3

Les attaques réseau indirectes

Beaucoup d'attaques peuvent impacter le réseau de manière directe ou indirecte. Les chapitre 1 et 2 ont détaillé les attaques réseau proprement dites. Le présent chapitre traite des autres types d'attaques susceptibles d'impacter le réseau de manière indirecte en provoquant des phénomènes de saturation ou de congestion du réseau. Ces autres formes d'attaques réseau s'appuient principalement sur les faiblesses des applications.

Les virus sont les vecteurs des attaques les plus fréquentes contre les systèmes et réseaux informatiques. De par leur mode de reproduction, ils sont capables de saturer le réseau et de le placer en déni de service, ce qui impacte en premier lieu le réseau local. Par réplication de ce type de programme, des attaques par déni de service distribué peuvent en outre être lancées. Enfin, le phénomène de réplication des virus peut impacter le réseau d'entreprise lui-même, comme on l'a constaté avec le virus SQL Hammer.

Un autre impact sur le réseau a pour origine les attaques par les relais, qui impactent la disponibilité non pas du réseau mais des services réseau, ce qui revient, de manière indirecte, à rendre le réseau indisponible.

Nous détaillons dans ce chapitre les attaques par virus informatiques ainsi que les attaques par relais.

Attaques par virus

On peut qualifier de virus tout programme, sous quelque forme que ce soit, capable de se reproduire par lui-même.

Les virus ont pour caractéristique commune une volonté de nuire. Cette volonté peut prendre la forme d'une routine, ou programme, qui, une fois activée, use de tous les moyens à sa disposition pour empoisonner la vie de l'utilisateur.

Les principaux impacts réseau recherchés par les virus sont les suivants :

- perturber l'utilisation de la machine en faisant apparaître, par exemple, des images à l'écran ou en modifiant constamment le design de l'interface graphique ;
- consommer inutilement toutes les ressources mémoire et de calcul de la machine ;
- se reproduire autant que possible sur le disque dur de l'utilisateur, consommant le processeur et l'espace disque de celui-ci ;
- se reproduire sur les disques durs des autres utilisateurs par l'intermédiaire du partage de fichiers en réseau ;
- se reproduire en s'envoyant dans des courriers électroniques émis au nom de l'utilisateur attaqué aux contacts présents dans son carnet d'adresses.

S'il ne s'agissait que de telles nuisances, les virus ne seraient qu'un mal bénin. Malheureusement, des virus aux effets beaucoup plus dévastateurs ont fait leur apparition, notamment les suivants :

- Attaques des cartes mères des ordinateurs et flash de leur BIOS. L'ordinateur devient inutilisable et doit repartir chez le constructeur.
- Effacement des données, soit en plaçant en séquence des octets aléatoires sur le disque, soit en effaçant des fichiers au hasard, d'une manière plus ou moins rapide. Dans certains cas, il n'est pas possible de récupérer les données perdues.
- Reproduction des virus à une cadence folle *via* le réseau, saturant celui-ci, malgré les technologies au gigabit par seconde. Les virus attaquent des serveurs réseau, s'installent sur ceux-ci et les utilisent pour se reproduire. Le nombre de sources de propagation augmente de façon exponentielle, saturant non seulement les réseaux locaux mais également les réseaux WAN d'entreprise et même Internet.
- Installation des virus sur les machines afin de permettre à leurs auteurs d'en prendre le contrôle (chevaux de Troie). Dans certains cas, le virus prévient son auteur par e-mail, message ICMP, etc., afin que celui-ci sache où se trouvent les machines infectées.

Les virus ont donc un degré de nuisance variable. Quel que soit ce dernier, ils doivent être éradiqués, car ils font peser une menace constante sur les systèmes informatiques.

Cycle de vie d'un virus informatique

Les virus informatiques, tout comme les virus biologiques, se caractérisent par un cycle de vie, qui s'étend de leur création à leur destruction, en passant par leur reproduction, leur activation et leur découverte.

Création

La création d'un virus désigne le temps que passe un programmeur à construire son virus afin qu'il soit le plus efficace possible.

En règle générale, la programmation se fait en assembleur afin d'optimiser la taille du virus, qui doit demeurer la plus petite possible à des fins de discrétion.

Certains programmes mettent à la portée de n'importe qui la création de virus informatiques. Appelés *virii generator,* ces programmes produisent des assemblages de virus ayant déjà fait leurs preuves.

Nimda et CodeRed ont été créés *via* de tels programmes.

Reproduction

Par nature, les virus cherchent à se reproduire. Un virus correctement conçu se reproduit un grand nombre de fois avant de s'activer. C'est là le meilleur moyen de s'assurer de sa pérennité.

La reproduction est le procédé par lequel le virus est copié en un endroit stratégique afin que sa diffusion soit la plus rapide et la plus vaste possible.

L'ancienne méthode consistant à infecter un programme très populaire puis à le distribuer cède la place à des virus envoyés par courrier électronique — en utilisant les automatismes telle que l'API de messagerie de Microsoft MAPI (Messaging Application Programming Interface), par exemple — profitant des facilités et insécurités offertes par les programmes destinés à communiquer.

Les outils peer-to-peer de partage de fichiers tels que eMule, tout comme ceux destinés au « chat » (ICQ, MSN, etc.) ou à la téléphonie (Skype, Internet Phone, etc.), sont également devenus un vecteur privilégié de propagation de virus.

