Contournement d'une passerelle antivirus

Guillaume Arcas and Stéphane Clodic

guillaume.arcas@retiaire.org
stephane.clodic@retiaire.org

Résumé Ce document décrit les stratégies et les techniques qu'un attaquant peut mettre en oeuvre pour contourner une passerelle antivirus. Après une présentation des fondements théoriques, nous les détaillerons en nous plaçant dans la position de l'attaquant. Nous illustrerons notre propos à partir d'exemples de filtrage des protocoles SMTP et HTTP qui sont les plus couramment utilisés par les internautes... et les virus.

1 Introduction

Un article récent du magazine MISC [1] s'ouvrait sur cette maxime : « Quiconque - utilisateur ou administrateur - ignorerait encore la capacité de nuisance des virus et vers informatiques s'expose à de graves et douloureuses déconvenues. »*. Les tendances obersvées ces derniers mois - professionnalisation et criminilisation de l'activité - en renforcent la portée et l'acuité.

Si les « développeurs » d'hier cherchaient une certaine forme de gloire et la renommée - par des moyens certes répréhensibles - les motivations de leurs successeurs sont plus sonnantes et trébuchantes - les moyens restant tout aussi sinon plus encore répréhensibles. Les coûts engendrés par les « oeuvres » des premiers étaient essentiellement indirects. Ils couvraient les opérations de restauration ou de reconstitution des données perdues ou corrompues et de nettoyage ou de reconstruction des systèmes infectés. Quand il y avait vol ou détournement, c'était généralement celui de ressources systèmes ou réseaux. Les effets de celles des seconds sont beaucoup plus directs pour leurs victimes : le vol de données, d'identifiants et d'identités ayant pour but de détourner ou d'extorquer des fonds.

Les solutions antivirus occupent depuis ces dernières années une place de choix dans l'arsenal défensif des ordinateurs et des réseaux connectés à Internet.

Ces solutions prennent souvent la forme de passerelles spécialisées dans l'analyse et le nettoyage de certains protocoles, au premier rang desquels SMTP, tant il est vrai que la messagerie électronique est encore (mais pour combien de temps ?) le principal point d'entrée et vecteur de propagation des infections informatiques.

Depuis ses débuts, la lutte antivirale est une lutte continuelle entre l'épée et le bouclier associée à un jeu du chat et de la souris lui-même doublé d'une course aux armements. Dit plus sobrement, les créateurs de virus n'ont pas tardé à inclure dans leurs productions des mécanismes anti-antivirus dont les objectifs sont d'empêcher la détection et d'interdire l'analyse du code. La présentation

exhaustive de ces techniques dépasse très largement le cadre du présent document dans lequel nous nous intéresserons plus particulièrement aux mécanismes utilisés pour contourner les passerelles antivirus.

Pourquoi s'intéresser aux passerelles ?

La nature profonde et les motivations des attaquants nous poussent à croire que ces protections seront tôt ou tard prises pour cibles comme le sont déjà les logiciels installés sur les postes de travail. Si les entreprises sont dans l'ensemble sensibilisées et pour la plupart assez bien équipées contre le risque viral, il n'en est pas encore de même pour les particuliers. Or avec la démocratisation récente et galopante des moyens de connexion permanente et à haut débit à Internet, cette catégorie d'utilisateurs risque de devenir une cible privilégiée. Le fait que la plupart des internautes ne sont pas des professionnels de l'informatique ni de la sécurité accroît leur exposition et leur vulnérabilité à l'heure du « click connect & go » généralisé. L'absence de règles et d'équipes d'administration, de déploiement et de maintenance des systèmes informatiques rend la tâche plus aisée encore à l'attaquant qui ira toujours au plus facile. Enfin, voler 10 euros à 10.000 personnes que rien ne lie présente beaucoup moins de risque que voler 1 millions d'euros à une société du CAC40.

Pour illustrer ce fait, citons un cas, survenu dans les tous premiers mois de l'année 2005. Une équipe de pirates allemands a utilisé des virus et des procédés d'hameçonnage (phishing) pour extorquer à plusieurs milliers d'internautes des petites sommes d'argent. Le montant ainsi détourné a été réinvesti dans la location de serveurs d'où ont été lancé de vastes opérations de reconnaissance et de compromission de systèmes à travers le monde. L'objectif final était vraisemblablement de constituer une armée de machines zombies destinée à faire chanter des sites de commerce ou de jeux en ligne.

