contains IPs and their value between 0 and 1, lower is less spammy, higher is more spammy. This value is called the grip value.

☐ **Don't auto-download the Griplist file** ***(noGriplistDownload)***

Set this checkbox, if you don't use the **Griplist**. You have to disable also **noGriplistUpload** to download the **Griplist**.

☐ **Store Assp-Header into Spam Collection** ***(StoreASSPHeader)***

Add "X-Assp-" to the collected spam-mails.

☑ **Add Envelope-Recipient Header** ***(AddIntendedForHeader)***

Adds two lines to the email header: "X-Assp-Intended-For: user@domain" and "X-Assp-Envelope-From: user@domain".

☑ **Block Outgoing Spam-Prob header** ***(NoExternalSpamProb)***

Check this box if you don't want your X-Assp-Spam-Prob header on external mail
Note this means mail from local users to local users will also be missing the header.

☑ **Add Spam Header** ***(AddSpamHeader)***

Adds a line to the email header "X-Assp-Spam: YES" if the message is spam, or "X-Assp-Spam: YES (Probably)" if it is possibly spam.

Add Custom Header ***(AddCustomHeader)***

X-Spam-Status:yes

Adds a line to the email header if the message is spam. For example: **X-Spam-Status:yes**

☑ **Add Graphical Level Header** ***(AddLevelHeader)***

Adds a line to the email header "X-Assp-Spam-Level: **** " showing the total message score represented by stars (1 - 20), every star represents five scoring points.

☐ **Add X-ASSP-Original-Subject Header** ***(AddSubjectHeader)***

Adds a line to the email header "X-ASSP-Original-Subject: the subject".

☑ **Add Spam Reason Header** ***(AddSpamReasonHeader)***

Adds a line to the email header "X-Assp-Spam-Reason: " explaining why the message is spam.

Notes On Spam Control

Notes

Copy Spam & Ham

Copy Spam and Send to this Address *(sendAllSpam)*

If this is set, ASSP will deliver a copy of spam mails to this address. For example: spammaster@mydomain.com. The literal USERNAME is replaced by the user part of the recipient, the literal DOMAIN is replaced by the domain part of the recipient. For example: USERNAME@Spam.DOMAIN, USERNAME+Spam@DOMAIN, catchallspamthis@DOMAIN. Separate multiple entries by comma or space. To deliver copy of spams based on the domain name (only some special hosted domains), use **ccSpamInDomain** .

Copy Spam and Send to this Address per Domain* *(ccSpamInDomain)*

If the domain of the recipient-address is matches one in this list, ASSP will deliver an additional copy of spam emails of a domain to this address (even if **sendAllSpam** is not set). For example: monitorspam@example1.com|monitor@example2.com.

Copy Spam SMTP Destination *(sendAllDestination)*

IP address and port to connect to when Spam messages are copied. If blank they go to the main SMTP Destination. eg "10.0.1.3:1025", "SSL:10.0.1.3:465", "1025", etc.

Copy Spam to these Recipients Only* *(ccSpamFilter)*

Restricts Copy Spam to these recipients. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Copy Spam to these Recipients always* *(ccSpamAlways)*

Copy Spam to these recipients regardless of collection mode. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Do Not Copy Spam Regex* *(ccSpamNeverRe)*

Never Copy Spam regardless of collection mode. Put anything here to identify messages which should not be copied.

Do Not Copy Messages Above This MessageTotal score *(ccMaxScore)*

Messages whose score exceeds this threshold will not be copied. For example: 75

☑ **Restrict Copy Spam to MaxBytes** *(ccMaxBytes)*

CCMail will cut off Spam mails, thereby reducing the load considerably (recommended).

☐ **Prepend Spam Subject to Copied Spam** *(spamSubjectCC)*

If set, **spamSubject** gets prepended to the subject of the copied message.

☑ **Prepend Spam Tag to Copied Spam** *(spamTagCC)*

The check which caused the spam detection will be prepended to the subject of the message. For example: [DNSBL]

Copy Not-Spam SMTP Destination *(sendAllHamDestination)*

IP address and port to connect to when Ham messages are copied. If blank they go to the Spam SMTP Destination. eg "10.0.1.3:1025", "SSL:10.0.1.3:465",, "1025", etc.

Copy Incoming Not-Spam and Send to this Address *(sendHamInbound)*

If you put an address in this box ASSP will forward a copy of notspam messages from outside to this address. The literal USERNAME is replaced by the user part of the recipient, the literal DOMAIN is replaced by the domain part of the recipient. For example: archiv@mydomain.com, USERNAME@mybackup.domain, catchallforthis@DOMAIN

Copy Outgoing Not-Spam and Send to this Address *(sendHamOutbound)*

If you put an address in this box ASSP will forward a copy of outgoing notspam messages to this address.

Copy Ham Filter* *(ccHamFilter)*

Copy Not-Spam to these addresses only. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Do Not Copy Ham Filter* *(ccnHamFilter)*

Do Not Copy Ham to these addresses. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

☐ **ccMail Recipient Replacement** *(ccMailReplaceRecpt)*

The recipient replacement (**ReplaceRecpt**) rules from the "Recipients/Local Domains" section, will be used to replace ccMail recipients. For example: **sendHamInbound** = USERNAME@yourspamdomain.lan - in this case you are able to detect the target domain "yourspamdomain.lan" in a rule and you can replace the recipient/domain depending on its values and/or on the senders address.

Notes On CC Messages

Notes

SPAM Lover and SPAM Hater

☐ **Suppress SpamSubject to Spam-Lover-Messages** *(spamSubjectSL)*

If set, **spamSubject** and **spamTag** does NOT get prepended to the subject of the Spam-Lover-Message.

☑ **Suppress SpamTags to Spam-Lover-Messages** *(spamTagSL)*

If set, spamTags (the method used to catch spam) does NOT get prepended to the subject of the Spam-Lover-Message.

☐ **Group SpamLovers and Not SpamLovers per mail** *(groupSpamLovers)*

If set, the first envelope recipient consider a mail to be for spamlovers or not. If the first envelope recipient is any SpamLover, all other (following) envelope recipients must be also any SpamLover (or reverse) - if not, their address will be not accepted by ASSP for this single mail and '452 too many recipients' will be sent.

All Spam-Lover* *(spamLovers)*

postmaster|abuse

Messages to Spam-Lovers are processed and filtered by ASSP, but (optionally) get tagged with **spamSubject** (if would be blocked) and are not blocked. When a Spam-Lover is not the sole recipient of a message, the message is processed normally, and if it is found to be spam, it will not be delivered to the Spam-Lover.
delaySpamLovers are not included here and must be set additionally.
Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com). Default: postmaster|abuse.
For example: fribo*@thisdomain.com|jhanna|@sillyguys.org

This option and all SpamLover-Options (...SpamLovers) below accept a second score parameter like "user@your-domain.com=>70"
If such a parameter is defined in any option for an entry and the recipient address matches this entry and the message score exceeds the parameter value, the message will be considered spam.
If there are multiple possible matches for a recipient address found, the generic longest match (and assigned value) will be used.
ASSP will use the highest found value for all envelope recipients of an email.
Notice: the settings for [Local]PenaltyMessageLimit and [Local]PenaltyMessageLow will be overwritten for the mail, if a match is found.
The according ...Low limit is calculated as:
for incoming mails: value - (**PenaltyMessageLimit** - **PenaltyMessageLow**)
or
for outgoing and local mails: value - (**LocalPenaltyMessageLimit** - **LocalPenaltyMessageLow**)

Regular Expression to Identify Spam-Lover* *(SpamLoversRe)*

If a message matches this regular expression it will be considered a Spam-Lover message.

Bayesian Spam-Lover* *(baysSpamLovers)*

Regular Expression to Identify Bayesian Spam-Lover* *(baysSpamLoversRe)*

If a message matches this regular expression it will be considered a Bayesian Spam-Lover message. For example: password|news

☑ **Do not store Bayesian Spam-Lover in SpamDB** *(baysSpamLoversRed)*

If set (recommended), mail to Bayesian Spam-Lover will be stored in the **discarded** folder (not in the Spam/Notspam folder).

Blacklisted Domains Spam-Lover* *(blSpamLovers)*

Bomb Spam-Lover* *(bombSpamLovers)*

HELO Blacklisted Spam-Lover* *(hlSpamLovers)*

Valid/Invalid Helo* *(hiSpamLovers)*

Bad Attachment Spam-Lover* *(atSpamLovers)*

SPF Failures Spam-Lover* *(spfSpamLovers)*

DNSBL Failures Spam-Lover* *(rblSpamLovers)*

URIBL Failures Spam-Lover* *(uriblSpamLovers)*

Unsigned SRS Bounces Spam-Lover* *(srsSpamLovers)*

No Delaying Spam-Lover* *(delaySpamLovers)*

Invalid Sender Spam-Lover* *(isSpamLovers)*

Missing MX Spam-Lover* *(mxaSpamLovers)*

Invalid/Missing PTR Spam-Lover* *(ptrSpamLovers)*

Penalty Box Blocking Spam-Lover * *(pbSpamLovers)*

Country Blocking Spam-Lover * *(sbSpamLovers)*

All Spam-Haters* *(spamHaters)*

Spam-Haters are used to override Spam-Lovers. Example: If you have set your entire domain as a Spam-Lover(s), but there are still some addresses you still wish to block spam for. If you add those addresses to the Spam-Haters field allows messages to only those addresses to be blocked while still allowing the messages to the other Spam-Lovers pass through. The message will only be blocked if all recipients are Spam-Haters. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).
For example: *fribo@thisdomain.com|jhanna|@sillyguys.org

Bayesian Spam-Hater* *(baysSpamHaters)*

DNSBL Failures Spam-Hater* *(rblSpamHaters)*

HELO Blacklisted Spam-Hater* *(hlSpamHaters)*

☐ **Switch Spam-Lover to Message Scoring** *(switchSpamLoverToScoring)*

Put the filter automatically in "Message Scoring Mode" when **DoPenaltyMessage** is set (instead of stopping spam processing altogether).

Notes On Spam-Lover

Notes

No Processing - IP's, Domains, Addresses and Limits

No Processing IPs* ***(noProcessingIPs)***

file:files/ipnp.txt [Edit file]

Mail from any of these IP's will pass through without processing. (some attachments may be processed)
For example: 145.145.145.145|146.145.
To define IP's only for specific email addresses or domains (recipients) you must use the file:... option
An entry (line) may look as follows:
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

It is possible to define a predefined group on any or both sides of the '=>' separator, like:
[ipgroup]=>[usergroup]|user@mydomain

NOTICE: the following combination of two entries, will lead in to a user/domain based matching - the global entry will be ignored!
145.146.0.0/16 # comment
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

All fields marked by '*' accept a filepath/filename : 'file:files/ipnp.txt'.

No Processing Addresses* ***(noProcessing)***

Mail solely to or from any of these addresses are proxied without processing. The envelope sender and recipients are checked. Like a more efficient version of Spam-Lovers & redlist combined. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com). If you register TO addresses here, all recipients for a single mail must be marked as noprocessing to flag the mail as "noprocessing".

No Processing Addresses From* ***(noProcessingFrom)***

Mail solely from any of these addresses are proxied without processing. Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com). Wildcards are supported (fribo*@example.com).

No Processing Domains* ***(noProcessingDomains)***

sourceforge.net
Domains from which you want to receive all mail and proxy without processing. Your ISP, domain registration, mail list servers, stock broker, or other key business partners might be good candidates. Note this matches the end of the address, so if you don't want to match subdomains then include the @. Note that buy.com would also match spambuy.com but .buy.com won't match buy.com. For example: sourceforge.net|@google.com|.buy.com

Regular Expression to Identify No Processing Mail* ***(npRe)*** (.*)

If a message matches this Perl regular expression ASSP will treat the message as a 'No Processing' mail. For example: 169\.254\.122\.|172\.16\.|\[autoreply\].

Message Size Limit ***(npSize)***

500000
ASSP will treat incoming messages larger than this SIZE (in bytes) as 'No Processing' mail, after the header part of the mail and **MaxBytes** of the mail body are received. IP-, handshake- and header- checks will be done regardless the noprocessing flag (which is in this case ignored for these checks), all actions that require the full mail are skipped. Empty or 0 disables the feature.
Please see also **neverQueueSize** .

Message Size Limit Outgoing ***(npSizeOut)***

500000
ASSP will treat outgoing messages larger than this SIZE (in bytes) as 'No Processing' mail, after the header part of the mail and **MaxBytes** of the mail body are received without any error. Empty or 0 disables the feature.
Please see also **neverQueueSize** .

Process Only These Addresses* ***(processOnlyAddresses)***

If the Enable Process Only Addresses check box is checked, mail solely to or from any of the addresses in this list (envelope only) will be processed by ASSP. All others will be proxied without processing. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).
Note that if an address matches both the NoProcessing and the OnlyTheseProcessing lists, the NoProcessing rules take precedence.

☐ **Enable Process Only Addresses** ***(poTestMode)***

Notes On No Processing
[Notes]

Whitelisting and RWL(DNSWL)

Whitelisted IPs* ***(whiteListedIPs)***

file:files/ipwl.txt [Edit file] [edit Groups file]

They contribute to the Whitelist and to Notspam. For example: 145.145.145.145|146.145.|146.145.0.0/16. It is recommended to use the CIDR notation.
To define IP's only for specific email addresses or domains (recipients) you must use the file:... option
An entry (line) may look as follows:
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

It is possible to define a predefined group on any or both sides of the '=>' separator, like:
[ipgroup]=>[usergroup]|user@mydomain

NOTICE: the following combination of two entries, will lead in to a user/domain based matching - the global entry will be ignored!
145.146.0.0/16 # comment
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

All fields marked by '*' accept a filepath/filename : 'file:files/ipwl.txt'.

Regular Expression to Identify Non-Spam* ***(whiteRe)***

If an incoming email matches this Perl regular expression, it will be considered whitelisted.
For example: Secret Ham Password|307\D{0,3}730\D{0,3}4[12]\d\d
For help writing regular expressions click **here**.
IMPORTANT: The body is scanned in a later stage AFTER all sender related checks are performed. So a white regular expression here might not prevent the message to be blocked by eg. invalid PTR. Set the sender related checks to score only if you want to make sure that the white regular expression will be seen. Some things you might include here are your office phone number or street address, spam rarely includes these details. .

Whitelisted Domains and Addresses* ***(whiteListedDomains)***

file:files/whitedomains.txt [Edit file]

Domains and addresses from which you want to receive all mail. Your ISP, domain registration, mail list servers, stock broker, or other key business partners might be good candidates. Be careful not to put widely used or local domains here like google.com or hotmail.com or mydomain.com. Note this matches the end of the address, so if you don't want to match subdomains then include the @. Note that example.com would also match spamexample.com but .example.com won't match example.com. Wildcards are supported. For example: sourceforge.net|group*@google.com|.example.com

It is possible to make email addresses whitelisted only for a set of local domains and/or local users. Use wildcards (* and ?) to define domains. Use the following syntax to do this:
@anydomain=>@any_local_domain - for domain to domain
@.anydomain=>*@any_local_domain - for any sub-domain to domain
user@anydomain=>*@*.any_local_domain - for user to any sub-domain

It is possible to define more than one entry at the left and the right side of the definition (=>), like:
@anydomain|@other_domain=>*@any_local_domain|*@other_local_domain - always separate multiple entries by pipes
It is also possible to use a GroupDefinition in any or both sides, like:
[sendergroup]=>[recipientgroup]
[sendergroup1]|[sendergroup2]|*@domain=>[recipientgroup1]|[recipientgroup2]|user@local_domain

NOTICE - that the local email addresses and domains are not checked to be local once

☐ **Enable Realtime Whitelist Validation** ***(ValidateRWL)***

RWL: Real-time white list. These are lists of IP addresses that have somehow been verified to be from a known good host. Senders that pass RWL validation will pass IP-based filters. This requires an installed **Net::DNS** module in PERL.

☐ **Whitelist all RWL Validated Addresses** ***(RWLwhitelisting)***

If set, the message will also pass Bayesian Filter and URIBL.

RWL Service Providers* ***(RWLServiceProvider)***

file:files/dnsrws.txt [Edit file]

Host Names of RWLs to use separated by "|".
Examples are:
list.dnswl.org|query.bondedsender.org|cml.anti-spam.org.cn|iadb.isipp.com|hul.habeas.com

If you use a local provider of the list.dnswl.org zone, your local provider zone name has to contain 'list.dnswl.org' - for example:
list.dnswl.org.yourdns.local

because list.dnswl.org provides special return codes (127.0.X.Y)
where X defines the category and Y the trust value!

For list.dnswl.org or any equivalent local provider, it is possible to override the reported trust value based on the reported category. To do this, use the following syntax in the service provider definition:
serviceprovider:category=>trust_value[,category_from-category_to=>-trust_value][,*=>+trust_value]
* is used, if no other match is found. Any or all categories may be defined for the override. If no override is found for a category, the reported trust value is used.
+ and - are math operations to the reported trust value.

The currently by dnswl.org provided categories are:

2 = Financial services

3 = Email Service Providers
4 = Organisations
5 = Service/network providers
6 = Personal/private servers
7 = Travel/leisure industry
8 = Public sector/governments
9 = Media and Tech companies
10 = some special cases
11 = Education, academic
12 = Healthcare
13 = Manufacturing/Industrial
14 = Retail/Wholesale/Services
15 = Email Marketing Providers

The returned trust values by list.dnswl.org are:

0 = none
1 = low
2 = medium
3 = high

override example: list.dnswl.org:15=>0,2=>+1,5=>-2
For list.dnswl.org set the trust for category 15 to zero regardless the reported trust value, increase the trust value by one for category 2 and decrease the trust value for the category 5 by 2.

Maximum Replies ***(RWLmaxreplies)***

4

A reply is affirmative or negative reply from a RWL. The RWL module will wait for this number of replies (negative or positive) from the RWLs listed under Service Provider for up to the Maximum Time below. This number should be equal to or less than the number of RWL Service Providers listed to allow for randomly unavailable RWLs.

Minimum Hits ***(RWLminhits)***

1

A hit is an affirmative response from a RWL. The RWL module will check all of the RWLs listed under Service Provider, and flag the email with a RWL pass flag if equal to or more than this number of RWLs return a positive whitelisted response. This number should be less than or equal to Maximum Replies above and greater than 0

Maximum Time ***(RWLmaxtime)***

10

This sets the maximum time to spend on each message performing RWL checks

Don't Validate RWL for these IPs* ***(noRWL)***

Enter IP addresses that you don't want to be RWL validated, separated by pipes (|). For example: 145.145.145.145|146.145.

☑ **Add X-Assp-Received-RWL Header** ***(AddRWLHeader)***

Add X-Assp-Received-RWL header to header of all mails processed by RWL.

RWL Cache Refresh Interval ***(RWLCacheInterval)***

7

IP's in cache will be removed after this interval in days. 0 will disable the cache. Show RWL Cache

PrivacyLevel of the Whitelist ***(WhitelistPrivacyLevel)***

global & private(legacy)

Sets the privacy level of the **whitelistdb** . If a (local) user adds an email address to the whitelist:

(0) global & private - this email address is automatically whitelisted for all other local users
(1) domain & private - this email address is automatically whitelisted for all other local users in the same local domain
(2) private only - this email address is only whitelisted for this single local user

(0-1) unless another user has removed this email address from his whitelist. Default is zero, which is the legacy setting.
NOTICE: independent from this setting, the **whitelistdb** is filled with all three entries (global,domain,private), to make it possible to change this value.

Max Whitelist/Personal Black Days ***(MaxWhitelistDays)***

365

This is the number of days an address will be kept on the whitelist and personal blacklist without any email to/from this address. Set it to 0 to keep the entries infinity.

☐ **Reject All But Whitelisted Mail** ***(WhitelistOnly)***

Check this if you don't want Bayesian filtering and want to reject all mail from anyone not whitelisted. To do this related to local user addresses, use **InternalAndWhiteAddresses** and switch this option off.

☐ **Only Email-Interface Addition to Whitelist.** ***(NoAutoWhite)***

Check this box to allow additions to the whitelist by email interface only.

No AutoWhite Addresses* ***(NoAutoWhiteAdresses)***

Mail solely to or from any of these addresses are excluded from automatic whitelist additions. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Senders are compared/added to the whitelist *(NotGreedyWhitelist)*

check all addresses - one white match - add all

Normal operation includes addresses in the FROM, SENDER, REPLY-TO, ERRORS-TO, or LIST-* header fields.
This allows nearly all list email to be whitelisted.
If set to 'check all addresses - one white match - add all', one match in any of this fields is enough to get white and all addresses will be added to whitelist.
If set to 'consider whitelisted, if the envelop sender is white', only the envelope sender address is compared and possibly updated.
If set to 'check all addresses - all must matches for white - update all', all found sender addresses in all fields must be already whitelisted for a message to get a whitelisted state and all addresses will updated in whitelist. Notice: this setting will overwrite a match in **whiteListedDomains** , if a not whitelisted sender is found.
If any address is found in redlist, no whitelist addition will be done and the message gets not white.
If the penalty score of a message has reached **PenaltyMessageLow**, no whitelist addition will be done.
This setting is ignored, for mails to add/remove whitelist entries via email-interface.

How add Greedy Senders to Whitelist *(GreedyWhitelistAdditions)*

envelope only

Defines what sender addresses are added to the whitelist if a message is considered to be from a whitelisted sender. **NotGreedyWhitelist** is considered in determining if a message is from a whitelisted sender.

☐ **Only local or authenticated users contribute to the whitelist. *(WhitelistLocalOnly)***

Normal operation allows all local, authenticated, or whitelisted users to contribute to the whitelist.
Check this box to not allow whitelisted users to add to the whitelist.

☑ **Only users with a local domain in mailfrom contribute to the whitelist. *(WhitelistLocalFromOnly)***

Check this box to prevent sender with non-local domains from contributing to the whitelist. (for example: redirected messages).

☐ **Whitelist mails from authenticated users. *(WhitelistAuth)***

Mails from authenticated users will be processed as whitelisted.

Save Whitelist [s] *(UpdateWhitelist)*

3600

Save a copy of the white list every this many seconds. Empty or Zero will prevent any saving and the cleanup of old records.

Notes On Whitelist

Notes

Local Recipients and Domains & Transparent Recipients and Domains

Mails to these Recipients are Handled in Transparent-PROXY Mode* ***(transparentRecipients)***

Mails to any of these recipients or domains are handled transparent immediatly **after** a possible SRS check, BATV processing, Recipient-Replacement, RFC822 checks, ORCPT check and a feature match is found in the currently processed "RCPT TO:" SMTP command (envelope recipient).
What means "transparent handled" ? ASSP acts like a transparent Proxy. No filter actions are taken for the mail. Nothing is analyzed. Nothing is verfied. Nothing is stored. Nothing is logged (except reply codes if configured) - only debugging will work.
NOTICE: If a connection is moved in to the transparent proxy mode, this connection will stay in this mode until "MAIL FROM:" or "RSET" is used or the connection is closed by any peer.
You can list specific addresses (user@mydomain.com), addresses at any local domain (user), or entire domains (@mydomain.com). Wildcards are supported (fribo*@domain.com). (|).
For example: fribo@thisdomain.com|jhanna|@sillyguys.org or place them in a plain ASCII file one address per line - file:files/transparentuser.txt.

☐ **remove Foreign BCC** ***(removeForeignBCC)***

Remove foreign BCC: header lines from the mail header. The remove is done before the **DoHeaderAddrCheck** is done!

☐ **Check TO,CC and BCC headers** ***(DoHeaderAddrCheck)***

If enabled TO: , CC: and BCC: header lines are checked the following way:

1. a possible recipient replacement is done
2. local email address validation is done - if OK, the next address or headerline is processed
3. **spamtrapaddresses** will be detected - scored with **stValencePB** - **mail is blocked** (**noPenaltyMakeTraps** is honored)
4. a local but not valid TO/CC/BCC: address will be detected - scored with **irValencePB**
5. a RelayAttempt will be detected if a BCC address is not local - scored with **rlValencePB** - **mail is blocked**

The check 3 and 4 honors whitelisting , noprocessing and noBlockingIPs
Enable this check only, if assp is configured to validate local domains and email addresses!
NOTICE: that **removeForeignBCC** take place before this check is done - step 5 will be never reached if **removeForeignBCC** is enabled!
Using this feature can lead in to alot of address lookups. LDAP or VRFY address verifications may take a very long (possibly too long) time!

Catchall Address for Messages to Postmaster ***(sendAllPostmaster)***

ASSP will deliver messages addressed to all postmasters of your local domains to this address. For example: postmaster@mydomain.com

☐ **Skip Spam Checks for Postmaster Catchall** ***(sendAllPostmasterNP)***

Catchall Address for Messages to Abuse ***(sendAllAbuse)***

ASSP will deliver messages to all abuse addresses of your local domains to this address. For example: abuse@mydomain.com

☐ **Skip Spam Checks for Abuse Catchall** ***(sendAllAbuseNP)***

Validate addresses to conform with RFC 822 ***(DoRFC822)***

sender

If activated, the envelope sender and/or each envelope recipient is checked to conform with the email format defined in RFC 822. For an invalid sender address '**nofromValencePB**' is used for scoring - for invalid recipient addresses, each is scored with **irValencePB** .
For the sender address in addition a top level domain existence and DNS name server registration check is done.
The default setting is 'sender' - recommended settings are 'sender' or 'both'!

Lookup valid Local Addresses from here* ***(LocalAddresses_Flat)***

These email addresses are the list of your local addresses. You can list specific addresses (user@mydomain.com), addresses at any local domain (user), or entire domains (@mydomain.com). Wildcards are supported (fribo*@domain.com). (|).
For example: fribo@thisdomain.com|jhanna|@sillyguys.org or place them in a plain ASCII file one address per line - file:files/localuser.txt.
NOTICE: The VRFY definition described below is depricated in this configuration parameter - use localDomains instead!
You can use entries like @mydomain.com=>[SSL:]vrfyhost:port to VRFY users on your MTA, for more information read **localDomains**. You can use an entry like ALL=>vrfyhost:port to define a VRFY host for all domain entries (better use **Groups**).
If the port :465 is defined for VRFY-MTA, or "SSL:" is prepended to the VRFY-MTA, a SSL connection will be used (read **DoVRFY**).
Notice: If an equal domain entry is defined in **localDomains** , the entry in **localDomains** will be used!
If you define **only one domain definition line - using ALL**
ALL=>[SSL:]vrfyhost:port
here and **ldaplistdb** is configured and **DoVRFY** is enabled and **LDAPFail** is set to ON, local domains will be additionaly collected in to **ldaplistdb** from verfied addresses, domains and URL's (eg. **DoLocalSenderAddress** , local recipient checks). The postmaster account must exists for every local domain and subdomain at the MTA!
Using such a configuration, you must know what you are doing and have a properly configured MTA! Be carefull, the URIBL check (ValidateURIBL) can lead in to alot of domain lookups and verifications (possibly several hundred per mail). The same applies to the header recipient address validation (DoHeaderAddrCheck)!

☐ **Use Addresses without '@' as Domains** ***(LocalAddresses_Flat_Domains)***

Will handle entries without '@' as full domains

Reject These Local Addresses* ***(RejectTheseLocalAddresses)***

If ANY recipient is on reject list, message will not be delivered. Used for disabled legitimate accounts, where a user may have left the company. This stops wildcard mailboxes from getting these messages.

Local Domains* ***(localDomains)***

putYourDomains.com|here.org

Check local domains against these addresses. Add a fake domain like 'assp-nospam.org' for the email interface if you run MS Exchange. When mailing to eg. 'spam@assp-nospam.org' MS Exchange forwards it outbound to ASSP who handles the different options. As in every field marked by '*' separate addresses with | or use file 'file:files/localdomains.txt'. Wildcards are supported.
For example: *mydomain.com|*.mydomain.com|here.org
Use the syntax:
*mydomain.com=>smtp.mydomain.com|other.com=>SSL:mx.other.com:port|other2.com=>mx.other.com:port,mx2.other.com:port to verify the recipient addresses with the SMTP-VRFY (if VRFY is not supported 'MAIL FROM:' and 'RCPT TO:' will be used) command on other SMTP servers.
The entry behind => must be the hostname:port or ip-address:port of the MTA which is used to verify 'RCPT TO' addresses with a VRFY command! If :port is not defined, port :25 or :465 (in case SSL: is defined) will be used.br /> You can use an entry like ALL=>vrfyhost:port to define a VRFY host for all local domain entries that don't have a MTA defined (better use **Groups**). Separate multiple VRFY hosts for failover by comma ",". You have to enable the SMTP 'VRFY' command on your MTA - the 'EXPN' command should be enabled! This requires an installed **Net::SMTP** module in PERL.
If the port :465 is defined for VRFY-MTA, or "SSL:" is prepended to the VRFY-MTA, a SSL connection will be used (read **DoVRFY**).
If you have configured LDAP and enabled **DoLDAP** and ASSP finds a VRFY entry for a domain, LDAP search will be done first and if this fails, the VRFY will be used. So VRFY could be used for LDAP backup/fallback/failover!
It is recommended to configure '**ldaplistdb**' in the 'File Paths and Database' section when using this verify extension - so ASSP will store all verified recipients addresses there to minimize the queries on MTA's. There is no need to configure LDAP, but both VRFY and LDAP are using **ldaplistdb**. Please go to the 'LDAP setup' section to configure **MaxLDAPlistDays** and **LDAPcrossCheckInterval** or start a crosscheck now with **forceLDAPcrossCheck**. This three parameters belong also to VRFY.
Notice: if an equal domain entry is defined in **LocalAddresses_Flat** (depricated !!!), the entry in **localDomains** will be used!

☑ **Verify Recipients with SMTP-VRFY** ***(DoVRFY)***

If activated and the format 'Domain=>MTA:Port' is encountered in **localDomains** and/or **LocalAddresses_Flat**, recipient addresses will be verified with SMTP-VRFY (if VRFY is not supported 'MAIL FROM:' and 'RCPT TO:' will be used). If you know that VRFY is not supported with a MTA, you may put the MTA into **VRFYforceRCPTTO**. Don't forget to configure **LDAPFail** (belongs also to VRFY) to your needs!
If the SMTP-SSL port :465 is defined with a MTA, or "SSL:" is prepended to the MTA definition and the module IO::Socket::SSL is available, a SSL connection will be used for the SMTP-VRFY-session.

☐ **Enable STARTTLS for VRFY** ***(enableTLS4VRFY)***

If enabled and the module IO::Socket::SSL is available and STARTTLS is supported by the VRFY-MTA and the SMTP-VRFY-session is not in SSL-mode, assp will try to use the STARTTLS command to secure the SMTP-VRFY-session.

SMTP VRFY-Query Timeout ***(VRFYQueryTimeOut)***

5

The number of seconds ASSP will wait for an answer of the MTA that is queried with the VRFY command to verify a recipient address.

Force the usage of RCPT TO* ***(VRFYforceRCPTTO)***

Define MTA's here for which you want ASSP to force the usage of MAIL FROM:,RCPT TO: instead of the VRFY command. The definition of each MTA has to be the same as defined in **LocalAddresses_Flat** and/or **localDomains** (after the '=>') for example:
smtp.mydomain.com|SSL:mx.other.com:port|10.1.1.1|10.1.1.2:125 .

☐ **Disable VRFY and EXPN for External Clients** ***(DisableVRFY)***

If you have enabled VRFY and/or EXPN on your MTA to make assp able to verify addresses and you do not want external clients to use VRFY and EXPN - select this option.

☐ **Do LDAP lookup for valid local addresses** ***(DoLDAP)***

Check local addresses against an LDAP database before accepting the message.
Note: Checking this requires filling in the other LDAP parameters below.
This requires an installed **Net::LDAP** module in PERL.

☐ **Do Not Validate Local Addresses if in NoProcessing List** ***(LocalAddressesNP)***

If a recipient is found in NoProcessing, the user validation is skipped.

Catchall per Domain* ***(CatchAll)***

ASSP will send to this addresses/domain if no valid user is found in LocalAddresses_Flat/LDAP.
For example: catchall@domain1.com|catchall@domain2.com

Catchall for All Domains ***(CatchAllAll)***

ASSP will send to this address if no valid user is found in LocalAddresses_Flat/LDAP and no match is found in Catchall per Domain.
For example: catchall@domain.com

☐ **Move ISP Connection with wrong Recipient Address to NULL** ***(CatchallallISP2NULL)***

If set, ASSP will move all ISP connections with wrong recipient addresses to a NULL-connection. The ISP will receive "250 OK" until the mail has passed, but the mail will not be sent to your MTA. This is done after **CatchAll** but before **CatchAllAll** is checked.

NULL Connection Addresses* ***(NullAddresses)***

ASSP will dump a message silently when encountering such an address in "MAIL FROM:" or "RCPT TO:". Accepts specific addresses (null@example.com), user parts (nobody) or entire domains (@example.com).

Accept Mail from Local Domains only* ***(InternalAddresses)***

These local addresses accept mail only from local domains. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Accept Mail from Local Domains and Whitelisted Senders only* ***(InternalAndWhiteAddresses)***

These local addresses accept mail only from local domains and whitelisted external senders. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Separation Character for Subaddressing ***(SepChar)***

+

RFC 3598 describes subaddressing with a Separation Character. A star ('*') is not allowed as Separation Character. Everything between Separation Character and @ is ignored (including Separation Character). For Example = '+' will allow user+subaddress@domain.com.

☐ **Support Bang Path** ***(EnableBangPath)***

If set, ASSP will support addresses like domainx!user and will convert them to user@domainx .

Maximum recipient verification Errors ***(MaxVRFYErrors)***

0

The maximum number of failed 'RCPT TO' or 'VRFY' commands encountered before the connection is dropped. You can leave this field at 0, if you are using '**DoLDAP**', '**LocalAddresses_Flat**'! If configured, ASSP will drop the connection, if the count of '550 unknown user' errors, received from your 'smtpDestination'(MTA), reached this value!

Block Max Duplicate Recipients ***(DoMaxDupRcpt)***

score

Block remote servers that uses the same recipient address more times, than the number defined in **MaxDupRcpt** in the RCPT TO: command. Scoring is done with **mdrValencePB** . This check is skipped for outgoing, noprocessing, whitelisted and spamlovers mails. If a message has to be delayed, this check will score before the delay if set to block or score - and score and/or block on the next server request.

Maximum Allowed Duplicate Recipient Addresses ***(MaxDupRcpt)***

0

The maximum number of duplicate recipient addresses that are allowed in the sequence of the RCPT TO: commands!
The number per mail is calculated by 'number of RCPT TO: commands - number of unique recipient addresses'.
For example: if one address is used three times or two addresses are used each two times, will result in the same count - 2. Or if both is the case in one mail, the count will be 4.

Enable recipient replacement* ***(ReplaceRecpt)***

file:files/rcptreplrules.txt Edit file

recommented if used: file:files/rcptreplrules.txt - default empty ! This enables recipient replacement. If you do not use file:, separate the rules with |. The replacement will be done before any ASSP check. Use this option carefully - for example: if you have enabled DKIM check, the DKIM check will fail, if the recipient of the mail was modified. For a more detailed description of the rules and options, read the file: files/rcptreplrules.txt!

No-Valid-Local-User Reply ***(NoValidRecipient)***

550 5.1.1 User unknown: EMAILADDRESS

SMTP reply for invalid Users. Default: '550 5.1.1 User unknown: EMAILADDRESS'
The literal EMAILADDRESS (case sensitive) is replaced by the fully qualified SMTP recipient (e.g., thisuser@yourcompany.com).

Notes On Local Addresses

Notes

Validate HELO and EHLO

Regular Expression to Immediatly Blocked Invalidate HELO* *(invalidHeloRe)*

ylmf-pc

Regular expression to check and block immediatly incoming HELOs.
This option blocks and drops a connection immediatly by sending a reply "554 5.7.1 the connection is rejected - bad host identity detected", if the sent HELO matches this regular expression. The check is done immediatly after the HELO is sent by a peer. It can be used to block BOT's, that are using different IP's but the same or similar HELO every time.
Notice: this option ignores all other settings in assp!
The penalty box score for the connected IP is increased by **ihValencePB** .
For example: ylmf-pc

Use the Helo Blacklist *(useHeloBlacklist)*

score

Use the list of blacklisted-helo hosts built by rebuildspamdb.

Use the Helo Goodlist *(useHeloGoodlist)*

bonus

Use the list of known good helo hosts built by rebuildspamdb.
bonus - the message/IP get a bonus of the weighted negative value of **hlValencePB**
whitelisted - the message is processed as whitelisted

The good helos and weights are stored together with the helo blacklist.

Do Score Suspicious Helos *(DoIPinHelo)*

score

Score servers with IP number in Helo and check for mismatch with sending IP.

☑ **Enforce Check of Forged Helos Before Delaying *(ForceFakedLocalHelo)***

If set, ASSP will check Forged Helos before DELAYING. Collecting, Testmode, CopySpam, Spam-Lover and private/domain whitelist (**WhitelistPrivacyLevel**) is ignored.

Block Forged Helos *(DoFakedLocalHelo)*

block

Block remote servers that claim to come from our Local Domains/Local IP's/Local Host.

☑ **Use Local Domain List for Blocking Forged Helos *(DoFakedUseLocalDomain)***

If set, **DoFakedLocalHelo** will use **localDomains**.

☐ **Do Not Block Whitelisted *(DoFakedWL)***

Disable "Block Forged Helo's" for whitelisted addresses (not recommended).

☐ **Do Not Block Noprocessing *(DoFakedNP)***

Disable "Block Forged Helo's" for addresses identified as noprocessing (not recommended).

Local Domains,IP's and Hostnames* *(myServerRe)*

Local Domains, IP's and Hostnames are often use to fake (forge) the Helo. Include all IP addresses and hostnames for your server here, localhost is already included. Include Local Domains of your choice here, if you deactivated the automatic use of the local domain list. For example: 11.22.33.44|mx.YourDomains.com|here.org

Don't Validate HELO for these IP's* *(noHelo)*

127.0.0.0/8|::1

Enter IP addresses that will be excluded from all HELO checks. Default setting is 127.0.0.0/8|::1.
For example: 127.0.0.1|::1|192.168.

Don't process these HELO's* *(heloBlacklistIgnore)*

HELO / EHLO greetings on this list will be excluded from all HELO checks. For example: host123.isp.com|host456.*.com

☑ **Enforce Early Helo Checks *(ForceValidateHelo)***

If set, ASSP will Validate/Invalidate the format of HELO before DELAYING. Collecting, Testmode, CopySpam, Spam-Lover and private whitelist (**WhitelistPrivacyLevel**) is ignored.

Validate Format of HELO *(DoValidFormatHelo)*

block

If activated, the HELO is checked against the expression below. If the Regular Expression matches, the HELO is validated as being ok.

Regular Expression to Validate Format of HELO* *(validFormatHeloRe)*

file:files/validhelo.txt Edit file

Validate Format HELO will check incoming HELOs according to rfc1123.
For example: ^(?:\w[\w\.\-]*\.\w{2,6})$ or ^(?:(?:[a-z\d][a-z\d\-]*)?[a-z\d]\.)+[a-z]{2,6}$

Invalidate Format of HELO *(DoInvalidFormatHelo)*

block
If activated, the HELO is checked against the expression below. If the Regular Expression matches, the HELO is invalidated as being not ok.

Regular Expression to Invalidate Format of HELO** ***(invalidFormatHeloRe)***

file:files/invalidhelo.txt Edit file

Invalidate Format HELO will check incoming HELOs for this.
For example: ^\d+\.\d+\.\d+\.\d+$|^[^\.]+\.?$,

☑ **Do Valid/Invalid/Black Helo for Whitelisted** ***(DoHeloWL)***

Do valid/invalid Helo for whitelisted addresses.

☑ **Do Valid/Invalid/Black Helo for Noprocessing** ***(DoHeloNP)***

Do valid/invalid Helo for noprocessing addresses.

Notes On Validate Helo
Notes

Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

Do Blacklisted Addresses and Domains *(DoBlackDomain)*

block

☑ **Do Blacklisting Addresses and Domains for White *(DoBlackDomainWL)***

Do blacklisting addresses & domains in messages which are marked whitelisted by **whiteRe**, **whiteListedDomains**, **whiteListedIPs**, **whitelistdb**, **DoOrgWhiting** or **ValidateRWL** .

☑ **Do Blacklisting Addresses and Domains for NoProcessing *(DoBlackDomainNP)***

Do blacklisting addresses & domains in messages which are marked noprocessing by **npRe**, **noProcessingDomains**, **noProcessingIPs** or **noProcessing**.

Blacklisted Addresses and Domains* *(blackListedDomains)*

file:files/blackdomains.txt Edit file

Addresses & Domains from which you always want to reject mail, they only send you spam. Note this matches the end of the address, so if you don't want to match subdomains then include the @. Note that buy.com would also match spambuy.com but .buy.com won't match buy.com. abc@def.com will match abc@def.com but won't match bbc@def.com. Wildcards are supported. For example: cc|info|biz|seller@bayer.com|sell*@basf.com

It is possible to make email addresses blacklisted only for a set of local domains and/or local users. Use wildcards (* and ?) to define domains. Use the following syntax to do this:
@anydomain=>@any_local_domain - for domain to domain
@.anydomain=>*@any_local_domain - for any sub-domain to domain
user@anydomain=>*@*.any_local_domain - for user to any sub-domain

It is possible to define more than one entry at the left and the right side of the definition (=>), like:
@anydomain|@other_domain=>*@any_local_domain|*@other_local_domain - always separate multiple entries by pipes
It is also possible to use a GroupDefinition in any or both sides, like:
[sendergroup]=>[recipientgroup]
[sendergroup1]|[sendergroup2]|*@domain=>[recipientgroup1]|[recipientgroup2]|user@local_domain

NOTICE - that the local email addresses and domains are not checked to be local once

Check Message IDs *(DoMsgID)*

score

Score messages with missing/suspicious/invalid Message-ID. Scoring is done by **midmValencePB** / **midsValencePB** / **midiValencePB** .

Don't Validate Message-IDs for these IPs* *(noMsgID)*

127.0.0.|192.168.|10.

Enter IP addresses that you don't want to be Message-ID validated, separated by pipes (|). For example: 127.0.0.1|192.168.

Regular Expression to Validate Format of Message-ID* *(validMsgIDRe)*

^.+\@.+\..+$

Check Message IDs will check incoming messages for valid Message-IDs.
For example: ^.+\@.+\..+$

Regular Expression to Invalidate Format of Message-ID *(invalidMsgIDRe)***

Check Message IDs will check incoming messages for invalid Message-IDs.

Validate Remote Sender with Local Domain Address *(DoNoValidLocalSender)*

block

If activated, each remote sender with a local domain is checked against the *Local Addresses File* and/or LDAP.

☑ **Early "Remote Sender with Local Domain Address" Check *(ForceNoValidLocalSender)***

If set, ASSP will check Remote Sender with Local Domain Address before Delaying a message.
Collecting, Testmode, CopySpam, and Spam-Lover settings are ignored.

Block Local Address from External Sender *(DoNoSpoofing)*

score

If activated, each external sender address built with a domain in **localDomains** is regarded a spoofed address. An external sender is a sender from an IP not in **acceptAllMail** and not authenticated. Scoring is done with **slValencePB**.

Do Spoofing Check ONLY for these IP's* *(onlySpoofingCheckIP)*

Enter IP's that you want to be checked for spoofing. If this is set, ONLY these IP's will be checked. For example:145.145.145.145|145.146.

Do Spoofing Check ONLY for these Addresses/Domains* *(onlySpoofingCheckDomain)*

Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com). Wildcards are supported (fribo*@example.com). If set, ONLY these addresses/domains will be checked for spoofing.

Don't do Spoofing Check for these IP's* *(noSpoofingCheckIP)*

Enter IP's that you don't want to be checked for spoofing. For example:145.145.145.145|145.146.

Don't do Spoofing Check for these Addresses/Domains* ***(noSpoofingCheckDomain)***

Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com). Wildcards are supported (fribo*@example.com).

☐ **Do NoSpoofing for from:** ***(DoNoSpoofing4From)***

Do the NoSpoofing check also for header 'from:' addresses.

Reversed Lookup ***(DoReversed)***

score

If activated, each sender IP is checked for a PTR record. This requires an installed **Net::DNS** module in PERL.

☑ **Do Reversed Lookup for Whitelisted** ***(DoReversedWL)***

Do reversed lookup for whitelisted addresses.

☑ **Do Reversed Lookup for Noprocessing** ***(DoReversedNP)***

Do reversed lookup for noprocessing addresses.

Reversed Lookup FQDN ***(DoInvalidPTR)***

score

If activated - and Reversed Lookup is activated -, the PTR-FQDN record is checked against the Regex. This requires an installed **Net::DNS** module in PERL.

Regular Expression to Invalidate Format of PTR** ***(invalidPTRRe)***

file:files/invalidptr.txt Edit file

Validate Format PTR will check PTR records for this.
For example: ^\d+\.\d+\.\d+\.\d+$|^[^\.]+\.?$ or file:files/invalidptr.txt

Regular Expression to Validate Format of PTR* ***(validPTRRe)***

file:files/validptr.txt Edit file

Validate Format PTR will check PTR records for this.
For example: static or file:files/validptr.txt

Reversed Lookup Cache Refresh Interval ***(PTRCacheInterval)***

7

IP's in cache will be removed after this interval in days. 0 will disable the cache. Show PTR Cache

Validate MX or A Record ***(DoDomainCheck)***

score

If activated, the sender address and each address found in the following header lines (ReturnReceipt:, Return-Receipt-To:, Disposition-Notification-To:, Return-Path:, Reply-To:, Sender:, Errors-To:, List-...:) is checked for a valid MX or A record. Scoring is done for non existing MX record and non existing A record - a messages fails (block), if both records are not found.

Validate Domain MX Cache Refresh Interval ***(MXACacheInterval)***

7

IP's in cache will be removed after this interval in days. 0 will disable the cache. Show MX Cache

Check for Existing From Header Tag and Address ***(DoNoFrom)***

score

If enabled, the MIME header is checked for a valid From: header tag. The scoring value is set with fromValencePB.

☑ **Do DoNoFrom for Whitelisted** ***(DoNoFromWL)***

Check for existing From header and address for whitelisted addresses.

☑ **Do DoNoFrom for NoProcessing** ***(DoNoFromNP)***

Check for existing From header and address for noprocessing addresses.

Remove Disposition Notification Headers ***(removeDispositionNotification)***

To remove any headers : "ReturnReceipt: , Return-Receipt-To: and Disposition-Notification-To:" from not whitelisted and not noprocessing incoming mails, define the unwanted headers as regular expression.
for example: Disposition-Notification-To
or: Disposition-Notification-To|Return-Receipt-To
or: Disposition-Notification-To|Return-Receipt-To|ReturnReceipt
or any other possible combination. Notice: do NOT define the trailing ":"!
Define this to prevent unwanted whitelisting of spammers that request a Disposition Notification. Another way to prevent autowhitelisting because of an autoresponder is to use **redRe** .

Validate DomainKeys Identified Mail ***(DoDKIM)***

score

If activated, DomainKeys Identified Mails are checked for the right signature and contents. All DKIM parameters belongs also to the old

DomainKey specification. This requires an installed **Mail::DKIM::Verifier** module in PERL. In addition DKIM is used to process Domain-based Message Authentication, Reporting & Conformance - described in **DMARC** (DMARC requires also **ValidateSPF** to be enabled).

☐ **Validate DomainKeys Identified Mail strictly *(DoStrictDKIM)***

The DKIM test will fail, if the mail was modified by a mailhop. In this case the from address, the from domain, the to domain, the DKIM-signature by itself and the prefix of the digest-verification are valid, only the lower digest value differs! This may happen, if a mailhop has modified any other headerfield like X-...! If unchecked a mail will only pass, if the author policy and sender policy are accept or neutral!

Do not any DKIM Check for these Addresses * *(noDKIMAddresses)*

Mail from or to any of these envelope addresses will not be tagged and checked for DKIM. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).

Exclude these IP's from any DKIM Check* *(noDKIMIP)*

Enter IP's that you want to exclude from DKIM check, separated by pipes (|).

Validate DKIM-Pre-Check-Cache Refresh Interval *(DKIMCacheInterval)*

7

Domains's in cache will be removed after this interval in days. 0 will disable the cache.
If activated a DKIM-pre-check will be done. If ASSP finds a DKIM-Signature in the mail header, it checks the DNS records of the sending domain for valid DKIM configurations and writes a record in to the DKIM-pre-check-cache, if it finds such configuration.
If ASSP does not find a DKIM-Signature in the mail header, it also checks the DNS records of the sending domain for valid DKIM configurations. If it find such a configuration, the mail is considered spam, because it should have a DKIM-Signature.
The next mail from a domain that is found in this cache, must have a DKIM-Signature to pass the DKIM-pre-check. How ever, some DNS records are wrong or inaccurate and will cause ASSP to block mails because of this - register such domains and/or IP's in **noDKIMAddresses** and/or **noDKIMIP** .

Show DKIM Cache

☑ **Add X-Assp-DKIM Header *(AddDKIMHeader)***

Add X-Assp-DKIM header.

