Lutte antispam et surveillance réseau dans une université

Nicolas Grenèche

Centre de Ressources Informatique (CRI) - Université d'Orléans
Projet SDS - LIFO et ENSI de Bourges
nicolas.greneche@univ-orleans.fr

RMLL 2010 : Topic Sécurité

Sommaire

UNIVERSITE D'ORLEANS

Plan

UNIVERSITE D'ORLEANS

SI à l'université d'Orléans

Equipe de recherche SDS (Sécurité et Distribution des Systèmes) au sein du LIFO (Laboratoire d'Informatique Fondamentale d'Orléans).

- Formalisation de propriétés de sécurité ;
- Système de protection mandataire garantissant les propriétés ;
- Système d'exploitation sécurisé pour l'Internaute (en position de gagner le défi sécurité de l'ANR) ;
- Application pour le calcul intensif, les pots de miel, les cartes à puce, les réseaux informatiques ;

UNIVERSITÉ D'ORLÉANS

Organisation de l'université

- 4 UFR, 4 IUT, 1 IUFM, 1 OSU et 1 école d'ingénieur ;
- Des sites distants (Bourges, Chartres, Blois, Tours et Châteauroux / Issoudun ;
- Accès federé vers RENATER par RECIA ...

UNIVERSITÉ D'ORLÉANS

Organisation de l'université

- 4 UFR, 4 IUT, 1 IUFM, 1 OSU et 1 école d'ingénieur ;
- Des sites distants (Bourges, Chartres, Blois, Tours et Châteauroux / Issoudun ;
- Accès federé vers RENATER par RECIA ... Mais pas toujours ! ;
- Un service central chargé de l'infrastructure commune (mail, web etc.) ;
- Des correspondants réseaux dans chaque composantes.

UNIVERSITÉ D'ORLÉANS

Plan

UNIVERSITE D'ORLEANS

Le routage des mails et le DNS 1/3

Problème : à quel serveur doit-on s'adresser pour envoyer un mail destiné à "local@domaine.com" ?

- Le MTA effectue une requête DNS afin de connaître l'enregistrement MX de "domaine.com".

Commande :

```
$ host -t mx gmail.com
gmail.com mail is handled by 5 gmail-smtp-in.l.google.com.
gmail.com mail is handled by 10 alt1.gmail-smtp-in.l.google.com.
gmail.com mail is handled by 10 alt2.gmail-smtp-in.l.google.com.
```

- En cas d'échec de la requête DNS, cela ne veut pas dire qu'il n'y a pas d'enregistrement MX pour le domaine concerné : le client doit réessayer un peu plus tard...
- Si la requête DNS ne renvoie aucun enregistrement MX, alors le MTA doit prendre comme MX l'hôte lui-même.

UNIVERSITE D'ORLEANS

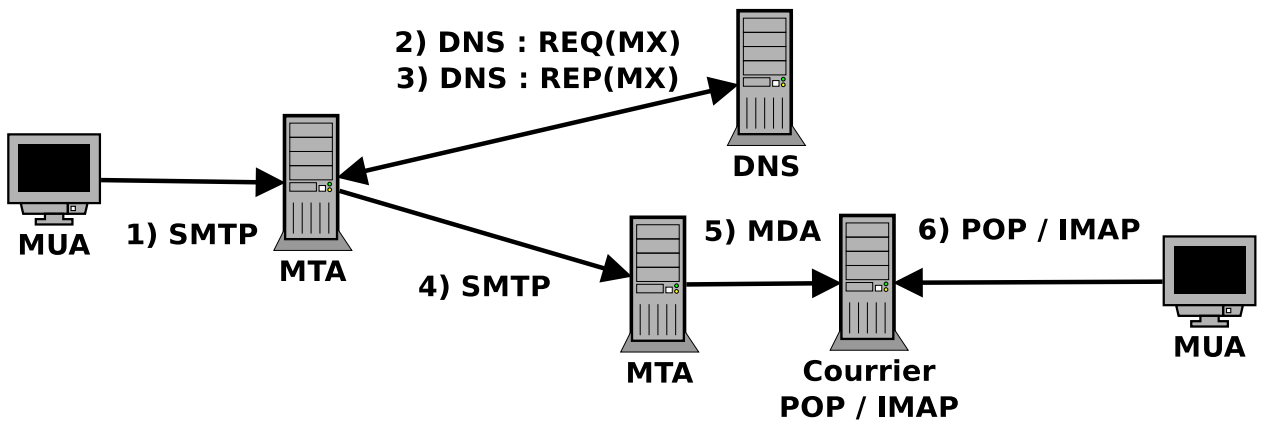
Le routage des mails et le DNS 2/3

- Le serveur prend ensuite la liste des serveurs destinataires par ordre croissant de préférence et se connecte ensuite sur le port TCP/25 pour envoyer le message par SMTP
  - En cas d'échec permanent, le client retourne un message d'erreur à l'expéditeur (bounce message)
  - En cas d'échec temporaire, le client réessaye d'envoyer le message en utilisant les serveurs de mails ayant une priorité inférieure. S'il ne restent plus de serveurs dans la liste, le message est alors mis en attente
- L'utilisation des MX est spécifiée dans la RFC 974

UNIVERSITÉ D'ORLÉANS

Le routage des mails et le DNS 3/3

Problème : à quel serveur doit-on s'adresser pour envoyer un mail destiné à "local@domaine.com" ?



Plan

UNIVERSITE D'ORLEANS

Les DNSBL (DNS BlackList) 1/2

Ces listes d'adresses IP de MTA de spammeurs connus ou supposés sont utilisées par les outils de lutte contre le spam.

