SpamAssassin

Instructor: Vuong Thi Nhung

FIT - HANU

2/10/2015

1. Regular expression

- Abbreviated as regex, regexp
- Often called a pattern, is an expression that describes a set of strings.
- Are used to give a concise description of a set without having to list all elements
- Are written in a formal language that can be interpreted by a regular expression processor, a program that either serves as a parser generator or examines text and identifies parts that match the provided specification.

Regex operation

- Alternation: A vertical bar separates alternatives
  - gray|grey
    - can match "*gray*" or "*grey*".
- Grouping: Parentheses are used to define the scope and precedence of the operators
  - gr(a|e)y
    - describe the set of "*gray*" and "*grey*"

Quantification

- A quantifier after a token (such as a character) or group specifies how often that preceding element is allowed to occur.
  - ?: zero or one of the preceding element
    - colou?r matches both "*color*" and "*colour*".
  - *: zero or more of the preceding element
    - ab*c matches "*ac*", "*abc*", "*abbc*", "*abbbc*", and so on.
  - +: one or more of the preceding element
    - ab+c matches "*abc*", "*abbc*", "*abbbc*", and so on, but not "*ac*".

Metacharacters

. Matches any single character

[ ] A bracket expression. Matches a single character that is contained within the brackets.

[^] Matches a single character that is not contained within the brackets.

^ Matches the starting position within the string.

$ Matches the ending position of the string or the position just before a string-ending newline.

SA Regex example

- /test/
  - a simple case-sensitive for the string "test"
- /\btest\b/
  - a \b can be used to indicate where a word-break must exist for a match
- /\btest\b/i
  - case-insensitive by adding an i to the end
- /www\.example\.com\/OrderPic\//
  - This rule will look for web links to www.example.com/OrderPic/

- Study more about regex in the Internet
  - http://en.wikipedia.org/wiki/Regular_expression

2. Spam Assassin concepts

- Lots of rules to determine if a mail is spam or not
  - "Fuzzy logic": rules are assigned scores, based on our confidence in their accuracy
  - These are combined to produce an overall score for each message
  - If over a user-defined threshold, the mail is judged as spam
- No one rule, alone, can mark a mail as spam

Spam Assassin concepts

- Combines many systems for a "broad-spectrum" approach:
  - Detect forged headers
  - Spam-tool signatures in headers
  - Text keyword scanner in the message body
  - DNS blacklists
  - Razor, DCC (Distributed Checksum Clearinghouse), Pyzor

SA Rules

- Most SA tests consists of the same basic components:
  - A test name, consisting of up to 22 uppercase letters, numbers, or underscores. Names that begin T_ refer to rules in testing.
  - A more verbose description of the test, which is used in the reports generated by SpamAssassin. Typically, descriptions are up to 50 characters long.
  - An indication of where to look. Tests can be applied to the message headers only, the message body only, uniform resource identifiers (URIs) in the message body, or the complete message. When testing the message body, the body can be analyzed in its raw state, after MIME-decoding the text, or after MIME-decoding, stripping of HTML, and removal of all line breaks.
  - A description of what to look for. Tests can specify a header to check for existence, a Perl regular expression pattern to match, a DNS-based blacklist to query, or a SpamAssassin function to evaluate.

SA Rules (cont)

- Optional test flags that control the conditions under which the test is applied or other exceptional features.
- A score or scores for the test. Tests can have a single score that is always used, or they can have separate scores for messages that test positive under each of four conditions:
  - When the Bayesian classifier and network tests are not in use
  - When the Bayesian classifier is not in use, but network tests are
  - When the Bayesian classifier is in use, but network tests are not
  - When the Bayesian classifier and network tests are both in use

SA Rule example

- header FROM_STARTS_WITH_NUMS From =~ /^\d\d/
- describe FROM_STARTS_WITH_NUMS From: starts with nums
- score FROM_STARTS_WITH_NUMS 0.390 1.574 1.044 0.579

- Meaning:
  - The “header” directive defines it as a test that will be applied to the message headers and gives the test name (FROM_STARTS_WITH_NUMS)
  - and the test itself, a match of the *From* header against the regular expression /^\d\d/. That regular expression denotes a string that begins with two digits.

- The “describe” directive provides a human-readable description of the test that SpamAssassin will insert in reports when the test matches.
- The score directive determines how many points SpamAssassin will add to the spam score of a message if the test matches.