Les virus peuvent aussi mettre à profit des failles de sécurité réseau et des configurations laxistes, telles que le partage de périphérique disque sur le réseau ou de produits comme Microsoft Windows, pour se déposer sur les disques et se lancer à l'insu des utilisateurs, connectés à Internet par ADSL, par exemple.

Activation

Les virus disposant d'une capacité destructive ne s'activent généralement que lorsque certaines conditions sont réunies.

Certains ne s'activent qu'à compter de dates prédéfinies, tandis que d'autres possèdent un système de compte à rebours interne. D'autres encore détectent des situations particulières, telles que relation réseau, présence d'un logiciel particulier ou d'une configuration spéciale, etc.

Actuellement, la plupart des virus qui recherchent une visibilité maximale par souci de notoriété s'activent dès leur installation et tentent le plus rapidement possible de se reproduire en grand nombre. Leur objectif est de saturer les équipes ou les outils chargés de les nettoyer, ce qui peut être aussi efficace qu'un virus lent resté non détecté pendant une longue période.

Découverte

La découverte d'un virus est la phase où l'existence du virus est détectée et où celui-ci est isolé.

Cette phase apparaît généralement après l'activation du virus, mais il arrive que cela se produise avant. Une machine équipée d'un logiciel de détection d'intrusion capable de signer tous les fichiers peut, par exemple, être avertie avant l'activation d'un virus du changement de taille d'un fichier exécutable.

Une fois le virus isolé, il peut être transmis aux autorités compétentes, notamment la NCSA (National Computer Security Association), à Washington, et le CERT (Computer Emergency Response Team), à l'Université de Carnegie Mellon.

Le virus est alors analysé, documenté et distribué aux développeurs de logiciels antivirus. Ces derniers ajoutent sa signature à leur base de données et développent des contre-mesures.

Destruction

La phase de destruction des virus peut être considérée comme utopique. Pour pouvoir considérer un virus comme détruit, il faudrait s'assurer qu'il n'en existe plus aucune souche sur la planète. Outre l'exemplaire probablement conservé par l'auteur du virus, il est évidemment impossible d'affirmer que 100 p. 100 des ordinateurs ne sont plus infectés par ce virus.

On considère généralement que cette phase est atteinte lorsque le virus cesse de faire peser une menace réelle.

Typologie des virus

Il existe différents types de virus, dont le comportement, la mise en place ou la capacité d'être détectés sont extrêmement variables.

Les sections qui suivent détaillent les virus les plus importants, à savoir :

- virus de secteur d'amorçage ;
- virus à infection de fichiers (parasites) ;
- virus non résidents mémoire ;
- virus résidents mémoire ;
- virus multiformes ;
- virus furtifs ;
- virus polymorphes (mutants) ;
- virus réseau et vers (*worms*) ;
- virus flibustiers (*bounty hunters*) ;
- bombes logiques ;
- chevaux de Troie.

Les virus de secteur d'amorçage

Ces virus ont pour principe de se placer sur le secteur 0 du disque dur. Ce secteur étant lancé par l'ordinateur au démarrage pour initialiser le système d'exploitation, c'est évidemment un emplacement privilégié.

Du fait qu'il se lance avant le système d'exploitation, le virus dispose de possibilités supplémentaires pour empêcher sa détection. Il peut, par exemple, détourner des interruptions pour rester invisible d'un antivirus mais également se doter de facilités de reproduction.

En règle générale, le contenu par défaut du secteur 0 est copié dans un autre secteur, et le virus s'installe sur le secteur 0 pour être lancé. Par la suite, il charge lui-même le contenu précédent du secteur 0.

Il faut habituellement éteindre physiquement la machine (coupure de tension) pour que ces virus cessent d'être une menace. Bien sûr, le logiciel antivirus doit être lancé sans passer par la phase standard de démarrage du disque dur, *via* une disquette par exemple, faut de quoi le virus se recharge.

Les virus à infection de fichiers (parasites)

Les virus parasites ont pour méthode de se placer au sein de programmes exécutables sur le système d'exploitation, par exemple avec un suffixe en .com, .exe ou .sys sous Windows.

Ils sont exécutés chaque fois qu'un des fichiers programme infecté est lancé par l'utilisateur. Cela signifie que, contrairement aux virus placés sur le secteur 0, ils ne disposent que des privilèges de l'utilisateur, ce qui encourage la pratique de la séparation des privilèges, l'utilisateur ne disposant que des droits dont il a réellement besoin sur sa station de travail.

Ces fichiers infectés sont habituellement modifiés pour privilégier le fonctionnement du virus par rapport à celui du programme avant son infection. Ils s'installent au début ou à la fin du programme.

Pendant son exécution, le virus se duplique sur d'autres programmes sans que l'utilisateur en ait conscience, voire commence son action nuisible sur le système (altération ou destruction de données, etc.). La taille du programme s'en trouve modifiée, rendant sa détection aisée. Précisons que certains virus savent utiliser des zones vides au sein de programmes pour éviter d'en modifier la taille.

Les virus non résidents mémoire

Dans la plupart des cas, les virus qui ne sont pas résidents en mémoire sont ceux qui se greffent sur des fichiers.

Le programme viral s'active en totalité dès la première étape de lancement du fichier infecté. Dans la plupart des cas, d'autres fichiers se trouvent infectés rapidement et deviennent eux-mêmes des vecteurs de propagation.

Rappelons qu'un fichier infecté est généralement lancé par un utilisateur qui ne bénéficie pas des privilèges d'administrateur sur le système. Cela prouve, s'il en était besoin, que

la séparation des privilèges est une des clés de la réduction des risques de reproduction des virus avec des droits d'administrateur.

