Mail::SpamAssassin::Conf - SpamAssassin配置指南

中国反垃圾邮件联盟 王兴宇 译

2004/11/13

名称

Mail::SpamAssassin::Conf — SpamAssassin 配置指南

中文版翻译 — 中国反垃圾邮件联盟 (http://anti-spam.org.cn) 王兴宇

如果译文中有任何错误，欢迎给我发邮件指出，不胜感激。如果希望讨论 SpamAssassin 的相关内容，请到中国反垃圾邮件联盟论坛的 SpamAssassin 版讨论。

译文版本 — 1.0

英文版原地址 — http://spamassassin.apache.org/full/3.0.x/dist/doc/Mail_SpamAssassin_Conf.html

---

纲要

```
 # 注释文本

 rewrite_header Subject        *****SPAM*****

 full PARA_A_2_C_OF_1618       /Paragraph .a.{0,10}2.{0,10}C. of S. 1618/i
 describe PARA_A_2_C_OF_1618   Claims compliance with senate bill 1618（根据上议院第1618号法令）

 header FROM_HAS_MIXED_NUMS    From =~ /\d+[a-z]+\d+\S*@/i
 describe FROM_HAS_MIXED_NUMS  From: contains numbers mixed in with letters（From: 信头中混和了数字）

 score A_HREF_TO_REMOVE        2.0

 lang es describe FROM_FORGED_HOTMAIL Forzado From: simula ser de hotmail.com（西班牙语：“ 信件假称其来自Hotmail.com ”）
```

描述

SpamAssassin 使用传统的UNIX风格的配置文件，这些配置文件可以保存在 /usr/share/spamassassin 和 /etc/mail/spamassassin 目录中。

以 # 开始的行是注释行，其中不包含任何有效的配置。文件中的空白字符是无所谓的，但是最好不要放在行首，因为将来可能使用行首空白来表示续行。

不过在当前，每个规则或配置必须放在一行，多行仍然不被支持。

路径中能使用 ~ 字符来表示用户的主目录。

下面在适当的情况下，默认值会被列在括号中。

用户参考

这些选项能被用于站点级配置和用户级配置中，通过它们可以定制 SpamAssassin 处理进入的邮件的方式。

评分选项

required_score n.nn (默认值: 5)

设定一个邮件被判定为垃圾邮件的分数线。n.nn 可以是整数或者实数。 默认值为5.0，对于单个用户使用，它基本上就可以符合需要了；但是对于应用于整个服务器，应该设置得更保守一些（更高一些），比如设置为8.0或10.0等等。通常不推荐设置为自动删除或丢弃那些被判定为垃圾邮件的邮件，否则可能招致用户的强烈抗议；除非评分特别的高，比如15.0或者更高，才可以考虑直接删除。这个选项以前称之为 required_hits ，现在虽然还可以用，但是不推荐使用旧的名称。

score 测试规则名 n.nn [n.nn n.nn n.nn]

指定一个测试规则的评分（命中后的评分）。评分可以是正的或者负的整数或实数。测试规则名是一个测试规则的名称，如： FROM_ENDS_IN_NUMS 。

如果只列出了一个评分，那么测试后总是返回该评分。

如果列出了4个评分，那么 SpamAssassin 在不同的使用情况下返回不同的评分。第一个评分用于贝叶斯测试和网络测试都被取消的情况下（0号评分集）；第二个评分用于贝叶斯测试被取消，但使用网络测试的情况下（1号评分集）；第三个评分用于使用贝叶斯测试，但网络测试被取消的情况下（2号评分集）；第四个评分用于贝叶斯测试和网络测试都使用的情况下（3号评分集）。

设置一个测试规则的评分为0将导致该规则根本不会进行测试。

如果评分使用括号“ () ”括起来，那么该行所有其后的评分都被增加同等的比例。例如，“ (3) ”表示在所有的评分集中提高该评分3点。“ (3) (0) (3) (0) ”表示提高0号和2号评分集3点评分。

如果一个测试规则没有给定评分，它将会被指定一个默认评分1.0；除非是以“ T_ ”开头的测试规则（用于指出该测试规则还在试验中），它会被指定为评分0.01。

注意，以“ __ ”开头的测试规则名是一个间接规则，它被用于组成元测试规则，这些规则不会被计算评分也不会列在测试命中报告中。注意，虽然间接规则的评分不会被计算，但是设置间接规则的评分为0将导致该规则的根本不会进行测试。

黑白名单选项

whitelist_from 邮件地址

它用于指定通常发送被误判为垃圾邮件的发信地址；它对于那些拥有很多律师的公司很适用，:> 一笑。 如果垃圾邮件发送者假称其是你白名单中的地址，那么可能会带来一些麻烦，所以一般不要使用它来绕开 SpamAssassin 的检测。如果你要把你自己的域放入到白名单中，你需要要知道，垃圾邮件发送者经常会假称其是来自你的域的。推荐的解决方法是使用下面介绍的 whitelist_from_rcvd 来替代这个选项。

白名单和黑名单都可以使用通配符。如 friend@somewhere.com ， *@isp.com ，或 *.domain.net 都是可以的。需要注意的是只支持 * 和 ?（匹配单个字符），其他的元字符匹配不被支持。由于一些安全的原因，这里也不支持正则表达式。

可以在一行中使用空格分隔开写入多个邮件地址，也可以使用多个 whitelist_from 行。

如果信头设置了 Resent-From 地址，那么就检查它，否则检查来自下列信头的全部地址：

Envelope-Sender
Resent-Sender
X-Envelope-From
From

此外，如果SMTP通讯中的信封信息可用的话，那么“ 信封发件人（envelope sender） ”也会被检查。

范例：

whitelist_from joe@example.com fred@example.com
whitelist_from *@example.com

unwhitelist_from 邮件地址

它用于覆盖 whitelist_from 选项。举例说，在 local.cf 中指定了一个站点级通用的白名单地址 whitelist_from，在用户级的配置 user_prefs 中可以使用这个选项来覆盖那个通用的白名单地址选项。这个选项所匹配的邮件地址也必须同样被前面的 whitelist_from 选项所匹配。

范例：

unwhitelist_from joe@example.com fred@example.com
unwhitelist_from *@example.com

whitelist_from_rcvd 邮件地址 该邮件中继的反向DNS解析名称

它对“ Received ”信头进行检查，是对 whitelist_from 的补充。 第一个参数是一个白名单的地址，第二个参数是该邮件中继点的反向DNS解析的名称。

第二个参数用于在邮件从互联网发送到你的内部网的邮件服务器时候进行的反向DNS查询匹配。它可以是一个完全限定的主机名或主机名的域部分，换言之，如果连接到你的邮件服务器的主机的IP可以被反向解析为“ sendinghost.spamassassin.org ”，那么你可以写为 sendinghost.spamassassin.org 或 spamassassin.org 。

注意，这需要你的 internal_networks 被正确配置。简言之，除了在一个复杂的网络里、或关闭了DNS检查、或在使用 -L 参数等情况下，设置这个参数可以得到不错的效果。

译者注：鉴于国内很多的ISP或IP拥有者很少进行IP反向解析的注册，所以，很多情况下IP的反向DNS解析并不能得到结果，那么设置此选项并无意义。

范例：

whitelist_from_rcvd joe@example.com example.com
whitelist_from_rcvd *@axkit.org sergeant.org

def_whitelist_from_rcvd 邮件地址 该邮件的反向DNS解析名称

类似于 whitelist_from_rcvd，但是它用于 SpamAssassin 的默认白名单中。这个白名单的评分

较低，因为它常常是垃圾邮件发送者假称的地址。

whitelist_allows_relays 邮件地址

指定 whitelist_from_rcvd 中的哪些邮件地址可以不使用 white_from_rcvd 中对应的邮件中继发信，而使用其它的中继服务器发信。默认情况下，发信地址在 white_from_rcvd 中，但是中继服务器却不是列出的那个，这种情形会命中一个测试伪造的测试规则。将该地址放入到 white_allows_relay 中防止命中。

白名单和黑名单都可以使用通配符。如 friend@somewhere.com， *@isp.com，或 *.domain.net 都是可以的。需要注意的是只支持 * 和？（匹配单个字符），其他的元字符匹配不被支持。由于安全的原因，这里不支持正则表达式。