- Souci : proposer un moyen de consultation simple ayant une faible latence et pouvant s'adapter à une sollicitation importante ;
- En 1997, Paul Vixie a donc eu l'idée de créer la première blacklist En s'appuyant sur DNS. Ce protocole satisfait tous les prérequis mentionnés ;
- En créant une zone DNS sur un nom de domaine donné (par exemple bl.spamcop.net) et en la peuplant avec les adresses IP des MTA de spammeurs, on obtient le premier mécanisme DNSBL (DNS BlackList).

UNIVERSITÉ D'ORLÉANS

Les DNSBL (DNS BlackList) 2/2

```
1265059581.978307 00:14:22:1b:c2:05 >
00:16:3e:53:ed:23, ethertype IPv4 (0x0800), length
87:  @IP_BLACK.60325 > @IP_DNS.53:  61620+ A?
22.84.62.187.bl.spamcop.net.  (45)

1265059581.999310 00:16:3e:53:ed:23 >
00:14:22:1b:c2:05, ethertype IPv4 (0x0800), length
140:  @IP_DNS.53 > @IP_BLACK.60325:  61620 NXDomain
0/1/0 (98)
```

Efficacité : épure autour de 80% des spams (c'est-à-dire qu'environ 80% des messages à destination de notre MTA viennent de serveur blacklistés).

Le Greylist (1/2)

Basé sur le fait que les serveurs de spams ou machines ayant un virus ne gèrent pas les files d'attentes de messages en ne réessayant pas un renvoi lors d'un échec temporaire (erreur SMTP 4XX), le greylisting consiste à rejeter temporairement (quelques minutes) tout message.

- Implication : beaucoup de messages arrivent en retard (c'est problème potentiel pour des sites envoyant un mail de confirmation d'inscription ayant une faible durée de vie) ;
- Solution : listes blanches (listes de domaines où d'IP bypassant le greylisting).

Le Greylist (2/2)

Pour cela , on forme un triplet (adresse IP de l'expéditeur, adresse email de l'expéditeur, adresse email du destinataire) à partir de tout mail qui arrive.

- Si le triplet est inconnu, on le met dans une base (il devient gris) et on refuse temporairement le mail ;
- Si le triplet revient dans un intervalle de temps défini et paramétrable, on l'accepte et on le blanchit ;
- Si le triplet est blanc (fait partie d'une liste blanche) ou blanchi, on l'accepte ;
- Tout triplet gris est détruit au bout d'un certain délai ;
- Tout triplet blanchi et non réutilisé est détruit au bout d'un certain délai.

UNIVERSITÉ D'ORLÉANS

Bayes

Les filtres Bayésiens se basent sur les probabilités conditionnelles.

- Sachant que le résultat du lancer d'un dé équilibré est pair, quelle est la probabilité que le résultat de ce lancer soit deux ? Intuitivement on trouve 1/3 (1 chance sur 3) ;
- Le filtre tente de déterminer la probabilité qu'un message soit un spam sachant qu'il contient tel ou tel mot clé ;
- La phase d'apprentissage ajuste ces différentes valeurs en fonction de l'environnement.

Un système de ce type seul pose de gros problèmes notamment avec su spam / ham dans une langue autre que le français et rare sur le domaine concerné. Par exemple, si un seul message en portugais est reçu lors de la phase d'apprentissage et qu'il s'avère que c'est un spam alors tous les messages suivants en portugais risquent d'être catégorisés spams. En revanche, couplé à un système d'apprentissage automatique type Spamassassin, ça fonctionne très bien.

Plan

UNIVERSITE D'ORLEANS

SA : Overview

Spamassassin (que nous nommerons SA par la suite) ne détruit pas forcément les messages (c'est juste une question de réglages). Il tente de séparer le bon grain de l'ivraie en appliquant un mécanisme de notation pour chaque email. Cette notation est basée sur les règles situées dans le répertoire pointé par la variable DEF_RULES_DIR du script SA. Chacune des règles possède un score. Si la somme des scores du message dépasse un certain seuil, le message est considéré comme spam. On y trouve des règles de différents types :

- Blacklists ;
- Pattern matching ;
- Bayes ;
- Modules externes (Razor, Pyzor etc.).

UNIVERSITÉ D'ORLÉANS

SA : Blacklists (1/2)

Les blacklists sont built-in dans SA. Elles sont déclarées dans le fichier 20_dnsbl_tests.cf. Par exemple pour spamhaus on retrouve les lignes suivantes :

```
header __RCVD_IN_ZEN eval:check_rbl('zen',
'zen.spamhaus.org.')
describe __RCVD_IN_ZEN Received via a relay in
Spamhaus Zen
tflags __RCVD_IN_ZEN net
reuse __RCVD_IN_ZEN
```

UNIVERSITE D'ORLEANS

SA : Blacklists (2/2)

Ainsi une requête DNS est faite vers le serveur zen.spamhaus.org pour chaque message transitant par SA.

- Installer un cache DNS (dnscache) ;
- On laisse les DNSRBL activées car SA travaille sur les en-têtes Received du message. De cette manière, les MTA intermédiaires sont aussi testés.