Le virus activé ne dispose que des privilèges d'utilisateur tant que les permissions sur le système sont bien paramétrées. Si l'utilisateur disposait ne serait-ce que d'une permission d'écriture sur un programme lancé par le système, le virus pourrait gagner ce privilège et devenir encore plus néfaste pour le système.

Il est possible d'éradiquer le virus avec un logiciel approprié en redémarrant la machine infectée et en lançant l'antivirus avant tout autre programme infecté.

Les virus résidents mémoire

Les virus résidant en mémoire sont indépendants du lancement d'un programme par l'utilisateur.

Disposant de suffisamment de privilèges sur le système pour se loger en mémoire, ils ont la capacité de parasiter le fonctionnement du système au niveau assez bas des interruptions.

De plus, du fait qu'ils sont installés en mémoire, ces virus peuvent être hors de portée de certains logiciels antivirus tout en continuant leurs actions néfastes.

Une fois actif, le virus infecte chaque programme exécuté qui n'est pas déjà infecté. Cela permet une propagation très efficace. Il faut éteindre physiquement la machine par une coupure de tension pour que le virus cesse d'être une menace.

Le logiciel antivirus doit être lancé sans passer par la phase standard de démarrage du disque dur, *via* une disquette par exemple, faute de quoi le virus se recharge.

Les virus multiformes

On appelle virus multiforme un regroupement de différents types de virus.

Il existe peu de virus sous cette forme. Dans le cas le plus fréquent, il s'agit de l'association d'un virus sur secteur d'amorçage et d'un virus par infection de fichiers. Ils infectent à la fois les fichiers programme et la procédure de démarrage du système d'exploitation.

Les virus furtifs

Les virus furtifs sont également appelés intercepteurs d'interruptions, car ils prennent le contrôle des interruptions logicielles du système d'exploitation afin de lui faire croire que le système est sain.

Cette prise de contrôle de la table d'interruptions s'effectue au tout début de la zone mémoire. Lorsqu'un programme émet une requête d'interruption, celle-ci est habituellement redirigée vers la table d'interruptions qui gère les commandes et permet au programme de faire son travail.

En cas d'infection par un virus furtif, celui-ci intercepte les requêtes et peut les rediriger où il le désire et effectuer toute opération possible selon son bon plaisir.

Cette capacité des virus furtifs à contrôler la table d'interruptions leur permet de se cacher de manière extrêmement efficace, rendant leur détection particulièrement ardue.

Les virus polymorphes (mutants)

Comme les logiciels antivirus détectent les comportements curieux des programmes, tels les fichiers qui voient leur taille modifiée sans raison ou des signatures particulières (séquences de bits au sein des fichiers exécutables), certains virus sont capables de déjouer ces méthodes de détection.

Appelés polymorphes, ces virus ont la capacité de chiffrer ou de modifier leur code de programmation à chaque nouveau clone, ce qui rend chaque copie unique et différente des autres. Les systèmes de détection se trouvent mis en échec par ce type de virus, car il n'existe pas de méthode pour les détecter.

Ces virus sont de plus en plus populaires depuis l'apparition des moteurs de mutation, mis au point par une personne ou un groupe se faisant appeler Dark Avenger (le vengeur noir). Propagé sur plusieurs serveurs, son code de programmation a été rendu public. Il est livré avec un jeu complet d'instructions permettant de transformer n'importe quel virus normal en virus polymorphe.

Les virus réseau et les vers (worms)

Les vers sont le type de virus que l'on rencontre aujourd'hui le plus fréquemment. Depuis la généralisation de l'accès public à Internet en haut débit, mais également du fait d'un déficit de conscience sécuritaire dans le grand public comme au sein des entreprises, ces virus trouvent un terrain propice à leur diffusion.

Prenant pour cibles les systèmes d'exploitation qui offrent des services réseau, ils utilisent ces derniers pour se répandre chez l'utilisateur, et ce selon deux grandes méthodes :

- En infectant un serveur qui fournit des ressources à une communauté d'utilisateurs, par exemple Netware, Microsoft ou un service comme le Web, ils se propagent à la communauté entière en modifiant les programmes pendant leur transmission vers l'utilisateur. On parle en ce cas de virus réseau.
- En utilisant une vulnérabilité d'un service réseau, ils attaquent le service, le pénètrent et l'utilisent pour se propager. C'est dans ce cas qu'on parle de ver. Profitant de la puissance processeur et réseau du serveur qu'ils attaquent, les vers tentent d'infecter le plus rapidement possible d'autres machines (CodeRed, Nimda, SQL Hammer, etc.). Le choix de l'algorithme de sélection des adresses réseau à infecter ainsi que la cadence d'envoi de l'infection sont les critères définissant l'efficacité du virus.

Les virus flibustiers (bounty hunters)

Ces virus extrêmement rares ont pour vocation de mettre en échec certaines solutions logicielles antivirus. Ils sont bien sûr redoutables contre la solution attaquée.

Les bombes logiques

Une bombe logique est un virus qui attend un événement pour se déclencher. Cet événement, déterminé par le programmeur malveillant, peut être une date particulière, une combinaison de touches, une action spécifique ou un ensemble de conditions précises.

Un employé mal intentionné peut implanter une bombe logique chargée de vérifier si son nom disparaît des listes du personnel de l'entreprise ou son compte d'un serveur et nuire à l'entreprise après qu'il l'a quittée en détruisant ou corrompant des données, par exemple.