La mise en oeuvre de passerelles antivirus commence ainsi à faire partie de la panoplie des services - gratuits ou payants - des fournisseurs d'accès et de services[1] . Il y a donc tout lieu de penser que les attaques contre ces passerelles seront tôt - si l'on est pessimiste - ou tard - si l'on est optimiste - intégrées aux codes malveillants. D'où l'intérêt d'en comprendre le fonctionnement dans un premier temps, de dresser un état de l'art dans un second puis de suivre les quelques pistes non encore étudiées afin de mieux organiser la défense.

2 Un (tout) petit peu de théorie

2.1 Contexte

Commençons par planter le décor : un attaquant souhaite faire entrer dans un système d'information un code viral.

Dans la suite de ce document et pour les besoins de la démonstration, nous faisons les hypothèses et utilisons les définitions suivantes :

1. Nous appellons indistinctement « attaquant » le créateur de virus et son code en action ;

[1] Citons Yahoo !Mail et Cegetel

2. Par « système d'information » nous comprenons aussi bien le réseau d'entreprise ou d'une collectivité que l'ordinateur individuel du particulier ;
3. Par « contournement » nous entendons « tout dispositif technique - passif ou actif - visant à contrevenir à une politique de sécurité fondée sur l'analyse de code exécutable potentiellement ou expressément malveillant ». Cela inclut le camouflage de code, le leurre des dispositifs de protection, l'utilisation de canaux cachés ou de moyens détournés, les attaques contre les protocoles.
4. Le code malveillant a les caractéristiques propres aux virus et vers : notamment, il doit, une fois installé sur un hôte, chercher à se reproduire de manière autonome, non discriminante et automatique. Il doit en outre pouvoir faire l'objet d'une détection via une signature. Il ne s'agit donc pas d'une attaque contre une cible unique préalablement identifiée, action qui selon nous relève du piratage. Notre attaquant ne va pas développer un code spécifique pour arriver à ses - funestes et répréhensibles - fins.
5. Les techniques de contournement sont destinées à être intégrées au code malveillant. Dans le cas d'un ver, elles seront ainsi automatiquement exploitées pour pénétrer d'autres systèmes. Elles devront donc idéalement être génériques et non liées à un produit en particulier.
6. Il existe au moins un logiciel utilisé pour mettre en oeuvre la politique antivirale en au moins un point du système cible. Ce logiciel utilise des signatures pour détecter les fichiers suspects ou infectés.

Pour finir, précisons que l'objectif de l'attaquant est bien de passer à travers les mailles du filet antivirus et non de le détruire ou l'endommager. Nous n'étudierons donc pas les techniques qui visent à faire planter les logiciels antivirus ni celles qui portent atteinte à leur disponibilité. Ces techniques ont en effet pour résulter d'empêcher l'analyse virale... mais rendent souvent impossible le passage du code viral.

Nous ne discuterons donc pas des attaques de type « Archive Bombing » qui consistent à faire parvenir à l'antivirus une archive récursive ou compressée de telle manière que son extraction ou sa décompression font exploser les ressources mémoire ou physique du serveur.

Nous nous plaçons donc bien dans l'optique d'un attaquant qui veut leurrer l'antivirus.

2.2 Fonctionnement d'une passerelle antivirus

Nous allons brièvement décrire le fonctionnement d'une passerelle antivirus. Cela nous permettra de mieux comprendre quels sont les angles d'attaque que l'on peut exploiter pour la leurrer.

De la même façon qu'un antivirus installé sur poste de travail traque les virus sur le(s) disque(s) et dans la mémoire de son hôte, une passerelle antivirus traque toute tentative d'infection à partir d'un point de contrôle idéalement unique (notion de passage obligé) sur le réseau. La principale différence avec le logiciel destiné au poste de travail tient en ce que l'analyse se fait quasi-exclusivement sur le contenu des fichiers à partir de leur signature ou empreinte.

