Filtrage de messagerie et analyse de contenu

Philippe LAGADEC

DGA/CELAR
Philippe.Lagadec (at) dga.defense.gouv.fr

Résumé La messagerie est un des services les plus utilisés sur Internet et sur les réseaux d'entreprise. L'ouverture de ce service vers Internet nécessite un filtrage efficace pour se prémunir des nombreux risques inhérents au protocole de messagerie SMTP/MIME : virus, vers, contenus actifs, vulnérabilités des clients de messagerie et des serveurs, chevaux de Troie, usurpation d'identité, relayage, spam, mailbombing...Cet article présente les objectifs d'un filtrage de messagerie sécurisé, ainsi que les différentes techniques généralement utilisées, dont l'analyse de contenu. Il détaille ensuite les problèmes et les limites de ces techniques de filtrage, en apportant quelques exemples concrets comme le filtrage de scripts dans HTML ou de macros dans les documents Office, le "zip de la mort", le filtrage par nom ou par contenu. Le cas particulier des webmails est aussi abordé. En conclusion les techniques de filtrage actuelles sont imparfaites mais nécessaires, et quelques pistes d'amélioration sont proposées. Cet article est complémentaire de celui publié pour SSTIC03 à propos des formats de fichiers et du code malveillant.

1 Introduction

La messagerie est aujourd'hui un des services les plus utilisés sur Internet et sur les réseaux d'entreprise. Elle permet une communication à la fois rapide, asynchrone et bon marché, en complément du téléphone, du courrier postal et du fax. C'est également un moyen simple et universel d'échange de fichiers. Ce service devenu incontournable comporte cependant de nombreux risques en termes de sécurité informatique. Cet article dresse tout d'abord un bilan général des risques encourus, et présente certaines solutions de protection classiques, en particulier l'analyse de contenu. Il a pour but de montrer les limites actuelles de ces solutions techniques. Pour cela, on se place dans le cas général d'un réseau local connecté à Internet, dont la passerelle d'interconnexion offre un service de messagerie. Le réseau interne comprend au moins un serveur de messagerie, ainsi que des postes clients. Les utilisateurs se servent d'un logiciel client de messagerie pour émettre et recevoir des messages en passant par le serveur interne.

2 Les risques liès à la messagerie SMTP

Les risques liés au service de messagerie sont nombreux, et touchent tous les éléments qui constituent l'architecture de messagerie : réseau, serveurs, clients,

systèmes d'exploitation, applications, utilisateurs. Ces risques concernent essentiellement le protocole SMTP [1], sur lequel s'appuie le service de messagerie standard d'Internet, ainsi que ses extensions (par exemple MIME pour les pièces jointes [2]. Les paragraphes suivants rappellent succinctement la nature de ces différents risques :

2.1 Virus et Vers

Les virus et les vers constituent un des principaux fléaux d'Internet. Selon la définition couramment utilisée, un virus se présente sous la forme d'un fichier, qui infecte la machine du destinataire si jamais celui-ci l'exécute. Il peut alors se camoufler ou s'attacher à certains fichiers présents pour se reproduire, et sa diffusion est essentiellement assurée par l'utilisateur lorsque celui-ci transmet les fichiers infectés à d'autres machines. Un ver est quant à lui capable de se diffuser tout seul, généralement sans action de l'utilisateur, en se servant simplement des mécanismes ou des vulnérabilités de la machine pour en infecter d'autres. Dans le domaine de la messagerie, on voit apparaître chaque semaine de nouvelles infections, qui correspondent souvent à des hybrides virus/vers. Ceux-ci se présentent généralement sous la forme de messages contenant une pièce jointe infectée. Selon les cas, celle-ci peut être automatiquement exécutée à la visualisation du message, ou bien l'utilisateur est simplement invité à l'ouvrir. Une fois lancée, cette pièce jointe peut se comporter comme un virus pour infecter le système, et comme un ver pour se répliquer toute seule par messagerie en envoyant des messages infectés à toutes les adresses collectées sur la machine. Elle peut également déclencher une charge utile portant atteinte à la machine. La messagerie est aujourd'hui un des vecteurs les plus courants pour ce type d'infection, en raison de sa simplicité, de son efficacité, et de ses lacunes en termes de sécurité. Les vers les plus rapides se répandent dans le monde entier en moins d'une journée. La plupart des réseaux uniquement protégés par un antivirus sont donc très vulnérables au début de chaque nouvelle infection, dans l'intervalle de temps ou les signatures de l'antivirus n'ont pas été mises à jour.