Les chevaux de Troie

Pour pouvoir prendre le contrôle d'une machine, il n'y a pas énormément de possibilités : soit l'agresseur dispose des authentifications nécessaires (compte, mot de passe, etc.), soit il utilise une vulnérabilité pour pénétrer le système à l'insu de son propriétaire, soit encore il incite le propriétaire à mettre en place lui-même le moyen lui permettant d'entrer dans le système. C'est le rôle du cheval de Troie.

L'agresseur emballe son cheval de Troie dans un programme qui attire l'utilisateur. Celui-ci installe le programme et met lui-même en place le moyen de pénétration de l'agresseur. Il s'agit souvent d'un programme qui écoute sur un port TCP de son choix et qui en attend la connexion de l'agresseur.

Une nouvelle forme de cheval de Troie est apparue depuis quelques années par laquelle le virus prend l'initiative de se connecter à un serveur. L'objectif est de permettre à son concepteur d'atteindre la machine infectée malgré la présence d'un pare-feu, en remontant le flux sortant initié par le cheval de Troie.

D'autres chevaux de Troie coupent simplement le pare-feu ou ajoutent une exception afin que le port sur lequel ils écoutent soit accessible depuis n'importe quelle adresse réseau externe.

Tableau 3.1 Exemples de ports d'écoute de chevaux de Troie

Port 1234 Ultors Trojan
Port 1243 BackDoor-G, SubSeven, SubSeven Apocalypse
Port 1245 VooDoo Doll port 1269Mavericks Matrix
Port 1349 (UDP)BO DLL
Port 1509 Psyber Streaming Server
Port 1600 Shivka-Burka
Port 1807 SpySender
Port 1981 Shockrave
Port 12076 Gjamer
Port 12223 Hack´99 KeyLogger
Port 12345 GabanBus, NetBus, Pie Bill Gates, X-bill
Port 12346 GabanBus, NetBus, X-bill

Tableau 3.1 Exemples de ports d'écoute de chevaux de Troie *(suite)*

Port 12361 Whack-a-mole
Port 30303 Sockets de Troie
Port 30999 Kuang2
Port 31337 Baron Night, BO client, BO2, Bo Facil
Port 31337 (UDP)BackFire, Back Orifice, DeepBO
Port 31338 (UDP)Back Orifice, DeepBO
Port 31339 NetSpy DK
Port 31666 BOWhack
Port 33333 Prosiak
Port 33911 Spirit 2001a
Port 34324 BigGluck, TN
Port 40421 Agent 40421, Masters Paradise
Port 40422 Masters Paradise
Port 47262 (UDP)Delta Source
Port 50505 Sockets de Troie
Port 50766 Fore, Schwindler
Port 53001 Remote Windows Shutdown
Port 54320 Back Orifice 2000
Port 54321 School Bus
Port 54321 (UDP)Back Orifice 2000
Port 60000 Deep Throat
Port 61466 Telecommando
Port 65000 Devil

Techniques de codage d'un virus

En dehors du mécanisme d'infection utilisé pour pénétrer un système et de la charge finale d'un virus, celui-ci doit déployer des techniques spécifiques pour lutter contre la détection virale, notamment les suivantes :

- Polymorphisme : consiste à faire muter le code du virus lors d'une infection afin de rendre difficile la lutte antivirale en évitant de créer une signature du virus facilitant sa détection.
- Furtivité : consiste à camoufler le virus afin de rendre sa détection difficile par un antivirus. Dans ce contexte, le virus doit lutter efficacement contre sa propre surinfection afin de limiter sa détection et ainsi augmenter sa furtivité.
- Blindage : consiste à rendre difficile l'analyse du code associé au virus. La combinaison de la cryptographie et de la virologie fournit des méthodes de blindage robustes.

La figure 3.1 illustre l'infection d'un fichier par un virus appliquant la technique du polymorphisme.

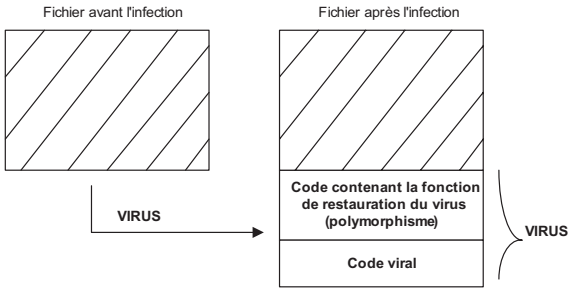


Figure 3.1
Infection d'un fichier par un virus usant de polymorphisme

Le virus Whale a été le premier virus à embarquer une fonction de détection de débogueur consistant à surveiller les interruptions système. Lors d'une telle détection, le virus Whale bloque le clavier et se désinfecte en mémoire.