Les notions d'analyse heuristique ou d'émulation, qui permettent dans certains cas de détecter des codes pour lesquels ils n'existe pas de signature, ne sont, à notre connaissance, que rarement utilisées sur des passerelles. La raison en est relativement simple et facile à comprendre : si un logiciel destiné au poste de travail connait l'environnement logiciel et matériel (notamment le type de processeur) sur lequel il est installé - et notamment le système d'exploitation de son hôte - une passerelle ne peut raisonnablement émuler et interpréter les appels systèmes et commandes spécifiques à plusieurs systèmes d'exploitation différents et micro-processeurs. Une passerelle se doit donc d'être « neutre » en la matière et ne doit faire aucune présomption qui soit basée sur des critères tels que : système d'exploitation cible, architecture matérielle sous-jacente, logiciels présents ou non sur les hôtes destinataires du code analysé, utilitaires de compression, etc. Nous reviendrons plus loin dans ce document sur les raisons - évidentes - de cette neutralité.

Autre facteur qui compromet l'emploi des méthodes d'analyse autre que par signature : le temps. L'utilisation d'une passerelle n'est en effet pas neutre sur la chaîne de flux prise dans son ensemble. Si la passerelle constitue un point de passage obligé - chose souhaitable - elle représente une étape suplémentaire sur le trajet des données. Dans le cas de protocoles non ou faiblement interactifs tels que SMTP, cette étape supplémentaire et la latence qu'elle génère ne sont pas bloquantes. Par contre, les protocoles fortement interactifs comme HTTP ou FTP ne peuvent supporter un temps d'analyse trop long : les logiciels clients peuvent avoir comme fâcheuse habitude de fermer les sessions trop lentes (« timeout » ou l'utilisateur perdre rapidement patience. D'où la nécessité d'utiliser les méthodes les plus rapides. Ce sont les plus simples... et les plus simplistes !

Le tableau ci-dessous présente une vision légèrement simplifiée du traitement des flux soumis à la passerelle :

Il est important de comprendre que l'antivirus (terme qui désigne dans la suite de cet article aussi bien la passerelle antivirus que le logiciel antivirus à proprement parler) n'analyse pas les paquets « à la volée » pris en dehors de leur contexte applicatif.

La première conséquence est que l'analyse induit - nous l'avons déjà dit - inévitablement une latence dans le cheminement des flux, plus ou moins importante suivant la taille et le nombre de fichiers contenus dans une session.

Dans la plupart des cas, notamment dans celui des échanges de messages électroniques, cette latence n'a qu'une faible importance, compte tenu du caractère asycnhrone des échanges entre utilisateurs. Mais dans d'autres cas, notamment dans celui des flux HTTP, cette latence peut être pénalisante et constituer un point faible apparent aux yeux de l'utilisateur et donc une tentation pour ce dernier d' « échapper » à la passerelle.

La seconde conséquence est que l'antivirus doit savoir comment interpréter des flux réseau pour les remettre dans un contexte applicatif.

Deux cas de figure :

1. L'antivirus utilise des fonctions de « décodage » embarquées dans son code. L'avantage que présente cette option est de rendre l'antivirus autonome. L'in-

Etape	Entrée	Sortie
1. Reconstruction du contexte applicatif	Trafic IP	Session applicative Exemple : dans le cas du trafic SMTP, la sortie est le message électronique dans sa totalité : en-tête contenu, pièces jointes.
2. Recherche et extraction des fichiers contenus dans la sessions	Session reconstruite l'étape précédente	Fichiers dans leur forme « brute ». Exemple : toujours dans dans le cas d'un message électronique, il s'agit des pièces jointes seules ou du contenu du message.
3. Analyse antivirus	Fichiers extraits à l'étape précédente	Pour chaque fichier un code retour OK / INFECTE ; Voir tableau ci-dessous pour le détail de cette étape.
4. Traitement du code retour	Code retour : OK/INFECTE	OK : fichier remis dans sa forme d'entrée. INFECTE : message d'alerte. Le fichier peut être joint tel quel, mis en quarantaine sur la passerelle ou détruit.
5. Reconstruction de la session	Sortie précédente	Session reconstruite à l'étape 2.
6. Réémission sur le réseau	Session reconstruite	Trafic IP.

Tab. 1. Traitement des flux soumis à la passerelle

convénient est d'en alourdir le code et d'en multiplier les fonctions internes et, par voie de conséquence, le nombre potentiel de bogues. La maintenance et la mise à jour du code devient également problématique puisqu'il doit prendre en compte les évolutions du chaque protocole supporté et la correction de chaque faiblesse pour chacun d'eux ;

2. Il fait appel à des programmes externes chargés de reconstruire le contexte applicatif. Exemple : le serveur mandataire Squid peut être utilisé pour traiter les flux HTTP ou FTP. L'antivirus agit comme redirecteur. L'analyse est donc pilotée par le serveur Squid dont la configuration peut alors comporter des failles qui permettront à l'attaquant de contourner l'antivirus.