2.2 Messages non sollicités, Spam

Un spam est un message commercial reçu par un utilisateur sans qu'il l'ait sollicité, souvent envoyé en masse par des logiciels automatisés. Ces logiciels utilisent des listes d'adresses de messagerie collectées sur les sites web et dans les newsgroups, ou achetées à des fournisseurs d'accès peu scrupuleux. Certains virus récents ont même été conçus pour la collecte de telles adresses ainsi que pour le relayage de spam. De nombreuses études annoncent que début 2004 la proportion de spam dans les messages échangés sur Internet atteint environ 50%, ce qui en fait un problème majeur pour tous les utilisateurs du réseau. Les deux conséquences principales du spam sont la réduction des ressources informatiques (bande passante du réseau et performances des serveurs), ainsi que la perte de temps pour les utilisateurs (lecture et suppression des spams).

2.3 Usurpation d'identité

Le protocole SMTP standard ne fournit pas de moyen d'authentifier de façon sûre l'émetteur d'un message. Il est ainsi très simple de forger un message SMTP en se faisant passer pour n'importe quel émetteur, que celui-ci existe ou non. Cette possibilité est très souvent utilisée par les virus et les spammeurs afin de camoufler la source réelle de l'envoi.

2.4 Pièces jointes, chevaux de Troie, spyware

Le protocole MIME ([2]) est principalement utilisé pour composer des messages riches, pouvant contenir du texte au format HTML ainsi que des fichiers en pièces jointes. Un fichier en pièce jointe à l'apparence anodine peut très bien contenir du code malveillant, qui s'exécute lorsque l'utilisateur ouvre le fichier ([3]). Ce code peut porter atteinte à la sécurité du poste et des données de l'utilisateur sans que ce dernier s'en rende compte. Cela peut aboutir par exemple à la prise de contrôle de la machine à distance (cheval de Troie), ou à la fuite d'informations privées ou confidentielles vers Internet (spyware).

2.5 Contenus actifs

Lorsque le corps d'un message est au format HTML, il est possible d'y inclure des scripts tout comme dans une page HTML sur le Web. Ces scripts peuvent être exécutés par le client de messagerie à la visualisation du message (suivant le logiciel utilisé et son paramétrage), et porter atteinte à la sécurité du système s'il n'est pas correctement protégé.

2.6 Vulnérabilités des clients de messagerie

Comme tout logiciel, un client de messagerie peut comporter des vulnérabilités. Si celles-ci concernent les fonctions de décodage ou d'affichage des messages, on peut aboutir à des vulnérabilités exploitables à distance par l'envoi d'un simple message, avec exécution de code en local. Un exemple simple serait un client de messagerie qui provoquerait un débordement de tampon lorsque le sujet d'un message dépasse 1024 caractères. Un attaquant peut alors forger un message avec un sujet très long, et faire exécuter le code de son choix par la machine vulnérable qui recevrait ce message, dans le contexte de l'utilisateur.

2.7 Vulnérabilités des serveurs de messagerie

Un serveur de messagerie peut aussi souffrir de telles vulnérabilités, comme on l'a vu dans le cas de Sendmail en 2003 [4]. Dans ce cas un attaquant peut parvenir à faire exécuter du code directement sur le serveur à l'aide d'un simple message SMTP.

2.8 Relayage

La plupart des serveurs de messagerie peuvent être configurés pour jouer le rôle de relais de messagerie ("*open relay*"). Dans ce mode de fonctionnement un serveur accepte tous les messages qui lui sont adressés, même si l'adresse du destinataire ne correspond pas à son domaine local. Il renvoie alors ces messages vers le serveur du domaine destinataire (enregistrement MX du DNS de ce domaine). Si un serveur est configuré de cette façon comme relais ouvert sur Internet sans contrôle particulier, il peut être utilisé par les spammeurs ou les créateurs de virus pour envoyer des messages en camouflant leur adresse source. Il est alors beaucoup plus difficile de retrouver leur trace. De plus, le relayage peut être considéré comme une complicité dans le cas d'une attaque basée sur la messagerie.

2.9 Mailbombing

Le mailbombing est une attaque en déni de service qui consiste à émettre un grand nombre de messages afin de bloquer la boîte aux lettres d'un utilisateur ou bien de saturer les ressources d'un serveur.