Senders need to SMIME or PGP Sign All Mail* *(signedSenders)*

file:files/signedSenders.txt Edit file

Domains and addresses which have to SMIME or PGP sign or encrypt all mail. If a match is found for a sender and the email is not signed or encryped, the mail will be rejected!
If configured, this check is done regardless any other assp setting - it will affect all incoming mails!
If a match is found and the mails is signed or encrypted, the mail will be processed as whitelisted mail!
Note this matches the end of the address, so if you don't want to match subdomains then include the @. Note that example.com would also match spamexample.com but .example.com won't match example.com. Wildcards are supported. For example:
sourceforge.net|group*@google.com|.example.com

It is possible to make the senders signing requirement recipient dependend (eg: on a set of local domains and/or local users). Use wildcards (* and ?) to define domains.
Use the following syntax to do this:
@anydomain=>@any_local_domain - for domain to domain
@.anydomain=>*@any_local_domain - for any sub-domain to domain
user@anydomain=>*@*.any_local_domain - for user to any sub-domain

It is possible to define more than one entry at the left and the right side of the definition (=>), like:
@anydomain|@other_domain=>*@any_local_domain|*@other_local_domain - always separate multiple entries by pipes
It is also possible to use a GroupDefinition in any or both sides, like:
[sendergroup]=>[recipientgroup]
[sendergroup1]|[sendergroup2]|*@domain=>[recipientgroup1]|[recipientgroup2]|user@local_domain

NOTICE - that the local email addresses and domains are not checked to be local once

Sender Validation Error *(SenderInvalidError)*

554 5.7.1 REASON .

SMTP error message to reject invalid senders. The literal REASON is replaced by (missing MX, missing PTR, invalid Helo, invalid user, missing signature) depending on the check.

Notes On Validate Sender

Notes

IP Blocking

Simple IP Greylisting ***(DelayIP)***

Enable simple delaying for IP's in black penaltybox with totalscore above this value. A value of zero or empty disables this feature.

Simple IP Greylisting Embargo Time ***(DelayIPTime)***

5

Enter the number of minutes for which delivery, related with IP address of the sending host, is refused with a temporary failure. Default is 5 minutes.

Do Deny Connections from these IP's ***(DoDenySMTP)***

block

If activated, the IP is checked against (**denySMTPConnectionsFrom**) Deny Connections from these IP's.

Deny Connections from these IP's* ***(denySMTPConnectionsFrom)***

Manually maintained list of IP's which should be blocked. IP's in **noPB**, **noDelay**, **acceptAllMail**, **ispip**, **whiteListedIPs**, **noProcessingIPs**, whitebox (PBWhite) will pass. For example: file:files/blockip.txt.
To define IP's only for specific email addresses or domains (recipients) you must use the file:... option
An entry (line) may look as follows:
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

It is possible to define a predefined group on any or both sides of the '=>' separator, like:
[ipgroup]=>[usergroup]|user@mydomain

NOTICE: the following combination of two entries, will lead in to a user/domain based matching - the global entry will be ignored!
145.146.0.0/16 # comment
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

Do not block Connections from these IP's* ***(noBlockingIPs)***

Manually maintained list of IP's which should not be blocked. For example: 145.145.145.145|145.146.
To define IP's only for specific email addresses or domains (recipients) you must use the file:... option
An entry (line) may look as follows:
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

It is possible to define a predefined group on any or both sides of the '=>' separator, like:
[ipgroup]=>[usergroup]|user@mydomain

NOTICE: the following combination of two entries, will lead in to a user/domain based matching - the global entry will be ignored!
145.146.0.0/16 # comment
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

Do Deny Connections from these IP's Strictly ***(DoDenySMTPstrict)***

block

If activated, the IP is checked against ('**denySMTPConnectionsFromAlways**') Deny Connections from these IP's Strictly.

Deny Connections from these IP's Strictly* ***(denySMTPConnectionsFromAlways)***

file:files/denyalways.txt Edit file

Manually maintained list of IP's which should **strictly** be blocked after address verification and before body and header is downloaded. Contrary to *denySMTPConnectionsFrom* IP's in **noDelay**, **acceptAllMail**, **ispip**, **whiteListedIPs**, **noProcessingIPs**, whitebox (PBWhite) will **not** pass if listed here.

Do also Deny Connections from these IP's ***(DoDropList)***

disabled

If activated, the IP is checked against the Droplist in addition to '**denySMTPConnectionsFromAlways**' and/or '**denySMTPConnectionsFrom**'. The **droplist** is downloaded if a new one is available and contains the Spamhaus DROP List. See "http://www.spamhaus.org/drop/drop.lasso".

Drop also Connections from these IP's* ***(droplist)***

file:files/droplist.txt Edit file

Automatically downloaded (http://www.spamhaus.org/drop/drop.lasso) list of IP's which should be blocked right away. This list could be used in addition to **denySMTPConnectionsFrom** and/or **denySMTPConnectionsFromAlways**!

☐ **Do Strictly Deny Connections Early** ***(denySMTPstrictEarly)***

IP's in **denySMTPConnectionsFromAlways** will be denied right away.

Do an Enhanced Origin IP Address Detection in the Mail Header ***(enhancedOriginIPDetect)***

all but most origin

If selected, ASSP will analyze the mail headers "RECEIVED:" lines for IP's on the mail routing way to detect spam bots, that uses open relay or hijacked mail servers for mail delivery.
Local and private IP's, and IP's listed in **ispip**, **acceptAllMail**, **whiteListedIPs**, **noProcessingIPs**, **noDelay** and **noPB** will be ignored.
The detected IP's will be additionally checked for IP-Blocking, DNSBL and IP-Frequency - the same way like the connected IP. These IP's are also additionally used for the maximum mail size calculation in **MaxRealSizeAdr** and MaxRealSizeExternalAdr.
Default setting is 'all but most origin', which ignores the first of multiple detected public IP address, that was involved in the mail transport (possibly a user device).

Check Frequency - Maximum Connections Per IP *(DoFrequencyIP)*

disabled

Select the action, if **maxSMTPipConnects** is reached.

Maximum Frequency of Connections Per IP *(maxSMTPipConnects)*

10

The maximum number of SMTP connections an IP Address can make during the **IP Address Frequency Duration**. If a server makes more than this many connections to ASSP within the (**maxSMTPipDuration**) IP Address Frequency Duration it will be banned from future connections until the (**maxSMTPipExpiration**) IP Address Frequency Expiration is reached. This can be used to prevent server overloading and DoS attacks. 10 connections are typically enough. If left blank or 0, there is no limit imposed by ASSP. IP's in **noPB**, **noDelay**, **acceptAllMail**, **ispip**, **whiteListedIPs**, **noProcessingIPs**, whitebox (PBWhite) are excluded from SMTP session limiting, whitelisted and noprocessing addresses are honored

Maximum Frequency of Connections Per IP Duration *(maxSMTPipDuration)*

90

The window (in seconds) during which the (**maxSMTPipConnects**) IP Frequency (see above for more details) will be scrutinized for each IP. The default is 90 seconds.

Expiration of Maximum Frequency *(maxSMTPipExpiration)*

7200

The number of seconds that must pass before an IP address blocked by the (**maxSMTPipConnects**) IP Address Frequency setting is allowed to connect again. The default is 7200 (seconds) .

Check Number of IP's Per Domain *(DoDomainIP)*

disabled

This check is skipped if the IP and domain have passed the SPF-check. If **ValidateSPF** is enabled and an IP/Domain reaches the **maxSMTPdomainIP** limit, the MaintThread starts a background SPF check to prevent blocking good mails in future.

Limit Number of IP's Per Domain *(maxSMTPdomainIP)*

10

The number of IP(subnet) switches a domain may have during the (**maxSMTPdomainIPExpiration**) Limit Different IP's Per Domain Expiration. If a domain switches more often than this it will be banned from future connections until the Expiration is reached. This can be used to prevent server overloading and DoS attacks. 10 connections are typically enough. If left blank or 0, there is no limit imposed by ASSP. IP's in **noPB**, **noDelay**, **acceptAllMail**, **ispip**, **whiteListedIPs**, **noProcessingIPs**, whitebox (PBWhite) are excluded, whitelisted and noprocessing addresses are honored.

Expiration of Limit Number *(maxSMTPdomainIPExpiration)*

7200

The number of seconds that must pass before a domain blocked by the (**maxSMTPdomainIP**) Limit Subnet IP's Per Domain setting (see above for more details) is allowed to connect again. The default is 7200 (seconds).

Do Not Limit Different IP's For These Domains* *(maxSMTPdomainIPWL)*

gmx.de|t-online.de|yahoo.com|hotmail.com|gmail.com

This prevents specific domains from limiting. For example: yahoo.com|hotmail.*.com|gmail.com

Notes On IP Blocking

Notes

SenderBase and WhoisIP

☐ **SenderBase Testmode** *(sbTestMode)*

Use Whois Queries instead or after or before of SenderBase Queries *(enableWhois)*

disabled

If enabled, WHOIS queries to IP-Whois-Servers

"ARIN" => "whois.arin.net" - (which will possible redirect to)

"RIPE" => "whois.ripe.net"
"APNIC" => "whois.apnic.net"
"KRNIC" => "whois.krnic.net"
"LACNIC" => "whois.lacnic.net"
"AFRINIC" => "whois.afrinic.net"

will be done instead/after/before (WHOIS only/SenderBase first/WHOIS first) the Senderbase queries to CISCO's Ironport servers to get informations about an IP address. ARIN will be the first queried WHOIS server.
For the two '...first' options, the alternative second check is done, if the first check fails or assp has got no result for the county code.
This is useful, if your DNS-servers don't get answers for senderbase queries or senderbase queries are too slow.
In most cases WHOIS queries are much more faster than senderbase queries!
NOTICE: you must open the WHOIS-port (43) for TCP on your firewall for outgoing traffic from assp (if not already done)!

Do Organization Whiting *(DoOrgWhiting)*

whiting

If activated, each sending IP address has its assigned organization looked up. Scoring is set with **sworgValencePB**.

Whitelisted Organizations, Domains and Hosts in SenderBase** *(whiteSenderBase)*

file:files/whiteorg.txt Edit file

If the organization, domain or hostname in the **SenderBase** IP description matches this Perl regular expression, the message will be considered non-spam. For example file:files/whiteorg.txt
NOTICE: If only the hostname matches an entry and **DoOrgWhiting** is set to "whiting", the domain+organization pair will not be added to the white organizations!

edit White-Org-List

Do Organization Blocking *(DoOrgBlocking)*

monitor

If activated, each sending IP address has its assigned organization looked up. Scoring is set with **sborgValencePB**, Testmode is set with **sbTestMode**.

Blacklisted Organizations, Domains and Hosts in SenderBase** *(blackSenderBase)*

If the organization, domain or hostname in the **SenderBase** IP description matches this Perl regular expression, the message will be considered spam.

Do Country Blocking *(DoCountryBlocking)*

monitor

If activated, each sending IP address has its assigned country looked up.

Blocked Country Codes** *(CountryCodeBlockedRe)*

CN|KR|RU|JP|TR|TH|PL|LT|CL|RO|UA|GR|HU|SA|IN|GB|IE|PT|MD|PE|CZ|TW|BR|C

Messages from IP's based in these countries will be blocked. For example:
CN|KR|RU|JP|TR|TH|PL|LT|CL|RO|UA|GR|HU|SA|IN|IE|PT|MD|PE|CZ|TW|BR|CL. "all" will block all foreign countrycodes which are not in 'Suspicious Country Codes' or 'Ignore Country Codes'. See: **English country names and code elements**.

Do Country Code Scoring *(DoSenderBase)*

score

If activated, each sending IP address has its assigned country looked up.

Ignore Countries* *(NoCountryCodeRe)*

US|CA|DE

Messages from IP's based in these countries will be ignored. For example: US|CA|DE

Suspicious Country Codes** *(CountryCodeRe)*

CN|KR|RU|JP|TR|TH|PL|LT|CL|RO|UA|GR|HU|SA|IN|IE|PT|MD|PE|CZ|TW|BR|CL|ID

Messages from IP's based in these countries will increase the MessageScore. For example:
CN|KR|RU|JP|TR|TH|PL|LT|CL|RO|UA|GR|HU|SA|IN|IE|PT|MD|PE|CZ|TW|BR|CL|ID|PH

Home Country Codes** *(MyCountryCodeRe)*

Put here your own country code(s) (for example: US). Messages from IP's based in these countries will decrease, messages from other countries will increase the MessageScore.

☑ **Score Foreign Countries** *(ScoreForeignCountries)*

Messages from foreign countries will increase the total messageScore using **sbfccValencePB**.

Country Cache Refresh Interval *(SBCacheExp)*

3

IP's in cache will be removed after this interval in days. 0 will disable the cache. show cache

PenaltyBox - Message and IP Scoring

Do PenaltyBox - IP History ***(DoPenalty)***

monitor/messageScoring

The PenaltyBox is a temporary position of low esteem awarded for a perceived misdeed. It scores IP's based on some events (**baValencePB** see penalty scores)and writes them into a BlackBox (PBBlack). If the score per specified time interval surpasses the threshold the message is rejected (and the IP is marked for blocking). They continue to get scored up to the Extreme Threshold.
These top performers can get a special treatment **PenaltyExtreme** when **DoPenaltyExtreme** is enabled. The WhiteBox (PBWhite) stores IP's which should not be put into the BlackBox (PBBlack). The WhiteBox is always enabled. If an address is in the whitelist or whitedomain, the IP goes into the WhiteBox. The WhiteBox is one of the sources Delaying/Greylisting uses to determine when delaying should not be done.
Entries in *Don't do penalties for these IP's* or *ISP/Secondary MX Servers* will prevent from penalties. Select 'monitor/messageScoring' to fill WhiteBox (PBWhite) and BlackBox (PBBlack). 'monitor/messageScoring' is also the right choice if you do not want to block IP's but rather score a message in 'Message Scoring Mode'.

Show BlackBox | Show White Box

Message Scoring Mode ***(DoPenaltyMessage)***

block

If this feature is selected, the total score for all checks during a message is used to determine if the email is Spam. If the combined score is greater than the **Low MessageLimit** (**PenaltyMessageLow**) and less than or equal the **High MessageLimit** (**PenaltyMessageLimit**) the message will not be blocked but tagged. If the combined score is greater than the **High MessageLimit** (**PenaltyMessageLimit**), the message will be blocked.

Message Scoring Mode for Local and Outgoing Mails ***(DoLocalPenaltyMessage)***

disabled

If this feature is selected, the total score for all checks during a local or outgoing message is used to determine if the email is Spam. If the combined score is greater than the **Local Low MessageLimit** (**LocalPenaltyMessageLow**) and less than or equal the **Local High MessageLimit** (**LocalPenaltyMessageLimit**) the message will not be blocked but tagged. If the combined score is greater than the **Local High MessageLimit** (**LocalPenaltyMessageLimit**), the message will be blocked.

☐ **Message Scoring on End** ***(MsgScoreOnEnd)***

ASSP will wait using the '**DoPenaltyMessage**' action, until all configured possible checks are finished. Use this, to force calculating a complete message score over all values, including all bonus values.

Low MessageLimit ***(PenaltyMessageLow)***

40

MessageMode will not block messages whose score exceeds this threshold during the message but will tag them. For example: 40

Low MessageLimit for Local and Outgoing Mails ***(LocalPenaltyMessageLow)***

40

MessageMode will not block local and outgoing messages whose score exceeds this threshold during the message but will tag them. For example: 40

High MessageLimit ***(PenaltyMessageLimit)***

50

MessageMode will block messages whose score exceeds this threshold during the message. For example: 50

High MessageLimit for Local and Outgoing Mails ***(LocalPenaltyMessageLimit)***

50

MessageMode will block local and outgoing messages whose score exceeds this threshold during the message. For example: 50

☑ **Add IP/Message Scoring Header** ***(AddScoringHeader)***

Adds a line to the email header "X-Assp-XXX-Score: ", where XXX may be IP, Message or both.

Don't do Profiling for these IP's* ***(noPB)***

Enter IP's that you don't want to be penalized. These IP's will also be automatically removed from BlackBox (PBBlack). For example: 127.0.0.1|172.16.

Don't do WhiteBox for these IP's* ***(noPBwhite)***

Enter IP's that you want to be penalized. These IP's will also be automatically removed from WhiteBox (PBWhite).

Expiration Time for WhiteBox Entries ***(WhiteExpiration)***

30

The WhiteBox (PBWhite) is always activated. The WhiteBox (PBWhite) is similar to the Whitelist - but it is not a whitelist: content-related checks like Bayesian, URIBL, Bomb will be done, IP-related checks will be skipped. WhiteBox (PBWhite) entries will expire after this specified number of days. For example: 30

Do Damping on Messagescore [0...99] ***(DoDamping)***

5

If **DoPenalty** and **DoPenaltyMessage** are set not to disabled and **DoDamping** is not set to 0, ASSP will slowdown the spammers traffic speed proportional to the current message score - because slowing down their speed will reduce spam everywhere.
The delay in seconds per receive/read cycle is calculated by the division [messagescore / **DoDamping**] . A recommended value is 5 (default is 0). In this case the delay for a message score of 50 would be 10 seconds.
Do not use this option, if you have a highly frequented system, because the spammers connections will stay possibly a long time on your system, and you system could possibly reach the sessions limit (**maxSMTPSessions**).
Damping is never done for: noprocessing, whitelisted, nodelay, ISP, redlisted, **noPB**, outgoing/releayed and contentonly addresses, IP's, messages.
Damping may not be done for forced checks, relay attemps, messages reaching maxerrors, **spamtrapaddresses** and if any block condition is

found - because ASSP will no more read from those connections and closes such connections immediately - but ASSP will try to keep the connection open for the calculated time, before it closes the connection.
Using this option or using a too low value (long delay) could possibly prevent ASSP from receiving spam messages, for example for spamlovers or **sendAllSpam** . Some Servers could give up sending data, because of too long delays.

Max time Used for Damping *(maxDampingTime)*

30
The maximum time in second, that is used for one damping cycle if **DoDamping** is not set to 0, even if the calculated value caused by **DoDamping** is higher. For example: 30

PenaltyBox Trap Addresses * *(spamtrapaddresses)*

put|your@penaltytrap.com|addresses|@here.org
Mail to any of these addresses will be blocked and the scoring value is added. Whitelist and **noPenaltyMakeTraps** will be ignored. Nothing will be stored in the Spam Collection, if these addresses are not checked for validity. TO: and CC: addresses will be also checked - BCC: addresses only, if '**removeForeignBCC**' is not set. If you want to use these addresses as permanent honeypott addresses (with collection), it is better to define them in **spamaddresses** and to enable **DoNotBlockCollect** . Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).

PenaltyTrap Reply *(PenaltyTrapPolite)*

550 5.1.1 User unknown: EMAILADDRESS
SMTP reply for invalid Users. Default: '550 5.1.1 User unknown: EMAILADDRESS'
The literal EMAILADDRESS (case sensitive) is replaced by the fully qualified SMTP recipient (e.g., thisuser@example.com).

Do Heavy Used Invalid Addresses as PenaltyBox Trap Addresses *(DoPenaltyMakeTraps)*

make traps, only collect them
If set to 'make traps, only collect them', the frequency of Invalid Addresses is stored, no other action taken. If set to 'do not make them but block' or 'make traps and block them', addresses in heavy use will act like **spamtrapaddresses** (PenaltyBox Trap Addresses). If **UseTrapToCollect** is also set they will work like **spamaddresses** and collect the mails.

Invalid Addresses Limit *(PenaltyMakeTraps)*

10
Minimum number of times an address must appear before it will be used as Trap. For example 10.

Exceptionlist for Traps* *(noPenaltyMakeTraps)*

Addresses which should not be used for traps. This list is also opponent to **spamtrapaddresses** . Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Invalid Addresses Refresh Interval *(PBTrapInterval)*

3
Addresses will be removed after this interval in days. For example 3. Show Invalid Addresses

☑ **Use IP Netblocks *(PenaltyUseNetblocks)***
Perform the IP address checks of the sending host based on the /24 subnet rather than on the specific IP.

Penalty Reply *(PenaltyError)*

If set SMTP reply for Penalty Deny. eg: '554 5.7.1 Error, send your mail to postmaster@LOCALDOMAIN to ensure delivery'. The literal LOCALDOMAIN will be replaced by the recipient domain. The literal LOCALUSER will be replaced by the recipient user part. For example:554 5.7.1 Mail appears to be unsolicited -- send error reports to postmaster@LOCALDOMAIN.

Penalty Interval *(PenaltyDuration)*

60
IP's will be kept in the BlackBox (PBBlack) if their score exceeds the Penalty Limit during this interval (minutes).

Penalty Limit *(PenaltyLimit)*

50
PB will block IP's whose score exceeds this threshold during the Penalty Interval.
Successful ASSP checks will increase the internal score per IP. For example: 50

Expiration Time *(PenaltyExpiration)*

360
Penalties will expire after this number of minutes after the first creation of the PenaltyBox record. If set to zero the Penalty BlackBox (PBBlack) will be deleted and started from scratch.

Clean Up PB Databases [s] *(CleanPBInterval)*

3
Delete outdated entries from blackbox (PBBlack) and whitebox (PBWhite) databases every this many hours.
Defaults to 3 hours.

PenaltyBox Extreme IP Profiling *(DoPenaltyExtreme)*

disabled
If set PBextreme will block IP's whose score meet or exceed Extreme Scoring Threshold. **DoPenaltyExtreme** blocks after the header is done, based on the IP's score from previous and current SMTP session

Enforce Early PenaltyBox Extreme IP Profiling *(DoPenaltyExtremeSMTP)*

disabled
If set PBextreme will block IP's whose score meet or exceed Extreme Scoring Threshold before DELAYING, based on the IP's score from previous SMTP sessions. This can be set independently from **DoPenaltyExtreme** above. Whitelist, Collecting, Testmode, CopySpam, Spam-Lover is ignored.

Don't do Extreme Profiling for these IP's* ***(noExtremePB)***

Enter IP's that you don't want to be extreme penalized. IP's in **noPB** are already included. For example: 127.0.0.1|172.16.

Don't do Extreme Profiling for Mails from any of these Addresses* ***(noExtremePBAddresses)***

Mails from any of these addresses will not be extreme profiled if **DoPenaltyExtremeSMTP** is not set. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).

Extreme Scoring Threshold ***(PenaltyExtreme)***

150

PBextreme will use this to determine candidates for special treatment. For example: 150.

☐ **Penalize Whitelisted** ***(ExtremeWL)***

Enable extreme penalties for whitelisted addresses.

☐ **Penalize NonProcessing** ***(ExtremeNP)***

Enable extreme penalties for addresses on the **noProcessing** list.

Expiration Time for Extreme Penalties ***(ExtremeExpiration)***

7

Extreme penalties will expire after this number of days. For example: 7

☐ **Do Export Penalty BlackBox Extreme** ***(DoExtremeExport)***

☐ **Append Export File** ***(DoExtremeExportAppend)***

Do not overwrite the export file but append to it.

Export BlackBox Extreme File Interval [s] ***(exportInterval)***

6

Exported Penalty Black Box Extreme File every this hours.
Defaults to 6 hours.

Exported BlackBox Extreme File ***(exportExtremeBlack)***

file:files/exportedextreme.txt Edit file

IP's in Penalty BlackBox (PBBlack) which surpassed the extreme level will be regularly stored into this file. This file may be used for setting the firewall or similar applications. The file can be downloaded via the STATS-interface " **webStatPort** "! The download URL, used by your firewall, should look like: http://assp.domain.local:55553/extremeblack .

☐ **Do Not Score IP's in Redlisted Messages** ***(DoNotPenalizeRed)***

IP's matching Red Regex or Redlist will not collect scoring values from PenaltyBox.

☐ **Do Not Score IP's From Bounce/Null-Senders** ***(DoNotPenalizeNull)***

IP's matching **BounceSenders** will not be IP-penalized.

Bad SMTP Authentication, default=60 + ***(autValencePB)***

60

Message/IP scoring

This option and all other *ValencePB options with a "+" at the end of the description, accepts a second comma or pipe separated value like: "20,10" .
In this case the first value is used for message scoring and the second value is used for IP scoring.
If only the first value is defined, this value is used for both scoring mechanism.
If a *ValencePB option is related to any feature which allowes the usage of weighted penalties, the message scoring value is used to calculate the weighted penalty and the result is used calculating (result * ipscorevalence / messagescorevalence) for IP scoring.

Bad Attachment, default=20 + ***(baValencePB)***

20

Message/IP scoring

Backscatter detection, default=10 + ***(backsctrValencePB)***

10

Message scoring

Bayesian, default=49 + ***(baysValencePB)***

49

Message/IP scoring

Bayesian for Local Messages, default=55 + *(bayslocalValencePB)*

55

Message/IP scoring

Bayesian HAM Bonus, default=0 + *(bayshamValencePB)*

0

Message/IP scoring bonus (zero or negative value only)

Hidden-Makov-Model, default=49 + *(HMMValencePB)*

49

Message/IP scoring

Hidden-Makov-Model for Local Messages, default=55 + *(HMMlocalValencePB)*

55

Message/IP scoring

Hidden-Makov-Model HAM Bonus, default=0 + *(HMMhamValencePB)*

0

Message/IP scoring bonus (zero or negative value only)

Blacklisted Domain, default=20 + *(blValencePB)*

20

Message/IP scoring

Bomb Suspicious - scoring only, default=10 + *(bombSuspiciousValencePB)*

10

Message scoring

Bomb Expression, default=20 + *(bombValencePB)*

20

Message/IP scoring

Bomb Black Expression, default=20 + *(blackValencePB)*

20

Message/IP scoring

Domain Key Verification failed, default=15 + *(dkimValencePB)*

15

Message/IP scoring

Domain Key Verification OK, default=0 *(dkimOkValencePB)*

0

Message Scoring Bonus

Empty Recipients, default=5 + *(erValencePB)*

5

Message/IP scoring

Early Talker Scoring, default=25 + *(etValencePB)*

25

Message/IP scoring for clients who talk before server's greeting is sent. A value of zero will disable this check - otherwise assp scores the IP and droppes the connection.

Forged HELO, default=150 + *(fhValencePB)*

150

Message/IP scoring

Suspicious HELO: IP in HELO, default=39 + *(fiphValencePB)*

39

Message/IP scoring

Suspicious HELO: IP in HELO mismatch, default=60 + *(fiphmValencePB)*

60

Message/IP scoring

Invalid Local Sender, default=20 + *(flValencePB)*

20

Message/IP scoring

Spoofed Local Sender, default=20 + *(slValencePB)*

20

Message/IP scoring

Blacklisted/Good HELO, default=20 + *(hlValencePB)*

20

Message/IP scoring

Internal Only Address, default=25 + *(iaValencePB)*

25

Message/IP scoring

Domain Changing IP Frequency, default=150 + *(idValencePB)*

150

Message/IP scoring

IP Frequency, default=150 + *(ifValencePB)*

150

Message/IP scoring

Timeout Score *(idleValencePB)*

0

For IP scoring with **smtpIdleTimeout**.

IP Parallel Sessions, default=5 + *(iplValencePB)*

5

Message/IP scoring

Invalid HELO, default=10 + *(ihValencePB)*

10

Message/IP scoring

Invalid Recipient, default=10 + *(irValencePB)*

10

Message/IP scoring

Subject Frequency, default=150 + *(isValencePB)*

150

Message/IP scoring

Duplicate Recipient, default=10 + *(mdrValencePB)*

10

Message/IP scoring

Missing Message-ID, default=10 + *(midmValencePB)*

10

Message/IP scoring

Suspicious Message-ID, default=10 + *(midsValencePB)*

10

Message/IP scoring

Invalid Message-ID, default=10 + *(midiValencePB)*

10

Message/IP scoring

Invalid FBMTV check, default=25 + *(fbmtvValencePB)*

25

Message/IP scoring

Invalid BATV check, default=25 + *(batvValencePB)*

25

Message/IP scoring

Max Errors Exceeded, default=10 + *(meValencePB)*

10

Message/IP scoring

Message Scoring Limit Exceeded, default=10 + *(msValencePB)*

10

IP scoring

Missing MX, default=10 + *(mxValencePB)*

10

Message/IP scoring

Missing MX & A Record, default=15 + *(mxaValencePB)*

15
Message/IP scoring

No From Score, default=50 + *(nofromValencePB)*
50
For Message/IP scoring in **DoNoFrom**.

Extreme Bad IP History, TotalScore larger than PenaltyExtreme, default=25 *(pbeValencePB)*
25
Message Scoring

Bad IP History, TotalScore larger than PenaltyLimit, default=15 *(pbValencePB)*
15
Message Scoring

Good IP History (IP in PB WhiteBox), default=-15 *(pbwValencePB)*
-15
Message Scoring Bonus

GRIP value (+ if > 0.7,- if < 0.3), default=5 *(gripValencePB)*
5
Message scoring

Message OK, default=-25 *(okValencePB)*
-25
IP Bonus

Missing PTR Record, default=10 + *(ptmValencePB)*
10
Message/IP scoring

Invalid PTR Record, default=15 + *(ptiValencePB)*
15
Message/IP scoring

DNSBL Neutral, default=35 + *(rblnValencePB)*
35
Message/IP scoring

DNSBL Failed, default=100 + *(rblValencePB)*
100
Message/IP scoring

Failed Relay Attempt, default=10 + *(rlValencePB)*
10
Message/IP scoring

Spam Collect Address, default=25 *(saValencePB)*
25
IP scoring

Script Expression, default=25 + *(scriptValencePB)*
25
Message/IP scoring

No Organization and No CountryCode, default=10 + *(sbnValencePB)*
10
For Message/IP scoring in DoOrgBlocking/DoCountryBlocking

White Organizations Score, default=-25 *(sworgValencePB)*
-25
Bonus for Message/IP scoring in DoOrgWhiting

Suspicious Country Code, default=10 *(sbsccValencePB)*
10
Message scoring

Blocked Country Code Score, default=25 + *(bccValencePB)*
25
For Message/IP scoring in PenaltyBox (**DoPenalty**)

Foreign Country Code Score, default=10 + *(sbfccValencePB)*

10
message scoring in PenaltyBox (**DoPenaltyMessage**)

Home Country Code Score, default=-10 + *(sbhccValencePB)*

-10
Bonus for Message/IP Scoring in PenaltyBox (**DoPenalty**)

Blocked Organizations Score, default=25 + *(sborgValencePB)*

25
For Message/IP scoring in PenaltyBox (**DoPenalty**)

SPF Pass Score, default=-10 *(spfpValencePB)*

-10
Bonus for Message/IP scoring with SPF

SPF Neutral, default=5 + *(spfnValencePB)*

5
Message/IP scoring

SPF Softfailed, default=5 + *(spfsValencePB)*

5
Message/IP scoring

SPF None, default=0 + *(spfnonValencePB)*

0
Message/IP scoring

SPF Unknown, default=0 + *(spfuValencePB)*

0
Message/IP scoring

SPF Error, default=5 + *(spfeValencePB)*

5
Message/IP scoring

SPF Failed, default=10 + *(spfValencePB)*

10
Message/IP scoring

SRS Validate Bounce Failed Score, default=10 + *(srsValencePB)*

10
For Message/IP scoring in **SRSValidateBounce**

Penalty Trap Address, default=50 + *(stValencePB)*

50
For Message/IP scoring

OK, Is a SSL/TLS connection, default=-10 + *(tlsValencePB)*

-10
Message Scoring/IP scoring Bonus for SSL/TLS connections

URIBL Neutral, default=20 + *(uriblnValencePB)*

20
Message/IP scoring

URIBL Failed, default=25 + *(uriblValencePB)*

25
Message/IP scoring

Virus suspicious, default=25 *(vsValencePB)*

25
Message scoring

Virus detected, default=50 + *(vdValencePB)*

50
Message/IP scoring

TestRe Valence, default=20 + *(teValencePB)*

20
Valence for testing testRe

Notes On Penalty Box
Notes

Delaying - Greylisting

☑ **Enable Delaying/Greylisting** *(EnableDelaying)*

Enable Greylisting (also called Delaying) as described at **Greylisting-whitepaper**.
It's a new method of blocking significant amounts of spam at the mailserver level, but without resorting to heavyweight statistical analysis or other heuristical approaches.

☐ **Whitelisted Greylisting** *(DelayWL)*

Enable Greylisting for whitelisted mails. This also enables Geylisting for SPF-Cache-OK listed IP's and mails from white organizations, which are normally not greylisted.

☐ **NoProcessing Greylisting** *(DelayNP)*

Enable Greylisting for noprocessing mails.

☐ **Spam-Lovers Greylisting** *(DelaySL)*

Enable Greylisting for Spam-Lovers.

☑ **Add X-Assp-Delayed Header** *(DelayAddHeader)*

Add X-Assp-Delayed header to header of all delayed or whitelisted mails.

Embargo Time *(DelayEmbargoTime)*

5

Enter the number of minutes for which delivery, related with new 'triplet' (IP address of the sending host + mail from + rcpt to), is refused with a temporary failure. Default is 5 minutes.

Wait Time *(DelayWaitTime)*

28

Enter the number of hours to wait for delivery attempts related with recognized 'triplet'; delivery is accepted immediately and the 'tuplet' (IP address of the sending host + sender's domain) is safelisted. Default is 28 hours.

Expiry Time *(DelayExpiryTime)*

36

Enter the number of days for which whitelisted 'tuplet' is considered valid. Default is 36 days.

☑ **Use IP Netblocks** *(DelayUseNetblocks)*

Perform the IP address checks of the sending host based on the /24 subnet it is at rather than the specific IP.
This feature may be useful for legitimate mail systems that shuffle messages among SMTP clients between retransmissions.

☑ **Normalize VERP Addresses** *(DelayNormalizeVERPs)*

Some mailing lists (such as Ezmlm) try to track bounces to individual mails, rather than just individual recipients, which creates a variation on the VERP method where each email has its own unique envelope sender. Since the automatic whitelisting (called savelisting to make a difference to the standard whitelisting) that is built into Greylisting depends on the envelope addresses for subsequent mails being the same, the greylisting filter will attempt to normalize the unique sender addresses, when this option is checked.

☐ **Add myName to Triplets** *(DelayWithMyName)*

If set, **myName** is added to every delay triplet (not to tuplets). This is useful and recommended, if you are using more than one ASSP host with shared databases for **delaydb**. This option makes the triplets unique to every ASSP host, because it is allowed for SMTP-hosts, to request a backup MX immediately after the primary MX, without waiting 5 minutes (**DelayEmbargoTime**) between the two requests.

☑ **Use MD5 for DelayDB** *(DelayMD5)*

Message-Digest algorithm 5 is a cryptographic hash function and adds some level of security to the delay database. Must be set to off if you want to list the database with DelayShowDB/DelayShowDBwhite. This requires an installed **Digest::MD5** module in PERL.

Show Delay/Greylisting Database *(DelayShowDB)*

file:delaydb [Show file]

The directory/file with the delay database file. If you change the filename in section Filepath (**delaydb**) you must change it here too.

Show Delay/Greylisting Save Database *(DelayShowDBwhite)*

file:delaydb.white [Show file]

The directory/file with the save delay database file. If you change the filename in section Filepath (**delaydb**) you must change it here too.

☑ **Expire Spamming Safelisted Tuplets** *(DelayExpireOnSpam)*

If a safelisted 'tuplet' is ever associated with spam, viruses, failed rbl, spf etc, it is deleted from the safelist.
This renews the temporary embargo for subsequent mail involving the tuplet.

Clean Up Delaying Database [s] *(CleanDelayDBInterval)*

10800

Delete outdated entries from triplets and safelisted tuplets databases every this many seconds.
Defaults to 3 hour.

Don't Delay these IPs* *(noDelay)*

file:files/nodelay.txt [Edit file]

Enter IP addresses that you don't want to be delayed, separated by pipes (|). There are misbehaving MTAs that will not be able to get a

legitimate email through a Greylisting server because they do not try again later. An INCOMPLETE list of such mailers is available at **cvs.puremagic.com/viewcvs/Greylisting/schema/whitelist_ip.txt**.
When using mentioned list remember to add trailing dots in IP addresses which specify subnets (eg. 192.168 -> 192.168.).
For example: 127.0.0.1|172.16..
To define IP's only for specific email addresses or domains (recipients) you must use the file:... option
An entry (line) may look as follows:
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

It is possible to define a predefined group on any or both sides of the '=>' separator, like:
[ipgroup]=>[usergroup]|user@mydomain

NOTICE: the following combination of two entries, will lead in to a user/domain based matching - the global entry will be ignored!
145.146.0.0/16 # comment
145.146.0.0/16=>*@local.domain|user@mydomain|user2@*.mydomain # comment

Do not Delay these Addresses* ***(noDelayAddresses)***

Enter senders and/or recipient email addresses that you don't want to be delayed, separated by pipes (|). You can list specific addresses (user@anydomain.com), addresses at any domain (user), or entire domains (@anydomain.com). Wildcards are supported (fribo*@domain.com). (|).
For example: fribo@anydomain.com|jhanna|@sillyguys.org or place them in a plain ASCII file one address per line:file:files/nodelayuser.txt.
Groups definitions are also allowed to be used.

Reply Code to Refuse Delayed Messages ***(DelayError)***

451 4.7.1 Please try again later

SMTP reply code to refuse delayed messages. Default: 451 4.7.1 Please try again later

Notes On Delaying

Notes

Validate SPF, DMARC and SRS

Enable SPF Validation *(ValidateSPF)*

score

Enable Sender Policy Framework Validation as described at **openspf** and Domain-based Message Authentication, Reporting & Conformance - described in **DMARC** (DMARC requires also **DoDKIM** to be enabled).
This requires an installed **Mail::SPF** module in PERL. Testmode is set with **spfTestMode**, scoring is set with **spfValencePB**. If you need more information about the syntax of SPF records, visit **SPF_Record_Syntax**.

☑ **Do SPF Version 2 Validation** *(SPF2)*

Enable Sender Policy Framework Validation Version 2. Default is ON.
This requires an installed **Mail::SPF** object-oriented Perl module that supersedes the old Mail::SPF::Query module.
It is highly recommended to disable the load of Mail::SPF::Query by turning OFF **useMailSPFQuery** , if this option is set to ON.

☐ **Whitelisted SPF Validation** *(SPFWL)*

Enable Sender Policy Framework Validation for whitelisted users also.

☐ **noProcessing SPF Validation** *(SPFNP)*

Enable Sender Policy Framework Validation for nonprocessed messages also.

☐ **Local and outgoing mail SPF Validation** *(SPFLocal)*

Enable Sender Policy Framework Validation for local and outgoing messages also. Don't forget to configure your DNS-server for SPF and/or to configure **SPFoverride** / **SPFfallback** / **SPFlocalRecord**, if you enable this option.

☑ **Enable SPF Background Check** *(enableSPFbackground)*

SPF background checks are initiated by some features (for example **DoDomainIP**) to fillup the SPFCache. The collected results are later used to prevent blocking good mails.

☑ **Add Received-SPF Header** *(AddSPFHeader)*

Add Received-SPF header to header of all mails processed by SPF.

SPF Failed Reply *(SPFError)*

554 5.7.1 failed SPF: SPFRESULT

SMTP reply for SPF failed messages. Default: '554 5.7.1 failed SPF: SPFRESULT'
The literal SPFRESULT (case sensitive) is replaced by the actual result.

Skip SPF Processing* *(noSPFRe)*

Put anything here to identify these messages in mailfrom or header

Override Domains* *(SPFoverride)*

Set override to define SPF records for domains that do publish (or not) but which you want to override anyway. If you specify only domains the Local SPF Record (**SPFlocalRecord**) below will be used as default. Wildcards are supported. For example: abc.com=>v=spf1 a/24 mx/24 ptr -all|cello.ch=>v=spf1 ip4:213.46.243.0/26 ~all|abc.com|*.def.com .
To generate a SPF record for a domain:
- go to **http://www.senderbase.org**
- lookup the domain information in "Look up your network"
- right beside "Addresses in domain used to send email" click on export, and export the list in to plain text
- copy and past the list in to an editor and generate a comma separated IP list
- go to an online SPF record generator - for example: **http://www.royhochstenbach.com/projects/spfgenerator** and generate the SPF record
- put "domain=>SPF-record" in any of **SPFoverride** or SPFfallback
- define the policy as strict as possible

Fallback Domains* *(SPFfallback)*

Set fallback to define "pretend" SPF records for domains that don't publish them yet. If you specify only domains the Local SPF Record (**SPFlocalRecord**) below will be used as default. Wildcards are supported. For example: abc.com=>v=spf1 a/24 mx/24 ptr -all|cello.ch=>v=spf1 ip4:213.46.243.0/26 ~all|abc.com|*.def.com

Local SPF Policy *(LocalPolicySPF)*

v=spf1 a/24 mx/24 ptr ~all

If the sending domain does not publish its own SPF Records this will be used.
The default is v=spf1 a/24 mx/24 ptr ~all
This option applies to Mail::SPF::Query module only.

Fallback/Override SPF Record *(SPFlocalRecord)*

v=spf1 a/24 mx/24 ptr -all

Used in Fallback/Override Domains
The default is v=spf1 a/24 mx/24 ptr -all

Strict SPF Processing Regex* *(strictSPFRe)*

file:files/strictspf.txt Edit file

Softfail/Neutral will be failed for these sending addresses. Put anything here to identify the addresses

Block SPF Processing Regex* ***(blockstrictSPFRe)***

@ebay.com|@paypal.com

All failed messages will be blocked for these sending addresses. Put anything here to identify the addresses.

☐ **Additional SPF Check on the Header from** ***(DoSPFinHeader)***

Do an additional SPF check on the header from: address if it is in **blockstrictSPFRe** *** this check breakes RFC rules ***.

☐ **Fail SPF Softfail Validations** ***(SPFsoftfail)***

Intentionally fail SPF softfail status responses. The possible results of a query are:
pass:The client IP address is an authorized mailer for the sender. The mail should be accepted subject to local policy regarding the sender.
fail:The client IP address is not an authorized mailer, and the sender wants you to reject the transaction for fear of forgery.
softfail:The client IP address is not an authorized mailer, but the sender prefers that you accept the transaction because it isn't absolutely sure all its users are mailing through approved servers. The softfail status is often used during initial deployment of SPF records by a domain.
neutral:The sender makes no assertion about the status of the client IP.
none:There is no SPF record for this domain.
permerror & temperror:The DNS lookup encountered an error during processing.
unknown:The domain has a configuration error in the published data or defines a mechanism that this library does not understand.

☐ **Fail SPF Neutral Validations** ***(SPFneutral)***

Intentionally fail SPF neutral status responses

☐ **Fail SPF Error Responses** ***(SPFqueryerror)***

Intentionally fail SPF 'error' status responses

☐ **Fail SPF None and Unknown Responses** ***(SPFnone)***

Intentionally fail SPF 'none' status responses

☐ **Fail SPF Unknown Responses** ***(SPFunknown)***

Intentionally fail SPF 'unknown' status responses

SPF Cache Refresh Interval ***(SPFCacheInterval)***

7

SPF records in cache will be removed after this interval in days. 0 will disable the cache. Show SPF Cache

☐ **Enable SPF/DNS/Whois/Senderbase Debug output to ASSP Logfile** ***(DebugSPF)***

Enables verbose debugging of SPF/DNS/Whois/Senderbase queries within the related modules.

Notes On SPF

Notes

☑ **Enable DMARC Check** ***(DoDMARC)***

If enabled and **ValidateSPF** and **DoDKIM** are enabled and the sending domain has published a DMARC-record/policy, assp will act on the mail according to the senders DMARC-policy using the results of the SPF and DKIM check. It is save to leave this feature ON, it will not produce false positives!
If you have published a DMARC-record and you want to collect statisical data, look at **dmarcian.com**

Don't Check DMARC for these Addresses/Domains* ***(noDMARCDomain)***

Put any sender domain (or address) in to this list, for which you want to disable the DMARC check - for example if an invalid DMARC record is published.
Use '**noDMARCReportDomain**' if you only want to disable DMARC reports.
Accepts entire domains (@example.com) (specific addresses (user@example.com) and user parts (user) are accepted, but not usefull!).
Wildcards are supported (@*example.com or @*.example.com).

From Address for DMARC Reports ***(DMARCReportFrom)***

The email address to be used as FROM: address to send **DMARC** reports. If blank, no DMARC reports will be sent! If only the user name is defined, assp will add the domain name that belongs to the report.

Don't send DMARC reports to these Addresses/Domains* ***(noDMARCReportDomain)***

Put any DMARC report recipient domain or address (ruf/rua) in to this list - for example if DMARC reports could be never delivered for any reason.
Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com). Wildcards are supported (fribo*@example.com).

☐ **Enable Sender Rewriting Scheme** ***(EnableSRS)***

Enable Sender Rewriting Scheme as described at **www.openspf.org/SRS**.
This requires an installed **Mail::SRS** module in PERL.
You should use SRS if your message handling system forwards email for domains with published spf records and there SPF record not includes your MX.
NOTICE: In case your local users are forwarding mails (e.g. from external domains) to external domains (external mail accounts) and these foreign domains bounces back (e.g. out_of_office / vacation), your MTA (**smtpDestination**) will possibly get mails from external domains to be delivered to external domains!
Note that you have to setup the outgoing path (Relay Host and Port) to let ASSP see and rewrite your outgoing traffic.
Testmode is set with **srsTestMode**.

Alias Domain ***(SRSAliasDomain)***

thisdomain.com
SPF requires the SMTP client IP to match the envelope sender (return-path). When a message is forwarded through an intermediate server, that intermediate server may need to rewrite the return-path to remain SPF compliant.
For example: thisdomain.com

Secret Key *(SRSSecretKey)*

A key for the cryptographic algorithms -- Must be at least 5 characters long.

Maximum Timestamp Age *(SRSTimestampMaxAge)*

2
Enter the maximum number of days for which a timestamp is considered valid. Default is 2 days. After this number of days a SRS bounce is no longer valid!

Hash Length *(SRSHashLength)*

6
The number of bytes of base64 encoded data to use for the cryptographic hash.
More is better, but makes for longer addresses which might exceed the 64 character length suggested by RFC2821.
This defaults to 6, which gives 6 x 6 = 36 bits of cryptographic information, which means that a spammer will have to make 2^36 attempts to guarantee forging a SRS address.

Enable Bounce Recipient Validation *(SRSValidateBounce)*

disabled
Bounce messages that fail reverse SRS validation (but not a valid SMTP probe) will receive a 554 5.7.5 [Bounce address not SRS signed] SMTP error code.
Testmode is set with **srsTestMode**, scoring is set with **srsValencePB**.

Don't Rewrite These Addresses* *(SRSno)*

Don't rewrite addresses when messages come from these addresses. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).
For example: fribo@thisdomain.com|jhanna|@sillyguys.org

Don't Validate Bounces From these IPs* *(noSRS)*

Enter IP addresses that you don't want to validate bounces from, separated by pipes (|). For example: 127.0.0.1|172.16..

Notes On SRS
Notes

DNSBL - RBL Validation

Enable DNS Blacklist Validation ***(ValidateRBL)***

block

This requires an installed **Net::DNS** module in PERL.

☐ **Early DNSBL Cache Blocking** ***(ForceRBLCache)***

If set, ASSP will use cached DNSBL hits to block messages before other tests. **testmode** will override this. **spamlover settings** will be ignored.

Don't do DNSBL for these IPs* ***(noRBL)***

Enter IP addresses that you don't want to be DNSBL validated, separated by pipes (|). For example: 127.0.0.1|172.16..

☐ **Whitelisted DNSBL Validation** ***(RBLWL)***

Enable DNSBL for whitelisted users also

☑ **Add X-Assp-DNSBL Header** ***(AddRBLHeader)***

Add X-Assp-DNSBL header to messages with positive reply from DNSBL.

DNSBL Failed Reply ***(RBLError)***

554 5.7.1 DNS Blacklisted by RBLLISTED

SMTP reply for DNSBL failed messages. Default: '554 5.7.1 DNS Blacklisted by RBLLISTED'
The literal RBLLISTED (case sensitive) is replaced by the actual service providers(s).

RBL Service Providers* ***(RBLServiceProvider)***

file:files/dnsbls.txt Edit file

Names of DNSBLs to use separated by "|". You may set for every provider a weight like zen.spamhaus.org=>50|bl.spamcop.net=>25.
Defaults are:
zen.spamhaus.org=>1|bl.spamcop.net=>1|psbl.surriel.com=>2|ix.dnsbl.manitu.net=>2|
l2.apews.org=>3|combined.njabl.org=>1|safe.dnsbl.sorbs.net=>1|dnsbl-1.uceprotect.net=>2|
dnsbl-2.uceprotect.net=>2|dnsbl-3.uceprotect.net=>2|blackholes.five-ten-sg.com=>3".
DNSBL providers can get a "weight" like bl.spamcop.net=>1.
The value of the weight can be set directly like=>45 or as a divisor of **RBLmaxweight**. Low numbers < 6 are divisors . So if **RBLmaxweight** = 50 (default) bl.spamcop.net=>50 would be the same as bl.spamcop.net=>1, bl.spamcop.net=>2 would be the same as bl.spamcop.net=>25.
If the sum of weights surpasses **RBLmaxweight**, the DNSBL check fails. If not, the DNSBL check is scored as "neutral" even with **RBLmaxhits** reached. Setting **Showmaxreplies** will allow ALL replies to contribute to the total weight regardless of RBLmaxhits.
Some RBL Service Providers, like blackholes.five-ten-sg.com, provides different return codes in a single DNS-zone: like 127.a.b.c - where a,b,c are used to identify a weight or type (or what ever) of the returned entry. If you want to care about special return codes, or if you want to use different weights for different return codes, you should use the following enhanced entry syntax:

RBL-Service-Provider=>result-to-watch=>weight (like:)
blackholes.five-ten-sg.com=>127.0.0.2=>3
blackholes.five-ten-sg.com=>127.0.0.5=>4
blackholes.five-ten-sg.com=>127.0.?.*=>5

You can see, the wildcards * (multiple character) and ? (single character) are possible to use in the second parameter. Never mix the three possible syntax types for the same RBL Service Provider. A search for a match inside such a definition is done in reverse ASCII order, so the wildcards are used as last.
Some RBL Service Providers, provides different return codes using a bitmask in any part of the reply. To define weights for bitmasks, place a single 'M' in front of the mask number, like

sp.com=>127.0.0.M2=>25
sp.com=>127.0.0.M4=>41
sp.com=>127.0.M1.5=>56
sp.com=>127.0.M64.*=>11
sp.com=>127.0.0.2=>22
sp.com=>127.0.*.*=>1

Valid bitmasks are 1,2,4,8,16,32,64 and 128. The resulting weight will be the weight sum of all matching bitmasks (if no full qualified definition is found). For example: a return code of 127.0.0.6 for sp.com will result in a weight of 66 (25+41), a reply of 127.0.0.2 will result in 22
Because each single bitmask indicates a set of 128 numbers you should prevent the usage of something like 127.0.M16.M1 - this will lead in to a set of (128*128) 16384 addresses, which is really too much!
For the same service provider, first define all bitmask definitions, after that all full qualified definitions and than all definitions with wildcards, like in the example above! If your definition order is wrong, the resulting weights will be unexpected!

