SpamAssassin

Réunion RÉSIST
Université des Sciences Sociales
Toulouse - 25 novembre 2002

Denis Ducamp
Hervé Schauer Consultants - http://www.hsc.fr/

Plan de la présentation

- Le spam
- SpamAssassin
  - procmail
  - spamc + spamd
  - Intégration SMTP : postfix et autres
  - Relais POP3 / IMAP
  - Auto White List (AWL)
  - BL DNS : razor et autres
  - Performances
  - Problèmes résiduels
- Autres techniques
  - SpamTraps
  - Méthode "Bayes"
  - Listes blanches semi-automatiques : ASK

HSC

- Cabinet de consultants en sécurité Unix, Windows, TCP/IP et Internet depuis 1989
  - 13 consultants
- Expérience de la sécurité Unix depuis 1987
  - Expérience de la sécurité Internet depuis 1991
  - Expérience de la sécurité Windows depuis 1997
- Veille en vulnérabilités
  - vendue depuis Juin 1997
- Veille technologique et stratégique de l'actualité en sécurité
  - vendue depuis Janvier 2000
- 6 à 8 conférences internationales par an
  - Blackhat, CanSecWest, Defcon, Eurosec, IETF, ISSE, RSA, SANS, Usenix...
- Types de prestations
  - Études
  - Installations
  - Investigations et enquêtes après incident
  - Analyses, audits et tests d'intrusion
  - Formations & Tutoriels

HSC et le logiciel libre

- Contribution à Sendmail (01/1992)
- Rediffusion de la conférence des BBS LINUX.FR en newsgroup resif.info.linux (01/1993)
- Création de la hiérarchie de news fr. et de fr.comp.os.linux (04/1993)
- Dossier Unix sur PC de Tribunix (07/1993)
- Installation en clientèle du premier 386BSD (1994)
- Installation en clientèle du premier Linux (01/1995)

- Présentations et formations en français
  - Formation Sendmail (1992)
  - Formation Bind (1992)
  - Présentations de PGP (1994), ssh, SSL, Apache avec certificats
  - Sensibilisation à l'intérêt du logiciel libre en sécurité (1997)
  - Tutoriel sécurité Linux dans le cadre de Linux-Expo (1999)
  - Formation Postfix (1999), (support de cours en ligne)
  - Formation Sécurité Linux (1999), (support de cours en ligne)
- Publication de logiciels libres
  - Babelweb, Filterrules, Nstreams, XML-logs, PktFilter, etc.

Le spam

- Un spam est un message électronique non sollicité
  - un exemple est l'UCE : Unsolicited Commercial E-mail
- Son nom vient d'une marque américaine de corned beef
- Par opposition un message licite peut être appelé "ham"
- Certains ISP déclarent avoir dépassé un taux de 25% de spams
  - certains arrivant déjà à 50%
- Le principal problème est le temps perdu par les destinataires
  - l'utilisation des ressources sur les serveurs peut être importante
  - notamment sur les serveurs SMTP, relais ouverts sur Internet

La lutte anti-spam

- La lutte anti-spam demande des moyens humains et systèmes
  - elle réclame des ressources systèmes
    - comme tout filtre de contenu
  - devant être mise en place et administrée
  - et enseignée aux utilisateurs
- Différentes méthodes complémentaires se font concurrence :
  - filtrage de contenu
  - listes noires DNS (BL DNS)
  - catalogues de spams
  - classifications statistiques
  - listes blanches semi-automatiques
- Côté utilisateurs les principaux problèmes sont
  - les faux négatifs et surtout les faux positifs
  - l'éducation des utilisateurs