Plusieurs virus apparus ces dernières années ont révélé de multiples vecteurs de propagation réseau :

- Nimda utilise les ressources NetBIOS de partage de fichiers, ainsi que les serveurs Microsoft IIS ne disposant pas de correctifs de certaines vulnérabilités, le service TFTP (Trivial File Transfer Protocol) et la messagerie électronique par l'exploitation d'une vulnérabilité d'Outlook, etc.
- NetSky est un virus qui se propage sous différentes variantes par e-mail. Il se présente sous la forme d'un message dont le titre et le corps sont aléatoires et qui possède un fichier joint. Le virus est lancé si le fichier est exécuté.
- Mydoom (et ses variantes) est un virus qui se propage par e-mail. Il se présente sous la forme d'un message au titre aléatoire, accompagné d'un fichier joint dont l'extension est, par exemple, .BAT, .CMD, .EXE, .PIF, .SCR ou .ZIP. et dont l'icône est faussement celle d'un simple fichier texte. Le virus est lancé si le fichier est exécuté.
- Welchia (et ses variantes) est un virus qui cible les ordinateurs vulnérables à une faille RPC de Microsoft. Si une machine connectée à Internet n'est pas à jour dans ses correctifs, Welchia l'infecte à l'insu de l'utilisateur puis scanne le réseau à la recherche de nouvelles machines vulnérables.
- Bagle (et ses variantes) est un virus qui se propage par e-mail. Il se présente sous la forme d'un message dont le titre est « Hi » et qui comporte un fichier joint au nom aléatoire, dont l'extension est en .EXE et l'icône est celle de la calculatrice Windows. Le virus est lancé si le fichier est exécuté.
- Sasser (et ses variantes) est un virus ciblant les ordinateurs vulnérables à la faille LSASS de Microsoft. Si une machine connectée à Internet n'est pas à jour dans ses correctifs, Sasser l'infecte *via* le port TCP 445 à l'insu de l'utilisateur puis scanne le réseau à la recherche de nouvelles machines vulnérables.

Voici une liste non exhaustive des extensions des fichiers susceptibles d'être infectées par un virus : ACE, ACM, ACV, ARC, ARJ, ASD, ASP, AVB, AX, BAT, BIN, BOO, BTM, CAB, CLA, CLASS, CDR, CHM, CMD, CNV, COM, CPL, CPT, CSC, CSS, DLL,

DOC, DOT DRV, DVB, DWG, EML, EXE, FON, GMS, GVB, HLP, HTA, HTM, HTML, HTA, HTT, INF, INI, JS, JSE, LNK, MDB, MHT, MHTM, MHTML, MPD, MPP, MPT, MSG, MSI, MSO, NWS, OBD, OBJ, OBT, OBZ, OCX, OFT, PCI, PIF, PL, PPT, PWZ, POT, PRC, QPW, RAR, SCR, SBF, SH, SHB, SHS, SHTML, SHW, SMM, SYS, TD0, TGZ, TT6, TLB, TSK, TSP, VBE, VBS, VBX, VOM, VWP, VXE, VXD, WBK, WBT, WIZ, WPC, WPD, WML, WSH, WSC, XML, XLS, XLT, ZIP.

Détection virale et théorie de la complexité

La théorie de la complexité a été développée dans les années 1970 dans le but de classifier des problèmes selon des classes de complexité. L'objectif initial était de savoir si, pour un problème donné, il existait un algorithme permettant de trouver une solution en un temps polynomial, c'est-à-dire susceptible de rendre ces problèmes traitables.

Plusieurs classes ont été définies, pointant des problèmes de plus en plus difficiles, comme l'illustrent les exemples suivants :

- Si G est un graphe orienté valué et s et t deux sommets, trouver un chemin de coût minimal de s à t ? Plusieurs algorithmes, notamment Bellman et Dijkstra, permettent de donner la solution en un temps polynomial en fonction de la taille du graphe G.
- Le problème du voyageur de commerce consiste à trouver un cycle, ou circuit hamiltonien, de coût minimal dans un graphe valué complet. Il s'agit d'un problème difficile, dont aucun algorithme connu ne permet de trouver une solution optimale en un temps polynomial. En revanche, on peut construire une solution non optimale à l'aide de méthodes dites « gloutonnes » et améliorer cette solution de base avec des méthodes dites « méta-heuristiques ».

Les bases de la formalisation de la théorie de la complexité viennent des travaux de Alan Turing sur l'existence effective d'un programme permettant de résoudre un problème. L'idée initiale était de savoir si un programme permettrait de répondre à coup sûr à un problème donné ? Alan Turing a montré qu'il existait des problèmes non décidables, qu'aucun programme ne permettait de résoudre.

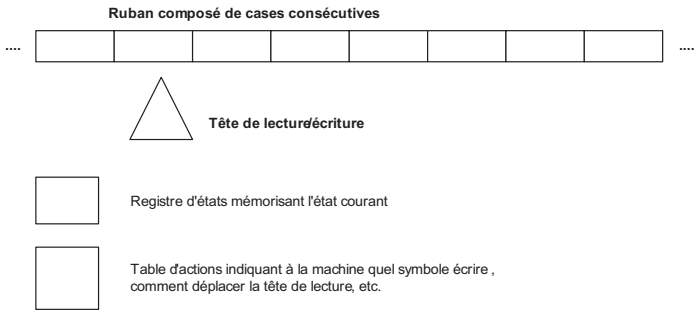
Pour le démontrer, Alan Turing a créé la fameuse machine de Turing, qui est un modèle abstrait du fonctionnement d'un ordinateur et de sa mémoire afin de donner une définition précise au concept d'algorithme (ou procédure mécanique).

La figure 3.2 illustre une machine de Turing composée d'un ruban, d'une tête de lecture/écriture, d'un registre d'états et d'une table d'actions. C'est ce modèle qui est encore aujourd'hui largement utilisé en informatique théorique, en particulier pour résoudre les problèmes de complexité algorithmique et de calculabilité.

Fondées sur l'approche de Turing, différentes classes de problèmes ont été définies, telles que les suivantes :

- Classe P : problèmes solubles en un temps polynomial.

