Rendre son infrastructure sécuritaire

Fabrice Prigent
Centre de Ressources Informatiques, Cellule Réseau
Université Toulouse 1 Sciences Sociales, 2, Rue du Doyen Gabriel Marty 31042 TOULOUSE
Fabrice.Prigent@univ-tlse1.fr

Résumé

Dans un contexte changeant, où les agressions n'ont pas d'horaires, la défense périmétrique statique n'est plus suffisante, la juxtaposition d'outils de sécurité non plus.

Une des adaptations possibles consiste à fédérer l'ensemble de l'infrastructure informatique de l'organisme, ses processus et ses matériels afin de pouvoir :

- *détecter les comportements déviants, généralement liés aux agressions (automatiques ou non) ;*
- *lancer, de manière automatique, des processus de mise en quarantaine afin d'isoler la menace, en avertissant de manière claire et précise les divers protagonistes de l'incident ;*
- *permettre la suppression du processus d'agression pour terminer par la « libération » de la ou des machines incriminées.*

On retrouve des principes dans l'approche dite de la défense en profondeur [1]. Les résultats de l'application de cette approche sur une université seront ensuite présentés.

Mots clefs

Sécurité, détection, intrusion, quarantaine, IDS, anormalité

1 Une journée en enfer

Pourquoi moi ? Qu'ai-je bien pu faire pour mériter cette journée d'octobre 2004. Le casse-pied du jour est arrivé par portable, ou clé USB, ou encore par mail. Le virus est inconnu, (il ne sera repéré que 72 heures plus tard) mais il me plaît déjà : 2 IGR, 1 IGE, 1 ASI et 1 TCH en train de courir partout derrière lui, interruption d'une rencontre avec des huiles pour vérifier le portable de la « personnalité extérieure », une désactivation en catastrophe de bâtiments entiers.

Selon la définition de Larry Wall, je suis un bon informaticien : je suis fainéant. Et là je suis servi : courir, désactiver des prises, désinfecter les postes...

J'ai pourtant tous mes gadgets : 1 antivirus de messagerie de marque X, 1 antivirus sur les postes de marque Y, 1 firewall aux petits oignons avec ses 4000 lignes de règles. Je lui ai même ajouté un IDS pour faire bonne mesure. Mes serveurs ont les IPTables activées.

Mais bien sûr, ça ne marche pas : Le firewall n'a rien vu passer, et n'a donc rien bloqué. Les IPTables font bling bling, juste histoire de saturer mes logs, l'IDS me dit que tout va bien, et les antivirus tamponnent le message « approuvé » sur ce programme atteint d'ubiquité galopante.

Je vais m'enfermer dans les toilettes pour parler à mes pieds....

1.1 Fric, Sexe, et Mafias

Le contexte de l'agression informatique a énormément évolué avec les années. Nos petits hackers d'hier avec leurs lunettes et leur acné juvénile désireux de marquer de leur empreinte un site web, ou bien les barbus révoltés contre la société de consommation qui s'attaquaient aux grandes firmes ne sont plus.

Les nouveaux agresseurs sont des ingénieurs, bien propres sur eux, que l'explosion de la bulle internet a laissé sur le carreau et qui s'insèrent dans une nouvelle économie quasi mafieuse : je crée un virus (ou un ver, un site infecté, etc. rayez la mention inutile) pour prendre le contrôle de PC. Je vends ou je loue ces PC à des sociétés qui diffusent du SPAM. Ces sociétés vendent leur capacité de diffusion à d'autres sociétés qui vendent des pilules bleues, des montres avec des vrais morceaux de plastique dedans, des comptes en banque nigérians, des sites avec pack demoiselles en cuir et exploitation de faille intégrée, etc.

1.2 Pourquoi c'est toujours moi ?

Mais pourquoi moi ? Je ne suis qu'un administrateur d'une université en sciences sociales. Je n'ai pas de secret d'état, de découvertes qui révolutionneront le monde pour les vingt ans qui viennent, je ne gagne pas d'argent avec mon site web.

Certes, mais j'ai des ordinateurs et de la bande passante. Alors, forcément la probabilité que ça me tombe dessus est forte vu le prix d'un botnet[1]. Au mauvais endroit, au mauvais moment, je vous dis...

1.3 On ne me la fait pas à moi !

Mais si ! La compétence des meilleurs pirates s'est accrue (pas celle des scripts kiddies). Certes le fait de cliquer sur un exécutable attaché à un mail en dialecte indonésien est toujours possible par la grâce d'un QI disputant son niveau avec celui d'une huître, mais cela se raréfie.

Des liens youtube, des messages web semblant provenir de votre antivirus arrivent à piéger les plus aguerris des utilisateurs.

[1] Botnet : réseau de PC contrôlés par un pirate.

