Quel avenir pour la sécurité Windows ?

Nicolas Ruff

EdelWeb
nicolas.ruff@edelweb.fr

1 Introduction

Les logiciels Microsoft (tels que MS-DOS et Windows 3.1) ont longtemps traîné une réputation d'instabilité chronique plutôt pittoresque (qui ne connaît pas le fameux "écran bleu" ?). Aujourd'hui les dernières générations logicielles (Windows 2000 et XP) doivent faire face à une menace beaucoup plus grande : celle de l'ouverture aux réseaux, Internet en particulier. Les défaillances logicielles ne sont plus alors de simples nuisances pour l'utilisateur, mais des failles majeures dans lesquelles s'engouffrent virus, spammeurs et pirates informatiques.

Après avoir longtemps ignoré ce problème, Microsoft se devait de réagir face à plusieurs attaques de grande envergure (historiquement Melissa et Code Red). Plusieurs annonces fortement médiatisées autour de la sécurité ont alors été faites en plus haut lieu (formation de tous les développeurs, audit de code), sans empêcher pour autant de nouvelles attaques à large diffusion (Slammer, Blaster, Klez, MyDoom, la liste est longue).

Qu'en est-il exactement et qui faut-il croire ?

Cette présentation a pour vocation de retracer un historique de la sécurité chez Microsoft, les grandes campagnes, ce qu'elles ont apportées à la sécurité Windows et les failles qu'elles ont laissées. Cet historique abordera également le présent de Windows, avec les innovations techniques majeures (pourtant mal documentées) de Windows 2003 et de Windows XP SP2.

Ayant passé en revue les imperfections de tous les mécanismes susmentionnés, cette présentation se conclura par quelques éléments de prospective sur les solutions de protection matérielles déjà annoncées et leur efficacité attendue, ainsi que les nouveaux risques auxquels ces solutions devront faire face dans un avenir très proche.

2 Un peu d'histoire

2.1 Lorsque la sécurité n'existait pas...

Le 29 juillet 1996 sort Windows NT4.

Corrigeant la plupart des défauts de jeunesse identifiés dans Windows NT 3.51, Windows NT4 allait marquer le début de la conquête par Microsoft du segment des réseaux locaux d'entreprise, jusque là partagé entre plusieurs systèmes dont principalement des systèmes Unix.

La famille Windows NT se caractérise par une architecture complètement différente des logiciels qui ont fait jusque là le succès de Microsoft (MS-DOS, Windows 3.1 et Windows 95). En effet Windows NT possède un vrai noyau 32 bits, exploitant pleinement les capacités de protection matérielle des processeurs Intel, et ne reposant plus sur une antique couche MS-DOS pour son amorçage.

A l'époque le principal avantage perçu de cette architecture est avant tout la stabilité : le dysfonctionnement d'une application utilisateur n'affecte plus le fonctionnement global du système. Bien que cette assertion soit vraie en théorie, en pratique il s'avère que les nombreux bogues de composants critiques, et surtout des pilotes s'exécutant en mode noyau, font de NT4 un système qui ne tient pas toutes ses promesses. D'ailleurs d'après Microsoft, 90% des "écrans bleus" sont provoqués par des pilotes d'affichage.

Cette nouvelle architecture présente également un intérêt non négligeable du point de vue de la sécurité. La principale préoccupation de l'époque étant les virus, il est vrai que le système NT4 leur porte un coup dur : absence de système DOS pour lequel sont conçus la majorité des virus (avec les techniques bien connues "Interrupt Hijacking" et "Terminate and Stay Resident"), secteur d'amorçage exotique, impossibilité d'accéder directement au disque dur, complexité du format PE, etc. Les concepteurs de virus se tournent alors vers les virus "macro", qui présentent l'avantage d'être multi plateformes.

Toutefois les fonctions réseau avancées et les mécanismes de sécurité robustes[1] de Windows NT4 intéressent des groupes de hackers, tels le célèbre L0pht. Dès lors les premières attaques commencent à apparaître sur Internet : attaque en rejeu sur l'aléa du protocole LM, outil de cassage de mots de passe L0phtCrack, attaque Red Button permettant l'énumération des comptes de manière anonyme....

Face à la grogne des clients et à la multiplication des alertes, Microsoft se voit contraint de réagir et inaugure le 1er bulletin de sécurité le 1er juin 1998 (MS98-001 : "*Disabling Creation of Local Groups on a Domain by Non-Administrative Users*").

Dès lors la question de la sécurité des logiciels va aller croissant, face à des incidents de plus en plus nombreux et étendus (qui vont de pair avec la croissance du réseau mondial).

- 26 mars 1999
  - Propagation mondiale de Melissa (macro-virus Word).
- 19 juillet 2001
  - Propagation du ver Code Red sur les serveurs Web IIS 5.
- 25 janvier 2003
  - Propagation du ver Slammer sur SQL Server et MSDE. De nombreuses infrastructures sensibles sont perturbées.
- 12 août 2003
  - Propagation du ver Blaster, qui exploite une faille RPC présente dans toutes les versions de Windows. Un grand nombre de "home users" sont

[1] Par opposition à Windows 9x.

affectés. Chacun des vers ci-dessus exploite une faille pour laquelle un correctif est disponible depuis plusieurs mois ?