2.10 Web bugs

Un "*web bug*" consiste à exploiter l'affichage d'un message au format HTML dans un client de messagerie pour tracer à distance l'ouverture du message sans que l'utilisateur s'en aperçoive. Il s'agit généralement d'une référence dans le code HTML du message vers une image qui se trouve sur un serveur Web, ce serveur web étant administré par l'émetteur du message. Lorsque le client de messagerie affiche le message, il télécharge automatiquement l'image indiquée, et le serveur web est ainsi prévenu que le message est en cours de visualisation, avec diverses informations supplémentaires comme l'adresse IP du client ainsi que la nature de son logiciel d'affichage. C'est donc un moyen très efficace pour un spammeur, qui a toujours besoin de valider les adresses de messagerie qu'il utilise. Cela permet également de contrôler l'efficacité d'un mailing commercial, de façon plus sûre et plus directe qu'un accusé de réception.

3 Les objectifs d'un service de messagerie sécurisé

Compte tenu de tous les problèmes cités précédemment, un réseau utilisant la messagerie doit se protéger pour réduire les risques au minimum. Le niveau de protection et les contraintes associées doivent être adaptés aux besoins de sécurité du système protégé, notamment en termes de disponibilité, de confidentialité et d'intégrité. Bien sûr, toute mesure de protection doit aussi être compatible avec les besoins des utilisateurs pour que le service soit opérationnel.

Voici une liste non exhaustive des objectifs à atteindre pour assurer la sécurité "idéale" d'une messagerie connectée à Internet :

- Les virus et les vers doivent être détectés et bloqués avant qu'ils puissent atteindre les machines internes, en minimisant leur impact sur les performances du service.
- Le système doit aussi être protégé lorsqu'une nouvelle infection débute, et que les signatures de l'antivirus ne sont pas encore mises à jour.
- Les messages de spam doivent être détectés et bloqués au plus tôt, en minimisant le temps perdu pour l'utilisateur, ainsi que les ressources utilisées au niveau du serveur.
- Un utilisateur doit être protégé contre les messages ou les pièces jointes contenant du code potentiellement dangereux.
- Un message provenant de l'extérieur ne doit pas pouvoir être pris pour un message interne, afin d'éviter l'usurpation d'identité.
- Le service de messagerie ne doit pas pouvoir être utilisé comme relais par un tiers pour camoufler ses envois.
- Si le besoin en disponibilité est important, le système doit être protégé contre les attaques de type mailbombing.
- Les vulnérabilités découvertes dans les serveurs et les clients de messagerie doivent être corrigées au plus tôt.

La plupart de ces objectifs peuvent être atteints en mettant en place un filtrage de messagerie efficace. Il est souvent nécessaire de compléter ce filtrage par des mesures organisationnelles complémentaires, comme la sensibilisation des utilisateurs et le suivi des correctifs publiés pour les divers logiciels employés.

Pour que le service soit opérationnel, les mesures de sécurité doivent respecter les besoins fonctionnels d'une messagerie connectée à Internet et les besoins des utilisateurs :

- Le service doit accepter les messages conformes aux RFC (cf. [SMTP] et [MIME]), mais il doit souvent aussi être compatible avec les très nombreux serveurs qui ne respectent pas à la lettre ces standards.
- Les différents types de fichiers normalement employés par les utilisateurs doivent pouvoir être échangés comme pièces jointes.
- La politique de filtrage doit rester compréhensible par l'utilisateur pour qu'il puisse la respecter et comprendre la raison d'un éventuel blocage.
- Tout message bloqué ou modifié doit donner lieu à une notification claire pour l'utilisateur.
- La modification d'un message ou d'une pièce jointe pour raison de sécurité ne doit pas rendre le message illisible (perte de mise en forme) ou la pièce jointe inexploitable.
- Les faux-positifs (messages bloqués ou modifiés à tort) doivent être réduits au strict minimum, afin d'éviter la perte de données importantes.
- Le filtrage doit respecter les droits des utilisateurs, étant donné que la messagerie est considérée comme une communication privée.

En définitive, garantir la sécurité d'un réseau utilisant un service de messagerie connecté à Internet est loin d'être une tâche simple.

4 Les techniques de filtrage

Pour protéger un système utilisant un service de messagerie connecté à Internet, la méthode la plus employée est le filtrage des messages, en plaçant une passerelle de messagerie en coupure entre Internet et le serveur interne. Cette passerelle peut être constituée d'un simple pare-feu ou bien d'une DMZ plus évoluée. Suivant le résultat des divers filtres mis en oeuvre, un message peut être accepté tel quel, placé en quarantaine ou bien modifié. Les paragraphes suivants présentent différentes techniques de filtrage envisageables.