Maximum Replies ***(RBLmaxreplies)***

7

A reply is affirmative or negative reply from a DNSBL.
The DNSBL module will wait for this number of replies (negative or positive) from the DNSBLs listed under Service Provider for up to the Maximum Time(**RBLmaxtime**).
This number should be equal to or less than the number of DNSBL Service Providers listed to allow for randomly unavailable DNSBLs.

Maximum Hits ***(RBLmaxhits)***

2

A hit is an affirmative response from a DNSBL.
The DNSBL module will check all of the DNSBLs listed under Service Provider. If the number of hits is greater or equal Maximum Hits, the email is flagged **Failed**.
If the number of hits is greater 0 and less Maximum Hits, the email is flagged **Neutral**

RBL Maximum Weight ***(RBLmaxweight)***

50
A weight is a number representing the trust we put into a DNSBL.
The DNSBL module will check all of the DNSBLs listed under Service Provider. If the total of weights is greater or equal Maximum Weight, the email is flagged **Failed**.
If the total of weights is greater 0 and less Maximum Weight, the email is flagged **Neutral**

Maximum Time *(RBLmaxtime)*

15
This sets the maximum time in seconds to spend on each message performing DNSBL checks. Default is 15.

Socket Timeout *(RBLsocktime)*

1
This sets the DNSBL socket read timeout in seconds.

DNSBL Expiration Time *(RBLCacheExp)*

24
IP's in cache will be removed after this interval in hours. 0 will disable the cache. Show DNSBL Cache

Notes On DNSBL
Notes

URIBL and Obfuscation Detection

Enable URI Blocklist Validation *(ValidateURIBL)*

block

Enable URI Blocklist. Messages that fail URIBL validation will receive **URIBLError** SMTP error code. This requires an installed **Net::DNS** module and an installed **Email::MIME** module in PERL.
0 = disabled, 1 = block, 2 = monitor, 3 = messagescore .

☐ **Do URI Blocklist Validation for Whitelisted** *(URIBLWL)*

URIBL check is done ignoring all spamlovers and testmodes!

☐ **Do URI Blocklist Validation for NoProcessing** *(URIBLNP)*

URIBL check is done ignoring all spamlovers and testmodes!

☐ **Do URI Blocklist Validation for Local Mails** *(URIBLLocal)*

☑ **Do URI Blocklist Validation for ISP/Secondary** *(URIBLISP)*

URIBL Service Providers* *(URIBLServiceProvider)*

file:files/uribls.txt Edit file

Domain Names of URIBLs to use separated by "|". You may set for every provider a weight like multi.surbl.org=>50|black.uribl.com=>25.
The value of the weight can be set directly like=>45 or as a divisor of **URIBLmaxweight** . Low numbers < 6 are divisors . So if **URIBLmaxweight** = 50 (default) multi.surbl.org=>50 would be the same as multi.surbl.org=>1, multi.surbl.org=>2 would be the same as multi.surbl.org=>25.
If the sum of weights surpasses **URIBLmaxweight**, the URIBL check fails. If not, the URIBL check is scored as "neutral" even with **URIBLmaxhits** reached. Setting **Showmaxreplies** will allow ALL replies to contribute to the total weight regardless of URIBLmaxhits.
Some URIBL Service Providers, like multi.surbl.org and black.uribl.com , provides different return codes in a single DNS-zone: like 127.a.b.c - where a,b,c are used to identify a weight or type (or what ever) of the returned entry. If you want to care about special return codes, or if you want to use different weights for different return codes, you should use the following enhanced entry syntax:

URIBL-Service-Provider=>result-to-watch=>weight (like:)
multi.surbl.org=>127.0.0.2=>2
multi.surbl.org=>127.0.0.4=>3
multi.surbl.org=>127.0.0.?=>4
multi.surbl.org=>127.0.0.*=>5

You can see, the wildcards * (multiple character) and ? (single character) are possible to use in the second parameter. Never mix the three possible syntax types for the same URIBL Service Provider. A search for a match inside such a definition is done in reverse ASCII order, so the wildcards are used as last.
Some URIBL Service Providers, provides different return codes using a bitmask in any part of the reply. To define weights for bitmasks, place a single 'M' in front of the mask number, like

sp.com=>127.0.0.M2=>25
sp.com=>127.0.0.M4=>41
sp.com=>127.0.M1.5=>56
sp.com=>127.0.M64.*=>11
sp.com=>127.0.0.2=>22
sp.com=>127.0.*.*=>1

Valid bitmasks are 1,2,4,8,16,32,64 and 128. The resulting weight will be the weight sum of all matching bitmasks (if no full qualified definition is found). For example: a return code of 127.0.0.6 for sp.com will result in a weight of 66 (25+41), a reply of 127.0.0.2 will result in 22
Because each single bitmask indicates a set of 128 numbers you should prevent the usage of something like 127.0.M16.M1 - this will lead in to a set of (128*128) 16384 addresses, which is really too much!
For the same service provider, first define all bitmask definitions, after that all full qualified definitions and than all definitions with wildcards, like in the example above! If your definition order is wrong, the resulting weights will be unexpected! Default is: multi.surbl.org|black.uribl.com

URIBL Country Code TLDs* *(URIBLCCTLDS)*

file:files/URIBLCCTLDS.txt Edit file

List of **two level country code TLDs** and **three level country code TLDs** used to determine the **base** domain of the uri. Two level TLDs will be checked on third level, third level TLDs will be checked on fourth level. Any not listed domain will be checked in level two.

Maximum URIs *(URIBLmaxuris)*

0

More than this number of URIs in the body will increase spam probability. Enter 0 to disable feature.

Maximum Unique Domain URIs *(URIBLmaxdomains)*

0

More than this number of unique domain URIs in the body will increase spam probability. Enter 0 to disable feature.

☑ **Disallow Obfuscated URIs** *(URIBLNoObfuscated)*

When enabled, messages with obfuscated URIs of types [integer/octal/hex IP, other things!] in the body will get increased spam probability and if weights are used, the double weight will be used. If a very strong obfuscated IP is detected (like: 0x9A3F0800CEBF9E37 or 0xCE.191.0236.0x37), URIBL will fail!

☐ **Check for 'DOT' in URI** *(URIBLcheckDOTinURI)*

When enabled, assp will also check for the used word 'DOT' instead of a '.' in URI's like 'example**dot**com or example**!d o-t_**com' .
Enable this feature only, if you don't expect any problems in your national language (using 'dot' + a toplevel domain in any words).

Maximum Replies *(URIBLmaxreplies)*

2
A reply is affirmative or negative reply from a URIBL.
The URIBL module will wait for this number of replies (negative or positive) from the URIBLs listed under Service Provider for up to the Maximum Time below. This number should be equal to or less than the number of URIBL Service Providers listed to allow for randomly unavailable URIBLs.

Maximum Hits *(URIBLmaxhits)*

1
A hit is an affirmative response from a URIBL.
The URIBL module will check all of the URIBLs listed under Service Provider,
and flag the email with a URIBL failure flag if more than this number of URIBLs return a positive blacklisted response.
This number should be less than or equal to Maximum Replies above and greater than 0. If the number of hits is greater or equal Maximum Hits, the email is flagged **failed** in every case! If the number of hits is greater 0 and less Maximum Hits, the email is flagged **neutral**.
This behavior could be changed to your needs by using weighted values for the **URIBLServiceProvider** .

URIBL Maximum Weight *(URIBLmaxweight)*

50
A weight is a number representing the trust we put into a URIBL.
The URIBL module will check all of the URIBLs listed under **URIBLServiceProvider** for every URI found in an email. If the total of weights for a URI is greater or equal this Maximum Weight, the email is flagged **Failed**.
If the total of weights is greater 0 and less Maximum Weight, the email is flagged **Neutral** . If not defined or set to zero only the hit count will used to detect a fail or neutral state.

Maximum Time *(URIBLmaxtime)*

10
This sets the maximum time in seconds to spend on each message performing URIBL checks.

Socket Timeout *(URIBLsocktime)*

1
This sets the URIBL socket read timeout in seconds.

Whitelisted URIBL Domains* *(URIBLwhitelist)*

doubleclick.net|www.w3.org|schemas.microsoft.com
This prevents specific domains from being checked by URIBL module. For example: doubleclick.net|www.w3.org|schemas.microsoft.com or file:files/URIBLwhitelist.txt. Domains already listed in **noProcessingDomains** and **whiteListedDomains** will be honored.

Don't Check Messages from these Addresses* *(noURIBL)*

Don't validate URIBL when messages come from these addresses. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).
For example: fribo@thisdomain.com|jhanna|@sillyguys.org

Bad URI IP's* *(URIBLIPRe)*

Every IP in a URI and every IP resolved for a hostname in a URI is checked against this list of IP's or networks. For example:145.145.145.145|145.146.|1.2.0.0/16
This high security feature will follow the rules in **URIBLWL**, **URIBLNP**, **URIBLLocal** and **URIBLISP** - but if a match is found, it will block the email (ignores scoring, monitoring, testmodes and spamlover).

☑ **Add X-Assp-Received-URIBL Header *(AddURIBLHeader)***
Add X-Assp-Received-URIBL header to messages with positive reply from URIBL.

☐ **Add X-Assp-Detected-URI Header *(AddURIS2MyHeader)***
URI's detected with URIBLOK are added to our header lines (X-Assp-Detected-URI:).

URIBL Cache Refresh Interval for Hits *(URIBLCacheInterval)*

1
Domains in cache will be removed after this interval in days. Empty or 0 will disable the cache. Show URIBL Cache

URIBL Cache Refresh Interval for Misses *(URIBLCacheIntervalMiss)*

0.5
Domains in cache with status=2 (miss) will be removed after this interval in days. Empty or 0 will prevent caching of non-hits.

Reply Code to Refuse Failed URIBL Message *(URIBLError)*

554 5.7.1 Blacklisted by URIBLNAME Contact the postmaster of this domain for resol
SMTP reply code to refuse failed URIBL message. The literal URIBLNAME (case sensitive) is replaced by the names of URIBLs with negative response. If this field is empty, client connection is simply dropped.

Notes On URIBL
Notes

Attachment Validation and Protection

External Attachment Blocking ***(DoBlockExes)***

disabled

This requires an installed **Email::MIME** module in PERL.

External Attachment Blocking Level ***(BlockExes)***

no check

Set the level of Attachment Blocking to 1-3 for attachments that should be blocked, set level to 4 for attachments that should be allowed. Choose 0 for no attachment blocking.

Whitelisted & Local Attachment Blocking ***(BlockWLExes)***

no check

Set the level of Attachment Blocking to 0-4 for whitelisted & local senders. Choose 0 for no attachment blocking.

NoProcessing Attachment Blocking ***(BlockNPExes)***

no check

Set the level of Attachment Blocking to 0-4 for no processing senders. Choose 0 for no attachment blocking.

Level 1 rejected File Extensions ***(BadAttachL1)***

exe\-bin|exe|scr|pif|vb[es]?|jse?|ws[cfh]?|sh[sb]?|li?nk|bat|cmd|com|ht[ab]|ps1?

This regular expression is used to identify Level 1 attachments that should be blocked.
Separate entries with a pipe |. The dot . is assumed to precede these, so don't include it.
For example:
ad[ep]|asx|ba[st]|chm|cmd|com|cpl|crt|dbx|exe|exe\-bin|hlp|ht[ab]|in[fs]|isp|js|jse|lnk|md[abez]|mht|ms[cipt]|nch|pcd|pif|prf|ps1?|reg|sc[frt]|sh[bs]|vb|vb[es]|wms|ws[cfh]
If you've installed the ASSP_AFC Plugin (at least version 2.10) and 'exe-bin' is defined (on any level), the Plugin will detect executable files based on there binary content. Detected will be all executables, libraries and scripts for DOS and Windows (except .com files), MS office macros (VBA), MAC-OS and linux ELF (for all processor architectures).
If you want to skip the detection for a specific executable type, define any combination of the tags below like: 'exe-bin|:WSH|:MSOM|:WIN' - notice the leading collon for the exceptions!

:WIN - windows executables
:MOS - Mach-O executables
:PEF - Classic MacOS executables
:ELF - ELF (linux) executables
:WSH - windows shell scripts
:MMC - windows MMC Console Files
:ARC - static library (linux,unix)
:CSC - common scripts (basic,java,perl,php,powershell....)
:MSOM - microsoft office macros

Level 2 rejected File Extensions ***(BadAttachL2)***

This regular expression is used to identify Level 2 attachments that should be blocked.
Level 2 already includes all rejected extensions from Level 1.
For example:
(ad[ep]|asx|ba[st]|chm|cmd|com|cpl|crt|dbx|exe|hlp|ht[ab]|in[fs]|isp|js|jse|lnk|md[abez]|mht|ms[cipt]|nch|pcd|pif|prf|reg|sc[frt]|sh[bs]|vb|vb[es]|wms|ws[cfh]).zip

Level 3 rejected File Extensions ***(BadAttachL3)***

This regular expression is used to identify Level 3 attachments that should be blocked.
Level 3 includes Level 2 and Level 1.
For example:
zip|url

Level 4 Allowed File Extensions ***(GoodAttach)***

This regular expression is used to identify attachments that should be allowed. All others are blocked. Separate entries with a pipe |. The dot . is assumed to precede these, so don't include it.
For example:
ai|asc|bhx|dat|docx?|eps|gif|htm|html|ics|jpg|jpeg|hqx|od[tsp]|pdf|p7[mscz]|ppt|rar|rpt|rtf|snp|txt|xls|zip|7z

User based Good and Bad Attachments* ***(UserAttach)***

This set of regular expression is used to identify attachments that should be allowed or blocked for specified users and/or domains. Separate entries with a any of '=> , ; space'. Separate multiple regex entries with pipe '|'. The dot . is assumed to precede the regex, so don't include it anywhere (except the user name).
To define entries you have to use the 'file:...' option. Define one entry per line - comments are not allowed in a definition line.
The syntax of an entry is as follows:
username => good => goodAttachRegex , good-out => goodoutRegex , good-in => goodinRegex , block => blockAttachRegex , block-out => blockoutRegex , block-in => blockinRegex
username - Mail solely to or from any of these addresses. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com) or a Group definition [GROUP]. Wildcards are supported (fribo*@domain.com).

good => goodAttachRegex - good attachment for incoming and outgoing mails
good-out => goodoutRegex - good attachment for outgoing mails
good-in => goodinRegex - good attachment for incoming mails
block => blockAttachRegex - bad attachment for incoming and outgoing mails
block-out => blockoutRegex - bad attachment for outgoing mails
block-in => blockinRegex - bad attachment for incoming mails

For example:

user@domain.tld => good => ai|asc|bhx|dat|doc|eps|gif|htm|html|ics|jpg|jpeg|hqx|od[tsp]|pdf|ppt|rar|rpt|rtf|snp|txt|xls|zip
*@domain.tld => good => ai|asc|bhx , good-out => eps|gif , good-in => htm|html , block => pdf|ppt , block-out => rar|rpt , block-in => xls|exe\-bin

At least one of the above option must be defined in a line - a maximum of all (six) could be defined, if this makes sense.
This feature replaces the above level definitions. If at least one valid regular (not zip:... from the ASSP_AFC Plugin) attachment blocking rule is defined here, all level definitions are ignored for all emails!
The defined blocking rules for the sender and the first envelope recipient are combined together using an OR logic.
good, good-out and good-in - and also - block, block-out and block-in - will be logical OR combined according to the mail flow.
Notice: if a bad attachment is found on a user based attachment check, the penalty box IP address scoring is skipped.

Reply Code to Refuse Rejected Attachments *(AttachmentError)*

550 5.7.1 These attachments are not allowed -- Compress before mailing.

The literal 'FILENAME' will be replaced with the name of the blocked attachment!

☑ **Refuse Uuencoded Mails *(BlockUuencoded)***

Reply to Refuse Uuencoded Mails *(UuencodedError)*

554 5.7.1 This message is uuencoded and will be blocked.

For example: 554 5.7.1 This mail is uuencoded and will be blocked

Notes On Attachment Blocking

Notes

Virus Protection using ClamAV and OS-FileScanner

Do Not Scan Messages from/to these Addresses* *(noScan)*

Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).

Do Not Scan Messages from these IP's* *(noScanIP)*

Enter IP addresses that you don't want to be scanned for virus , separated by pipes (|). For example: 145.145.145.145|145.146.

Skip Virus RegEx* *(NoScanRe)*

Put anything here to identify messages which should not be checked for viruses.

No-Blocking Virus Scan Scoring Regex *(SuspiciousVirus)***

file:files/suspiciousvirus.txt [Edit file]

If a ClamAV or FileScan result matches this expression it will be scored with the suspicious virus score (**vsValencePB**) and the message will not be blocked.
It is possible to weight such results. Every weighted regex that contains at least one '|' has to begin and end with a '~' - inside such regexes it is not allowed to use a '~', even it is escaped - for example: ~abc\~|def~=>23 or ~abc~|def~=>23 - instead use the octal (\126) or hex (\x7E) notation , for example ~abc\126|def~=>23 or ~abc\x7E|def~=>23 . Every weighted regex has to be followed by '=>' and the weight value. For example:
Phishing\.=>1.45|~Heuristics|Email~=>50
or
~(Email|HTML|Sanesecurity)\.(Phishing|Spear|(Spam|Scam)[a-z0-9]?)\.~=>4.6|Spam=>1.1|~Spear|Scam~=>2.1 .
The multiplication result of the weight and the penaltybox valence value will be used for scoring, if the absolute value of weight is less or equal 6. Otherwise the value of weight is used for scoring.

☑ **Scan Whitelisted Senders *(ScanWL)***

☐ **Scan No Processing Senders *(ScanNP)***

☐ **Scan Local Senders *(ScanLocal)***

☐ **Scan Copied Spam and Forwarded Ham Mails *(ScanCC)***

Reply Code to Refuse Infected Messages *(AvError)*

554 5.7.1 Mail appears infected with \[$infection\].

Reply code to refuse infected messages. The string $infection is replaced with the name of the detected virus.
For example: 554 5.7.1 Mail appears infected with \[$infection\] -- disinfect and resend.

Send Virus Report To These Addresses *(EmailVirusReportsTo)*

If set, an email containing the Message ID, Remote IP, Message Subject, Sender email address, Recipient email address, and the virus detected will be sent to these addresses. For example: admin@domain.com .
It is possible to define multiple addresses separated by pipe (|) e.g: admin@domain.com|virusalert@domain.com .
In addition, a leading 'IN:' or 'OUT:' can be specified in front of each address for incoming or outgoing/local mails. e.g: commonvirusalert@domain.com|IN:inboundvirusalert@domain.com|OUT:localvirusalert@domain.com .
The literals 'USER' and 'DOMAIN' will be replaced by the user part and domain part of the sender for outgoing/local mails and the recipient for incoming mails.

☐ **Add Full Header To Virus Report To Mail Address Above *(EmailVirusReportsHeader)***

If set the full message headers will also be added to Virus Reports.

Send Virus Report To Recipient *(EmailVirusReportsToRCPT)*

disabled [edit report file: reports/virusreport.txt]

If set the intended recipient of the message will be sent a copy of the Virus Report. If "for HAM only" is selected, the report will only be sent, in case the mail is not detected as SPAM before the virus check is done.

Use File System Virus Scanner *(DoFileScan)*

disabled

If activated, the message is written to a file inside the '**FileScanDir**' with an extension of '**maillogExt**'. After that ASSP will call '**FileScanCMD**' to detect if the temporary file is infected or not. The temporary created file(s) will be removed.
The infected file will be stored in a special folder, if the **SpamVirusLog** is set to 'quarantine' and the filepath to the **viruslog** is set.
Please check the setting of **FileLogScan** before you enable this option!

File Scan Directory *(FileScanDir)*

c:/assp/virusscan

Define the full path to the directory where the messages are temporary stored for the file system virus scanner. This could be any directory inside your file system. The running ASSP process must have full permission to this directory and the files inside!

File Scan Command *(FileScanCMD)*

ASSP will call this system command and expects a returned string from this command. This returned string is checked against '**FileScanBad**'

and/or '**FileScanGood**' to detect if the message is OK or not! If the file does not exists after the command call, the message is consider infected. ASSP expects, that the file scan is finished when the command returns!
The literal 'FILENAME' will be replaced by the full qualified file name of the temporary file.
The literal 'NUMBER' will be replaced by the threadnumber and could be used to name logfiles and to redirect them to STDOUT.
The literal 'FILESCANDIR' will be replaced with the value of FileScanDir.
Any case sensitive literal starting and ending with an asterix (*) like '*rcpt*' or '*mailfrom*' will be replaced by the quoted runtime connection variable of Con{fh}->{literal} (this->{literal}). You need to know the assp internals!
If a code reference is defined for the internal variable $main::FileScanCMDbuild_API in lib/CorrectASSPcfg.pm , assp will call '$FileScanCMDbuild_API->(\$cmd,$this)' before running the command. The first parameter, the command (**FileScanCMD**), is submitted as a reference to a scalar, which must be modified in place. If you want assp not to scan the message, set this variable to undef. The second submitted parameter is the reference to the client connection parameter HASH - $Con{fh} (eg. $this)
All outputs of this command to STDERR are automatic redirected to STDOUT.
FileScan will not run, if **FileScanCMD** is not specified.
If you have your online/autoprotect file scanner configured to delete infected files inside the '**FileScanDir**', define 'NORUN' in this field! In this case **FileScanGood** and **FileScanBad** are ignored. If there is a need to wait some time for the autoprotect scanner, write 'NORUN-dddd', where dddd are the milliseconds to wait!
Depending on your operating system it may possible, that you have to quote (' or ") the command, if it contains whitespaces. The replaced file name will be quoted by ASSP if needed.

RegEx to Detect 'BAD' in Returned String* ***(FileScanBad)***

Put anything here to identify bad messages by the string returned from the **FileScanCMD**. If defined and this regular expression matches, the message is consider infected.

RegEx to Detect 'GOOD' in Returned String* ***(FileScanGood)***

Put anything here to identify good messages by the string returned from the **FileScanCMD**. If defined and this regular expression matches and '**FileScanBad**' does not, the message is consider not infected.
If both **FileScanBad** and **FileScanGood** are defined, **FileScanBad** has not to match and **FileScanGood** has to match, to consider a mail not infected!

FileScan Responds Regex* ***(FileScanRespRe)***

A regular expression that will be used over the text returned from the **FileScanCMD**. The result of this regex is used as virus name ($infection) in **AvError**. For example: infected by ([^\r\n]+)

Scan Resent and Stored Files for Virus with FileScan ***(FileLogScan)***

scan resend folder and collected files

If virus check is enabled (**DoFileScan**), every file/mail (except reports - eg. n10000123456.eml) in the '**resendmail**' folder and if selected, every collected file is scanned for virus before it is sent or stored.
If a virus is found, the file/mail is not (re)sent (it will get the extension '.virus') and a notification mail will be sent to local users. Infected collected files are moved in to the **SpamVirusLog** folder.
To force the resend of a virus infected mail, the header tag 'X-ASSP-ForceResend:' must be added to the file!
If 'scan resend folder and collected files' is selected, it could be possible, that the virus scanner (**FileScanCMD**) forces a very high system workload.
If you are not sure what to set here, leave the setting at the default 'scan resend folder only'!
If the ASSP_AFC Plugin is installed and configured to be used, the files in the resend folder will be scanned by FileScan and ClamAV if any of **FileLogScan** or **ClamAVLogScan** is configured.
Under normal conditions the scan will be done by the SMTP-worker, if assp is under a havy workload, the scan request will be transfered to the High-Workers (10000/10001).

☐ **Use ClamAV** ***(UseAvClamd)***

If activated, the message is checked by ClamAV, this requires an installed File::Scan::ClamAV Perl module and a running Clamd . It is not recommended to use ClamAV on heavy-load systems, because of resulting system overload, stuck workers or timeouts.
The infected file will be stored in a special folder, if the **SpamVirusLog** is set to 'quarantine' and the filepath to the **viruslog** is set.
Please check the setting of **ClamAVLogScan** before you enable this option!

Port or file socket for ClamAV ***(AvClamdPort)***

A socket specified in the clamav.conf file - LocalSocket. For example /tmp/clamd. If the socket has been setup as a TCP/IP socket (see the TCPSocket option in the clamav.conf file), then specify the TCP socket. For example: 3310 .
For remote host TCP connections define the hostname or IP-address in front of the port definition - example: clamhost:3310 or 192.168.0.1:3310 . If the hostname is not defined, localhost will be used as default.
It is possible to define multiple hosts to balance the workload - define them separated by pipe (|) - example: clamhost:3310|192.168.0.1:3310
If multiple hosts are defined, they are used in a random round-robin mode.

ClamAV Bytes ***(ClamAVBytes)***

60000

The number of bytes per message or file that will be submited to ClamAV and FileScan for virus scanning. Values of 100000 or larger are not recommended, because while a thread is waiting for the scanner result, it could not get new connections.

Scan Resent and Stored Files for Virus with ClamAV ***(ClamAVLogScan)***

scan resend folder and collected files

If virus check is enabled (**UseAvClamd**), every file/mail (except reports - eg. n10000123456.eml) in the '**resendmail**' folder and if selected, every collected file is scanned for virus before it is sent or stored.
If a virus is found, the file/mail is not (re)sent (it will get the extension '.virus') and a notification mail will be sent to local users. Infected collected files are moved in to the **SpamVirusLog** folder.
To force the resend of a virus infected mail, the header tag 'X-ASSP-ForceResend:' must be added to the file!
If 'scan resend folder and collected files' is selected, it could be possible, that the virus scanner (clamd) forces a very high system workload.
If you are not sure what to set here, leave the setting at the default 'scan resend folder only'!
If the ASSP_AFC Plugin is installed and configured to be used, the files in the resend folder will be scanned by FileScan and ClamAV if any of **FileLogScan** or **ClamAVLogScan** is configured.

Under normal conditions the scan will be done by the SMTP-worker, if assp is under a havy workload, the scan request will be transfered to the High-Workers (10000/10001).

ClamAV Timeout *(ClamAVtimeout)*

10

ClamAV will timeout after this many seconds.
default: 10 seconds.

Notes On Virus Control

Notes

Perl Regular Expression Filter and Spambomb Detection

☐ **Allow Internal Variables in Regex** *(AllowInternalsInRegex)*

Allow internal variables to be used in regular expressions - replaces something like ${$EmailDomainRe} with the value of (?^u:(?:[a-zA-Z0-9_][a-zA-Z0-9_\-]*(?:\.[a-zA-Z0-9_][a-zA-Z0-9_\-]*)*\.(?:xn--[a-zA-Z0-9\-]+|[a-zA-Z0-9_][a-zA-Z0-9_]+)|\[[0-9][0-9\.]*\.[0-9]+\]))

Regular Expression to early Identify Spam in Handshake and Header Part* *(preHeaderRe)*

file:files/preheaderre.txt [Edit file]

Until the complete mail header is received, assp is processing the handshake and header content line per line, but the first mail content check is done after the complete mail header is received.
It is possible, that some content (malformed headers, forbidden characters or character combinations) could cause assp to die or to run in to a unrecoverable exception.
Use this regular expression to identify such incoming mails based on a line per line check, at the moment where a single line is received.
This setting does not affect any other and is not affected by any other configuration setting, except that this check is only done for incoming mails.
If a match is found, assp will immediately send a '421 <**myName**> closing transmission' reply to the client and will immediately terminate the connection.
Default setting is file:files/preheaderre.txt

☐ **Do Bomb/Script Regular Expressions Checks for Whitelisted** *(bombReWL)*

☐ **Do Bomb/Script Regular Expressions Checks for NoProcessing** *(bombReNP)*

☐ **Do Bomb/Script Regular Expressions Checks for Local Messages** *(bombReLocal)*

☑ **Do Bomb/Script Regular Expressions Checks for ISPIP** *(bombReISPIP)*

Maximum Penalty on Bombs per Mail per Check *(bombMaxPenaltyVal)*

70

Depending on the configuration, it could be possible that a message gets a very high penalty value on a bomb-check. This value limits the maximum penalty per mail for every single bomb-check that is enabled.

Maximum time spend on Bomb Search *(maxBombSearchTime)*

5

Maximum time in seconds that is spend on every configured bomb check. This time check is done, after every found bomb. So it is possible that the bomb search takes longer as the defined value, if no bomb is found or a single search takes more time. Default is 5.

Even if any of the following bomb parameters is set to "block", but the sum of the resulting weighted penalty value is less than the corresponding "Penalty Box Valence Value" (because of lower weights) - only scoring will be done!
A description of how of weighting regular expressions is done and working, could be found at the bottom this web page.

☐ **Transliterate non-Roman characters in to Roman** *(DoTransliterate)*

If enabled, ASSP tries to transliterate non-Roman characters in an email it to Roman characters. These transliterations are than additionally used in the bomb checks.
For example - the (character) sequence '年光通信产业会回归高增长轨道' will be transliterated to 'Nian Guang Tong Xin Chan Ye Hui Hui Gui Gao Zeng Chang Gui Dao' .
To transliterate something, use the 'Mail Analyzer'.
To make this feature working, the Perl module **Text::Unidecode** must be installed.

Use BombHeader Regular Expressions on Header Part *(DoBombHeaderRe)*

block

If activated, each message-header is checked against **bombSenderRe**, **bombHeaderRe**, **bombSubjectRe** and **bombCharSets** Regular Expressions. If you use **sendAllSpam**, be aware that only the header will be shown in the spamcopy.
The scoring value is the sum of all valences(weights) of all found bombs - **bombValencePB** .

**Envelope Blocking Regular Expression ** *(bombSenderRe)*

emailserver3\.com|\d\d\d\d\d\d\@tom\.com

Part of **DoBombHeaderRe**: expression to identify sender (mailfrom,ip,helo).

Regular Expression to Identify Spam in Header Part *(bombHeaderRe)*

file:files/bombheaderre.txt [Edit file]

Part of **DoBombHeaderRe**: header will be checked against this Regex if **DoBombHeaderRe** is enabled. For example file:files/bombheaderre.txt

Regular Expression to Identify Spam in Subject *(bombSubjectRe)*

Part of **DoBombHeaderRe** : the mail header will be checked against this Regex if **DoBombHeaderRe** is enabled. If **DoBombHeaderRe** is enabled, the mail subject will be automatically checked against RFC2047 (for NON printable characters in the undecoded MIME content).

Maximum allowed Subject Length *(maxSubjectLength)*

200=>100

If set to a value greater than 0, assp will check the length of the Subject of the mail. If the Subject length exceeds this value, the message score will be increased by '**bombValencePB**' and the string that is checked in '**bombSubjectRe**' will be trunked to this length. It is possible to define a special weight using the syntax 'length=>value', in this case the defined absolute value will be used instead of '**bombValencePB**' to

increase the message score. If the subject is too long and this weight is equal or higher than '**bombMaxPenaltyVal**' no further bomb checks will be done on the subject.

Regular Expression to Identify Foreign Charsets** ***(bombCharSets)***

charset=(?:BIG5|CHINESEBIG|GB2312|KS_C_5601|KOI8-R|EUC

Part of **DoBombHeaderRe**: header will be checked against this Regex if **DoBombHeaderRe** is enabled. The literal UNKNOWN will detect all wrong defined MIME character sets.
Part of **DoBombRe** : every MIME-part header will be checked against this Regex if **DoBombRe** is enabled.
For example:
charset=(?:BIG5|CHINESEBIG|GB2312|KS_C_5601|KOI8-R|EUC-KR|ISO-2022-JP|ISO-2022-KR|ISO-2022-CN|CP1251|UNKNOWN).

Maximum Hits for Bombs in Header and Sender ***(bombHeaderReMaxHits)***

1

A hit is a found Bomb in header and sender - **bombSenderRe** , **bombHeaderRe** , **bombSubjectRe** , **bombCharSets** .
If the number of hits is greater or equal Maximum Hits, the email is flagged **Failed** (possibly blocked and/or scored).
If the number of hits is greater 0 and less Maximum Hits, the email is flagged **Neutral** (possibly scored)

Use Bomb Regular Expressions ***(DoBombRe)***

block

If activated, each message is checked against **bombRe** and BombData Regular Expressions.
The scoring value is the sum of all valences(weights) of all found bombs - **bombValencePB** .

Regular Expression for Header and Data Part** ***(bombRe)***

file:files/bombre.txt Edit file

Header and Data will be checked against this Regular Expression if **DoBombRe** is enabled. For example:
IMG [^>]*src=['"]cid|<BODY[^>]*>(<[^>]+>|\n|\r)*<IMG[^>]+>(<[^>]+>|\n|\r)*</BODY>
If you want to search for attachment names, define a line with 'attachment:the_attachment_name'.

Regular Expression to Identify skipped Tags in Header Part* ***(bombSkipHeaderTagRe)***

file:files/bombskipheadertagre.txt Edit file

Regular Expression to define header tags, that will be skipped for **bombSuspiciousRe**, **bombHeaderRe**, **bombRe** and **blackRe** - like 'DKIM-Signature|Domainkey-Signature' - the always followed collon (:) is added by assp. For example
file:files/bombskipheadertagre.txt

Maximum Hits for Bombs in Header and Data ***(bombReMaxHits)***

1

A hit is a found Bomb in header and data - **bombRe** .
If the number of hits is greater or equal Maximum Hits, the email is flagged **Failed** (possibly blocked and/or scored).
If the number of hits is greater 0 and less Maximum Hits, the email is flagged **Neutral** (possibly scored)

BombData Regular Expression for Data Part** ***(bombDataRe)***

Data part will be checked against the Regular Expression if **DoBombRe** is enabled. For example:
IMG [^>]*src=['"]cid|<BODY[^>]*>(<[^>]+>|\n|\r)*<IMG[^>]+>(<[^>]+>|\n|\r)*</BODY>
If you want to search for attachment names, define a line with 'attachment:the_attachment_name'.

Maximum Hits for Bombs in Data ***(bombDataReMaxHits)***

1

A hit is a found Bomb in data - **bombDataRe** .
If the number of hits is greater or equal Maximum Hits, the email is flagged **Failed** (possibly blocked and/or scored).
If the number of hits is greater 0 and less Maximum Hits, the email is flagged **Neutral** (possibly scored)

Suspicious Expression for Scoring Only** ***(bombSuspiciousRe)***

Sender, Header and Data will be checked for scoring only. Put here anything which might be suspicious. **bombSuspiciousValencePB** will be used to increase the score.
For example:
unsubscribe

NOTICE: BombSuspiciousRe is processed per default for all mails (incoming and outgoing) regardless of noprocessing and whitelisting! Only noBombScript is observed in every case.
To change this behavior, use the enhanced regular expression syntax (NWIL) described at the bottom of the GUI!

Don't Check Messages from these Addresses* ***(noBombScript)***

Don't detect spam bombs or scripts in messages from these addresses. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).

☐ **Do Test Regular Expression** ***(DoTestRe)***

If activated, each message is checked against the Test Regular Expression below. This provides a way to test regex strings on live mail.

Test Regular Expression** ***(testRe)***

Use this to test your regular expressions. Test valence is **teValencePB** .

Spam Bomb Error *(bombError)*

554 5.7.1 Delivery not authorized, message refused -- .

SMTP error message to reject spam bombs. For example: 554 5.7.1 Delivery not authorized, message refused -- send report to mailto:postmaster@mydomain.tld or call +12.34.56.78.90

☑ **Add Reason** *(bombErrorReason)*

Add matching expression to Spam Bomb Error

Use Black Regular Expression to Identify Spam Strictly *(DoBlackRe)*

disabled

Each incoming message (except **acceptAllMail**) is checked against the BlackRe to Identify Spams. No Optout.
The scoring value is the sum of all valences(weights) of all found bombs - **blackValencePB** .

BlackRe - Regular Expression to Identify Spam Strictly** *(blackRe)*

file:files/blackre.txt Edit file

If an incoming email (except **acceptAllMail**) matches this Perl regular expression it will be strictly considered spam . For example: \breplica watches\b|\bMegaDik\b|\bcock\b|\bpenis\b|\bpills\b|\bOriginal Viagra\b|\bbetter sex life\b|\baverage penis\b|\benlargement\b|\borgasm\b|\berections\b|\bViagra\b|\bbig dick\b|\bsperma\b|\bSexual\b|\bErectionsk\b|\bStamina\b|\bsildenafil\b|\bcitrate\b|\bErectile\b

Maximum Hits for Identify Spam Strictly *(blackReMaxHits)*

1

A hit is a found Bomb for Identify Spam Strictly. - **blackRe**
If the number of hits is greater or equal Maximum Hits, the email is flagged **Failed** (possibly blocked and/or scored).
If the number of hits is greater 0 and less Maximum Hits, the email is flagged **Neutral** (possibly scored)

Use Regular Expression to Identify Mobile Scripts *(DoScriptRe)*

disabled

Each message is checked against the Expression to Identify Mobile Scripts.
The scoring value is the sum of all valences(weights) of all found bombs - **scriptValencePB** .

Regular Expression to Identify Mobile Scripts** *(scriptRe)*

Spam mails may contain mobile scripting code, eg activex and java or php. You can use this feature to block those messages.
Leave this blank to disable the feature. For example:
\<applet|\<embed|\<iframe|\<object|\<script|\<?php|onmouseover|onload|onfocus|onblure|onclick|javascript:

Maximum Hits for Identify Mobile Scripts *(scriptReMaxHits)*

1

A hit is a found mobile scripting code for Identify Mobile Scripts - **scriptRe** .
If the number of hits is greater or equal Maximum Hits, the email is flagged **Failed** (possibly blocked and/or scored).
If the number of hits is greater 0 and less Maximum Hits, the email is flagged **Neutral** (possibly scored)

Script Error *(scriptError)*

554 5.7.1 Your email contains html scripting code -- please resend as plain text.

SMTP error message to reject scripts. For example: 554 5.7.1 Your email appears to be spam -- send an error report to mailto:postmaster@mydomain.tld or call +12.34.56.78.90

Notes On Bomb Regex

Notes

Hidden Markov Model and Bayesian Options

Bayesian Check *(DoBayesian)*

disabled

If activated, the message is checked based on Bayesian factors in **spamdb** for global and private entries. Private **spamdb** entries have a five times higher weight than global entries. This needs a fully functional **spamdb** built by rebuildspamdb. For starters it is best practice to put this inactive and build the **spamdb** collection with the help of DNSBL ,URIBL and **spamaddresses**. Scoring is done with **baysValencePB** for external mails, **baysIocalValencePB** is used for outgoing and internal mails - both values are multiplied with the detected **baysProbability** . It is possible to score (in and out) with a bonus for HAM with **bayshamValencePB** (**bayshamValencePB** * (1 - **baysProbability**)).
Both, the Bayesian-check and the Hidden-Markov-Model-check (below), are using Perl version depending (Perl 5.12 and higher) **Unicode** features to recognize any possible character. How ever, some east asian languages (and some others) have graphemes, that contains multiple **unicode code points**. If you need (or want) assp to process all text as a sequence of **UAX #29 Grapheme Clusters**, the Perl module **Unicode::LineBreak** is required.

Hidden Markov Model Check *(DoHMM)*

disabled

If activated, the message is checked based on a **Hidden Markov Model** for global and private entries. Private HMM entries have a five times higher weight than global entries. This needs a fully functional **HMMdb** database built by rebuildspamdb. For starters it is best practice to put this in monitoring mode and build the HMM collection with the help of DNSBL ,URIBL and **spamaddresses**. Scoring is done with **HMMValencePB** for external mails, **HMMIocalValencePB** is used for outgoing and internal mails - both values are multiplied with the detected hmmProbability. It is possible to score (in and out) with a bonus for HAM with **HMMhamValencePB** (**HMMhamValencePB** * (1 - **baysProbability**)).
The perl module **BerkeleyDB** version 0.34 or higher and BerkeleyDB version 4.5 or higher is required (to store temporary data) to use this feature and '**useBerkeleyDB**' must be set to ON.
If this option is disabled, the rebuildspamdb task will **NOT** build a valid HMM database!
Compared to the Bayesian option, the Hidden Markov Model will produce results that are much more exact. How ever, it is possible, that HMM gets no result on very small messages, for this reason it is recommended to use both Bayesian and HMM. If you enable both checks, check your settings for **baysValencePB**, **HMMValencePB**, **baysIocalValencePB** and **HMMIocalValencePB** - eg. divide them by 2. or set the bayes values to 1/3 and the HMM values to 2/3.
NOTICE that using this option requires a **very fast database server** behind, if **HMMusesBDB** is set to OFF. The Bayesian- and HMM check together can produce **4000 and much more SQL queries per second**.
Keep in mind, that all backups and exports of the HMM database could require several 100MB of diskspace, if the file count in the corpus is very large.

Do Bayesian depends on HMM results *(BayesAfterHMM)*

This value is ignored, if **DoHMM** is not enabled or set to monitor or **DoBayesian** is disabled.
The Bayesian check will only run, if the spam/ham probability of the HMM check is in a given value range or the HMM check has given too few results or the confidence (**baysConf**) of the detection is too low.
Leave this blank to run the Bayesian check every time, independent from any HMM result (default).
To set this value, define a probability value range like 0.4-0.6 or 0.3-0.7 - eg: best set it according to the setting of **baysProbability** ([1 - **baysProbability**]-**baysProbability**).

Ignore a database version missmatch *(ignoreDBVersionMissMatch)*

Spam and HMMDB

The status of assp is changed to "not healthy" if the current version of any of Spamdb or **HMMdb** is not equal to the required database version. Such a missmatch is automatically corrected with the next successful rebuildspamdb. How ever, if you are unable to solve this problem for any reason, you should set this value to keep the status of assp "healthy".

☑ **Use BerkeleyDB for the Hidden Markov Model database** *(HMMusesBDB)*

If enabled (default), the Hidden Markov Model database uses BerkeleyDB - notice: in this case no database import, backup or export are provided for the **HMMdb**. This value is completely ignored, if **DBdriver** is set to 'BerkeleyDB' and **spamdb** is set to 'DB:'. Switch this parameter to OFF, if you want to use the same database engine for the **HMMdb** like **spamdb** is configured.
Changing this value requires a restart of assp. Possibly a forced rebuildspamdb is required after the restart.

Use also private entries for the Bayesian Spamdb and Hidden Markov Model databases *(DoPrivatSpamdb)*

NO

If enabled, private entries (based on the local recipient and/or the report sender email address) will be added to the Bayesian and HMM databases. These private entries have a three times higher priority for users (full email address) and two times higher priority for domains (domain part of the email address) than global entries. To enable this option "**spamdb**" must be set to use a database "DB:" first!
Setting this option to ON, will increase the record count for the spamdb and the HMM databases dramaticaly!

Bayesian and HMM Check Timeout *(BayesMaxProcessTime)*

15

The Bayesian- and HMM checks are the most memory and CPU consuming tasks that ASSP is doing on a message. If such tasks running to long on one message, other messages could run in to SMTPIdleTimeout. Define here the maximum time in seconds that ASSP should spend on Bayesian Checks for one message. Default is 60.

☐ **Bayesian/HMM Check on Whitelisted NON Local Senders/Messages** *(BayesWL)*

If enabled, the Bayesian/HMM check is done on whitelisted NON local senders/messages.

☐ **Bayesian/HMM Check on NoProcessing Messages** *(BayesNP)*

If enabled, the Bayesian/HMM check is done on NoProcessing messages.

☐ **Bayesian/HMM Check on Local Senders** *(BayesLocal)*

If enabled, the Bayesian/HMM check is done on local and outgoing messages

Skip Bayesian and HMM Check* *(noBayesian)*

Mail from/to any of these addresses are ignored by Bayesian- and HMM check, mails will not be stored in spam/notspam collection. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com)

Skip Bayesian and HMM Check for this local senders* ***(noBayesian_local)***

Mail from any of these local addresses are ignored by Bayesian- and HMM checks, mails will not be stored in spam/notspam collection. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com)

Do Bayesian and HMM Check ONLY for this local senders* ***(Bayesian_localOnly)***

Only mail from any of these local addresses are processed by the Bayesian- and HMM checks, except they are also defined in **noBayesian_local** . **BayesLocal** must be switched on to make this option working. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com)

Maximum most significant results used per mail to calculate Bayesian- and HMM-Probability ***(maxBayesValues)***

60

Maximum count of most significant values used to calculate the Bayesian/HMM-Spam-Probability and the confidence of that probability.
The Bayesian/HMM Spam probability will be fine with 30 and will get more exact, than higher this value is - until a value of 60.
The confidence of the Bayesian/HMM Spam probability will get better, than higher this value is.
Values above 60 are possible, but could lead in to a performance penalty, without getting a better spam detection.
If the HMM check gets less than (**maxBayesValues** / 3 + 1) results, the HMM check is set to scoring for the mail.
If the HMM check gets less than (**maxBayesValues** / 12 + 1) results, the HMM check is set to monitoring for the mail.
Default is '60', minimum is '30'.

Bayesian and HMM Probability Threshold ***(baysProbability)***

0.6

Messages with spam-probability below or equal this threshold are considered Ham. Recommended '0.6'. If you change this value, check your setting of **BayesAfterHMM** .
A resulting Spam-Probability above this value is multiplied with baysValencePB_local or **baysValencePB** to get the penaltybox scoring value for the IP- and message score. In other words, the penaltybox scoring value is weighted by the Spam-Probability in case Spam is detected.
A resulting Spam-Probability below this value but higher than (1 - **baysProbability**) is stated as 'UNSURE' . In this case the half score will be added to the message score but not to the IP score and the message will not be blocked.

The following default Bayesian math (prob = p1 / (p1 + p2)) is used to calculate the SpamProb value for 'n' found Bayesian-Word-Pairs or HMM-Sequences, each with a spam-weight 'p' - where 0<p<1 :

'SpamProb' = $(p_1 * p_2 * ... * p_n) / (p_1 * p_2 * ... * p_n + (1 - p_1) * (1 - p_2) * ... * (1 - p_n))$

Bayesian and HMM Confidence Threshold ***(baysConf)***

0

Spam-Mails having a confidence below this threshold are passed in TestMode . Spam-Mails having a confidence above this threshold are blocked. Set this only above 0 if you are familiar with the bayesian statistics used in ASSP.
Messages that are processed by the bayesian and HMM check get a spam-probability score and a confidence score. The confidence score in assp is a quality indicator. A confidence near 0 would mean the probability score is like a wild guess. A confidence score near 1 would mean that it's pretty sure that the bayesian analysis result is correct. The confidence threshold is an allowance to process a Bayesian/HMM Spam as-if in Bayesian TestMode, if the message's *confidence* score is lower than the confidence threshold. Set this level to a specfic value, let's say .001 (which is a good one for starting), then:
- messages with spam-probability higher than 0.6 and a confidence of less than 0.001 would come through as in test mode
- messages with spam-probability higher than 0.6 and a confidence of more than 0.001 would be blocked
- messages with spam-probability less than 0.6 would pass
The 0.6 threshold can be set in **baysProbability** .
The confidence of the probability value is also used in BayesAfterHMM.
Carefully set this parameter above 0, if the bayesian corpus norm (shown by the rebuildspamdb log) is less than 0.6 or higher than 1.4 .

The following math is used to calculate the SpamProbConfidence value for 'n' found Bayesian-Word-Pairs or HMM-Sequences doing 'q' database queries, each result with a spam-weight 'p' - where 0<p<1 :

extreme_confidence_count = $|(0 < p_{1...n} < 0.01)| - |(0.99 < p_{1...n} < 1)|$
extreme_confidence_count = 0 - if (extreme_confidence_count < 0 and SpamProb > 0.5) or (extreme_confidence_count > 0 and SpamProb <= 0.5) == TRUE;
extreme_confidence_count = abs(extreme_confidence_count)
mail_confidence = $abs((P_1 * P_2 * ... * P_k) - ((1 - P_1) * (1 - P_2) * ... * (1 - P_k)))$ - for all elements $P_{1...k}$ in $(0.01 < p_{1...n} < 0.99)$
corpus_confidence = $1 / ((abs(1 - corpus\_norm) + 1)^{int(abs(1 - corpus\_norm) * 10)})$ - the exponent is limited to a maximum of 4
q = max(n , min(q , **maxBayesValues**))
SpamProbConfidence = $0.01^{extreme\_confidence\_count} * mail\_confidence * corpus\_confidence * (n / q)^2$

The SpamProbConfidence is limited to a maximum of 1.0 .
All extreme values 'p' having a spam weight less than 0.01 or higher than 0.99 with a corresponding extreme value like (0.001 <-> 0.999) are ignored for the mail_confidence calculation.

empty or zero = disabled.

Show the Bayesian and Hidden-Markov-Model confidence distribution!

☑ **Reduce Scoring for Low Confidence** ***(baysConfidenceHalfScore)***

Spam-Mails having a confidence below the threshold, will get half of the normal penalty score for Bayesian and HMM hits.

☐ **Add Bayes and HMM Probability Header** ***(AddSpamProbHeader)***

Adds a line to the email header "X-Assp-Spam-Prob: 0.0123" and/or "X-Assp-HMM-Spam-Prob: 0.0123" Probability ranges from 0 to +1 where > 0.6 = spam.

☐ **Add Bayes and HMM Confidence Header** ***(AddConfidenceHeader)***

Adds a line to the email header "X-Assp-Bayes-Confidence: 0.0123" and/or "X-Assp-HMM-Confidence: 0.0123".