Sans parler des failles XSS[2], CSRF[3], et autres billevesées qui permettent, grâce à de magnifiques bibliothèques tel que le MPACK[4], de transformer le site web des autres, en plate-forme d'infection.

Nous n'oublierons pas le modèle multi-couche qui fait que même si le produit initial est sécurisé, ses 3000 plugins ne le sont pas tous (et loin de là !).

La psychologie fait aussi désormais partie du paquetage de l'auteur de virus. La conclusion est sans appel : le ver, le virus ou l'agression réussira. En tout cas, sur le poste d'un utilisateur.

2 Virus : 58 minutes pour vivre.

Le constat est clair. La défense doit réussir tout le temps, l'attaque une seule fois. Ne jetons pas nos firewalls, antivirus et IDS, mais bon : il faut être réaliste. Et puis sans rebondissement, on n'a pas de bon film.

Le premier constat, c'est que mon employeur est rarement une cible autre que pour créer des botnets. Préoccupons nous donc de ce qui va nous saboter des journées entières : les attaques automatiques rapides.

2.1 Ver et virus

Les agressions les plus virulentes d'Internet ont majoritairement trois buts : constituer un botnet, envoyer des spams, attaquer des sites. Les deux derniers buts nécessitent que le premier ait été atteint, que ce soit par réalisation personnelle, ou par location/achat.

Les vers et les virus sont clairement le meilleur moyen à l'heure actuelle (cela ne va pas forcément durer) de se constituer un botnet.

La différence entre ver et virus se fait principalement sur le mode de propagation : le virus nécessite une intervention humaine pour se propager, le ver se propageant seul par l'exploitation de vulnérabilités. Nous emploierons le terme virus pour désigner les deux éléments dans la suite de l'article.

2.1.1 Des réalisations de haut niveau

Les virus sont actuellement conçus par des ingénieurs tout à fait au courant des dernières technologies. Ils sont aidés en cela par des études théoriques de grande qualité sur :

- les méthodes optimales de propagation [2] ;
- les algorithmes efficaces de chiffrement ;
- les analyses de rootkits créés par des firmes ayant pignon sur rue.

[2] Cross Site Scripting : utilisation de failles sur un serveur web acceptant des informations « externes » pour faire exécuter des javascript malicieux par les clients.

[3] Cross Site Request Forgery : utilisation de l'authentification d'un client sur un site sécurisé pour lui faire exécuter des actions imprévues. Par exemple en lui envoyant un mail avec une image dont le SRC serait l'url du script permettant un virement et qui est placé sur le site bancaire sur lequel la victime est authentifiée.

[4] MPACK est un programme vendu aux alentours de 700 dollars, mis à jour régulièrement et qui permet de créer des serveurs web d'intrusions. On pourra aussi se tourner vers zunker et Icepack.

2.1.2 Des fonctionnalités impressionnantes

L'amélioration régulière des processus de fabrication [3] (des traces d'IDE ont été trouvées dans certains virus), la constante progression des divers domaines de la sécurité informatique donne aux nouveaux virus des capacités :

- de polymorphisme, afin de compliquer la réalisation des signatures virales ;
- de camouflage, par des procédés de plus en plus élaborés , jusqu'à la virtualisation[4] ;
- de « nettoyage », afin de désactiver les firewall, antivirus et antispyware locaux qui pourraient les découvrir ;
- de blindage, afin de retarder leur analyse par les auteurs d'antivirus par le biais de la cryptographie, de codes morts, de branchement aléatoire et d'anti-debuggage ;
- de mise à jour : pour changer la signature, pour obtenir de nouveaux modes de propagation, de nouveaux modes d'attaque ou encore de nouvelles fonctionnalités ;
- de génération de réseaux P2P pour les canaux de contrôle et de mise à jour.

2.1.3 Un délai de protection allongé

Chacune des fonctionnalités précédentes va rendre évidemment beaucoup plus longue et complexe la mise à niveau des détections.

Une rapide recherche des délais moyens "courants" dans l'ensemble des phases avant la mise en place d'une signature donne les chiffres suivants :

- délai d'analyse d'un code viral : de 10 minutes à 24 heures, voire 48 heures pour les plus résistants ;
- délai de réalisation d'une signature correcte (sans trop d'effet de bord) : de 10 minutes à 4 heures ;
- délai moyen de mise à disposition : 1 heure ;
- délai moyen de mise à jour : 1 heure.

Cela nous amène rapidement à considérer un temps minimal de 3 heures entre l'apparition d'un nouveau virus et la mise en production de la parade. Même le virus le plus mal conçu va être libre d'agir pendant au moins une heure.

2.2 L'infection

En général elle est due à la négligence de l'utilisateur :

- un clic de souris malencontreux lors de la consultation d'un site web ou d'un mail ;
- un oubli de mise à niveau d'un logiciel ;
- l'introduction d'un logiciel de provenance douteuse.