4.1 Filtrage sur les champs SMTP/MIME : émetteur, destinataire...

Les différents champs utilisés par les protocoles SMTP et MIME permettent un filtrage simple suivant divers critères. Un des filtrages les plus importants consiste à interdire tout message provenant de l'extérieur avec une adresse émetteur correspondant au domaine interne, afin d'éviter l'usurpation d'identité d'un utilisateur local. Ce filtrage est souvent dénommé "antispoofing". De même, tout message entrant doit avoir une adresse destinataire correspondant au domaine interne, afin d'éviter les problèmes de relayage. Pour les messages sortants, il est nécessaire de vérifier que l'émetteur appartient bien au domaine local, et que le destinataire n'en fait pas partie.

4.2 Filtrage sur le texte et l'objet du message

Les logiciels de filtrage de messagerie fournissent souvent des fonctions de détection de mots-clés portant sur le corps ou le sujet du message. Cette technique est essentiellement utilisée pour bloquer les messages comportant un ou plusieurs mots faisant partie d'une liste noire. Elle est efficace contre des spams ou des virus qui utilisent des mots facilement reconnaissables.

4.3 Filtrage des pièces jointes sur le nom et le "content-type"

Un fichier placé en pièce jointe d'un message grâce au format MIME est précédé d'un entête contenant plusieurs informations. Voici un exemple d'entête MIME :

```
Content-Type: application/octet-stream; name="fichier.xyz"
Content-Transfer-Encoding: base64
Content-Disposition: inline; filename="fichier.xyz"
```

On y trouve en particulier le nom du fichier (champs "name" et "filename") et une indication sur son type de contenu (champ "Content-Type"). En effet, certains systèmes d'exploitation comme Windows se basent exclusivement sur le nom d'un fichier (ou plutôt l'extension après le dernier point) pour reconnaître son type de contenu, alors que d'autres stockent cette information à part. Ce type de filtrage permet de reconnaître facilement certaines pièces

jointes à risque, comme les exécutables et scripts Windows, qui comportent des extensions connues : EXE, COM, BAT, CMD, SCR, VBS, JS, VBE, JSE...

Il est cependant impossible de garantir un filtrage parfait à partir de l'entête MIME, car ces champs sont purement déclaratifs, et rien ne garantit que l'indication "Content-Type" correspond effectivement au contenu du fichier. Par exemple, de nombreux virus contournent ce filtrage en envoyant un exécutable tout en le déclarant de type "image/gif" ou "audio/x-wav". (Ce qui pouvait provoquer accessoirement l'exécution automatique de la pièce jointe dans d'anciennes versions d'Outlook Express) De plus, on peut constater qu'une même entête MIME contient deux champs possibles pour indiquer le nom du fichier, alors que celui-ci n'en a qu'un lorsqu'il est extrait du message. Comme les clients de messagerie ne prennent pas tous en compte ces 2 champs de la même façon, il est là aussi possible d'en tirer partie pour contourner un filtrage basé sur un des deux champs uniquement. Il est également important de remarquer que certains types de fichiers peuvent être renommés avec des extensions inconnues ou peu utilisées, tout en restant potentiellement dangereux s'ils contiennent du code malveillant. Un exemple répandu est celui des documents Microsoft Office [3].

4.4 Analyse de contenu

Au lieu de se contenter de vérifier le nom d'un fichier, certains logiciels de filtrage en analysent le contenu intégral. Cette approche a pour avantage de détecter la plupart des fichiers renommés, en reconnaissant correctement leur type. Comme il existe de nombreux formats de documents qui peuvent optionnellement contenir du code sous forme de macro-commandes (documents MS Office) ou de scripts (HTML, PDF), l'analyse de contenu permet également de distinguer les fichiers purement statiques de ceux qui pourraient présenter un risque [3]. L'analyse de contenu est aussi employée pour filtrer le code HTML du corps des messages, afin de détecter les scripts. Cela permet en outre de se protéger contre les vulnérabilités exploitées par les virus pour provoquer l'ouverture automatique des pièces jointes infectées dès la visualisation des messages. (exemple des balises < IFRAME > sous Outlook Express)

4.5 Formats conteneurs et analyse récursive

Certains formats de fichiers peuvent contenir d'autres fichiers, qui sont extraits par l'utilisateur lorsqu'il veut les ouvrir. Les exemples les plus connus sont les archives compressées (ZIP, RAR, TAR.GZ, CAB...), mais il en existe beaucoup d'autres qui offrent cette fonctionnalité (documents Office, RTF, PDF, SHS...), comme le montre [3]. Si l'on se contente de filtrer les pièces jointes uniquement sur le nom de fichier, il est très simple de camoufler un exécutable ou un virus dans un fichier conteneur. Certains virus récents ont utilisé cette méthode, ce qui montre qu'un exécutable infecté dans un simple fichier ZIP est toujours efficace face aux barrières actuelles. Pour obtenir un filtrage complet, il est donc nécessaire que l'analyse de contenu soit récursive, en extrayant tous les fichiers trouvés dans les formats conteneurs.