Notes On Bayesian
Notes

Outgoing Message Tagging, NDR Validation and Backscatter Detection

Do Message-ID Tagging and Validation (FBMTV) *(DoMSGIDsig)*

disabled

If activated, the message-ID of each outgoing message will be signed with a unique Tag and every incoming mail will be checked against this Tag. This tagging mode is called FBMTV "Forwarder(s) Bounce Message-ID Tag Validation" and it is worldwide unique to ASSP. This Tag is build nearly the same way, as BATVTag is build for the sender address. This Tag will be removed from any incoming email, to recover the original references in the mail header! If anything is changed on this option inside the mail, no DKIM-check will be done! Before activating **DoMSGIDsig**, please configure **MSGIDpreTag** and MSGIDsec!
If activated and a bounced mail from null sender or postmaster contains no valid signature the configured action is taken.
If activated and any other mail contains a valid signature (eg. because it is an answer/reply to a tagged mail), this mail will be flagged as noprocessing and whitelisted !
This check requires an installed **Digest::SHA1** module in Perl.

Message-ID pre-Tag for MSGID-TAG-generation *(MSGIDpreTag)*

sig

To use Message-ID signing and to create the MSGID-Tags, a pre-Tag is needed. This Tag must be 2-5 characters [a-z,A-Z,0-9] long. Default is 'sig'.

Message-ID Secrets for MSGID-TAG-generation* *(MSGIDSec)*

0=key0|1=key1|2=key2|3=key3|4=key4|5=key5|6=key6|7=key7|8=key8|9=key9

To use Message-ID signing and to generate the MSGID-Tags, at leased one secret key is needed, up to ten keys are possible.
The notation is : generationnumber[0-9]=secretKey. For example(do not use!): 0=jk09Z|1=oPLmn4g|.... . Multiple pairs are separated by pipes (|). Default is 0=key0|1=key1|2=key2|3=key3|4=key4|5=key5|6=key6|7=key7|8=key8|9=key9 . Do not defines spaces, tabs and '=' as part of the keys(secrets)!
Values that contains any default are not valid, please change them, to prevent detecting strange ASSP-signatures as valid local signatures!
For this reason, please define your secrets as unique as possible! The secrets are used randomly to build the Message-ID-Tags.

Do FBMTV For These Addresses Only* *(MSGIDsigAddresses)*

Mail to any of these addresses will be tagged and checked by FBMTV. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). If empty, FBMTV is done for all addresses.

Skip Message-ID signing, mail content dependent* *(noMSGIDsigRe)*

Use this to skip the Message-ID tagging depending on the content of the email. If the content of the email matches this regular expression (checking **MaxBytes** only), FBMTV will not be done. For example: 'I am out of office' .

☐ **Skip Message-ID signing for Redlisted mails** *(noRedMSGIDsig)*

If selected, FBMTV will not be done for redlisted emails!

Do BATV Tagging and Validation *(DoBATV)*

disabled

If enabled any sender address of outgoing mails is mangled with a **BATV-Tag**. Any incoming bounced mail is checked for a valid BATV-Tag. All valid (local) BATV-Tags will be removed from incoming mails - so whitelisting, delaying and all other recipient and sender based checks will use the normal addresses. If the BATV-check is successful, no MSGID-signing-check and DNS-Backscatter-check will be done! If any BATVTag was removed, no DKIM-check will be done! BATV-address-replacement is done, before the recipient replacement rules are processed!
This check requires an installed **Digest::SHA1** module in Perl.

BATV Secrets for BATV-TAG-generation* *(BATVSec)*

0=key0|1=key1|2=key2|3=key3|4=key4|5=key5|6=key6|7=key7|8=key8|9=key9

To use **BATV** and to create the BATV-Tags, at leased one secret key is needed, up to ten keys are possible.
The notation is : generationnumber[0-9]=secretKey. For example: 0=key0|1=KEYX45rt|.... . Multiple pairs are separated by pipes (|). Default is 0=key0|1=key1|2=key2|3=key3|4=key4|5=key5|6=key6|7=key7|8=key8|9=key9 . Do not defines spaces, tabs and '=' as part of the keys (secrets)! The secrets are use randomly to build the BATV-Tags.

☐ **remove strange BATV-Tags from incoming mails** *(removeBATVTag)*

Any strange BATV-signature will be removed from the sender address and the real sender address will be used! Using this together with **remindBATVTag** keeps your clients addressbooks (also whitelist, **delaydb** ...) clean from BATV-Tags. This will also work, if **DoBATV** is disabled. If you do not use **remindBATVTag** and the MTA behind ASSP sends a bounced mail back - this mail will fail on BATV on the recipients site. If any BATVTag was removed, no DKIM-check will be done!

☐ **store incoming strange BATV-Tags to remind them for outgoing bounce mails** *(remindBATVTag)*

If defined, any incoming stange BATV-signature will be stored and any recipient of outgoing bounce mails will be checked against this list. If there is found a valid (not older than 7 days) BATV-Tag for that recipient, it will be mangled in to the recipient address. This will also work, if **DoBATV** is disabled.

Do DNS-Backscatter Detection *(DoBackSctr)*

disabled

If activated, the IP-address of each message received for null sender,bounced or postmaster will be checked against the list below. DNS **base** checks requires an installed **Net::DNS** module in Perl.
For more information about backscatter detection please read **http://www.backscatterer.org/?target=usage**.

Backscatter-DNS Cache Refresh Interval *(BackDNSInterval)*

7

IP's in cache will be removed after this interval in days. 0 will disable the cache and the usage of **downloadBackDNSFile** and **localBackDNSFile**. Show Backscatter-DNS Cache

ServiceProvider for Backscatterer Detection* *(BackSctrServiceProvider)*

ips.backscatterer.org

ServiceProvider for DNS check on Backscatterer. Possible value is ips.backscatterer.org for DNS check.

☐ **Download the Backscatterer DNS-IP-List** ***(downloadBackDNSFile)***

If selected, the complete IP-list is downloaded to a local file. If **useDB4IntCache** is set, the list is stored in a BerkeleyDB database (**BackDNS2**). Otherwise the records will be stored in the **pbdb** cache BackDNS . The download will be skipped, if **useDB4IntCache** is not set and **mysqlSlaveMode** is set. IP's are checked on this file first, if the IP is not found on this list, a DNS query is done. It is recommended to use this option for ISP's and users with more than 1000 bounced mails a day. See wget-mirrors.uceprotect.net/rbldnsd-all/ips.backscatterer.org.gz

Local File for the Backscatterer DNS-IP-List ***(localBackDNSFile)***

file:files/backdnslist.txt Edit file

The name of the local file that is used for this IP-list. The content of this file is filled in to the 'Backscatter-DNS Cache' (**BackDNSInterval**). IP's from this list will be removed after one day from the cache.

The following configurations are valid for all Backscatter Detection Options!

☐ **Send 250 OK to ISP if any Backscatter Detection fails** ***(Back250OKISP)***

If any Backscatter check fails for a bounced mail that is coming from an ISPIP, ASSP will send "250 OK" to the ISP, but will discard the mail, if the check is configured to block!

☐ **Do Backscatter Detection checks for Whitelisted mail** ***(BackWL)***

Tagging will be always done, if not excluded by address or domain!

☐ **Do Backscatter Detection checks for No Processing mail** ***(BackNP)***

Tagging will be always done, if not excluded by address or domain!

Regular Expression to Skip all BackScatter Checks* ***(noBackSctrRe)***

If the contents of a mail matches these regular expressions, all BackScatter checks will be skipped.

Do not any Backscatter detection for this Addresses * ***(noBackSctrAddresses)***

Mail to and from any of these addresses will not be tagged and checked by any backscatter option. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).

Exclude these IP's from any Backscatter detection* ***(noBackSctrIP)***

Enter IP's that you want to exclude from FBMTV and Backscatter check, separated by pipes (|).

Notes On Backscatter Detection

Notes

TestModes and SPAM Tagging

Prepend Spam Subject *(spamSubject)*

Setting a filter to testmode will tell ASSP not to reject the mail but rather build up the whitelist and spam and notspam collections. This can go on for some time without disturbing normal operation. After this very important phase TestMode can be used to tag the message: if TestMode and the message is spam Spam Subject gets prepended to the subject of the email. For example: [SPAM]

☐ **Prepend Spam Tag** *(spamTag)*

If checked, the method(s) ASSP used which caught the spam will be prepended to the subject of the email. For example; [DNSBL]

☐ **All Test Mode ON** *(allTestMode)*

Turn all of the individual testmodes on - regardless of the individual test mode settings.

☐ **Bayesian/Hidden-Markov-Model Test Mode** *(baysTestMode)*

Bayesian/HMM Test Mode User Addresses* *(baysTestModeUserAddresses)*

These users are in test mode / mark subject only for bayesian spam, even with test mode above off

☐ **BlackDomain Test Mode** *(blTestMode)*

☐ **Helo Blacklist Test Mode** *(hlTestMode)*

☐ **Forged Local Domain Test Mode** *(flsTestMode)*
-> **DoNoValidLocalSender**

☐ **SPF Test Mode** *(spfTestMode)*

☐ **DNSBL Test Mode** *(rblTestMode)*

☐ **Bad Attachment Test Mode** *(attachTestMode)*

☐ **URIBL Test Mode** *(uriblTestMode)*

☐ **SRS Test Mode** *(srsTestMode)*

☐ **Bomb Regex Test Mode** *(bombTestMode)*

☐ **Script Regex Test Mode** *(scriptTestMode)*

☐ **Missing MX Record Test Mode** *(mxaTestMode)*

☐ **Reversed Lookup Test Mode** *(ptrTestMode)*

☐ **Invalid Helo Test Mode** *(ihTestMode)*

☐ **Forged Helo Test Mode** *(fhTestMode)*

☐ **Message Scoring Test Mode** *(msTestMode)*

☐ **DKIM Test Mode** *(dkimTestMode)*

☐ **Penalty Box Test Mode** *(pbTestMode)*

☐ **Switch Testmode to Message Scoring** *(switchTestToScoring)*

Put the filter automatically in "Message Scoring" when **DoPenaltyMessage** is set (instead of stopping spam processing altogether).

Notes On Testmode

Notes

Email Interface for Reports and List Control

☑ **Enable Email Interface** ***(EmailInterfaceOk)***

Checked means that you want ASSP to intercept and parse mail to the following usernames at any localdomains. The domains '@assp.local' and '@assp-nospam.org'are automatically a local domain and can be used for the email-interface.

NOTICE: It is possible to define any MIME-header lines in any report file after the first (subject) line. This makes it possible to define MIME encoding and/or charset settings.
If a definition of MIME encoding and/or charset is found in a report file, assp converts the report from UTF-8 in to the defined encodings. **Don't forget to terminate your MIME-header with an empty line!**

It is also possible to include files at any line of such a file, using the following directive
include filename
where filename is the relative path (from c:/assp) to the included file like reports/mime-header.txt (one file per line). The line will be internaly replaced by the contents of the included file!

Admin Mail Address ***(EmailAdminReportsTo)***

If set internal warnings/infos will be sent to this address. For example: admin@domain.com

Email Interface Reports Destination ***(EmailReportDestination)***

Port to connect to when Email Interface or Block reports are send. If blank they go to the main smtpDestination.
If you need to connect to the **EmailReportDestination** host using native SSL, write 'SSL:' in front of the IP/host definition. In this case the Perl module **IO::Socket::SSL** must be installed and enabled (**useIOSocketSSL**).
eg 10.0.1.3:1025 SSL:10.0.1.3:465, etc.

Authorized Addresses* ***(EmailAdmins)***

Mail from any of these addresses can add/remove to/from redlist, spamlovers, noprocessing, blacklist. May request an **EmailBlockReport** for a list of users. Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com)

Accept Mails (Reports) for these local domains only* ***(EmailInterfaceDomains)***

Enable the EmailInterface and BlockReports for these local domains ONLY (NOT RECOMMENDED). If used, you have also to define '@assp.local' (if required). If not used, all localdomains and '@assp.local' take place (see **EmailInterfaceOk**). Accepts entire domains (@domain.com|domain.com)

Accept Mails (Reports) from these external addresses* ***(EmailSenderOK)***

Allow these external domains/addresses to report to the email interface (NOT RECOMMENDED). The reply address for the reports must be set to a local one. By default, ASSP only accepts reports from local or authenticated users. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com)

Not Authorized Addresses* ***(EmailSenderNotOK)***

Edit report file: reports/denied.txt

Mail from any of these addresses are not accepted from Email Interface, except "Help Report", "Analyze Report" and "Block Report/Resend". Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com). The user will get informed about the denied request.

Ignore Not Authorized Addresses* ***(EmailSenderIgnore)***

Mail from any of these addresses are not accepted from Email Interface. Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com). The user will get not informed about the denied request.

Help Address ***(EmailHelp)***

asshelp | Edit report file: reports/helpreport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request for help. Do not put the full address here, just the user part. For example: asshelp

Report Spam Address ***(EmailSpam)***

asspspam | Edit report file: reports/spamreport.txt

Any mail sent or forwarded by local/authenticated users to this username will be interpreted as a spam report. Multiple attachments get truncated to MaxBytesReports. Do not put the full address here, just the user part.
For example: asspspam . Use a fake domain like @assp.local or @assp-nospam.org when you send the email- so the full address would be then asspspam@assp.local.
You can sent multiple mails as attachments and/or zipped file(s). Each attached email-file must have the extension defined in "**maillogExt**". In this case only the attachments will be processed. To use this multi-attachment-feature an installed **Email::MIME** module in PERL is needed. It is also possible to send MS-outlook '.msg' files (possibly zipped). To use this MS-outlook-feature in addition an installed **Email::Outlook::Message** module in PERL is needed.

Report Ham (Not-Spam) Address ***(EmailHam)***

asspnotspam | Edit report file: reports/notspamreport.txt

Any mail sent or forwarded by local/authenticated users to this username will be interpreted as a false-positive report. Multiple attachments get truncated to MaxBytesReports. Do not put the full address here, just the user part.
For example: asspnotspam . Use a fake domain like @assp.local or @assp-nospam.org when you send the email - so the full address would be

then asspspam@assp.local.
You can sent multiple mails as attachments and/or zipped file(s). Each attached email-file must have the extension defined in "**maillogExt**". In this case only the attachments will be processed. To use this multi-attachment-feature an installed **Email::MIME** module in PERL is needed. It is also possible to send MS-outlook '.msg' files (possibly zipped). To use this MS-outlook-feature in addition an installed **Email::Outlook::Message** module in PERL is needed.

Email Interface Forward Reports Destination ***(EmailForwardReportedTo)***

Host and Port to forward **EmailSpam** and **EmailHam** reports to - eg "10.0.1.3:1025".
If you use more than one assp instance and your users are reporting spam and ham mails to multiple or all of them, but only one (but not this instance) is doing the rebuildspamdb and the corpus folders are not shared between the instances,
define the "host:port" of the central assp (rebuild-) instance here. Every report to **EmailSpam** and **EmailHam** (but only these!) will be forwarded to the defined host(s) and NO other local action will be taken. If the forwarding to all defined hosts fails, the request will be processed locally. To define multiple hosts for failover, separate them by pipe (|).

Reply to Spam/Not-Spam Reports ***(EmailErrorsReply)***

REPLY TO SENDER

Send Copy of Spam/Ham-Reports TO ***(EmailErrorsTo)***

Email sent from ASSP acknowledging your submissions will be sent to this address. For example: admin@domain.com

Combined Spam/Ham Report & Whitelist Check ***(EmailErrorsModifyWhite)***

modify whitelist
If set to 'modify whitelist' Ham Reports will add email addresses to the Whitelist, Spam Reports will remove addresses from the Whitelist, also a copy of a file in the GUI to **correctedspam** (remove) and **correctednotspam** (add) will modify the Whitelist for the found addresses. If set to 'show whitelist' Spam Reports will show if addresses are whitelisted.

Combined Spam Report and NoProcessing Deletion ***(EmailErrorsModifyNoP)***

modify noprocessing
If set to 'modify **noProcessing**' Spam Reports will remove email addresses from **noProcessing** list. If set to 'show **noProcessing**' Spam Reports will show if addresses are on **noProcessing** list, also a copy of a file in the GUI to **correctedspam** (remove) and **correctednotspam** (show) will modify the **noProcessing** list for the found addresses.

Add to Whitelist Address ***(EmailWhitelistAdd)***

asspwhite | Edit report file: reports/whitereport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to add addresses to the whitelist. Do not put the full address here, just the user part.
For example: asspwhite
If an address is added to whitelist, it will be removed from the Personal Blacklist of the sending user.

Remove from Whitelist Address ***(EmailWhitelistRemove)***

asspnotwhite | Edit report file: reports/whiteremovereport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to remove addresses from the whitelist. Do not put the full address here, just the user part.
For example: asspnotwhite

Reply to Add to/Remove from Whitelist ***(EmailWhitelistReply)***

REPLY TO SENDER

Send Copy of Whitelist-Reports TO ***(EmailWhitelistTo)***

Email sent from ASSP acknowledging your submissions will be sent to this address. For example: admin@domain.com

Add to Redlist Address ***(EmailRedlistAdd)***

asspred | Edit report file: reports/redreport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to add the sender address to the redlist. Only the users defined in **EmailRedlistTo**, **EmailAdmins** and **EmailAdminReportsTo** are able to define a list of email addresses in the mail body. Do not put the full address here, just the user part.
For example: asspred.

Remove from Redlist Addresses ***(EmailRedlistRemove)***

asspnotred | Edit report file: reports/redremovereport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to remove the sender address from the redlist. Only the users defined in **EmailRedlistTo**, **EmailAdmins** and **EmailAdminReportsTo** are able to define a list of email addresses in the mail body. Do not put the full address here, just the user part.
For example: asspnotred

Reply to Add to/Remove from Redlist ***(EmailRedlistReply)***

REPLY TO SENDER

Send Copy of Redlist-Reports TO ***(EmailRedlistTo)***

Email sent from ASSP acknowledging your submissions will be sent to this address. For example: admin@domain.com

Add to SpamLover Addresses *(EmailSpamLoverAdd)*

asspspamlover | Edit report file: reports/slreport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to add the sender address to **spamLovers**. Only the users defined in **EmailSpamLoverTo**, **EmailAdmins** and **EmailAdminReportsTo** are able to define a list of email addresses in the mail body. Do not put the full address here, just the user part.
For example: asspspamlover. To use this option, you have to configure **spamLovers** with "file:..." for example "file:files/spamlovers.txt" !

Remove from SpamLover Addresses *(EmailSpamLoverRemove)*

asspnotspamlover | Edit report file: reports/slremovereport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to remove the sender address from **spamLovers**. Only the users defined in **EmailSpamLoverTo**, **EmailAdmins** and **EmailAdminReportsTo** are able to define a list of email addresses in the mail body.
Do not put the full address here, just the user part.
For example: asspnotspamlover

Reply to Add to/Remove from SpamLovers *(EmailSpamLoverReply)*

REPLY TO SENDER

Send Copy of Spamlover-Reports TO *(EmailSpamLoverTo)*

Email sent from ASSP acknowledging your submissions will be sent to this address. For example: admin@domain.com

Add to NoProcessing Addresses *(EmailNoProcessingAdd)*

asspnpadd | Edit report file: reports/npreport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to add the sender address to the **noProcessing** addresses. Only the users defined in **EmailNoProcessingTo**, **EmailAdmins** and **EmailAdminReportsTo** are able to define a list of email addresses in the mail body. Do not put the full address here, just the user part.
For example: asspnpadd. To use this option, you have to configure **noProcessing** with "file:..." for example "file:files/noprocessing.txt" !

Remove from noProcessing Addresses *(EmailNoProcessingRemove)*

asspnprem | Edit report file: reports/npremovereport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to remove the sender address from **noProcessing** .
Do not put the full address here, just the user part. Only the users defined in **EmailNoProcessingTo**, **EmailAdmins** and **EmailAdminReportsTo** are able to define a list of email addresses in the mail body.
For example: asspnprem. To use this option, you have to configure **noProcessing** with "file:..." for example "file:files/noprocessing.txt" !

Reply to Add to/Remove from noProcessing *(EmailNoProcessingReply)*

REPLY TO SENDER

Send Copy of NoProcessing-Reports TO *(EmailNoProcessingTo)*

Email sent from ASSP acknowledging your submissions will be sent to this address. For example: admin@domain.com

Add to BlackListed Addresses *(EmailBlackAdd)*

assp-black | Edit report file: reports/blackreport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to add the sender address to the **blackListedDomains** addresses. Only the users defined in **EmailAdmins** and **EmailAdminReportsTo** are able to request an addition. Do not put the full address here, just the user part.
For example: assp-black. To use this option, you have to configure **blackListedDomains** with "file:..." for example "file:files/blacklisted.txt" !

Remove from BlackListed Addresses *(EmailBlackRemove)*

assp-notblack | Edit report file: reports/blackremovereport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to remove the sender address from **blackListedDomains** .
Do not put the full address here, just the user part. Only the users defined in **EmailAdmins** and **EmailAdminReportsTo** are able to request a removal.
For example: assp-notblack. To use this option, you have to configure **blackListedDomains** with "file:..." for example "file:files/blacklisted.txt" !

Spam/NotSpam Report will modify Personal Blacklist * *(EmailErrorsModifyPersBlack)*

Spam Reports will add email addresses to the Personal Blacklist, NotSpam Reports will remove addresses from the Personal Blacklist, if the report senders address matches.
Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). Wildcards are supported (fribo*@domain.com).
Default is *@* , which matches all addresses.

Add to Personal BlackListed Addresses *(EmailPersBlackAdd)*

assp-persblack | Edit report file: reports/persblackreport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to add the listed address(es) to the personal

blackListed addresses. Do not put the full address here, just the user part.
For example: assp-persblack.
The add and remove is done via email-interface, by sending specific email addresses to '**EmailPersBlackAdd**' and '**EmailPersBlackRemove**'.
A local user can force a complete report about all his personal black list entries by defining an email address that begins with 'reportpersblack' in a remove or add request : eg: reportpersblack@anydomain.com or by sending an empty body.
Any mail address sent to this username will be removed from the whitelist if possible.
Globalized adding an address to all local users is not supported - use **EmailBlackAdd** instead.
The following wildcard combinations are allowed for an email address to support personal blacklisting of domains:

full_sender_address
*@sender_domain or @sender_domain
@*sender_domain or *@*sender_domain
@*.sender_domain or *@*.sender_domain

Remove from Personal BlackListed Addresses *(EmailPersBlackRemove)*

assp-persnotblack | Edit report file: reports/persblackremovereport.txt

Any mail sent by local/authenticated users to this username will be interpreted as a request to remove the listed address(es) from the personal blackListed addresses .
Do not put the full address here, just the user part.
For example: assp-persnotblack.
The add and remove is done via email-interface, by sending specific email addresses to '**EmailPersBlackAdd**' and '**EmailPersBlackRemove**'.
A local user can force a complete report about all his personal black list entries by defining an email address that begins with 'reportpersblack' in a remove or add request : eg: reportpersblack@anydomain.com or by sending an empty body.
Only an admin can force a complete cleanup of all personal black entries for a specific email address for all local users - sending an email to '**EmailPersBlackRemove**' with the address followed by ',*' in the body eg: address_to_remove@the_domain.foo,* - be careful modifying personal entries of other users!
The same wildcard combinations like in **EmailPersBlackAdd** are supported.
Notice: a remove request for a specific email address will remove ALL entries from the users personal blacklist, that would block this email address (also all matching wildcard entries)!

Reply to Add to/Remove from BlackListed *(EmailBlackReply)*

REPLY TO SENDER

Send Copy of Black-Change-Reports TO *(EmailBlackTo)*

Email sent from ASSP acknowledging your submissions will be sent to this address. For example: admin@domain.com

Request Analyze Report *(EmailAnalyze)*

asspanalyze | Edit report file: reports/analyzereport.txt

Any mail sent or forwarded by local/authenticated users to this username will be interpreted as a request for analyzing the mail. Do not put the full address here, just the user part. For example: asspanalyze
Use a fake domain like @assp.local or @assp-nospam.org when you send the email- so the full address would be then asspanalyze@assp.local.
You can sent multiple mails as attachments and/or zipped file(s). Each attached email-file must have the extension defined in "**maillogExt**". In this case only the attachments will be processed. To use this multi-attachment-feature an installed **Email::MIME** module in PERL is needed. It is also possible to send MS-outlook '.msg' files (possibly zipped). To use this MS-outlook-feature in addition an installed **Email::Outlook::Message** module in PERL is needed.

Reply to Analyze Request *(EmailAnalyzeReply)*

SEND TO SENDER

Send Copy of Analyze-Reports *(EmailAnalyzeTo)*

A copy of the Analyze-Report will be sent to this address. For example: admin@domain.com

Spam and Ham Reports will trigger an additional Analyze Report *(DoAdditionalAnalyze)*

NO ADDITIONAL REPORT
Additional Analyze Report will be generated for Spam and Ham Reports. Setting the TO Address accordingly and choosing **EmailAnalyzeTo** will send the Analyze Report to the admin only.

From Address for Reports *(EmailFrom)*

<spammaster@yourdomain.com>
Email sent from ASSP acknowledging your submissions will be sent from this address.

☑ **Allow '=' in Addresses *(EmailAllowEqual)***
Allow '=' in addresses to be whitelisted or redlisted.

Do Not Reply To These Addresses* *(EmailSenderNoReply)*

Email sent from ASSP acknowledging your submissions will not be sent to these addresses. Accepts specific addresses (user@example.com), user parts (user) or entire domains (@example.com).
Analyze-, PersonalBlackList- and all virus related reports are ignored by this feature (are sent even a user is listed here).
A Report copy to **EmailAnalyzeTo**, **EmailBlackTo**, **EmailNoProcessingTo**, **EmailSpamLoverTo**, **EmailRedlistTo**, **EmailWhitelistTo** and **EmailErrorsTo** is also ignored by this feature.

Notes On Email Interface
Notes

File Paths and Database

Directory Base *(base)*

c:/assp

All paths are relative to this folder.
Note: Display only.

Spam Collection *(spamlog)*

spam

The folder to save the collection of spam mails. This directory will be used in building the **spamdb** . For example: spam

Not-spam Collection *(notspamlog)*

notspam

The folder to save the collection of not-spam mails. This directory will be used in building the **spamdb** . For example: notspam

OK Mail *(incomingOkMail)*

okmail

The folder to save non-spam (message ok). These are messages which are considered as HAM, but are not stored in the standard HAM folder because of our policy to use only confirmed HAM messages (whitelisted or local) for **spamdb** . If you want to keep copies of ok mail then put in a directory name. This directory will not be used in building the **spamdb** . Default: okmail

Discarded Spam *(discarded)*

discarded

The folder to save **discarded** spam-messages. These are Spam messages which are not stored for building the **spamdb** but for resending with an **EmailBlockReport**. If you want to keep copies of **discarded** Spam then put in a directory name. Default: **discarded**

Attachment/Virus Collection *(viruslog)*

quarantine

The folder to save rejected attachments and viruses. Leave this blank to not save these files (default). If you want to keep copies of rejected content then put in a directory name. Note: you must create the directory. This directory will not be used in building the **spamdb** . For example: quarantine

False-negative Collection *(correctedspam)*

errors/spam

Spam that got through -- counts double. This directory will be used in building the **spamdb** . For example: errors/spam

False-positive Collection *(correctednotspam)*

errors/notspam

Good mail that was listed as spam, count 4x. This directory will be used in building the **spamdb** . For example: errors/notspam

try to resend this files *(resendmail)*

resendmail

ASSP will try to resend the files in this directory to the original recipient. The files must have the "**maillogExt**" extension and must have the SMTP-format. For example: **resendmail**. This requires an installed **Email::Send** module in PERL.

Extension for Mail Files *(maillogExt)*

.eml

Enter the file extension (include the period) you want appended to the mail files in the mail collections.
Leave it blank for no extension - this setting will prevent several features from working. Never use '.msg' - this is an extension used by MS-outlook! For Example: .eml

Spam/HMM Bayesian Database Files *(spamdb)*

spamdb

The output file from rebuildspamdb. Write only "DB:" to use a database table instead of a local file, in this case you need to edit the database parameters below. The Hidden Makov Model is only available if this parameter is set to DB: .

It is recommended to use a database for all possible lists and caches for best performance, less memoryusage and stability! If you do not want to install a database engine like MySql or Oracle, use BerkeleyDB! Please read the section **DBdriver** !
If you set this value to "DB:" and you want **HMMdb** to use the same database backend like **spamdb**, don't forget to disable **HMMusesBDB** !

Last Run Rebuildspamdb

Last Run Rebuildspamdb

Email Whitelist Database File *(whitelistdb)*

whitelist

The file with the whitelist.
Write only "DB:" to use a database table instead of a local file, in this case you need to edit the database parameters below.

Email Redlist Database File *(redlistdb)*

redlist

The file with the redlist.
Write only "DB:" to use a database table instead of a local file, in this case you need to edit the database parameters below.

Personal Blacklist Database File *(persblackdb)*

persblack

The file with the personal blacklist. The check of the personal black list is done shortly after the RCPT TO: command. This command will be rejected if an entry is found - any other setting except **send250OK** and **send250OKISP** will be ignored.
Each entry is represented by two comma separated values TO,FROM (and an expiration date).

TO could be any of : email address, [subdomain.]domain.tld, @[subdomain.]domain.tld, *@[subdomain.]domain.tld - the last three entry options could be only added and removed by editing the list in the GUI !
FROM could be any of : email address or any [@][subdomain.][domain.]TLD variant (wildcards are allowed). All values are supported by the email interface for all local users.
Write only "DB:" to use a database table instead of a local file, in this case you need to edit the database parameters below.

LDAP Database ***(ldaplistdb)***

ldaplist

The file with the LDAP-cache database. Local email addresses and local domains, which are successfully validated using LDAP (**DoLDAP**) or VRFY (**DoVRFY**) are cached in this list, to prevent repeated LDAP and/or VRFY lookups.
Write only "DB:" to use a database table instead of a local file, in this case you need to edit the database parameters below.
If an email address or domain is reported as invalid by LDAP or VRFY, this result is cached for (min) ten to (max) fifteen minutes in the LDAP-Not-Found-Cache edit LDAP-Not-Found Cache , to prevent repeated LDAP and/or VRFY lookups. Those cache entries are not removed from the cache, while the rebuildspamdb task is running!

Delaying Database ***(delaydb)***

delaydb

The file with the delay database.
Write only "DB:" to use a database table instead of a local file, in this case you need to edit the database parameters below.

PenaltyBox Database ***(pbdb)***

pb/pbdb

The directory/file with the penaltybox database files. For removal of entries from BlackBox (PBBlack) use **noPB** . For removal of entries from WhiteBox (PBWhite) use **noPBwhite**. For whitelisting IP's use **whiteListedIPs** or **noProcessingIPs** . For blacklisting use **denySMTPConnectionsFrom** and **denySMTPConnectionsFromAlways** .
Write only "DB:" to use a database table instead of a local file.
Show BlackBox Show White Box

Admin Users Database ***(adminusersdb)***

The file with the GUI-Admin-Users database - default to set is 'adminusers'.
Write only "DB:" to use a database table instead of a local file, in this case you need to edit the database parameters below. Before setting this parameter, please set **adminusersdbpass** to a value of your choice!

To use this database shared between multiple ASSP's, set all ASSP to **mysqlSlaveMode** (except the master) and the **adminusersdbpass** must be the same on all installations! If you want to change the **adminusersdbpass**, first change it on the master.

☐ **Admin Users Database uses no Binary Data (ASCII only)** ***(adminusersdbNoBIN)***

Select this, if **adminusersdb** is set to "DB:" and your database engine does not accept or has problems with binary data (eg. Postgres). If you change this value, you have to stop all assp and to cleanup both tables (adminusers and adminusersright) **before** restarting assp!. To keep your data do the following: do an **ExportMysqlDB** - change this value - stop assp - drop or clean both tables - start assp - do an **ImportMysqlDB** .

Admin Users Database PassPhrase ***(adminusersdbpass)***

45WPcXBk5dhLo

The passphrase that is used to encrypt the **adminusersdb**. This has to be the same on all ASSP installations that are sharing the **adminusersdb**. If you want to change it, first change it on the master installation and than on the slaves. Do not forget to configure '**mysqlSlaveMode**' first. An empty value is not valid!

GreyIPlist Database ***(griplist)***

griplist Show file

The file with the current Grey-IP-List database -- make this blank if you don't use it.

☐ **Use BerkeleyDB for Griplist** ***(useDB4griplist)***

If selected ASSP uses 'BerkeleyDB' instead of 'orderedtie' for **griplist**. Depending on your settings for **OrderedTieHashTableSize** this could spend some memory and/or result in better performance. The perl module **BerkeleyDB** version 0.34 or higher and BerkeleyDB version 4.5 or higher is required to use this feature.

database hostname or IP ***(myhost)***

You need **Tie::RDBM** to use a database instead of local files.
This way you can share whitelist, **delaydb**, redlist, **spamdb**, **HMMdb**, ldaplist, **adminusersdb**, personal blacklist and penaltybox (all that can be set to 'DB:') between servers.
ASSP uses permanent connections to the database server, so set the connection timeout in your server configuration to a very high value (eg - mysql: interactive_timeout=28800 , wait_timeout=28800).

database driver name ***(DBdriver)***

mysql

The database driver used to access your database - DBD-driver.
Please read this section very carefull!
The following drivers are available on your system:
BerkeleyDB, ADO, AnyData, CSV, DBM, ExampleP, File, Firebird, Gofer, LDAP, Log, MVS_FTPSQL, Mock, Multiplex, ODBC, Oracle, Ovrimos, Pg, PgPP, Proxy, SQLite, Sponge, Sprite, Template, TemplateSS, mysql, mysqlPP
If you can not find the driver for your database in this list, you should install it via cpan or ppm!
- or if you have installed an ODBC-driver for your database and DBD-ODBC, just create a DSN and use ODBC.
If assp is running on windows and you want to use a MSSQL server as backend, don't use the ODBC driver - use the ADO driver with the DSN definition!
Useful are ADO|DB2|Informix|ODBC|Oracle|Pg|Sybase|mysql|Firebird - but any other SQL compatible database should also work.

syntax examples: driver,option1,option2,...,...

ADO[,DSN=mydsn[;Provider=sqloledb]]
ODBC,DSN=mydsn|driver=\{SQL Server\},Server=server_name
DB2
Informix
Oracle,SID=1|INSTANCE_NAME=myinstance|SERVER=myserver|SERVICE_NAME=myservice_name,[PORT=myport]
Pg[,PORT=myport]
Firebird[,PORT=myport]
Sybase,SERVER=myserver,[PORT=myport]
mysql[,PORT=myport][,mysql_socket=/path/to/mysql.sock][,AutoCommit=1][,mysql_auto_reconnect=1]

Notice: assp requires permanent database connections. Set database engine parameter like 'client-timeout' or 'connection-timeout' to very high values (eg: 1/2 or 1 day)! ASSP requires one database connection per thread (typical 8 connections), plus up to five connection for imports, exports and internal processing. Set the maximum allowed database connection in your database server configuration according!

Instead using local files for hashes and lists via shared memory, it is recommended to use **BerkeleyDB** (Perl-module) version 0.34 or higher for highest performance and less memory usage. The BerkeleyDB (engine) version 4.5 or higher is required to use BerkeleyDB.
If you specify BerkeleyDB here, the values for **myhost**, **mydb**, **myuser** and **mypassword** will be ignored. All possible BerkeleyDB option must be defined here - the option for '-Filename' is already set by ASSP! Options could be defined for example:
BerkeleyDB,-Pagesize=>number,-Env=>[-Cachesize=>number,-Mode=>mode,...,...],...,...
If '-Env=>[-Cachesize=>number]' (number in bytes) is specified, this cache size will be used at minimum for every single list. Setting the cache size is not recommended (as long as do not you really know what you do), because ASSP does automatically calculate the right cache for every list. You may setup configuration values for any BerkeleyDB, creating a file **DB_CONFIG** (case sensitive) in the corresponding directory ./tmpDB/[list]. Please use the BerkeleyDB documentation if you don't know the syntax of this file. Any value defined in that file will overwrite the corresponding internal ASSP configuration for this DB.
As with each other database engine, you should know how to handle BerkeleyDB large shared BDB-environments (CDB - DB_INIT_CDB and DB_INIT_MPOOL), how to repair database files and all the other important stuff. ASSP has several buildin mechanism to detect and repair corrupt BerkeleyDB files, but they may not work in every case!
If you have specified BerkeleyDB here and your system shows unexpected SEGV or ASSP died unexpected, think about the BDB settings. If you can't fix such an issue, it may be an good idea to switch over to MySQL or another database engine.
KEEP IN MIND: BerkeleyDB files are shared opened and accessed by all threads using BDB-CDB. The last terminated thread closes the BDB-files (shutdown the BDB-engine) for the systems file system. It is important, that (especially) linux and unix system shutdown scripts are waiting until ALL assp/perl processes are ended (this may take up to one minute - see **MaxFinConWaitTime**)! Otherwise, the kernel will kill the assp/perl process at shutdown and the BerkeleyDB DB-files and environment-files WILL BE DESTROYED and cause to 100% unexpected behavior or crashes at the next start or run! The same applies to Windows systems, if assp is not running as system service - the windows system-service-manager will wait until the process is finished.

The options for all drivers and their possible or required order depends on the DBD driver used, please read the driver's documentation, if you do not know the needed option.
The username, password, host and databasename are always used from this configuration page.

database name *(mydb)*

This database must exist before starting ASSP, necessary tables will be created automatically into this database.

☐ **This is a slave of more then one assp-computers accessing the same database *(mysqlSlaveMode)***

If you are running more then one assp-computers accessing the same or **(better because of SPOF)** a bidirectional replicated database this is a slave-assp and no database maintenance will be done by this one!
Maintenance should only be done by the first assp - the master!
Maintenance for file based caches and lists will always be done!

database username *(myuser)*

This user must have CREATE privilege on database to create tables automatically

database password *(mypassword)*

Database Maximum Cache Age *(DBCacheMaxAge)*

0

Setting this value above zero, enables an internal database cache for every defined table to reduce the concurrent database queries and to prevent possible record access collisions, which could cause stuck workers on some systems
The value defines the maximum age in seconds a record will exists untouched in the table cache.
Be careful, setting this value too high in a database replication environment could cause unexpected query results, because this cache is NOT shared between multiple assp instances.
If set, a value of 10 seems to be popular in any case. A value that is too small will produce overhead without any advantage. A value that is too high may cause database consistency problems.

import directory *(importDBDir)*

mysql/dbimport

The folder to import the used tables of the database from.
The schema of the files must be the assp-schema.
Files can be:
- pbdb.back.db.(add|rpl)
- pbdb.batv.db.(add|rpl)
- pbdb.black.db.(add|rpl)
- pbdb.dkim.db.(add|rpl)
- pbdb.mxa.db.(add|rpl)
- pbdb.ptr.db(add|rpl)
- pbdb.rbl.db.(add|rpl)
- pbdb.rwl.db.(add|rpl)
- pbdb.sb.db.(add|rpl)
- pbdb.spf.db.(add|rpl)
- pbdb.trap.db.(add|rpl)
- pbdb.uribl.db.(add|rpl)
- pbdb.white.db.(add|rpl)
- ldaplist.(add|rpl)

- redlist.(add|rpl)
- whitelist.(add|rpl)
- persblackdb.(add|rpl)
- spamdb.(add|rpl)
- spamdb.helo.(add|rpl)
- delaydb.(add|rpl)
- delaydb.white.(add|rpl)
- adminusers.(add|rpl)
- adminusersright.(add|rpl)

Use the extension "add" or "rpl" to add or replace the records to the tables.
Only files for database-enabled tables will be imported ! The import will be done at ASSP start or if the option below is used.
Imported files will be renamed to *.OK !
For example: mysql/dbimport
If you plan to import in to BerkeleyDB - do the following:
- set **DisableSMTPNetworking** to on - set all needed DB parameters - collect your import files - restart assp and wait until all imports are finished - restart assp - set **DisableSMTPNetworking** to off

☐ **Prevent Bulk Import** ***(preventBulkImport)***

Do not select, if you are using MySQL! Doing a Bulk-Import of data, ASSP modifies the properties of table columns. This could result in breaking some configured DB features like DB-replication in MSSQL and possibly other database engines. If selected, ASSP will do a line per line insert/update (which takes much more time) without modifying the tables properties.

Fill the Import Folder ***(fillUpImportDBDir)***

[ ]

If set to a value between 1 and 9, the corresponding backup file for any list/hash that configured to use a database will be copied from the **backupDBDir** to the **importDBDir**. The resulting file name will has an extension of ".rpl", so a possible import will replace the current table content. If a value of "L" is defined, the last backup will be used. Possible values are L or 1 - 9 or blank. Any configured value will be reset to blank after the copy is finished.

☐ **import all files from the importDBDir Directory in to the database - now.** ***(ImportMysqlDB)***

All files from the "**importDBDir**" will be imported in to database . Please define the directory above, before using the import!
[Apply Changes and Run DB Import Now (if checked)] [Refresh Browser]

export directory ***(exportDBDir)***

[mysql/dbexport]
The folder to export the used tables of the database.
The schema of the files is the assp-schema.(starts with [LF], one record per line, lines are terminated with [LF], keys and values are separated by \002 or \x02)
Ten versions of the database and hash exports are available!
Exports are done unencrypted, it is recommended to secure this folder!
For example: mysql/dbexport

☐ **export all tables from the database and plain hash files** ***(ExportMysqlDB)***

All tables of the database and all plain hash files will be exported to the "**exportDBDir**" directory. Please define the directory above, before using the export function!
In addition the running configuration and all encrypted option files in use will be exported.
If you plan to upgrade the OS or perl, or you plan to move to a new system or a different OS - it is recommended to do an export first!
NOTICE: both encrypted tables/hashes, AdminUsersRight and AdminUsers, will be exported unencrypted (eg. in plain text), the same applies to the exported configuration file and the exported option files!
If possible, assp will compress the config files, option files and the AdminUsersRight and AdminUsers to the file 'config.zip' in the "**exportDBDir**" directory.
If possible, assp will encrypt the config.zip to config.zip.aes using openssl or Crypt::CBC. To decrypt this file, use the OS commandline:

openssl enc -d -aes-256-cbc -in config.zip -out config.zip.aes -pass pass:PASSWORD

NOTICE: The password / key, used for the export encryption function, may change at the next assp start or if the assp.cfg gets an external update! Record the password after each export!

[Apply Changes and Run the Export NOW (if checked)] [Refresh Browser] [show the decryption password]

backup directory ***(backupDBDir)***

[mysql/dbbackup]
The folder to backup the used tables of the database.
The schema of the files is the assp-schema.
Ten versions of backups are available!
For example: mysql/dbbackup

backup database Interval [s] ***(backupDBInterval)*** 

[2]
backup the database (all tables used by assp at the time) every this hours.
Defaults to 2 hours.

☑ **copy the last DB-backup to the original location** ***(copyDBToOrgLoc)***

If DB-backup is enabled, the last backupversion is also copied to the original location.
If database connections are failed, while ASSP is running, ASSP will switch over to use these files instead of DB-tables.
DB-tables will not be imported from here, this must be done from the **importDBDir**!

ASSP Logfile ***(logfile)***

[logs/maillog.txt]
Blank if you don't want a log file. Change it to maillog.log if you don't want auto rollover. NOTE: Changing this field requires restarting ASSP before changes take effect.

Max Age of Logfiles *(MaxLogAge)*

0

The maximum file age in days of logfiles. If a **logfile** is older than this number in days, the file will be deleted. Default is 0 - recommended is 30. A value of 0 disables this feature and no **logfile** will be deleted because of its age.

Runtime MaxLogAge [s] *(MaxLogAgeSchedule)*

1

Runtime hour for deleting old logfiles. Set a number between 0 and 23. 0 means midnight, 1 is default.

PID File *(pidfile)*

pid

Blank is not a valid value!
You have to restart ASSP before you get a pid file in the new location.
This file is used to detect a clean shutdown of ASSP - in this case it does not exist at startup!

Notes On File Path

Notes

Collecting SPAM and HAM

Spam Collect Addresses* ***(spamaddresses)***

Mail to any of these addresses are always spam and will contribute to the spam-collection unless from someone on the whitelist - for example honeypott addresses. Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com). The addresses are not validated, they are readdressed to ccallspam, however you can supersede this by putting a valid address into **sendAllCollect** below.

Catchall Address for Collect Addresses ***(sendAllCollect)***

ASSP will readdress messages addressed to Collect Addresses to this address.
For example: collect@mydomain.com

☐ **Use Collect Addresses for Testing Your Environment** ***(DoNotBlockCollect)***

If set ASSP will block messages from Collect Addresses **after** other checks are performed. That may help to test and control activated filters.

☐ **Use Penalty Trap Addresses To Collect** ***(UseTrapToCollect)***

If set ASSP will use addresses from **DoPenaltyMakeTraps** and **spamtrapaddresses** to collect spams.

Do Not Collect Messages from/to these Addresses* ***(noCollecting)***

Accepts specific addresses (user@domain.com), user parts (user) or entire domains (@domain.com).

Do Not Collect Messages - Content Based* ***(noCollectRe)***

If the content of a collected file (incl. X-ASSP-... headers) matches this regular expression, it will be deleted from the collection after the mail is completely processed.
If the ASSP_ARC plugin is used, the file will be deleted from the collection after it was archived. This is the only "no collect" option which removes an already collected file, all other options will prevent assp from creating a collection file - if set to "no collection". The check is limited to **MaxBytes** or at max 100000 Bytes.

☑ **Do Not Collect Red-Re Matching Mails** ***(DoNotCollectRedRe)***

Mails (Spam/Ham) matching Red Regex (**redRe**) will not be stored in the collection folders.

☑ **Do Not Collect Redlisted Mails** ***(DoNotCollectRedList)***

Mails (Spam/Ham) matching Redlist will not be stored in the collection folders.

☑ **Do Not Collect Bounced Mails** ***(DoNotCollectBounces)***

Mails matching <Bounce Senders> will not be collected.

☐ **Do Not Collect Mail** ***(NoMaillog)***

Check this if you're using Whitelist-Only and don't care to save mail to build the Bayesian database.

Max Files ***(MaxFiles)***

14000

If you're not using subjects as file names (**UseSubjectsAsMaillogNames**), this is the maximum number of files to keep in each collection (spam & nonspam)
It's actually less than this -- files get a random number between 1 and **MaxFiles**.

Files Distribution ***(FilesDistribution)***

1

This defines how file names are chosen in each collection. If set to 1, names are uniformly distributed. If set between 0.01 and 0.99, names distribution is exponential -- files get lower numbers more frequently. This prevents from corpus being refreshed too quickly, especially when **MaxFiles** is set to low value (ex. 3000). This setting is ignored if **UseSubjectsAsMaillogNames** is set to ON, which highly recommended.
Recommended: 0.5, Default: 1

☑ **Use Subject as Maillog Names** ***(UseSubjectsAsMaillogNames)***

You can turn this on to help you manually identify mail in your spam and non-spam collections. This will prevent ASSP from controlling the number of files in your collections(-> **MaxFiles**). It is recommended to switch on **MaintBayesCollection** and to setup **MaxNoBayesFileAge** to your needs, if you have switched on this option.

Max Number of Duplicate File Names ***(MaxAllowedDups)***

50

The maximum number of logged files with the same filename (subject) that are stored in the spam folder (**spamlog**), if **UseSubjectsAsMaillogNames** is selected. Default is 0. A low value reduces the number of possibly duplicate mails, assuming that mails with the same subject will have the same content. A value of 0 disables this feature. If this number of files with the same filename is reached, the oldest file with the same subject will be moved to the **discarded** folder, which has to be defined (in addition to **spamlog**) for this feature to work.

Regular Expression to Identify allowed duplicate Subjects* ***(AllowedDupSubjectRe)***

Messages their subject matches this regular expression will be collected regardless the setting in **MaxAllowedDups** .

☐ **Use Unicode to build Maillog Names** ***(UseUnicode4MaillogNames)***

If you have switched on **UseSubjectsAsMaillogNames** and your default (local language) characterset (please setup **ConsoleCharset**) needs 8 Bit like "KOI8-r","CP-866","Windows-1251","Windows-1252","ISO-8859-X","X-Mac-Cyrillic","JIS_X0201" or any other (or is UTF-8) - and you

want to have readable filenames in the maillog and on the console screen, you can switch on this option. The resolution of some characters written to the console could be incorrect depending on your operating system. This requires an installed **Email::MIME** module in PERL.
If in addition the module **Win32::Unicode** is installed on windows platforms, assp will generate unicode filenames for the collected corpus files (already on nix systems).

☐ **Use Unicode to build Subjects in Maillog** ***(UseUnicode4SubjectLogging)***

If you have switched on **UseUnicode4SubjectLogging** and your default (local language) characterset (please setup **ConsoleCharset**) needs 8 Bit like "KOI8-r","CP-866","Windows-1251","Windows-1252","ISO-8859-X","X-Mac-Cyrillic","JIS_X0201" or any other (or is UTF-8) - and you want to have a readable subject in the maillog and on the console screen, you can switch on this option. The resolution of some characters written to the console could be incorrect depending on your operating system. This requires an installed **Email::MIME** module in PERL.

Max Length of File Names ***(MaxFileNameLength)***

50

The maximum character count that is used from the mail subject to build the file name of the logged file, if **UseSubjectsAsMaillogNames** is selected. This could be useful, if your mail clients having trouble to build the resend file name (right button - URL) correctly in block reports. Every non printable character will be replaced by a 4 byte string in this link.