Le nombre de bulletins de sécurité émis annuellement par Microsoft depuis 1999 se situe entre 60 et 100, et ce malgré des efforts notables pour diminuer cet indicateur devenu une valeur de référence dans la compétition entre éditeurs.

2.2 Round 1 : "Microsoft Security Initiative"

Code Red et Slammer sont les plus illustres représentant de la classe de vulnérabilités la plus répandue : le débordement de tampon alias "buffer overflow". Ce type de vulnérabilité est rendu possible par la conjonction de deux facteurs : un langage de programmation bas niveau (ici le C) laissant au développeur le soin de dimensionner ses variables, et un manque de sensibilisation des développeurs à une pratique de programmation "sécurisée". A la décharge de ces pauvres développeurs, il faut bien admettre qu'une grande partie du code des systèmes actuels a été écrit à une époque où le concept même de "buffer overflow" n'avait pas été découvert.

Face à ce constat, Bill Gates annonce personnellement les moyens de réponse que met en oeuvre Microsoft :

- Un arrêt temporaire (2 mois) des nouveaux développements, au profit d'un audit global du code existant. Des outils automatisés, tels que PREFix et PREFast, sont développés en interne par Microsoft à cette fin.
- Une formation des développeurs aux problèmes de sécurité, reprenant l'excellent ouvrage "Writing Secure Code" de Michael Howard, aux éditions Microsoft Press.

Le coût estimé par Microsoft de cette opération est de 200 millions de dollars (une broutille comparativement au chiffre d'affaire de la société).

Au final, il est indéniable que cette opération a profité à la qualité du code source de Windows, compte tenu du nombre de bogues documentés a posteriori et corrigés silencieusement par Microsoft. Le nombre de bogues corrigés lors de cette opération n'a pas été annoncé officiellement par Microsoft, quant au nombre de bogues qui ont échappé à la correction ? il est par définition impossible à connaître !

Ce qui est sûr, c'est qu'au mois de juillet 2003 le groupe polonais LSD annonçait la découverte d'un "buffer overflow" trivial dans le service RPC, avec pour conséquence la propagation du ver Blaster. Ce bogue a été identifié par rétro-ingénierie sans accès au code source.

Cette démarche montre donc ses limites sur un code complexe (plus de 30 millions de lignes de code), dont la conception initiale remonte à plus de 10 ans.

2.3 Round 2 : "Get Secure, Stay Secure"

Malgré l'amélioration de la qualité du code source et la correction (souvent silencieuse) de nombreux bogues, le parc Windows déployé reste attaqué avec succès partout de par le monde. Microsoft identifie deux causes à ce problème rédhibitoire :

"*La configuration par défaut du système n'est pas sécurisée.*"

Cette configuration a été conçue pour fonctionner dans tous les environnements (professionnel et domestique) et maximiser la fonctionnalité ; or le besoin de rétro-compatibilité en entreprise impose l'activation d'options archaïques et dangereuses (tels que les sessions nulles sur SMB). On se souvient que Code Red s'est propagé de manière importante car le serveur Web IIS était installé par défaut avec Windows 2000 Serveur, et que le gestionnaire d'extensions ".IDA" était actif par défaut.

Au final, et d'après le "top 10" du SANS, l'attaque la plus souvent utilisée contre un Windows 2000 consiste à se connecter à distance via un compte administrateur local sans mot de passe ?

Windows XP innove donc en la matière, avec une surface d'attaque réduite par rapport à Windows 2000, et une configuration adaptée à chaque utilisation. Ainsi sur un poste en "Workgroup", il n'est pas possible de se connecter via le réseau avec un mot de passe vide, et les sessions nulles sont interdites.

Le concept n'a toutefois pas été poussé jusqu'au bout, puisque le nouveau service UPnP était démarré par défaut ? et rapidement identifié comme vulnérable à un "buffer overflow" !

Windows 2003 Server diminue encore la surface d'attaque par rapport à ses prédécesseurs : chaque applicatif système doit être installé individuellement, en fonction des besoins.

Philosophiquement il est intéressant de constater que les applicatifs système de Windows 2003 ne sont pas forcément moins bogués, mais font l'objet de moins de tentatives d'attaque à partir du moment où la base installée est plus faible. Après tout, quitte à investir du temps dans la recherche de nouvelles attaques, autant maximiser sa portée potentielle ?

Pour les entreprises, Microsoft publie également à titre gracieux des guides de sécurisation, destinés à remplacer les guides "non officiels" fleurissant un peu partout sur Internet (NSA, SANS...).

On peut reprocher à ces guides d'être volumineux (plus de 400 pages pour le guide Windows 2000), et de nécessiter un gros effort d'interprétation et d'adaptation au contexte de l'entreprise (pas toujours évident avec des guides rédigés majoritairement en anglais). Toutefois ces guides constituent indéniablement une mine d'informations pour les personnes désireuses de s'impliquer dans la sécurité de leur parc Microsoft.

Parallèlement des outils gratuits tels que MBSA assistent l'utilisateur (nécessairement éclairé malgré l'effort de pédagogie du produit) dans la configuration sécurisée de son poste de travail.

Malgré tout, des vulnérabilités restent présentes dans le coeur historique (et archaïque) du système, celui qui échappe à toute démarche de sécurisation proactive. Le meilleur exemple est le ver Blaster, utilisant une faille dans les services RPC. Désactiver un tel service dans Windows est suicidaire, et les options de configuration peu nombreuses ?