Une fois reconstitué le contexte applicatif, l'antivirus doit identifier les fichiers contenus dans la session, les extraire et les analyser un par un.

Cette seconde étape suit un processus classique qui n'est pas à proprement parler spécifique au mode de fonctionnement de la passerelle et que résume le tableau ci-dessous :

Warning : TRIAL RESTRICTION – Table omitted !

Notes :

1. Les étapes 1 et 2 peuvent être successives, à savoir qu'une archive peut être compressée et sera donc décompresssée avant d'en extraire les fichiers contenus, ou optionnelles si le fichier n'est ni une archive ni compressé.
2. Dans certains cas, la dernière étape peut être complétée par un nettoyage du fichier quand le logiciel utilisé apporte cette fonctionnalité.

Pour la première étape, là encore, l'antivirus peut s'appuyer sur des fonctionnalités de décompression et de désarchivage internes ou bien sur les utilitaires du système sur lequel il est installé. Les avantages et les inconvénients sont les mêmes que ceux décrits précédemment pour l'interprétation des protocoles.

Intéressons-nous à l'étape 3.

Ce sont les techniques d'analyse statique qui sont utilisées pour identifier les codes viraux.

Ces techniques reposent sur ce qu'il est convenu d'appeler des signatures. Une signature peut être :

1. L'empreinte cryptographique (hachage) d'un fichier. Ce type de signature a pour avantage de s'appuyer sur un calcul fiable ou réputé tel et d'être relativement facile et rapide à obtenir. Par contre, il ne permet pas de détecter des variantes et il suffit à l'attaquant de ne modifier ne serait-ce qu'un seul bit du fichier qui a servi à calculer l'empreinte pour rendre celle-ci inexploitable. Cela n'est pas compliqué à obtenir : il suffit d'inclure des données aléatoires dans les parties du code non utiles pour fausser le calcul de l'empreinte.
2. Une suite d'éléments caractéristiques du contenu du fichier : suite hexadécimale, texte, etc. Dans cette forme, une signature peut être simple (constituée d'une chaine unique) ou complexe (type : recherche la chaine hexadécimale X à tel adresse du fichier ET la chaine Y N octets plus loin, etc.)
3. La signature que le virus insère dans dans les fichiers infectés pour éviter la surinfection ;
4. Des caractéristiques « externes » d'un fichier : nom, taille, encodage, etc.
5. Un « mix » de tout cela.

Idéalement, une signature virale doit permettre d'identifier à coup sûr un code malveillant et ne doit pas générer de faux positifs (surtout lorsque l'option de détruire un fichier suspect a été retenue).

2.3 Techniques de base de contournement

Compte tenu de ce qui a été exposé précédemment, il devrait apparaitre clairement à tout un chacun que les techniques de contournement sont multiples (pour ne pas dire légion).

Des deux tableaux qui précèdent on peut tirer les angles d'attaques suivants :

1. incapacité à traiter un protocole ;
2. incapacité à identifier et extraire un fichier d'une session reconstituée ;
3. incapacité à décoder le fichier compressé / archivé ;
4. absence de signature ; nous considèrerons dans la suite de cet article que ce cas de figure est « exceptionnel » même si c'est un risque bien réel auquel sont exposés les administrateurs systèmes de manière quotidienne !

Il est entendu que le terme « incapacité » doit être compris dans un sens très large. Nous ne nous intéresserons cependant qu'aux cas dans lesquels cette incapacité résulte d'une attaque et non d'un bogue du logiciel. Par contre, nous retenons le cas d'une attaque contre la configuration des logiciels utilisés.

Pour contourner la passerelle, l'attaquant peut donc agir sur :

1. le protocole ;
2. le format du vecteur de transport du code viral (par exemple le message électronique, la page web, etc.) ;
3. le format du fichier contenant le code viral ;
4. la configuration de la passerelle (par exemple en faisant l'hypothèse que l'analyse ne se fait pas sur tous les flux mais seulement sur les flux entrants) ;
5. le comportement de l'utilisateur (ce dernier est en effet l'élement déclencheur de l'infection, bien entendu à l'insu de son plein gré...).