☑ **Maintenance for Bayesian Collection** ***(MaintBayesCollection)***

Set this to on, if you want ASSP to run a maintenance tasks on the bayesian collection folders (**spamlog** , **notspamlog** , **correctedspam** , **correctednotspam**). ASSP will delete the oldest files until the number of files per folder reaches **MaxFiles**. If you want ASSP to delete files because of their age instead of the number of files (**MaxFiles**), setup **MaxBayesFileAge** and/or **MaxCorrectedDays** to your needs. **MaintBayesCollection** is useful, if **UseSubjectsAsMaillogNames** is set to on and **doMove2Num** is set to off, because in this case the number of files in every collection folder will grow infinite. If set to On, the rebuildspamdb task will also do the cleanup.

Max Age of Bayes Files ***(MaxBayesFileAge)***

31

The maximum file age in days of every file in every bayesian collection folder (**spamlog** , **notspamlog**). If **MaintBayesCollection** is set to on and a file is older than this number in days, the file will be deleted. Default is 31. A value of 0 disables this feature and no file will be deleted because of its age. To use different values for **spamlog** and **notspamlog**, define two space separated values - the first for **spamlog** and the second for **notspamlog**, like '30 60'. The rebuildspamdb task will ignore files older than this days (if not zero).
It is not recommended to enable this option, if you use the bayesian engine of ASSP and **doMove2Num** is set to ON. A better solution in this case is, to have **MaintBayesCollection** take care of deletions (by date) and change this setting to 0.

Max Corrected File Age ***(MaxCorrectedDays)***

10000

This is the number of days a error report will be kept in the **correctedspam** and **correctednotspam** folders. These folders are the longterm memory of ASSP, therefore the default is 10000 days (more than 27 years). To use different values for **correctedspam** and **correctednotspam**, define two space separated values - the first for **correctedspam** and the second for **correctednotspam**, like '1000 1500'. The rebuildspamdb task will ignore files older than this days (if not set to zero).

Max Age of non Bayes Files ***(MaxNoBayesFileAge)***

31

The maximum file age in days of every file in every non bayesian collection folder (**incomingOkMail** , **discarded** , **viruslog**). If defined and a file is older than this number in days, the file will be deleted. Default is 31. A value of 0 disables this feature and no file will be deleted because of its age. To use different values for **incomingOkMail** and **discarded** and **viruslog**, define three space separated values - the first for **incomingOkMail** and the second for **discarded** and the third for **viruslog**, like '31 45 60'

Runtime for MaintBayesCollection and MaxNoBayesFileAge [s] ***(MaxFileAgeSchedule)***

1

Runtime hour for deleting old collected files (bayes and non bayes). Set a number between 0 and 23. 0 means midnight, 1 is default. If empty a cleanup will not be scheduled. This could be fine, if a rebuildspamdb is scheduled, which will also do the cleanup based on the settings of **MaintBayesCollection** , **MaxBayesFileAge** and **MaxCorrectedDays** - but it will not maintain **incomingOkMail** , **discarded** and **viruslog** based on **MaxNoBayesFileAge** !

Max Bytes ***(MaxBytes)***

4000

How many bytes of the message body will ASSP look at - the message header is always included in all checks? Mails stored in the collecting folders will be truncated to this size, if **StoreCompleteMail** is disabled. The average of Ham messages (message body) is 6K, the average of Spam messages is 3K. Usually the spam folder will be filled quicker than the notspam folder, therefore set this value to 4000 to get more wordpairs per Ham Message. When both folders are close to the maxfiles limit, reduce it to 3000.

Store the Complete Mail ***(StoreCompleteMail)***

no limit

If set, ASSP will look at **MaxBytes**, but if possible it will store the complete mail up to the number of bytes configured. This could be useful for example, if you want resend blocked messages. Be careful using this option, your disk could be filled up very fast!

Non Spam ***(NonSpamLog)***

notspam folder

Where to store whitelisted/local non spam messages. Default: notspam folder (**notspamlog**).

OK Mail ***(baysNonSpamLog)***

no collection

Where to store non spam (message ok) messages. These are messages which are considered as HAM, but should not stored in the standard HAM folder because of our policy to use only confirmed HAM messages (whitelisted or local) for SpamDB. Set **incomingOkMail** accordingly if you choose 'okmail folder'. Default: no collection

Store Spam ***(SpamLog)***

enabled

Set this to 'disabled' if you do not want to store any Spam regardless of settings in. Default: enabled (store in folder **spamlog**).

NoProcessing OK Mails ***(noProcessingLog)***

no collection

Where to store noprocessing OK mails.

NoProcessing rejected Attachments ***(npAttachLog)***

discard folder & sendAllSpam

Where to store noprocessing rejected mail+attachments. Recommended: discard folder (**discarded**) & **sendAllSpam**

Whitelisted rejected Attachments ***(wlAttachLog)***

discard folder & sendAllSpam

Where to store whitelisted rejected mail+attachments. Recommended: discard folder (**discarded**) & **sendAllSpam**

External rejected Attachments ***(extAttachLog)***

discard folder & sendAllSpam

Where to store external rejected mail+attachments. Recommended: discard folder (**discarded**) & **sendAllSpam**

Virus Infected ***(SpamVirusLog)***

quarantine

Where to store virus infected messages. Recommended: quarantine (quarantine)

Spam Bombs ***(spamBombLog)***

discard folder

Where to store spam bombs. Recommended: discard folder (**discarded**)

Scripts ***(scriptLog)***

spam folder & sendAllSpam

Where to store scripted messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

Blacklisted Domains ***(blDomainLog)***

spam folder & sendAllSpam

Where to store blacklisted domain messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

Blacklisted Helos ***(spamHeloLog)***

discard folder & sendAllSpam

Where to store spam helo messages. Recommended: discard folder (**discarded**) & **sendAllSpam**

Forged Helos ***(forgedHeloLog)***

no collection

Where to store forged helo messages. Recommended: no collection

Invalid Helos ***(invalidHeloLog)***

discard folder

Where to store invalid helo messages. Recommended: discard folder (**discarded**)

Spam Collect Addresses ***(spamBucketLog)***

spam folder

Where to store mails addressed to Spam Collect Addresses. Recommended: spam folder (**spamlog**)

Bayesian Spams ***(baysSpamLog)***

discard folder & sendAllSpam

Where to store Bayesian spam messages. Recommended: discard folder (**discarded**) & **sendAllSpam**

SPF Failures ***(SPFFailLog)***

spam folder & sendAllSpam

Where to store SPF Failure spam messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

DNSBL Failures ***(RBLFailLog)***

spam folder & sendAllSpam

Where to store DNSBL Failure spam messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

URIBL Failures ***(URIBLFailLog)***

spam folder & sendAllSpam

Where to store URIBL Failure spam messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

SRS Failures ***(SRSFailLog)***

spam folder & sendAllSpam

Where to store SRS Failure (not signed bounces) spam messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

Missing/Invalid Pointer ***(spamPTRLog)***

spam folder & sendAllSpam

Where to store Missing/Invalid Pointer rejected messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

Missing MX Record ***(spamMXALog)***

spam folder & sendAllSpam

Where to store Missing MX record rejected messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

Invalid Local Sender *(spamISLog)*

no collection

Where to store messages from a local domain with an unknown userpart. Recommended: no collection

Blocked Country *(spamSBLog)*

spam folder & sendAllSpam

Where to store messages from a blocked country. Recommended: spam folder (**spamlog**) & **sendAllSpam**

Message Limit Blocks *(spamMSLog)*

spam folder & sendAllSpam

Where to store Message Scoring Limit rejected messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

PenaltyBox Blocks *(spamPBLog)*

spam folder & sendAllSpam

Where to store PB rejected messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

DKIM failed *(DKIMLog)*

spam folder & sendAllSpam

Where to store DKIM rejected messages. Recommended: spam folder (**spamlog**) & **sendAllSpam**

Backscatter check failed *(BackLog)*

discard folder

Where to store backscatter (MSGID-signing, BATV, DNS-Backscatter) rejected messages. Recommended: no collection

Non Spam Collection Frequency *(freqNonSpam)*

1

Store every n'th non spam message. If you set the value to 10 then every 10th message is logged. These frequency settings are for ASSP users with a mature installation who experience heavy mail or spam volumes. Enter a larger value if the non spam corpus is being refreshed too quickly. Default Value = 1, log every message. Leave it at the default value 1, if you use BlockReports.

Spam Collection Frequency *(freqSpam)*

1

Store every n'th spam message. The same as for non spam but helps prevent spam corpuses being skewed by flooding. It is recommended that this be set depending on spam volume. Default value = 1, log every message. Leave it at the default value 1, if you use BlockReports.

Notes On Collecting

Notes

Logging and Notifications

Notification Email To *(Notify)*

Email address(es) to which you want ASSP to send a notification email per default, if a matching log entry (**NotifyRe** , **NoNotifyRe**) is found.
Separate multiple entries by comma ",".
NOTICE: that groups are not allowed to be used here!

Do Notify, if log entry matches* *(NotifyRe)*

Regular Expression to identify loglines for which a notification message should be send.
useful entries are:
Info: new assp version - to get informed about new available assp versions
info: autoupdate: new assp version - to get informed about an autoupdate of the running script
adminupdate: - for config changes
admininfo: - for admin information
option list file: - for option file reload
error: - for any error
warning: - for any warning
restart - to detect a ASSP restart
notification: too many recipients - for local frequency abuse once per day and sender
warning: too many recipients - for every local frequency abuse
MainThread started - to detect a start of ASSP
Admin connection - for GUI logon

You may define a comma separated list (after '=>') of recipients in every line, this will override the default recipient defined in 'Notify'.
for example: adminupdate:=>user1@yourdomain.com,user2@yourdomain.com.
NOTICE: that groups are not allowed to be used for the second parameter!
As third parameter after a second ('=>') you can define the subject line for the notification message.
for example: adminupdate:=>user1@yourdomain.com,user2@yourdomain.com=>configuration was changed
or: adminupdate:=>=>configuration was changed.

Do NOT Notify, if log entry matches* *(NoNotifyRe)*

Regular Expression to identify loglines for which no notification message should be send.
for example:
user root - if root does anything
\[root.*?\] - if root changes the config

☑ **File name logging** *(fileLogging)*
Show file names of collected spam/notspam in log. Will be automatically set to on, if **inclResendLink** is not set to disabled.

☑ **Subject logging** *(subjectLogging)*
Show subject of mail in log

Subject Start Delimiter *(subjectStart)*

[
Start delimiter of subject in log

Subject End Delimiter *(subjectEnd)*

]
End delimiter of subject in log

☑ **Regex Match logging** *(regexLogging)*
Show matching regex in log, note that all lists (like eg. noprocessing-list) are used as regex.

☑ **Worker logging** *(WorkerLogging)*
Show Workername in Log.

☐ **IP Matches Logging** *(ipmatchLogging)*
Enables logging of IP addresses matches in the maillog. Will show a comment instead of the range if there is text after the IP ranges (and before any number sign) eg. 182.82.10.0/24 AOL

☐ **Logging Address Matches** *(slmatchLogging)*
Enables logging of address matches in the maillog.

☐ **Add RegEx Match Header** *(AddRegexHeader)*

☑ **Unique ID logging** *(uniqeIDLogging)*
Add unique string to log

Prepend Unique ID logging *(uniqueIDPrefix)*

m1-
Prepend ID. For example: m1-

☑ **Spam Tag Logging** *(tagLogging)*
Add spam tag to log.

SMTP Status Code Reply Logging *(replyLogging)*

enabled - exclude [123]XX

☑ **Logging Records include IP & MailFrom** *(expandedLogging)*

☐ **SYSLOG Centralized Logging** *(sysLog)*

Enables logging to UNIX or Network Syslog.
Needs the Perl module Sys::Syslog for local UNIX/LINUX or Windows Eventlog logging.
If enabled and **useSysSyslog** is enabled and any of **sysLogIp** or **sysLogPort** is not set, local UNIX/LINUX or Windows Eventlog logging is used. It is not recommended to log to the Windows Eventlog!

Syslog Port (UDP) *(sysLogPort)*

514

Port for Network Syslog logging.

Syslog Facility *(SysLogFac)*

mail

Syslog Facility. Valid are kern, user, mail, daemon, auth, syslog, lpr, news, uucp, cron, authpriv, ftp, local0, local1, local2, local3, local4, local5, local6

Syslog IP *(sysLogIp)*

IP Address or hostname of your Network Syslog Daemon for Syslog logging.

☑ **ASSP local logging** *(asspLog)*

ASSP manages local logging. The logs (**logfile**) are stored inside the directory where ASSP is installed.

Roll the Logfile How Often? *(LogRollDays)*

1

ASSP closes and renames the log file after this number of days. Leave this at the default value 1, if you use BlockReporting.

LogName Date Format *(LogNameDate)*

YY-MM-DD

The standard name for the **logfile** is YY-MM-DD.maillog.txt, use this option to set it to your needs.
possible values are:
YY-MM-DD (default)
YYYY-MM-DD
MM-DD

Date/Time Format in LogDate *(LogDateFormat)*

MMM-DD-YY hh:mm:ss

Use this option to set the logdate. The default value is 'MMM-DD-YY hh:mm:ss'. The following (case sensitive !) replacements will be done:

YYYY - year four digits
YY - year two digits
MMM - month (three characters) alpha numeric - like Oct Nov Dec
MM - month numeric two digits
DDD - day (three characters) alpha numeric - like Mon Tue Fri
DD - day numeric two digits
hh - hour two digits
mm - minute two digits
ss - second two digits

NOTICE: If you change this value, BlockReports and Griplist-uploads will not work for log entries in the past (from now)!
An value has to be defined for every part of the date/time, the date must be the first part. Allowed separators in date part are '_ -./' - in time part '-_.:' .

Date/Time Language *(LogDateLang)*

English

Select the language for the day and month if **LogDateFormat** contains DDD and/or MMM.
NOTICE: If you change this value, BlockReports and Griplist-uploads will not work for log entries in the past (from now)!

☐ **Silent Mode** *(silent)*

Checked means don't print log messages to the console. **AsADaemon** overrides this.

☐ **General Debug Mode** *(debug)*

Checked sends debugging info to a .dbg file. Debug is enabled for all Threads, all the time! **debugIP** and **debugRe** will be ignored! Leave this unchecked unless there is a program error you are trying to track down.

Debug these IPs* *(debugIP)*

Enter IP addresses that you want to be debugged, separated by pipes (|). The local and the remote IP of the connection will be checked!
Not blank sends debugging info to a .dbg file. Leave this blank unless there is a program error you are trying to track down.
This can be IP address of the SMTP service monitoring agent. For example: 127.0.0.1|172.16.

Regular Expression to Identify Debug-Messages* *(debugRe)*

Put anything here to identify messages that you want to be debugged. Not blank sends debugging info to a .dbg file. Leave this blank unless there is a program error you are trying to track down.

Run this Code to switch on Debug *(debugCode)*

Put a code line here, to detect messages that you want to **debug**. The code line has to return 0 or 1. A return of 1 will switch on debug.
for example:

$Con{$fh}->{isbounce}
This code line will switch on **debug** for all bounce messages.

($Con{$fh}->{relayok} && $Con{$fh}->{isbounce})
This code line will switch on **debug** for all outgoing bounce messages.

($Con{$fh}->{ispip} && $Con{$fh}->{cip} =~ /^193\.2\.1\./)
This code line will switch on **debug** if the messages is from ISP and the IP of the server that was connected to the ISP begins with 193.2.1. .

To use this option, you need to know the internal ASSP variables and their usage!

☐ **Do not write Body to Debug *(debugNoWriteBody)***

If selected, the sent message body data will not be written to the **debug** file.

☐ **Database Connection Debug Mode *(DataBaseDebug)***

Select to **debug** the database connections!

☐ **Connection Timeout Debug Mode *(ConTimeOutDebug)***

Select to **debug** SMTP connections that are running in to timeout!

☑ **Ignore MIME Errors *(IgnoreMIMEErrors)***

If selected - Errors, based on wrong email MIME contents, will not be written to log!

Don't Log these IPs* *(noLog)*

Enter IP addresses that you don't want to be logged, separated by pipes (|). The local and the remote IP of the connection will be checked!
This can be IP address of the SMTP service monitoring agent. For example: 127.0.0.1|172.16.

Regular Expression to Identify NoLog-Messages* *(noLogRe)*

Put anything here to identify messages that you don't want to be logged.

Regular Expression to Identify Messages from/to Problematic Addresses * *(allLogRe)*

Put anything here to identify messages from/to addresses you want to look at for problem solving. Messages identified will also be set to **StoreCompleteMail**.

Regular Expression to Identify skipped Log Lines* *(noLogLineRe)*

Put anything here to identify log Lines that you don't want to be logged.

Connections Logging *(ConnectionLog)*

nolog

Session Limit Logging *(SessionLog)*

standard

Enables Logging for 'Deny SMTP Connections From' *(denySMTPLog)*

standard

Enable RWL logging *(RWLLog)*

standard

Enable LDAP logging *(LDAPLog)*

standard
ATTENTION: diagnostic will possibly write credential information in clear text to the log!

Enable VRFY logging *(VRFYLog)*

standard

Enable User Validation logging *(ValidateUserLog)*

standard

Enable PenaltyBox logging *(PenaltyLog)*

standard

Enable PenaltyBox logging *(PenaltyExtremeLog)*
standard

Enable Message Scoring logging *(MessageLog)*
standard

Enable Message-ID signing logging *(MSGIDsigLog)*
standard

Enable DNS-Backscatter detection logging *(BacksctrLog)*
standard

Enable BATV logging *(BATVLog)*
standard

Enable Validate Sender Logging *(ValidateSenderLog)*
standard

Enable SenderBase Logging *(SenderBaseLog)*
standard

Enable Greylisting/Delaying logging *(DelayLog)*
standard

Enable Bomb logging *(BombLog)*
standard
If set to verbose, the reporting to the **logfile** and the X-ASSP- scoring header will show the complete list of all hits. Otherwise only the highest match will be shown.

Enable Attachment logging *(AttachmentLog)*
standard

Enable SPF logging *(SPFLog)*
standard

Enable DNSBL logging *(RBLLog)*
standard

Enable URIBL logging *(URIBLLog)*
standard

Enable ClamAV logging *(ScanLog)*
standard

Enable DKIM logging *(DKIMlogging)*
standard

Enable thread action logging *(WorkerLog)*
nolog

Enable central Perl-signal logging *(SignalLog)*
standard
nolog will handle the Perl signals without any output (this should be never set!!!), standard will write a message to log, verbose will write a message to log and to file debugSignal.txt

Enable Bayesian Logging *(BayesianLog)*
standard
Enables verbose logging of Bayesian checks in the maillog.

Enable Conversion logging *(ConvLog)*
standard

Enable Maintenance logging *(MaintenanceLog)*
standard

Enable Report logging *(ReportLog)*
standard

Enable Scheduler logging *(ScheduleLog)*
standard

Enable SNMP logging ***(SNMPLog)***

standard

☐ **Show All Possible Hits** ***(Showmaxreplies)***

Show hits until maxreplies instead of stopping at maxhits (RBL,URIBL,RWL).

RegEx Length in Log ***(RegExLength)***

32

Defines how many bytes of a matching Regular Expression will be shown in the log
Some matching Regular Expressions are too long for one line. Default: 32

☐ **Send NOOP Info** ***(sendNoopInfo)***

Checked means you want ASSP to send a "NOOP Connection from IP" message to your SMTP server.

Notes On Logging

Notes

LDAP Setup

LDAP Host(s) *(LDAPHost)*

localhost

Enter the DNS-name(s) or IP address(es) of the server(s) that run(s) the **LDAP** database. Second entry is backup. For example: localhost. Separate entries with pipes: LDAP-1.domain.com|LDAP-2.domain.com . To use a different than the default LDAP port, define host:port.

Use SSL with LDAP (ldaps) *(DoLDAPSSL)*

no

ASSP will use 'ldaps (SSL port 636)' instead of ldap (port 389) or 'ldaps (TLS over port 389)'. The Perl module **IO::Socket::SSL** must be installed to use SSL or TLS!

LDAP Query Timeout *(LDAPtimeout)*

15

timeout when connecting to the remote server. The default is 15 seconds.

LDAP Login *(LDAPLogin)*

Most LDAP servers require a login and password before they allow queries.
Enter the DN specification for a user with sufficient permissions here.
For example: cn=Administrator,cn=Users,DC=yourcompany,DC=com

LDAP Password *(LDAPPassword)*

Enter the password for the specified LDAP login here.

LDAP Version *(LDAPVersion)*

3

Enter the version for the specified LDAP here.

LDAP Root container for Local Domains *(ldLDAPRoot)*

The LDAP lookup will use this container and all sub-containers to match the local domain query.
The literal DOMAIN is replaced by the domain part of SMTP recipient (eg. domain.com) during the search.
For example: DC=yourcompany,DC=com.
If you use DOMAIN here, you must check "LDAP failures return false" below or non local domains will be treated as local. If not defined, **LDAPRoot** will be used.

LDAP Filter for Local Domains *(ldLDAPFilter)*

This filter is used to query the LDAP database. This strongly depends on the LDAP structure.
The filter must return an entry if the domain must be relayed.
The literal DOMAIN is replaced by the domain name during the search.
for example: (&(|(|(|(|(&(objectclass=user)(objectcategory=person))(objectcategory=group))(objectclass=publicfolder))(!(objectclass=contact)))(objectclass=msExchDynamicDistributionList))(proxyaddresses=smtp:*@DOMAIN))

LDAP Root container for Local Addresses *(LDAPRoot)*

The LDAP lookup will use this container and all sub-containers to match the local email address query.
The literal DOMAIN is replaced by the domain part of SMTP recipient (eg. domain.com) during the search.
For example: DC=yourcompany,DC=com.
If you use DOMAIN here, you must check "LDAP failures return false" below or non local domains will be treated as local.

LDAP Filter for Local Addresses *(LDAPFilter)*

This filter is used to query the LDAP database. This strongly depends on the LDAP structure.
The filter must return an entry if the recipient address matches with that of any user.
The literal EMAILADDRESS is replaced by the fully qualified SMTP recipient (eg. user@domain.com) during the search.
The literal USERNAME is replaced by the user part of SMTP recipient (eg. user) during the search.
The literal DOMAIN is replaced by the domain part of SMTP recipient (eg. domain.com) during the search.
For example: (proxyaddresses=smtp:EMAILADDRESS) or (|(mail=EMAILADDRESS)(mailaddress=EMAILADDRESS)) or (&(|(|(|(|(&(objectclass=user)(objectcategory=person))(objectcategory=group))(objectclass=publicfolder))(!(objectclass=contact)))(objectclass=msExchDynamicDistributionList))(proxyaddresses=smtp:EMAILADDRESS))

Clean Up local LDAP/VRFY Database [s] *(LDAPcrossCheckInterval)*

12

Delete outdated entries from the LDAP/VRFY cache. Check the LDAP cache to the LDAP server and/or VRFY-MTA and delete not existing entries. Defaults to 12 hours. Is only used, if **ldaplistdb** is defined in the database section!

Show local LDAP Database *(LDAPShowDB)*

file:ldaplist Show file

The directory/file with the LDAP cache database file. If you change **ldaplistdb** in section Filepath you must change it here too.

☐ **force to run LDAP/VRFY-CrossCheck - now.** *(forceLDAPcrossCheck)*

ASSP will force to run an LDAP/VRFY-CrossCheck now!

Apply Changes and Run LDAP VRFY-CrossCheck Now (if checked) Refresh Browser

Max LDAP/VRFY cache Days *(MaxLDAPlistDays)*

30

This is the number of days an address will be kept on the local LDAP/VRFY cache without any email to this address.

LDAP - Destination to Local IP-address Mapping* *(ldapLocalIPAddress)*

You need to use the "file: ..." option for this parameter!
On windows systems at least Vista/2008 is required!
On multihomed systems with multiple default gateways, it could be required to define the local IP address (source) used for outgoing LDAP connections.
This parameter allows to define local IP addresses used for specific targets (IP's or hosts) - based on the local address, the system will use the right gateway/interface.
Define one entry per line, comments (#) are allowed. The syntax for an entry is 'target=>local-IP'.
target could be any of: IP(4/6) network, IP(4/6) address, hostname, domain-name with wildcard (*).

for example:
22.* => 192.168.1.1 # IP4 Network
2222:333:* => FE81::1 # IP6 Network
22.23.24.25 => 10.1.1.1, # host IP4
1:2:3:4:5:6:7:8 => FE94::5 # host IP6
*.domain.com => 10.1.1.1 # domain
host.domain.com => 192.168.1.1 # host
* => 172.16.1.1 # default - if not defined, the system default is used

NOTICE: assp will NOT check, that the local IP address is available and bound to a local interface! It will also NOT check the system routing table! YOU SHOULD KNOW WHAT YOU DO!

☐ **LDAP/VRFY failures return false *(LDAPFail)***

If checked, when an error occurs in LDAP or VRFY lookups, the test fails.

Notes On LDAP

Notes

DNS-Client Setup

☑ **Use Local DNS** *(UseLocalDNS)*

Use system default local DNS Name Servers. To use system default local DNS Servers and the configured **DNSServers** (below), unselect this option and define the system default local DNS Servers in addition below!
To **debug** the DNS queries, switch on **DebugSPF**, even you don't use the SPF-check.
All configured or local DNS Name Servers will be checked this may take some time if the servers are responding slow- please wait after apply changes!

☑ **Reuse DNS UDP Sockets** *(DNSReuseSocket)*

If selected, assp will try to reuse DNS-UDP sockets as long as this is possible. Otherwise each DNS-query will create a new UDP socket for each DNS-Server. It is recommended to set this to on, because assp could use DNS-queries very extensive, which possibly forces the assp system and/or your DNS-servers to run out of available UDP sockets.

☐ **Show DNS Name Servers Response Time in Log** *(DNSResponseLog)*

You can use this to arrange **DNSServers** for better performance.

DNS Name Servers* *(DNSServers)*

208.67.222.222|208.67.220.220

DNS Name Servers IP's to use for DNSBL(RBL), RWL, URIBL, PTR, **SPF2**, SenderBase, NS, and DMARC lookups. Separate multiple entries by "|" or leave blank to use system defaults. At least TWO DNS-servers should be defined or used by the system!
For example: 208.67.222.222|208.67.220.220 (**OpenDNS**).
An DNS-query for the domain 'sourceforge.net' is used per default to measure the speed of the used DNS-servers. If you want assp to use another domain or hostname for this, append '=>domain.tld' at the end of the line - like: 208.67.222.222|208.67.220.220=>myhost.com
To define the domain if you use the local DNS-servers '**UseLocalDNS**' without defining any DNS-servers here, simply write '=>myhost.com'.
To **debug** the DNS queries, switch on **DebugSPF**, even you don't use the SFF-check.
NOTICE: don't define any public , ISP or open DNS-Servers (eg 208.67.222.222 208.67.220.220 8.8.8.8 8.8.4.4) , if you use any of the following assp checks: DNSBL(RBL), RWL, URIBL, SenderBase ! It is recommended in EVERY case to install (and to use) at least two local DNS-Servers!
NOTICE: the DNS-server order can be changed by assp. Please read this section completely.
All configured or local DNS Name Servers will be checked this may take some time if the servers are responding slow - please wait after apply changes!

Limit the Number of used DNS-Servers *(DNSServerLimit)*

0

If set to a number > zero, assp will use the defined number of fastest responding nameservers (**DNSServers**) for DNS queries.
Otherwise, all nameserver are used every time.
Notice: This value is not checked against the number of defined **DNSServers** - don't set nonsense here!

Minimum TTL used for config reload *(host2IPminTTL)*

300

Minimum TTL used for config reload options, if hostnames are defined for any IP in regular expressions.

DNS / WHOIS - Destination to Local IP-address Mapping* *(dnsLocalIPAddress)*

You need to use the "file: ..." option for this parameter!
On windows systems at least Vista/2008 is required!
On multihomed systems with multiple default gateways, it could be required to define the local IP (source) address used for DNS connections. This parameter allows to define local IP addresses used for specific targets (IP's or hosts) - based on the local address, the system will use the right gateway/interface.
Define one entry per line, comments (#) are allowed. The syntax for an entry is 'target=>local-IP'.
target could be any of: IP(4/6) network, IP(4/6) address, hostname, domain-name with wildcard (*).

for example:
22.* => 192.168.1.1 # IP4 Network
2222:333:* => FE81::1 # IP6 Network
22.23.24.25 => 10.1.1.1 # host IP4
1:2:3:4:5:6:7:8 => FE94::5 # host IP6
*.domain.com => 10.1.1.1 # domain
host.domain.com => 192.168.1.1 # host
* => 172.16.1.1 # default - if not defined, the system default is used

NOTICE: assp will NOT check, that the local IP address is available and bound to a local interface! It will also NOT check the system routing table! YOU SHOULD KNOW WHAT YOU DO!

Maximum DNS Response Time change *(maxDNSRespDist)*

50

Maximum DNS Server response time change in milliseconds. The query order of the used nameservers is changed, if any responds time exceeds this value. Set the value to zero or empty to use a fixed DNS-Server order list.

DNS Query Timeout *(DNStimeout)*

2

Global DNS Query Timeout for DNSBL, RWL, URIBL, PTR, SPF, MX and A record lookups. The default is 2 seconds.

DNS Query Retry *(DNSretry)*

1

Global DNS Query Retry. Set the number of times to try the query. The default is 1.

DNS Query Retrans *(DNSretrans)*

1

Global DNS Query Retransmission Interval. Set the retransmission interval. The default is 1.

Notes On DNS Setup

Notes

General Server Setup

Charset for STDOUT and STDERR *(ConsoleCharset)*

System Default

Set the characterset/codepage for the console output to your local needs. Default is "System Default" - default conversion. To display nonASCII characters on the console screen, setup **UseUnicode4MaillogNames** . Restart is required!

☑ **Normalize Unicode to NFKC *(normalizeUnicode)***

If set (which is the default and recommended), all regular expressions and both, the Bayesian and the HMM engine, are normalizing all characters in there setup and the checked content, according to unicode **NFKC**.
In addition some extended (assp unique) unicode normalization is done for the unicode blocks "Enclosed Alphanumerics", "Enclosed Alphanumeric Supplement" , "Enclosed CJK Letters And Months" and "Enclosed Ideographic Supplement" - like: ① ② ⑳ ⑴ 1. ⓫ ⓘ ⓐ Ⓐ 🄰 ㊊ 特 ㍾ ㈱. Those characters are decomposed by compatibility, then recomposed by canonical equivalence (eg. to LATIN or CJK).
If this value is changed, and your system processes alot of NON-Latin mails, it is recommended to run a rebuildspamdb.
This feature requires a Perl version 5.012000 (5.12.0) or higher.
NOTICE: the rebuildspamdb task can take up to double the time, if this feature is enabled and non-LATIN mails are processed!

☑ **Enable the 8BITMIME SMTP Extension *(enable8BITMIME)***

If enabled (not default) assp offers and supports the 8BITMIME SMTP extension, if the connected peers offers and supports 8BITMIME.

☐ **Send 250 OK *(send250OK)***

Set this checkbox if you want ASSP to reply with '250 OK' instead of SMTP error code '554 5.7.1'. This will turn ASSP in some form of tarpit.

Run ASSP as a Daemon *(AsADaemon)*

No

In all NON-Windows OS (eg. Linux/BSD/Unix/OSX...) fork and close STDOUT and STDERR file handles.
Similar to the command "perl assp.pl &", but better.
If "externally controlled" is selected, ASSP simply ends and you have to restart assp from your daemon or watchdog script
If "run **AutoRestartCmd** on restart and wait" is selected, assp starts the OS command defined in **AutoRestartCmd** - assp will **NOT !** automatically terminate - the started command has to terminate/kill and to (re)start assp - like "service assp restart"!
If "run **AutoRestartCmd** on restart and exit" is selected, assp starts the OS command defined in **AutoRestartCmd** and terminates immediately!
It is important, that (especially) linux and unix system shutdown scripts are waiting until ALL assp/perl processes are ended (this may take up to one minute - see **MaxFinConWaitTime**)! Otherwise, the kernel will kill the assp/perl process at shutdown and the possibly used BerkeleyDB DB-files and environment-files WILL BE DESTROYED and cause to 100% unexpected behavior or crashes at the next start or run!

requires ASSP restart

Run as UID *(runAsUser)*

The *nix user name to assume after startup (*nix only). use the autorestart features careful, because any restart from inside ASSP will be done with the permission of this user!
Examples: assp, nobody
requires ASSP restart

Run as GID *(runAsGroup)*

The *nix group to assume after startup (*nix only). If you need to define supplementary groups, configure in addition **runAsGroupSupplementary** .
Examples: assp, nobody
requires ASSP restart

Run with supplementary groups *(runAsGroupSupplementary)*

The *nix supplementary groups to assume after startup (*nix only) - requires **runAsGroup** to be configured.
Examples: group1|group2
requires ASSP restart

Change Root *(ChangeRoot)*

The new root directory to which ASSP should chroot (*nix only). If blank, no chroot jail will be used. Note: if you use this feature, be sure to copy or link the etc/protocols file in your chroot jail. Think about your automatic restart configuration (eg. perl location) if you use this feature! And think about what happens, if perl requires to load a module on demand or a system call is done by assp! Leave this blank, if you do not really know what you do!
requires ASSP restart - in most cases, this feature will not work with all possible configuration setups !

☐ **Set ASSP File Permission on Startup *(setFilePermOnStart)***

If set, ASSP sets the permission of all ASSP- files and directories at startup to full (0777) - without any function on windows systems!

☐ **Check ASSP File Permission on Startup *(checkFilePermOnStart)***

If set, ASSP checks the permission of all ASSP- files and directories at startup - all files must be writable for the running job - the minimum permission is 0600 - without any function on windows systems!

☐ **Automatic Restart after Exception *(AutoRestart)***

If ASSP detects a main exception and it runs not as service or daemon, it will try to restart it self automatically! If running as daemon on nix/MAC , ASSP uses the action defined in **AsADaemon** to restart.

Automatic Restart ASSP on new or changed Script *(AutoRestartAfterCodeChange)*

If selected, ASSP will restart it self, if it detects a new or changed running script. An automatic restart will not be done, if ASSP is not running as

a service on windows or as daemon on linux/MAC, and **AutoRestartCmd** is not configured. If running as daemon on linux/MAC (**AsADaemon**) ASSP simply ends - you have to restart assp from your daemon script. Leave this field empty to disable the feature. Possible values are 'immed and 1...23' . If set to 'immed', assp will restart within some seconds after a detected code change. If set to '1...23' the restart will be scheduled to that hour. A restart at 00:00 is not supported.

Auto Update the Running Script (assp.pl) *(AutoUpdateASSP)*

no auto update

No action will be done if 'no auto update' is selected. You'll get a hint in the GUI (top) and a log line will be written, if a new version is availabe at the download location.
If 'download only' is selected and a new assp version is available, this new version will be downloaded to the directory c:/assp/download (assp.pl) and the syntax will be checked. The still running script will be saved version numbered to the download directory.
If 'download and install' is selected, in addition the still running script will be replaced by the new version.
Configure (**AutoRestartAfterCodeChange**), if you want the new version to become the active running script.
If this value is changed to 'download and install', the autoupdate procedure will be scheduled immediately.
If set, ASSP (on windows systems with ActivePerl installations) will search for updated Perl modules in all registered PPM repositories

new available perl modules

The installation of some modules could require manual configuration and the installation fails or an upgrade is not recommended. In this case put the case sensitive module names (one per line) in the following file. never upgrade these modules

If this value is set to 'download and install', ASSP will try an autoupdate of the new available modules. It is possible, that some modules could not be installed, because the XS module parts are still in use. In this case follow the instruction - click the "new available perl modules" button above. To disable the automatic Perl module update - set "**noModuleAutoUpdate**" below.
Click this button to see the log file for the updated modules module upgrade log
The perl module **Compress::Zlib** is required to use this feature.

☐ No Automatic Perl Module update *(noModuleAutoUpdate)*

If set, ASSP will skip the automatic Perl module update.
On NIX systems this value is ignored, if **runAsUser** is used! The automatic perl module upgrade is only done, if assp is running as user 'root'.

OS-shell command for AutoRestart *(AutoRestartCmd)*

The OS level shell-command that is used to autorestart ASSP, if it runs not as a windows service! A possible value for your system is:
cmd.exe /C start "ASSPSMTP restarted" "C:\Perl\bin\perl.exe" "c:/assp/assp.pl" "c:/assp"
Leave this field blank, if ASSP runs inside an external loop (inside the OS like assp.sh or assp.cmd).
If running on NIX systems and **runAsUser** and/or **runAsGroup** is used, don't forget to switch back to root permissions in the script!
For daemon actions in /etc/init.d (see **AsADaemon**), 'sudo -b' in front of the command may be required in case **runAsUser** and/or **runAsGroup** is used - like:
sudo -b /etc/init.d/assp restart or sudo -bs /etc/init.d/assp restart
In this case, the user in **runAsUser** must be able to 'sudo' without providing a password (sudoers , wheel)!

Restart Timeout *(RestartEvery)*

0

ASSP will automatically terminate and restart after this many seconds. Use this setting to periodically reload configuration data, combat potential memory leaks, or perform shutdown/startup processes. This will only work properly if ASSP runs as a Windows service or in a script that restarts it after it stops or **AutoRestartCmd** is configured. Alternative to this field you can use **ReStartSchedule**, to schedule restarts.

Schedule Cron time for ASSP Restart [s] *(ReStartSchedule)*

noschedule

If **not** set to "noschedule" (noschedule is default), ASSP uses scheduled times to shutdown or restart (**AutoRestartCmd**)! The syntax is the same like in **"Vixie" cron**! To disable this Scheduler leave this field blank! **Never write quotes in to this field!**
This requires an installed **Schedule::Cron** module in PERL.

Time and Date specification

Entry is the specification of the scheduled time in crontab format, which contains five mandatory time and date fields. Entry can be either a plain string, which contains a whitespace separated time and date specification.

The time and date fields are (taken mostly from "Vixie" cron):

field	values
minute	0-59
hour	0-23
day of month	1-31
month	1-12 (or as names)
day of week	0-7 (0 or 7 is Sunday, or as names)
seconds	0-59 (optional) **not supported inside ASSP !!!**

A field may be an asterisk (*), which always stands for "first-last".

Ranges of numbers are allowed. Ranges are two numbers separated with a hyphen. The specified range is inclusive. For example, 8-11 for an "hours" entry specifies execution at hours 8, 9, 10 and 11.

Lists are allowed. A list is a set of numbers (or ranges) separated by commas. Examples: "1,2,5,9", "0-4,8-12".

Step values can be used in conjunction with ranges. Following a range with "/number" specifies skips of the numbers value through the range. For example, "0-23/2" can be used in the hours field to specify command execution every other hour (the alternative in the V7 standard is "0,2,4,6,8,10,12,14,16,18,20,22"). Steps are also permitted after an asterisk, so if you want to say "every two hours", just use "*/2".

Names can also be used for the "month" and "day of week" fields. Use the first three letters of the particular day or month (case doesn't matter).

Note:
The day of a command's execution can be specified by two fields -- day of month, and day of week. If both fields are restricted (ie, aren't *), the command will be run when either field matches the current time. For example, "30 4 1,15 * 5" would cause a command to be run at 4:30

am on the 1st and 15th of each month, plus every Friday

Examples:

8 0 * * *	==>	8 minutes after midnight, every day
5 11 * * Sat,Sun	==>	at 11:05 on each Saturday and Sunday
0-59/5 * * * *	==>	every five minutes
42 12 3 Feb Sat	==>	at 12:42 on 3rd of February and on each Saturday in February
32 11 1-15/2 */3 *	==>	at 11:32 every two days from the first to the 15. every third month

In addition, ranges or lists of names are allowed.
If you want to define multiple entries separate them by "|"

Memory Limit in MB that ASSP could use *(MemoryUsageLimit)*

The memory limit in megabyte the assp process could use at maximum on your system. Set this to empty or zero to disable the feature. The check is done using the schedule defined in **MemoryUsageCheckSchedule** . If the assp process uses more memory than the limit at a scheduled time and assp is able to restart it self - a restart will be done within 15 seconds. The user running assp must have read access to /proc on nix systems or must have read access to the WMI provider on windows systems!

Schedule(s) to check the ASSP process memory usage [s] *(MemoryUsageCheckSchedule)*

0-59/10 * * * *

The schedule(s) that is used to check the current memory usage of the assp process compared to the **MemoryUsageLimit**. Default value is (0-59/10 * * * *), which means every 10 minutes. This requires an installed **Schedule::Cron** module in PERL.

My Name *(myName)*

ASSP.nospam

ASSP will identify itself by this name in the email "Received:" header and in the helo when sending report-replies. Usually the fully qualified domain name of the host.
Examples: assp.mydomain.com, mail.mydomain.org

It is highly recommended to change this value - do NOT use the default value ASSP.nospam in production environments! Because the same hostname can be used by any other server, that uses assp and sends emails to your system.
If you change this value after assp was running for a long time, add the old value to **myNameAlso** .

Additional My-Name-Definitions *(myNameAlso)*

If **myName** was changed or you use shared folders (multiple ASSP) for the corpus files, define the old or other host names here - separate multiple entries by pipe, space or comma. ASSP will use this host names in addition to **myName**, to detect the received headerlines while the rebuildspamdb is running and in the mail analyzer.

My Helo *(myHelo)*

How ASSP will identify itself when connecting to the target MTA.
The values used for incoming and outgoing/local mails are separated by "|" - for example:

SENDERHELO - IP - MYNAME - FQDN | MYNAME

The left part "SENDERHELO - IP - MYNAME - FQDN" is used for incoming mails, the right part "MYNAME" is used for outgoing mails.
If any part is empty or the complete parameter is not defined, the helo of the sending host is used.
Using the "IP" literal, you can tell your local MTA the connected IP address.
Any RFC compatible text can be used. DO NOT define the SMTP command HELO/EHLO, the command used by the sending host will take place!
The following case sensitive literals will be replaced with:

IP - the IP address of the connected host
MYNAME - the value defined in myName
FQDN - the local operating system hostname
SENDERHELO - the helo text received from the connected host

Hide IP and/or Helo *(HideIPandHelo)*

Replace any of these information (ip=127.0.0.1 helo=anyhost.local) in our received header for outgoing mails. Use the syntax ip=127.0.0.1 and/or helo=anyhost.local .

Override the Server SMTP Greeting *(myGreeting)*

Send this SMTP greeting (eg. 220 MYNAME is ready - using ASSP VERSION) instead of your MTA's SMTP greeting to the client. If not defined (default), the MTA's greeting will be sent to the client. The literal MYNAME will be replaced with **myName** and the literal VERSION will be replaced by the full version string of assp. If the starting '220 ' is not defined, assp will add it to the greeting.

assp.cfg* *(asspCfg)*

file:assp.cfg Edit file

For internal use only - it is assp.cfg file. Do not change this value.

☐ **Automatic Reload ConfigFile *(AutoReloadCfg)***

If selected and the assp.cfg file is changed externally, ASSP will reload the configuration from the file automatically.

assp.cfg version *(asspCfgVersion)*

2.5.5(16366)
ASSP will identify the assp.cfg file. Do not change this.

Schedule Configuration Changes* ***(ConfigChangeSchedule)***

Use this option to schedule configuration changes. You must use the file option like 'file:files/configchangeschedule.txt' to define schedules - an empty value disables this feature.
Define one schedule per line - comments are not allowed in a schedule definition line!
The line has to start with the schedule string (see **ReStartSchedule**) followed by the variable (or hidden variable) name to change, followed by ':=', followed by the value to change the variable to - like:

8 0 * * * myNameAlso:=otherhost1.mydomain.tld
0 6 * * *|0 10 * * * myNameAlso:=otherhost2.mydomain.tld
0 1 * * * debug:=1
0 2 * * * debug:=

The schedule string can contain multiple schedule definitions separated by pipe'|'. You will get errors if:
- the schedule definition is wrong
- the variable name is wrong (does not exists)
- the syntax of the value is wrong
Notice - assp will only check the syntax at definition time - the logical correctness of the value will be checked at the scheduled time! So, assp will (for example) not check any dependencies at definition time - if a dependency is wrong, the change request at the scheduled time will fail!
Notice - all configuration changes are done with 'root' permission! For this reason, this configuration parameter is only visible to root and it is stored encrypted!

For advanced users ONLY:
Using the following extension, requires a deep internal knowledge of the assp code!
It is also possible to schedule a call to an internal assp subroutine. The name of the subroutine has to begin with a '&', the parameters that should passed to the subroutine must be in '()' - like:
0 6 * * * &subname(var1,var2,..,...)
0 7 * * * &subname()
Notice: the subroutine will be called in the MainThread and syntax check will be done at run time - possible errors are shown in the log!

Proxy Server ***(proxyserver)***

The Proxy Server to use when uploading global statistics and downloading the greylist.
Examples: 192.168.0.1:8080, 192.168.0.1

Proxy User ***(proxyuser)***

The Proxy-UserName that is used to authenticate to the proxy.

Proxy Password ***(proxypass)***

The password for Proxy-UserName that is used to authenticate to the proxy.

Web Admin Port ***(webAdminPort)***

55555
The port on which ASSP will listen for http connections to the web administration interface. If you change this, after you click Apply you must change the URL on your browser to reconnect. You may also supply an IP address or hostname to limit connections to a specific interface. Separate multiple entries by pipe "|"!
Examples: 55555, 192.168.0.5:12345, myhost:12345, 192.168.0.5:22345|myhost:12345

☐ **Use https instead of http** ***(enableWebAdminSSL)***

If selected the web admin interface will be only accessible via https. If you change this, after you click Apply you must change the URL on your browser to reconnect. This requires an installed **IO::Socket::SSL** module in PERL.
A server-certificate-file "certs/server-cert.pem" and a server-key-file "certs/server-key.pem" must exist and must be valid!
If you do not have valid certificates, you may generate both files online with **www.mobilefish.com** or you may use OpenSSL to generate **Self-signed SSL certificates**! More configuration options are **webSSLRequireCientCert**, **SSLWEBCertVerifyCB** and **SSLWEBConfigure** .

Web Admin Password - Masterpassword (root) ***(webAdminPassword)***

45WPcXBk5dhLo
The password for the web administration interface for user root(minimum of 5 characters).
DO NOT use the digits "45" as the first two characters of the password or you will be not able to login ever again!
If root is logged on, no other logins are allowed. Always use the "logoff"-button as root to terminate the session - closing the browser without logoff could cause other session to be disallowed for up to 15 minutes.

Only Allow Admin Connections From* ***(allowAdminConnectionsFrom)***

An optional list of IP addresses and/or hostnames from which you will accept web admin connections. Blank means accept connections from any IP address.
Note: if you make a mistake here, you may disable your web administration interface and be forced to manually edit your configuration file to fix it.
Examples:
127.0.0.1|172.16.

☑ **HTTP and HTTPS require enabled browser cookies** ***(httpRequireCookies)***

Cookie based http session ID's are used by assp to handle different requests from the same IP (eg behind NAT). Switch this off, if you are unable to use cookies in your browser. If switched off, a security hole is opened for connection that are using NAT - it could be possible that a second workstation (behind NAT) is able to login to the GUI, without user credentials if the same OS and browser version is used.

Status Response Literal for a Healthy State of ASSP *(webStatHealthyResp)*

healthy

This option must be set and it must be different to **webStatNotHealthyResp**. This literal will be given back in stat requests, if ASSP is working healthy.

Status Response Literal for a Not Healthy State of ASSP *(webStatNotHealthyResp)*

not healthy

This option must be set and it must be different to **webStatHealthyResp**. This literal will be given back in stat requests, if ASSP is working not healthy.

Raw Statistics Port *(webStatPort)*

55553

The port on which ASSP will listen for http or telnet connections to the statistics interface. You may also supply an IP address to limit connections to a specific interface. Only one value is supported!
The stats are available via browser or telnet (or telnet similar socket). Using telnet, press ENTER two times to get the healthy state (' healthy [CRLF]' or ' not healthy [CRLF]' in a single line), this is the recommended methods to get the 'UP'-state of assp from nagios or any other external script.
Type 'stat[ENTER][ENTER]' to get the STATS in raw text where each line is terminated with '[CR]LF' (CR is send in any case, if the request contains CR).
The HTML/browser output are LF terminated STAT lines.
If you have configured " **exportExtremeBlack** ", your firewall (pfsense/pfBlockerNG or snort) may download the extreme black IP list using this interface - append "/extremeblack" to the URL.
The download URL, used by your firewall, should look like: http://assp.domain.local:55553/extremeblack
Examples: 55553, 192.168.0.5:12345

☐ **Use https instead of http *(enableWebStatSSL)***

The web stat interface will be only accessible via https. This requires an installed **IO::Socket::SSL** module in PERL.
A server-certificate-file "certs/server-cert.pem" and a server-key-file "certs/server-key.pem" must exits and must be valid! More configuration options are **statSSLRequireClientCert**, **SSLSTATCertVerifyCB** and **SSLSTATConfigure** .

Only Allow Raw Statistics Connections From* *(allowStatConnectionsFrom)*

127.0.0.1

An optional list of IP addresses from which you will accept raw statistical connections. Blank means accept connections from any IP address.
Examples:
127.0.0.1|172.16.

☑ **Enable HTTP Compression in GUI *(EnableHTTPCompression)***

Enable HTTP Compression for faster web administration interface loading. The perl module **Compress::Zlib** is required to use this feature.