可以在一行中使用空格分隔写入多个邮件地址，也可以使用多个 whitelist_allows_relays 行。

这里列出的邮件地址不必被前面的 whitelist_from_rcvd 所完全匹配，它只需要匹配信头中的地址就行。

范例：

```
whitelist_allows_relays joe@example.com fred@example.com
whitelist_allows_relays *@example.com
```

unwhitelist_from_rcvd 邮件地址

它用于覆盖 whitelist_from_rcvd 选项。举例说，在 local.cf 中指定了一个站点级通用的白名单地址 whitelist_from_rcvd，在用户级的配置 user_prefs 中可以使用这个选项来覆盖那个通用的白名单地址选项。所匹配的地址也必须同样被前面的 whitelist_from_rcvd 选项所匹配。

范例：

```
unwhitelist_from_rcvd joe@example.com fred@example.com
unwhitelist_from_rcvd *@axkit.org
```

blacklist_from 邮件地址

它用于指定那些通常被漏判为非垃圾邮件的垃圾邮件发送地址。格式与 whitelist_from 相同。

unblacklist_from 邮件地址

它用于覆盖 blacklist_from 选项。举例说，在 local.cf 中指定了一个站点级通用的黑名单地址 blacklist_from，在用户级的配置 user_prefs 中可以使用这个选项来覆盖那个通用的黑名单地址选项。

范例：

```
unblacklist_from joe@example.com fred@example.com
unblacklist_from *@spammer.com
```

whitelist_to 邮件地址

如果给定的地址出现在信头中的收信人那里（Resent-To, To, Cc, 明显的信封收件人等），邮件将作为非垃圾邮件处理。 它常用于整个站点使用了 SpamAssassin 但是某些用户不希望他们的任何邮件被过滤。与 whitelist_from 的格式相同。

有三个级别的接收白名单：whitelist_to， more_spam_to 和 all_spam_to。 在第一个接收白名单中的用户仍然可能会被过滤一些垃圾邮件，但是在最后一个接收白名单中的用户不会被过滤任何垃圾邮件。

白名单信头检查将按照如下顺序，如果设置了 Resent-To 或 Resent-Cc 就使用它们，否则检查来自下列信头中的全部地址：

To
Cc
Apparently-To
Delivered-To
Envelope-Recipients
Apparently-Resent-To
X-Envelope-To
Envelope-To
X-Delivered-To
X-Original-To
X-Rcpt-To
X-Real-To

more_spam_to 邮件地址

参见上面。

all_spam_to 邮件地址

参见上面。

blacklist_to 邮件地址

如果给定的地址出现在信头的收件人中（Resent-To, To, Cc, 明显的信封收件人等），邮件将被作为垃圾邮件处理。与 blacklist_from 的格式相同。

基本消息标记选项

rewrite_header { subject | from | to } 字符串

默认情况下，SpamAssassin 不会对那些被判定为垃圾邮件的信件主题、发信人和收信人等信息进行修改，以标识其是垃圾邮件。如果设置了这个选项，信件主题、发信人和收信人会被加上特定的字符串来表明该邮件是垃圾邮件。对于发信人和收信人的修改是在地址后面增加一个括在括号里面的RFC 2822格式的注释；对于信件主题的修改则是替换原先的主题。注意，在 report_safe 没有设置为0的情况下，你可以使用 _REQD_ 和 _SCORE_ 标记来重写信件主题，否则你也许不能通过正常的方式去掉 SpamAssassin 的标记。在重写发信人和收信人时，字符串不能包含圆括号（会被转换为方括号）。

add_header { spam | ham | all } 信头 字符串

可以对各种类型的信件（垃圾邮件、非垃圾邮件和全部邮件）增加 SpamAssassin 的定制信头。所有的定制信头都会以 X-Spam- 开始（如信头 Foo 将显示为 X-Spam-Foo ）。信头只能使用下列字符：所有的大小写英文字符、所有的数字和下划线及中划线。([A-Za-z0-9_-])。

字符串可以包含下面提到的“ 模板标记 ”。如果需要的话，还可以使用\n和\t来增加回车符和制表符。使用\\来表示一个反斜线字符。其它的转义字符无效，只被简单的去掉反斜线。

如果 fold_headers 被设置为1，信头会被折叠起来（ 即通过行首空格进行续行，以避免较长的行）。但是注意，通过\n手工换行的信头将不会被自动折叠（即可能会出现很长的信头），即使这个信头需要折叠起来。

你能够通过 add_header 来改变部分已有的 X-Spam- 信头（如 SpamAssassin 默认增加的）。

删除特定的信头请参见 clear_headers 。

以下是一些例子（这些是默认增加的）：

```
 add_header spam Flag _YESNOCAPS_
 add_header all Status _YESNO_, score=_SCORE_ required=_REQD_ tests=_TESTS_
autolearn=_AUTOLEARN_ version=_VERSION_
 add_header all Level _STARS(*)_
 add_header all Checker-Version SpamAssassin _VERSION_ (_SUBVERSION_) on
_HOSTNAME_
```

remove_header { spam | ham | all } 信头

可以删除各种类型的信件（垃圾邮件、非垃圾邮件和全部邮件）中的 SpamAssassin 的定制信头，这些信头是以 X-Spam- 开头的（所以信头名应该前缀以 X-Spam- ）。

删除全部的 SpamAssassin 定制信头参见 clear_headers 。

注意，X-Spam-Checker-Version 信头是不能删除的，因为邮件管理员和开发人员需要使用它来诊断问题。如果没有这个信头，甚至都不知道 SpamAssassin 是否在运行。

clear_headers

清除全部的 SpamAssassin 定制信头。你可以在任何的 add_header 前使用这个，以防止默认的 SpamAssassin 信头被添加到信头中。

注意，X-Spam-Checker-Version 信头是不能被删除的，因为邮件管理员和开发人员需要使用它来诊断问题。如果没有这个信头，甚至都不知道 SpamAssassin 是否运行。

report_safe { 0 | 1 | 2 } (默认值: 1)

如果这个选项被设置为1，当收到的信件被判定为垃圾邮件时，不是修改原信件，而是创建一个新的报告信件，并且将原信件作为一个RFC 822格式的附件附上（确保原信件保持原样，且容易恢复）。

如果这个选项被设置为2，原信件以文本方式附加到报告信件中。之所以采用这个选项是由于安全的原因，某些不完善的邮件客户端会在用户没有要求的情况下自动的载入附件，这可能会带来一些安全问题。这个选项也许会导致附加的信件和原信件保存出来或看起来并不太一样。

如果这个选项设置为0，收到的垃圾邮件只在信头中增加一些 X-Spam- 信头而不修改信体。此外，X-Spam-Report 信头会被增加到垃圾邮件中，你可以设置 report_safe 为0后使用 remove_header 来去掉这些 SpamAssassin 的定制信头。

如果你要复制原信件的信头到被判定的邮件中，参见 report_safe_copy_headers 。

语言选项

ok_languages xx [yy zz ...] (默认值: all)

这个选项指定了使用那些语言的邮件可以被处理。 SpamAssassin 会试图通过邮件中的文本自动检测所用的语言。

注意，所用语言并不是总能顺利的识别，如果不能识别，SpamAssassin 将不处理该邮件。

这个选项的值决定了 UNWANTED_LANGUAGE_BODY 规则的触发条件。

你可以使用小写的二或三字符语言类型缩写，而不是使用该语言的英文全名。如果你使用的语言没有在下面列出或你要允许任何语言，你可以使用配置值 all 。默认的配置值是 all 。

范例：

```
ok_languages all       (允许所有语言)
ok_languages en        (只允许英文)
ok_languages en zh ja  (允许英文、中文和日文)
```

注意，如果指定了多个 ok_languages 行则只有最后一个有效。

可用的语言如下：

af - 南非荷兰语
am - 埃塞俄比亚的阿姆哈拉语
ar - 阿拉伯语
be - 白俄罗斯语
bg - 保加利亚语
bs - 波斯尼亚语
ca - 西班牙的加泰罗尼亚语
cs - 捷克语
cy - 英国的威尔士语
da - 丹麦语
de - 德语
el - 希腊语

en - 英语
eo - 世界语
es - 西班牙语
et - 爱沙尼亚语
eu - 巴斯克语
fa - 波斯语
fi - 芬兰语
fr - 法语
fy - 荷兰的语弗里斯兰
ga - 爱尔兰的盖尔语
gd - 苏格兰的盖尔语
he - 希波来语
hi - 北印度语
hr - 克罗地亚语
hu - 匈牙利语
hy - 亚美尼亚语
id - 印尼语
is - 冰岛语
it - 意大利语
ja - 日本语
ka - 乔治亚语
ko - 韩语
la - 拉丁语
lt - 立陶宛语
lv - 拉脱维亚语
mr - 马拉地语
ms - 马来语
ne - 尼泊尔语
nl - 荷兰语
no - 挪威语
pl - 波兰语
pt - 葡萄牙语
qu - 盖丘亚语
rm - 磊蒂亚-罗曼语
ro - 罗马尼亚语
ru - 俄罗斯语
sa - 梵语
sco - 苏格兰语
sk - 斯洛伐克语
sl - 斯洛文尼亚语
sq - 阿尔巴尼亚语
sr - 塞尔维亚语
sv - 瑞典语
sw - 班图语
ta - 泰米尔语
th - 泰国语
tl - 塔加路语
tr - 土耳其语
uk - 乌克兰语
vi - 越南语
yi - 犹太人的依地语
zh - 中文（包括简体和繁体）

zh.big5 - 中文（繁体）
zh.gb2312 - 中文（简体）

ok_locales xx [yy zz ...] (默认值: all)
这个选项指定了那些地区性（国家代码）的邮件可以被处理。使用这些国家的语言字符集的邮件不会被标记为外文垃圾邮件。