Deux solutions contre ces problèmes: mettre un firewall, et mettre un antivirus. Sachant que :

- le premier ne protège que des vers ;
- le second que des virus ;
- l'antivirus n'est jamais plus efficace que sa dernière mise à jour.

Il devient évident qu'un virus passera à un moment où un autre l'antivirus et qu'il s'empressera de désactiver tout ce qui pourrait lui nuire. D'autant plus que nous aurons droit à :

– la désactivation d'un procédé de protection "parce qu'il est gênant" ;
– l'oubli de la mise à jour anti-virale ;
– la consultation des mails après 3 semaines de congés, avant même que l'antivirus soit mis à jour.

2.3 La propagation

La propagation virale peut être de plusieurs types : l'envoi massif de mails pour contaminer des correspondants, utilisation des partages Microsoft (ou éventuellement NFS, Appleshare), ou utilisation de vulnérabilités dans les systèmes d'exploitation concernés.

Le marketing nous propose la vision idyllique suivante : nos antivirus protègent vos postes fixes, donc rien à craindre. Contre les postes portables infectés, une zone de quarantaine est définie. Le poste y est alors ausculté. Traduction : on vérifie qu'il a les signatures antivirales à jour, le firewall actif, et les bonnes clés de registre.

Mais, cf ci-dessus, les antivirus ne seront pas à jour à un moment ou un autre, les virus savent se cacher, y compris d'un programme supplémentaire de protection aussi « intelligent » que ses prédécesseurs.

2.4 Les cons, ça ose tout, c'est même à ça qu'on les reconnaît.

Il est bien évidemment hors de question d'abandonner la partie. Si nous ne pouvons identifier un ver ou un virus inconnu, il est malgré tout possible de le repérer :

– quand il se propage ;
– quand il attaque.

Nous n'avons droit à aucune erreur, certes, mais le terrain est à nous. Un agresseur automatique va forcément utiliser les services réseaux que nous mettons à sa disposition. A nous de les surveiller correctement.

Le principe est aussi simple que la vie : il faut profiter de tout, mais n'abuser de rien. Les agresseurs automatiques ne l'ont pas compris.

2.4.1 Le terrain

L'ensemble des détections va supposer la mise en place sur le réseau de l'établissement de plusieurs éléments : un firewall, un proxy transparent, un ordinateur capable de regrouper les renseignements : un simple serveur de log fera l'affaire.

2.4.2 Les virus de messagerie

Ils ont deux caractéristiques : ils ont quasiment tous un serveur SMTP embarqué, et ont une cadence de tir impressionnante. Il suffit donc d'avoir un firewall qui bloque le SMTP vers l'extérieur et journalise les refus. Un simple comptage par IP va permettre d'identifier le poste infecté.

Les faux positifs sur cette méthode sont dûs principalement à des personnes ayant un serveur smtp extérieur configuré sur leur poste. La difficulté se contourne aisément en faisant un comptage par le couple (IP source, IP destination).

On abandonne à leur sort les propriétaires de mac (ou autres) qui ont installé un serveur SMTP sur leur poste.

Mais, me direz-vous, il existe des virus utilisant les services locaux, et ceux-là tu ne les vois pas. Moi non ! Mais le serveur SMTP oui. Il pourra détecter deux éléments différents : une fréquence d'envoi trop élevée (seul superman peut envoyer plus de 10 mails par minute), ou des changements d'expéditeur trop rapides (10 par heure est vraiment un maximum).

On exclura évidemment les serveurs SMTP internes de cette détection.

2.4.3 Les vers netbios

Le principe est identique : qui aurait l'idée de laisser passer un protocole Microsoft sur le réseau ? Donc, il suffit d'un firewall, des logs et une remontée d'alerte si l'abus est manifeste.

Les faux positifs ont la même origine : une configuration du poste pour utiliser, par exemple, les imprimantes de son organisme d'origine. Même motif, même solution.

2.4.4 Les vers à vulnérabilité

Derrière cette dénomination se cache l'ensemble des vers utilisant des failles de sécurité présentes sur les services mis à disposition.

Le repérage ne peut se faire qu'avec une structure plus lourde nécessitant la présence d'un coordinateur, la mise en place de filtres à journalisation, idéalement sur toutes les machines de l'établissement, et un travail de réglage.

Les vers de ce type vont généralement tenter de faire un minimum de propagation locale et donc faire un test du service sur les machines locales. Si les refus sont systématiquement remontés vers la même machine, celle-ci va pouvoir identifier une attaque, et donc repérer l'IP concernée.

2.4.5 Les mises à jour P2P et les attaques

Les virus modernes utilisent de plus en plus des processus de mise à jour P2P, ce qui rend l'idée même de blocage des serveurs de mise à jour inopérante.