HTTP - Destination to Local IP-address Mapping* *(httpLocalIPAddress)*

You need to use the "file: ..." option for this parameter!
On windows systems at least Vista/2008 is required!
On multihomed systems with multiple default gateways, it could be required to define the local IP address (source) used for outgoing HTTP connections.
This parameter allows to define local IP addresses used for specific targets (IP's or hosts) - based on the local address, the system will use the right gateway/interface.
Define one entry per line, comments (#) are allowed. The syntax for an entry is 'target=>local-IP'.
target could be any of: IP(4/6) network, IP(4/6) address, hostname, domain-name with wildcard (*).

for example:
22.* => 192.168.1.1 # IP4 Network
2222:333:* => FE81::1 # IP6 Network
22.23.24.25 => 10.1.1.1 # host IP4
1:2:3:4:5:6:7:8 => FE94::5 # host IP6
*.domain.com => 10.1.1.1 # domain
host.domain.com => 192.168.1.1 # host
* => 172.16.1.1 # default - if not defined, the system default is used

NOTICE: assp will NOT check, that the local IP address is available and bound to a local interface! It will also NOT check the system routing table! YOU SHOULD KNOW WHAT YOU DO!

☐ **Enable Floating Menu Panel in GUI *(EnableFloatingMenu)***

Allow the left menu panel on the web administration interface to float.

☐ **Hide the Alpha Index Menu Panel in GUI *(hideAlphaIndex)***

Removes the index panel on the left side in the GUI, but the index is accessible by clicking on the "SEARCH icon" in the left-hand top menu.

Sliding Speed of the Alpha Index Menu Panel in GUI *(IndexSlideSpeed)*

normal

Adjust the sliding speed of the Alpha Index Menu Panel in GUI to your needs.

☑ **Remember the last GUI position *(RememberGUIPos)***

If selected, the GUI will remember the last topic of the main menu, that had the focus, was changed, that were jumped to or that were clicked on.

☑ **Show Internal Names in the GUI *(EnableInternalNamesInDesc)***

Show the internal names in the web interface. The internal names are used in the configuration file (assp.cfg), in the application code, and in the menu bar on the left side of the GUI.

☐ **Jump to the End of the Maillog** *(MaillogTailJump)*

Causes the browser window to jump to the bottom of the maillog instead of sitting at the top of the display.

Maillog Tail Bytes *(MaillogTailBytes)*

10000

The number of bytes that will be shown when the end of the maillog is viewed. The default value is 10000.

Cache Cleaning Interval [s] *(CleanCacheEvery)*

6

This period (in hours) determines how frequently ASSP does cache-housekeeping.

Statistics Save Interval [s] *(SaveStatsEvery)*

30

This period (in minutes) determines how frequently ASSP statistics are written to a local file.

☑ **Upload Consolidated Spam Statistics** *(totalizeSpamStats)*

ASSP will upload its statistics to be consolidated with the **global ASSP totals**. This is a great marketing tool for the ASSP project — please do not disable it unless you have a good reason to do so. No private information is being disclosed by this upload.

☐ **Enable Graphical Statistics Collection** *(enableGraphStats)*

ASSP will collect statistical data in files located in the '/logs' folder (scoreGraphStats-YYYY-MM.txt , statGraphStats-YYYY-MM.txt). If data are collected and the module lib/ASSP_SVG.pm is installed and the files images/stat.gplot, images/svg_style.css, images/svg_defs.svg and images/svg.js are installed and your browser supports SVG, assp will show graphical statistic data, if you click on a line in the 'Info and **Stats**' view.
If **baysConf** is configured, assp will also collect statistical data about the Bayesian and HMM confidence distribution - the file names are confidenceGraphStats-YYYY-MM.txt.
It is recommended to set '**SaveStatsEvery**' to a value of 5 or 10 minutes, if this option is enabled!
ASSP will delete '*GraphStats...txt'-files if they are over one year old. If you don't need some of that files any longer, remove them manually!

Reload Option Files Interval [s] *(ReloadOptionFiles)*

300

If set not to zero, ASSP reloads configuration option files (file:.....) every this many seconds if they have changed. It is not recommended (and could make ASSP unavailable) to use rsync or any external tool to snychronize caches and list permanently. If you need to snychronize data between ASSP installations, you better use a database of your choice!

Ordered-Tie Hash Table Size *(OrderedTieHashTableSize)*

10000

The number of cached entries allowed in the hash tables used by ASSP. This belongs to **griplist**, if **useDB4griplist** is not set and to temporary lists used by the rebuild **spamdb** process, if **useDB4Rebuild** is set and BerkeleyDB is not available. Larger numbers require more RAM but result in fewer disk hits. The default value is 10000. Adjust down to use less RAM. Adjust up to speedup.

TCP and SSL Read/Write Buffer Size *(TCPBufferSize)*

Define the buffer size in byte used for TCP- and SSL socket read and write operations - defaults to empty.
Any or all of the following four values can be defined:

tcprcv - TCP receive buffer size
tcpsnd - TCP send buffer size
sslrcv - SSL receive buffer size
sslsnd - SSL send buffer size

Multiple value definition have to be separated by comma or pipe, like: tcprcv = 65536, tcpsnd = 65536, ...
Possible size values are 8192-9999999 , special value for sslrcv and sslsnd is zero.
If a value is not specified for tcprcv or tcpsnd, the according TCP buffer size reported by the system is used - but at least 8192 byte.
If a value is not specified for sslrcv or sslsnd, a value of 16384 byte is used, which is the maximum size of a single SSL frame of the SSL layer.
If a value of zero is specified for sslrcv or sslsnd, the according system TCP socket buffer size is used.
Under normal conditions any setting here will be not required. But, if you notice a bad SSL transmission performance in relation to the speed of plan TCP sockets, it may help to set both SSL buffer size to the size of the according system TCP buffer.
like: sslrcv = 0, sslsnd = 0

Never internaly Queue Mails larger than this Size *(neverQueueSize)*

20971520

Default is 20971520 (20MB) - lowest possible value is 1000000. Any mail that is announced to be or grows larger than this size in byte, will not be queued for actions and checks that requires the complete mail to be internaly queued.
skipped actions are: DKIM signature generation and charset conversions
skipped checks are: all Plugins in level 2 (complete mail) and the full mail DKIM check
Please also check **npSize** and **npSizeOut** .

☐ **Use BerkeleyDB for Internal Caches** *(useDB4IntCache)*

ASSP uses some internal caches that could grow to a large number of entries. Switch this to ON, apply and restart assp, if you want ASSP to use less memory and be a little slower. If Changed to ON, ASSP will import the current internals in to the databases at the next restart. The perl module **BerkeleyDB** version 0.34 or higher and BerkeleyDB version 4.5 or higher is required to use this feature.

Module Call Timeout *(ALARMtimeout)*

10

Global Timeout for SPF checks. The default is 10 seconds.

Thread Control - be careful changing the following green options!

Number of SMTP-Threads *(NumComWorkers)*

5

Number of SMTP-Threads to be used! Typical and default is 5. 10 should be enough for 200.000 connections a day. 15 should be the absolute maximum. Values above 7 will mostly not increase performance. Configurable values are between 2 and 29. Restart ASSP if you changed this and you are using any database connection! A restart of assp is required if tis value was increased.

Reserved Number of Outbound-SMTP-Threads on relayPort *(ReservedOutboundWorkers)*

0

Number of SMTP-Threads to be reserved for relayed (outbound) connections on **relayPort** ! This number of Threads will be exclusive reserved for connections on **relayPort** . For example: NumComWorkers=7 and ReservedOutboundWorkers=2 - mails on **listenPort** , **listenPort2** and **listenPortSSL** are using worker 1-5 and mails on **relayPort** using worker 7-1 ! If you are not using the **relayPort**, do not reserve any workers.

☑ **automatically restart died threads *(autoRestartDiedThreads)***

If defined, a (for any reason) died thread will be automatically restarted!

Maximum time to wait for SMTP-Workers to finish connections *(MaxFinConWaitTime)*

45

The maximum time in seconds to wait for SMTP-Workers to finish connections, in case of a shutdown or restart of ASSP. Default is 45. Configurable values are 10 to 599.

☑ **Monitor the MainThread *(MonitorMainThread)***

If defined, the MainThread will be monitored for healthy by the MaintThread (Worker 10000)!

Enable Higher Performance *(EnableHighPerformance)*

off

If set, the SMTP-Worker-Threads will get new pending connections faster - using less wait states. The speed to interrupt the workers by the MainThread is increased. Using this feature will increase the CPU usage of the system! An too high setting, may lead in to stuck workers, or in worth case, in to a much lower perfomance.
If there is any doubt about this setting, leave this feature off!

thread cycle time *(ThreadCycleTime)*

3000

Time in microseconds (for SMTP workers and MainThread) to give each other thread to run in high CPU-workload conditions. Default value is 3000, typical and valid values are between 1000 and 9999. A higher value will reduces CPU usage but cause ASSP to run more slowly!

MaintenanceThread cycle time *(MaintThreadCycleTime)*

3000

Time in microseconds (for MaintThread) to give each other thread to run in high CPU-workload conditions. Default value is 3000, typical and valid values are between 10 and 9999. A higher value will reduce CPU usage but cause ASSP to run more slowly!

RebuildSpamDBThread cycle time *(RebuildThreadCycleTime)*

30

Time in microseconds (for RebuildSpamDBThread) to give each other thread to run in high CPU-workload conditions. Default value is 30, typical values are between 10 and 1000. You can set this to 0, if your OS honors system-yield-calls (0 is not recommended on Windows OS) and your system is fast enough! A higher value will reduce CPU usage but cause ASSP to run more slowly!

Stack Size use by every Thread *(ThreadStackSize)*

0

The stack size in MB that is used by every thread. Default is 0, which means to use the default system stack size. 16 MB is the default system stack size on windows platforms. This system value may differ on different platforms. To get the default stack size on linux use the shell command "ulimit -a". Try to increase this value, if you get "out of memory" errors while running assp. Changing this value requires an assp restart to take effect.

Use This IO Engine *(IOEngine)*

IO::Poll

Depending on your operating system and your Perl version, it could be necessary to use the non default restricted **IOEngine** 'IO::Select'. Try this 'IO::Select', if you see unexpected early closed connections or a large amount of SMTP timouts in the log. For Strawberry-Perl on Windows it recommended to use 'IO::Select'. You have to restart ASSP, if you have changed this value!
On most OS or Perl distributions the IO::Select is restricted to a maximum of 1024 concurrent active file descriptors (disk files and [TCP,UDP,unix] sockets) within a single "fd-set" object. This depends on the setting of the C-compiler option "FD_SETSIZE" while your perl was compiled.
If this C-compiler option was set too low at the Perl compile time - you will see errors like:
"can not open file .. too many opened files"
or
"can not create socket to .. too many opened files"
If this happens, you will need to recompile your perl with a higher value, set for "FD_SETSIZE".
NOTICE that some OS are not supporting the setting of "FD_SETSIZE" because of a hard coded value for "__FD_SETSIZE" - for example linux.
On most Unix and all Windows OS it is supported.

NOTICE that a too low setting of 'ulimit -n' may cause the same errors on all nix OS.

Minimum Poll/Select Wait Time *(MinPollTime)*

2

The time in milliseconds that ASSP will at least wait for IO::Poll/IO::Select events! A higher value will reduce CPU usage but cause ASSP to run more slowly! Default is 2.

CPU priority for SMTP-Threads *(WorkerCPUPriority)*

0

Set the priority for the Workers in relation to all other processes/threads on the system. Than higher the value - than lower the priority. Default is 0 (system default is 0). Possible values are 0,1 and 2. This requires installed **Thread::State** module. It is recommended to run the Workers on lower priority, if ASSP has to process most of the time a large number of mails at one moment (number of mails > **NumComWorkers**).

Cpu Affinity for assp *(asspCpuAffinity)*

-1

Set the Cpu Affinity for all threads . Default is -1 (for use all CPU's). Possible values are comma or space separated CPU numbers starting with zero (0) or -1 for all CPU's. This requires installed **Sys::CpuAffinity** module. This feature will possibly not work on MacOS and OpenBSD and on any OS, if the system contains more than 32 CPU's.

pre allocate memory for every mail *(PreAllocMem)*

100000

ASSP pre-allocates this number of bytes in mainstorage two times (in/out) for every mail to avoid memoryfracmentation (particularly in ASSP long run conditions). The memory will be allocated, if the DATA command is received from the server. Default is 100000 - this is enough for most of the mails. If ASSP receives the SIZE command from the server, the pre-allocation-memory will be calculated on that value. Question: Is it better to increase this value? Answer: Yes, it is - but be careful, this may cause ASSP running in out of memory errors!

☑ **Freeup Memory Garbage *(FreeupMemoryGarbage)***

If defined, all Threads will try to recover memory every five minutes!

Connection Transfer Timeout *(ConnectionTransferTimeOut)*

30

Global Timeout for MainThread to transfer a connection to any Worker. If no Worker is able to take the new SMTP-connection (for any reason), the new connection will be dropped! The default is 30 seconds.

☑ **Show Performance DATA in SMTP Connection screen *(ShowPerformanceData)***

If defined, performance data will be shown in top of the SMTP connection screen!

end of Thread Control

☑ **Use Local Time *(UseLocalTime)***

Use local time and timezone offset rather than UTC time in the mail headers.

Notes On Server Setup

Notes

Rebuild Hidden Markov Model and Bayesian Database

Schedule Cron time for RebuildSpamdb [s] *(RebuildSchedule)*

noschedule

If **not** set to "noschedule" (noschedule is default) , ASSP uses scheduled times to run the RebuildSpamdb! The syntax is the same like in **"Vixie" cron**! To disable the Scheduler write "noschedule"! **Never write quotes in to this field!**
This requires an installed **Schedule::Cron** module in PERL.
It is possible to define more than one scheduled time per day to keep the Bayesian and HMM databes up to date, but this is not required - use '**newReportedInterval**' instead.
If a file c:/assp/rebuilddebug.txt exists, the rebuild task will write the **debug** output to this file.

Time and Date specification

Entry is the specification of the scheduled time in crontab format, which contains five mandatory time and date fields. Entry can be either a plain string, which contains a whitespace separated time and date specification.

The time and date fields are (taken mostly from "Vixie" cron):

field	values
minute	0-59
hour	0-23
day of month	1-31
month	1-12 (or as names)
day of week	0-7 (0 or 7 is Sunday, or as names)
seconds	0-59 (optional) **not supported inside ASSP !!!**

A field may be an asterisk (*), which always stands for "first-last".

Ranges of numbers are allowed. Ranges are two numbers separated with a hyphen. The specified range is inclusive. For example, 8-11 for an "hours" entry specifies execution at hours 8, 9, 10 and 11.

Lists are allowed. A list is a set of numbers (or ranges) separated by commas. Examples: "1,2,5,9", "0-4,8-12".

Step values can be used in conjunction with ranges. Following a range with "/number" specifies skips of the numbers value through the range. For example, "0-23/2" can be used in the hours field to specify command execution every other hour (the alternative in the V7 standard is "0,2,4,6,8,10,12,14,16,18,20,22"). Steps are also permitted after an asterisk, so if you want to say "every two hours", just use "*/2".

Names can also be used for the "month" and "day of week" fields. Use the first three letters of the particular day or month (case doesn't matter).

Note:
The day of a command's execution can be specified by two fields -- day of month, and day of week. If both fields are restricted (ie, aren't *), the command will be run when either field matches the current time. For example, "30 4 1,15 * 5" would cause a command to be run at 4:30 am on the 1st and 15th of each month, plus every Friday

Examples:

8 0 * * *	==>	8 minutes after midnight, every day
5 11 * * Sat,Sun	==>	at 11:05 on each Saturday and Sunday
0-59/5 * * * *	==>	every five minutes
42 12 3 Feb Sat	==>	at 12:42 on 3rd of February and on each Saturday in February
32 11 * * * 0-30/2	==>	11:32:00, 11:32:02, ... 11:32:30 every day

In addition, ranges or lists of names are allowed.
If you want to define multiple entries separate them by "|"

☑ **Use BerkeleyDB/DB_File or orderedtie for the RebuildSpamDB Internal Caches** *(useDB4Rebuild)*

The RebuildSpamDB thread creates some internal temprary caches, which can grow to a very large number of entries. Switch this on (default), if you want this thread to use less memory and be possibly a little slower.
Adjust **RebuildThreadCycleTime** to a lower value (between 0 and 30) to speed up the RebuildSpamDB thread.
The perl module **BerkeleyDB** version 0.34 or higher and BerkeleyDB version 4.5 or higher is required to use this feature. DB_File (Berkeley V1) will be used if BerkeleyDB is not available - this is not recommended! If both BerkeleyDB and DB_File are not available, the rebuild thread will hold all temporary data in RAM - the same way, this option were set to "OFF".

☑ **Replace the old Records in Spamdb and Spamdb.helo** *(ReplaceOldSpamdb)*

If selected (default), the new created records for Spamdb and Spamdb.helo will replace the old (belongs not to HMM, which is replaced every time). If not seleted, the new records will be added to Spamdb and Spamdb.helo .

☐ **Do move2num Before Rebuild** *(doMove2Num)*

Renames files to numbers before the rebuild is started. If this is done, some other features like 'MailLogTail' and 'Block-Report' will be unable to find the files! Setting this option to "ON" is not recommended!

Interval for processing new Reported Mails *(newReportedInterval)*

10 5

File count and interval definition (count minutes) for processing new reported mails (**correctedspam** , **correctednotspam**) - process if at least 'first value' mails are reported but every 'second value' minutes. defaults to '10 5'
Set the first value to zero to disable this feature.
If enabled, new reported mails or files moved in to the corpus via GUI are used, to immediately update the Spamdb and **HMMdb** with the new information.
This will keep the databases continuously uptodate and the **RebuildSchedule** interval could be increased, if there are enough files in the corpus and your corpus norm is fine.
If you need to copy/move several files from outside assp in to the corpus and you want assp to process them immediately, copy/move the files in to the subfolder "error/.../newManuallyAdded".

Max Days of Keep Deleted ***(MaxKeepDeleted)***

0

The maximum number in days deleted files in the bayesian collection folders (**spamlog** , **notspamlog**) will be kept. This is necessary when **EmailBlockReport** is used to handle the file and the file is meanwhile deleted. The list of files that are maked for deletion is stored in trashlist.db .

Automatic Corpus Correction ***(autoCorrectCorpus)***

0.6-1.4-4000-14

(Syntax: a.a[a]-b.b[b]-cccc-dd or empty - default is "0.6-1.4-4000-14") If the corpus norm (the weight between spamwords/hamwords) is less than "a" (0.6 - too much ham) or greater than "b" (1.4 - too much spam), assp will delete the excess (oldest) files from the corresponding folder (**spamlog** , **notspamlog**). ASSP will keep a minimum of "c" (4000) files in the folder and will never delete files that are younger than "d" (14) days. This cleanup will run at the end of the rebuildspamdb task. So the corrected file corpus will take effect at the next rebuildspamdb! If this value is defined, assp will use the middle value of "a" and "b" ((a+b)/2) as target corpusnorm and will try to reach this value, using (as many as possible) but only such a count of files in the folders **spamlog** and **notspamlog** as required!

File Processing time Limit ***(RebuildFileTimeLimit)***

1 5

(Syntax: a[.aa] b[.bb] - default is "1 5")
Define one, or two space or comma separated values.
If the first value is not zero and the processing time of a single corpus file exceeds the first value in seconds, this will be shown in the rebuild log.
If the second value is not zero and the processing time of a single corpus file exceeds the second value in seconds, the file will be moved to the folder "c:/assp/rebuild_error" to prevent future runtime penalties.

Notification Email To ***(RebuildNotify)***

Email address(es) to which you want ASSP to send a notification email after the rebuild task is finished. The file rebuildrun.txt is included in this notification. Separate multiple entries by "|".

☐ **Run the Rebuild in Test Mode** ***(RebuildTestMode)***

If selected, all rebuildspamdb tasks will not populate the **spamdb** and hmmdb - and no data will be sent to the griplist-Server.

☑ **Keep rebuildspamdb.pm compatible to assp.pl** ***(forceRebuildDowngrade)***

Keep rebuildspamdb.pm compatible to assp.pl in case of an assp.pl version downgrade.

☐ **Run RebuildSpamdb now** ***(RunRebuildNow)***

If selected, RebuildSpamdb will be started immediately.

Apply Changes and Run Rebuild SpamDB Now (if checked) | Refresh Browser

Last Result Of Rebuildspamdb

Last Run Rebuildspamdb

Rebuildspamdb-debug-output - create the file 'rebuilddebug.txt' to enable the **debug** mode - delete the file to stop the **debug** mode for the rebuildspamdb task

Rebuildspamdb-debug-output

normfile - shows current:
Corpus-Norm , Corrected-SpamFiles , Corrected-NotSpamFiles , Spamlog-Files , NotSpamlog-Files , SpamWords/File , Hamwords/File , Spamwords , Hamwords

normfile

Notes On RebuildSpamdb

Notes

CharacterSet Conversions and TNEF Processing

inbound charset conversion table* ***(inChrSetConv)***

If defined, characterset conversion for inbound mails will be done. For example: if your email server does not understand UTF-8, ASSP will convert the mail parts to the characterset of your choice. The rules specified here are used to convert text parts of inbound mails from one to another characterset.
*Example:*UTF-8=>ISO-8859-1|ISO-8859-15=>ISO-8859-1
This requires an installed **Email::MIME** module in PERL.
This conversions are done for all (inbound,CC,report ..) mails except relayed mails. The converted mail will be not available on disk except DEBUG.

outbound charset conversion table* ***(outChrSetConv)***

If defined, characterset conversion for outbound mails will be done. For example: if your email server is unable to send mails in UTF-8, ASSP will convert the mail parts to UTF-8. The rules specified here are used to convert text parts of outbound mails from one to another characterset.
*Example:*ISO-8859-1=>UTF-8|ISO-8859-2=>UTF-8|windows-1250=>UTF-8
This requires an installed **Email::MIME** module in PERL.
This conversions are done only for relayed mails!

☐ **convert inbound MS-TNEF attachments to MIME** ***(doInFixTNEF)***

convert inbound MS-TNEF attachments like winmail.dat to MIME parts/attachments. If a TNEF-file is attached by other than Exchange (like application/octet-stream) no conversion will be done.
In addition to **Email::MIME** this requires both installed **Convert::TNEF** and **MIME::Types** module in PERL.

☑ **keep the MS-TNEF part in inbound mail** ***(keepInTNEF)***

keep inbound MS-TNEF attachments like winmail.dat in MIME parts. If unchecked and the conversion is successful, the original attachment will be removed from mail!

☐ **convert outbound MS-TNEF attachments to MIME** ***(doOutFixTNEF)***

convert outbound MS-TNEF attachments like winmail.dat to MIME parts/attachments. If a TNEF-file is attached by other than Exchange (like application/octet-stream) no conversion will be done.
In addition to **Email::MIME** this requires both installed **Convert::TNEF** and **MIME::Types** module in PERL.

☑ **keep the MS-TNEF part in outbound mail** ***(keepOutTNEF)***

keep outbound MS-TNEF attachments like winmail.dat in MIME parts. If unchecked and the conversion is successful, the original attachment will be removed from mail!

☐ **convert NoProcessing mails** ***(convertNP)***

Set this to on, if noprocessing mails should be converted, which is normally not the case.

☐ **convert DKIM mails** ***(doDKIMConv)***

DKIM messages could normally not modified. If checked, conversions will be done on DKIM messages - you have to disable the DKIM check on your email server (MTA)!

☐ **TNEFDEBUG (only in dev)** ***(TNEFDEBUG)***

prints TNEF conversion **debug** info to screen.

Notes On Character Conversions / TNEF
Notes

SSL Proxy and TLS support

How to Handle STARTTLS Requests *(DoTLS)*

drop TLS

If set to "drop TLS", any STARTTLS request will be removed from the protocol stack and no connection will ever go in to any TLS mode!
If set to "TLS to Proxy" and both peers (client and server) supports TLS, both connection will be moved in to a transparent Proxy mode. All data will be encrypted and unreadable to ASSP.
If set to "do TLS", ASSP will be the "man in the middle". ASSP will try to move both connections in to TLS. All data will be readable to ASSP - so all checks could be done. If any of the peers does not support TLS, ASSP will fake this (250-STARTTLS) to the other peer. So it could be possible, that the connection to the client is going in to TLS mode, even if TLS is not supported by the server. If a client does not request TLS (STARTTLS) even it has got the (250-STARTTLS), ASSP tries to start a TLS session to server, if he has sent (250-STARTTLS)! This behavior belongs to incoming and outgoing messages. This option requires the installed perl module **IO::Socket::SSL**!
For "do TLS" a server-certificate-file " **SSLCertFile** " and a server-key-file " **SSLKeyFile** " must exist and must be valid!
If you do not have valid certificates, you may generate both files online with **www.mobilefish.com** or you may use OpenSSL to generate **Self-signed SSL certificates**! If you have installed OpenSSL (must be in PATH) and installed and enabled IO::Socket::SSL and ASSP is unable to find valid certificates - ASSP will try to create them at startup!

SSL-failed-Cache

SSL version used for transmission *(SSL_version)*

SSLv23:!SSLv3:!SSLv2

Sets the version of the SSL protocol used to transmit data. The default is SSLv23:!SSLv3:!SSLv2.
The IO::Socket::SSL POD explains:
Sets the version of the SSL protocol used to transmit data.
'SSLv23' and the older definition 'SSLv2/3' (of the same) uses a handshake compatible with SSL2.0, SSL3.0 and TLS1.x, while 'SSLv2', 'SSLv3', 'TLSv1', 'TLSv1_1' or 'TLSv1_2' restrict handshake and protocol to the specified version.
All values are case-insensitive. Instead of 'TLSv1_1' and 'TLSv1_2' one can also use 'TLSv11' and 'TLSv12'. Support for 'TLSv1_1' and 'TLSv1_2' requires recent versions of Net::SSLeay and openssl.

Independent from the handshake format you can limit to set of accepted SSL versions by adding !version separated by ':'.

The default **SSL_version** is 'SSLv23:!SSLv3:!SSLv2' which means, that the handshake format is compatible to SSL2.0 and higher, but that the successful handshake is limited to TLS1.0 and higher, that is no SSL2.0 or SSL3.0 because both of these versions have serious security issues and should not be used anymore.
You can also use !TLSv1_1 and !TLSv1_2 to disable TLS versions 1.1 and 1.2 while still allowing TLS version 1.0.

Setting the version instead to 'TLSv1' might break interaction with older clients, which need a SSL2.0 compatible handshake. On the other side, some clients just close the connection when they receive a TLS version 1.1 request. In this case setting the version to 'SSLv23:!SSLv2:!SSLv3:!TLSv1_1:!TLSv1_2' might help.

If not set, the following defaults are used by IO::Socket::SSL :
for local listeners : SSLv23:!SSLv3:!SSLv2
for client connections : SSLv23:!SSLv3:!SSLv2

SSL key cipher list *(SSL_cipher_list)*

If this option is set, the cipher list for the connection will be set to the given value, e.g. something like 'ALL:!LOW:!EXP:!ADH' or 'DEFAULT:!aNULL:!RC4:!MD5'. Look into the OpenSSL documentation (**http://www.openssl.org/docs/apps/ciphers.html#CIPHER_STRINGS**) for more details. Setting this value causes the 'SSL_honor_cipher_order' flag to be switched on (BEAST vulnerable)
If this option is not used (default) the IO::Socket::SSL builtin defaults are used, which are suitable for most cases.

The IO::Socket::SSL defaults are :
for local listeners : ECDHE-ECDSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-GCM-SHA256 DHE-RSA-AES128-GCM-SHA256 ECDHE-ECDSA-CHACHA20-POLY1305 ECDHE-RSA-CHACHA20-POLY1305 ECDHE-ECDSA-AES256-SHA ECDHE-RSA-AES256-SHA DHE-RSA-AES256-SHA ECDHE-ECDSA-AES128-SHA ECDHE-RSA-AES128-SHA DHE-RSA-AES128-SHA AES128-GCM-SHA256 AES256-SHA AES128-SHA DES-CBC3-SHA ECDHE-RSA-AES256-GCM-SHA384 ECDHE-ECDSA-AES256-GCM-SHA384 AES256-SHA256 DHE-DSS-AES256-SHA256 DHE-DSS-AES256-SHA DHE-DSS-AES128-SHA EDH-DSS-DES-CBC3-SHA !EXP !MEDIUM !LOW !eNULL !aNULL !RC4 !DES !MD5 !PSK !SRP

for client connections : ECDHE-ECDSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-GCM-SHA256 DHE-RSA-AES128-GCM-SHA256 ECDHE-ECDSA-CHACHA20-POLY1305 ECDHE-RSA-CHACHA20-POLY1305 ECDHE-ECDSA-AES256-SHA ECDHE-RSA-AES256-SHA DHE-RSA-AES256-SHA ECDHE-ECDSA-AES128-SHA ECDHE-RSA-AES128-SHA DHE-RSA-AES128-SHA AES128-GCM-SHA256 AES256-SHA AES128-SHA DES-CBC3-SHA ECDHE-RSA-AES256-GCM-SHA384 ECDHE-ECDSA-AES256-GCM-SHA384 AES256-SHA256 DHE-DSS-AES256-SHA256 DHE-DSS-AES256-SHA DHE-DSS-AES128-SHA EDH-DSS-DES-CBC3-SHA !EXP !MEDIUM !LOW !eNULL !aNULL !RC4 !DES !MD5 !PSK !SRP

Disable SSL support on listenPorts *(NoTLSlistenPorts)*

This disables TLS/SSL on the defined listenPorts, if **DoTLS** is set to "do TLS". All other SMTP listeners will support TLS/SSL, if **DoTLS** is set to "do TLS". This option works for **listenPort** , **listenPort2** and **relayPort** . The listener definition here has to be the same like in the port definitions. Separate multiple entries by "|".
Examples: 25, 127.0.0.1:25, 127.0.0.1:25|127.0.0.2:25

Force TLS to Proxy on this Ports *(TLStoProxyListenPorts)*

If a STARTTLS command is received on a port that is defined here, the connection will be moved in to the transparent proxy mode every time - independent from the setting of **DoTLS** . This option works for **listenPort** , **listenPort2** and **relayPort** . The listener definition here has to be the same like in the port definitions. Separate multiple entries by "|".
Examples: 25, 127.0.0.1:25, 127.0.0.1:25|127.0.0.2:25

SSL Certificate File (PEM format) *(SSLCertFile)*

c:/assp/certs/server-cert.pem

Full path to the file containing the server's SSL certificate or certificate-chain, for example : /usr/local/etc/ssl/certs/assp-cert.pem or c:/assp/certs/server-cert.pem. A general cert.pem file is already provided in 'assp/certs/server-cert.pem'.

SSL Key File (PEM format) *(SSLKeyFile)*

c:/assp/certs/server-key.pem
Full path to the file containing the server's SSL private key, for example: /usr/local/etc/ssl/certs/assp-key.pem or c:/assp/certs/server-key.pem. A general key.pem file is already provided in 'assp/certs/server-key.pem'

SSL Private Key Password *(SSLPKPassword)*

Optional parameter. If your private key ' **SSLKeyFile** ' is password protected, assp will need this password to decrypt the server's SSL private key file.

SSL Certificate Authority File *(SSLCaFile)*

Optional parameter to enable chained certificate validation at the client side. Full path to the file containing the server's SSL certificate authority. If you provide the ca-certificate or certificate-chain together with the certificate file in the **SSLCertFile** parameter, leave this field blank. For example : /usr/local/etc/ssl/certs/assp-ca.crt or c:/assp/certs/server-ca.crt. A general ca.crt file is already provided in 'c:/assp/certs/server-ca.crt'. The default value is empty and leave it empty as long as you don't know, how this parameter works.

Exclude these IP's from TLS* *(noTLSIP)*

Enter IP's that you want to exclude from starting SSL/TLS, separated by pipes (|). For example, put all IP's here, that making trouble to switch to TLS every time, what will prevent ASSP from getting mails from or sending mails to this hosts.

Ban Failed SSL IP *(banFailedSSLIP)*

both
If set (recommended is 'both'), an IP that fails to connect via SSL/TLS will be banned for 12 hour from using SSL/TLS.
Privat IP's and IP addresses listed in **acceptAllMail** will get one more try to correct the mistake.
This is done per default ('both'), to prevent possible DoS attacks via SSL/TLS.
Those IP's are stored in the SSL-failed-Cache. This cache is cleaned up at startup.
disable - disables this feature, which is highly NOT recommended
private only - only private IP's and IP's in **acceptAllMail** will be banned (they have two tries)
public only - only public IP's will be banned
both - private and public IP's will be banned
edit SSL-failed-Cache

Exclude these IP's from SSL-failed-Cache* *(noBanFailedSSLIP)*

Enter IP's that you want to exclude from being added to the SSL-failed-Cache, separated by pipes (|).

☐ **Send EHLO *(sendEHLO)***

If selected, ASSP sends an EHLO even if the client has sent only a HELO. This is useful to force the usage of TLS to the server or to satisfy XCLIENT/XFORWARD helo offers, because EHLO is needed before STARTTLS or XCLIENT/XFORWARD can be used.

Retry SSL on "SSL want a read first" error *(SSLRetryOnError)*

1
Define the number of SSL/TLS negotiation retries done with a half second delay after **SSLtimeout** , if the peer was not ready after STARTTLS or at the **listenPortSSL** , because of a "SSL want a read/write first" SSL handshake error.

SSL Timeout (0-999) *(SSLtimeout)*

5
SSL/TLS negotiation will timeout after this many seconds. default is : 5 seconds.

Debug Level for SSL/TLS *(SSLDEBUG)*

no Debug
Set the debug-level for SSL/TLS. Than higher the level, than more information are written to STDOUT!

☐ **Web-Client requires valid SSL Certificate for GUI Requests *(webSSLRequireCientCert)***

If enabled and **enableWebAdminSSL** is set to ON, each browser session is forced to provide a valid SSL client certificate. If no certificate is provided by the client, the connection will fail! To extend the verification of the certificate, use **SSLWEBCertVerifyCB** . Per default are used 'SSL_VERIFY_PEER | SSL_VERIFY_FAIL_IF_NO_PEER_CERT | SSL_VERIFY_CLIENT_ONCE'
To create a PKCS12 from the PEM formated cert- and key file you can use openssl, like :
openssl pkcs12 -export -clcerts -in client.pem -inkey client.key -out client.p12
The file client.p12 could now be imported in to your browser.
!!! Install a valid certificate in to your browser BEFORE you enable this option - otherwise the GUI will get inaccessible !!!
NOTICE: This option will possibly not work if you use any self signed certificate!

CallBack to Verify Web-Client Certificates for GUI Connections *(SSLWEBCertVerifyCB)*

If used, assp will call the defined subroutine as SSL->SSL_verify_callback in an eval closure submitting the original ARRAY of parameters (see the IO::Socket::SSL documentation).
The subroutine has to return 1 on certificate verification success - otherwise 0.
You can use/modify the module lib/CorrectASSPcfg.pm to implement your code. For example

```
sub checkWebSSLCert {
   my ($OpenSSLSays,$CertStackPtr,$DN,$OpenSSLError, $Cert)=@_;
   my $subject = Net::SSLeay::X509_NAME_oneline(Net::SSLeay::X509_get_subject_name($Cert));
   my $chain = Net::SSLeay::PEM_get_string_X509($Cert);
   ...any code...;
   my $success = eval{verify($Cert);};
   return $OpenSSLSays if $@;
   my $user = eval{get_owner($Cert);};
   return $OpenSSLSays if $@;
```

```
    my $pass = get_pass($user);};
    @main::ExtWebAuth = ($user,$pass)
    return $success;
}
```

Now, if you set this parameter to 'CorrectASSPcfg::checkWebSSLCert' - assp will call
CorrectASSPcfg::checkWebSSLCert->(@_);
The variable '@main::ExtWebAuth' could be used to authenticate the user to the GUI related to the used certificate. The username must be provided as first element of the array. The password could be provided as second element of the array - this is not recommended and it is not required! If the used certificate is valid and a known adminusername (root is provided) is stored as first element in '@main::ExtWebAuth', the user will be automatically logged on to the GUI.
NOTICE: This option will possibly not work if you use any self signed certificate!

Call to Configure SSL-Listener-Parameters for GUI Connections *(SSLWEBConfigure)*

If used, assp will call the defined subroutine in an eval closure submitting a reference to the assp predefined SSL-Socket-Configuration-HASH. The HASH could be modified in place to your needs - please read the documentation of IO::Socket::SSL, Net::SSLeay and OpenSSL. Return values are ignored.
You can use/modify the module lib/CorrectASSPcfg.pm to implement your code. For example

```
sub configWebSSL {
    $parms = shift;
    $parms->{timeout} = 10;
    $parms->{'SSL_check_crl'} = 1;
    $parms->{'SSL_crl_file'} = '/assp/certs/crl/crllist.pem';
    return;
}
```

Now, if you set this parameter to 'CorrectASSPcfg::configWebSSL' - assp will call
CorrectASSPcfg::configWebSSL->(\%sslparms);
NOTICE: This option will possibly not work if you use any self signed certificate!

☐ **Client requires valid SSL Certificate for STAT Requests *(statSSLRequireClientCert)***

If enabled and **enableWebStatSSL** is set to ON, each session is forced to provide a valid SSL client certificate. If no certificate is provided by the client, the connection will fail! To extend the verification of the certificate, use **SSLSTATCertVerifyCB** . Per default are used 'SSL_VERIFY_PEER | SSL_VERIFY_FAIL_IF_NO_PEER_CERT | SSL_VERIFY_CLIENT_ONCE'
NOTICE: This option will possibly not work if you use any self signed certificate!

CallBack to Verify Client Certificates for STAT Connections *(SSLSTATCertVerifyCB)*

Please read the description of **SSLWEBCertVerifyCB** .
NOTICE: This option will possibly not work if you use any self signed certificate!

Call to Configure SSL-Listener-Parameters for STAT Connections *(SSLSTATConfigure)*

If used, assp will call the defined subroutine in an eval closure submitting a reference to the assp predefined SSL-Socket-Configuration-HASH. Please follow the description for **SSLWEBConfigure** .
NOTICE: This option will possibly not work if you use any self signed certificate!

☐ **SMTP-Client requires valid SSL Certificate for SMTP SSL Connections *(smtpSSLRequireClientCert)***

If enabled, each client or server requesting a connection at the **listenPortSSL** requires a valid SSL client certificate. If no certificate is provided by the client, the connection will fail! To extend the verification of the certificate, use **SSLSMTPCertVerifyCB** . Per default are used 'SSL_VERIFY_PEER | SSL_VERIFY_FAIL_IF_NO_PEER_CERT | SSL_VERIFY_CLIENT_ONCE'
NOTICE: This option will possibly not work if you use any self signed certificate!

CallBack to Verify Client Certificates for SMTP Connections *(SSLSMTPCertVerifyCB)*

Please read the description of **SSLWEBCertVerifyCB** .
NOTICE: This option will possibly not work if you use any self signed certificate!

Call to Configure SSL-Listener-Parameters for SMTP Connections *(SSLSMTPConfigure)*

If used, assp will call the defined subroutine in an eval closure submitting a reference to the assp predefined SSL-Socket-Configuration-HASH. Please follow the description for **SSLWEBConfigure** .
NOTICE: This option will possibly not work if you use any self signed certificate!

SSL and TLS support
Notes

Global PenaltyBox Network

GPB Client Registration Name ***(globalClientName)***

The Name of this global-client for registration on the global-server. This entry has to be the full qualified DNS-Name of the IP-address over which ASSP is doing HTTP-requests! If you are using a HTTP-Proxy, this should be the public IP-address of the last Proxy in chain! This DNS-Name has to be resolvable worldwide and the resolved IP-address has to match the ASSP-HTTP-connection-IP-address. It is not possible to use an IP-address in this field! Dynamic DNS-Names like "yourdomain.dyndns.org" are supported!
To become a member of the exclusive global-penalty-box-users, you will need a subscription and you will have to pay a yearly maintenance fee. To get registered and/or to get more information, please send an email with your personal/company details and the **globalClientName** to "assp.globalpb@thockar.com".
The name of this client has to be known by the global server **before** it can be registered from here. Please wait until you have confirmation that your client name is known by the global server.
If assp is unable to connect to the GPB-server for registration, check the IP - and - clientname relation! You may also try to set this parameter to the value '**clean**' one time - this will reset all GPB-internals and GPB-configuration parameters to there default value.
Make sure, the used assp version and the perl modules are uptodate!
In addition to **Compress::Zlib** this requires an installed **LWP::UserAgent** module in PERL.

GPB Client Registration Password ***(globalClientPass)***

If the global client is registered on the global-server, you will see a number of "*" in this field. This field is readonly.

GPB Client Subscription Expiration Date ***(globalClientLicDate)***

The date of license/subscription expiration for this global client. If this date is exceeded, no upload and download of global PB will be done! This field is readonly.

☐ **Enable the Global-Black-Penalty** ***(DoGlobalBlack)***

Enables the merge of the Black-Penalty-Box-Entries, if the client is registered on the global-PB-server. Upload and download of the black penalty entries are done independent from this setting as long as any of **GPBDownloadLists** or **GPBautoLibUpdate** is activated.

Value for Global-Black-PB Entries + ***(globalValencePB)***

20

This penalty-value will be given to downloaded Black-Penalty-Box-Entries. As long as entries have the "GLOBALPB" state, they will never become extreme-Black. It is recommended to set this value above **PenaltyLimit**!

Expiration for Global-PB-Black Records ***(globalBlackExpiration)***

48

Global-Black-Penalties will expire after this number of hours.

☐ **Enable the Global-White-Penalty** ***(DoGlobalWhite)***

Enables the merge of the White-Penalty-Box-Entries, if the client is registered on the global-PB-server. Upload and download of the white penalty entries are done independent from this setting as long as any of **GPBDownloadLists** or **GPBautoLibUpdate** is activated.

Expiration for Global-PB-White Records(days) ***(globalWhiteExpiration)***

7

Global-White-Penalties will expire after this number of days.

Download List and Regex Updates from GPB-Server ***(GPBDownloadLists)***

download and install

Select, if assp should download updates for lists and regular expressions from the global penaltybox server. Downloads will be done to the 'download' folder. If install is selected, the downloaded lines will merged in to the defined files (file:...). If you want to disable a specific line in any of your files, do not delete the line, instead comment it out - putting a '#' or ';' in front of the line. If any list is not configured using the 'file:...' option, only the download will be done, even if install is selected. To disable a line that was added by the GPB-server to your file - simply commend the line out (# or ;). If you remove such a line, it could be possibly added again by the next GPB check. To change a line that was added by the GPB-server to your file - disable the line and customize a copied line to your needs.

Download Plugin and Library Updates from GPB-Server ***(GPBautoLibUpdate)***

download and install

Select, if assp should download updates for Plugins or Library-Files (../lib) from the global penaltybox server. Downloads will be done to the 'download' folder. If install is selected, the downloaded Plugins and/or modules will be installed in to there original location, if an older version of the file still exists. If an older version is not found, only the download will be done. To activate updated Plugins or modules a restart of assp is required. This feature will not force an automatic restart of assp!.

Notes On Global Penalty Box

Notes

Block Reporting - Schedule and Instant

☑ **Enable extra Logging for BlockReports** *(ExtraBlockReportLog)*

Maillogs could grow to a very large size. Enable this feature to log only loglines with blocking information to an extra file. These files will be named as "b" + **logfile** . Using this option will speed up Block Reporting. Before you switch on this option, you should run "grep"[linux/MacOS] or "find"[Windows] to create the "b" - file from the maillogs.
linux/MacOS - grep "\[spam found\]" *maillog.txt > bmaillog.txt
Windows - find "[spam found]" *maillog.txt > bmaillog.txt

Request Block Report *(EmailBlockReport)*

assp block

Any mail sent by local/authenticated users to this username will be interpreted as a request to get a report about blocked emails. Do not put the full address here, just the user part. For example: asspblock
Leading digits/numbers in the mail subject will be interpreted as "report request for the last number of days". If the number of days is not specified in the mail subject, a default of 5 days will be used to build the report.
All characters behind the "number of days" will be interpreted as a regular expression to overwrite the **BlockReportFilter** - leading and trailing white spaces will be ignored.
Users defined in **EmailBlockTo**, **EmailAdmins**, **BlockReportAdmins** and **EmailAdminReportsTo** are 'Admins' and can request a report for multiple users. They have to use a special syntax with '=>' in the body of the report request. The syntax is:
QueryAddress=>ReportRecipient=>ReportDays - there are many possible combinations of this three parameters. For example:
user@domain and user@domain=>user@domain - will send a report for this user to this user
*@domain (better use) *@domain=>* - will send a report for every blocked user in this domain to this user
user@domain=>recipient@any-domain - will send a report for user@domain to recipient@any-domain
*@domain=>recipient@any-domain - will send a report for every blocked user in this domain to recipient@any-domain
It is possible to define a group (**Groups**) in the first parameter like:
[user@domain]=>recipient@any-domain
The group name must be a lower case email address of a local domain without any wildcard. This will create a combined block report for all email addresses defined in this group - useful, if someone has multiple email addresses and wants to get a single report.
If the group name is equal to a real existing email address of a user, and this user requests a block report using this email address (MAIL FROM:), a combined block report for the group will be generated.
A third parameter is possible to set, which defines the number of days for which the report should be created. The default (if empty or not defined) is one day. This value is used to calculate the 'next run date'. For example:
*@domain=>recipient@any-domain=>2 - creates a report for two days.
@domain=>=>14 - creates a report for 14 days.
user@domain=>=>3 or user@domain=>*=>3 - creates a report for three days. The second parameter is here empty or *.
To overwrite the defined **BlockReportFilter**, you can define a fourth parameter, which contains the regular expression to use.
@domain=>=>14=>virus|newsletter - creates a report for 14 days and skips all lines that contains the words 'virus' or 'newsletter'.
If an admin emails a block report request and specifies a filter in the subject of the email and a fourth parameter in the body, both regular expressions will be merged in to a single regex for each line.
If you or a user want the default **BlockReportFilter** to become part of the overwrite regex, the literal '$BRF' should be included in the regex like:
@domain=>=>14=>virus|$BRF|newsletter - or even in the subject of the email
In this case the literal '$BRF' will be replaced by the BlockReportFilter.
Only Admins are able to request blockreports for non local email addresses. For example:
user@non_local_domain=>recipient@any-domain=>4
*@non_local_domain=>recipient@any-domain=>4
This will result in an extended blockreport for the non local address(es). Replace 'non_local_domain' with the domain name you want to query for.
It is possible to change the complete design of the BlockReports to your needs, using a html-css file. A default css-file 'blockreport.css' is in the image folder as is a default icon file 'blockreporticon.gif' and a default header-image-file 'blockreport.gif'. These are optional files - If assp can not find these files in its image folder, it will use the default hardcoded css and icon. If the file 'blockreport.gif' is not found 'logo.gif' will be used.
To change any content, use the Blockreport::modify module in the lib folder. You'll need some Perl skills to do that.

Edit blockreport_sub.txt file
Edit blockreport_html.txt file
Edit blockreport_text.txt file

Request Blocked Email Domain *(EmailBlockReportDomain)*

@assp.local

Set this to the domain to which the users can send a request to receive blocked messages. For example: @assp.local. Notice the leading required '@'!

Reply to Block-Report Request *(EmailBlockReply)*

REPLY TO SENDER

Queue User Block Report Requests *(QueueUserBlockReports)*

run instantly

How to process block report requests for users (not **EmailBlockTo**, **EmailAdmins**, **BlockReportAdmins**, **EmailAdminReportsTo**).
'run instantly' - the request will be processed instantly (not stored).
'store and run scheduled' - (deprecated) the request will be stored/queued, runs permanently scheduled at **BlockReportSchedule** until it will be removed from queue - a '+' in the subject is not needed
To add a request to queue, the user has to send an email to **EmailBlockReport**. Leading digits/numbers in the mail subject will be interpreted as "report request for the last number of days". If the number of days is not specified in the mail subject, a default of 5 days will be used to build the report.
If 'run instantly' is selected, but a user wants to schedule a permanent request, a leading '+' before the digits in subject is required.
To remove a request from queue the user has to send an email to **EmailBlockReport** with a leading '-' in the subject.

Edit user report queue

Runtime for Queued Requests [s] *(QueueSchedule)*

0

Runtime hour for reports in **QueueUserBlockReports**. Set a number between 0 and 23. 0 means midnight and is default

Forward The Blockreportrequest to other ASSP *(BlockRepForwHost)*

Edit report file: reports/blockreportforwarderror.txt

If you are using more than one ASSP (backup MX), define the IP-address and **relayPort** (x.x.x.x:ppp - for SSL use SSL:x.x.x.x:ppp) of the other ASSP here (separate multiple entries by "|"). The Blockreportrequest will be forwarded to this ASSP and the user will get a blockreport from every ASSP. The forwarded request has the same sender and recipient like the original request. So **EmailBlockReport** and **EmailBlockReportDomain** have to be configured identically on all ASSP!!!! Resend requests are automatic forwarded to the right (or next) host, if ASSP finds the hostname in the subject of the request. If you have more than two ASSP, the logical sending structure must be a star. If ASSP(A) (the sun) is in the middle and you have also ASSP(B), ASSP(C) and ASSP(D) (satellites), ASSP(A) should know C,B and D, and B,C and D should only know A.
If a forward host is unreachable, the forward request will be queued for a maximum of 24 hours and the user will be informed sending the 'reports/blockreportforwarderror.txt' file.
The perl module **Net::SMTP** is required to use this feature (for SSL - Net::SMTP::SSL is required).