如果你收到了很多外文的垃圾邮件，而且不能收到这种语言写非垃圾邮件，这个选项也许会有帮助。注意，所有的ISO-8859-*字符集和Windows代码页字符集默认总是可以使用的。

设置为all可以允许所有的字符集。这是默认值。

这个选项的设置决定了规则 CHARSET_FARAWAY、CHARSET_FARAWAY_BODY 和 CHARSET_FARAWAY_HEADERS 的触发条件。

范例：

ok_locales all (允许全部地区)
ok_locales en (仅允许全部地区)
ok_locales en zh ja (仅允许英文、中文和日文)

注意，如果指定了多个 ok_locales 行则只有最后一个有效。

可用的地区如下：

en - 西方通用字符集
ja - 日文字符集
ko - 韩文字符集
ru - 斯拉夫语字符集
th - 泰语字符集
zh - 中文（包括简体和繁体）字符集

网络测试选项

use_dcc (0 | 1) (默认值: 1)
是否使用DCC（分布式校验值交换中心），它是一个类似于Razor的系统。

dcc_timeout n (默认值: 10)
设置等待DCC返回结果的超时为多少秒。

dcc_body_max 数值
dcc_fuz1_max 数值
dcc_fuz2_max 数值
这个选项设置了 SpamAssassin 判定DCC校验值匹配时，这个邮件的 body/fuz1/fuz2 校验值

必须被报告多少次。

几乎所有的DCC客户端都会自动的汇报它们处理邮件生成的校验值，所以你应该设置这个值为一个尽量高的值，如：999999 （这是DCC多数的计数值）。

这些选项的默认值都是999999。

use_pyzor (0 | 1) (默认值: 1)
是否使用 Pyzor，它也是一个类似于 Razor 的系统。

pyzor_timeout n (默认值: 10)
设置等待Pyzor返回结果的超时为多少秒。

pyzor_max 数值

这个选项设置了 SpamAssassin 判定 Pyzor 校验值匹配时，这个邮件的 body 校验值必须被报告多少次。

选项的默认值是5。

pyzor_options [选项 ...]
Pyzor 的命令行附加选项。注意，由于安全的原因，只允许使用全部大小写字母、全部数字、下划线、中划线和斜线（[A-Za-z0-9_-/]）。

spamcop_from_address 邮件地址 (默认值: none)
设定手工报告到 SpamCop 时的发信人地址。你可以使用你正常通讯的邮件地址。如果没有设置这个选项的话，将使用一个匿名地址（none）作为发信地址。

spamcop_to_address 邮件地址 (默认值: 报告地址)
你个人的SpamCop的报告邮件地址。你需要在 http://www.spamcop.net/ 注册以获得一个报告地址。如果没有设置这个选项的话，您将报告到 SpamCop 的一个通用报告地址，这样的报告只能在 SpamCop 获得一个较低的权重。

trusted_networks IP地址[/掩码] ... (默认值: none)
这个选项设置信任网络或主机。“ 信任 ”是指这些网络上的中继主机确信其不会被垃圾邮件发送者所操纵、也不会是开放转发和开放代理。一个信任主机能够毫无顾忌的转发垃圾邮件而不会被识别，甚至不需要伪造信头。SpamAssassin 也绝不会在 DNS黑名单中查询信任网络中的主机。

在你的域中的邮件服务器和内部中继服务器应使用 internal_networks 指定。当有除了你的邮件服务器和内部中继服务器外的可信任的主机时，可以在 trusted_networks 指定它们。

如果指定了一个 / 及其后的掩码，这是一个CIDR风格的网络地址；如果没有指定掩码，但是有少于4个的IP地址单元并后缀以一个点的话，它是指所有前面的IP地址单元相同的网络地址；如果没有指定掩码也没有后缀的点，它是指一个单一IP地址，就像指定了/32掩码一样。

范例：

```
trusted_networks 192.168/16 127/8       # 全部的 192.168.*.* 和 127.*.*.*
trusted_networks 212.17.35.15           # 仅指该IP地址
trusted_networks 127.                   # 全部的 127.*.*.*
```

trusted_networks 的定义是叠加的，多个选项指定的信任网络都会作为信任网络。可以使用 clear_trusted_networks 清除前面定义的信任网络。

如果没有设置这个选项，但是设置了 internal_networks，那么 internal_networks 的值将作为这个选项的默认值。

如果你打开了DNS检查功能，SpamAssassin 可以即时推算你的信任网络，所以这个选项并不是十分必要。（感谢 Scott Banister 和 Andrew Flury 提出这种推算的思路。）推算流程如下：

- 如果信头中最后（最顶部）一个Received行的“ from ”地址与“ by ”地址是相同的一个/16网段内，那么它是可信任的。
- 如果“ from ”地址是私有保留网段的，那么它是可信任的。
- 如果“ by ”的地址是私有保留网段的，那么它是可信任的。

clear_trusted_networks
清除前面定义的信任网络列表。

internal_networks IP地址[/掩码] ... (默认值: none)
这个选项设置内部网络或主机。“ 内部 ”是指在该网络的中继服务器是你的域中的邮件服务器或内部中继。它的格式同上面的 trusted_networks 一样。

该选项用于在检查拨号或动态IP地址黑名单时，它用来检测“ 单跳发送（direct-to-MX）”的垃圾邮件。信任的中继会直接从拨号连接接受邮件而不需要它们列在 internal_networks 中，它们只需列在 trusted_networks 中。

如果设置了 trusted_networks 而没有设置 internal_networks，那么 trusted_networks 的值将作为这个选项的默认值。

如果既没有指定 trusted_networks 也没有指定 internal_networks，即没有本地地址。换言之，任何连接到运行 SpamAssassin 的主机的主机都被认为是外部的。

clear_internal_networks
清除前面定义的内部网络列表。

use_razor2 (0 | 1) (默认值: 1)
是否使用Razor2。

razor_timeout n (默认值: 10)
设置等待Razor返回结果的超时为多少秒。

skip_rbl_checks { 0 | 1 } (默认值: 0)

默认情况下， SpamAssassin 会运行RBL（实时黑名单，也叫做DNSBL）检查。如果你的ISP已经为你做了RBL检查，你可以该值设置为1来跳过这个检查。

译者注：国外的多数RBL由于对国内的情况不是很了解，所以对中国的IP地址进入RBL的处理比较轻率，通常不建议使用国外的RBL。替代的选择可以使用中国反垃圾邮件联盟推出的RBL服务。

rbl_timeout n (默认值: 15)

所有的DNS查询都是在整个测试开始的时候进行，并且在整个测试结束时候读取结果。这个选项设置了最大的DNS查询等待时间。在大多数DNS查询都成功完成的情况下，SpamAssassin 将不会浪费时间来等待剩下的那些查询，可能它们根本没有回应了。当剩余没有完成的请求越少时，等待的时候也越短。对于默认的15秒的等待时间，下面是一个表格说明了当剩余请求有多少时会等待多少时间：

剩余的查询	100%	90%	80%	70%	60%	50%	40%	30%	20%	10%	0%
等待时间	15	15	14	14	13	11	10	8	5	3	0

此外，只要其它的查询返回了结果，剩下的查询的等待时间都会变得更短，但是等待时间总不会超过 rbl_timeout 所指定的时间。

举个例子，如果邮件检查开始时候有20个查询，当有16个（剩下20%）返回了结果后，剩下的4个查询必须在开始后的5秒钟内完成，否则就会放弃这几个查询。

dns_available { yes | test[: 服务器1 服务器2...] | no } (默认值: test)

默认情况下， SpamAssassin 会查询一些默认主机以判断DNS是否工作正常。做这个测试的原因是有可能因为网络链接断开导致的延时和某些情况下由于连接失败导致的DNS不可用。SpamAssassin 默认包括了13个测试的服务器，并且每次随机取出其中3个测试。