Génial ! Ce comportement est le même que trois autres plaies de nos réseaux : les attaques DDOS qui proviendraient de chez nous, les logiciels de P2P, et le fameux logiciel chiffré de TOIP, VOIP, VidéoOIP, etc.

La détection se fait tout simplement en regardant le nombre d'ouvertures de session par minute pour chaque adresse IP sur le firewall (module ipt_recent de IPTables par exemple).

Pour éviter les faux positifs on exclura certains serveurs particulièrement bavards.

2.4.6 Les spywares orientés

Autre plaie, même si moins dangereuse, les spywares à affichage incongru (style demoiselles en tenue d' Eve) sont aussi repérables de manière globale. On doit les éradiquer car ils cachent souvent des problèmes plus graves (chevaux de Troie, keylogger, etc.). Aucune supposition ne peut être faite sur l'utilisateur, car, les postes sont parfois allumés par

d'autres personnes que leur propriétaire pour ce genre de consultation.

La détection se fera simplement grâce à des dispositifs de filtrage d'url (transparents ou non) avec des listes noires conséquentes qui signaleront les accès anormaux.

A voir ensuite, le seuil de tolérance adéquat pour que cette détection ait une utilité sans devenir un père fouettard.

2.4.7 Les bots web

Basés sur la recherche de vulnérabilité dans les programmes en langage PHP, ASP, perl et autres, ils sont souvent la cause des défigurations. Leur repérage nécessite un travail plus lourd au niveau de chaque serveur de l'établissement, et une configuration spécifique des proxies.

Hormis les bots travaillant par googlehacking[5], les bots web recherchent les sites vulnérables en tentant directement de se connecter aux urls les plus alléchantes (http://...../wp.cgi ou http://..../awstats.cgi) auxquelles ils ajoutent les paramètres idoines.

Leur détection, s'ils sont internes, se fera tout simplement en comptabilisant le nombre d'erreurs 4xx ou 5xx dans les journaux des proxies. Un seuil dépassé indiquera, sinon un poste infecté, du moins un poste « à surveiller ».

Si le bot se concentre sur le réseau local, ou si l'on souhaite repérer les agressions extérieures, on pourra recenser tout simplement les urls à paramètres (GET ou POST) utilisées en temps normal dans l'établissement grâce à des expressions régulières. Le travail est long et fastidieux au début, mais permet de repérer très facilement un grand nombre de tentatives d'intrusions web : une requête ne faisant pas partie de la liste devient de suite très suspecte.

2.4.8 Trop c'est top !

Les agressions actuelles évolueront forcément, avec des méthodes plus astucieuses, plus élaborées et sur des protocoles pas vraiment prévus. Mais les agresseurs automatiques en feront toujours trop. C'est ça qui est top.

On se plongera avec enthousiasme dans l'analyse de fréquences de tout un tas de paramètres plus amusants les uns que les autres :

- le nombre de requêtes DNS formulées par les postes ;
- la CPU des commutateurs ;
- les échecs d'authentification aux divers services ;
- l'évolution du nombre de communications UDP ;
- etc.

Tout cet ensemble de petites analyses finissent par permettre une vision pro active des problèmes.

3 Piège de cristal

« Allo, central, j'ai 15 terroristes allemands, dont 2 blessés, ils sont équipés d'armes lourdes. ». Bravo, McClane, le repérage a eu lieu, mais maintenant, il va falloir se bouger un peu. Savoir que l'ennemi est là n'est qu'un début.

Premier problème : les terroristes attaquent à la Noël,... ou pendant les congés, ou le vendredi soir, bref .... quand on n'est pas là. Il faut donc réagir automatiquement. Ceci suppose que le serveur syslog est capable de donner des ordres aux équipements.

Deuxième problème : quel type d'action peut être effectué sur un poste infecté sachant que le tir au fusil à pompe sur machine infectée n'est pas encore une discipline olympique validée ? Il faut trouver autre chose, l'idée principale étant de rendre inoffensif l'ordinateur (et éventuellement l'utilisateur).

Troisième problème : la réaction automatique est aveugle et surtout relativement muette. Or, se faire "geler" sur place sans que l'on sache pourquoi est rarement du goût de nos utilisateurs. Allez savoir pourquoi.

3.1 Réagir

La réaction suppose une autorité informée et avec des prérogatives importantes. Un serveur SMTP peut décider d'agir de manière purement locale, mais cela limitera la protection.

Il devient alors évident qu'un ordinateur (au sens littéral du terme : celui qui ordonne) devra imposer à l'ensemble du système informatique des mesures restrictives, et ce en toute connaissance de cause.

Les choses à considérer sont nombreuses :

- faire considérer par tous les éléments de la sécurité, cet "ordonnateur" ;
- sécuriser les communications entre l'ordonnateur et les "acteurs" (par le chiffrement, l'isolation, le filtrage, l'authentification forte) ;
- lui fournir tous les éléments nécessaires à sa décision (journaux, état, etc.).