3 Vif du sujet

Nous allons dans cette section passer en revue les angles cités précédemment en les abordant d'un point de vue pratique.

3.1 Techniques de contournement simples

Attaques fondées sur le protocole L'objectif de cette classe d'attaques est simple : interdire la reconstruction de la session.

La difficulté de la chose réside dans le fait que si la passerelle ne doit pas pouvoir reconstruire cette session, le logiciel client destinataire finale doit pouvoir le faire sans problème. A moins de « tomber » sur des logiciels qui, croyant bien faire, ne se formalisent pas trop des violations de certaines RFCs et passent outre certains types d'anomalies protocolaires, cette catégorie d'attaques présente peu d'intérêt. Pour cette raison, nous nous intéresserons aux attaques fondées sur le protocole plutôt qu'aux attaques contre un protocole.

En effet, la façon la plus simple d'interdire la reconstruction de la session est d'utiliser les protocoles chiffrés. Cela ne constitue pas à proprement parler une attaque mais c'est un moyen très efficace.

Ainsi, un mandataire Squid se contentera de faire transiter les flux via une directive CONNECT au nez et à la barbe de l'antivirus.

Il faut pour cela que le code viral sache utiliser ces protocoles. Il est possible que le code comporte les fonctionnalités nécessaires à la mise en oeuvre des

protocoles chiffrés. C'est le cas le plus favorable car le code serait autonome mais c'est aussi le moins réaliste compte tenu de la complexité et du coût de développement de ces fonctions et du caractère aléatoire de réussite.

Une attaque couramment utilisée et qui s'appuie sur le protocole consiste à empêcher non pas la reconstruction de la session mais sa transmission à l'antivirus.

Dans le cas d'une passerelle SMTP, cela revient à interrompre ou générer une exception dans le dialogue SMTP. Le client (c'est-à-dire l'attaquant) initialise une connexion normale avec le serveur. L'adresse envoyée par l'attaquant lors de la commande MAIL FROM est usurpée mais valide. La phase d'envoi des données (DATA) est volontairement mal terminée afin que le serveur renvoie à l'expéditeur apparent du message un avis d'erreur. Très souvent, cet avis est accompagné du message d'origine, pièces jointes incluses. L'expéditeur dont l'adresse a été usurpée se voit donc remettre un message contenant neuf fois sur dix un virus, le tout, comble de l'ironie, en provenance de la passerelle antivirus... Cette méthode permet ainsi des attaques par rebonds (ou par réflexion).

Format du vecteur Les attaques de cette seconde classe ont pour objectif, la session étant reconstruite, d'empêcher la détection ou l'extraction des fichiers qu'elle contient. En résumé, il s'agit pour l'attaquant de cacher à l'antivirus la présence de fichiers pour que la session soit déclarée correcte et sans danger puis transmise à son destinataire final.

L'une des méthodes les plmus répandues consiste à « jouer » avec l'encodage MIME.

Cette méthode présent également l'intérêt d'être multi-protocoles. Elle peut « servir » à contourner autant que faire se peut les méthodes de filtrage antispam et antiphishing fondées sur la détection d'URL.

SMTP Dans le cas d'un message électronique, cela peut consister à utiliser un type d'encodage MIME volontairement biaisé ou « obfusqué ».

Une technique très simple consiste à jouer sur la casse utilisée pour les champs des en-têtes et varier, par exemple, lettres majuscules et minuscules. De nombreuses passerelles ne savaient alors plus traiter les messages ainsi formatés.

Il existe d'autres techniques fondées sur l'encodage MIME. Il est ainsi possible de modifier le type associé à une pièce jointe. L'attaquant déclare un fichier exécutable comme étant un fichier son (Content-type : audio/x-wav). Dans certains cas, le logiciel client adaptera son comportement au type du fichier joint alors que l'antivirus aura fait confiance au type déclaré.

Autre variante : la déclaration du bon type de document mais sans indication du nom du fichier joint. L'usage veut que dans pareil cas ce soit le clainet de massagerie qui décide de la façon d'ouvrir ce document sans nom. Malheureusement, de nombreuses passerelles antivirus ignoraient tout simplement ces pièces et ne les analysaient pas.