你可以指定你自己的测试列表：

```
dns_available test: domain1.tld domain2.tld domain3.tld
```

注意， DNS检查的是NS记录。

SpamAssassin 的网络规则测试是并发进行的。这也许会导致需要打开的文件描述符超过了系统限制，推荐将文件描述符的限制至少增加到256以上。

学习选项

use_bayes (0 | 1) (默认值: 1)

是否使用 SpamAssassin 内建的朴素贝叶斯（Bayes）风格的分类器。这是所有的贝叶斯相关的选项的主开关。

use_bayes_rules (0 | 1) (默认值: 1)

是否使用 SpamAssassin 内建的朴素贝叶斯风格的分类器规则。这个选项允许你在停止自动学习使用手动学习的情况下禁止那些贝叶斯规则。

auto_whitelist_factor n (默认值: 0.5, 范围 [0 至 1])

该选项设置了历史平均评分如何影响邮件的最终评分。基本上，规则是跟踪发信人所发信件的历史平均评分，当我们计算了某邮件的实际评分后，其最终评分为：

最终评分 = 实际评分 + (平均评分 - 实际评分) * 因数

就是说，如果因数（factor）设置为0.5，那么邮件的评分会向平均评分偏移一半的差值；如果因数设置为0.3，那么邮件的评分会向平均评分偏移三分之一的差值；如果因数设置为1，其评分总是使用平均值；如果因数设置为0，其评分总是使用该邮件实际的评分。

auto_whitelist_db_modules 模块名 ... (默认值: 见下)
指定了那个数据库模块用于自动白名单（auto-whitelist）存储数据文件。SpamAssassin 使用能被PERL正确载入的第一个模块。选项格式为：

首选模块 第二个可选模块 第三个可选模块 ...

即，使用空格分隔开的模块名。默认为：

DB_File GDBM_File NDBM_File SDBM_File

bayes_auto_learn (0 | 1) (默认值: 1)
设置 SpamAssassin 是否自动通过高评分邮件（或低评分邮件，用于非垃圾邮件）的“ 学习 ”提高系统识别能力。目前“ 学习 ”只支持朴素贝叶斯风格的分类器。

注意，决定是否学习时，以下情况将被忽略（不学习）：

- tflags 设置为“ learn ”（贝叶斯规则）

- tflags 设置为“ userconf ”（用户白名单/黑名单规则等）

- tflags 设置为“ noautolearn ”

此外，自动学习与当前使用哪个评分集有关，它仅仅发生在使用评分集0或评分集1的时候。邮件测试所得的评分和自动学习的评分是不同的。

bayes_auto_learn_threshold_nonspam n.nn (默认值: 0.1)
低于该评分线的邮件将自动的作为非垃圾邮件学习。

bayes_auto_learn_threshold_spam n.nn (默认值: 12.0)
高于该评分线的邮件将自动的作为垃圾邮件学习。

注意， SpamAssassin 要求邮件信头至少有评分3，邮件信体至少有评分3才会自动作为垃圾邮件学习。因此，这个选项的最小值不能小于6。

bayes_ignore_header 信头
如果你收到的邮件被上游邮件系统过滤过，比如ISP的邮件过滤或邮件列表的过滤，且这些过滤增加了新的信头（多数如此），这些信头可能会给贝叶斯分类器一些不正确的指示。为了避免这种情况，可以使用这个选项列出这些信头：如：

```
bayes_ignore_header X-Upstream-Spamfilter
bayes_ignore_header X-Upstream-SomethingElse
```

bayes_ignore_from 邮件地址

贝叶斯分类器和自动学习功能不会处理来自这里列出地址的邮件。如果 sa-learn 使用 --use-ignores 选项的话，也会忽略这些邮件。可以列出一个或多个地址，格式参见 whitelist_from 。

来自特定发件人的垃圾邮件也许包含了许多经常出现在非垃圾邮件中的词汇。举个例子，某个人也许会收取他常去的书店的邮件，但是不希望收到来自其他书店的类似邮件。如果那些不想收到的信件作为垃圾邮件被学习的话，那么任何讨论书籍的邮件，包括来自他想收到的书店的邮件也很可能被判定为垃圾邮件。这个让人麻烦的书店的邮件地址应该列在这里。

那些发送合理的垃圾邮件的人，或收到了包含了垃圾邮件常见词汇的非垃圾邮件的人，可能会担心一些垃圾邮件被作为非垃圾邮件处理。发送垃圾邮件的邮件列表和地址等可以放到这个列表中。

bayes_ignore_to 邮件地址

贝叶斯分类器和自动学习功能不会处理发送到这里列出地址的邮件。参见 bayes_ignore_from 。

bayes_min_ham_num (默认值: 200)
bayes_min_spam_num (默认值: 200)

为正确起见，贝叶斯分类器在一定数量的垃圾邮件和非垃圾邮件被学习之前不会被启用。他们的默认值是200，你可以针对需要调高或降低这两个数值。

bayes_learn_during_report (默认值: 1)

贝叶斯系统默认情况下会学习那些被分析并报告为垃圾邮件的邮件（ spamassassin -r ）。你可以设置这个选项为0来关闭这个学习功能。

bayes_sql_override_username

用于 BayesStore::SQL 存储。

如果设置了这个选项， BayesStore::SQL 模块所设置的 username 将被覆盖。这个选项用于实现一个全局或分组的贝叶斯数据库。

bayes_use_hapaxes (默认值: 1)

指定贝叶斯分类器是否使用 hapaxes (仅仅出现了一次的词汇/字串）。它能提高命中率但是会增大数据库的大小。

bayes_use_chi2_combining (默认值: 1)

指定贝叶斯分类器是否使用卡方（chi-squared）合并替代 Robinson/Graham 式的朴素贝叶斯合并。卡方算法会生成更“敏感”的结果，但是也许不太在乎邮件的大小等信息。

译者注：根据 SpamAssassin 维护团队的经验，卡方合并要比传统的朴素贝叶斯合并的效果要好，所以这里默认是1。

bayes_journal_max_size (默认值: 102400)

SpamAssassin 将不定时的同步日志和数据库。通常是一天做一次同步，但是如果日志文件大小超过了这个选项所设置的值，将会同步更多次。该值的单位是字节。如果该值设置为0，同步就不再进行。

bayes_expiry_max_db_size (默认值: 150000)

指定了贝叶斯字串数据库的最大大小。当达到了最大大小时，贝叶斯系统将视乎大小，保留原来的75%且至少有10万个字串。一般15万个字串的数据库会占用8M的空间。

bayes_auto_expire (默认值: 1)

如果设置为1，贝叶斯系统将自动丢弃旧的字串。仅在数据库中的字串数量超过了 bayes_expiry_max_db_size 时才会自动丢弃旧的字串。

bayes_learn_to_journal (默认值: 0)

如果设置了这个选项，那么 SpamAssassin 在学习的时候会将结果写入到日志而不是直接写入到数据库中。降低了更新时对数据库的锁定情况的发生，但是也导致了对日志文件的更多读写和数据库更新的延迟。

其它选项

lock_method 方式

选择一个文件锁定方式来保护磁盘上的数据库文件。默认情况下，在UNIX上 SpamAssassin 会使用“ NFS安全 ”锁定方式；不过，如果你确认你用于贝叶斯和自动白名单的数据库决不会通过NFS方式来访问，你可以“ 非NFS安全 ”的锁定方式。这会相对快一些，但是如果同时有一个或多个通过NFS方式访问的客户端访问时，可能会造成数据库文件的破坏。

注意，不同的操作系统使用不同的锁定方式。

支持下列锁定方式：

nfssafe - 一个“ NFS安全 ”的锁定方式
flock - 简单的UNIX flock() 锁定
win32 - Win32 平台上使用 sysopen (..., O_CREAT|O_EXCL) 方式锁定

nfssafe 和 flock 只能用于 UNIX 上，win32 只能用于 Windows 。默认情况下，SpamAssassin 根据操作系统的不同使用 nfssafe 或 win32 锁定方式。

fold_headers { 0 | 1 } (默认值: 1)

默认情况下，SpamAssassin 添加的信头会使用空白进行折叠。换言之，它们将会断成多行而不是使用一个很长的行，其后的行添加前置的制表符来表示对前一行的续行。

可以通过这个选项来禁止折叠，不过要注意可能会生成很长的行。

report_safe_copy_headers 信头 ...