3.2 Isoler

Le nettoyage à distance d'un poste étant quasiment impossible (poste invité, pas de droit administrateur dessus, aucune routine disponible), la seule solution est donc la mise en quarantaine du poste infecté. Plusieurs méthodes sont disponibles.

3.2.1 La hache

Méthode très efficace : elle consiste à couper l'accès réseau du poste par la désactivation, par exemple, d'un port de commutateur. Mais elle a plusieurs inconvénients :

- elle nécessite une parfaite connaissance en temps réel de son réseau (telle machine sur tel port) ;
- elle coupera tous ceux qui sont derrière le port et dans ce cas : quid des hubs, des bornes WiFi et autres équipements non finement manageables ;
- elle ne permet en aucun cas d'avertir l'utilisateur. Celui-ci changera donc de prise en traitant au passage les informaticiens du réseau d'incapables, de bons à rien, d'empêcheurs de travailler en rond, de fonctionnaires obtus, etc.

[5] Googlehacking: utilisation des moteurs de recherche, tel google, pour repérer en toute discrétion les applications web vulnérables, les fichiers de mots de passe en libre accès, etc.

3.2.2 La xénophobie

Cette approche ordonne à tous les serveurs, routeurs, firewalls, proxies, de refuser toute connexion provenant de la machine infectée.

On peut le faire de manière explicite ou brutale. La méthode explicite nécessitera d'empêcher la connexion au niveau applicatif. Plusieurs problèmes apparaissent alors :

- si l'applicatif est vulnérable, l'interdiction ne servira peut-être à rien ;
- cette interdiction est plus compliquée à mettre en oeuvre ;
- elle est aussi plus longue (rafraîchissement de processus).

La méthode brutale, par des filtres réseaux par exemple, est rapide et efficace. Mais elle est aussi compréhensible pour l'utilisateur qu'un manuel en japonais. Celui-ci traitera donc les informaticiens du réseau d'incapables, de bons à rien...

3.2.3 Le ghetto

Mixage des deux méthodes précédentes : il consiste à placer le port de la machine infectée dans un VLAN de quarantaine préalablement configuré. Cette technique conserve certaines contraintes :

- une excellente vision de son réseau est obligatoire ;
- les dégâts collatéraux pour ceux qui partagent le port.

Mais elle compense ceci en permettant d'avertir l'utilisateur, et surtout de faciliter l'interdiction xénophobe car celle-ci aura pu être préparée et mise en place avant l'infection.

3.3 Informer

Un utilisateur privé de réseau est forcément mécontent, et donc plus enclin à déverser des paroles peu amènes sur sa hotline.

Heureusement, un utilisateur qui se rend compte qu'il vient de faire une ânerie (parce qu'on le lui dit) est beaucoup plus réceptif.

Avertir l'utilisateur et lui expliquer la raison de son blocage est donc une bonne idée. Surtout que cela permet de lui donner des explications sur comment s'en sortir. A condition de lui laisser les billes pour le faire.

3.3.1 Réduction de voilure

L'isolement du contrevenant a pour fâcheux corollaire d'empêcher complètement l'utilisateur de se sortir seul de sa situation. Et là on touche à ma qualité première d'informaticien. Je ne veux pas travailler à sa place.

L'idée consistera donc à ne pas complètement le fermer, mais à lui laisser les outils nécessaires à sa propre désinfection.

La réduction de voilure va donc consister à autoriser l'accès à des services de désinfection en ligne, de mises à jour de système et toute autre aide à la remise en état.

Cela passe forcément par la mise en place d'un filtrage d'url par liste blanche sur un proxy transparent.

Les autres sites devront être bloqués par un message explicite, avec une notice de décontamination claire.

4 Retour en enfer

Ça y est. Il y est arrivé. Son poste est désinfecté. Mais il ne peut plus travailler. Allons-nous le laisser se prélasser dans une inactivité calme et relaxante ? Pas question ! Au turbin ! Et plus vite que ça encore....

Partant du postulat que je ne suis pas là, ou pas joignable, ou en train de faire des choses plus importantes tels que parler à mes pieds, il faut que la remise en route des accès se fasse de manière tout aussi automatique que la fermeture des accès.

Par chance, la philosophie qui dirige le traitement est « refuser l'abus ». Si nous avons estimé que dix actes d'agressions par heure étaient le seuil de déclenchement, il est évident qu'au bout d'une heure de calme, le statut d'agresseur disparaît. L'ordinateur est donc enlevé de la liste des machines à bloquer.

4.1 Ouaip, mais là ça coince.

Effectivement, il y a des points qui coincent.

Pendant qu'il est en quarantaine, il ne peut rien faire, et en particulier aucun acte d'agression. Comment va-t-on pouvoir constater sa réelle accalmie ? En notant la persistance de ses connexions illicites si cela est possible.