4.6 Filtrage antivirus

Le couplage d'un serveur de messagerie avec un logiciel antivirus est aujourd'hui quasiment indispensable. Chaque message est analysé avec ses pièces jointes, et une base de signatures connues permet la détection de la plupart des virus. Il est important de noter que suivant les fonctionnalités et le paramétrage de l'antivirus, celui-ci n'effectue pas forcément une analyse complète et récursive de tous les formats de fichiers, et en particulier des formats conteneurs. Par exemple, il est très rare qu'un antivirus seul soit en mesure de détecter un exécutable infecté qui serait inclus dans un document RTF.

4.7 Filtrage antispam

Aujourd'hui le spam est un problème crucial pour la messagerie, cependant il n'existe aucune technique de filtrage capable de se protéger de façon complète et sûre. A l'heure actuelle, la meilleure solution est de combiner diverses techniques complémentaires pour distinguer les messages de spam des messages normaux. A la suite de cette détection, les messages considérés comme spams peuvent être directement supprimés, ou bien il est possible d'annoter le sujet du message pour que l'utilisateur puisse faire le tri lui-même. La seconde solution permet d'éviter de perdre un message normal en cas de faux-positif, au prix d'une perte de temps pour l'utilisateur. Voici une liste des principales techniques mises en oeuvre aujourd'hui pour lutter contre le spam :

Liste noire d'émetteurs : Cette technique consiste à maintenir une liste noire d'émetteurs connus pour avoir déjà envoyé du spam, et à refuser tout nouveau message en provenance de ces émetteurs. Il peut s'agir d'adresses individuelles de messagerie (par exemple *adam.smith@provider.com*) ou de domaines entiers (**@provider.com*). Il est possible de s'abonner à des listes (gratuites ou payantes) régulièrement mises à jour sur Internet, pour profiter automatiquement des spams détectés par des communautés d'utilisateurs. La principale limite de cette technique est que les spammeurs utilisent constamment de nouvelles adresses pour émettre, au fur et à mesure que celles utilisées sont placées en liste noire.

Liste noire de serveurs open relay : Comme indiqué précédemment, les spammeurs emploient souvent des serveurs ouverts au relayage afin de camoufler leurs traces. On peut trouver sur Internet des listes noires tenues à jour comme [5], qui indiquent les serveurs de messagerie open relay qui ont été détectés. Pour l'utiliser, chaque serveur de messagerie doit refuser tout message provenant d'un des serveurs open relay placé dans la liste noire.

Vérification DNS : Sur Internet, chaque domaine doit contenir un enregistrement "MX" (*Mail eXchanger*) dans son DNS, pour indiquer l'adresse IP de son serveur de messagerie principal. Lorsqu'on reçoit un message d'un autre serveur,

il est possible de vérifier si l'adresse IP de ce serveur correspond bien à l'enregistrement MX du domaine de l'adresse e-mail de l'émetteur. Par exemple, si mon serveur reçoit un message envoyé par le serveur 192.168.12.34 dont l'émetteur est *adam.smith@provider.com*, il faut vérifier si l'enregistrement DNS MX du domaine provider.com a bien l'adresse IP 192.168.12.34. Cette technique permet de se protéger contre les messages forgés qui correspondent à un domaine existant, et qui ne seraient pas émis par le serveur officiel. Elle oblige cependant à une gestion stricte du DNS et des enregistrements MX, ce qui n'est pas toujours possible. (cas des serveurs de messagerie secondaires)

Filtrage par mots-clés : Comme indiqué plus haut, il est possible de dresser une liste noire de mots-clés qui sont très souvent employés dans les spams ("viagra", "xxx", "porn"...), et très peu souvent dans les messages normaux. Il s'agit donc d'un indicateur simple pour la détection de spams. Cependant cette technique est de moins en moins efficace, car les spammeurs utilisent diverses méthodes de camouflage qui rendent la détection automatique de mots-clés de plus en plus difficile : remplacement de lettres par des chiffres à la forme similaire (zéro à la place de O), insertion de points entre les lettres, insertion de commentaires HTML, de lettres minuscules colorées en blanc... Les logiciels de filtrage antispam doivent être constamment améliorés pour prendre en compte les nouvelles formes de camouflage. Un autre défaut de cette technique est le nombre élevé de faux-positifs et de faux-négatifs, car les spams tendent à ressembler de plus en plus à des messages normaux.