如果使用 report_safe，一些原邮件的信头被复制到包装的信头里面（From, To, Cc, Subject, Date 等）。如果你希望其他的信头也被复制到这里，你可以使用这个选项。你可以在一行里面使用空格分隔开列出多个信头，或者使用多行这个选项。

envelope_sender_header 信头
如果SMTP服务器提供了“ MAIL FROM: ”通讯信息（信封发信人），SpamAssassin 会试图从邮件中发现该信息。这个选项用于指定信封发信人这个“ 伪信头 ”，这个“ 伪信头 ”可用于各种检查，比如SPF等。

默认情况下，几种MTA使用不同的信头，如：

X-Envelope-From
Envelope-Sender
X-Sender
Return-Path

如果可以通过查找一些特征（比如所邮件中所替换的特定信头，或fetchmail的特定信头等）可以安全的确定这些，那么 SpamAssassin 会使用它们。 然而，某些邮件服务器的配置可能会导致选择了错误的信头。（更多的讨论请参见 SpamAssassin 的 BugZilla 里面的2142号错误。）

为了避免选择错误，可以使用 envelope_sender_header 来指明这个信头。这个信头中包含了用于指明信封发信人的地址。

如果信头像在SMTP通讯中一样包含了“ < ”或“ > ”字符，这两个字符将被去掉。

如果该信头没有找到或者信头中没有包含“ @ ”符号， SpamAssassin 将自动使用查找特征的方式来确定这个信封发信人。

（MTA开发者注意，我们希望将来使用一个单独的与其后的垃圾邮件扫描器不同的信头。http://wiki.apache.org/spamassassin/EnvelopeSenderInReceived 是一个更好的使用 Received 信头的建议）

范例：

envelope_sender_header X-SA-Exim-Mail-From

describe 测试规则名 描述 ...
用于描述一个测试规则。这个描述会出现在细节报告中。

注意，以“ __ ”开始的测试是为元规则所保留，它们不会被计分和列出在“ 命中的测试 ”报告中。

同时注意，习惯上描述文本不要超过50个字符。

report_charset 字符集 (默认值: 无)
设置附加了垃圾邮件原信件的报告邮件文本的 MIME Content-Type 的字符集。

report 报告模板文字
设置附加了垃圾邮件的报告邮件的报告模板，参见 /usr/share/spamassassin 中的 10_misc.cf 中的例子。

如果你设置了这个，请不要超过每行78个字符。每个 report 行累加到现存的模板上（前面的 report 行），可以使用 clear_report_template 来清除前面的模板定义。

能够使用上面说的特定标记。

clear_report_template
清除 report 模板。

report_contact 联系地址
设置上面报告中使用的 _CONTACTADDRESS_ 的值。默认值是“ the administrator of that system ”，后跟上运行本软件的系统的主机名。

report_hostname 使用的主机名
设置上面报告中使用的 _HOSTNAME_ 的值。默认情况下是运行本软件的主机名。

unsafe_report 报告模板文字.
设置附加了包含非文本的垃圾邮件的报告邮件的报告模板，参见 /usr/share/spamassassin 中的 10_misc.cf 中的例子。

每个 unsafe_report 行累加到现存的模板上（前面的 unsafe_report 行），可以使用 clear_unsafe_report_template 来清除前面的模板定义。

能够使用上面说明的特定标记。

clear_unsafe_report_template
清除 unsafe_report 模板。

规则定义和特权设置

这些设置与上面的设置不同，它们被称之为“ 特权设置 ”。只有用户在通过 procmailrc 文件或 forward 文件调用 SpamAssassin 时，或在系统管理员编辑 /etc/mail/spamassassin 下的配置文件时才能使用它们。由于安全及效率的原因，通过 spamc 来访问 spamd 的用户是不允许在他们的 user_prefs 文件中使用这些“ 特权设置 ”，除非设置了 allow_user_rules 选项（而且，也只能使用下面列出的这些特权设置）。

allow_user_rules { 0 | 1 } (默认值: 0)
这个选项允许用户在他们的 user_prefs 中创建可以用于 spamd 的规则（也只能创建规则）。默认是不允许用户创建规则的，因为这样可能会造成一些安全漏洞，如果 spamd 是以 root 身份运行的话，就有可能授予了用户 root 级别的访问权限。这并不是一个好的做法，除非你能够通过别的方法确保安全地运行用户规则。如果不是很有把握，不要打开这个选项。 此外，这个选项会导致每当一封邮件递交给用户时，如果他的 user_prefs 里面有自己定义的规则，那么 SpamAssassin 每次都会重新编译所有的规则，这会显著的增大服务器的负载。所以强烈建议不要允许用户自己定义规则！

注意，现在即便打开了这个选项， spamd 也不会使用 user_prefs 中的规则来修改系统已经存在的规则定义。

header 测试规则名 信头 操作符 /模式/修饰符 [if-unset: 字符串]
这个选项用于定义一个信头测试规则。测试规则名 是一个测试规则名，
如“ FROM_ENDS_IN_NUMS ”。信头是一个邮件信头名，如“ Subject ”、“ To ”等。

信头后面后缀上“ :raw ”可以防止使用 quoted-printable 或 base-64 编码的字符串自动进行解码。

信头后面后缀上“ :addr ”可以去掉除了信头中的第一个邮件地址外的其他部分。例如，以下所有信头处理后都只剩下“ example@foo ”了：

example@foo
example@foo (Foo Blah)
example@foo, example@bar
display: example@foo (Foo Blah), example@bar ;
Foo Blah <example@foo>
``Foo Blah'' <example@foo>
```'Foo Blah''' <example@foo>

信头后面后缀上“ :name ”可以去掉除了信头中第一个真实名字外的其他部分。例如, 以下所有信头处理后只剩下“ Foo Blah ”了：

example@foo (Foo Blah)
example@foo (Foo Blah), example@bar
display: example@foo (Foo Blah), example@bar ;
Foo Blah <example@foo>
``Foo Blah'' <example@foo>
```'Foo Blah''' <example@foo>

可以使用以下的几个“ 伪 ”信头：

ALL 表示所有的任何信头。
ToCc 表示“ To ”和“ Cc ”信头。
EnvelopeFrom 是SMTP通讯过程中，如果“ MAIL FROM: ”通讯信息可以传递到信头中的话，使用这个“ 伪 ”信头来代表该信息。
MESSAGEID 表示信件中所有的 Message-Id 信头。一些邮件列表软件会将原来的 Message-Id 信头改名为 Resent-Message-Id 或 X-Message-Id ，然后使用自己的 Message-Id 信头。这个“ 伪 ”信头返回以上全部三种信头，使用回车符分隔开。

操作符是 =~ （符合其后的正则表达式）或 !~ （不符合其后的正则表达式）。 模式是一个 Perl 风格的正则表达式，修饰符是对正则表达式的修饰（请参见 Perl 中有关正则表达式的部分）。 注意，即便你使用了 x 修饰符，也不支持多行的正则表达式。

如果使用了 [if-unset: 字符串] 标记，那么如果邮件中没有发现该信头，就会使用该字符串来进行模式匹配。

测试规则名不能用数字开头，只能使用英文字母、数字和下划线。建议不要使用小写字

母、名字不要超过22个字符。也不能使用中划线。

注意，以“ __ ”（两个下划线）开头的测试规则被保留用于元规则，他们不会被计分和列在“ 测试命中 ”报告中。以“ T_ ”开头的测试规则被保留用于试验，它们应该给予很低的评分。

如果你增加或修改一个测试规则，请使用 spamassassin --lint 来测试一下是否有语法错误。这可以避免出现错误消息或导致其他的测试被忽略。

header 测试规则名 exists:信头
定义一个“ 信头存在 ”测试规则。信头是一个要测试存在与否的信头名。这是上面的信头测试规则的一个简化版本。

header 测试规则名 eval:评估函数([参数])
定义一个邮件信头的评估测试。评估函数是Mail::SpamAssassin::EvalTests 中定义的测试函数。参数是可选的。

header 测试规则名 eval:check_rbl('名单名称', '名单地址' [, '测试结果'])
检查一个 DNSBL （DNS方式的黑名单RBL或白名单）。它会从邮件的 Received: 信头中取出所有的IP地址，如果IP地址不在 trusted_networks 中，那么对这些IP地址进行DNSBL查询。以下几点需要注意：
重复或保留的IP地址
重复的IP地址仅仅被查询一次。保留的IP地址（如192.168.0.1、127.0.0.1等）不查询。保留的IP地址列在 http://www.iana.org/assignments/ipv4-address-space， http://duxcw.com/faq/network/privip.htm， http://duxcw.com/faq/network/autoip.htm 或 ftp://ftp.rfc-editor.org/in-notes/rfc3330.txt 中。