Pour désactiver une DNSBL particulière, ajouter les lignes suivantes dans le fichier .spamassassin/user_prefs de l'utilisateur exécutant le script SA :

```
score RCVD_IN_ZEN 0
```

SA : Pattern Matching

Le pattern matching travaille à la fois sur les en-têtes et sur le corps du message. Le langage de définition des règles est assez simple. Il définit un triplet constitué d'un domaine d'application de la règle (body, header, uri, rawbody), d'un identifiant et d'une expression régulière. Le domaine d'application header concerne les en-têtes, par exemple :

```
header __LOCAL_FROM_NEWS From ~= /news@example\.com/i
```

Cette expression va matcher avec tous les messages dont l'en-tête From vaut news@example.com. Body fait la même chose sur le corps du message. Rawbody travaille sur le message avec son pré-traitement par Spamassassin. Principalement, les balises HTML ainsi que les sauts de ligne sont conservés. Enfin, uri matche spécifiquement les URI localisées dans les parties HTML des messages.

UNIVERSITÉ D'ORLÉANS

SA : Bayes

Le filtre Bayésien de SA repose sur l'apprentissage. La commande sa-learn construit les bases de connaissance de spam et de ham. SA va décomposer les messages pour analyser la fréquence des mots. Ainsi, par apprentissage statistique, on obtient la probabilité qu'un message soit un spam sachant qu'il contient tel mot. SA dispose aussi d'un mode auto-apprentissage très efficace.

SA : Intégration

On trouve SA sous deux formes :

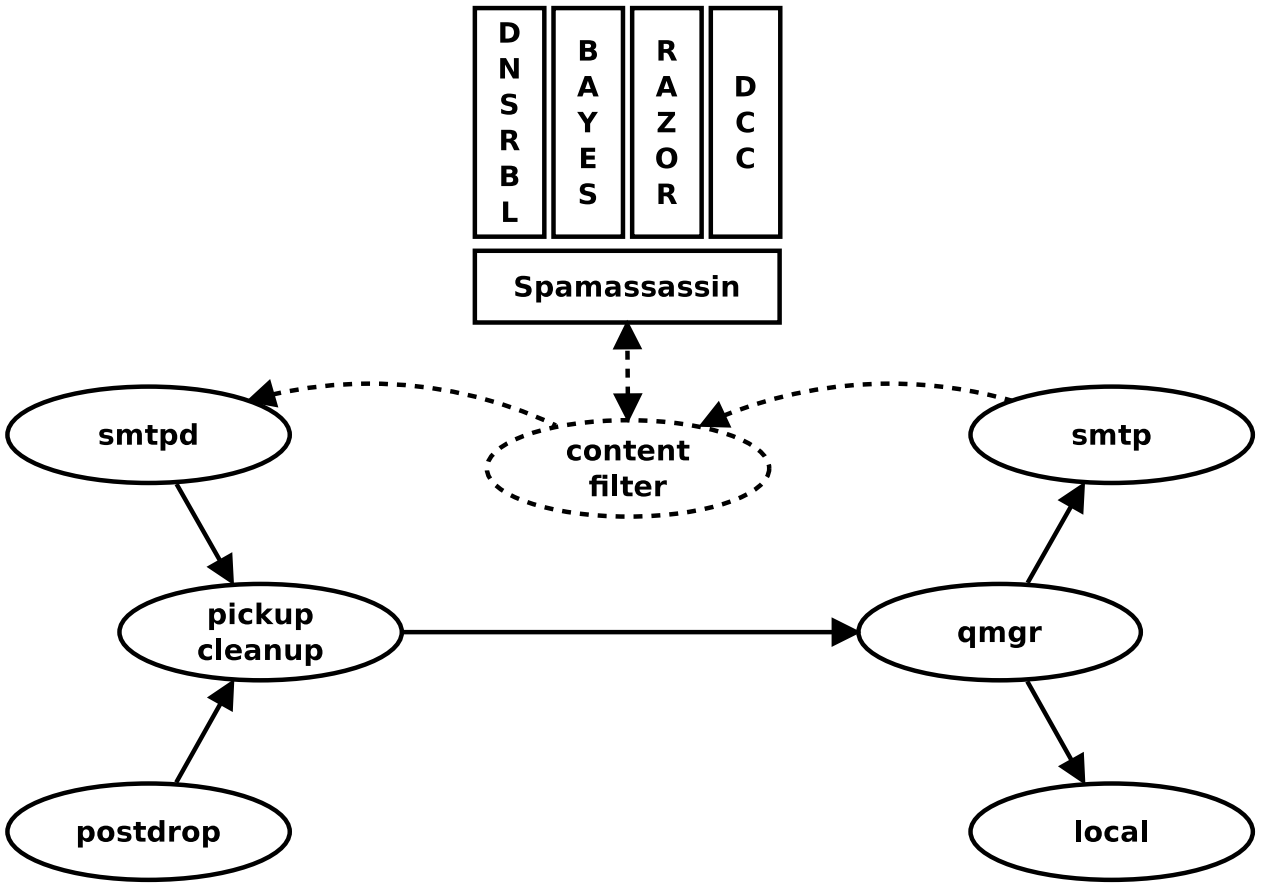
- Script Perl jouant le jeu de règles sur son entrée standard ;
- Application client serveur. Cette implémentation optimisée comprend deux parties : Spamd et Spamc.
  - Spamd charge les règles en mémoire de manière résidente ;
  - Spamc est un client écrit en C qui attend les messages à traiter sur l'entrée standard pour les soumettre au démon Spamd.

On peut choisir de l'intégrer à de multiples niveaux : MTA, MDA et MUA.

UNIVERSITÉ D'ORLÉANS

SA : Intégration

Intégration à un MTA :



Efficacité : 3% des messages résiduels (derrière le blacklist et le greylist). 7% après 6 mois d'apprentissage.