"*Les utilisateurs n'appliquent pas les correctifs de sécurité.*"

Effectivement, les efforts de correction de code mettent du temps à se répercuter sur la base installée, y compris en entreprise où les modifications du système d'information sont rarement préventives.

Or la principale cause de propagation des vers est bien la non application des correctifs de sécurité, on citera en particulier Slammer qui est apparu un an après le correctif adéquat ?

On notera au passage que Microsoft a nié publiquement[2] l'utilisation de "0-day"[3] pour Windows, alors que plusieurs exemples patents ont été documenté tels que la compromission d'un site militaire américain via la faille WebDAV, ou la propagation quotidienne de virus via des failles IE.

Toujours est-il que Microsoft s'est concentré sur la mise à jour de ses produits : activation par défaut de Windows Update dans Windows X, ; outils gratuits de gestion des correctifs (MSUS, HFNetChk), envoi de CDs gratuits sur simple demande, changement de la politique de publication (deuxième mardi de chaque mois) afin de permettre aux entreprises de s'organiser.

Le bilan que l'on peut tirer de cette initiative est mitigé : les entreprises qui se sont dotées de moyens de gestion des correctifs arrivent à obtenir des résultats, avec néanmoins les limites suivantes :

- Gestion du nomadisme.
- Gestion des masters (intégration difficile des correctifs dans les images d'installation, suivi de version et compatibilité applicative),
- Gestion de parc (identification des postes concernés et déploiement massif, problème du retour arrière et des postes non gérés).
- Continuité de service (qualification des correctifs, compatibilité applicative et problème du redémarrage).

En ce qui concerne les PME et les utilisateurs domestiques, voire certains grands groupes, tout reste à faire. La meilleure preuve en est la propagation fulgurante du ver Blaster en août 2003. De plus les outils gratuits disponibles chez Microsoft (Windows Update, MSUS) ne gèrent que la plateforme Windows / IE et non les autres applicatifs Microsoft, d'où la propagation du ver Slammer sur SQL Server et MSDE. A la décharge des utilisateurs, il faut également admettre que les virus exploitent aujourd'hui des failles IE plus rapidement qu'elles ne sont corrigées. Il semble toutefois que 2004 s'annonce comme l'année du "patch management", avec du côté de Microsoft la sortie du nouveau "Microsoft Update".

2.4 Round 3 : "La défense périmétrique"

Puisque la sécurisation du coeur historique de Windows semble vouée à l'échec, et qu'il existera toujours une majorité d'irréductibles utilisateurs qui n'appliqueront pas les correctifs de sécurité et les guides de configuration, la nouvelle approche de Microsoft dans le SP2 de Windows XP consiste à isoler complètement l'utilisateur du réseau, en plus des corrections sensibles apportées

[2] We have never had vulnerabilities exploited before the patch was known `http://news.co.uk/1/hi/technology/3485972.stm`

[3] Vulnérabilité non documentée et *a fortiori* non patchée.

aux principaux vecteurs d'attaque du système (Internet Explorer, Outlook Express, RPC, COM/DCOM...).

Cette approche radicale a des conséquences non négligeables sur la compatibilité applicative, comme nous le verrons plus loin dans la présentation du SP2.

Il ne faut pas confondre défense périmétrique et défense en profondeur, un autre concept hérité du domaine militaire, applicable aux systèmes d'information.

La défense en profondeur consiste à opposer aux attaquants plusieurs lignes de défense successives. On peut citer par exemple la séparation des privilèges dans le démon SSH et la notion de "prison" (jail) obtenue grâce à la commande chroot. Lorsque ces mécanismes sont en place, la compromission d'un service réseau ne permet pas d'obtenir immédiatement les droits les plus élevés sur le système. Or actuellement avec Windows XP, même SP2, l'utilisateur reste administrateur local du poste et de nombreux services réseau sensibles (tels que RPC) tournent sous le compte SYSTEM ?

Avec une défense périmétrique, la moindre brèche permet de compromettre complètement le système.

3 Les nouveautés majeures

3.1 Windows XP/2003 : les révolutions silencieuses

Protection de pile Lors de la publication en juillet 2003 de la faille DCOM/RPC ayant donné lieu au ver Blaster, aucun code d'exploitation pour Windows 2003 n'a été rendu public alors que ce système était annoncé comme vulnérable par le groupe LSD et Microsoft.

A contrario le bogue MS04-006 dans le service WINS provoque un déni de service sur Windows 2003 uniquement, alors que toutes les versions de Windows sont annoncées comme vulnérables.

La raison technique profonde de ces comportements est une nouveauté introduite dans le compilateur Visual Studio.NET via l'option de compilation /GS. Il s'agit d'un mécanisme de protection de pile à l'exécution (runtime), basé sur l'insertion d'une valeur aléatoire en pile (appelée canary ou cookie) avant tout appel de fonction manipulant des buffers.

Ce mécanisme n'est pas nouveau, il existe sous Unix dans le produit StackGuard par exemple. Visual Studio 6 intègre également une protection de ce type, via les options /GZ et /RTC, mais celles-ci impactent trop les performances pour être utilisées en mode "Release" (elles avaient été pensées dans un but de déboguage uniquement).