“ 名单名称 ”参数
它也称作“ zone ID ”（译者注：即给所查询的DNSBL的一个名字，如用 CBLPLUS 代表 cblplus.anti-spam.org.cn. ）。如果你要查询像 NJABL 或 SORBS 这样的由多个黑名单合并而成的多重DNSBL,你可以通过 check_rbl_sub() 使用它来比较查询返回的结果。

如果上面取出的多个IP地址的DNSBL查询不止一个返回了命中结果，并不会重复计分，因为对于每封邮件来说，该规则只被触发一次（即只要有一个IP地址在黑名单中就算触发了该规则）。

“ 名单地址 ”参数
这是DNSBL的根区（译者注：即DNSBL的服务地址，如cblplus.anti-spam.org.cn.），使用点结尾。

“ 测试结果 ”参数
这是一个和下面的 check_rbl_sub() 一样的子测试参数，它是可选的。

查询除了第一跳外的所有IP地址
可以在“ 名单名称 ”后缀上“ -notfirsthop ”来查询除了第一跳外的所有IP地址。它被用于在查询动态地址黑名单（动态地址是用于动态的分配给拨号、ISDN、ADSL等连接的IP地址）。 邮件的第一跳也许是动态地址，但是至少应该有一跳以上的邮件递交跳数，这是合理的情况，所以在这种情况下不应该因第一跳在动态地址黑名单里面而增加评分。但是

如果只有一跳，那么无论如何都会被查询的，因为邮件应该是通过它的外发邮件服务器进行递交的，而不是直接发送到接收邮件的服务器。

查询可信任与否的IP地址
当查询一个“ 正向 ”的DNSBL（DNS白名单）时，你不能信任那些“ Received ”信头中没有被列在可信任中继名单中的IP地址（信任的中继是不进行查询的）。为了查询第一个可信任的IP地址，可以在“ 名单名称 ”名称后缀上“ -firsttrusted ”，这会查询连接到最远可信任的中继的IP地址。

此外，你能够通过后缀“ -untrusted ”来查询所有不信任的IP地址。

注意，这需要 SpamAssassin 能知道那个中继是可信任的。在简单的环境里， SpamAssassin 能够很好的自行推测。在复杂的环境，通过手工设定 trusted_networks 可以得到更好的结果。

header 测试规则名 eval:check_rbl_txt('名单名称', '名单地址')
同 check_rbl() 一样，只是查询的是TXT类型的DNS记录而不是A类型的DNS记录。如果所查询的DNSBL支持TXT查询，返回的结果是一行文字，用来说明该地址被列入黑名单的原因，通常是一个可以查询黑名单数据库的链接。

header 测试规则名 eval:check_rbl_sub('名单名称', '测试结果')
创建某DNSBL查询的子测试。如果你要查询一个像 relays.osirusoft.com 那样的多重DNSBL，你可以使用对应的“ 名单名称 ”来比较 chek_rbl 查询得到结果。如果DNSBL查询返回多个IP地址时，“ 测试结果 ”可以是一个IPv4地址；如果DNSBL查询返回一个包含掩码的IP地址时，“ 测试结果 ”可以是一个代表掩码的正十进制整数；如果是一个SenderBase查询（对 sa.senderbase.org 的TXT查询），“ 测试结果 ”是一个以“ sb: ”开头的表达式；如果前面的都不符合，它还可以是一个正则表达式。

注意：这个“ 名单名称 ”必须和前面的 check_rbl() 中的名字完全一样，包括后缀的“ -notfirsthop ”等。

译者注：中国反垃圾邮件联盟所推出的DNSBL服务的设置如下，它们可以放到 /etc/mail/spamassassin/local.cf 中，但是不必全部放入和使用，通常根据需要使用其中一个就可以了，推荐使用CBL-。设定的评分可以自己的情况自行调整：
CBL（返回值是127.0.8.2）：

```
header   RCVD_IN_CASA_CBL eval:check_rbl('CBL','cbl.anti-spam.org.cn.')
describe RCVD_IN_CASA_CBL Relay in CASA CBL, http://anti-spam.org.cn/services/rbl.php#cbl
tflags   RCVD_IN_CASA_CBL net
score    RCVD_IN_CASA_CBL 2.0
```

CDL（返回值是127.0.8.4）：

```
header   RCVD_IN_CASA_CDL eval:check_rbl('CDL','cdl.anti-spam.org.cn.')
describe RCVD_IN_CASA_CDL Relay in CASA CDL, http://anti-spam.org.cn/services/rbl.
php#cdl
tflags   RCVD_IN_CASA_CDL net
score    RCVD_IN_CASA_CDL 3.0
```

CBL+（返回值是127.0.8.6）：

header RCVD_IN_CASA_CBLPLUS eval:check_rbl('CBLPLUS','cblplus.anti-spam.org.cn.')
describe RCVD_IN_CASA_CBLPLUS Relay in CASA CBL+, http://anti-spam.org.cn/services/rbl.php#cblplus
tflags RCVD_IN_CASA_CBLPLUS net
score RCVD_IN_CASA_CBLPLUS 2.0

虽然CBL+是CBL和CDL的综合，但是CBL+不是一个多重列表，其返回值是固定的一个127.0.8.6。

CBL-（返回值是127.0.8.5）：

header RCVD_IN_CASA_CBLLESS eval:check_rbl('CBLLESS','cblless.anti-spam.org.cn.')
describe RCVD_IN_CASA_CBLLESS Relay in CASA CBL-, http://anti-spam.org.cn/services/rbl.php#cblless
tflags RCVD_IN_CASA_CBLLESS net
score RCVD_IN_CASA_CBLLESS 3.0

CML（返回值是127.0.8.1）：

header RCVD_IN_CASA_CML eval:check_rbl('CML','cml.anti-spam.org.cn.')
describe RCVD_IN_CASA_CML Relay in CASA CML (whitelist), http://anti-spam.org.cn/services/cml.php
tflags RCVD_IN_CASA_CML net nice
score RCVD_IN_CASA_CML -1.0

这是一个白名单，评分是负值，可以降低计分。

body 测试规则名 /模式/修饰符
定义一个信体模式测试。模式是一个Perl的正则表达式。

“ body ”这里指的是邮件信体里面的普通文本；任何非文本的MIME部分都会忽略，如果需要的话， Quoted-Printable 或 Base 64 编码的文本都会被解码。邮件的主题信头也作为了邮件信体的第一个段落处理。在模式匹配前，所有的HTML标记和换行都会被忽略。

body 测试规则名 eval:评估函数([参数])
定义一个邮件信体的评估测试，参见上面。

uri 测试规则名 /模式/修饰符
定义一个uri的模式测试。模式是一个Perl的正则表达式。

“ uri ”这里指的是邮件信体中所有的URI，测试会对每一个URI进行测试，如果发现了匹配，增加其对应的评分。用这个测试来替代使用“ body ”来测试信体中的URI，它会更精确的匹配在URL的两端，同时也速度更快。

rawbody 测试规则名 /模式/修饰符
定义一个原始信体模式测试。模式是一个Perl的正则表达式。

“ raw body ”这里指的是邮件信体里面所有的文本。Quoted-Printable 或 Base 64 编码的文本都会被解码，但是HTML代码和换行仍旧保留。模式是逐行进行匹配的。

rawbody 测试规则名 eval:评估函数([参数])
定义一个原始邮件信体的评估测试，参见上面。

full 测试规则名 /模式/修饰符
定义一个整个邮件的模式测试。模式是一个Perl的正则表达式。

整个邮件包括了完整的信头和信头，其中包括MIME编码的数据，如图像、其它附件、MIME边界等等。

full 测试规则名 eval:评估函数([参数])
定义一个整个邮件的评估测试，参见上面。

meta 测试规则名 逻辑表达式
定义一个逻辑表达式（元规则）来测试其他的测试是否命中或未命中。例如：

meta META1 TEST1 && !(TEST2 || TEST3)