Plan

UNIVERSITE D'ORLEANS

Extrusion ?

L'extrusion de spams résulte de la compromission des machines du réseau interne. Ces machines tentent soit de se connecter directement sur le port TCP/25 de serveurs distants, soit d'utiliser vos MTA émetteurs pour envoyer du spam. Le premier cas est simple à bloquer : interdiction de sortir sur le port TCP/25 au niveau du pare-feu. Pour le deuxième c'est un peu plus compliqué, il faut authentifier la source avant de relayer. On a deux types de population amenées à utiliser un service de relai de messagerie authentifié : les non-interactifs (le serveur qui envoie son logwatch journalier) et les interactifs (l'utilisateur et son MUA).

UNIVERSITE D'ORLEANS

Extrusion : le cas du webmail (1/2)

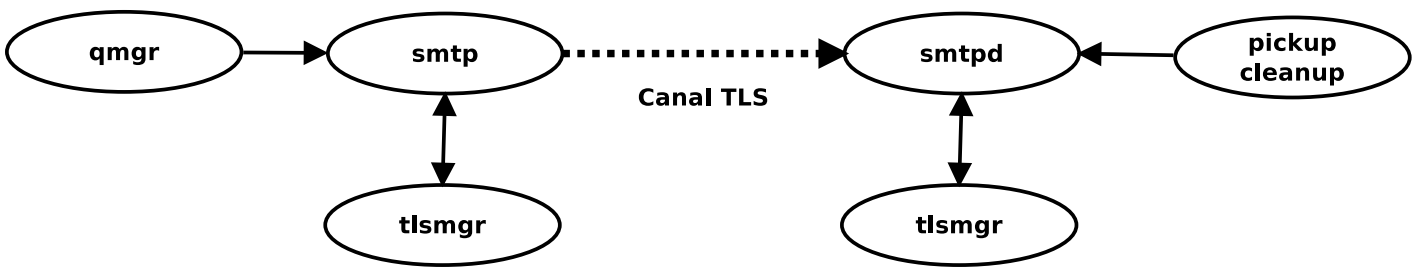
Une campagne de phishing a ciblé l'université pour extorquer les logins / mots de passe des utilisateurs. Fatalement, certains on répondu. Les conséquences :

- Accroissement du spam global en entrée ;
- Affinage du spam (meilleur taux de pénétration des antispams) ;
- Phishing plus ciblés (par exemple champs From en CNRS, destinataires ciblés) ;
- Envoi de spams via des robots implémentant la librairie Horde à d'autres domaines universitaires ;
- DoS sur nos MTA : blacklist (hotmail, gmail and co) et impact sur les ressources (latence et taille des files d'attente).

UNIVERSITÉ D'ORLÉANS

Extrusion : le cas du webmail (2/2)

Un webmail n'est pas fiable pour l'emission de messages. Il est relayé en toute confiance par un MTA. On authentifie un session HTTP et non SMTP. Pour l'envoi sécurisé de courrier, la mise en place d'un serveur SMTPS attaquable par des MUA lourds est nécéssaire. SMTPS = SMTP sur TLS avec une authentification (le plus souvent LDAP ou Kerberos via la SASL).



Extrusion : enseignements

Experience(s) tirée(s) de ces incidents :

- Bien dissocier les circuits d'entrée et de sortie des mails ;
- Proposer un service SMTPS ouvert ;
- Pour le circuit de sortie, multiplier les instances de MTA dans le but de confiner les services d'emission blacklistés.

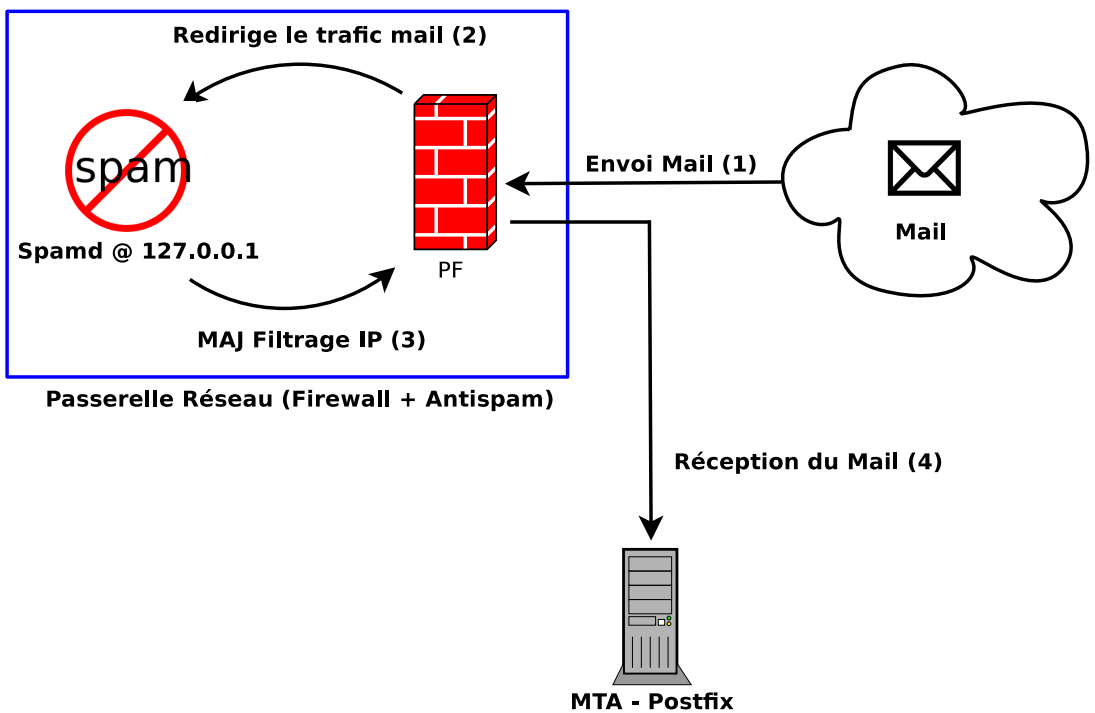
UNIVERSITE D'ORLEANS

Plan

UNIVERSITE D'ORLEANS

OpenBSD Spamd (1/4)