La nouveauté est que l'ensemble de Windows 2003 Server, y compris le code "hérité" (de type WINS, RPC) et IIS 6.0, a été compilé avec cette option, ce qui le rend beaucoup plus robuste vis-à-vis des possibilités d'exploitation malveillante en cas de débordement de buffer. (Cette technologie n'élimine pas les débordements de buffer, mais bien la possibilité de les exploiter dans la plupart des cas).

D'autres produits majeurs, tels que le ".NET Framework" ou Office 2003, ont également été compilés avec cette option. Microsoft est tellement satisfait des premiers retours d'expérience que l'ensemble du SP2 pour Windows XP sera également recompilé avec /GS.

On notera que le projet Fedora (version communautaire de Red Hat Linux) a également été compilé en grande partie avec la technologie "exec-shield".

Gestionnaire d'exceptions Sans rentrer dans les détails, une technique d'exploitation des débordements de buffer bien connue et mise en ?uvre par Code Red en 2001 consiste à modifier l'adresse du gestionnaire d'exception (située en pile). Cette technique permet de s'affranchir de la protection /GS vue précédemment.

Face à cette menace, plusieurs nouveautés concernant le traitement des exceptions ont été introduites dans Windows XP :

- Le *dispatcher* d'exceptions refuse de transférer le contrôle à un gestionnaire, si celui-ci réside dans la pile.
- Les registres EAX, EBX, ESI et EDI sont effacés avant le traitement de l'exception, afin d'éviter la fuite d'adresses mémoire.
- Les gestionnaires d'exception peuvent être déclarés dans une section spécifique du format de fichier exécutable PE. Aucun autre gestionnaire ne peut alors être appelé par le programme à l'exécution. Cette technologie n'est quasiment jamais utilisée en pratique, car la plupart des gestionnaires d'exception sont dynamiques.

.NET Framework A partir de Windows 2003, le ".NET Framework" est installé en standard avec le système. Certains composants de Windows 2003 sont des assemblies ".NET".

Par conception, et sauf bogue majeur dans le Framework, les applications ".NET" sont immunisées aux problèmes de débordement de buffer pour deux raisons :

- Il s'agit d'un langage interprété, les assemblies contenant en fait du "pseudo-code". Le Framework effectue des contrôles forts à l'exécution.
- Les types tableau ne sont pas gérés par le développeur mais par le langage (la notion de pointeur n'existe pas).

Options de sécurité De nombreuses options de sécurité ont été ajoutées dans Windows XP, et ces options sont activées par défaut. Les mêmes options ont été reprises dans Windows 2003. Parmi les plus significatives on notera :

- Les comptes avec mot de passe vide ne peuvent pas être utilisés pour des opérations réseau (mode workgroup uniquement).
  - L'accès au partage administratif `C$` avec le compte administrateur local et un mot de passe vide a longtemps été l'une des failles de conception les plus exploitées de Windows 2000 (cf. Top 10 du SANS).
- L'utilisateur "anonyme" n'appartient plus au groupe "tout le monde".

  - Ainsi les connexions anonymes ne peuvent pas être utilisées pour se connecter aux partages réseau, dont les permissions par défaut sont "tout le monde : lecture seule".
- La permission par défaut sur les nouveaux partages n'est plus "tout le monde : contrôle total" mais "tout le monde : lecture seule".
- L'énumération via une connexion anonyme des comptes et groupes est désactivée dans la configuration par défaut.
  - Cette option est présente depuis Windows NT4 SP4, mais rarement activée auparavant.

3.2 Windows XP : les nouveautés du SP2

Introduction Avec la sortie du SP2 pour Windows XP (prévu pour cet été), Microsoft compte frapper un grand coup. Ce Service Pack ne corrige pas seulement des bogues, mais ajoute également de nombreuses fonctions de sécurité, et active par défaut des options de sécurité généralement inutilisées. L'esprit de ce Service Pack est de bloquer à la source toutes les techniques utilisées jusqu'ici pour attaquer un système Windows. Par exemple :

- Vers de type Code Red, Slammer, Blaster.
- Firewall intégré en mode "deny all" sur les connexions entrantes.
- Spam via le service d'affichage des messages.
- Désactivation des services "alerter" et "messenger" par défaut.
- Bogues IE non corrigés.
- Restriction de la zone poste de travail (cf. outil QwikFix), blocage des popups....

Le document décrivant les nouveautés du SP2 fait à lui seul plus de 150 pages, on peut le trouver à l'adresse suivante : `http://www.microsoft.com/technet/prodtechnol/winxppro/maintain/winxpsp2.mspx`

La "liste à la Prévert" de ces nouveautés est la suivante (il est bien entendu impossible de rentrer dans les détails ici, le lecteur curieux est invité à se reporter au document Microsoft et aux analyses tierces publiées sur Internet) :

- Protection réseau
  - Services "Alerter" et "Messenger" désactivés par défaut.
  - Support Bluetooth natif.
  - Ajout des fonctions "rechercher..." et "sélectionner des utilisateurs, des ordinateurs ou des groupes" aux outils d'administration
  - Restrictions COM / DCOM / RPC
    - Désactivation de l'accès anonyme par défaut.
    - Journalisation accrue.
    - Granularité des permissions accrue.
  - Redirecteur WebDAV.
    - "Basic auth" interdit sur HTTP.
- Support du flag NX.
  - Processeurs AMD64 et Itanium uniquement.
  - Désactivable dans le panneau de configuration (globalement ou par application).