Des attaques plus précises existent enfin, qui se fondent sur des particularités de Microsoft Outlook, comme celle qui consiste à utiliser la syntaxe CLSID

comme extension du fichier joint. Cette syntaxe se caractérise par l'emploi d'accolades dans l'extension du fichier, accolades qui pouvaient tromper les passerelles sur le type réel du fichier transmis. Là encore, l'objectif de l'attaquant est d'échapper à l'analyse.

HTTP Des techniques similaires sont aussi utilisées pour passer outre les règles de filtrage des flux HTTP. Il existe même des outils qui automatisent les transformations et les opérations de camouflage d'URL. Citons à titre d'exemple pHproxy (http ://ice.citizenlab.org/projects/phproxy/).

Une trarnsformation des plus aisées consiste à encoder les données en BASE64, format que la plupart des outils de filtrage - antivirus inclus - ne traitent pas systématiquement.

Autre exemple valable pour le serveur mandataire HTTP Squid : l'insertion d'une chaine de caractères dans une URL permettait, avec des versions 2.4 de cet outil, d'échapper à certaines ACL, notamment celles responsables de la redirection des flux vers un antivirus.

Format du fichier La manipulation des formats de fichiers est peut-être - encore - la méthode la plus souvent employée pour tromper un antivirus.

L'attaquant peut agir principalement sur deux manettes : le format d'archivage et la compression. Il peut aussi insérer du code viral dans des formats de fichiers considérés - à tort - comme sûr auparavant, comme ce fut récemment le cas avec les fichiers JPEG (à quand les virus PDF ?). Sans oublier qu'il peut également chercher à insérer son code dans des formats de fichiers non encore utilisés, et bénéficier ainsi d'un effet de surprise.

Archivage Agir sur le format d'archivage d'un fichier consiste à exploiter le facteur temps qui joue « contre » l'antivirus. Bien souvent, cela conduit les développeurs ou les administrateurs des passerelles à faire des choix pour réduire les temps d'analyse.

Par exemple, il peut être décidé de ne pas extraire les archives au-delà d'un certain seuil de récursion, la récursion dans ce contexte étant lde fait pour une archive d'en contenir elle-même une autre, cette dernière en contenant elle-même une autre, et ainsi de suite (effet « Vache qui rit »).

D'une part l'extraction consécutive de plusieurs archives est un processus consommateur en temps. D'autre part, cela présente un danger : celui d'épuiser les ressources - mémoire ou disque - de la passerelle dans les cas où l'archive est volontairement piégée.

La solution consiste à déclarer infectée toute archive qui dépasse un certain seuil de récursion, avec le risque de générer de nombreux faux-positifs et de devoir céder face au mécontentement des utilisateurs. En sens inverse, accepter de laisser passer un fichier au-delà de ce seuil sans analyser son contenu constitue une voie de pénétration royale pour les virus.

Le pire des cas de figure reste celui où l'antivirus possède des limites internes ou utilise des paramètres de configuration non documentés qui aboutissent à un

« laisser passer » inconscient. Ce cas de figure s'est déjà vu : certains produits, toujours dans l'optique de réduire les temps de traitement, se contentaient par défaut de ne rechercher les codes viraux que dans les premiers kilo-octets des fichiers analysés.

La multiplicité des formats joue aussi en faveur de l'attaquant, parfois dans des cas que l'on attendrait pas. Des versions encore récentes de l'antivirus ClamAV ne reconnaissaient pas le format TAR utilisé pour diffuser le code même de ce logiciel. Le code étant accompagné de fichiers de test, il était possible d'analyse l'archive avec le moteur ClamAV sans que celui-ci n'y détecte les fichiers de test qu'il reconnaissait par contre fort bien une fois l'archive extraite...

Compression Les remarques précédentes s'appliquent également au mode de compression. Un fichier peut être compressé plusieurs fois dans le but de ralentir le processus d'analyse. La compression récursive peut aussi être utilisée pour construire des fichiers piégés dans l'ouverture aboutit à la mise hors service de l'antivirus par épuisement de ses ressources.

Une autre méthode plus subtile d'utilisation de la compression pour leurrer l'antivirus consiste à utiliser des formats peu utilisés dans l'espoir - souvent exaucé - que la passerelle, ne sachant ouvrir les fichiers ainsi compressés, ne saura les analyser. Récemment, le format RAR a ainsi été retenu dans ce but. Heureusement, ce type de faille est généralement rapidement comblée par les éditeurs.