Comme son nom l'indique, OpenBSD Spamd est un système antispam disponible dans le système de base d'OpenBSD. Spamd est un daemon qui va s'intercaler entre le monde extérieur et votre MTA en simulant un faux MTA. La passerelle reçoit le mail (1), le trafic est redirigé vers Spamd (2). Spamd le traite et met éventuellement à jour les règles de filtrage de PF (3). Si le mail est licite, alors le vrai MTA reçoit le message.



OpenBSD Spamd (2/4)

Blacklist :

Les blacklists résultent de la composition de plusieurs listes d'IP de MTA spammeurs. Ces listes sont récupérées soit sur des systèmes distants via HTTP ou FTP soit dans un fichier plat local. Une fois constituées, ces blacklists sont chargées dans la table <spamd> de PF via l'utilitaire spamd-setup. On notera que Spamd ne propose pas de travailler avec DNSRBL. Il est possible que le fait de s'appuyer sur une résolution DNS pour faire prendre une décision au pare-feu soit rédhibitoire pour les développeurs d'OpenBSD.

UNIVERSITÉ D'ORLÉANS

OpenBSD Spamd (3/4)

Greylist :

Spamd maintient une base de données des triplets de connexion nommée spamdb. Si un serveur ne réémet pas le message dans un temps acceptable, il est passé dans la blacklist. Si il réussit, il est mis en whitelist. Il n'aura pas à repasser par le processus d'élimination lors de la prochaine connexion, mais sera directement redirigé vers le MTA. La récupération des logs de greylisting se fait via l'outil spamdb.

```
$ spamdb | grep nico@garnett.fr
GREY|62.209.218.70|usgs.gov|<jraber@valkyrie.net>
|<nico@garnett.fr>|1194181594|1194195994|1194195994|3|0
```

La spamdb est synchronisable entre plusieurs machines dans une optique de redondance.

OpenBSD Spamd (4/4)

Spamtrap + greylisting = greytrapping

Le spamtrap est l'action de dédier des adresses email à la réception du spam (des adresses non utilisées ou présentes sur des listes de spammeurs font l'affaire). Ces adresses sacrifiées sont insérées dans la spamdb via une commande spamdb :

```
spamdb -T -a 'marketing@garnett.fr'
```

Ensuite, si une machine présente en greylist tente d'envoyer un mail à l'adresse de spamtrap marketing@garnett.fr, alors elle est ajoutée dans la table PF <spamd-greytrap> qui blacklist pendant 24h. Cette combinaison du spamtrap et du greylisting est appelée greytrapping.

Qsmtpd (1/2)

Qsmtpd est un programme en PERL traitant la partie transaction SMTP (il ne s'occupe pas du tout des aspects livraison / relai). Dans la documentation, il est dit qu'il est le mod_perl du mail. Les buts de ce projet sont multiples :

- S'intégrer facilement dans une infrastructure existante ;
- Un système de plugins évolutifs pour filtrer les mails ;
- Performance / Disponibilité (il fonctionne nativement avec les daemontools de DJB, un fichier run est même fourni dans l'archive).

Le plus simple pour intégrer Qsmtpd est de l'installer en coupure du MTA réel. Dès lors, les mails vont devoir passer par ses plugins avant d'atteindre le MTA pour être livrés ou relayés.

Qsmtpd (2/2)

Les principaux plugins sont les suivants :

- check_earlytalker : identification des machines zombies qui parlent plus tôt qu'elles ne devraient lors de la session SMTP. Le nombre de faux positifs est zéro ;
- check_spamhelo : des contrôles sont fait sur l'argument de la commande HELO tels que la propre IP du serveur, le propre nom de domaine DNS, etc. ;
- dnsbl : contient une liste de DNSBL à consulter ;
- virus/* : branche un antivirus sur Qsmtpd, on peut en chainer plusieurs ;
- ident/p0f : prise d'empreinte passive de l'OS initiant la connexion vers Qsmtpd.

Qsmtpd est un outil très pratique pour mettre en place un filtrage antispam sur des systèmes de messagerie qui en étaient dépourvus. Il a été le premier à intégrer SPF, URIBL (contrôle dans le corps du message des URL présentes pour confrontation avec une blacklist) ainsi que la méthode early_talker.

Plan

UNIVERSITE D'ORLEANS

DCC (1/2)

DCC (Distributed Checksum Clearinghouse) est distribué sous deux formes :

- Libre pour les particuliers et les sociétés dont l'activité n'a rien avoir avec l'analyse de flux de messagerie ;
- Commerciale avec une fonction de réputation pour les MTA.

Le client DCC remonte chaque message entrant au serveur, il ne cherche pas à déterminer si c'est un spam ou un ham. Il ne soumet pas une simple empreinte du message. Un traitement aussi simpliste ne résisterait pas aux altérations mineures (ajouts d'espaces, modification des liens publicitaires etc.). Il fait d'abord une empreinte classique des sections suivantes du message : IP du MTA source, Env_From, From, Message-ID, le dernier champ Received et du corps.