- API "AES" (Attachment Execution Service).
  - Point d'entrée pour un filtrage antivirus.
  - Commune à IE / OE / Messenger.
- Windows Messenger.
  - Blocage des fichiers "dangereux" envoyés par des inconnus.
  - Nickname obligatoirement différent de l'adresse email.
- Outlook Express.
  - Lecture en texte par défaut (rendu RTF au lieu de HTML).
  - Pas de téléchargement du contenu HTML externe.
- Wireless Provising Service.
  - Envoi de paramètres de configuration par les hotspots...
  - Le Wireless Network Registration Wizard permet de donner son numéro de carte bleue aux opérateurs WiFi OEM...
- Windows Media Player.
  - Installation obligatoire de Media Player 9.
- Maintenance
  - Les correctifs de sécurité n'apparaissent plus dans ajout/suppression de programme.
  - Client "Windows Update v5" / "Microsoft Update".
  - Calcul du RSoP.
  - "Security Center"
    - Alerte l'utilisateur sur les fonctions de sécurité suivantes : antivirus, firewall, mises à jour.
  - Windows Installer 3.0

Les deux composants les plus fondamentalement modifiés par le SP2 sont ICF (le Firewall intégré de Windows XP) et IE.

ICF Le produit ICF devient un Firewall personnel à part entière, capable de rivaliser avec des produits du commerce sauf sur un point (et non des moindres) : ICF est incapable de filtrer les connexions sortantes.

En revanche il bénéficie logiquement d'une meilleure intégration avec Windows que ses concurrents, et peut être configuré soit par script, soit de manière centralisée par GPO.

La principale nouveauté est la configuration par défaut en mode "deny all" sur les connexions entrantes pour les postes en Workgroup. Lorsque le poste est joint à un domaine, seul le sous-réseau local est autorisé à établir des connexions entrantes.

ICF intègre également d'une API permettant aux applications "ICF-aware" d'ouvrir dynamiquement des ports. Cette API comprend des fonctions telles que *INetFwAuthorizedApplication* (s'ajouter à la "white list"), *INetFwOpenPort* (ouvrir un port en écoute), *INetFwProfile* (modifier les paramètres globaux du Firewall).

Ceci représente bien évidemment une voie royale pour le code malveillant, aussi quelques restrictions ont-elles été apportées par Microsoft, dont les suivantes :

- Seuls les processus exécutés sous les comptes LocalSystem, LocalService ou NetworkService pour accéder à l'API *INetFwOpenPort*
- Dans tous les cas, SVCHOST ne peut pas accéder à l'API.

Enfin des problématiques spécifiques ont été traitées avec des "astuces" telles que :

- Traitement spécial des RPC et du problème des ports dynamiques : la clé *PrivilegedRpcServerPermission* permet de lister les applications hors "white list" qui peuvent néanmoins effectuer des appels RPC.
- Traitement des requêtes UDP : la réponse est attendue pendant 90 secondes, avant fermeture du port.
- Traitement des broadcasts et des multicasts UDP : la réponse est attendue pendant 3 secondes avant fermeture du port.

Par défaut, ICF bloque efficacement les attaques réseau "frontales" contre un utilisateur domestique en Workgroup. Il s'avèrera probablement inefficace contre du code malveillant exécuté directement sur le poste de l'utilisateur (ouvrant par exemple une porte dérobée), et les nombreuses possibilités de contournements prévues par conception laissent entrevoir des scénarios d'attaque potentiels.

IE Les nouveautés dans IE sont également très nombreuses et vont bien au-delà de la simple correction de bogues. Rien de révolutionnaire toutefois, puisque Microsoft intègre principalement des technologies déjà éprouvées (ex. outil QwikFix, blocage des popups).

Parmi les principales nouveautés on peut citer :

- La barre d'information, affichant les alertes de sécurité pour la page en cours.
- Une seule popup de sécurité par page et non plus par composant.
  - Popup de sécurité = popup de type "voulez-vous exécuter les contrôles ActiveX sur cette page ?"
- Gestion facilitée des "add-ons". Les add-ons (souvent utilisés par des codes malveillants de type Spyware pour s'installer de manière résidente) comprennent :
  - Les plug-ins de navigation (de type "Google Bar").
  - Les "binary behaviors" (extensions de rendu HTML).
- Blocage des popups.
- Nouvelles "Feature Control" (options de sécurité).
  - "MIME sniffing" : reconnaissance des types de fichier par signature et pas par extension. La base de signatures utilisée, et les moyens de mise à jour, restent inconnus à ce jour.
  - Pas d'exécution dans un contexte plus privilégié que l'URL de base.
  - Limites sur la création et le déplacement des fenêtres par script (permet d'éviter les fenêtres masquées ou affichées hors écran).

Conclusion Il est difficile de tirer des conclusions définitives sur le SP2 de Windows XP car à la date de rédaction de cet article, seule une RC1 est disponible.

On notera toutefois un gros effort d'amélioration de la sécurité Windows de la part de Microsoft, salué par la presse spécialisée américaine.

