

# AdjustRuleScore

## Adjusting Rule Scores

[SpamAssassin](#) is distributed with rules designed to differentiate between spam and ham. Emails are tested to see which rules apply, and the scores of those rules that do apply to the email are added together. If the resulting score is high enough (equal or greater than the Required Hits parameter), the email is declared to be spam.

The primary SA documentation at [http://spamassassin.apache.org/full/3.1.x/dist/doc/Mail\\_SpamAssassin\\_Conf.html#scoring\\_options](http://spamassassin.apache.org/full/3.1.x/dist/doc/Mail_SpamAssassin_Conf.html#scoring_options) defines the `required_hits` parameter, and states, 5.0 is the default setting, and is quite aggressive; it would be suitable for a single-user setup, but if you're an ISP installing [SpamAssassin](#), you should probably set the default to be more conservative, like 8.0 or 10.0.

There are two basic reasons why you might want to change the default scores of the distribution rule set:

1. You've modified the required hits parameter (up or down), and therefore need to modify at least some of the default scores to properly flag your spam.
1. Non-spam has different attributes for different people. Medical organizations and users will need to score medical and drug-related rules lower than the general population, while mortgage brokers will need to lower the scores of mortgage and debt-related rules.

To determine which scores should be modified for your system,

1. Examine spam that did not get properly flagged. Look in the headers, and see what rules hit for this email. Those are the rules that can have their scores increased.
1. Examine non-spam that was wrongly flagged as spam. Look in the headers to find those rules that can have their scores decreased.

It's best if you have a corpus of spam and non-spam which can be searched to determine the frequency with which various rules hit. Assuming you can collect such a corpus, there are several ways to scan that corpus and count rule hits, from using `grep` and `wc` to using the search functions of an email client like The Bat! You can also use the [MassCheck](#) functionality installed with SA.

Given you have identified a rule which might need to have its score changed, scan your corpus to determine the hit frequency for that rule. You can then increase or decrease the score to better characterize your email.

### Example:

Given a conservative system, which has significantly increased the required hits parameter

```
required_hits 9.00
```

and receives a spam email which scores low:

```
X-Spam-Checker-Version: SpamAssassin 2.60 (1.212-2003-09-23-exp) on
server14.arteryserver14.net
X-Spam-Status: No, hits=1.3 required=9.0 tests=BAYES_50,
HTML_FONTCOLOR_UNKNOWN,HTML_FONTCOLOR_UNSAFE,HTML_MESSAGE,
MIME_HTML_ONLY,PLING_PLING autolearn=no version=2.60
X-Spam-Level: *
```

The rules which match are shown in the X-Spam-Status header. You'll find their scores listed in the `rules/50_scores.cf` file within the SA directories.

- score HTML\_FONTCOLOR\_UNKNOWN 0.100 0.100 0.283 0.100
- score HTML\_FONTCOLOR\_UNSAFE 0.100
- score HTML\_MESSAGE 0.160 0.001 0.100 0.100
- score MIME\_HTML\_ONLY 0.666 0.100 0.248 0.320
- score PLING\_PLING 1.047 1.325 1.157 0.650
- score BAYES\_50 0 0 0.001 0.001

BAYES\_50 you don't want to do anything about – Bayes does not (yet) know whether this email is spam or ham, so it gives effectively no score. Feed this email to Bayes as spam, and that will help identify future emails as spam.

The highest scoring rule is PLING\_PLING (testing for !!! in the subject header). Scan your corpus – how many emails match this rule? If it matches 300 spam and 3 ham, then at your site it's a very strong spam indicator, and the score should likely be increased. Do so by adding a line like the following to your `local.cf` or `user_prefs` file:

```
score PLING_PLING 2.1 2.7 2.3 1.3 # Double default scores
```

Simply doubling this one rule will not cause this spam to be correctly flagged, but it's a start. That plus having Bayes learn the spam may be enough. This score change, plus maybe one or two additional rules of your own or from [CustomRulesets](#) can be enough.