Filtrage Bayesien : Il s'agit d'une technique statistique qui associe des probabilités aux différents mots-clés rencontrés dans un message, à l'aide d'un apprentissage progressif. En combinant les différentes probabilités obtenues suivant la méthode de Bayes, on calcule la probabilité globale qu'un message soit du spam, ce qui permet de le classer. La méthode décrite par Paul Graham [6,7] permet d'obtenir un taux de détection excellent (au-delà de 99%) avec très peu de faux-positifs, après un temps d'apprentissage suffisant. La difficulté de cette technique est l'apprentissage, qui doit être supervisée en désignant manuellement les messages normaux et les spams. De plus, cet apprentissage doit être adapté aux messages normaux reçus par chaque utilisateur pour que le filtrage soit optimal. Les résultats et les contraintes sont donc différentes selon l'implantation du filtre, qui peut être mis en ?uvre sur le serveur (exemple de SpamAssassin [8]) ou bien dans le client de messagerie (exemple de Mozilla [9]).

Filtrage comportemental : En analysant le trafic de messagerie dans le temps, il est possible de distinguer les envois normaux des envois de spam, généralement massifs et automatisés. Cette méthode est par exemple employée dans le logiciel j-chkmail [10], en complément d'autres techniques.

Autres méthodes : Etant donné qu'aucune technique de filtrage ne donne pleine satisfaction, la recherche de nouvelles solutions est très active dans le domaine de la lutte antispam.

5 Les limites du filtrage

Malgré toutes les techniques disponibles pour le filtrage de messagerie, il est actuellement difficile de se protéger de façon complète et sûre contre tous les risques présentés au début de cet article. En effet, chaque technique de filtrage possède des faiblesses et peut être contournée. Voici quelques exemples de problèmes connus.

5.1 Filtrage des types de fichiers par nom ou par contenu

Comme indiqué précédemment, le simple filtrage par nom de fichier ne suffit pas pour détecter toutes les pièces jointes qui pourraient contenir du code potentiellement dangereux. Par exemple, un document Word avec une macro auto-exécutable peut être renommé avec une extension inconnue ".xyz", et contourner la politique de filtrage. De même, l'analyse du contenu des fichiers n'est pas strictement suffisante pour détecter le type d'un fichier. Par exemple, il est parfois difficile d'identifier à coup sûr un fichier XML ou HTML en ne se basant que sur son contenu. D'autre part, certains formats contiennent des marqueurs en début de fichier, d'autres à la fin. Il est ainsi possible de forger des fichiers dont le contenu s'apparente à deux formats distincts. La détection des fichiers "à risque" doit donc obligatoirement combiner la vérification du nom du fichier et l'analyse de son contenu. Il peut être utile de corréler cette analyse avec le champ MIME "Content-Type" pour déceler une éventuelle combinaison anormale.

5.2 Récursivité de l'analyse de contenu

Les formats de fichiers conteneurs permettent souvent de contourner les logiciels de filtrage et les antivirus, par exemple en incorporant un fichier exécutable dans un document RTF ou Word (objet OLE Package). Si l'on veut obtenir une détection complète de tout code exécutable dans les pièces jointes, le filtrage doit se faire de façon récursive dans tous les formats conteneurs autorisés. Il est difficile voir impossible d'obtenir ce résultat à l'aide des logiciels de messagerie disponibles sur le marché actuel, car leur support des formats conteneurs est partiel. La seule possibilité est de limiter les types de conteneurs autorisés (ZIP, TAR.GZ, CAB...), au détriment des utilisateurs, qui ont parfois besoin des formats conteneurs moins connus (par exemple les objets OLE Package).

5.3 Performances

Il est bien sûr évident que l'analyse de contenu récursive est beaucoup plus lourde en termes de ressources qu'une simple analyse sur le nom des fichiers.

Pour certains systèmes cela constitue un problème bloquant, car il faut disposer de ressources matérielles adéquates, et le temps de transit d'un message dans le logiciel de filtrage n'est pas négligeable.