Figure 3.2
Fonctionnement de la machine de Turing



- Classe NP : problèmes vérifiables en un temps polynomial. Si l'on dispose d'une solution « certifiée », on peut vérifier cette dernière en un temps polynomial.
- Classe NP-complet : problèmes appartenant à NP et étant aussi « difficiles » à résoudre que n'importe quel problème de classe NP. Dit autrement, s'il existe un quelconque problème NP-complet soluble en un temps polynomial, tout problème NP-complet a un algorithme à temps polynomial.

Un virus est un programme caractérisé par la définition suivante : « Séquence de symboles qui, interprétée dans un environnement donné (adéquat), modifie d'autres séquences de symboles dans cet environnement, de manière à y inclure une copie de lui-même, cette copie ayant éventuellement évolué. »

Les travaux de Fred Cohen et Leonard Adleman dans les années 1984-1989 ont permis de formaliser le problème de la décidabilité de la détection virale à l'aide de machines de Turing. Le résultat de ces travaux montre que toute détection virale absolue est impossible. En d'autres termes, il n'existe aucun programme capable de détecter à coup sûr un virus.

Il ne faut pas en conclure que les logiciels antivirus ne servent à rien, puisqu'ils permettent déjà de détecter et d'éradiquer la base existante des virus connus. En revanche, la lutte antivirale ne doit pas uniquement reposer sur des programmes antiviraux, mais s'appuyer aussi sur d'autres techniques.

Par exemple, les recommandations suivantes doivent être suivies par les utilisateurs afin de compléter les dispositifs de lutte antivirale :

- n'ouvrir et ne transmettre aucun message e-mail provenant d'un expéditeur inconnu ou incertain ;
- n'ouvrir et ne transmettre aucune pièce jointe dans un e-mail provenant d'un expéditeur inconnu ou incertain ;
- n'ouvrir et ne transmettre aucun fichier ou message attaché ayant un aspect suspect ou inattendu ;
- ne copier aucun fichier inconnu ou ne faire aucune confiance à sa source ;
- utiliser un programme antivirus fiable et le mettre à jour le plus souvent possible ;
- faire des copies de secours régulières des données importantes.

Technologies de lutte antivirale

Il existe différentes manières de traiter les menaces des virus, tant au niveau architectural que technique.

Les méthodes de détection ont grandement évolué ces dernières années afin de tenir compte à la fois des techniques de programmation des virus, mais aussi des mutations des virus lors de leur phase de reproduction. Pour nécessaires qu'elles soient, ces méthodes restent insuffisantes pour la détection des futurs virus.

La scanérisation

Le procédé de scanérisation repose sur une base de signatures de virus pour la phase de détection. Une signature est une portion de code propre à un virus qui permet de l'identifier. Il s'agit en quelque sorte de l'empreinte digitale du virus.

Lorsque l'existence d'un nouveau virus est avérée, celui-ci est extrait du programme qu'il infecte pour être analysé et sauvegardé. Le programme de scanérisation effectue alors une comparaison entre les éléments qu'il découvre et ceux présents dans la base de données de signatures sur laquelle il s'appuie. S'il y a correspondance, le fichier est considéré comme infecté. Sinon, le fichier est considéré comme sain. Les programmes sérieux effectuent également une analyse des zones de fichiers et du secteur d'amorçage.

Le point faible de ce genre de programme est que toute infection par un virus inconnu risque de passer inaperçue. Si un fichier infecté par un virus dont la signature ne figure pas encore dans la base de signatures de l'antivirus est utilisé, l'antivirus ne peut s'en rendre compte.

Demander à un logiciel antivirus utilisant cette technique de trouver les virus qui ne sont pas encore répertoriés équivaudrait à rechercher une aiguille dans une meule de foin. Sans aucune idée de son emplacement ni de sa physionomie, la tâche est presque impossible. La scanérisation n'est donc une méthode fiable que si l'on recherche des virus connus.

Certains scanners antivirus se contentent de vérifier le début et la fin des fichiers. L'impression de sécurité est alors illusoire, car de nombreux virus en circulation sont capables de se greffer au cœur même des fichiers. Cette manière de procéder est donc des plus dangereuses puisqu'elle laisse passer des virus bien que leur signature soit connue.

Tests d'intégrité

Les tests d'intégrité s'appuient sur l'analyse de la taille en octet des fichiers. L'installation d'antivirus implémentant cette fonctionnalité doit s'effectuer sur un système parfaitement sain, car ces logiciels créent une multitude de petits fichiers leur servant de référence et comprenant des informations précises à propos de la taille des divers fichiers présents sur le disque dur.

Par la suite, ces antivirus effectuent une comparaison permanente entre la taille effective des fichiers analysés et les fichiers de référence correspondants afin de détecter toute modification suspecte.

Le reproche principal que l'on peut adresser à cette méthode est qu'elle n'effectue aucune action préventive, la détection d'un virus n'étant possible qu'après infection. Le rapport d'infection est délivré *a posteriori,* une fois que le virus a fait son office, et l'antivirus est incapable d'identifier la source de l'infection.

De plus, toute modification effectuée par les programmes eux-mêmes engendre des alertes intempestives. Les antivirus fondés sur cette seule méthode ne peuvent en aucun cas être considérés comme efficaces et ne procurent nullement la prévention nécessaire.

Analyse comportementale

La recherche de comportements anormaux du fait de la présence de virus dans un environnement informatique est généralement effectuée par un programme résident en mémoire. Ces programmes sont de type TSR (Terminate and Stay Resident), c'est-à-dire qu'ils doivent être à même d'analyser les requêtes dirigées vers la table d'interruptions.