S'il est dans un VLAN de quarantaine, son adresse IP a changé. Comment va-t-on repérer qu'il est à nouveau calme ? Même chose que précédemment, mais en faisant le lien avec son ancienne adresse IP.

5 Le scénario est bien, mais le film ?

5.1 Le paysage local

L'université Toulouse 1 est une université de taille moyenne (16000 étudiants, 2000 personnels) en sciences sociales. L'utilisation de l'informatique y est courante mais pas vitale.

Le service informatique est composé de 23 personnes qui veillent sur près de 2500 PC, à 99% sous environnement Microsoft.

5.2 L'historique de la sécurité

Octobre 2004 a été le point de départ de la mise en place de la défense réactive, automatique et globale décrite plus haut. Les années précédentes avaient été marquées par quelques infections assez larges (10 % du parc en général), deux intrusions sur des serveurs linux « individuels ».

5.3 La détection

5.3.1 Le firewall externe

Comme décrit sur la Figure 1, le réseau est protégé par un firewall de type Linux/IPTables qui filtre les entrées et les sorties.

Un script de 30 lignes en perl[6], detection_agression.pl, suit au fil de l'eau les journaux à la recherche d'infractions intéressantes (SMTP, Netbios, mais aussi DNS, SNMP).

[6] Ce script, comme tous les autres scripts décrits dans cet article, est disponible sur simple demande.

Les informations collectées sont alors insérées dans une base de données sur le coordonnateur.

Le firewall envoie aussi les alertes de l'IDS. (Un snort dont les règles ont été particulièrement travaillées).

Enfin, il fait de même avec les machines ouvrant trop de connexions simultanément.

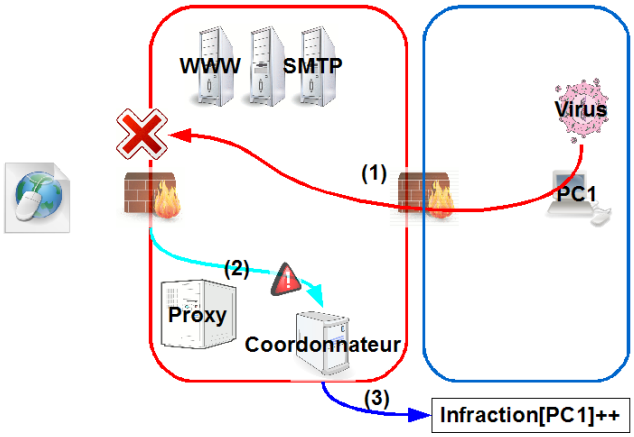

Figure 1: Détection par le firewall

5.3.2 *Le firewall interne*

Il se contente de renvoyer ses logs sur le coordonnateur. Mais il redirige aussi toute requête web vers les proxies.

5.3.3 *Les serveurs de messagerie*

Les serveurs de messagerie (des postfix) ont un « policy-daemon » (142 lignes) dédié à la surveillance des expéditeurs internes. Il étudie pour cela :

- la variation du nombre d'expéditeurs ;
- la fréquence d'envoi de mail.

Puis il décide du blocage éventuel de certaines adresses IP. Celles-ci sont alors signalées au coordonnateur.

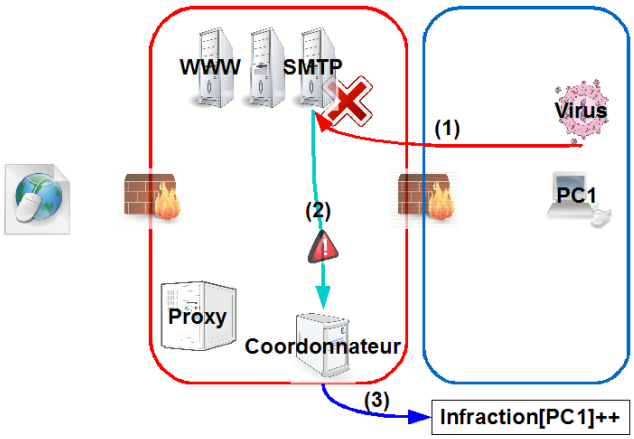

Figure 2: Détection par les serveurs applicatifs (ici SMTP)

5.3.4 *Les serveurs web*

Les serveurs web font tourner un programme (129 lignes) qui analyse les fichiers access_log en les comparant à des expressions régulières (entre 100 et 200 par serveur). Toute infraction est envoyée à la base de données sur le coordonnateur.

5.3.5 *Les serveurs proxy*

Les serveurs web, en collaboration avec les serveurs DHCP, diffusent l'information wpad.dat et wspad.dat, afin d'indiquer où sont les serveurs proxy.

Ceux-ci sont dotés d'un système de filtrage d'urls (squidguard) associé à une base libre.

Deux instances tournent : une normale, une transparente.

Le nombre de blocages n'est pas (encore) comptabilisé.