UNIVERSITE D'ORLEANS

DCC (2/2)

Ensuite, il applique un algorithme de *fuzzy checksums*. La façon dont fonctionne cet algorithme n'est pas claire et peu (ou pas) documentée. De plus, il évolue pour s'adapter aux nouvelles formes de spam. L'idée principale est de ne prendre l'empreinte que des parties fixes du message. Par exemple, si il commence par 'cher nico@garnett.fr' et qu'il fait moins de 5 Ko alors on prend l'empreinte de la seconde ligne et le checksum 'Fuz1' est ajouté dans la demande du client. Si le message pesait plus que 5 Ko, on ajouterait également à la demande un checksum 'Fuz2' qui serait l'empreinte de la 12e ligne.

Efficacité : détéction de 37% du spam transitant par SA.

Razor (1/2)

Razor diffère de DCC au niveau de la collecte. L'utilisateur doit soumettre manuellement une empreinte du corps de ses spams via razor-report. En général on script plutôt derrière une adresse de spamtrap. La base de données distribuée Razor ne contient donc que des empreintes de spams (contrairement à DCC qui lui contient des emails sans assertions sur le fait que ce soient des spams ou des hams). Razor opère avec deux moteurs. Le premier e4 opère sur des sous sections de chaque composant MIME du message en calculant leur empreinte SHA1. Le second e8 travaille sur les URL.

Razor (2/2)

Le problème qui se pose immédiatement est la confiance que l'on doit accorder à chaque rapport. Ainsi, depuis la version 2, la soumission nécessite que l'utilisateur dispose d'une clé GPG pour signer ses rapports. En comparant les soumissions effectuées via chaque clé, Razor maintient une base de confiance de ses signatures (notées de 0 à 100). C'est à l'utilisateur de définir le seuil de confiance à partir duquel il veut travailler.

UNIVERSITE D'ORLEANS

Pyzor

Pyzor est un projet totalement libre implémentant les principes de Razor et développé en Python. La principale différence par rapport à Razor est dans le processus de soumission. Pyzor n'utilise pas GNUPG (c'est-à-dire que les soumissions ne sont pas signées). Il n'implémente pas non plus de mécanismes de whitelist (liste d'adresses email pour lesquelles le test n'est pas réalisé).

Dspam

Dspam est un logiciel libre constitué d'une bibliothèque (libdspam), de commandes et interfaces web afin d'offrir des filtres anti-spam adaptés à chaque utilisateur. Par ses filtres Bayésiens, il est capable d'apprendre grâce aux forwards des utilisateurs à une adresse dédiée.

Il stocke dans une base de données les signatures des spams. Il fonctionne avec plusieurs types de bases de données (SQLite, Berkeley DB, MySQL, PostgreSQL, Oracle) et plusieurs MTA (Sendmail, Postfix, Exim4, etc.).

Il peut être installé comme passerelle MTA servant à filtrer les spams. Dans ce cas, il n'est plus nécessaire d'activer le filtrage Bayésien côté SA.

UNIVERSITÉ D'ORLÉANS

Plan

UNIVERSITE D'ORLEANS

SPF

Comme le serveur expéditeur peut facilement se faire passer pour un nom de domaine usurpé, SPF (Sender Policy Framework) est une norme (RFC 4408) qui définit par un enregistrement DNS les adresses IP des serveurs autorisés à envoyer au nom du domaine par la commande MAIL FROM ou HELO. Lors d'une session SMTP, le serveur de réception vérifie par une requête DNS l'enregistrement TXT du domaine concerné. Voici par exemple l'enregistrement SPF (type TXT) sur le DNS de l'Université d'Orléans :

```
univ-orleans.fr.  IN TXT ''v=spf1 mx
ip4:194.167.30.0/24 -all''
```

Signification de la syntaxe SPF :

- v= Version de SPF. Ce paramètre doit être à spf1 ;
- ip4= autorisation de la classe IPv4 à émettre ;
- mx= tous les serveurs mx du domaine sont autorisés ;
- all= toute autre machine est interdite.

SenderID

SenderID (RFC 4406) ressemble un peu à SPF du fait qu'il est définit par un enregistrement DNS les adresses IP des serveurs autorisés à envoyer au nom du domaine mais diffère par les champs testés. Cette fois-ci ce sont les champs From, Sender, Resent-From et Resent-Sender qui sont vérifiés lors d'une session SMTP, après une requête DNS de l'enregistrement TXT du domaine concerné.

La syntaxe SenderID est pratiquement identique, seule diffère la version v=spf2. On peut préciser le type de version suivant le champ que l'on veut tester.

Microsoft pousse à son utilisation avec ses serveurs Exchange (Microsoft est à l'origine de ce protocole dérivé de SPF).

Problème du forward

Les systèmes se basant sur une politique qualifiant tel ou tel émetteur en fonction de l'IP pose quand même un soucis en cas de redirection (forward) de messagerie. Si A envoie un message à B et que B a configuré une redirection vers C. C va interroger l'enregistrement SPF de A alors que c'est un MTA de B qui lui a envoyé le message. Ainsi, si on utilise un système antispam avec des scores, il est préférable de considérer ce genre de systèmes comme du bonus (si c'est OK, on réduit le score de spam, sinon c'est 0).