Le comportement des virus au sein des applications peut presque toujours être considéré comme anormal. C'est ce qu'on appelle l'activité virale. Peuvent être considérées comme activités virales les requêtes d'écriture dans le secteur d'amorçage, les requêtes d'ouverture de programmes en écriture et les tentatives de programmes cherchant à se loger en mémoire. Certaines opérations communément effectuées par les virus permettent d'élaborer un système de règles visant à différencier un comportement normal d'un comportement viral.

L'analyse fondée sur de telles règles permet de détecter de manière très performante les virus connus et inconnus et d'arrêter une tentative d'infection avant même qu'elle ait la possibilité d'endommager un fichier.

Les pièges à virus ainsi constitués offrent de multiples avantages. Ils peuvent empêcher toutes sortes de programmes pernicieux d'endommager le système d'information et se révèlent particulièrement efficaces contre les virus suivants :

- virus connus et inconnus ;
- chevaux de Troie ;
- bombes logiques.

Un inconvénient mineur de l'analyse comportementale réside dans le fait qu'elle est incapable d'identifier les virus détectés. Seul le procédé de scanérisation permet d'obtenir ce genre de renseignement.

Mécanismes réseau de lutte antivirale

Les dénis de service exploitent généralement de fausses adresses IP sources afin de masquer l'origine des attaques. De telles adresses sont généralement choisies parmi les adresses IP dites réservées, ou BOGONS (RFC 1918). Ces BOGONS doivent être filtrés par les opérateurs de télécommunications en périphérie de leurs réseaux afin de limiter leur exploitation à des fins de déni de service. Ces filtres ne sont malheureusement pas appliqués de manière systématique.

La limitation en terme de bande passante d'un protocole tel que ICMP peut limiter les dénis de service fondés sur de tels messages. En revanche, la limitation de la bande passante par protocole réseau reste un exercice périlleux et souvent voué à l'échec de par la nature non prédictible des trafics.

D'autres mécanismes réseau, tels que l'URPF (Unicast Reverse Forwarding Protocol), permettent de n'autoriser un trafic que si l'adresse source est présente dans les tables de routage. Ces mécanismes peuvent toutefois s'avérer complexes à mettre en œuvre, et, en théorie, ils ne protègent pas des dénis de service.

Les techniques de puits de routage réseau, ou *black* ou *sink hole,* sont réalisées par les opérateurs de télécommunications auxquels est connecté le système visé par un déni de service.

Elles fonctionnent de la façon suivante :

1. Une fois qu'un déni de service est détecté sur une adresse IP, le responsable de cette adresse avertit son opérateur de télécommunications.
2. L'opérateur indique au processus de routage du réseau, généralement le protocole BGP (Border Gateway Protocol), que le trafic à destination de cette adresse IP doit être mis systématiquement au rebut (black hole) ou être redirigé vers un équipement dédié ayant la capacité de l'analyser et de le filtrer afin de séparer le trafic légitime de celui de l'attaque (sink hole).

Utilisation malicieuse de la cryptographie

La cryptographie permet généralement de se protéger contre de nombreuses faiblesses de sécurité et de contrôler la sécurité des systèmes d'information. Cette science peut cependant être aussi utilisée par les auteurs de virus afin de renforcer leur caractère nocif.

La première définition de la cryptographie malicieuse a été donnée par M. Yung et L. A. Young selon le modèle opératoire dit hybride (algorithmes de chiffrement symétrique et asymétrique) suivant :

1. Une paire de clés publique/privée est générée par l'auteur du virus.
2. Ce dernier insère uniquement la clé publique dans le corps du programme du virus et libère le virus sur le réseau. L'auteur du virus est le seul possesseur de la clé privée et ne la diffuse évidemment pas.
3. Lorsque le virus atteint un système par le biais d'une faiblesse de sécurité, il génère de manière aléatoire une clé A, qu'il utilise pour chiffrer les données du système à l'aide d'un algorithme de chiffrement symétrique.
4. Le virus chiffre la clé A avec la clé publique qu'il possède au moyen d'un algorithme de chiffrement asymétrique. Appelons la clé chiffrée A'.
5. Les données du système sont pris en otage, et une demande de rançon peut être exigée pour les récupérer.

6. Après paiement de la rançon, la clé A' est fournie à l'auteur du virus afin d'être déchiffrée avec la clé privée (rappelons que la clé A a été chiffrée préalablement avec la clé publique) et de fournir en retour la clé A ainsi que l'algorithme de chiffrement symétrique utilisé. Le couple constitué de la clé A et de l'algorithme de chiffrement symétrique permet de récupérer (déchiffrer) les données du système.

Bien que très peu de virus implémentent de tels mécanismes, ces derniers montrent qu'il est possible d'utiliser la cryptographie à des fins malfaisantes. La cryptographie peut aussi être utilisée afin de réaliser du polymorphisme du code viral (modification de la sémantique du code) et du blindage viral.

Les virus qui utilisent la cryptographie à des fins de blindage viral implémentent avant tout une gestion « environnementale » des clés associées. Sachant que la présence statique de clés dans du code viral peut compromettre le caractère nocif du virus, ce dernier gère ces clés à partir de l'environnement dans lequel il est présent. Il agit alors en aveugle et ignore si les clés sont disponibles à un instant *t*.

Dans le cas du virus Bratley, qui utilise un tel principe de blindage, E. Filiol a démontré que l'analyse de ce code révélait une complexité exponentielle et que les problèmes étaient considérés comme intraitables.