5.4 "Zip of Death"

Le "zip de la mort" est une vulnérabilité qui touche certains logiciels d'analyse de contenu et certains antivirus, et qui aboutit généralement à un déni de service. Pour effectuer l'analyse récursive d'un format conteneur, ceux-ci extraient sur disque les fichiers inclus dans un conteneur, et effectuent alors l'analyse de chaque fichier jusqu'à extraction complète. Si l'on prend un grand fichier rempli de caractères identiques (par exemple des espaces ou des zéros), et qu'on le place dans une archive compressée, on obtient un taux de compression élevé, de l'ordre de 1000 avec le format ZIP voire 1000000 avec RAR. En répétant l'opération plusieurs fois (archives imbriquées), on obtient un fichier de quelques kilo-octets, qui occupe un grand nombre de giga-octets une fois décompressé. Un fichier d'exemple connu et diffusé sur Internet fait ainsi 42 Ko, et contient plusieurs millions de fichiers de 4 Go chacun ! Si le logiciel de filtrage ou l'antivirus effectue la décompression sur disque sans contrôle particulier, cette décompression prend beaucoup de temps en occupant le serveur, et le disque utilisé peut être saturé. Afin d'éviter ce problème classique, il est indispensable de contrôler la taille d'un fichier inclus avant de l'extraire, et de limiter le temps et l'espace disque alloués à l'analyse d'un fichier. Ce contrôle doit être effectué quel que soit le format conteneur analysé.

5.5 Détection et nettoyage de scripts dans HTML, camouflage

De nombreux logiciels de filtrage proposent la détection de scripts dans le corps HTML des messages, ainsi que dans les pièces jointes au format HTML. En présence d'un script, il est possible de bloquer le message, ou bien de le nettoyer en supprimant le code du script tout en maintenant le reste de la mise en forme. Ce nettoyage n'est pas toujours parfait lorsqu'on teste les logiciels du marché, pour diverses raisons :

Emplacement des scripts : Un script peut apparaître sous diverses formes et à divers emplacements dans le code HTML : balises < SCRIPT >, évènements on-XXX (par exemple "onLoad" ou "onMouseOver"), URL d'un lien (par exemple `<A HREF="javascript:?">` ou encore `<BASE HREF="javascript:?">`)... Tous ces emplacements ne sont pas toujours correctement pris en compte.

Texte au format Unicode : Un fichier HTML est généralement au format texte ASCII (1 caractère est codé sur 1 octet soit 8 bits), mais Internet Explorer et certains autres navigateurs acceptent d'autres formats comme Unicode UCS-2 (1 caractère sur 16 bits) ou encore UTF-8 (caractères courants sur 8 bits, caractères étendus sur 16 bits ou plus). Certains logiciels de filtrage HTML peuvent être facilement contournés par cette méthode.

Caractères codés : Dans une URL, le format HTML autorise le codage de caractères sous la forme "`&#`" suivi de leur code ASCII en décimal ou en hexadécimal, puis d'un point-virgule optionnel. Par exemple, le caractère "j" peut être représenté sous les formes "`&#106`", "`&#106;`", "`&#6A`", "`&#6a`", "`&#6A;`", "`&#00106`"... i Comme tous les logiciels de filtrage ne sont pas capables de décoder ce type d'URLs, il est souvent possible de camoufler un script de cette façon :

```
<A HREF="&#106;avascript:alert('ceci est un script')">
```

Scripts imbriqués : Un script peut être nettoyé suivant plusieurs méthodes, par exemple en supprimant simplement son code, ou en le mettant dans un commentaire HTML. Si le logiciel de filtrage supprime simplement le code des scripts en une seule passe, il peut être vulnérable aux scripts imbriqués. Ce problème apparaît lorsque le code suivant est nettoyé :

```
<SCRI<SCRIPT> alert('script 1'); </SCRIPT>PT> /
alert('script 2'); </SCRIPT>
```

Si le script 1 est simplement supprimé, le script 2 apparaît sous une forme correcte et peut alors être exécuté par le client de messagerie ou le navigateur :

```
<SCRIPT> alert('script 2'); </SCRIPT>
```

Cela montre que le nettoyage de scripts par suppression doit être effectué en plusieurs passes, ou qu'un script supprimé doit être remplacé par un code neutre.

Détection et nettoyage de macros Certains logiciels proposent la détection et le nettoyage des macros dans les documents Office. Cependant cette détection ne concerne pas toujours tous les types de documents (Word, Excel, Powerpoint, Access, Project, ?), et elle n'est pas toujours efficace vis-à-vis des versions successives de ces formats de fichiers (Word 6, 95, 97, 2000, XP, 2003...).

Fichiers ou messages découpés Certains formats de fichiers comme les archives compressées (ZIP, RAR, ARJ...) offrent la possibilité d'être découpés en plusieurs parties, afin de ne pas dépasser une taille donnée, par exemple la taille d'une disquette. De même, le standard MIME [2] permet le découpage d'un message SMTP en plusieurs messages partiels, pour ne pas dépasser un seuil donné si la messagerie employée est limitée. Dans le cas d'un message partiel ou d'une pièce jointe découpée, l'analyse de contenu n'est pas toujours possible, car les structures ou les motifs recherchés peuvent être répartis sur des messages indépendants.