Send Copy of Block-Reports TO *(EmailBlockTo)*

Email sent from ASSP acknowledging your submissions will be sent to this address. For example: admin@domain.com

BlockReport Admins* *(BlockReportAdmins)*

A list of local addresses, which have the same rights like **EmailAdmins**, but only for all BlockReport functions (nothing else). Leave this field blank (default), to disable this feature.
This is useful, if a user must request BlockReports or resend mails for other users like an EmailAdmin and BlockReportAdmin can do it, but should not have other extended rights to use the EmailInterface.
Accepts specific addresses (user@domain.com), user parts (user). Wildcards are supported (fribo*@domain.com).
For example: fribo*@thisdomain.com|jhanna

Email Admin BlockReport User and Domain Restrictions* *(EmailAdminDomains)*

Use this parameter to restrict users registered in **EmailAdmins**, **BlockReportAdmins**, **EmailAdminReportsTo** and **EmailBlockTo** to a list of domains or users, for which they can request BlockReports.
It is possible to use defined GROUPS on both sites. The file: option is required. Use the following syntax to define an entry (one per line):
EmailAdminAddress=>*@domain1,*@domain2,user@domain3,...
EmailAdminAddress1|EmailAdminAddress2=>*@domain1,*@domain2,user@domain3,...
[group_of_EmailAdminAddresses]=>*@domain1,*@domain2,user@domain3,...
[group_of_EmailAdminAddresses]=>[group_of_domains],...
Wildcards are allowed to be used only in the domain definition - like *@*.domain.tld - separate multiple domains by comma.
If an address of an EmailAdmin or BlockReportAdmin is defined multiple times, all entries are used in an "AND" logic.
If a BlockReport is requested for a not allowed email address, the complete BlockReport request will be ignored.
If an **EmailAdmins** or **BlockReportAdmins** address is not registered in this parameter, he/she is able to request BlockReports for all domains.

Blocked Email Resend Requester* *(EmailResendRequester)*

A list of local addresses, which are allowed to request a resend of blocked emails for other users, even they are not **EmailAdmins** or **BlockReportAdmins** . Leave this field blank (default), to disable this feature.
This is useful, if a user gets automatic generated BlockReports (e.g via **BlockReportFile**) for a group of users and should be able to manage resends for them. Added here, the user is not allowed to request BlockReports for other users - in this case use **EmailAdmins**, **BlockReportAdmins** and **EmailAdminDomains** instead.
The resend is done to the recipient stored in the X-Assp-Intended-For: (requires **AddIntendedForHeader**) header field and the requester, if the address was found in a TO: header field.
Accepts specific addresses (user@domain.com), user parts (user). Wildcards are supported (fribo*@domain.com).
For example: fribo*@thisdomain.com|jhanna

File for Blockreportrequest *(BlockReportFile)*

A file with BlockReport requests. ASSP will generate a block report for every line in this file (file:files/blockreportlist.txt - file: is required if defined!) every day at midnight for the last day. The perl modules **Net::SMTP** and **Email::MIME** are required to use this feature. A report will be only created, if there is at least one blocked email found! The syntax is:
QueryAddress=>ReportRecipient=>ReportDays - there are many possible combinations of this three parameters. For example:
user@domain and user@domain=>user@domain - will send a report for this user to this user
*@domain (better use) *@domain=>* - will send a report for every blocked user in this domain to this user
@ - creates a report for all local users in all local domains
user@domain=>recipient@any-domain - will send a report for user@domain to recipient@any-domain
*@domain=>recipient@any-domain - will send a report for every blocked user in this domain to recipient@any-domain
It is possible to define a group (**Groups**) in the first parameter like:
[user@domain]=>recipient@any-domain
The group name must be a lower case email address of a local domain without any wildcard. This will create a combined block report for all email addresses defined in this group - useful, if someone has multiple email addresses and want's to get a single report.
An optional third parameter can define the number of days for which the report should be created. The default (if empty or not defined) is one day. This value is used to calculate the 'next run date'. For example:
*@domain=>recipient@any-domain=>2 - creates a report for two days.
@domain=>=>14 - creates a report for 14 days.
user@domain=>=>3 or user@domain=>*=>3 - creates a report for three days. The second parameter is here empty or *!
To overwrite the defined **BlockReportFilter**, you can define a fourth parameter, which contains the regular expression to use.
@domain=>=>14=>virus|newsletter - creates a report for 14 days and skips all lines that contains the words 'virus' or 'newsletter'.
A fifth parameter could be used to schedule (cron) a BlockReport. If this parameter is used, the line will be ignored at **BlockReportSchedule**.
For the syntax of the cron entry, please read **RebuildSchedule** . Multiple schedules in one line could be separated by pipe (|).
*@domain=>it_dep@domain=>7=>virus|newsletter=>0 0 * * 0 - creates a report every Sunday at 00:00 for the last seven days
*@domain=>it_dep@domain=>2=>virus|newsletter=>0 0 * * 2,4,6|0 12 * * 1 - creates a report every Tuesday,Thursday,Saturday at 00:00 and at every Monday at 12:00 for the last two days
Only Admins are able to request blockreports for non local email addresses. For example:
user@non_local_domain=>recipient@any-domain=>4
*@non_local_domain=>recipient@any-domain=>4
This will result in an extended blockreport for the non local address(es). Replace 'non_local_domain' with the domain name you want to query for.

Runtime BlockReportFile [s] *(BlockReportSchedule)*

0

Runtime hour for reports in **BlockReportFile**. Set a number between 0 and 23. 0 means midnight and is default.

☐ **Generate a BlockReport from BlockReportFile Now** *(BlockReportNow)*

If selected, ASSP will generate a block report from **BlockReportFile** now. Apply Changes and Run Block Report Now (if checked) | Refresh Browser

Max Search time per log File *(BlockMaxSearchTime)*

0

The maximum time in seconds, the Blockreport feature spends on searching in one log file. If this value is reached, the next log file will be processed. Default is 0. A value of 0 disables this feature and all needed log files will be fully processed.

The format of the Report Email *(BlockReportFormat)*

text and html

Block reports will be sent as multipart/alternative MIME messages. They normally contains two parts, a plain text part and a html part. Select "text only" or "html only" if you want to skip any of this parts.
To make it possible to detect a resent email, ASSP will add a header line "X-Assp-Resend-Blocked: **myName**" to each email!

My HTTP Name *(BlockReportHTTPName)*

The hostname for HTTP(S) links in AdminUsers Blockreports. If not defined the local hostname will be used. do NOT define an IP address here!

Regular Expression to Skip Log Records* *(BlockReportFilter)*

Virus|BlackDomain

Put anything here to identify messages which should not be reported in any Block Report. For example: Virus|BlackDomain.
For individual filter settings, it is possible to overwrite this value in the **BlockReportFile** for every single line and in every request per email using the subject line (read **EmailBlockReport**).

☐ **Collect multiple TopTen Statistics** *(DoT10Stat)*

enable the top ten statistic count (blocked IP's, blocked senders, blocked recipients) and the output in the GUI and BlockReports for admins.

Include a Resend-Link for every resendable email *(inclResendLink)*

in both

Block reports will be sent as multipart/alternative MIME messages. They contains two parts, a plain text part and a html part. If a blocked email is stored in any folder, it is possible to include a link for each email in to the report. Define here what you want ASSP to do. Default is "in both". If set to not to disabled " **fileLogging** " will be automatically set to on.

Which Link Should be included *(BlockResendLink)*

both

If HTML is enabled in **inclResendLink**, two links (one on the left and one on the right site) will be included in the report email by default. Depending on the used email clients it could be possible, that one of the two links will not work for you. Try out what link is working and disable the other one, if you want.

User which get the Left link only* *(BlockResendLinkLeft)*

List of users and domains that will get the left link only. The setting for **BlockResendLink** will be ignored for this entries!

User which get the right link only* *(BlockResendLinkRight)*

List of users and domains that will get the right link only. The setting for **BlockResendLink** will be ignored for this entries!

☑ **Delete Mails in Spam Folder** *(DelResendSpam)*

If selected, a user request to resend a blocked email will delete the file in the **spamlog** folder - an admin request will move the file to the **correctednotspam** folder.

Automatic add Resend Senders to Whitelist *(autoAddResendToWhite)*

no

If a BlockReport resend request is made by any of the selected users, the original sender of the resent mail will be added to whitelist, also a copy file to the resend folder will do that.
Notes On Block Reporting

Notes

SNMP Configuration

Enable the ASSP-SNMP Interface *(SNMP)*

disable

This enables the AgentX registration of assp to a **SNMP** master-AgentX. ASSP will be registered to the master-AgentX as 'assp_myName', the possible configuration file name will be assp_myName.conf . This option requires the installed perl module **NetSNMP::agent**. The product and needed librarys could be downloaded at **net-snmp.org**.
All configuration values are accessed using the **SNMPUser** account. The SNMP-permission and visibility is used from the configured user GUI-permissions.

The following OIDs (relative to the **SNMPBaseOID**) are available for SNMP-queries. The configuration values are changeable via snmp. The file mib/ASSP-MIB could be used in **SNMP** browsers to get a human readable view of the OID's (copy it to the net-snmp MIB file location - eg: [C:]/usr/share/snmp/mibs and the MIB location of your **SNMP** browser). Please keep in mind, that an extensive usage of **SNMP** queries will slow down assp.

.1 - runtime information
.1.0 - assp healthy status boolean 0/1
.1.1 - assp healthy status text
.1.2 - ASSP runtime status boolean 0/1 0=shutdown in progress - 1=running
.1.3 - ASSP runtime status text
.1.4 - ASSP version string
.1.5 - ASSP script name
.1.6 - Perl version string
.1.7 - Perl executable name
.1.8 - operating system name
.1.9 - hostname where ASSP is running on
.1.10 - IP-host where ASSP is running on
.1.11 - myName
.1.12 - URL to new ASSP version download
.1.13 - currently running tasks
.1.14 - current assp memory usage in MB
.1.20 - schedule information
.1.20.1 - next BerkeleyDB sync
.1.20.2 - next scheduled Config reload
.1.20.3 - next BATVTag cache cleaning
.1.20.4 - next general cache cleaning
.1.20.5 - next IP-per-Domain cache cleaning
.1.20.6 - next DelayDB cache cleaning
.1.20.7 - next Penaltybox cache cleaning
.1.20.8 - next Database Backup
.1.20.9 - next Database Connection Check
.1.20.10 - next DNS Connection Check
.1.20.11 - next hourly job runs (at)
.1.20.12 - next Database Export
.1.20.13 - next upload for Global-Black
.1.20.14 - next upload for Global-White
.1.20.15 - next Hash-File-Check (option files)
.1.20.16 - next LDAP-cross-Check
.1.20.17 - next RebuildSpamDB
.1.20.18 - next ResendMail
.1.20.19 - next ASSPFileDownload (assp.pl)
.1.20.20 - next Version File Download (version.txt)
.1.20.21 - next BackDNS File Download
.1.20.22 - next Code Change Check
.1.20.23 - next Droplist Download
.1.20.24 - next **Griplist** Download
.1.20.25 - next POP3Collect
.1.20.26 - next Save Stats
.1.20.27 - next TLDlist Download
.1.20.28 - next Sync Config
.1.20.29 - next **Groups** File Reload
.1.20.30 - next BlockReport Schedule
.1.20.31 - next File Age Schedule
.1.20.32 - next BlockReport Queue Schedule

.1.30.X - worker status (boolean) X = worker
.1.30.X.1 - worker time since last loop (text) X = worker
.1.30.X.2 - worker last action (text) X = worker

.1.31.0 - general database status (boolean) 0/1
.1.31.0.1 - general database status (text)
.1.31.X - database table status (boolean) 0/1 - X >= 1
.1.31.X.1 - database table name - X >= 1 related to .1.31.X

.2 - Configuration - X is the internal value number adapted from the language files
.2.H - heading description - H is the internal GUI heading number
.2.H.X - config value

.3 - assp module information - X is a counter up from zero
.3.X - module name
.3.X.1 - installed module version
.3.X.2 - required module version
.3.X.3 - module installation status
.3.X.4 - download URL for the module

.4 - assp runtime status
.4.1 - current stat - X is a counted number
.4.1.X - current stat value

.4.2 - cumulative stat - X is a counted number
.4.2.X - cumulative stat value

.4.3 - current total stat - X is a counted number
.4.3.X - current total value

.4.4 - cumulative total stat - X is a counted number
.4.4.X - cumulative total stat value
.4.5 - current scoring stat - X is a counted number
.4.5.X - current scoring stat value

.4.6 - cumulative scoring stat - X is a counted number
.4.6.X - cumulative scoring stat value
.5.0 - SNMP-API : is writeable - accepts internal subroutine command/call to be executed
.5.1 - the result of the last SNMP-API call (success or error)

SNMP Base OID *(SNMPBaseOID)*

.1.3.6.1.4.1.37058.2

The Base OID that should be used by assp. This OID will be registered to the master-AgentX. The master-AgentX will then redirect all requests for this OID and sub OID's to assp! The default setting .1.3.6.1.4.1.37058.2 is needed to use the MIB file mib/ASSP-MIB in **SNMP** browsers.

How to return Boolean Values *(SNMPreturnBOOL)*

ASN_BOOLEAN

How should assp return boolean values for status OIDs. Use another setting than the default ASN_BOOLEAN, if your **SNMP** application or browser does not understand it!

ASSP User Account used for SNMP Requests *(SNMPUser)*

root

The Admin Users account used for **SNMP** requests. If the user does no longer exists, the root account will be used!

Allow Config Changes via SNMP *(SNMPwriteable)*

allow

Allow configuration changes via **SNMP**. Do not forget to setup your **SNMP** configuration file to secure the access to **SNMP**. All configuration changes via **SNMP** are done using the **SNMPUser** account!

The Socket use to connect to the master-AgentX *(SNMPAgentXSocket)*

tcp:localhost:705

How to connect to the master-AgentX. Please read the **net-snmp** documentation for more details.
Notes On SNMP

Notes

POP3 Collecting

POP3 Configuration File* ***(POP3ConfigFile)***

file:files/pop3cfg.txt [Edit file]

The file with a valid POP3 configuration. Only the file: option is allowed to use.
If the file exists and contains at least one valid POP3 configuration line and **POP3Interval** is configured, assp will collect the messages from the configured POP3-servers.
Each line in the config file contains one configuration for one user.
All spaces will be removed from each line.
Anything behind a # or ; is consider a comment.
If the same POP3-user-name is used multiple times, put two angles with a unique number behind the user name. The angles and the number will be removed while processing the configuration.
e.g: pop3user<1> will result in pop3user - or - myName@pop3.domain<12> will result in myName@pop3.domain
It is possible to define commonly used parameters in a separate line, which begins with the case sensitive POP3-username "COMMON:=" - followed by the parameters that should be used for every configured user.
A commonly set parameter could be overwritten in every user definition.
Each configuration line begins with the POP3-username followed by ":=" : e.g myPOP3userName:=
This statement has to be followed by pairs of parameter names and values which are separated by commas (,) - the pairs inside are separated by an equal sign (=).

examples:
user@gmail.com:=POP3password=pop3_pass,POP3server=pop.gmail.com:995,SMTPsendto=demo@demo_smtp.local,POP3SSL=1,.....
user1<1>:=POP3password=pop3_pass,POP3server=pop3.server.com:110,SMTPsendto=demo@demo_smtp.local,.....
user1<2>:=POP3password=pop3_pass,POP3server=pop3.server2.com:110,SMTPsendto=demo@demo_smtp.local,.....

The following case sensitive keywords are supported in the configuration file:

POP3password=pop3_password
POP3server=POP3-server or IP[:Port]
SMTPsender=email_address
SMTPsendto=email_address or <TO:> or <TO:email_address>
SMTPserver=SMTP-server[:Port]
SMTPHelo=myhelo
SMTPAUTHuser=smtpuser
SMTPAUTHpassword=smtppass
POP3SSL=0/1
SIZElimit=maximum number of bytes in a single message

POP3SSL, SIZElimit, SMTPHelo, SMTPsender, SMTPAUTHuser and SMTPAUTHpassword are optional.
If POP3SSL is set to 1 - POP3S will be done! The Perl module **IO::Socket::SSL** is required for POP3S!
If SIZElimit is exceeded by a single message, the message will not be collected and a notification email will be sent to the recipient.
If SMTPsender is not defined, the FROM: address from the header line will be used - if this is not found the POP3username will be used.
If the <TO:> syntax is used for SMTPsendto, the mail will be sent to any recipient that is found in the "to: cc: bcc:" header lines if it is a local one.
If the <TO:email_address> syntax is used for SMTPsendto, the literals NAME and/or DOMAIN will be replaced by the name part and/or domain part of the addresses found in the "to: cc: bcc:" header lines. This makes it possible to collect POP3 mails from a POP3 account, which holds mails for multiple recipients.
For example: <TO:NAME@mydomain.com> or <TO:NAME@subdomain.DOMAIN> or <TO:central-account@DOMAIN>
If the <TO:> or <TO:email_address> syntax is used for SMTPsendto, "**localDomains**" and/or "**LocalAddresses_Flat**" must be configured to prevent too much error for wrong recipients defined in the "to: cc: bcc:" header lines. The POP3collector will not do any LDAP or VRFY query!
If you want assp to detect SPAM, use the **listenPort** or **listenPort2** as SMTP-server.
NOTICE: that the following characters and white spaces are not allowed to be used in the POP3password and SMTPAUTHpassword definitions: , = #
To use this feature, you have to install the perl script "assp_pop3.pl" in the assp- **base** directory.

POP3 Collecting Interval [s] ***(POP3Interval)***

0

The interval in minutes, assp should collect messages from the configured POP3-servers. A value of zero disables this feature.

☐ **POP3 Collector forks to a new Process** ***(POP3fork)***

If selected, the POP3 collection will be started in a new process (fork). This prevents the MaintThread from waiting until the POP3 collection has finished. Do not select this option, if you are testing the POP3 collection - to get all output from the collector! It is recommended to set this option after you've verified that the POP3 collector is running well.

☐ **POP3 Keep Rejected Mails on POP3 Server** ***(POP3KeepRejected)***

If selected, any collected POP3 mail that fails to be sent via SMTP (because of being SPAM - in case rejected by the SMTP server) will be kept on the POP3 server.

☐ **POP3 debug** ***(POP3debug)***

If selected, the POP3 collection will write **debug** output to the log file. Do not use it, unless you have problems with the POP3 collection!
Notes On POP3 collecting
[Notes]

Perl Module Setup

☑ **Use Module ASSP_FC** *(useASSP_FC)*

If selected, the perl module ASSP_FC will be loaded if it is installed. If not selected, ASSP will not load the perl module ASSP_FC even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module ASSP_SVG** *(useASSP_SVG)*

If selected, the perl module ASSP_SVG will be loaded if it is installed. If not selected, ASSP will not load the perl module ASSP_SVG even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **enableGraphStats** and maybe some other features.

☑ **Use Module ASSP_WordStem** *(useASSP_WordStem)*

If selected, the perl module ASSP_WordStem will be loaded if it is installed. If not selected, ASSP will not load the perl module ASSP_WordStem even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module AsspSelfLoader** *(useAsspSelfLoader)*

If selected, the perl module AsspSelfLoader will be loaded if it is installed. If not selected, ASSP will not load the perl module AsspSelfLoader even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Authen::SASL** *(useAuthenSASL)*

If selected, the perl module Authen::SASL will be loaded if it is installed. If not selected, ASSP will not load the perl module Authen::SASL even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **relayAuthUser** and maybe some other features.

☑ **Use Module BerkeleyDB** *(useBerkeleyDB)*

If selected, the perl module BerkeleyDB will be loaded if it is installed. If not selected, ASSP will not load the perl module BerkeleyDB even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **DoHMM** **HMMusesBDB** **downloadBackDNSFile** **spamdb** **useDB4griplist** **DBdriver** **importDBDir** **AsADaemon** **OrderedTieHashTableSize** **useDB4IntCache** **useDB4Rebuild** **SNMP** and maybe some other features.

☑ **Use Module Compress::Zlib** *(useCompressZlib)*

If selected, the perl module Compress::Zlib will be loaded if it is installed. If not selected, ASSP will not load the perl module Compress::Zlib even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **AutoUpdateASSP** **EnableHTTPCompression** **globalClientName** and maybe some other features.

☐ **Use Module Convert::TNEF** *(useConvertTNEF)*

If selected, the perl module Convert::TNEF will be loaded if it is installed. If not selected, ASSP will not load the perl module Convert::TNEF even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **doInFixTNEF** **doOutFixTNEF** and maybe some other features.

☐ **Use Module DB_File** *(useDB_File)*

If selected, the perl module DB_File will be loaded if it is installed. If not selected, ASSP will not load the perl module DB_File even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **useDB4Rebuild** and maybe some other features.

☑ **Use Module Digest::MD5** *(useDigestMD5)*

If selected, the perl module Digest::MD5 will be loaded if it is installed. If not selected, ASSP will not load the perl module Digest::MD5 even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **DelayMD5** and maybe some other features.

☑ **Use Module Digest::SHA1** *(useDigestSHA1)*

If selected, the perl module Digest::SHA1 will be loaded if it is installed. If not selected, ASSP will not load the perl module Digest::SHA1 even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **DoMSGIDsig** **DoBATV** and maybe some other features.

☑ **Use Module Email::MIME** *(useEmailMIME)*

If selected, the perl module Email::MIME will be loaded if it is installed. If not selected, ASSP will not load the perl module Email::MIME even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **ValidateURIBL** **DoBlockExes** **EmailSpam** **EmailHam** **EmailAnalyze** **UseUnicode4MaillogNames** **UseUnicode4SubjectLogging** **inChrSetConv** **outChrSetConv** **doInFixTNEF** **doOutFixTNEF** **BlockReportFile** and maybe some other features.

☑ **Use Module Email::Send** *(useEmailSend)*

If selected, the perl module Email::Send will be loaded if it is installed. If not selected, ASSP will not load the perl module Email::Send even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **resendmail** and maybe some other features.

☑ **Use Module File::ReadBackwards** *(useFileReadBackwards)*

If selected, the perl module File::ReadBackwards will be loaded if it is installed. If not selected, ASSP will not load the perl module File::ReadBackwards even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module File::Scan::ClamAV** ***(useFileScanClamAV)***

If selected, the perl module File::Scan::ClamAV will be loaded if it is installed. If not selected, ASSP will not load the perl module File::Scan::ClamAV even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **UseAvClamd** and maybe some other features.

☑ **Use Module IO::Socket::INET6** ***(useIOSocketINET6)***

If selected, the perl module IO::Socket::INET6 will be loaded if it is installed. If not selected, ASSP will not load the perl module IO::Socket::INET6 even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **enableINET6** and maybe some other features.

☑ **Use Module IO::Socket::SSL** ***(useIOSocketSSL)***

If selected, the perl module IO::Socket::SSL will be loaded if it is installed. If not selected, ASSP will not load the perl module IO::Socket::SSL even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **syncUsesSSL** **smtpDestination** **smtpDestinationSSL** **smtpAuthServer** **relayHost** **DoVRFY** **enableTLS4VRFY** **EmailReportDestination** **DoLDAPSSL** **enableWebAdminSSL** **enableWebStatSSL** **DoTLS** **SSL_version** **SSL_cipher_list** **SSLWEBCertVerifyCB** **SSLWEBConfigure** **POP3ConfigFile** and maybe some other features.

☑ **Use Module LWP::Simple** ***(useLWPSimple)***

If selected, the perl module LWP::Simple will be loaded if it is installed. If not selected, ASSP will not load the perl module LWP::Simple even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module MIME::Types** ***(useMIMETypes)***

If selected, the perl module MIME::Types will be loaded if it is installed. If not selected, ASSP will not load the perl module MIME::Types even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **doInFixTNEF** **doOutFixTNEF** and maybe some other features.

☑ **Use Module Mail::DKIM::Verifier** ***(useMailDKIMVerifier)***

If selected, the perl module Mail::DKIM::Verifier will be loaded if it is installed. If not selected, ASSP will not load the perl module Mail::DKIM::Verifier even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **DoDKIM** and maybe some other features.

☑ **Use Module Mail::SPF** ***(useMailSPF)***

If selected, the perl module Mail::SPF will be loaded if it is installed. If not selected, ASSP will not load the perl module Mail::SPF even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **ValidateSPF** **SPF2** **LocalPolicySPF** and maybe some other features.

☐ **Use Module Mail::SPF::Query** ***(useMailSPFQuery)***

If selected, the perl module Mail::SPF::Query will be loaded if it is installed. If not selected, ASSP will not load the perl module Mail::SPF::Query even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **SPF2** **LocalPolicySPF** and maybe some other features.

☐ **Use Module Mail::SRS** ***(useMailSRS)***

If selected, the perl module Mail::SRS will be loaded if it is installed. If not selected, ASSP will not load the perl module Mail::SRS even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **EnableSRS** and maybe some other features.

☑ **Use Module Net::CIDR::Lite** ***(useNetCIDRLite)***

If selected, the perl module Net::CIDR::Lite will be loaded if it is installed. If not selected, ASSP will not load the perl module Net::CIDR::Lite even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Net::DNS** ***(useNetDNS)***

If selected, the perl module Net::DNS will be loaded if it is installed. If not selected, ASSP will not load the perl module Net::DNS even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **ValidateRWL** **DoReversed** **DoInvalidPTR** **ValidateRBL** **ValidateURIBL** **DoBackSctr** and maybe some other features.

☑ **Use Module Net::IP** ***(useNetIP)***

If selected, the perl module Net::IP will be loaded if it is installed. If not selected, ASSP will not load the perl module Net::IP even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Net::LDAP** ***(useNetLDAP)***

If selected, the perl module Net::LDAP will be loaded if it is installed. If not selected, ASSP will not load the perl module Net::LDAP even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **ldLDAP** **DoLDAP** and maybe some other features.

☑ **Use Module Net::SMTP** *(useNetSMTP)*

If selected, the perl module Net::SMTP will be loaded if it is installed. If not selected, ASSP will not load the perl module Net::SMTP even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **enableCFGShare** **syncUsesSSL** **localDomains** **BlockRepForwHost** **BlockReportFile** and maybe some other features.

☑ **Use Module Net::SMTP::SSL** *(useNetSMTPSSL)*

If selected, the perl module Net::SMTP::SSL will be loaded if it is installed. If not selected, ASSP will not load the perl module Net::SMTP::SSL even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **syncUsesSSL** **BlockRepForwHost** and maybe some other features.

☑ **Use Module NetAddr::IP::Lite** *(useNetAddrIPLite)*

If selected, the perl module NetAddr::IP::Lite will be loaded if it is installed. If not selected, ASSP will not load the perl module NetAddr::IP::Lite even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☐ **Use Module NetSNMP::agent** *(useNetSNMPagent)*

If selected, the perl module NetSNMP::agent will be loaded if it is installed. If not selected, ASSP will not load the perl module NetSNMP::agent even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **SNMP** and maybe some other features.

☑ **Use Module PerlIO::scalar** *(usePerlIOscalar)*

If selected, the perl module PerlIO::scalar will be loaded if it is installed. If not selected, ASSP will not load the perl module PerlIO::scalar even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Regexp::Optimizer** *(useRegexpOptimizer)*

If selected, the perl module Regexp::Optimizer will be loaded if it is installed. If not selected, ASSP will not load the perl module Regexp::Optimizer even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Schedule::Cron** *(useScheduleCron)*

If selected, the perl module Schedule::Cron will be loaded if it is installed. If not selected, ASSP will not load the perl module Schedule::Cron even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **ReStartSchedule** **MemoryUsageCheckSchedule** **RebuildSchedule** and maybe some other features.

☑ **Use Module Sys::CpuAffinity** *(useSysCpuAffinity)*

If selected, the perl module Sys::CpuAffinity will be loaded if it is installed. If not selected, ASSP will not load the perl module Sys::CpuAffinity even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **asspCpuAffinity** and maybe some other features.

☑ **Use Module Sys::MemInfo** *(useSysMemInfo)*

If selected, the perl module Sys::MemInfo will be loaded if it is installed. If not selected, ASSP will not load the perl module Sys::MemInfo even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Sys::Syslog** *(useSysSyslog)*

If selected, the perl module Sys::Syslog will be loaded if it is installed. If not selected, ASSP will not load the perl module Sys::Syslog even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **sysLog** and maybe some other features.

☑ **Use Module Text::Unidecode** *(useTextUnidecode)*

If selected, the perl module Text::Unidecode will be loaded if it is installed. If not selected, ASSP will not load the perl module Text::Unidecode even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **DoTransliterate** and maybe some other features.

☑ **Use Module Thread::State** *(useThreadState)*

If selected, the perl module Thread::State will be loaded if it is installed. If not selected, ASSP will not load the perl module Thread::State even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **WorkerCPUPriority** and maybe some other features.

☑ **Use Module Tie::RDBM** *(useTieRDBM)*

If selected, the perl module Tie::RDBM will be loaded if it is installed. If not selected, ASSP will not load the perl module Tie::RDBM even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **myhost** and maybe some other features.

☑ **Use Module Unicode::GCString** *(useUnicodeGCString)*

If selected, the perl module Unicode::GCString will be loaded if it is installed. If not selected, ASSP will not load the perl module Unicode::GCString even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☐ **Use Module Win32::API::OutputDebugString** ***(useWin32APIOutputDebugString)***

If selected, the perl module Win32::API::OutputDebugString will be loaded if it is installed. If not selected, ASSP will not load the perl module Win32::API::OutputDebugString even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Win32::Daemon** ***(useWin32Daemon)***

If selected, the perl module Win32::Daemon will be loaded if it is installed. If not selected, ASSP will not load the perl module Win32::Daemon even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!

☑ **Use Module Win32::Unicode** ***(useWin32Unicode)***

If selected, the perl module Win32::Unicode will be loaded if it is installed. If not selected, ASSP will not load the perl module Win32::Unicode even it is installed and several features of ASSP will not be available! It is recommended to disable installed but unused modules to reduce the required memory. Requires ASSP restart!
This module is possibly used for **UseUnicode4MaillogNames** and maybe some other features.

ASSP_AFC-Plugin

Do the ASSP_AFC Plugin *(DoASSP_AFC)*

disabled

This plugin is an addon to the default attachment- and ClamAV- engine of ASSP. The default engines only scannes the first MaxBytes/ClamAVBytes of an email. If you enable this plugin, the complete mail will be scanned for bad attachments and/or viruses!
The default engine(s) will be disabled by this enhanced version. Before you enable this plugin, please go to the configuration section(s) and configure the values for attachments and/or ClamAV! This plugin requires an installed **Email::MIME** module in PERL.
This plugin is designed for- and running in call/run level 'complete mail'!

Select the ASSP_AFC Plugin Action *(ASSP_AFCSelect)*

do both

If you enable one or both options of this plugin, the complete mail will be scanned for bad attachments and/or viruses!

the priority of the Plugin *(ASSP_AFCPriority)*

6

Sets the priority of this Plugin within the call/run-level 'complete mail'. The Plugin with the lowest priority value is processed first!

☐ **Block Encrypted Compressed Attachments *(ASSP_AFCblockEncryptedZIP)***

If set, encrypted or password protected compressed attachments will be blocked or replaced according to **ASSP_AFCSelect** and **ASSP_AFCReplBadAttach** . This setting is a general switch - an override can be done using **UserAttach** !

Analyzing Compressed Attachments

Independend from the setting of **ASSP_AFCblockEncryptedZIP** this plugin provides several mechanism to analyze compressed attachments.
To enable the compressed attachment processing, **UserAttach** has to be configured!
To analyze compressed attachments, configure '**UserAttach**'. This plugin enhances the definiton options for **UserAttach**. In addition to the existing options, the following syntax could be used:
For example:
zip:user@domain.tld => good => ai|asc|bhx|dat|doc|eps|zip
zip:*@domain.tld => good => ai|asc|bhx , good-out => eps|gif , good-in => htm|html , block => pdf|ppt , block-out => rar|rpt , block-in => xls|exe\-bin|:MSOM|crypt\-zip|encrypt

Those definitions (notice the leading zip:) are only used inside compressed files.
The extension 'crypt-zip' could be used to allow or deni encrypted compressed attachments for users at any compression level.
The extension 'encrypt' could be used to allow or deni encrypted (eg. aes) for users.

If 'exe-bin' is defined, the Plugin will detect executable files based on there binary content. Detected will be all executables, libraries and scripts for DOS and Windows (except .com files), MS office macros(VBA), MAC-OS and linux ELF (for all processor architectures).
If you want to skip the detection for a specific executable type, specify exe-bin (which detects all executables) and then add exceptions to exclude specific types:Example: 'exe-bin|:MSOM|:WSH' - notice the single leading collon for the exceptions! This example will block all detected executable files except for MS Office Macro files (:MSOM) and Windows Shell Scripts (:WSH)

:WIN - windows executables
:MOS - Mach-O executables
:PEF - Classic MacOS executables
:ELF - ELF (linux) executables
:WSH - windows shell scripts
:MMC - windows MMC Console Files
:ARC - static library (linux,unix)
:CSC - common scripts (basic,java,perl,php,powershell....)
:MSOM - microsoft office macros

The following compression formats are supported by the common perl module Archive::Extract:
tar.gz,tgz,gz,tar,zip,jar,ear,war,par,tbz,tbz2,tar.bz,tar.bz2,bz2,Z,lzma,txz,tar.xz,xz.
The detection of compressed files is done content based not filename extension based. The perl modules File::Type and MIME::Types are required in every case!
Depending on your Perl distribution, it could be possible that you must install additionally 'IO::Compress::...' (for example: IO::Compress:Lzma) modules to support the compression methodes with Archive::Extract.
If the perl module Archive::Rar and a rar or unrar binary for your OS are installed (in PATH), the RAR format is also supported.
If the perl module Archive::Rar and a 7z/7za/7zip or p7zip executable is available at the system (in PATH), the following formats are supported: 7z, XZ, BZIP2, BZ2, GZIP, GZ, TAR.GZ, TAR, ZIP, WIM, AR, ARJ, CAB, CHM, CPIO, CramFS, DMG, EXT, FAT, GPT, HFS, IHEX, ISO, LHA, LZH, LZMA, MBR, MSI, NSIS, NTFS, QCOW2, RAR, RPM, SquashFS, UDF, UEFI, VDI, VHD, VMDK, WIM, XAR, Z.
If the perl module Archive::Libarchive::XS is available , the following formats are supported: 7z, XZ, BZIP2, BZ2, GZIP, GZ, TAR.GZ, TAR, ZIP, WIM, AR, ARJ, CPIO, EXT, IHEX, ISO, LHA, LZH, LZMA, NSIS, QCOW2, RAR, RPM, SquashFS, UDF, XAR, Z.

For performance reasons it is strongly recommended to install the module Archive::Libarchive::XS!
Currently supported compression formats are: 7z, 7zip, AR, ARJ, BZ2, BZIP2, CPIO, EAR, EXT, GZIP, GZ, IHEX, ISO, JAR, LBR, LHA, LRZ, LZ, LZ4, LZH, LZMA, LZR, NSIS, PAR, PAX, QCOW2, RAR, RPM, SquashFS, TAR, TBZ, TBZ2, UDF, WAR, XAR, Z, ZIP
Detected decompression executables are: C:\Perl\site\bin\rar.EXE , C:\Perl\site\bin\7z.EXE , libarchive 3.2.0
If multiple options are available to decompress a file, ASSP_AFC will use the following order: first Archive::Libarchive::XS, than Archive::Extract, than Archive::Rar + rar/unrar and last Archive::Rar + 7z.
Notice: you need to restart assp after installing any perl module and/or exexutable, to get them activated!

Maximum Decompression Level *(ASSP_AFCMaxZIPLevel)*

10

The maximum decompression cycles use on a compressed attachment (eg: zip in zip in zip ...). Default value is 10 - zero is not allowed to be used!

☐ **Replace Bad Attachments *(ASSP_AFCReplBadAttach)***

If set and AttachmentBlocking is set to block, the mail will not be blocked but the bad attachment will be replaced with a text!

Replace Bad Attachments Text *(ASSP_AFCReplBadAttachText)*

The attached file (FILENAME) was removed from this email by ASSP for policy reasons! The file was det

The text which replaces the bad attachment. The litteral FILENAME will be replaced with the name of the bad attachment! The litteral REASON will be replaced with the reason, because the attachment was rejected!

☐ **Replace Virus Parts** ***(ASSP_AFCReplViriParts)***

If set and virus scanning (UseClamAV) is enabled, the mail will not be blocked but the bad attachment or mail part will be replaced with a text!

Replace Virus Parts Text ***(ASSP_AFCReplViriPartsText)***

There was a virus (VIRUS) removed from this email (attachment FILENAME) by ASSP!

The text which replaces the bad mailparts that contains a virus. The litteral FILENAME will be replaced with the name of a bad attachment! The litteral VIRUS will be replaced with the name of the virus!

Increase MSG-Score on MSG Size ***(ASSP_AFCMSGSIZEscore)***

You can increase the message score of a mail because of its size (in byte). Define the size and scores in a comma separated list using the syntax 'size=>score[,othersize=>otherscore]'. The list will be processed in reversed numerical order of the size value. If the size of a mail is equal or higher as the defined size, the associated message score will be added. An possible definition could be:

500000=>10,1000000=>5,1500000=>0

which meens: if the message size is >= 1500000 byte no score will be added
if the message size is >= 1000000 byte and < 1500000 byte a score of 5 will be added
if the message size is >= 500000 byte and < 1000000 byte a score of 10 will be added
if the message size is < 500000 byte no score will be added.

This feature will not process incomming mails, whitelisted mails and mail that are noprocessing - except mails, that are noprocessing only because of there message size (**npSize**).

Detect Spam Attachments* ***(ASSP_AFCDetectSpamAttachRe)***

image\/

An regular expression used on the "Content-Type" header tag to detect MIME parts that should be checked to be known spam or not. The rebuildspamdb task will build **spamdb** entries for these attachements and inlines (in assp build 12022 and higher). The plugin will block an email, if a bad attachment is found and was not removed/replaced by any other rule in this plugin. Leave this blank to disable the feature.
for example:

image\/
application\/pd[ft]
application\/zip

Script to move large attachments to a web server ***(ASSP_AFCWebScript)***

If the size of an undecoded attachment exceeds the **ASSP_AFCinsize** or **ASSP_AFCoutsize** parameter, assp will call this script and will replace the attachment with the text returned by this script or executable.
If no text is returned by the script (a warning is written to the maillog.txt) or the returned text begins with the word "error", the attachment will not be replaced.
The script has to write the resulting text or error to STDOUT.
The resulting text could be any of plain text or html code. The MIME-enconding and the Content-Type value of the resulting MIME-part will be set accordingly.
The text should contain the link to download the attachment, possibly some explanation (eg. download life time), web login information or a web-session-identifier - what ever is needed to fit the requirements of your web server.
You have to define the full path to the script and all parameters that should be pass to the script. The literal FILENAME will be replaced with the attachment filename (including the full path) that was stored in the /transfer folder. Any literal starting with an '$', will be replaced by the according connection hash value or the global variable with the name.

for example:
$relayok will be replaced by $Con->{relayok} - which identifies if it is an incoming (1) or outgoing/local (0) mail

So a possible definition of this parameter could be:
'/usr/bin/move_attachment_to_web.sh $relayok FILENAME'
or
'c:/assp/move_attachment_to_web.cmd $relayok FILENAME'

The file has to be removed by the script. If not, assp will warn about this and will remove the file in the /transfer folder.
To keep the filenames unique, the assp message identifier is placed in front of the filename - like: M1-30438-02027_attachmentfilename.
Notice: if the filename contains unicode characters, assp will pass this characters in UTF-8 to your script!
Keep in mind, that if this script terminates it's own process - ASSP will die!

Attachment size incoming ***(ASSP_AFCinsize)***

1024

The size in KB of an attachment in incoming mails that must be reached, to call the **ASSP_AFCWebScript**. This parameter is ignored if left blank or set to zero.

Attachment size outgoing/local ***(ASSP_AFCoutsize)***

1024

The size in KB of an attachment in outgoing or local mails that must be reached, to call the **ASSP_AFCWebScript**. This parameter is ignored if left blank or set to zero.

SMIME sign outgoing mails* ***(ASSP_AFCSMIME)***

file:files/smime_cfg.txt Edit file

An "SMIME feature license" assigned to this host is required to use this feature!
Licenses are granted user based (10,50,100,250,500,1000) for a periode of two years.
An licensed user is an email address, that uses this feature at least one time, within the licensed periode.
For pricing information, please contact **Thomas Eckardt via email** or visit **www.thockar.com** .

Feature description:
This feature requires an installed Perl module **Crypt::SMIME** .

If configured, outgoing mails will be digitaly signed according to the SMIME specifications provided by the installed OpenSSL and Crypt::SMIME version - this is S/MIME Version 3.1 (specification is in RFC 3851) , newer version may support S/MIME Version 3.2 (specification is in RFC 5751).
It is possible to configure privat and/or corporate signatures. In any case, the "file:" option must be used - specify one configuration per line.
The domain or user is separated by "=>" from the signing configuration/policy. It is possible to use group definitions of domains and users using the [**Groups**] option. Define one line per domain or user or group.
Configuration entries are separated by comma.
Configuration entry pairs (tag and value) are separated by "=".
File definitions for the certificate and privat key have to include the full path to the file! Certificate and privat key have to be provided in PEM format
If you exchange any certificate or key file, click "Edit file" and save the file again to force a reload of the internal certificate store.
The domain / user part accepts full email addresses , domains and groups - wildcards are supported and must be used for domain definitions.
The domain / user part is compaired to the envelope sender - the first matching entry (in reverse generic order) will be used. Entries starting with a minus sign, explicit exclude the domain/user/group from SMIME processing.

certfile - is required and specifys the full path to the certificate to use. The subject of the certificate has to include a valid email address. In normal case, this email address is specified by the cert-subject-tag "emailAddress". The "FROM:" address in the mail header will be replaced by this email address and a "Reply-To:" line with the original sender is added (or replaced) to the mail header.
If the subject of the certificate specifys the email address in another tag, define this tag (NOT the email address) after "emailaddress=".

keyfile - is required and specifys the full path to the file that contains the privat key

keypass - the tag is required, the value is optional - defines the password required (or not) for the privat key

emailaddress - is optional - please read "certfile"
rcpt - is optional - include/[-]exclude mails to specified users and/or domains (recipients) - to exclude addresses, write a minus in front - separate multiple entries by space

examples:

- (1) user@your.domain => certfile=/certs/user_cert.pem, keyfile=/certs/user_key.pem, keypass=, rcpt=-otheruser@other.domain
- (2) *your.domain => certfile=/certs/corporate_cert.pem, keyfile=/certs/corporate_key.pem, keypass=mypassword
- (3) *@your.domain => certfile=/certs/corporate_cert.pem, keyfile=/certs/corporate_key.pem, keypass= , emailaddress=Email
- (4) -user4@your.domain
- (5) -*@*.your.domain
- (6) -[no_smime]

The first example specifys a privat signing policy which exclude the recipient otheruser@other.domain, the second and third example specifys a corporate signing policy (with and without subdomains). The fourth example excludes the user "user4@your.domain" from SMIME processing. The fives example excludes all subdomains of "your.domain" from SMIME processing. The last example excludes all domains, subdomains and users defined in the group "[no_smime]" from SMIME processing.

corporate SMIME signing:

Assume we define the following configuration line:
*@your.domain.com => certfile=/certs/corporate_cert.pem, keyfile=/certs/corporate_key.pem, keypass=
Now let's say, the subject of the specified certificate (corporate_cert.pem) contains .../emailAddress=central.office@your.domain.com/...
Your local user "mark.schmitz@your.domain.com" sends a mail to an external recipient. The related mail header is:

From: "Mark Schmitz" <mark.schmitz@your.domain.com>
Disposition-Notification-To: <mark.schmitz@your.domain.com>

After SMIME signing the mail, the related mail headers are the following:

From: "Mark Schmitz" <central.office@your.domain.com>
Disposition-Notification-To: <mark.schmitz@your.domain.com>
Reply-To: <mark.schmitz@your.domain.com>
References: assp-corp-smime-mark.schmitz@your.domain.com

The mail client of the recipient will validate the signature against the "From" address - which corresponds to the email address specified in the subject of the certificate -> VALID
Pressing the "REPLY/ANSWER" button, the mail client will provide "mark.schmitz@your.domain.com" as recipient address (To:) for the answer, using the entry in the "Reply-To:" header.
Notice, that some bad and/or older mail clients are ignoring the "Reply-To:" header tag - in such case an answered mail will go to "central.office@your.domain.com".
ASSP will help you a bit to prevent this. In addition to the required mail header changes, assp will add or enhance the "References:" mail header tag with a value of "assp-corp-smime-EMAILADDRESS" , where EMAILADDRESS is the original sender address.
If assp receives an answered mail, it will look for such an entry in the mail header and will add the found email address to the "To" header, if it is not already found there.

ASSP_DCC-Plugin

Do the ASSP_DCC Plugin *(DoASSP_DCC)*

disabled

This Plugin uses a service provided by www.rhyolite.com to detect spam on a statistical (checksum) base.
You have to open UDP port 6277 on your firewall for outgoing connections and dccifd must be installed an running. This port is used by dccifd to connect to the DCC-Servers.
Please notice that dccifd is not available on windows systems. To use DCC on windows you must install the DCC components on a second linux system and you have to configure **ASSP_DCCdccifd** to use an IP socket to connect to the dccifd. Please follow the installation instructions on **http://www.rhyolite.com/dcc/INSTALL.html**
DCC is a distributed, collaborative, spam detection and filtering network. Through user contribution, DCC establishes a distributed and constantly updating catalogue of spam in propagation that is consulted by email clients to filter out known spam. Detection is done with statistical signatures that efficiently spot mutating spam content. User input is validated through reputation assignments based on consensus on report and revoke assertions which in turn is used for computing confidence values associated with individual signatures.
This plugin is designed for- and running in call/run level 'complete mail'!

the priority of the Plugin *(ASSP_DCCPriority)*

8

Sets the priority of this Plugin within the call/run-level 'complete mail'. The Plugin with the lowest priority value is processed first!

☐ **Set the Plugin in Testmode** *(TestASSP_DCC)*

Set this Plugin in to Testmode. The Plugin returns true in any case!

Enable Plugin logging *(ASSP_DCCLog)*

standard

PenaltyBox valance for ASSP_DCC Plugin + *(ASSP_DCCValencePB)*

15

Message scoring for ASSP_DCC Plugin

☑ **process whitlisted mails** *(procWhiteASSP_DCC)*

Whitelisted mails will be processed by this Plugin!

location to log the failed mails *(ASSP_DCCLogTo)*

spamfolder & ccallspam

Where to store rejected mail for this Plugin. Recommended: spamfolder & ccallspam
1 = spamfolder, 2 = notspam folder, 3 = spamfolder & ccallspam, 4 = mailok folder, 5 = attachment folder, 6 = discard, 7 = discard & ccallspam.

Home Directory of DCC on linux *(ASSP_DCChomedir)*

/var/dcc

The home Directory of DCC on linux systems. dccifd will listen on a unix socket in this folder. This parameter will be ignored if **ASSP_DCCdccifd** is configured!

dccifd IP/Host Information *(ASSP_DCCdccifd)*

If you are running dccifd on a second system, define the IP address or hostname and port of that daemon here. For example: 192.168.0.100:11111 or dccifd.mydomain.com:11111 . If this parameter is configured, the setting of **ASSP_DCChomedir** will be ignored!

dccifd Socket Timeout *(ASSP_DCCTimeout)*

16

Define the maximum time in seconds, assp will wait for an Answer of the dccifd. Recommended setting are between 10 an 16 - default is 16 seconds.

DCC Auth Client IP *(ASSP_DCCClientIP)*

Define the IP address that is used to authenticate assp at the dccifd here.

DCC Auth Client Name *(ASSP_DCCClientName)*

Define the hostname that is used to authenticate assp at the dccifd here.

Report to DCC-Server *(ASSP_DCCReportToDCC)*

query only

Define how the reporting function of DCC should be used. If set to "query only" - no reporting is be done. If set to "report" of the current DCC result will be reported to the DCC servers. If set to "report and known spam" the same behavior like "report" belongs and additionaly - if the mail is still detected as SPAM by assp, this will be reported to the DCC servers.

ASSP_FakeMX-Plugin

Do the ASSP_FakeMX Plugin *(DoASSP_FakeMX)*

monitor

To explain it , let's say you have a domain "example.com" and let's also say that the domain has a single MX

example.com IN MX 10 mail.example.com

now, to adopt the "MX sandwich" (or Fake MX, as we call it) approach you'll need to add a couple MX records so, that the DNS will contain something like

example.com IN MX 10 mx00.example.com
example.com IN MX 20 mail.example.com
example.com IN MX 90 mx99.example.com

Now comes the trick, the "mx00" will point to an IP address on which there isn't (and will NEVER be) a listener on 25/tcp; this means that any connection attempted to mx00.example.com:25 will result into a TCP timeout error. The MX mail record (mail.example.com) will point to the real **listenPort** (and there may be more by the way) and the mx99, that is the last MX will point to another **listenPort** and to ASSP_FakeMXFakeMX

ASSP will answer connections on "mx99" *ALWAYS* with a reply of

421 Service temporarily unavailable, closing transmission channel.

Now the question - how will such a construct (the MX sandwitch) prevent spam?
Real mail servers will try to connect to mx00.example.com first. This will fail and they will next try mail.example.com , because it is the next MX in order, where they can connect and deliver the mail.
Some spam bots may also try to connect to mx00.example.com. This will also fail. But most spam bots never try a second MX - this is what we want - no bot - no spam.
A second type of spam bots are connecting to MX records in revers order. They connect to mx99.example.com first - which is a fault. The IP will get the configured score (**ASSP_FakeMXValencePB**). Future connections (even at the right MX records) from this IP can be blocked by the PenaltyBox or DelayIP.

NoProcessing IP's and senders can use the FakeMX without any blocking.
Whitelisted IP's and senders can use the FakeMX without any blocking as long as **procWhiteASSP_FakeMX** is not set.
ISP IP's can use the FakeMX without any blocking.
IP's listed in **acceptAllMail** can use the FakeMX without any blocking.

NOTICE: If you set this option to "block" and **TestASSP_FakeMX** is switched "OFF" - YOU NEED to switch "OFF" **EnableDelaying** FIRST !