De nombreuses critiques ont également été formulées par des experts reconnus lors de la conférence RSA 2004, au cours de laquelle Bill Gates en personne s'est déplacé pour présenter le SP2. Ces critiques sont les suivantes :

- Ajout de fonctionnalités (y compris de sécurité) = ajout de failles, le nombre de défauts dans le code étant directement proportionnel à la taille du code.
- Pas de changements fondamentaux dans l'architecture Windows, qui conserve les protocoles RPC, DCOM, SMB... et toutes les failles attenantes (ports dynamiques, accès anonymes, configuration complexe...).
- Fonctionnement en mode "pompier" : les protections apportées se basent sur les failles exploitées, pas sur les failles exploitables. Par exemple rien n'est prévu à l'heure actuelle pour lutter contre l'"API Hijacking", car cette technique commence à peine à être utilisée par les codes malveillants. Ainsi Bill Gates a raison en disant que "les virus et les hackers rendent Windows plus robuste"[4].
- La nouvelle configuration par défaut est très orientée "utilisateurs domestiques", mais inadaptée pour une entreprise.
- Microsoft concurrence clairement des produits commerciaux, malgré les nombreux rappels à l'ordre de la justice pour abus de position dominante. A l'heure actuelle il n'est pas prévu d'intégrer un antivirus dans le SP2, mais on se souviendra que Microsoft a racheté la société GeCAD et son produit antivirus RAV.
- Le gain sera nul si l'impact sur les applications est trop fort, car toutes les nouvelles fonctions de sécurité seront alors désactivées par les administrateurs pressés ou par les éditeurs d'applications.

En conséquence, on peut prévoir à la date de rédaction de cet article et sur la base de la version RC1 que le déploiement du SP2 s'annonce problématique. Il est impératif d'effectuer des tests de compatibilité exhaustifs avant tout déploiement massif et de prévoir la désinstallation du SP2. Le gain en sécurité obtenu par l'installation du SP2 est néanmoins important et justifie ces contraintes.

4 Vers une informatique de confiance ?

Personne ne sort réellement victorieux de la guerre entre l'attaque et la défense des systèmes Windows. A chaque nouvelle annonce de Microsoft succède une nouvelle classe d'attaque ou une nouvelle épidémie virale, qui remet en question les efforts entrepris.

Résultats : la faillibilité du système se banalise, la confiance des utilisateurs s'érode et le développement de la société numérique s'en trouve retardé, sans parler des conséquences économiques immédiates pour les entreprises infectées ou obligées d'investir une part importante de leur budget informatique dans le maintient de l'existant.

[4] `http://www.theregister.co.uk/content/55/35145.html`

Face à cette situation, voyons maintenant quelles sont les pistes explorées actuellement dans le domaine de la protection des systèmes Windows, censées donner un avantage durable à la défense sur l'attaque.

4.1 Prochaines pistes de recherche

Visual Studio 2005, nom de code "Whidbey" La prochaine version du compilateur phare de Microsoft devrait intégrer, sous une forme qui reste à définir, des technologies permettant (enfin) de développer facilement des applications respectant le principe du moindre privilège, c'est-à-dire ne nécessitant pas que l'utilisateur soit administrateur du poste.

Ceci serait bien évidemment une avancée majeure dans le domaine de la sécurité Windows, puisque pour le moment les applications "héritées" et les services nécessitent des privilèges maximum (administrateur ou SYSTEM) pour fonctionner.

Analyse de code source Dans le cadre de la démarche "Microsoft Security Initiative", Microsoft a développé en interne des outils d'audit de code appelés PREfast et PREfix.

Ces outils, déjà disponibles dans le DDK Windows 2003, seront probablement intégrés en standard dans la prochaine version du compilateur Visual Studio.

D'autre part Microsoft confirme sa volonté d'ouvrir de plus en plus largement son code source aux gouvernements, afin sans doute de ne pas se laisser distancer par son concurrent Linux sur l'important marché des administrations.

Aux Etats-Unis où le gouvernement durcit ses exigences de certification Critères Communs pour les fournisseurs de logiciels, Microsoft va sans doute être obligé de re-certifier Windows à chaque nouveau Service Pack.

Tout ceci concourre à une probable amélioration de la qualité générale du code source, bien que le nombre de vulnérabilités triviales identifiées ces derniers temps par des tiers n'ayant pas accès au code source puisse laisser dubitatif ...

Autres idées On retiendra également les idées suivantes, issues du monde Unix ou du Logiciel Libre, qui pourraient présenter un intérêt certain appliquées au monde Windows. Toutefois à ma connaissance Microsoft ne travaille pas officiellement sur ces sujets.

- *Protection de type PaX / GRSecurity.*- Bien que le support matériel des pages non exécutables soit intégré dans Windows XP SP2, cette protection nécessite un processeur AMD 64 ou Itanium. Compte tenu de la vitesse de renouvellement du parc matériel, de nombreux clients ne pourront pas bénéficier de cette protection avant un certain temps. Or des solutions de protection logicielle existent, implémentées sous Windows dans les produits peu connus SecureStack et Overflow Guard.
  GRSecurity intègre également des protections simples à mettre en ?uvre telles que l'allocation d'adresses aléatoires pour la pile et les exécutables.

L'efficacité de ces protections contre les attaques simples n'est plus à démontrer.