DKIM (1/2)

DKIM (Domain Key Identified Mail) est un mécanisme de signature d'email utilisant la cryptographie asymétrique. Un couple clé privée / clé publique est généré pour chaque serveur habilité à envoyer des emails à d'autres domaines. Un milter doit être installé sur le(s) MTA(s) émetteurs pour traiter les messages expédiés. Le milter crée une empreinte (SHA1 ou SHA256) composée du message et de quelques en-têtes. Il signe ensuite cette empreinte avec la clé privée. Les informations DKIM sont ajoutées au message :

```
DKIM-Signature:  v=1; a=rsa-sha256;
c=relaxed/relaxed; d=gmail.com; s=gamma;
h=domainkey-signature:mime-version:received:in-reply-to:ref
:date:message-id:subject:from:to:content-type
:content-transfer-encoding; bh=3oWI[...]; b=hu[...]
```

DKIM (2/2)

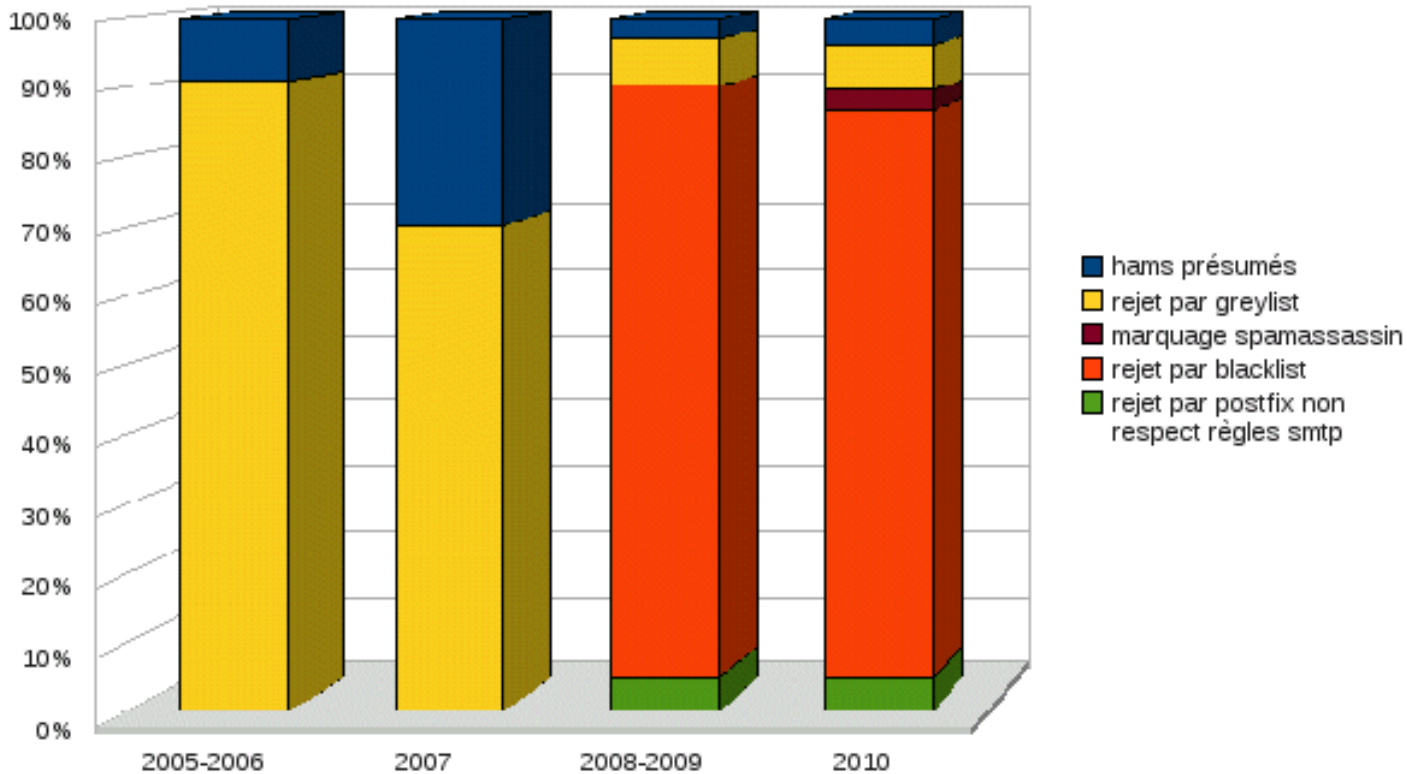
Le champ DKIM-Signature est décomposé en tags :

- v= Version de DKIM. Ce paramètre doit être à 1 ;
- a= Algorithme de hachage utilisé pour la génération de la signature ;
- c= Algorithmes de canonisation appliquées respectivement aux en-têtes et au corps du message. Deux algorithmes : simple et relaxed. Simple prépare les données pour la signature en pratiquant des altérations (suppression de lignes blanches, passage de la casse en minuscule etc.) plus légères que relaxed ;
- d= Domaine DNS cible des demandes d'obtention de clé publique ;
- s= Subdivision de l'espace de nommage défini par 'd=' ;
- h= Liste des en-têtes à signer ;
- bh= Empreinte du message canonisé ;
- b= Signature des en-têtes canonisés.