Le comptage des codes erreur 4xx et 5xx est en cours d'évaluation : pas assez d'alertes avérées pour l'instant.

5.3.6 *Le serveur DNS*

Il fait pour l'instant une journalisation intense, mais celle-ci n'est pas encore exploitée pour les mêmes raisons que les serveurs proxy : pas assez d'exemples d'infections avérées.

5.3.7 *Les autres serveurs*

Les autres serveurs envoient leurs logs directement sur le coordonnateur. Ils possèdent bien le petit script detection_agression.pl, mais il n'est pas actif.

5.3.8 *Le coordonnateur*

Le serveur coordonnateur, quant à lui, fait tourner, sur son énorme journal, le script detection_agression.pl pour remonter les évènements les plus intéressants des autres machines :

- échec d'authentification (SSH, pop, ftp, etc.) ;
- tentatives d'accès à des ports normalement non ouverts ;
- etc.

Il ajoute alors, sur sa base interne, les machines contrevenantes.

Sa base de données est constituée de 4 champs :

- IP d'origine ;
- IP destination ;
- type de l'infraction ;
- date de l'infraction.

5.4 La décision

Le coordonnateur évalue régulièrement les éléments de la table :

- toutes les 2 secondes, les infractions des cinq dernières minutes ;
- toutes les 5 minutes, les infractions de la dernière heure.

Si le test donne, pour une adresse IP d'origine interne, plus de 10 adresses IP destination (ou plus de 10 types d'infraction), la machine est alors considérée comme infectée.

On remarquera que rien n'empêche de faire ce traitement en considérant des adresses IP externes attaquantes. Il est même plus simple puisqu'alors nous ne considérons plus nécessaire la « multiplicité » des cibles : 10 infractions, et c'est la sanction.

5.5 L'action

Le coordonnateur a toutes les informations, et a surtout autorité sur tous les équipements.

Chaque machine possède un script qui décrit, pour une liste d'adresses IP donnée, les actions à faire en cas de mise en quarantaine.

Conformément à la Figure 3, cela va consister à donner les ordres suivants.

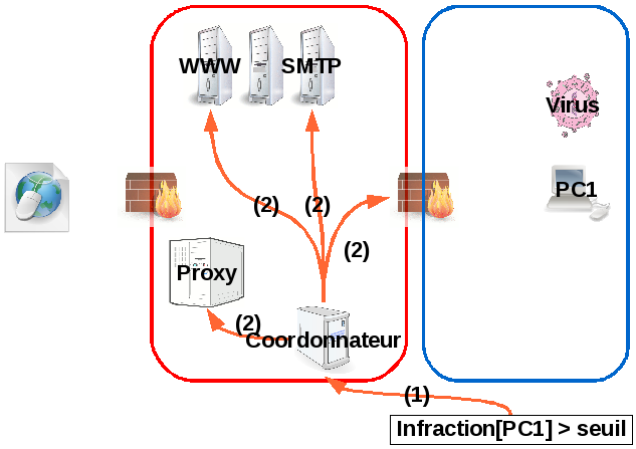

Figure 3: Lancement des ordres

5.5.1 Le firewall externe

Il bloque toute sortie de la machine.

5.5.2 Le firewall interne

Les rares exceptions dont pouvait bénéficier l'infecté sont annulées. Aucune communication n'est autorisée, hormis avec le réseau « serveurs ».

5.5.3 Les serveurs de messagerie

Aucun expédition n'est acceptée, mais le retrait de message reste possible (cf Figure 4, point 4).

5.5.4 Les serveurs web

Hormis le serveur web chargé de l'avertissement, les autres renvoient vers la page concernée.

5.5.5 Les serveurs proxies

L'infecté n'est autorisé à se connecter qu'à une liste blanche de sites de nettoyage (cf Figure 4, points 2 et 3).

5.5.6 Les autres serveurs

Toute communication est coupée, et chaque tentative est considérée comme une infraction, et envoyée comme telle dans la base de données.

5.5.7 Le coordonnateur

Il va d'abord avertir le correspondant informatique responsable du poste, en lui fournissant, si possible, le dernier expéditeur de mail connu pour le poste, afin de repérer les personnalités extérieures. Ceci est facilité par le script décrit en 5.3.3 qui stocke pour chaque IP le dernier expéditeur.

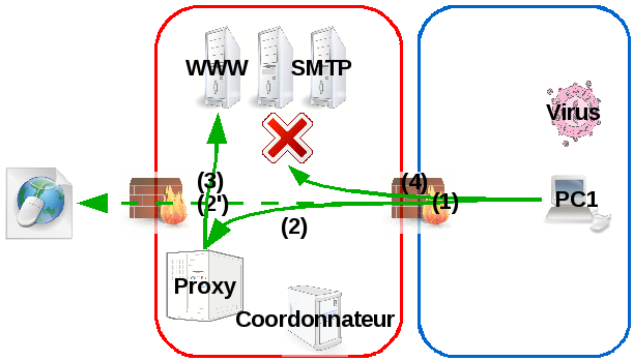

Figure 4: Blocage

Il va ensuite continuer à collecter les refus de connexion des serveurs bloqués.