Il faudrait pour bien faire imposer à l'ensemble de ces utilisateurs l'utilisation de quelques formats de compression - et d'archivage - bien contrôlés, mais cela induit une bien trop grande contrainte, notamment vis-à-vis des correspondants externes.

Panachage N'oublions pas de dire que les méthodes précédentes peuvent - et sont souvent - utilisées conjointement, parfois avec succès. Il n'est ainsi pas rare de trouver des archives compressées contenant des achives compressées avec un utilitaire différent.

Exemple (volontairement tiré par les cheveux) : une archive TAR compressée avec Gzip contenant une archive ZIP Bzippée.

Maquillage d'extensions Cette méthode assez ancienne ne devrait plus tromper grand monde. Pourtant, son utilisation est encore courante. Cependant, plus que tromper l'antivirus qui ferait encore confiance à la seule extension d'un fichier pour décider de son analyse, elle est plus vraisemblablement destinée à tromper la vigilance de l'utilisateur ou de son logiciel de messagerie qui s'arrêterait à la première extension visible pour vérifier le type du fichier ainsi transmis et cliquer.

Exemple fréquemment rencontré : naked_woman.jpg. [nombreux espaces] .exe

Nouveaux formats de fichiers Trouver de nouveaux vecteurs de propagation est un des sports favoris des créateurs de virus. Une telle découverte permet en effet

de bénéficier, pendant un certain temps, d'un effet de surprise et de prendre sur les éditeurs d'antivirus une longueur d'avance, même temporaire.

Récemment, les fichiers TNEF utilisés par Microsoft Outlook ont ainsi servis de supports à des infections jusqu'à ce que les antivirus soient mis à jour pour supporter ce format dont l'acronyme n'est pas dénué d'une certaine ironie : Transport Neutral Enconding Format.

Attaques par rebonds Nous avons décrit plus haut une attaque de ce type mais cette technique s'applique à d'autres cas.

L'exemple le plus simple consiste à utiliser un format de fichier peu usité sur certaines plates-formes mais plus répandus et inoffensifs sur d'autres, ce que le scénario suivant illustrera mieux qu'un long discours.

Un code viral pour plates-formes WinTel est envoyé dans un format lisible uniquement sur Apple Mac. Si la passerelle n'a été configurée que pour supporter les formats pour WinTel, ce fichier passera inaperçu. Dans une version simple de ce scénario, l'utilisateur destinataire du fichier demandera à un collègue Macistes de bien vouloir lui covnertir cette archive très importante reçue d'un ami russe dont il ignorait cinq minutes auparavant l'existence. Variante : le message qui véhicule l'archive comporte l'URL d'un site qui justement fourni l'utiliraire de décompression idoine qui permettra de visionner le contenu du fichier naked_girl.sit sous MS Windows...

Une variante plus élaborée consiste à utiliser des virus de macros MS Office dont certaines ne seront exécutables que sous une plate-forme donnée. L'idée là encore est de pousser l'utilisateur) diffuser ces fichiers à l'intérieur du réseau une fois franchies les barrières antivirus.

Ces techniques justifient qu'un logiciel antivirus pour passerelle doit être capable d'analyser des fichiers sans faire de présomptions quant aux plates-formes protégées. Cela s'applique aux formats d'archivage, de compression et aux macros.

Comportement de l'utilisateur Dernière manette sur laquelle il est possible d'agir : l'utilisateur. Disons le tout de suite : c'est la méthode parfois la plus simple et très souvent la plus efficace.

Cela peut aller de l'incitation de télécharger et d'installer lui-même le code viral sous des prétextes divers et variés, les visionneuses vidéos ayant, allez savoir pourquoi, beaucoup de succès, à la fourniture de données personnelles sans utilisation de code exécutable. L'hameçonnage (phishing) serait ainsi un exemple ultime d'infection comportementale.

L'utilisateur peut une fois encore être l'outil d'une attaque par rebonds. Par exemple, il peut être invité à télécharger depuis son ordinateur un utilitaire de sauvegarde du carnet d'adresses de son GSM, utilitaire qui contiendra un virus pour mobile. Le monde sans fil - WiFi, Bluetooth - n'est pas encore totalement intégré dans les politiques de protection antivirus même si il fait de plus en plus partie du périmètre du réseau de l'entreprise.

3.2 Techniques avancées

A côté des techniques somme toute relativement simples décrites dans les précédents paragraphes, il existe, certes sous forme encore théorique, des stratégies plus élaborées.