SpamAssassin

- http://spamassassin.org - écrit en perl - sous Artistic License
  - actuellement en version 2.43
- Destiné aux systèmes Unix
  - deux versions Windows commerciales existent :
    - plug-in Exchange et Outlook : http://www.deersoft.com
    - plug-in Eudora : http://www.spamnix.com
  - sinon : USING SpamAssassin WITH WIN32
    - http://www.openhandhome.com/howtosa.html
- Filtrage de contenu :
  - les entêtes SMTP
  - le corps des messages
- Peut gérer par utilisateur :
  - des configurations : règles et scores
  - des listes blanches automatiques
  - des statistiques "bayésiennes" (version 2.50)
- Peut également utiliser :
  - des listes noires DNS (BL DNS)
  - des catalogues de spams

SpamAssassin

- SpamAssassin associe à chaque test un certain nombre de points
  - quand le total des points dépasse une certaine limite
    - le message est considéré comme un spam
  - par défaut cette limite est à 5.

- Le nombre de points associés à chaque test est calculé
  - afin de minimiser les nombres de faux positifs et de faux négatifs
  - à partir de 2 corpus (spams/hams) et d'un algorithme génétique
    - les règles marquée "nice" ne peuvent avoir un score positif
  - un faux positif coûte beaucoup plus cher qu'un faux négatif
  - ces points sont enregistrés dans le fichier /usr/share/spamassassin/50_scores.cf

As Craig says, it only matters that we catch spam from the dumb 95% of spammers out there. If we miss the smart 5% that's OK, and we'll get them next time around.

Tests locaux

- SpamAssassin effectue deux types de tests locaux
  - soit pour détecter un spam
  - soit pour détecter un message légitime
- Les entêtes et le corps du message sont utilisés
- Entête :
  - anti_ratware : USER_AGENT*
  - ratware : RATWARE*
  - head_tests : NO_REAL_NAME, FROM_ENDS_IN_NUMS, TO_EMPTY, MSGID*, *DATE*, RCVD_IN*, MIME_BOUND*
- Le corps est décodé
  - décodage mime et html
  - HTML_*, MIME_*
  - *REMOVE*, *1618 (Senate Bill 1618), *FREE, DEAR*, EXCUSE*, *VIAGRA, LINES_OF_YELLING*, CLICK*, PENIS*, MORTGAGE*, *CREDIT*, NIGERIAN*
  - compensate : HABEAS_SWE, APPROVED_BY, BUGZILLA_BUG, MAJORDOMO...
  - uri_tests : *HTTP*, MAILTO_TO*, JAVASCRIPT_URI, WEIRD_PORT

Configuration

- L'administrateur peut personnaliser SpamAssassin au niveau du système
  - en modifiant le fichier /etc/mail/spamassassin/local.cf
- L'utilisateur peut personnaliser SpamAssassin grâce à son fichier ~/.spamassassin/user_prefs
- Toutes les options sont décrites dans le manuel Mail::SpamAssassin::Conf
- Les paramètres les plus utilisés sont :
  - #required_hits 8.00 # default: 5
    - pour changer la sensibilité de détection des spams
  - defang_mime 0 # default: 1
    - pour ne plus modifier le Content-type: en "text/plain" pour les spams
  - report_header 1 # default: 0
    - pour insérer le rapport dans les entêtes
  - #use_terse_report 1 # default: 0
    - pour n'avoir qu'un rapport condensé
  - #rewrite_subject 0 # default: 1
    - pour modifier le sujet
  - #subject_tag ~ # default: *****SPAM*****
    - pour changer la chaîne insérée dans le sujet

procmail

- Pour marquer son mail avec SpamAssassin

```
:0fw
* < 256000
| /usr/bin/spamassassin
```