- *Séparation des privilèges.*- Bien que Windows dispose d'un mécanisme d'impersonation via les API *Impersonate*()* et *RevertToSelf()* qui n'a rien à envier au *setuid()* Unix, ce mécanisme est trop peu utilisé par les développeurs, y compris chez Microsoft. Il s'agit là à mon avis d'une "culture sécurité" différente entre les développeurs Windows et Unix.
  La plus belle illustration de ce risque est une Shatter Attack élégante qui a été documentée récemment sur Internet :
  - En tant qu'utilisateur non privilégié, introduire un virus quelconque sur un poste : le service antivirus se déclenche et affiche une "popup".
  - Faire F1 pour obtenir l'aide.
  - Ouvrir le menu "Aller à l'URL..." et désigner "CMD.EXE".
  - CMD est exécuté avec les privilèges de son parent, c'est-à-dire SYSTEM.
- *Chroot.*- Il n'existe pas aujourd'hui de technologie fiable sous Windows permettant de limiter la visibilité du système de fichiers pour un processus donné.
  C'est ainsi que la première recommandation de tout guide d'installation IIS est de mettre la racine Web sur un disque différent du disque système, afin de limiter les conséquences d'une attaque en Directory Traversal, ce qui reste avouons le un "bricolage".
  Une telle technologie trouverait de nombreuses applications pratiques, par exemple dans les fermes de Terminal Server. A l'heure actuelle les "masquages" de disques ou de répertoires sont basées sur la "bonne volonté" de l'explorateur de fichiers, qui effectue lui-même les contrôles d'accès. N'importe quel explorateur tiers se rie de ces restrictions.

4.2 Sécurisation de l'environnement

Malgré toutes les protections vues précédemment et les efforts de R&D considérables, les utilisateurs continuent de cliquer sur les pièces jointes et de télécharger du contenu pirate sur Kazaa.

Windows reste donc un système exposé et vulnérable, principalement sur le poste client. Ce risque est particulièrement accru par la mobilité des postes (PC portables connectés sur des réseaux domestiques ou WiFi) et des données (clés USB, interfaces avec des téléphones portables).

Partant du constat que le poste client représente un risque, le concept de "défense en profondeur" fait son chemin dans les schémas de protection d'entreprise, et des solutions de contrôle externe commencent à apparaître.

On peut citer les annonces Cisco et Checkpoint sur des équipements réseau "intelligents" qui vérifient la configuration des postes et en particulier la mise à jour de l'antivirus. De son côté Microsoft n'est pas en reste avec une technologie de quarantaine des nomades native dans Windows 2003 Server.

Cette implémentation reste sommaire, puisque basée sur l'exécution d'un script côté client qui renvoie au service de quarantaine une valeur booléenne

indiquant la conformité du poste par rapport aux paramètres testés. Ces technologies représentent néanmoins un axe de développement futur important.

4.3 Sécurité matérielle

Le support du drapeau NX dans les processeurs IA-64 et AMD 64 démontre une collaboration étroite entre Microsoft et les principaux fondeurs de microprocesseurs, indispensable au succès des opérations futures que sont TCPA et NGSCB.

Intel a déjà annoncé sa volonté d'intégrer nativement un support TCPA dans ses processeurs. Reste l'inconnue NGSCB, dont le contour reste pour l'instant flou malgré les premières présentations publiques de Microsoft.

Sans rentrer dans la polémique, la date de pénétration sur le marché de ces technologies, le comportement des acheteurs et le gain réel en sécurité pour l'utilisateur final sont bien mal cernés aujourd'hui.

4.4 Les nouveaux risques

Dans ce dernier chapitre, je vais m'exercer au jeu dangereux de la prospective. En effet l'explosion des technologies va de pair avec l'apparition de nouveaux risques pour lesquels personne ne se pose encore la question de protections éventuelles, jusqu'à la première catastrophe. Parmi tous ces risques j'ai choisi de mettre l'accent sur les suivants :

- *La mobilité.*- Bien que le problème des postes nomades commence à être pris en compte après des coups durs (vers Slammer et Blaster entre autres), de nombreux autres aspects de la mobilité du code sont totalement occultés.
  L'exemple le plus frappant est l'explosion des applications pour téléphones portables grâce à la standardisation de la plateforme Java (MIDP). Les capacités des nouveaux téléphones en font des nomades à part entière (augmentation de la capacité de stockage à plusieurs Mo, connectivité Internet via GPRS, synchronisation automatique avec Outlook), quand ils ne sont pas directement équipés de Windows CE.
- *Intégration de plus en forte d'Internet dans les applications Microsoft.*- Il est indéniable qu'une grande partie de la stratégie de Microsoft tourne autour de l'osmose entre le poste de travail et les serveurs Microsoft : activation des produits, aide en ligne, compte Passport, aide et cliparts Office 2003 en ligne...
  Une connexion Internet rapide et permanente devient quasiment un prérequis pour pouvoir bénéficier pleinement des produits Microsoft. Par voie de conséquence, le poste est également de plus en plus exposé aux attaques provenant d'Internet.
- *Encapsulation des flux.*- Parallèlement à l'intégration très forte de la composante Internet dans les produits Microsoft, on notera une fâcheuse tendance à encapsuler les flux sur HTTP y compris à travers des extensions non normatives de ce protocole.