5.6 Les résultats

Ils sont de plusieurs ordres.

5.6.1 Alors combien de morts ?

Depuis 2004, plus aucune infection massive n'a eu lieu : la pire n'ayant eu d'impact que sur 20 postes autour d'un serveur Microsoft.

La désinfection virale, qui occupait 25% du temps en 2004, n'atteint pas les 2% depuis.

5.6.2 L'accueil du public

Il est très bon pour nous.

La sécurité du réseau est « visible » : le blocage d'un collègue est à la limite « rassurant ».

Les personnes bloquées sont réceptives. Elles savent qu'elle viennent de faire une bêtise, donc elles sont plus calmes.

Les problèmes sont résolus plus vite : les utilisateurs appellent en sachant ce qui se passe.

On passe moins de temps à gérer ces problèmes.

Le filtrage d'url, que certains considéraient comme abusif (surtout en cas d'erreur) devient soudain plus « acceptable » quand il bloque des affichages de type adware, et par là même indique leur présence.

5.6.3 Sérieusement, vous bloquez Skype ?

Oui. Une note ministérielle est une note ministérielle. Point barre. Et dans l'ensemble, c'est assez bien compris. Nous indiquons simplement l'existence de OpenWengo.

Le blocage de Skype n'est finalement qu'une conséquence de la mise en quarantaine des postes qui initient des connexions à une fréquence trop élevée. Seront donc bloquées de la même manière toutes les applications de type P2P : les logiciels type emule, les virus qui utilisent un réseau P2P pour les mises à jour, etc.

5.6.4 Les difficultés

Elles sont nombreuses :

- le réglage des divers seuils (fréquence des mails, des connexions, etc.) est dépendant de chaque site ;
- la démarche, dans son ensemble, nécessite beaucoup d'informations, et donc mettre en place le système de collecte est assez long ;

– la création des listes d'expressions régulières est aussi une opération fastidieuse ;
– faciliter l'utilisation d'un proxy est primordial : car celui-ci fait partie intégrante du système. C'est un point qui suppose beaucoup de petites actions simples ... à condition de les connaître ;
– faire accepter par les utilisateurs ce nouvel environnement est une tâche qui dépendra de la « personnalité » de l'entité. Cela peut aller du « très simple » à « l'impossible ».

6 Alors Die Hard 5 c'est pour quand ?

La bande annonce est sortie en novembre 2006 [5]. Utilisé massivement pour bloquer les postes internes infectés, ce dispositif a été complété d'un honeypot pour bloquer les attaques externes.

6.1 Et le ghetto ?

Il est fonctionnel, mais encore en test. Les vrais/faux positifs comme pour le logiciel « ToIP P2P », bloqueraient trop de personnes.

Il n'est pas compatible avec les bornes WiFi et les hubs.

Enfin, notre capacité à désigner l'emplacement exact d'une machine n'est fonctionnel qu'à 95% : trop de commutateurs non manageables restent à changer.

6.2 D'autres fonctionnalités ?

On attend de valider la détection sur le DNS et les proxies.

Nous n'utilisons pas encore les changements IP/ARP ou les remontées de p0f[7]. L'idée de repérer un changement dans le triplet IP/ARP/OS reste cependant très séduisante.

Enfin, il est possible que nous arrivions à installer un honeypot en interne pour mieux détecter les scans internes.

7 Conclusion

La démarche est rapide, efficace et d'un coût faible. Elle n'a besoin, ni de signatures, ni de mise à jour. Elle est adaptative et automatique.

Mais elle nécessite en contrepartie un très lourd travail préalable d'installation et de configuration (travailler pour mieux fainéanter après).

Elle suppose d'avoir une bonne vision de tout son réseau, et d'avoir les droits sur la quasi-totalité des équipements.

Enfin, c'est une méthode qui fonctionne chez nous, parce que nos utilisateurs sont réceptifs à nos arguments.

Bibliographie

[1] DCSSI, La défense en profondeur appliquée aux systèmes d'information, Juillet 2004

[2] Linux Magazine : Hors Série n° 32, Août 2007

[3] Tom Vogt : Simulating and optimising worm propagation algorithms, Septembre 2003

[4] Joanna Rutkowska : Subverting Vista Kernel for Fun and Profit, Syscan'06

[5] Fabrice Prigent : Défense par diversion et quarantaine MISC, 28 : Novembre 2006

[7] P0f : outil d'identification d'OS qui fonctionne de manière passive. http://lcamtuf.coredump.cx/p0f.shtml