- Au niveau du système, càd dans le fichier /etc/procmailrc
  - ajouter la ligne suivante avant tout filtre : DROPPRIVS=yes
- Pour le filtrer dans différentes boites suivant le score obtenu

```
:0:
* ^X-Spam-Level: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
Mail/spam-almost-certainly

      :0:
* ^X-Spam-Status: Yes
Mail/spam-probably
```


spamc + spamd

- spamd est une version "démonisée" de SpamAssassin
  - permet d'éviter l'initialisation de SpamAssassin à chaque message
  - environ 50% du temps processeur est économisé
- spamc est un mini client, écrit en C, à spamd
- spamc ne passe à spamd que les messages d'une taille inférieure à 250Ko (option -s)
- spamc communique avec spamd au travers d'une socket IPv4
  - spamd peut être sur un autre serveur (options -d et -p)
- spamd traite les messages sous l'identité de spamc
  - s'il est lancé sous root, sans l'option -u et que spamc est local
- spamd peut n'accepter les connexions que de certaines adresses
  - par défaut que 127.0.0.1

Intégration SMTP : postfix

- 2 méthodes - voir FILTER_README de postfix
- "Simple content filtering"
  - utilisation d'un script sur un fichier temporaire
  - voir "Filtering malware and spam with Postfix"
    - http://advosys.ca/papers/postfix-filtering.html
- "Advanced content filtering"
  - utilisation d'un relais SMTP filtrant
  - postfix récupère le message sur un autre smtpd (port TCP)
  - méthode bien plus stable : moins de risque de perte de message
- 2 relais SMTP filtrants sont disponibles :
  - spamproxyd : http://spamassassin.org/released/spamproxyd-20020226.tar.gz
  - spampd : http://www.worlddesign.com/index.cfm/rd/mta/spampd.htm

Autres serveurs SMTP

- Pas plus d'information que ces quelques URL :
- sendmail + milter
  - mimedefang : http://www.roaringpenguin.com/mimedefang/
  - spamass-milt : http://savannah.nongnu.org/projects/spamass-milt/
- qmail
  - qmail-scanner : http://qmail-scanner.sourceforge.net/
- exim
  - http://bogmog.sourceforge.net/document_show.php3?doc_id=28

Relais POP3 et IMAP

- spamproxy
  - en C pour Unix
  - http://www.rominet.net/spamproxy/
- SAPP SpamAssassin POP3 Proxy
  - en perl pour Windows
  - http://www.nickdafish.com/SAPP.htm
- Pop3proxy - a SpamAssassin-enabled POP3 proxy
  - en perl pour Windows.
  - http://mcd.perlmonk.org/pop3proxy/
- IMAPssassin - a SpamAssassin enabled IMAP client that allows a user to filter any IMAP mailbox
  - en perl pour Unix
  - http://sourceforge.net/projects/imapassassin

Auto White List (AWL)

- Système de suivie de score pour les expéditeurs
  - la base enregistre le score moyen de chaque expéditeur
  - une note est attribuée pour corriger le score du message traité
    - pour le rapprocher du score moyen de l'expéditeur
- Permet qu'un spammeur régulier soit toujours repéré ainsi
  - mais aussi qu'un utilisateur connu puisse vous envoyer des spams
- Ne pas utiliser de base globale
  - car elle prend en compte les préférences des utilisateurs

HSC BL DNS

- La qualité des bases est très variable
  - certaines devront certainement être désactivées
- Désactivation totale :
  - #skip_rbl_checks 1 # default: 0
- Attention : certains services sont commerciaux
  - et désactivés par défaut dans SpamAssassin
  - SpamCop : http://spamcop.net/bl.shtml
  - MAPS : http://mail-abuse.org/
- Autres pointeurs :
  - http://www.ordb.org

HSC razor

- http://razor.sourceforge.net/
- Catalogue distribué et constamment mis à jour de spams en propagation.
- Ce catalogue est utilisé par des clients pour filtrer les spams connus.
  - est mis à jour par des utilisateurs rapportant les spams inconnus.
  - via une chaîne unique d'identification de 20 caractères (SHA-Digest)
- Le protocole v2 permet d'associer un niveau de confiance aux rapporteurs : Truth Evaluation System (TeS)
  - afin de contrer les attaques contre certaines de diffusion listes
- Pour attraper les variantes de spam, d'autres hachés sont utilisés
  - nilsimsa - http://ixazon.dynip.com/~cmeclax/nilsimsa.html
  - des variances dans le message ne génère que peu de variance dans le hach é
- Désactivation :
  - score RAZOR_CHECK 0 # default: 2.640
  - #score RAZOR2_CHECK 0 # default: 3.910