Compte tenu de l'absence de Firewall applicatif intégrant ces extensions (sauf dans le produit Microsoft ISA Server), il devient difficile de filtrer efficacement les flux entre les postes clients du réseau interne et Internet. L'exemple le plus critique est l'encapsulation RPC sur HTTP (utilisé par OWA), compte tenu des nombreux bogues découverts dans les services RPC ces derniers temps.

– *Chevaux de Troie furtifs.*- Pour faire face au développement des logiciels de protection de type Firewall personnel, tout en répondant à la demande en matière de marketing en ligne, de plus en plus d'éditeurs de Spyware (voire de créateurs de virus) intègrent des techniques jusque là considérées comme offensives ("API Hijacking", masquage de processus, injection dans des processus autorisés...) - voir la longue liste des "Trojan.Downloader.xx" référencés chez les éditeurs d'antivirus.
Il existe encore peu d'outils de protection efficaces contre ces "malwares" qui sont proches de "rootkits", donc difficiles à détecter par définition. Seule une analyse "à froid" du disque permet de conclure de manière fiable sur l'infection d'un poste.
– *Les risques applicatifs.*- Avec l'explosion des services Web, les bogues applicatifs "classiques" (injection SQL, cross-site scripting, etc.) deviennent une réelle menace et l'une des premières causes de "défiguration" (defacement) de sites. Cette menace va aller croissante compte tenu de l'extension des technologies Web (XML, SOAP, ASP.NET, etc.).

Aucune des technologies précédentes (de type protection logicielle ou matérielle contre les "buffer overflow") ne protège contre ces risques, puisque nous sommes en présence de langages interprétés (Java, .NET, XML, scripts, etc.) et que le défaut se situe dans la logique intrinsèque de l'application - chose beaucoup plus difficile à détecter pour un outil.

5 Conclusion

Depuis quelques années, la sécurité logicielle est devenue un enjeu majeur dans un monde de plus en plus connecté. Windows, souffrant de tares de conception et d'une communauté d'utilisateurs peu sensibilisés aux problèmes de sécurité, a été victimes de nombreuses attaques fortement médiatisées compte tenu de sa pénétration importante du marché.

Face à cet état de fait, Microsoft a lancé plusieurs grands chantiers de sécurisation du système, dont le dernier en date (le SP2 pour Windows XP) va sortir cet été. Les bénéfices de chantiers précédents sont techniquement sensibles, mais n'ont eu qu'un faible impact pour l'utilisateur final submergé par les attaques de grande envergure qui continuent à se succéder (ver Blaster, virus Mimail, Bagle et Netsky).

Bien qu'un réel effort de sécurisation soit entrepris, l'accroissement des fonctionnalités et de la connectivité des systèmes Windows augmente parallèlement les risques.

L'équation à résoudre n'est pas simple puisque le système Windows est à la fois :

- Versatile :
  - Produits quasiment identiques en versions grand public, postes de travail, serveurs Web, contrôleurs de domaine...
- Intégré :
  - Les applications et les services exigent des privilèges important pour fonctionner.
  - La frontière entre le système et les applications est très floue.
  - La frontière entre le système et Internet de plus en plus également.
- Ouvert :
  - Ce système est leader du marché.
  - La diversité applicative énorme.
  - La communauté des développeurs n'est pas sensibilisée aux problèmes de sécurité.
- Hérité :
  - Le besoin de compatibilité protocolaire impose l'utilisation de protocoles peu sûrs.
  - Le besoin de compatibilité avec le parc logiciel impose également une configuration par défaut moins sécurisée.
  - Les nouvelles fonctions de sécurité doivent être comprises pour être utilisées par les développeurs.

Références

1. Interview David Aucsmith, *"We have never had vulnerabilities exploited before the patch was known"*. `http://news.bbc.co.uk/1/hi/technology/3485972.stm`
2. Interview Bill Gates, *"Les virus et les hackers rendent Windows plus robuste"*. `http://www.theregister.co.uk/content/55/35145.html`
3. Outil QwikFix, `http://www.pivx.com/main.html`
4. Produit SecureStack, `http://www.securewave.com/news/pr/pr019.html`
5. Produit Overflow Guard `http://datasecuritysoftware.com/`
6. SP 2 Preview, `http://www.microsoft.com/sp2preview`
7. Documentation officielle du SP2. La page du SP2 : `http://www.microsoft.com/technet/prodtechnol/winxppro/maintain/winxpsp2.mspx`
8. ICF, `http://www.microsoft.com/downloads/details.aspx?FamilyID=4454e0e1-61fa-447a-bdcd-499f73a637d1&DisplayLang=en`
9. Windows XP Service Pack 2 White Paper Overview, `http://download.microsoft.com/download/6/6/c/66c20c86-dcbe-4dde-bbf2-ab1fe9130a97/windows%20xp%20sp%202%20white%20paper.doc`
10. Analyses tierces du SP2 : Paul Thurrott sur `http://www.winsupersite.com/reviews/windowsxp_sp2_preview2.asp` - Steve Friedl sur `http://www.unixwiz.net/techtips/xp-sp2.html` et Nicolas Ruff sur `http://www.ossir.org/windows/supports/2004/2004-04-05/OSSIR-20040405%20-%20Nouveaut%E9s%20du%20SP2.pdf`