The Sysadmin's Daily Grind: FuzzyOCR

1000 MASTERPIECES

The latest trend is to hide spam in images. The admin's response: an OCR tool that extracts the texts and feeds them to the spam filter. BY CHARLY KÜHNAST

If you run Spamassassin, the FuzzyOCR [1] plugin is a good choice of image evaluation tool. FuzzyOCR isn't hard to install, except for having to fulfill a few dependencies. Make sure your version of Spamassassin is as up-to-date as possible; it should be anyway, of course, but version 3.1.4 is a must. You also need the NetPBM tools, from the Imagemagick *convert* binary, Giflib, two Perl modules, and *gocr* for optical character recognition. That may sound like a tall order, but most distributions actually have these tools on board, and the following command line will pick up the Perl modules for you:

```
cpan -i Digest::MD5
String::Approx
```

You are just a couple of steps away from image evaluation: you need *FuzzyOcr.cf*, and *FuzzyOcr.pm* in the Spamassassin directory, which is */etc/mail/spamassassin* on my machine. FuzzyOCR gives you a sample dictionary, *FuzzyOcr.words*, which contains the terms that you want FuzzyOCR to search the images for. You can modify the list to suit your own needs. Again, you need to copy this file to your Spamassassin directory.

The next step is to define the path for the logfile, and dictionary file in *FuzzyOcr.cf*. After doing so, FuzzyOCR should be ready to take up the fight, as Spamassassin will discover the module in its startup path, and automatically integrate the module.

Picture This

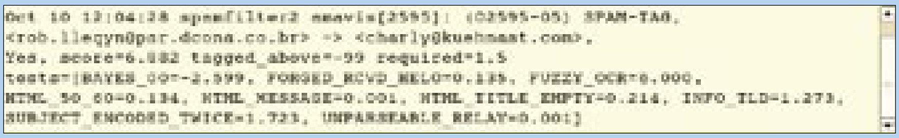
Although FuzzyOCR comes with a few sample spam files for test purposes, it is much more fun to try it out on your own junk. The Web is a weird and wonderful place: spam can occur in any common image format, and the MIME types are often declared incorrectly to cause even more confusion – GIFs prentending to be JPEGs, for example. FuzzyOCR reacts to obfuscation tactics by assigning additional minus points.

Spammers often resort to animated GIFs, especially ones that display junk pixels before revealing the spammer's message. It seems that many spammers rely on OCR engines only analyzing the first phase of the animation, but luckily, FuzzyOCR keeps going.

Crime and Punishment

Time to tweak *FuzzyOcr.cf*. If you have a Spamassassin version prior to 3.1.4, you will need to set an entry for *focr_pre314 = 1* here. Setting up the scores that FuzzyOCR assigns when it finds something suspicious is more important. The program is fairly draconian by default.

For example, a message with an attached image that matches two dictionary entries is given a spam score of four, and one and a half points are added if the MIME type is declared incorrectly. Two and half extra points are added for corrupt images, and five, if the error is not correctable. The points are added to give a grand total as shown in Figure 1.



```
Oct 10 12:04:28 spamfilter2 amavis[2555]: (02555-05) SPAM-TAG,
<rob.llegyn@par.dcona.co.br> -> <charly@kuehnast.com>,
Yes, score=6.882 tagged_above=-99 required=1.5
tests=[BAYES_00=-2.599, FORGED_RCVD_HELO=0.135, FUZZY_OCR=6.000,
HTML_50_60=0.134, HTML_MESSAGE=0.001, HTML_TITLE_EMPTY=0.214, INFO_TLD=1.273,
SUBJECT_ENCODED_TWICE=1.723, UNPARSEABLE_RELAY=0.001]
```

Figure 1: FuzzyOCR detects text in image files and assigns penalty scores if it discovers undesirable words.

Strict settings increase the danger of false positives: don't forget that Spamassassin will probably notice a few more issues with a spam message, and this can lead to incredibly high scores. My recommendation is to reduce the scores to about half their default values in *FuzzyOcr.cf*.

So what next? Just sit back, and wait to see what the spammers dream up next. ■

INFO

[1] FuzzyOCR: *http://users.own-hero.net/~decoder/fuzzyocr/*

THE AUTHOR

Charly Kühnast is a Unix System Manager at the data center in Moers, near Germany's famous River Rhine. His tasks include ensuring firewall security and availability and taking care of the DMZ (demilitarized zone).