Autres catalogues

- Pas plus d'information que ces quelques URL :
- dcc
  - http://www.rhyolite.com/anti-spam/dcc/
  - Distributed Checksum Clearinghouse
  - désactivation :
    - use_dcc 0 # default: 1
    - score DCC_CHECK 0 # default: 2.660
- pyzor
  - http://pyzor.sourceforge.net/
  - a Python implementation of Razor
  - désactivation :
    - score PYZOR_CHECK 0 # default: 2.500

Performances

- Attention à fetchmail + MTA + procmail + SpamAssassin
  - préférer fetchmail + procmail (mda=) + SpamAssassin
  - ou faire positionner un verrou par procmail lors du filtrage
    - :0fw:lockme.file
- Dans le cas MTA + procmail + SpamAssassin
  - limiter le nombre de délivrances locales simultanées pour un utilisateur
  - postfix (défaut) : local_destination_concurrency_limit = 2
- Le temps de traitement d'un message dépend du carré de sa taille
- Matt Sergeant <msergeant@startechgroup.co.uk>

We do about 10 million a day, but then we have over 400 mail servers. SpamAssassin can seriously overload a box, so be very careful.

I suggest allowing about 0.5 seconds per email, probably more. That means you're going to do about 100k mails per box, maximum (assuming a non-even spread).

Problèmes résiduels

- Apprendre aux utilisateurs à
  - utiliser correctement la messagerie :
    - inutile d'envoyer seulement du html avec couleurs et des fontes variées
    - présenter correctement ses réponses en citant comme il faut
  - ne pas détruire de façon automatique / systématique les messages marqués comme des spams
    - et consulter régulièrement ceux-ci pour en extraire les faux positifs
  - mettre à jour régulièrement les règles de filtrage
    - une version vieille de 6 mois est aussi caduque qu'un antivirus vieux de 6 mois
  - ne pas utiliser fetchmail + exim + SpamAssassin sans précaution
    - => Out Of Memory

Problèmes résiduels

- Apprendre aux administrateurs à
  - à ne pas détruire de façon automatique / systématique les messages marqués comme des spams
  - à ne pas marquer les messages sortants de leurs domaines
- Antivirus
  - SpamAssassin n'est pas un antivirus
    - la lutte virale est un autre problème
    - même s'il est possible d'écrire des règles de filtrage pour contrer certains virus

Les autres techniques

- SpamTraps
- Méthode "Bayes"
  - SpamBayes
    - http://spambayes.sourceforge.net/
  - SpamOracle
    - http://cristal.inria.fr/~xleroy/software.html
  - Bogofilter
    - http://www.bgl.nu/~glouis/bogofilter/
    - http://bogofilter.sourceforge.net/
    - http://www.bgl.nu/~glouis/bogofilter/test6000.html
  - Et de plus en plus d'autres :
    - http://sourceforge.net/projects/bayespam
    - http://spambayes.sourceforge.net/related.html
- Listes blanches semi-automatiques
  - ASK - Active Spam Killer
    - http://www.paganini.net/ask/

SpamTraps

- Adresses bidons
  - ne correspondant à rien
  - ou plus utilisées
- Ces adresses peuvent être diffusées
  - sur des pages web
  - dans des groupes de news
- Les messages reçus sur ces adresses sont des spams
  - utilisés pour constituer des BL et alimenter les catalogues
- Attention :
  - certains virus utilisent des adresses trouvées dans des caches web pour se dupliquer
  - ces adresses peuvent alors se retrouver abonnées à des listes de diffusion !