Plan

UNIVERSITE D'ORLEANS

Evolution des mesures antispams



Plan

UNIVERSITE D'ORLEANS

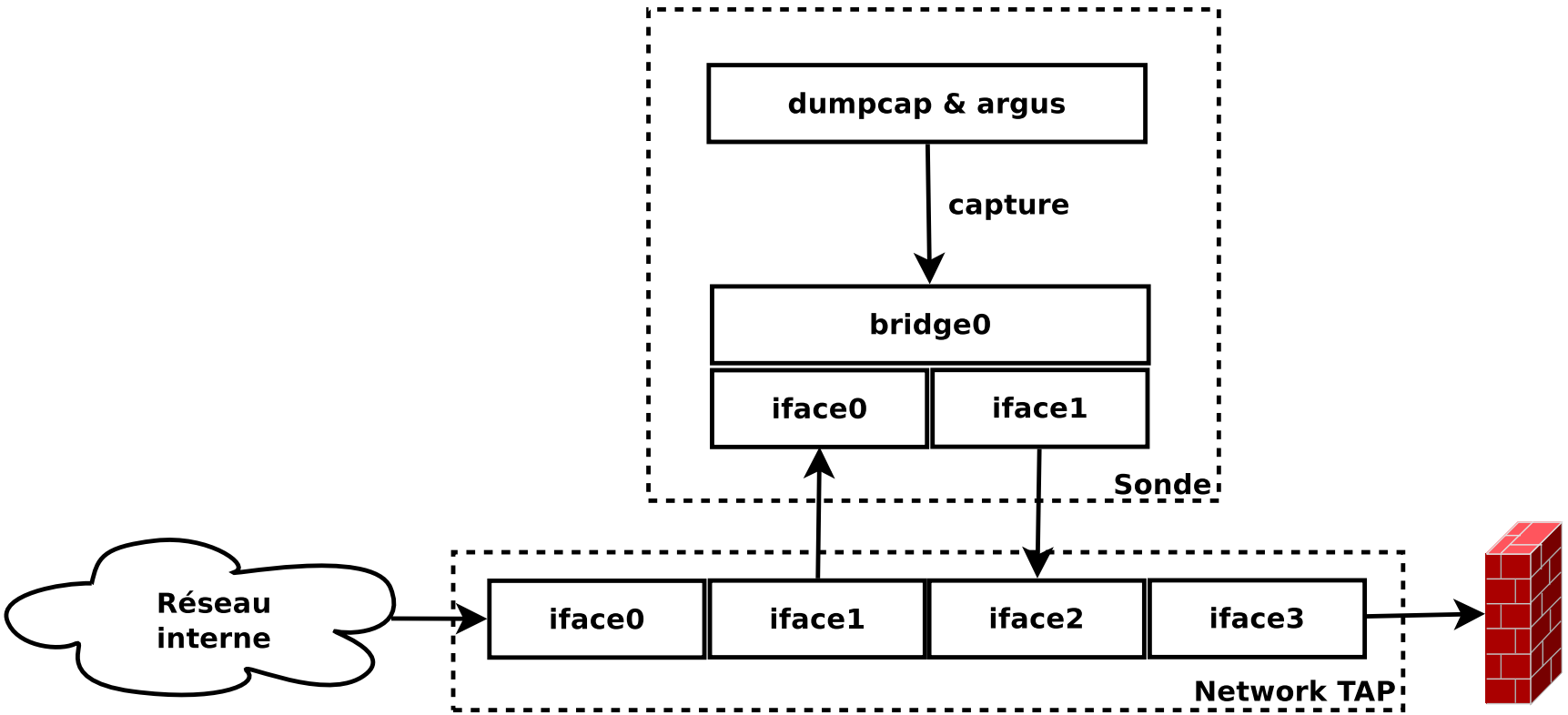
Placement de la sonde

Le placement optimale de la sonde dépend des ressources financières (puissance de la sonde) et humaines (quantité de flux à analyser). Pour notre part elle est placée à deux endroits :

- Sur la patte interne du pare-feu : fluxs des machines (non NATées) vers le world wide (wild ?) web sans tenir compte des agressions externes bloquées par le pare-feu ;
- Sur la patte DMZ du pare-feu : fluxs entre les machines externes et les services en DMZ ayant passés le pare-feu.

UNIVERSITÉ D'ORLÉANS

Schéma


dumpcap & argus
capture
bridge0
iface0
iface1
Sonde
Réseau
interne
iface0
iface1
iface2
iface3
Network TAP

Volumétrie

Quelques chiffres :

- Rotation de PCAP tout les 300 Mo, soit environ 40 secondes de traffic sur les pics d'utilisation ;
- 1 To de PCAP = traffic d'une journée et demi ;
- Fichier argus d'une journée = 5,2 Go en moyenne.

Implications :

- On oublie Snort (enfin pas tout à fait, excellent en ngrep amélioré) ;
- On oublie racluster (pratiquement 20 minutes pour traiter une requète sur une machine puissante) ;
- Focalisation sur les fluxs indépendament du payload (et du chiffrement).

UNIVERSITÉ D'ORLÉANS

On oublie réellement le PCAP ?

Collaboration avec Alcatel Lucent Bell Labs pour la détéction de botnets sur DNS :

- Fourniture des PCAP DNS ;
- Analyse en laboratoire ;
- Rapport des IP sources supposés compromises.

Et sur les fluxs ?

On se base sur les informations fournies par le CERT Renater :

- Fourniture de signatures toujours instanciables sous formes d'un filtre BPF ;
- Dans notre infrastructure, Argus permet de remonter facilement au moins 1 mois de fluxs réseaux ;
- Utilisation de ra sur les fichiers Argus pour detecter TOUTES les machines matchants le filtre ;
- Pour l'instant, on lance autant de commandes ra que de signatures BPF.

UNIVERSITE D'ORLEANS