Attaques par relais

Les attaques par relais peuvent impacter le réseau ainsi que les services réseau. Un relais peut être un système de messagerie fondé sur le protocole SMTP (Simple Mail Transfer Protocol) ou un système de résolution de noms de domaine à l'aide du protocole DNS (Domain Name Service).

En dehors des attaques classiques présentées au chapitre précédent, les sections qui suivent donnent quelques exemples de vecteurs d'attaques sur les relais.

Attaques par vers

Les vers sont de plus en plus utilisés dans les attaques réseau, notamment les attaques dites DDoS (Distributed Denial of Service). Récemment, SQL Hammer a provoqué la panique au sein d'Internet, venant après CodeRed et Nimda, qui avaient engendré d'énormes perturbations pendant plusieurs jours, voire semaines.

La partie du ver qui permet de définir si celui-ci impactera le réseau est celle chargée de sa reproduction. Sortant de sa discrétion, le ver peut chercher à se propager le plus rapidement possible. Il utilise pour cela le protocole UDP (User Datagram Protocol) et envoie autant de paquets qu'il le peut. Un seul système SQL Hammer, par exemple, pouvait surcharger une bande passante d'un gigabit par seconde.

La méthode de sélection des adresses IP victimes a également son importance. Si le ver génère les adresses au hasard, il infecte Internet avant l'entreprise où est situé le système

infecté et a de grandes chances de rencontrer des solutions de filtrage, qui ralentissent sa propagation.

S'il utilise la configuration réseau de la machine infectée pour tenter en premier la propagation locale, il a des chances d'être détecté plus tardivement, atout majeur pour un virus, voire d'infecter plus vite d'autres victimes.

Comme les vers CodeRed et Nimda l'ont démontré, ces méthodes changent du tout au tout l'efficacité et donc l'impact du ver sur le réseau. CodeRed réussissait à se propager à une cadence infernale, en tout cas beaucoup plus efficacement que Nimda, qui utilisait pourtant les mêmes vecteurs.

Attaques visant la saturation des systèmes relais

Diverses techniques d'attaques permettent de rendre indisponibles les relais ainsi que les services réseau qu'ils supportent, notamment les suivantes :

- Faire relayer un courrier électronique SMTP vers un grand nombre de destinataires en copie cachée, ou BCC (Blind Carbon Copy). Le relais reçoit un seul message mais doit générer un nouveau message pour chaque destinataire. Un simple message de 3 Mo envoyé à 1 000 destinataires implique pour le serveur de générer 1 000 messages de 3 Mo. L'attaque impacte donc le serveur et le réseau local et rend indisponible le service pour d'autres utilisateurs.
- Rendre indisponible la résolution de noms de domaines. Tout réseau fonctionne en s'appuyant sur des services partagés, tels que le service DNS de résolution de noms de domaine. Le service DNS permet d'accéder aux systèmes sans qu'il soit nécessaire d'avoir en tête les adresses IP. Toute atteinte à l'intégrité de ce service ou à sa disponibilité peut impacter la disponibilité des services réseau et donc le réseau lui-même.
- Saturer les ressources offertes par un système. Cela peut se faire par le biais du réseau, grâce aux inondations, ou par la saturation de la mémoire ou du processeur.

Les CERT (Computer Emergency Response Team)

Les CERT (Computer Emergency Response Team) ont été créés au début des années 1990 aux États-Unis suite à des incidents de sécurité survenus sur les réseaux de la recherche américains. De nos jours, de nombreux CERT sont en place dans la plupart des pays de la planète.

La mission d'un CERT est d'assister ses adhérents en matière de sécurité informatique, notamment dans le domaine de la prévention, de la détection et de la résolution d'incidents.

Les trois CERT suivants sont à l'œuvre en France :

- CERT Renater (secteur des universités et de la recherche) : *http://www.renater.fr/Securite/CERT_Renater.htm* ;
- CERTA (secteur des administrations) : *http://www.certa.ssi.gouv.fr/*;

- CERT-IST (secteurs de l'industrie, des services et du tertiaire) : *http://www.cert-ist.com/*.

Les CERT sont regroupés au sein d'une structure appelée FIRST (Forum of Incident Response and Security Team), qui assure notamment la cohérence des actions entreprises et normalise le mode de fonctionnement des différents CERT. Cette structure permet en outre de partager informations, expertises et outils.

En résumé

Les entreprises sont aujourd'hui bien conscientes de l'utilité d'une politique antivirale, surtout après les grandes attaques virales CodeRed, Nimda et SQL Hammer, qui ont causé des dégâts évalués en millions d'euros et de dollars aux entreprises mal protégées et ont démontré leur capacité à impacter les réseaux locaux ainsi que ceux des opérateurs de télécommunications.

On ne saurait pour autant être trop optimiste pour l'avenir. Le nombre de virus écrits dans le monde augmente en permanence. L'apparition de nouvelles fonctionnalités sur les téléphones portables et autres types d'équipements rattachés à des réseaux, comme celles permises par la technologie Java, augmente la probabilité de propagation des virus et les menaces qui pèsent sur les réseaux. Il existe d'ailleurs déjà des virus ou des preuves de concept de virus sur de tels appareils.

Comme l'a dit Sun Tzu dans son traité *L'Art de la guerre :* « Si tu te connais sans connaître ton ennemi, pour chaque victoire il y a aura également une défaite. » Il est important de connaître non seulement toutes les attaques possibles mais aussi les éléments à protéger, comme nous allons tenter de le montrer à la partie suivante de l'ouvrage.