More example

Testing for a *To*, *From*, or *Cc* header that mentions *friend@public.com* (this test is distributed disabled):

```
header FRIEND_PUBLIC        ALL =~ /^(?:to|cc|from):.*friend\@public\.com/im

describe FRIEND_PUBLIC      sent from or to friend@public.com

score FRIEND_PUBLIC         0
```

Testing for the existence of the *X-PMFLAGS* header:

```
header X_PMFLAGS_PRESENT        exists:X-PMFLAGS

describe X_PMFLAGS_PRESENT      Message has X-PMFLAGS header

score X_PMFLAGS_PRESENT         2.900 2.800 2.800 2.700
```

Testing for long lines of hexadecimal code in the message body:

```
body LARGE_HEX                  /[0-9a-fA-F]{70,}/

describe LARGE_HEX              Contains a large block of hexadecimal code

score LARGE_HEX                 0.633 1.595 1.193 1.160
```

Testing for a *Subject* header in all capital letters, by evaluating a SpamAssassin function:

```
header SUBJ_ALL_CAPS            eval:subject_is_all_caps( )

describe SUBJ_ALL_CAPS          Subject is all capitals

score SUBJ_ALL_CAPS             0.550 0.567 0 0
```

Testing for a message that includes HTML to open a new window with JavaScript (disabled by default):

```
body HTML_WIN_OPEN              eval:html_test('window_open')

describe HTML_WIN_OPEN          Javascript to open a new window

score HTML_WIN_OPEN             0
```

Testing for an HTTP (Hypertext Transfer Protocol) URI anywhere in the message that uses a numeric IP address

```
uri NUMERIC_HTTP_ADDR           /^https?\:\/\/\d{7,}/is

describe NUMERIC_HTTP_ADDR      Uses a numeric IP address in URL

score NUMERIC_HTTP_ADDR         2.899 2.800 2.696 0.989
```

Headers added to spam by SpamAssassin

```
Subject: *****SPAM***** Live your dream life!!                    MPNWSTU
X-Spam-Status: Yes, hits=12.9 required=5.0 tests=CLICK_BELOW,
        FORGED_MUA_EUDORA,FROM_ENDS_IN_NUMS,MISSING_OUTLOOK_NAME,
        MSGID_OUTLOOK_INVALID,MSGID_SPAM_ZEROES,NORMAL_HTTP_TO_IP,
        SUBJ_HAS_SPACES,SUBJ_HAS_UNIQ_ID autolearn=no version=2.60
X-Spam-Flag: YES
X-Spam-Checker-Version: SpamAssassin 2.60 (1.212-2003-09-23-exp)
X-Spam-Level: **********
```

Rules directory

10_misc.cf
20_body_tests.cf
20_head_tests.cf
20_uri_tests.cf
25_body_tests_es.cf
25_body_tests_pl.cf
25_head_tests_es.cf
25_head_tests_pl.cf
30_text_de.cf
30_text_es.cf
30_text_fr.cf
30_text_pl.cf
40_spam_phrases.cf
50_scores.cf
60_whitelist.cf
local.cf

10_misc.cf

- The *10_misc.cf* file
  - defines templates for the spam report that SpamAssassin attaches to spam messages,
  - definitions of headers that SpamAssassin adds to messages,
  - and default settings for the most common configuration options.

- *20_body_tests.cf:* This file defines most tests against message bodies, spam clearinghouses, message languages, and message locales.
- *20_head_tests.cf* : This file contains most of the tests that SpamAssassin performs against message headers. This includes tests for blacklisted and whitelisted addresses in the *From* and *To* headers

- *20_uri_tests.cf* : This file contains most of the tests that SpamAssassin performs against URIs that appear in messages.
- *25_head_tests_es.cf, 25_body_tests_es.cf, 25_head_tests_pl.cf, 25_body_tests_pl.cf:* These files contain header and body tests for Spanish (es) and Polish (pl) messages.

- *30_text_*.cf (de,es,fr,it,pl,sk):* These files don't define any new tests but provide translations of test descriptions and report templates into different languages, such as German (de), Spanish (es), French (fr), Italian (it), Polish (pl), and Slovak (sk). SpamAssassin 3.0 includes only German and French tests at the time of this writing.

- *50_scores.cf* : This file defines the scores associated with all of the tests defined in the other files. The scores are separated into a single file because they are generated by an algorithm that applies each test to a large corpus of spam and non-spam messages and adjusts the scores to minimize false positives and false negatives.
- *60_whitelist.cf* : The rules in this file set up default whitelists for several large well-known addresses and companies, such as Amazon.com.

4. Experiment

- Do it at home and write the report
- Scenario:
  - You have to write your own rule to detect a message with the subject “This is spam mail” as a spam.
  - SA will mark that message as a spam and move it to Junk folder