注意，英语的操作符（“ and ”、“ or ”）会被作为测试规则名处理，另外，并不支持异或操作。

meta 测试规则名 逻辑运算表达式
还能够定义一个逻辑运算表达式（元规则）来计算其他的测试结果的运算结果，命中的值是“ 1 ”，未命中的值是“ 0 ”。例如：

meta META2 (3 * TEST1 - 2 * TEST2) > 0

注意，不能使用Perl内建的运算符和函数，如abs()等，它们会被作为测试规则名处理。

如果你要定义一个元规则，但是不希望在测试每个子规则时将其评分计算到总的评分上，只在整个元规则匹配时才将元规则的评分计算到总的评分上时，可以给子规则名前加上“ __ ”（两个下划线），SpamAssassin 不会计入这些子规则的评分。

tflags 测试规则名 [{net|nice|learn|userconf|noautolearn}]
用于设置一个测试规则的标志。这些标志用于评分驱动的后台系统。关于这些标志对那些系统的作用的更多信息请参见 bayes_auto_learn 和 use_auto_whitelist 。有下列标志：
net
该测试是一个网络测试，在大量测试的系统或使用 -L 参数时，他们不会被运行，所以它的评分不会被计算进总的评分。

nice
该测试被用于补偿误判的邮件评分，它应该被指定为负值。

userconf
该测试在使用前需要用户配置（如 language- 类的测试）。

learn
该测试使用前要求经过学习。

noautolearn
该测试的评分不会被学习系统所学习。

priority 测试规则名 n （默认值：0）
指定一个测试的优先级。除了DNS和元测试外的所有测试都按照优先级的顺序进行测试。默认值是0。

管理员设置

这些设置与上面的设置不同，它们甚至比上面的“ 特权设置 ”还要“ 更特权 ”。无论 allow_user_rules 是否设置，它们绝不会用于用户的 user_prefs 配置中。

test 测试规则名 (ok|fail) 一些用于测试的字符串
定义一个回归测试字符串。你能给每个测试规则定义一个以上的回归测试字符串。直接定义一个测试规则匹配时候出现的字符串。

这些测试仅仅在测试环境中使用，它们不会影响到 SpamAssassin 的正常使用。

razor_config 配置文件名
指定存储 Razor 配置的配置文件名。当前它留待让 Razor 自己决定。

pyzor_path 路径
明确指定 Pyzor 的客户端的路径而不是让 SpamAssassin 在当前的 PATH 搜索路径中查找。注意，如果 Perl 解释器的“ 污染模式 ”打开的话，你需要指定这个路径，因为 PATH 会被清除掉。

dcc_home 路径
明确指定 dcc 的主目录。如果没有指定 dcc_path ，SpamAssassin 首先在 dcc_home/bin 中查找 dcc 客户端，如果没有找到，再在当前的 PATH 搜索路径中查找。 如果在 dcc_home 中找到了 dccifd 的UNIX套接字，那么使用它来替代 dccproc 。

dcc_dccifd_path 路径
明确指定 dccifd 的UNIX套接字的路径。 如果没有明确指定，默认会在 dcc_home 查找。如果找到了 dccifd 的UNIX套接字，那么使用它来替代 dccproc 。

dcc_path 路径
明确指定 dccproc 客户端的路径而不是让 SpamAssassin 在当前的 PATH 搜索路径中查找。注意，如果 Perl 解释器的“ 污染模式 ”打开的话，你需要指定这个路径，因为 PATH 会被清除掉。

dcc_options 选项（默认值： -R）
指定dccproc(8)的附加选项。注意由于安全原因，只能使用英文字母和“ － ”（[A-Z -]）。

默认值是 -R。

use_auto_whitelist (0 | 1) (默认值: 1)
设置是否使用自动白名单。自动白名单跟踪每个发信人的历史平均评分，然后将每个新的邮件的评分做向历史平均评分的逼近。这样根据以前的发送邮件的情况，能够提高或降低新的邮件的评分。

关于自动白名单的更多细节，请参阅 README 文件中的 Automatic Whitelist System 一节。自动白名单并不能替代静态白名单的作用。

注意，当觉得最终的评分时，下列情况的测试规则会被忽略：

- 测试规则的 tflags 设置为“ noautolearn ”

auto_whitelist_factory 模块名 (默认值: Mail::SpamAssassin::DBBasedAddrList)
另外的可替代的白名单模块。

auto_whitelist_path 路径 (默认值: ~/.spamassassin/auto-whitelist)
指定自动白名单的存放路径。默认情况下，每个用户都在自己的 ~/.spamassassin 目录里面有一个自动白名单文件 auto-whitelist，权限模式为0700。如果在整个站点应用 SpamAssassin，你应该保证该文件能被所有用户访问。

bayes_path 路径 (默认值: ~/.spamassassin/bayes)
指定贝叶斯的概率数据库的路径。使用这个路径，加上“ _toks ”、“ _seen ”等后缀创建几个数据库：默认情况下就是 ~/.spamassassin/bayes_seen 、 ~/.spamassassin/bayes_toks 等。

默认情况下，每个用户都在自己的 ~/.spamassassin 目录里面存放这些数据库，权限模式为0700或0600。如果整个站点应用 SpamAssassin，你可以让所有用户共享同一个数据库，从而降低磁盘的占用。（然而，贝叶斯过滤器在每用户使用自己单独的数据库时更加有效。）

auto_whitelist_file_mode (默认值: 0700)
指定自动白名单文件/目录的权限模式。

确保你指定的权限包含“ x ”（执行）权限，因为如果需要创建目录时，它需要执行权限才能正常使用。然而，如果创建的是文件，该文件并不会有任何执行权限（umask 被设置为111）。

bayes_file_mode (默认值: 0700)
指定贝叶斯字串数据库的权限模式。

确保你指定的权限包含“ x ”（执行）权限，因为如果需要创建目录时，它需要执行权限才能正常使用。然而，如果创建的是文件，该文件并不会有任何执行权限（umask 被设置为111）。

bayes_store_module 模块名
如果设置了该选项，该模块用于提供替换默认的贝叶斯存储方式。该模块必须遵循公布的存储规范。（参见 Mail::SpamAssassin::BayesStore ）。

bayes_sql_dsn DBI:数据库类型:数据库名:主机名:端口
该选项用于 BayesStore::SQL 存储方式。

这个选项指定的DSN用于连接到基于SQL方式的贝叶斯字串数据库。

bayes_sql_username 用户名
该选项用于 BayesStore::SQL 存储方式。

这个选项指定上面的DSN的连接用户名。

bayes_sql_password 密码
该选项用于 BayesStore::SQL 存储方式。

这个选项指定上面的DSN的连接密码。

user_scores_dsn （LDAP连接 | DBI:数据库类型:数据库名:主机名:端口）
如果你从一个SQL数据库中载入用户自定义的评分，那么在这里定义连接的DSN。例如：DBI:mysql:spamassassin:localhost

如果你从一个LDAP目录服务里面载入用户自定义的评分，那么也在这里定义连接的DSN。你需要写成LADP的URL格式，包含下列部分：LDAP主机、端口、base_DN（目录中条目的名称）、搜索范围（base、one 或 sub）、一个多值的用来控制配置的属性（空格分隔开的键和值，像在文件中一样），最后是一个过滤表达式来过滤出所要的用户名。注意，过滤表达式用在 sprintf 语句中，只有一个用户名参数：“__USERNAME__”，它会替换成实际的用户名。

例子：ldap://localhost:389/dc=koehntopp,dc=de?spamassassinconfig?uid=__USERNAME__

user_scores_sql_username 用户名
连接到上述DSN的用户名。

user_scores_sql_password 密码
连接到上述DSN的密码。

user_scores_sql_custom_query 查询语句
这个选项可以让你定制查询用户的评分和配置的SQL查询语句。查询结果需要按顺序返回配置名、配置值这两个字段才行。此外，你可以在SQL中使用以下的“变量”，它们会在查询时候被替代成当前值。当前支持以下变量：
TABLE
存储用户评分和配置的表名。当前它的值是“userpref”，如果需要，可以在定制查询里使用另外的表名。

USERNAME
当前用户的用户名。

MAILBOX
当前用户的用户名的“@”前的部分。

DOMAIN
当前用户的用户名的“ @ ”后的部分。这个值也许是空的。

查询语句必须是一个连续的行，以便能正常工作。
下面是几个查询语句的例子。注意，有一些为了阅读方便进行了换行，但是在你的配置中应该是一行。
当前的默认查询语句:
SELECT preference, value FROM _TABLE_ WHERE username = _USERNAME_ OR username = '@GLOBAL' ORDER BY username ASC

使用全局和域级别的默认值:
SELECT preference, value FROM _TABLE_ WHERE username = _USERNAME_ OR username = '@GLOBAL' OR username = '@~'||_DOMAIN_ ORDER BY username ASC