Méthode Bayes

- Cette méthode est basée sur la fréquence d'apparition des "mots" d'un message dans les messages précédents
  - pour déterminer s'il s'agit d'un spam ou d'un ham
- Ce système une fois mis en place ne doit pas être mis à jour
  - contrairement aux systèmes à base de règles de filtrage
- Ce système nécessite une période d'apprentissage pour savoir à quoi ressemblent les "mots" des hams et des spams d'un utilisateur
  - cette phase doit être effectuée de façon très rigoureuse
  - et continuer sans fin pour prendre en compte les changements de tendances
- Une fois correctement entraîné, le taux de reconnaissance peut être très élevé (99%) pour un utilisateur donné
  - pour un ensemble d'utilisateurs hétérogènes, ce taux sera plus faible (90%) => multiplication des "unsure"

Méthode Bayes

- Beaucoup de systèmes (tous ?) sont basés sur le papier de Paul Graham "A Plan For Spam" : http://www.paulgraham.com/spam.html
  - et devraient être associés à la catégorie "naive bayesian"
- La difficulté de mise en place de tous ces systèmes est l'apprentissage
  - ne peut être automatisé en se basant sur les résultats d'un programme utilisant une autre méthode
    - le résultat obtenu ne serait qu'une mise en oeuvre "bayésienne" de l'autre programme
  - doit être effectué sur un grand nombre de messages
    - reçus récemment par l'utilisateur à protéger
- Les ressources nécessaires (place disque...) peuvent être importantes
  - tous les "mots" appris devant être enregistrés
  - ceci empire si des bi-mots ou tri-mots sont utilisés
    - les expériences actuelles montrant que le gain est négligeable
- Très peu "user friendly" : apprentissage contraignant et nécessaire
  - et capable de mal classer un message après qu'il ait été enseigné

SpamBayes

- http://spambayes.sourceforge.net/ - écrit en python
  - Considéré comme l'état de l'art du filtrage de type Bayes
- A commencé, comme tous les autres, avec le papier de Paul Graham "A Plan For Spam" : http://www.paulgraham.com/spam.html
- Mais incorpore d'autres aspects pris de la page web de Graham Robinson : http://radio.weblogs.com/0101454/stories/2002/09/16/spamDetection.html
- Les messages sont classifiés en 3 groupes : spam / unsure / good
  - Un coût est associé à chacun de ces types de classement :
    - 10$: faux positif
    - 1.0$: faux négatif
    - 0.2$: unsure
  - La recherche se porte sur la détermination de ce qu'est un mot
    - afin d'optimiser le coût global

SpamAssassin 2.50

- Basé sur SpamBayes
- Toujours en développement
  - pas d'expiration des vieilles statistiques
  - pas d'apprentissage automatique
- Deux commandes permettent l'apprentissage :
  - sa-learn-spam
  - sa-learn-nonspam
- et une d'oublier :
  - sa-forget

Autres "Naive Bayesian"

- POPFile Automatic Email Sorting using Naive Bayes
  - open source POP3 proxy that does email classification and sorting using Naive Bayes
  - écrit en perl pour Windows, Unix et Macintosh
  - http://popfile.sourceforge.net/
- BogoFilter - fast Bayesian spam filter
  - Bogofilter is written in C. Supported platforms: Linux, FreeBSD, Solaris, OS X, and HP-UX.
  - http://bogofilter.sourceforge.net/
  - http://www.tuxedo.org/~esr/bogofilter/

Active Spam Killer - ASK

- http://www.paganini.net/ask/
- Quand un message arrive d'un expéditeur inconnu alors
  - une confirmation lui est renvoyée
  - le message reste en attente dans la queue
- Quand la réponse à la confirmation est reçue alors
  - l'expéditeur est rajouté à la liste blanche du destinataire
  - le message est délivré

That’s all folks...

Merci de votre attention.

Vous pouvez poser vos questions...

et faire connaître vos remarques...

puis réveiller discrètement ceux qui dorment ;-)

Bye, bye...