Code modulaire Cette technique consiste à envoyer le code viral en deux ou plusieurs composants d'apparence anodins mais qui, assemblés, révèlent toute leur capacité de nuisance. Elle s'inspire de ce qui se fait dans le domaine de la guerre chimique et bactériologique.

Un exemple imparfait consiste à envoyer le code actif dans un fichier ZIP avec mot de passe, et le mot de passe dans un second message.

Un exemple plus proche de l' « esprit » NBC[2] consisterait à envoyer dans deux messages distincts une charge inerte et son détonateur. On peut penser à une image piégée et à une visionneuse contenant les fonctions d'activation.

Utilisation de plusieurs canaux Les techniques précédentes peuvent s'enrichir, si l'on peut dire, en variant les canaux utilisés pour véhiculer les composants. L'exemple le plus simple consiste à envoyer l'URL d'une page piégée dans un message électronique. L'attaquant peut espérer que l'entreprise n'aura pas encore déployer d'antivirus pour flux HTTP. Les webmails sont des exemples involontaires de pénétration d'un réseau par des canaux détournés.

Il est entendu que les techniques de camouflage MIME décrites dans la section précédente sont tout à fait exploitables dans ces cas de figure.

4 Sometimes (sh)it happens...

Certaines des techniques évoquées ci-dessus sont loin d'être innovantes ni même théoriques. La liste (non exhautstive) suivante illustre que « ceux d'en face » n'ont pas attendu avant de les mettre en oeuvre :

1. **2004-11-29 : Zip Files Detection Evasion Vulnerability**.- De nombreux antivirus ont été victimes de cette faille qui permettait de camoufler la taille réelle d'un fichier ZIP. Pour une question de rapidité dans l'analyse, ces antivirus ne traitaient en effet pas les fichiers ZIP dont l'en-tête indiquait que leur taille était... égale à zéro ! Les dommages étaient d'autant plus grands que les messages parvenaient aux utilisateurs avec une indication faussement rassurante ajoutée par l'antivirus.

   Cette faille était exploitable sur les produits Computer Associates, Kaspersky, Sophos, et McAfee, pour ne citer que les plus connus.
2. **2004-09-27 : Reserved MS-DOS Name Scan Evasion Vulnerability**.- Une erreur dans l'architecture d'un logiciel antivirus a été exploitée pour le contourner. Cette faille permettait de faire passer des fichiers en utilisant un certain format pour leur nommage.

[2] Acronyme pour Nucléaire, Biologique, Chimique.

5 En guise de conclusion

Cette introduction aux techniques de contournement des passerelles antivirus montre, pour ceux qui en douteraient encore, qu'une politique de défense dans la profondeur est la meilleure des protections possibles. Si l'antivirus passerelle laisse passer un fichier, il faut qu'une solution installée sur le poste de travail l'intercepte. Ne serait-ce que parce que les techniques heuristiques et les émulateurs ne peuvent s'exécuter que sur les postes de travail.

L'offre en matière de logiciel Libre, même si elle se limite au projet ClamAV, permet de s'affranchir de la contrainte « surcoût financier ».

Il apparrait en outre qu'il devient nécessaire d'équiper les passerelles d'une brique supplémentaire dont la fonction sera de nettoyer les anomalies protocolaires et d'uniformiser certaines caractéristiques comme les formats d'encodage. Cette brique s'intègrera entre le serveur mandataire et l'antivirus. Elle est l'équivalent de la commande SCRUB pour le pare-feu PF d'OpenBSD/FreeBSD.

Ceci étant dit, l'utilisateur reste le maillon faible de la chaine.

Si cet appareil défensif - passerelles, antivirus sur chaque poste de travail, équipe dédiée à la sécurité et à sa supervision - est à la portée de nombreuses entreprises, il n'en est pas de même pour les particuliers. Or ces derniers constituent une cible privilégiée. Il peut donc être bon de se poser la question de savoir si leur protection, même a minima, ne devrait pas être une obligation contractuelle ou légale pour les FAI. Après tout, la gestion des autoroutes par des sociétés privées n'affranchit pas les automobilistes du respect forcé du code de la route...

Références

1. G. Arcas et S. Clodic, *ClamAV, l'antivirus qui vient du froid*, MISC- Le journal de la sécurité informatique, numéro 17, janvier 2005.