用户的配置可以覆盖全局配置:
SELECT preference, value FROM _TABLE_ WHERE username = _USERNAME_ OR username = '@GLOBAL' ORDER BY username DESC

user_awl_dsn DBI:数据库类型:数据库名:主机名:端口
如果你从一个SQL数据库中载入用户的自动白名单，那么在这里定义连接的DSN。例子：DBI:mysql:spamassassin:localhost

user_awl_sql_username 用户名
连接到上述DSN的用户名。

user_awl_sql_password 密码
连接到上述DSN的密码。

user_awl_sql_table 数据库表名
上述DSN中存储用户的自动白名单的数据库表名。

user_scores_ldap_username 用户名
用于连接到LDAP服务器。

例子：cn=master,dc=koehntopp,dc=de

user_scores_ldap_password 密码
用于连接到LDAP服务器。

loadplugin 插件模块名 [模块路径]
装入一个 SpamAssassin 插件模块。模块名是一个Perl的模块名，用于创建模块对象。

模块路径是装入模块的路径，如果指定的是一个相对路径，那是相对于当前配置文件的位置而言的。如果省略了路径参数，会从Perl的查找路径（@INC数组）中找到并载入。

参见Mail::SpamAssassin::Plugin 中的更多细节来写自己的插件。

预处理选项

include 文件名
从一个文件中引入配置。相对路径是相对于当前配置文件或用户的配置文件的位置。

if (conditional perl expression)
用于支持条件选择的配置。在 if 和 endif 之间的配置仅在条件表达式为真值时有效（对于Perl而言，就是该值是定义的且非0）。

由于安全的原因，条件表达式只接受Perl的部分功能，只能进行基本的算术比较。 允许下列输入：

数字、空白、算术运算符和括号
即以下字符：

( ) - + * / _ . , < = > ! ~ 0-9 空白

version
它被替换为当前运行的 SpamAssassin 的版本号。注意，SpamAssassin 内部使用的版本号是 x.yyyzzz 格式，这里 x 是主版本号， y 是辅版本号， z 是修订号。 所以3.0.0是3.000000，3.4.80是3.004080。

plugin(插件名)
如果该插件被载入，那么该函数返回1，否则返回 undef。

如果一个文件直到结束也没有使用 endif 来结束 if 语句，那么会触发一个警告，但是下一个配置文件将会继续进行处理（相当于在上个文件中末尾自动用 endif 结束了）。
例子：

```
if (version > 3.000000)
  header MY_FOO ...
endif

loadplugin MyPlugin plugintest.pm

if plugin (MyPlugin)
  header MY_PLUGIN_FOO  eval:check_for_foo()
  score  MY_PLUGIN_FOO  0.1
endif
```

ifplugin 插件模块名
同 if plugin(插件模块名) 一样。

require_version n.nnnnnn
指定包含该配置的文件需要运行在特定版本的 SpamAssassin 下。如果不同版本（更旧的或

者新的）试图从这个文件中读取配置，它会输出一个警告并忽略该配置文件。

注意，SpamAssassin 内部使用的版本号是 x.yyyzzz 格式，这里 x 是主版本号，y 是辅版本号，z 是修订号。 所以3.0.0是3.000000，3.4.80是3.004080。

version_tag 字符串
这个字符串会被添加到 X-Spam-Status 信头中的 SA版本后。当你修改了你的规则集，特别是在你想公开发布它的时候，你可以使用它。这个字符串的好的用法是使用你的姓或者名字缩写，然后跟上一个数字表示修改次数。

这个字符串是小写的，任何非字母的字符都会被替换为下划线。

范例：

```
version_tag myrules1   # version=2.41-myrules1
```

模板标记

下列标记可以作为变量在几个选项中使用。它们会被替换为相应的值。

一些标记可以使用扩号包括参数。参数是可选的，下面列出了它们的默认值。

YESNOCAPS	根据是否是垃圾邮件返回：“ YES ”/“ NO ”
YESNO	根据是否是垃圾邮件返回：“ Yes ”/“ No ”
SCORE(PAD)	邮件的评分。如果指定了PAD参数，且是空格或数字0时，评分会用空格或数字0进行填充（默认情况下是不填充）。例如 _SCORE(0)_ 将2.4填充成02.4，而 _SCORE(00)_ 将气填充成002.4。12.3则会分别填充成12.3和012.3
REQD	垃圾邮件评分标准线（即require值）
VERSION	版本号（如：3.0.0或3.1.0-r26142-foo1）
SUBVERSION	子版本号或代码修订日期（如：2004-01-10）
HOSTNAME	处理邮件的主机的主机名
REMOTEHOSTNAME	发送邮件的主机的主机名，只在 spamd 中可用
REMOTEHOSTADDR	发送邮件的主机的IP地址，只在 spamd 中可用
BAYES	贝叶斯评分
TOKENSUMMARY	所找到的新的、中立的、垃圾邮件的、非垃圾邮件的字串数量
BAYESTC	所找到的新的字串数量
BAYESTCLEARNED	所找到的出现过的字串数量
BAYESTCSPAMMY	所找到的垃圾邮件倾向的字串数量
BAYESTCHAMMY	所找到的非垃圾邮件倾向的字串数量
HAMMYTOKENS(N)	前N个最重要的非垃圾邮件字串（默认是5个）

SPAMMYTOKENS(N)	前N个最重要的垃圾邮件字串（默认是5个）
AWL	自动白名单的修正评分
DATE	检查时间，使用 rfc-2822 格式
STARS(*)	每一分的评分分值使用一个星号代表（可以使用任何字符）（RFC中限制右边最多有50个星号）
RELAYSTRUSTED	使用的可信任中继服务器
RELAYSUNTRUSTED	使用的非信任中继服务器
AUTOLEARN	自动学习状态（“ham”、“no”、“spam”、“disabled”、“failed”或“unavailable”）
TESTS(,)	使用逗号（或其它字符）分隔开的命中的评分测试规则
TESTSSCORES(,)	如上，只是加上了相应的分值（如：AWL=-3.0,...）
DCCB	DCC 的“Brand”
DCCR	DCC 的结果
PYZOR	Pyzor 的结果
RBL	RBL查询的完整的原始结果（使用DNS URI格式）
LANGUAGES	邮件中使用的可能的语言
PREVIEW	内容预览
REPORT	命中的评分测试规则的简要报告（用于信头报告中）
SUMMARY	命中的评分测试规则的标准报告（用于邮件报告中）
CONTACTADDRESS	“report_contact”的值

HAMMYTOKENS 和 SPAMMYTOKENS 标记有一个用于指定特定格式的可选的第二个参数，参见下面的 HAMMYTOKENS/SPAMMYTOKENS 标记格式 部分。

HAMMYTOKENS/SPAMMYTOKENS 标记格式

HAMMYTOKENS 和 SPAMMYTOKENS 有一个用于指定特定格式的可选的第二个参数：_SPAMMYTOKENS(N,FMT)_, _HAMMYTOKENS(N,FMT)_。格式如下：

short

只列出字串。例如，配置文件中加上：

add_header all Spammy _SPAMMYTOKENS(2,short)_

信头中会出现：

X-Spam-Spammy: remove.php, UD:jpg

指出了最高的两个垃圾邮件字串：“remove.php”和“UD:jpg”。（最后一个冒号后面的是字串，冒号前的标识符表示该字串的一些特性，这里 UD 的意思是“该字串看起来像是域名的一部分”）

compact

列出字串的概率、一个简短的无效分类的数量（参见例子）和字串。例如，在配置文件中

加上：

add_header all Spammy _SPAMMYTOKENS(2,compact)_

信头中会出现：

X-Spam-Spammy: 0.989-6--remove.php, 0.988-+--UD:jpg

分别指出了最高的两个垃圾邮件字串的概率是 0.989 和 0.988。第一个字串的无效分类的数量是6，意思是这个字串至少在6封没有被判定为垃圾邮件的邮件中出现过。第二个字串中的+表示无效分类的数量超过了9。

long

列出字串的概率、无效分类的数量、出现在非垃圾邮件中的次数、出现在垃圾邮件中的次数和字串的存在时间。例如，在配置文件中加上：

add_header all Spammy _SPAMMYTOKENS(2,long)_

信头中会出现：

X-Spam-Spammy: 0.989-6--0h-4s--4d--remove.php, 0.988-33--2h-25s--1d--UD:jpg

long 比 compact 提供了更多的信息，第一个字串出现在0个非垃圾邮件中（0 ham）和4个垃圾邮件中（4 spam）,最后出现是在4天前（4 day）；第二个字串出现在两个非垃圾邮件中（2 ham）和25个垃圾邮件中（25 spam），最后出现是在1天前（1 day）。（不像 compact，long 显示超过9个的无效分类数量而不是显示一个+）

本地化

使用 lang xx 开始的行仅在用户使用该语言时有效，允许在评分测试规则的描述和模板中使用语言定义。

参见

Mail::SpamAssassin spamassassin spamd

附录：配置选项索引