This Plugin is designed for- and running in call/run level 'SMTP-handshake'!

the priority of the Plugin *(ASSP_FakeMXPriority)*

5

Sets the priority of this Plugin within the call/run-level 'SMTP-handshake'. The Plugin with the lowest priority value is processed first!

☐ **set the Plugin in Testmode *(TestASSP_FakeMX)***

Set this Plugin in to Testmode. The Plugin returns true in any case!

Enable Plugin logging *(ASSP_FakeMXLog)*

standard

PenaltyBox valance for ASSP_FakeMX Plugin *(ASSP_FakeMXValencePB)*

200

IP scoring for ASSP_FakeMX Plugin

☑ **process whitlisted mails *(procWhiteASSP_FakeMX)***

Whitelisted IP's will be processed by this Plugin!

location to log the failed mails *(ASSP_FakeMXLogTo)*

no collection

not used

FakeMX listener *(ASSP_FakeMXFakeMX)*

The FakeMX for a MX sandwitch - must be predefined the same way in **listenPort** and here .

ASSP_OCR-Plugin

Do the ASSP_OCR Plugin *(DoASSP_OCR)*

monitor

This Plugin resolves the ASCII part of attached images.
This Plugin is designed for- and running in call/run level 'complete mail'!

the priority of the Plugin *(ASSP_OCRPriority)*

5

Sets the priority of this Plugin within the call/run-level 'complete mail'. The Plugin with the lowest priority value is processed first!

Enable Plugin logging *(ASSP_OCRLog)*

standard

☐ **process whitelisted mails *(procWhiteASSP_OCR)***

Whitelisted mails will be processed by this Plugin!

☑ **extract text from text files *(DoSimpleTextASSP_OCR)***

The text components of attached text/html or similar files will be extracted!

☑ **extract text from pdf files *(DoPDFTextASSP_OCR)***

The text components of attached pdf files will be extracted!

☐ **extract text from images inside pdf files *(DoPDFImageASSP_OCR)***

The text components of images inside of attached pdf files will be extracted!

☐ **extract text from attached image files *(DoImageASSP_OCR)***

The text components of attached images be extracted!

Full Path to ImageMagick Executable *(ASSP_OCRExec)*

C:\Program Files\ImageMagick\convert

The full path to the ImageMagick executable (convert). For example: c:/progams/ImageMagick/convert or /opt/ImageMagick/convert .
If not defined, ASSP will search for this executable and set this value automaticly, if any of the both Image options is set.
The path to ImageMagic must be defined in the systems PATH variable!
If the executable was not found, this value will be set to "convert not found in path". In this case set your systems PATH variable correct, restart ASSP and clear this value - ASSP will then retry to find convert!

maxsize of the converted images *(ASSP_OCRocrmaxsize)*

1024000

The maximum size of the converted images to scan with tesseract - default is 1024000

maximum number of allowed concurrent running image processing tasks *(ASSP_OCRocrmaxprocesses)*

3

The maximum number of concurrent running image processing tasks (tesseract / convert). This number should be less than the number of available CPU cores - default is 3. Changing this value requires an ASSP restart!

ASSP_Razor-Plugin

Do the ASSP_Razor Plugin *(DoASSP_Razor)*

disabled

This Plugin uses a service provided by www.cloudmark.com to detect spam on a statistical base.
You have to open port 2703 on your firewall for outgoing connections. This port is used by Razor to connect to the Razor-Servers.
Razor is a distributed, collaborative, spam detection and filtering network. Through user contribution, Razor establishes a distributed and constantly updating catalogue of spam in propagation that is consulted by email clients to filter out known spam. Detection is done with statistical and randomized signatures that efficiently spot mutating spam content. User input is validated through reputation assignments based on consensus on report and revoke assertions which in turn is used for computing confidence values associated with individual signatures.
This plugin is designed for- and running in call/run level 'complete mail'!

the priority of the Plugin *(ASSP_RazorPriority)*

7

Sets the priority of this Plugin within the call/run-level 'complete mail'. The Plugin with the lowest priority value is processed first!

☐ **Set the Plugin in Testmode *(TestASSP_Razor)***

Set this Plugin in to Testmode. The Plugin returns true in any case!

Enable Plugin logging *(ASSP_RazorLog)*

standard

Maximum Confidence by Razor for NOT SPAM *(ASSP_RazorMaxNotSpamConf)*

default

The Razor-Server will return a confidence/spam level for each mail between 0 and 100, where 0 meens no spam and 100 absolute spam. Under default conditions Razor uses a pre calculated default value, but if you want, you can set this to an absolute value between 0 and 99 or a value relative to the default (use "default-dd" or "default+dd" without quotes - dd must be digits). If the Razor-score is higher than this value, the mail will consider spam. To use the default value (recommended), set the value to the word "default".

PenaltyBox valence for ASSP_Razor Plugin + *(ASSP_RazorValencePB)*

15

Message/IP scoring for ASSP_Razor Plugin

☑ **process whitlisted mails *(procWhiteASSP_Razor)***

Whitelisted mails will be processed by this Plugin!

location to log the failed mails *(ASSP_RazorLogTo)*

spamfolder & ccallspam

Where to store rejected mail for this Plugin. Recommended: spamfolder & ccallspam
1 = spamfolder, 2 = notspam folder, 3 = spamfolder & ccallspam, 4 = mailok folder, 5 = attachment folder, 6 = discard, 7 = discard & ccallspam.

select a language file to change the display language

default edit readme

For defining any full filepaths, always use slashes ("/") not backslashes. For example: c:/assp/certs/server-key.pem !

Fields marked with one small (ˢ) - which are interval definitions - accept a single or a list of crontab entries separated by '|'. Such entries could be used to flexible schedule the configured task. Schedule definitions requires an installed **Schedule::Cron** module in PERL.
Time and Date specification
Entry is the specification of the scheduled time in crontab format, which contains five mandatory time and date fields. Entry can be either a plain string, which contains a whitespace separated time and date specification.
The time and date fields are (taken mostly from "Vixie" cron):

field	values
minute	0-59
hour	0-23
day of month	1-31
month	1-12 (or as names)
day of week	0-7 (0 or 7 is Sunday, or as names)
seconds	0-59 (optional) **not supported inside ASSP !!!**

A field may be an asterisk (*), which always stands for "first-last".
Ranges of numbers are allowed. Ranges are two numbers separated with a hyphen. The specified range is inclusive. For example, 8-11 for an "hours" entry specifies execution at hours 8, 9, 10 and 11.
Lists are allowed. An list is a set of numbers (or ranges) separated by commas. Examples: "1,2,5,9","0-4,8-12". Step values can be used in conjunction with ranges. Following a range with "/number" specifies skips of the numbers value through the range. For example, "0-23/2" can be used in the hours field to specify command execution every other hour (the alternative in the V7 standard is "0,2,4,6,8,10,12,14,16,18,20,22"). Steps are also permitted after an asterisk, so if you want to say "every two hours", just use "*/2".
Names can also be used for the "month" and "day of week" fields. Use the first three letters of the particular day or month (case doesn't matter).
Note: The day of a command's execution can be specified by two fields -- day of month, and day of week. If both fields are restricted (ie, aren't *), the command will be run when either field matches the current time. For example, "30 4 1,15 * 5" would cause a command to be run at 4:30 am on the 1st and 15th of each month, plus every Friday. In addition, ranges or lists of names are allowed.
Examples:

8 0 * * *	==>	8 minutes after midnight, every day
5 11 * * Sat,Sun	==>	at 11:05 on each Saturday and Sunday
0-59/5 * * * *	==>	every five minutes
42 12 3 Feb Sat	==>	at 12:42 on 3rd of February and on each Saturday in February
32 11 * * * 0-30/2	==>	11:32:00, 11:32:02, ... 11:32:30 every day

Fields marked with at least one asterisk (*) accept a list separated by '|' (for example: abc|def|ghi) or a file designated as follows (path relative to the ASSP directory): 'file:files/filename.txt'. Putting in the *file:* will prompt ASSP to put up a button to edit that file. *files* is the subdirectory for files. The file does not need to exist, you can create it by saving it from the editor within the UI. The file must have one entry per line; anything on a line following a number sign or a semicolon (# ;) is ignored (a comment).
It is possible to include custom-designed files at any line of such a file, using the following directive
include filename
where filename is the relative path (from c:/assp) to the included file like files/inc1.txt or inc1.txt (one file per line). The line will be internaly replaced by the contents of the included file!

Fields marked with two asterisk (**) contains regular expressions (regex) and accept a second weight value. Every weighted regex that contains at least one '|' has to begin and end with a '~' - inside such regexes it is not allowed to use a tilde '~', even it is escaped - for example: ~abc\~|def~=>23 or ~abc~|def~=>23 - instead use the octal (\126) or hex (\x7E) notation , for example ~abc\126|def~=>23 or ~abc\x7E|def~=>23 . Every weighted regex has to be followed by '=>' and the weight value. For example: Phishing\.=>1.45|~Heuristics|Email~=>50 or ~(Email|HTML|Sanesecurity)\.(Phishing|Spear|(Spam|Scam)[a-z0-9]?)\.~=>4.6|Spam=>1.1|~Spear|Scam~=>2.1 . The multiplication result of the weight and the penaltybox valence value will be used for scoring, if the absolute value of weight is less or equal 6. Otherwise the value of weight is used for scoring. It is possible to define negative values to reduce the resulting message score.

For all "bomb*" regular expressions and "blackRe", "scriptRe", "invalidFormatHeloRe", "invalidPTRRe" and "invalidMsgIDRe" it is possible to define a third parameter (to overwrite the default options) after the weight like: Phishing\.=>1.45|~Heuristics|Email~=>50:>N[+-]W[+-]L[+-]I[+-]. The characters and the optional to use + and - have the following functions:
use this regex (+ = only)(- = never) for: N = noprocessing , W = whitelisted , L = local , I = ISP mails . So the line ~Heuristics|Email~=>50:>N-W-LI could be read as: take the regex with a weight of 50, never scan noprocessing mails, never scan whitelisted mails, scan local mails and mails from ISP's (and all others). The line ~Heuristics|Email~=>3.2:>N-W+I could be read as: take the regex with a weight of 3.2 as factor, never scan noprocessing mails, scan only whitelisted mails even if they are received from an ISP .
If the third parameter is not set or any of the N,W,L,I is not set, the default configuration for the option will be used unless a default option string is defined anywhere in a single line in the file in the form !!!NWLI!!! (with + or - is possible).
If any parameter that allowes the usage of weighted regular expressions is set to "block", but the sum of the resulting weighted penalty value is less than the corresponding "Penalty Box Valence Value" (because of lower weights) - only scoring will be done!

If the regular expression optimization is used - ("perl module Regexp::Optimizer" installed and enabled) - and you want to disable the optimization for a special regular expression (file based), set one line (eg. the first one) to a value of 'assp-do-not-optimize-regex' or 'a-d-n-o-r' (without the quotes)! To disable the optimization for a specific line/regex, put <<< in front and >>> at the end of the line/regex. To weight such line/regex write for example: <<<Phishing\.>>>=>1.45=>N- or ~<<<Heuristics|Email>>>~=>50 or ~<<<(Email|HTML|Sanesecurity)\.(Phishing|Spear|(Spam|Scam)[a-z0-9]?)\.>>>~=>4.6 .

Using Perl 5.12 or higher, assp supports the usage of unicode block, unicode script and unicode character definitions in regular expressions, llke: \P{Balinese} \p{Script:Greek} \P{Hebrew} \p{script=katakana} \N{greek:Sigma} \x{263a}
It is recommended to switch off the regular expression optimization, if a unicode regular expression definition is used (at least for the line, where it is used)!

The literal 'SESSIONID' will be replaced by the unique message logging ID in every SMTP error reply.
The literal 'IPCONNECTED' will be replaced by the connected IP address in every SMTP error reply.
The literal 'IPORIGIN' will be replaced by the origin IP address in every SMTP error reply.
The literal 'NOTSPAMTAG' will be replaced by a random calculated TAG using **NotSpamTag**, in every SMTP permanent (5xx) error reply.
The literal 'MYNAME' will be replaced by the configuration value defined in 'myName' in every SMTP error reply.

If you define any SMTP-reply-code (like for example **SpamError**) as a **temporary** reply code (starting with **4** like **4**52 instead of the default **5** like **5**50), the connection will be dropped at it's current state, regardless any **collection** or **forwarding** setting. These actions may **finished incomplete** in this case!

If the internal name is shown in light blue like (uniqueIDPrefix) , this indicates that the configured value differs from the default value. To show the default value, move the mouse over the internal name. A click on the internal name will reset the value to the default.

IP ranges are defined as for example 182.82.10. CIDR notation is accepted (182.82.10.0/24). Hyphenated ranges can be used (182.82.10.0-182.82.10.255).
Text after the range (and before a number sign) will be accepted as comment to be shown in a match. For example:
182.82.10.0/24 Yahoo #comment to be removed
The short notation like 182.82.10. is only allowed for IPv4 addresses, IPv6 addresses must be fully defined as for example 2201:1::1 or 2201:1::/96
You may define a hostname instead of an IP, in this case the hostname will be replaced by all DNS-resolved IP-addresses, each with a /32 or /128 netmask. For example:
mta5.am0.yahoodns.net Yahoo #comment to be removed -> 66.94.238.147/32 Yahoo|... Yahoo|... Yahoo

'kill -HUP 4156' will load settings from disk. 'kill -NUM07 4156' will suspend or resume assp. 'kill -USR2 4156' will save settings to disk.

glosar

A

acceptAllMail - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

AddCustomHeader - SPAM Control

AddIntendedForHeader - SPAM Control

AddRBLHeader - DNSBL - RBL Validation

AddRWLHeader - Whitelisting and RWL(DNSWL)

AddSpamHeader - SPAM Control

AddSpamReasonHeader - SPAM Control

AddSubjectHeader - SPAM Control

AddURIS2MyHeader - URIBL and Obfuscation Detection

adminusersdbNoBIN - File Paths and Database

ALARMtimeout - General Server Setup

allowAdminConnectionsFrom - General Server Setup

AllowInternalsInRegex - Perl Regular Expression Filter and Spambomb Detection

allowStatConnectionsFrom - General Server Setup

AsADaemon - General Server Setup

ASSP_AFCDetectSpamAttachRe - ASSP_AFC-Plugin

ASSP_AFCMaxZIPLevel - ASSP_AFC-Plugin

ASSP_AFCoutsize - ASSP_AFC-Plugin

ASSP_AFCReplBadAttach - ASSP_AFC-Plugin

ASSP_AFCReplViriParts - ASSP_AFC-Plugin

ASSP_AFCSelect - ASSP_AFC-Plugin

ASSP_AFCWebScript - ASSP_AFC-Plugin

ASSP_DCCClientName - ASSP_DCC-Plugin

ASSP_DCChomedir - ASSP_DCC-Plugin

ASSP_DCCLogTo - ASSP_DCC-Plugin

ASSP_DCCReportToDCC - ASSP_DCC-Plugin

ASSP_DCCValencePB - ASSP_DCC-Plugin

ASSP_FakeMXLog - ASSP_FakeMX-Plugin

ASSP_FakeMXPriority - ASSP_FakeMX-Plugin

ASSP_OCRExec - ASSP_OCR-Plugin

ASSP_OCRocrmaxprocesses - ASSP_OCR-Plugin

ASSP_OCRPriority - ASSP_OCR-Plugin

ASSP_RazorLogTo - ASSP_Razor-Plugin

ASSP_RazorPriority - ASSP_Razor-Plugin

asspCfg - General Server Setup

asspCpuAffinity - General Server Setup

atSpamLovers - SPAM Lover and SPAM Hater

AttachmentLog - Logging and Notifications

AUTHrequireTLS - Network Setup for Incoming Mail and Authentication

autoAddResendToWhite - Block Reporting - Schedule and Instant

AutoReloadCfg - General Server Setup

AutoRestartAfterCodeChange - General Server Setup

autoRestartDiedThreads - General Server Setup

autValencePB - PenaltyBox - Message and IP Scoring

AvError - Virus Protection using ClamAV and OS-FileScanner

AddConfidenceHeader - Hidden Markov Model and Bayesian Options

AddDKIMHeader - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

AddLevelHeader - SPAM Control

AddRegexHeader - Logging and Notifications

AddScoringHeader - PenaltyBox - Message and IP Scoring

AddSpamProbHeader - Hidden Markov Model and Bayesian Options

AddSPFHeader - Validate SPF, DMARC and SRS

AddURIBLHeader - URIBL and Obfuscation Detection

adminusersdb - File Paths and Database

adminusersdbpass - File Paths and Database

allLogRe - Logging and Notifications

AllowedDupSubjectRe - Collecting SPAM and HAM

allowRelayCon - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

allTestMode - TestModes and SPAM Tagging

ASSP_AFCblockEncryptedZIP - ASSP_AFC-Plugin

ASSP_AFCinsize - ASSP_AFC-Plugin

ASSP_AFCMSGSIZEscore - ASSP_AFC-Plugin

ASSP_AFCPriority - ASSP_AFC-Plugin

ASSP_AFCReplBadAttachText - ASSP_AFC-Plugin

ASSP_AFCReplViriPartsText - ASSP_AFC-Plugin

ASSP_AFCSMIME - ASSP_AFC-Plugin

ASSP_DCCClientIP - ASSP_DCC-Plugin

ASSP_DCCdccifd - ASSP_DCC-Plugin

ASSP_DCCLog - ASSP_DCC-Plugin

ASSP_DCCPriority - ASSP_DCC-Plugin

ASSP_DCCTimeout - ASSP_DCC-Plugin

ASSP_FakeMXFakeMX - ASSP_FakeMX-Plugin

ASSP_FakeMXLogTo - ASSP_FakeMX-Plugin

ASSP_FakeMXValencePB - ASSP_FakeMX-Plugin

ASSP_OCRLog - ASSP_OCR-Plugin

ASSP_OCRocrmaxsize - ASSP_OCR-Plugin

ASSP_RazorLog - ASSP_Razor-Plugin

ASSP_RazorMaxNotSpamConf - ASSP_Razor-Plugin

ASSP_RazorValencePB - ASSP_Razor-Plugin

asspCfgVersion - General Server Setup

asspLog - Logging and Notifications

AttachmentError - Attachment Validation and Protection

attachTestMode - TestModes and SPAM Tagging

AUTHUserIPfrequency - SMTP Session Limits

autoCorrectCorpus - Rebuild Hidden Markov Model and Bayesian Database

AutoRestart - General Server Setup

AutoRestartCmd - General Server Setup

AutoUpdateASSP - General Server Setup

AvClamdPort - Virus Protection using ClamAV and OS-FileScanner

B

Back250OKISP - Outgoing Message Tagging, NDR Validation and Backscatter Detection

BackDNSInterval - Outgoing Message Tagging, NDR Validation and Backscatter Detection

BackLog - Collecting SPAM and HAM

BacksctrLog - Logging and Notifications

BackNP - Outgoing Message Tagging, NDR Validation and Backscatter Detection

BackSctrServiceProvider - Outgoing Message Tagging, NDR Validation and Backscatter Detection

backsctrValencePB - PenaltyBox - Message and IP Scoring

backupDBDir - File Paths and Database

backupDBInterval - File Paths and Database

BackWL - Outgoing Message Tagging, NDR Validation and Backscatter Detection

BadAttachL1 - Attachment Validation and Protection

BadAttachL2 - Attachment Validation and Protection

BadAttachL3 - Attachment Validation and Protection

banFailedSSLIP - SSL Proxy and TLS support

base - File Paths and Database

BATVLog - Logging and Notifications

BATVSec - Outgoing Message Tagging, NDR Validation and Backscatter Detection

batvValencePB - PenaltyBox - Message and IP Scoring

baValencePB - PenaltyBox - Message and IP Scoring

BayesAfterHMM - Hidden Markov Model and Bayesian Options

Bayesian_localOnly - Hidden Markov Model and Bayesian Options

BayesianLog - Logging and Notifications

BayesLocal - Hidden Markov Model and Bayesian Options

BayesMaxProcessTime - Hidden Markov Model and Bayesian Options

BayesNP - Hidden Markov Model and Bayesian Options

BayesWL - Hidden Markov Model and Bayesian Options

baysConf - Hidden Markov Model and Bayesian Options

baysConfidenceHalfScore - Hidden Markov Model and Bayesian Options

bayshamValencePB - PenaltyBox - Message and IP Scoring

bayslocalValencePB - PenaltyBox - Message and IP Scoring

baysNonSpamLog - Collecting SPAM and HAM

baysProbability - Hidden Markov Model and Bayesian Options

baysSpamHaters - SPAM Lover and SPAM Hater

baysSpamLog - Collecting SPAM and HAM

baysSpamLovers - SPAM Lover and SPAM Hater

baysSpamLoversRe - SPAM Lover and SPAM Hater

baysSpamLoversRed - SPAM Lover and SPAM Hater

baysTestMode - TestModes and SPAM Tagging

baysTestModeUserAddresses - TestModes and SPAM Tagging

baysValencePB - PenaltyBox - Message and IP Scoring

bccValencePB - PenaltyBox - Message and IP Scoring

BerkeleyDB_DBEngine -

blackListedDomains - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

blackRe - Perl Regular Expression Filter and Spambomb Detection

blackReMaxHits - Perl Regular Expression Filter and Spambomb Detection

blackSenderBase - SenderBase and WhoisIP

blackValencePB - PenaltyBox - Message and IP Scoring

blDomainLog - Collecting SPAM and HAM

BlockExes - Attachment Validation and Protection

BlockMaxSearchTime - Block Reporting - Schedule and Instant

BlockNPExes - Attachment Validation and Protection

BlockRepForwHost - Block Reporting - Schedule and Instant

BlockReportAdmins - Block Reporting - Schedule and Instant

BlockReportFile - Block Reporting - Schedule and Instant

BlockReportFilter - Block Reporting - Schedule and Instant

BlockReportFormat - Block Reporting - Schedule and Instant

BlockReportHTTPName - Block Reporting - Schedule and Instant

BlockReportNow - Block Reporting - Schedule and Instant

BlockReportSchedule - Block Reporting - Schedule and Instant

BlockResendLink - Block Reporting - Schedule and Instant

BlockResendLinkLeft - Block Reporting - Schedule and Instant

BlockResendLinkRight - Block Reporting - Schedule and Instant

blockstrictSPFRe - Validate SPF, DMARC and SRS

BlockUuencoded - Attachment Validation and Protection

BlockWLExes - Attachment Validation and Protection

blSpamLovers - SPAM Lover and SPAM Hater

blTestMode - TestModes and SPAM Tagging

blValencePB - PenaltyBox - Message and IP Scoring

bombCharSets - Perl Regular Expression Filter and Spambomb Detection

bombDataRe - Perl Regular Expression Filter and Spambomb Detection

bombDataReMaxHits - Perl Regular Expression Filter and Spambomb Detection

bombError - Perl Regular Expression Filter and Spambomb Detection

bombErrorReason - Perl Regular Expression Filter and Spambomb Detection

bombHeaderRe - Perl Regular Expression Filter and Spambomb Detection

bombHeaderReMaxHits - Perl Regular Expression Filter and Spambomb Detection

BombLog - Logging and Notifications

bombMaxPenaltyVal - Perl Regular Expression Filter and Spambomb Detection

bombRe - Perl Regular Expression Filter and Spambomb Detection

bombReISPIP - Perl Regular Expression Filter and Spambomb Detection

bombReLocal - Perl Regular Expression Filter and Spambomb Detection

bombReMaxHits - Perl Regular Expression Filter and Spambomb Detection

bombReNP - Perl Regular Expression Filter and Spambomb Detection

bombReWL - Perl Regular Expression Filter and Spambomb Detection

bombSenderRe - Perl Regular Expression Filter and Spambomb Detection

bombSkipHeaderTagRe - Perl Regular Expression Filter and Spambomb Detection

bombSpamLovers - SPAM Lover and SPAM Hater

bombSubjectRe - Perl Regular Expression Filter and Spambomb Detection

bombSuspiciousRe - Perl Regular Expression Filter and Spambomb Detection

bombSuspiciousValencePB - PenaltyBox - Message and IP Scoring

bombTestMode - TestModes and SPAM Tagging

bombValencePB - PenaltyBox - Message and IP Scoring

BounceSenders - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

C

CatchAll - Local Recipients and Domains & Transparent Recipients and Domains

CatchAllAll - Local Recipients and Domains & Transparent Recipients and Domains

CatchallallISP2NULL - Local Recipients and Domains & Transparent Recipients and Domains

ccHamFilter - Copy Spam & Ham

ccMailReplaceRecpt - Copy Spam & Ham

ccMaxBytes - Copy Spam & Ham

ccMaxScore - Copy Spam & Ham

ccnHamFilter - Copy Spam & Ham

ccSpamAlways - Copy Spam & Ham

ccSpamFilter - Copy Spam & Ham

ccSpamInDomain - Copy Spam & Ham

ccSpamNeverRe - Copy Spam & Ham

ChangeRoot - General Server Setup

checkFilePermOnStart - General Server Setup

ClamAVBytes - Virus Protection using ClamAV and OS-FileScanner

ClamAVLogScan - Virus Protection using ClamAV and OS-FileScanner

ClamAVtimeout - Virus Protection using ClamAV and OS-FileScanner

CleanCacheEvery - General Server Setup

CleanDelayDBInterval - Delaying - Greylisting

CleanPBInterval - PenaltyBox - Message and IP Scoring

ConfigChangeSchedule - General Server Setup

ConnectionLog - Logging and Notifications

ConnectionTransferTimeOut - General Server Setup

ConsoleCharset - General Server Setup

contentOnlyRe - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

ConTimeOutDebug - Logging and Notifications

convertNP - CharacterSet Conversions and TNEF Processing

ConvLog - Logging and Notifications

copyDBToOrgLoc - File Paths and Database

correctednotspam - File Paths and Database

correctedspam - File Paths and Database

CountryCodeBlockedRe - SenderBase and WhoisIP

CountryCodeRe - SenderBase and WhoisIP

D

DataBaseDebug - Logging and Notifications

DBCacheMaxAge - File Paths and Database

DBdriver - File Paths and Database

debug - Logging and Notifications

debugCode - Logging and Notifications

debugIP - Logging and Notifications

debugNoWriteBody - Logging and Notifications

debugRe - Logging and Notifications

DebugSPF - Validate SPF, DMARC and SRS

defaultLocalHost - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

DelayAddHeader - Delaying - Greylisting

delaydb - File Paths and Database

DelayEmbargoTime - Delaying - Greylisting

DelayError - Delaying - Greylisting

DelayExpireOnSpam - Delaying - Greylisting

DelayExpiryTime - Delaying - Greylisting

DelayIP - IP Blocking

DelayIPTime - IP Blocking

DelayLog - Logging and Notifications

DelayMD5 - Delaying - Greylisting

DelayNormalizeVERPs - Delaying - Greylisting

DelayNP - Delaying - Greylisting

DelayShowDB - Delaying - Greylisting

DelayShowDBwhite - Delaying - Greylisting

DelaySL - Delaying - Greylisting

delaySpamLovers - SPAM Lover and SPAM Hater

DelayUseNetblocks - Delaying - Greylisting

DelayWaitTime - Delaying - Greylisting

DelayWithMyName - Delaying - Greylisting

DelayWL - Delaying - Greylisting

DelResendSpam - Block Reporting - Schedule and Instant

denySMTPConnectionsFrom - IP Blocking

denySMTPConnectionsFromAlways - IP Blocking

denySMTPLog - Logging and Notifications

denySMTPstrictEarly - IP Blocking

detectMailLoop - SMTP Session Limits

DisableExtAUTH - Network Setup for Incoming Mail and Authentication

DisableSMTPNetworking - Network Setup for Incoming Mail and Authentication

DisableVRFY - Local Recipients and Domains & Transparent Recipients and Domains

discarded - File Paths and Database

DKIMCacheInterval - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DKIMgenConfig - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

DKIMLog - Collecting SPAM and HAM

DKIMlogging - Logging and Notifications

dkimOkValencePB - PenaltyBox - Message and IP Scoring

dkimTestMode - TestModes and SPAM Tagging

dkimValencePB - PenaltyBox - Message and IP Scoring

DMARCReportFrom - Validate SPF, DMARC and SRS

dnsLocalIPAddress - DNS-Client Setup

DNSResponseLog - DNS-Client Setup

DNSretrans - DNS-Client Setup

DNSretry - DNS-Client Setup

DNSReuseSocket - DNS-Client Setup

DNSServerLimit - DNS-Client Setup

DNSServers - DNS-Client Setup

DNStimeout - DNS-Client Setup

DoAdditionalAnalyze - Email Interface for Reports and List Control

DoASSP_AFC - ASSP_AFC-Plugin

DoASSP_DCC - ASSP_DCC-Plugin

DoASSP_FakeMX - ASSP_FakeMX-Plugin

DoASSP_OCR - ASSP_OCR-Plugin

DoASSP_Razor - ASSP_Razor-Plugin

DoBackSctr - Outgoing Message Tagging, NDR Validation and Backscatter Detection

DoBATV - Outgoing Message Tagging, NDR Validation and Backscatter Detection

DoBayesian - Hidden Markov Model and Bayesian Options

DoBlackDomain - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoBlackDomainNP - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoBlackDomainWL - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoBlackRe - Perl Regular Expression Filter and Spambomb Detection

DoBlockExes - Attachment Validation and Protection

DoBombHeaderRe - Perl Regular Expression Filter and Spambomb Detection

DoBombRe - Perl Regular Expression Filter and Spambomb Detection

DoCountryBlocking - SenderBase and WhoisIP

DoDamping - PenaltyBox - Message and IP Scoring

DoDenySMTP - IP Blocking

DoDenySMTPstrict - IP Blocking

DoDKIM - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

doDKIMConv - CharacterSet Conversions and TNEF Processing

DoDMARC - Validate SPF, DMARC and SRS

DoDomainCheck - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoDomainIP - IP Blocking

DoDropList - IP Blocking

DoExtremeExport - PenaltyBox - Message and IP Scoring

DoExtremeExportAppend - PenaltyBox - Message and IP Scoring

DoFakedLocalHelo - Validate HELO and EHLO

DoFakedNP - Validate HELO and EHLO

DoFakedUseLocalDomain - Validate HELO and EHLO

DoFakedWL - Validate HELO and EHLO

DoFileScan - Virus Protection using ClamAV and OS-FileScanner

DoFrequencyIP - IP Blocking

DoGlobalBlack - Global PenaltyBox Network

DoGlobalWhite - Global PenaltyBox Network

DoHeaderAddrCheck - Local Recipients and Domains & Transparent Recipients and Domains

DoHeloNP - Validate HELO and EHLO

DoHeloWL - Validate HELO and EHLO

DoHMM - Hidden Markov Model and Bayesian Options

DoImageASSP_OCR - ASSP_OCR-Plugin

doInFixTNEF - CharacterSet Conversions and TNEF Processing

DoInvalidFormatHelo - Validate HELO and EHLO

DoInvalidPTR - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoIPinHelo - Validate HELO and EHLO

DoLDAP - Local Recipients and Domains & Transparent Recipients and Domains

DoLDAPSSL - LDAP Setup

DoLocalPenaltyMessage - PenaltyBox - Message and IP Scoring

DoLocalSenderAddress - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

DoLocalSenderDomain - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

DoMaxDupRcpt - Local Recipients and Domains & Transparent Recipients and Domains

doMove2Num - Rebuild Hidden Markov Model and Bayesian Database

DoMsgID - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoMSGIDsig - Outgoing Message Tagging, NDR Validation and Backscatter Detection

DoNoFrom - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoNoFromNP - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoNoFromWL - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoNoSpoofing - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoNoSpoofing4From - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoNotBlockCollect - Collecting SPAM and HAM

DoNotCollectBounces - Collecting SPAM and HAM

DoNotCollectRedList - Collecting SPAM and HAM

DoNotCollectRedRe - Collecting SPAM and HAM

DoNotPenalizeNull - PenaltyBox - Message and IP Scoring

DoNotPenalizeRed - PenaltyBox - Message and IP Scoring

DoNoValidLocalSender - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoOrgBlocking - SenderBase and WhoisIP

DoOrgWhiting - SenderBase and WhoisIP

doOutFixTNEF - CharacterSet Conversions and TNEF Processing

DoPDFImageASSP_OCR - ASSP_OCR-Plugin

DoPDFTextASSP_OCR - ASSP_OCR-Plugin

DoPenalty - PenaltyBox - Message and IP Scoring

DoPenaltyExtreme - PenaltyBox - Message and IP Scoring

DoPenaltyExtremeSMTP - PenaltyBox - Message and IP Scoring

DoPenaltyMakeTraps - PenaltyBox - Message and IP Scoring

DoPenaltyMessage - PenaltyBox - Message and IP Scoring

DoPrivatSpamdb - Hidden Markov Model and Bayesian Options

DoReversed - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoReversedNP - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoReversedWL - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoRFC822 - Local Recipients and Domains & Transparent Recipients and Domains

DoSameSubject - SMTP Session Limits

DoScriptRe - Perl Regular Expression Filter and Spambomb Detection

DoSenderBase - SenderBase and WhoisIP

DoSimpleTextASSP_OCR - ASSP_OCR-Plugin

DoSPFinHeader - Validate SPF, DMARC and SRS

DoStrictDKIM - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

DoT10Stat - Block Reporting - Schedule and Instant

DoTestRe - Perl Regular Expression Filter and Spambomb Detection

DoTLS - SSL Proxy and TLS support

DoTransliterate - Perl Regular Expression Filter and Spambomb Detection

DoValidFormatHelo - Validate HELO and EHLO

DoVRFY - Local Recipients and Domains & Transparent Recipients and Domains

downloadBackDNSFile - Outgoing Message Tagging, NDR Validation and Backscatter Detection

droplist - IP Blocking

E

EmailAdminDomains - Block Reporting - Schedule and Instant

EmailAdmins - Email Interface for Reports and List Control

F

G

H

I

K

keepInTNEF - CharacterSet Conversions and TNEF Processing

keepOutTNEF - CharacterSet Conversions and TNEF Processing

L

LDAPcrossCheckInterval - LDAP Setup

LDAPFail - LDAP Setup

LDAPFilter - LDAP Setup

LDAPHost - LDAP Setup

ldaplistdb - File Paths and Database

ldapLocalIPAddress - LDAP Setup

LDAPLog - Logging and Notifications

LDAPLogin - LDAP Setup

LDAPPassword - LDAP Setup

LDAPRoot - LDAP Setup

LDAPShowDB - LDAP Setup

LDAPtimeout - LDAP Setup

LDAPVersion - LDAP Setup

ldLDAP - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

ldLDAPFilter - LDAP Setup

ldLDAPRoot - LDAP Setup

listenPort - Network Setup for Incoming Mail and Authentication

listenPort2 - Network Setup for Incoming Mail and Authentication

listenPortSSL - Network Setup for Incoming Mail and Authentication

LocalAddresses_Flat - Local Recipients and Domains & Transparent Recipients and Domains

LocalAddresses_Flat_Domains - Local Recipients and Domains & Transparent Recipients and Domains

LocalAddressesNP - Local Recipients and Domains & Transparent Recipients and Domains

localBackDNSFile - Outgoing Message Tagging, NDR Validation and Backscatter Detection

localDomains - Local Recipients and Domains & Transparent Recipients and Domains

LocalFrequencyInt - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

LocalFrequencyNumRcpt - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

LocalFrequencyOnly - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail

LocalPenaltyMessageLimit - PenaltyBox - Message and IP Scoring

LocalPenaltyMessageLow - PenaltyBox - Message and IP Scoring

LocalPolicySPF - Validate SPF, DMARC and SRS

LogDateFormat - Logging and Notifications

LogDateLang - Logging and Notifications

logfile - File Paths and Database

LogNameDate - Logging and Notifications

LogRollDays - Logging and Notifications

M

maillogExt - File Paths and Database

MaillogTailBytes - General Server Setup

MaillogTailJump - General Server Setup

MaintBayesCollection - Collecting SPAM and HAM

MaintenanceLog - Logging and Notifications

MaintThreadCycleTime - General Server Setup

MaxAllowedDups - Collecting SPAM and HAM

MaxAUTHErrors - SMTP Session Limits

MaxBayesFileAge - Collecting SPAM and HAM

maxBayesValues - Hidden Markov Model and Bayesian Options

maxBombSearchTime - Perl Regular Expression Filter and Spambomb Detection

MaxBytes - Collecting SPAM and HAM

MaxCorrectedDays - Collecting SPAM and HAM

maxDampingTime - PenaltyBox - Message and IP Scoring

maxDNSRespDist - DNS-Client Setup

MaxDupRcpt - Local Recipients and Domains & Transparent Recipients and Domains

MaxEqualXHeader - SMTP Session Limits

MaxErrors - SMTP Session Limits

MaxFileAgeSchedule - Collecting SPAM and HAM

MaxFileNameLength - Collecting SPAM and HAM

MaxFiles - Collecting SPAM and HAM

MaxFinConWaitTime - General Server Setup

MaxKeepDeleted - Rebuild Hidden Markov Model and Bayesian Database

MaxLDAPlistDays - LDAP Setup

MaxLogAge - File Paths and Database

MaxLogAgeSchedule - File Paths and Database

MaxNoBayesFileAge - Collecting SPAM and HAM

maxRealSize - SMTP Session Limits

MaxRealSizeAdr - SMTP Session Limits

maxRealSizeError - SMTP Session Limits

maxRealSizeExternal - SMTP Session Limits

MaxRealSizeExternalAdr - SMTP Session Limits

maxSize - SMTP Session Limits

MaxSizeAdr - SMTP Session Limits

maxSizeError - SMTP Session Limits

maxSizeExternal - SMTP Session Limits

MaxSizeExternalAdr - SMTP Session Limits

maxSMTPdomainIP - IP Blocking

maxSMTPdomainIPExpiration - IP Blocking

maxSMTPdomainIPWL - IP Blocking

maxSMTPipConnects - IP Blocking

maxSMTPipDuration - IP Blocking

maxSMTPipExpiration - IP Blocking

maxSMTPipSessions - SMTP Session Limits

maxSMTPSessions - SMTP Session Limits

maxSubjectLength - Perl Regular Expression Filter and Spambomb Detection

MaxVRFYErrors - Local Recipients and Domains & Transparent Recipients and Domains

MaxWhitelistDays - Whitelisting and RWL(DNSWL)

mdrValencePB - PenaltyBox - Message and IP Scoring

MemoryUsageCheckSchedule - General Server Setup

MemoryUsageLimit - General Server Setup

MessageLog - Logging and Notifications

meValencePB - PenaltyBox - Message and IP Scoring

N

O

P

Q

R

RBLmaxweight - DNSBL - RBL Validation
RBLServiceProvider - DNSBL - RBL Validation
rblSpamHaters - SPAM Lover and SPAM Hater
rblTestMode - TestModes and SPAM Tagging
RBLWL - DNSBL - RBL Validation
RebuildNotify - Rebuild Hidden Markov Model and Bayesian Database
RebuildTestMode - Rebuild Hidden Markov Model and Bayesian Database
redlistdb - File Paths and Database
RegExLength - Logging and Notifications
RejectTheseLocalAddresses - Local Recipients and Domains & Transparent Recipients and Domains
relayAuthUser - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail
RelayOnlyLocalDomains - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail
relayPort - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail
RememberGUIPos - General Server Setup
removeBATVTag - Outgoing Message Tagging, NDR Validation and Backscatter Detection
removeForeignBCC - Local Recipients and Domains & Transparent Recipients and Domains
ReplaceRecpt - Local Recipients and Domains & Transparent Recipients and Domains
ReportLog - Logging and Notifications
ReservedOutboundWorkers - General Server Setup
RestartEvery - General Server Setup
rlValencePB - PenaltyBox - Message and IP Scoring
runAsGroupSupplementary - General Server Setup
RunRebuildNow - Rebuild Hidden Markov Model and Bayesian Database
RWLLog - Logging and Notifications
RWLmaxtime - Whitelisting and RWL(DNSWL)
RWLServiceProvider - Whitelisting and RWL(DNSWL)
rblnValencePB - PenaltyBox - Message and IP Scoring
RBLsocktime - DNSBL - RBL Validation
rblSpamLovers - SPAM Lover and SPAM Hater
rblValencePB - PenaltyBox - Message and IP Scoring
RebuildFileTimeLimit - Rebuild Hidden Markov Model and Bayesian Database
RebuildSchedule - Rebuild Hidden Markov Model and Bayesian Database
RebuildThreadCycleTime - General Server Setup
redRe - SPAM Control
regexLogging - Logging and Notifications
relayAuthPass - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail
relayHost - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail
RelayOnlyLocalSender - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail
ReloadOptionFiles - General Server Setup
remindBATVTag - Outgoing Message Tagging, NDR Validation and Backscatter Detection
removeDispositionNotification - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM
ReplaceOldSpamdb - Rebuild Hidden Markov Model and Bayesian Database
replyLogging - Logging and Notifications
resendmail - File Paths and Database
ResetMaxAUTHErrorIPs - SMTP Session Limits
ReStartSchedule - General Server Setup
runAsGroup - General Server Setup
runAsUser - General Server Setup
RWLCacheInterval - Whitelisting and RWL(DNSWL)
RWLmaxreplies - Whitelisting and RWL(DNSWL)
RWLminhits - Whitelisting and RWL(DNSWL)
RWLwhitelisting - Whitelisting and RWL(DNSWL)

S

saValencePB - PenaltyBox - Message and IP Scoring
SBCacheExp - SenderBase and WhoisIP
sbhccValencePB - PenaltyBox - Message and IP Scoring
sborgValencePB - PenaltyBox - Message and IP Scoring
sbSpamLovers - SPAM Lover and SPAM Hater
ScanCC - Virus Protection using ClamAV and OS-FileScanner
ScanLog - Logging and Notifications
ScanWL - Virus Protection using ClamAV and OS-FileScanner
ScoreForeignCountries - SenderBase and WhoisIP
scriptLog - Collecting SPAM and HAM
scriptReMaxHits - Perl Regular Expression Filter and Spambomb Detection
scriptValencePB - PenaltyBox - Message and IP Scoring
send250OKISP - Network Setup, Limits and DKIM signing for Relaying, Outgoing and Local Mail
sendAllAbuseNP - Local Recipients and Domains & Transparent Recipients and Domains
sendAllDestination - Copy Spam & Ham
sendAllPostmaster - Local Recipients and Domains & Transparent Recipients and Domains
sendAllSpam - Copy Spam & Ham
SenderBaseLog - Logging and Notifications
sendHamInbound - Copy Spam & Ham
sendNoopInfo - Logging and Notifications
SaveStatsEvery - General Server Setup
sbfccValencePB - PenaltyBox - Message and IP Scoring
sbnValencePB - PenaltyBox - Message and IP Scoring
sbsccValencePB - PenaltyBox - Message and IP Scoring
sbTestMode - SenderBase and WhoisIP
ScanLocal - Virus Protection using ClamAV and OS-FileScanner
ScanNP - Virus Protection using ClamAV and OS-FileScanner
ScheduleLog - Logging and Notifications
scriptError - Perl Regular Expression Filter and Spambomb Detection
scriptRe - Perl Regular Expression Filter and Spambomb Detection
scriptTestMode - TestModes and SPAM Tagging
send250OK - General Server Setup
sendAllAbuse - Local Recipients and Domains & Transparent Recipients and Domains
sendAllCollect - Collecting SPAM and HAM
sendAllHamDestination - Copy Spam & Ham
sendAllPostmasterNP - Local Recipients and Domains & Transparent Recipients and Domains
sendEHLO - SSL Proxy and TLS support
SenderInvalidError - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM
sendHamOutbound - Copy Spam & Ham

SepChar - Local Recipients and Domains & Transparent Recipients and Domains

SessionLog - Logging and Notifications

setFilePermOnStart - General Server Setup

Showmaxreplies - Logging and Notifications

ShowPerformanceData - General Server Setup

SignalLog - Logging and Notifications

signedSenders - Validate Sender - Addresses, Domains, MsgID, PTR, MX and DKIM

silent - Logging and Notifications

slmatchLogging - Logging and Notifications

slValencePB - PenaltyBox - Message and IP Scoring

smtpAuthServer - Network Setup for Incoming Mail and Authentication

smtpDestination - Network Setup for Incoming Mail and Authentication

smtpDestinationRT - Network Setup for Incoming Mail and Authentication

smtpDestinationSSL - Network Setup for Incoming Mail and Authentication

smtpIdleTimeout - SMTP Session Limits

smtpLocalIPAddress - Network Setup for Incoming Mail and Authentication

smtpNOOPIdleTimeout - SMTP Session Limits

smtpNOOPIdleTimeoutCount - SMTP Session Limits

smtpSSLRequireClientCert - SSL Proxy and TLS support

SNMP - SNMP Configuration

SNMPAgentXSocket - SNMP Configuration

SNMPBaseOID - SNMP Configuration

SNMPLog - Logging and Notifications

SNMPreturnBOOL - SNMP Configuration

SNMPUser - SNMP Configuration

SNMPwriteable - SNMP Configuration

spamaddresses - Collecting SPAM and HAM

spamBombLog - Collecting SPAM and HAM

spamBucketLog - Collecting SPAM and HAM

spamdb - File Paths and Database

SpamError - SPAM Control

spamHaters - SPAM Lover and SPAM Hater

spamHeloLog - Collecting SPAM and HAM

spamISLog - Collecting SPAM and HAM

SpamLog - Collecting SPAM and HAM

spamLovers - SPAM Lover and SPAM Hater

SpamLoversRe - SPAM Lover and SPAM Hater

spamMSLog - Collecting SPAM and HAM

spamMXALog - Collecting SPAM and HAM

spamPBLog - Collecting SPAM and HAM

spamPTRLog - Collecting SPAM and HAM

spamSBLog - Collecting SPAM and HAM

spamSubject - TestModes and SPAM Tagging

spamSubjectCC - Copy Spam & Ham

spamSubjectSL - SPAM Lover and SPAM Hater

spamTag - TestModes and SPAM Tagging

spamTagCC - Copy Spam & Ham

spamTagSL - SPAM Lover and SPAM Hater

spamtrapaddresses - PenaltyBox - Message and IP Scoring

SpamVirusLog - Collecting SPAM and HAM

SPF2 - Validate SPF, DMARC and SRS

SPFCacheInterval - Validate SPF, DMARC and SRS

SPFError - Validate SPF, DMARC and SRS

spfeValencePB - PenaltyBox - Message and IP Scoring

SPFFailLog - Collecting SPAM and HAM

SPFfallback - Validate SPF, DMARC and SRS

SPFLocal - Validate SPF, DMARC and SRS

SPFlocalRecord - Validate SPF, DMARC and SRS

SPFLog - Logging and Notifications

SPFneutral - Validate SPF, DMARC and SRS

SPFnone - Validate SPF, DMARC and SRS

spfnonValencePB - PenaltyBox - Message and IP Scoring

SPFNP - Validate SPF, DMARC and SRS

spfnValencePB - PenaltyBox - Message and IP Scoring

SPFoverride - Validate SPF, DMARC and SRS

spfpValencePB - PenaltyBox - Message and IP Scoring

SPFqueryerror - Validate SPF, DMARC and SRS

SPFsoftfail - Validate SPF, DMARC and SRS

spfSpamLovers - SPAM Lover and SPAM Hater

spfsValencePB - PenaltyBox - Message and IP Scoring

spfTestMode - TestModes and SPAM Tagging

SPFunknown - Validate SPF, DMARC and SRS

spfuValencePB - PenaltyBox - Message and IP Scoring

spfValencePB - PenaltyBox - Message and IP Scoring

SPFWL - Validate SPF, DMARC and SRS

SRSAliasDomain - Validate SPF, DMARC and SRS

SRSFailLog - Collecting SPAM and HAM

SRSHashLength - Validate SPF, DMARC and SRS

SRSno - Validate SPF, DMARC and SRS

SRSSecretKey - Validate SPF, DMARC and SRS

srsSpamLovers - SPAM Lover and SPAM Hater

srsTestMode - TestModes and SPAM Tagging

SRSTimestampMaxAge - Validate SPF, DMARC and SRS

srsValencePB - PenaltyBox - Message and IP Scoring

SRSValidateBounce - Validate SPF, DMARC and SRS

SSL_cipher_list - SSL Proxy and TLS support

SSL_version - SSL Proxy and TLS support

SSLCaFile - SSL Proxy and TLS support

SSLCertFile - SSL Proxy and TLS support

SSLDEBUG - SSL Proxy and TLS support

SSLKeyFile - SSL Proxy and TLS support

SSLPKPassword - SSL Proxy and TLS support

SSLRetryOnError - SSL Proxy and TLS support

SSLSMTPCertVerifyCB - SSL Proxy and TLS support

SSLSMTPConfigure - SSL Proxy and TLS support

SSLSTATCertVerifyCB - SSL Proxy and TLS support

SSLSTATConfigure - SSL Proxy and TLS support

SSLtimeout - SSL Proxy and TLS support

SSLWEBCertVerifyCB - SSL Proxy and TLS support

SSLWEBConfigure - SSL Proxy and TLS support

statSSLRequireClientCert - SSL Proxy and TLS support

StoreASSPHeader - SPAM Control

StoreCompleteMail - Collecting SPAM and HAM

strictSPFRe - Validate SPF, DMARC and SRS

stValencePB - PenaltyBox - Message and IP Scoring

subjectEnd - Logging and Notifications

subjectFrequencyInt - SMTP Session Limits

subjectFrequencyNumSubj - SMTP Session Limits

subjectFrequencyOnly - SMTP Session Limits

subjectLogging - Logging and Notifications

subjectStart - Logging and Notifications

SuspiciousVirus - Virus Protection using ClamAV and OS-FileScanner

switchSpamLoverToScoring - SPAM Lover and SPAM Hater

switchTestToScoring - TestModes and SPAM Tagging

sworgValencePB - PenaltyBox - Message and IP Scoring

T

U

V

W