Fichiers chiffrés Il existe de nombreux formats de fichiers chiffrés : PGP, GPG, archives compressées protégées par mot de passe (ZIP, RAR, ARJ...), documents Office protégés en lecture par mot de passe (Word, Excel...), ou de nombreux autres formats moins connus. Dans tous les cas, l'analyse de contenu

est impossible. Il est donc nécessaire d'analyser le risque encouru pour tous les formats chiffrés que l'on autorise. Il existe un risque uniquement dans le cas où les utilisateurs disposent des logiciels nécessaires au déchiffrement.

Stéganographie La stéganographie est une technique apparentée au chiffrement, qui consiste à camoufler un fichier (ou toute information) dans les données d'un autre fichier en apparence anodin. Il est par exemple possible de camoufler un exécutable sous forme codée dans un commentaire d'un fichier HTML, ou dans les données d'un fichier image (PNG, JPEG, GIF...) ou d'un fichier audio (WAV, MP3, AU...). Tout fichier de données peut ainsi être utilisé comme conteneur. Néanmoins, le destinataire d'un tel fichier doit être en mesure d'extraire le contenu camouflé pour que cela présente un risque. Cet utilisateur doit donc effectuer une action volontaire voire disposer d'un logiciel spécifique, puisque les systèmes d'exploitation et les applications classiques ne sont pas conçus pour cela. D'un point de vue technique, un logiciel de filtrage de messagerie ne peut détecter le camouflage par stéganographie, à moins que les structures de codage soient connues.

Webmail Lorsque les utilisateurs du réseau interne ont accès à Internet via les protocoles HTTP et/ou HTTPS, ils peuvent utiliser les interfaces Web de nombreux fournisseurs pour envoyer et recevoir des messages personnels. Ces messages sont émis et reçus à partir d'un serveur situé sur Internet, en dehors du réseau interne. Ils ne sont donc jamais analysés par la passerelle de filtrage de messagerie. Lorsqu'un utilisateur reçoit un message avec pièce jointe dans une interface Webmail, il peut extraire les fichiers joints et les importer sur le réseau interne. Le téléchargement s'effectue alors grâce à un des protocoles HTTP, HTTPS ou FTP. Pour obtenir une protection homogène, il est donc nécessaire d'appliquer le même type de filtrage sur les fichiers téléchargés par ces protocoles que sur les pièces jointes des messages reçus par SMTP.

6 Conclusion

Le service de messagerie SMTP présente de nombreux risques pour un réseau connecté à Internet. Si le système contient des données sensibles, il est nécessaire de mettre en place des mécanismes de filtrage efficaces afin de se protéger, après avoir clairement analysé les menaces et défini une politique de sécurité. Il existe sur le marché de nombreux logiciels proposant des fonctions de filtrage de messagerie, avec diverses protections contre les virus, le spam et le code malveillant. Cependant le filtrage de messagerie est relativement complexe en raison des nombreux standards, protocoles et formats de fichiers mis en jeu. Les divers problèmes et limites des techniques de filtrage esquissés dans cet article montrent qu'il est difficile d'obtenir une protection complète et sûre d'un service de messagerie à partir des logiciels disponibles sur le marché. La recherche de nouvelles solutions plus efficaces est donc une necessité.

Références

1. RFC 2821 et 2822 (remplacent les RFC 821 et 822), *Simple Mail Transfer Protocol et Internet Message Format*, `http://www.rfc-editor.org`
2. RFC 2045 à 2049, *Multipurpose Internet Mail Extensions*, `http://www.rfc-editor.org`
3. P. Lagadec, Formats de fichiers et code malveillant, *SSTIC 2003*, `http://www.sstic.org/SSTIC03/interventions03.shtml` Une version mise à jour pour l'OSSIR est disponible sur `http://www.ossir.org/windows/supports/liste-windows-2003.shtml`
4. BUGTRAQ, Sendmail Header Processing Buffer Overflow Vulnerability, `http://www.securityfocus.com/bid/6991`
5. Open Relay Data Base, `http://www.ordb.org`
6. P. Graham, A plan for spam, `http://paulgraham.com/spam.html`
7. P. Graham, Better Bayesian filtering, `http://www.paulgraham.com/better.html`
8. SpamAssassin, `http://spamassassin.org`
9. Mozilla, `http://www.mozilla.org`
10. j-chkmail, `http://j-chkmail.ensmp.fr`