Editions Ishango - UAC
Revue Internationale Multidisciplinaire Etincelle

Ed. ishango

Article

De la vulnérabilité des informations numériques dans les réseaux informatiques : Cas de l'infiltration à travers le rootkit

Héritier Nsenge Mpia*

Faculté de Sciences Economiques et de Gestion, Université de l'Assomption au Congo, Butembo, P.O. Box 104, République Démocratique du Congo

* Correspondant: staniher@yahoo.fr

Abstract: Les réseaux informatiques constituent un monde complexe dans le domaine du computer science. Cette complexité se justifie par l'explosion des domaines tels que la sécurité informatique, le hacking, le contrôle illicite des PC à distance, etc. L'actualité de ces concepts n'est plus à démontrer dans l'univers des entreprises et des consommateurs de l'informatique. Leur innovation ininterrompue fait naître un regain d'intérêt en informatique et fait émerger d'autres aspects de la programmation: programmation système, programmation réseau. C'est l'émergence des outils de la nouvelles technologies de l'information et de la communication (NTIC). Certains de ces outils sont à la base de plusieurs conflits cybernétiques de notre ère et parviennent à mettre en brèche de nombreux systèmes dits sécurisés. Ainsi, ignorer l'existence de ces technologies aussi bien prometteuses que destructives est un danger pour des systèmes informatiques. En ce sens, il est important de connaître leur existence et de prendre des mesures de prévention afin de mettre, tant soit peu, à l'abri des risques les informations, qui sont on ne peut plus vitales pour la vie d'une entreprise.

Mots clés: Rootkit, infiltration, réseau informatique, vulnérabilité, sécurité informatique

Citation: Mpia, H.N. De la vulnérabilité des informations numériques dans les réseaux informatiques : Cas de l'infiltration à travers le rootkit. *Etincelle*, 2018, Vol. 21, no. 1. https://doi.org/10.61532/rime211111

Reçu: 29/03/2018
Accepté: 06/09/2018
Publié: 07/10/2018

Note de l'éditeur: Ishango-uac reste neutre en ce qui concerne les revendications juridictionnelles dans les cartes géographiques publiées et les affiliations institutionnelles des auteurs.

1. Introduction

La venue des nouvelles technologies est une véritable réussite dans la vie actuelle. Les réseaux informatiques, constituent un monde complexe dans le domaine du computer. Cette complexité se justifie par l'explosion des domaines tels que la sécurité informatique, le hacking, le contrôle illicite des PC à distance, etc. L'actualité de ces concepts n'est plus à démontrer dans l'univers des entreprises et des consommateurs de l'informatique. Leur innovation ininterrompue fait naître un regain d'intérêt en informatique et fait émerger d'autres aspects de la programmation : programmation système, programmation réseau, etc. C'est l'émergence des outils des nouvelles technologies de l'information et de la communication (NTIC). Néanmoins, certains de ces outils peuvent nuire à la sécurité de tout un système informatique. Ces technologies ont un impact ambivalent sur la société moderne dite numérique. En fait, on note l'utilité des outils informatiques dans la vie de l'homme. En effet, grâce au réseau et à la programmation informatique, l'accès à l'information n'exige plus un déplacement physique de l'individu. Ce qui nous pousse à affirmer sans ambages que les nouvelles technologies ont facilité le travail de l'homme. Mais, ces technologies ont déstabilisé également le fonctionnement et la sécurité des entreprises et ont fait naître un nouveau type de conflit que nous nommons, dans ce travail, *espiologicisme*. À vrai dire, les outils informatiques, surtout les réseaux ont un gain incommensurable, en ce sens qu'ils allègent les travaux de l'homme et l'effort manuel de la vie quotidienne. Toutefois, ces infrastructures sont pleines de vulnérabilités. Le Rootkit est une technologie très illustrative qui démontre combien la confidentialité, l'intégrité et la

disponibilité de l'information sont loin d'être considérées comme un mythe pour tous ceux qui exploitent le monde de l'Internet. En effet, aussi longtemps que le réseau internet et bien d'autres outils informatiques comprendront des vulnérabilités exploitables, les rootkits en tireront partie (Hoglund & Butler, 2006). À proprement parler, depuis quelques décennies, les hackers et ingénieurs en sécurité ont fait naître une nouvelle technologie du nom de rootkit, technologie jusqu'alors tenue cachée aux yeux de nombre d'internautes voire de certains informaticiens. Certes, des milliers des rootkits sont développés au monde (nombreux sont commercialisés).

Tous ces rootkits n'ont que pour mission l'infiltration des systèmes informatiques d'une façon insidieuse. Pour Lacombe (2009), le rootkit est un système parasite qui permet à un attaquant de maintenir dans le temps un contrôle frauduleux sur un système informatique. C'est ce qui arrive lorsque l'on utilise l'internet sans se préoccuper des risques imminents qui découlent des outils de sécurité. La méconnaissance de ces dangers est la cause de plusieurs dégâts incommensurables détectés actuellement en informatique. Nombre des internautes et même des administrateurs n'ont que la connaissance des virus et des vers informatiques, et ignorent les Rootkits, les Bootkits, etc. qui sont des outils capables de contrôler totalement un PC à distance et de manière discrète. C'est ainsi qu'ils sont victimes de maintes attaques au niveau de l'internet. Notre souci, dans cet article s'insère dans le cadre de sensibilisation des utilisateurs de l'internet de l'existence et du fonctionnement de la technologie des rootkit qui peut nuire à la sécurité de leurs systèmes informatiques afin de garder leurs systèmes informatiques intacts, puisque le système informatique est vital et tout ce qui le menace est potentiellement mortel. Ainsi, conjurer les menaces contre le système est un impératif car les menaces engendrent des risques et coûts humains et financiers (Bloch & Wolfhugel, 2009).

Précisons tout de même que le souci qui nous hante hic et nunc est de pouvoir montrer aux lecteurs que la cybercriminalité est le berceau des conflits dans notre société numérique. Elle peut engendrer une guerre plus que nucléaire entre les Etats puisque, de nos jours, plusieurs pays et entreprises font usage à ces technologies afin d'espionner leur cible. Nous sommes donc face à une source éblouissante des conflits. Si nous jetons un regard dans l'histoire, l'on se conviendrait avec William Ury (2001) que tous les conflits mal gérés se sont soldés par des résultats sanglants: guerre, violence, mutinerie, occupation, etc. Toutefois, l'objectif de cet article est d'éclairer les internautes des incidents informatiques qui surgissent des attaques et comment s'en préserver avec des simples techniques et manipulations de leurs ordinateurs.

Le reste de cet article est subdivisé en quatre points. Au premier abord, nous avons parlé des notions préliminaires, où nous avons abordé quelques concepts ayant trait aux attaques informatiques mais aussi nous avons essayé de parler du fonctionnement de rootkit et de la façon dont cet outil informatique s'infiltre dans le noyau Windows. Dans le deuxième, nous avons parlé de la méthodologie utilisée dans cette recherche. Dans le troisième point, nous avons tenté de montrer comment se protéger des certaines attaques informatiques pour éviter toute sorte de conflit interpersonnelle, interentreprises et interétats qui peuvent en découler. Le dernier point résume les efforts scientifiques de cette recherche en terme de conclusion.

2. Revue de littérature

Dans ce point, nous voulons mettre au clair certaines notions relatives au Rootkit. En effet, la technologie dont nous parlons est récente, et beaucoup de gens n'en connaissent pas les enjeux. C'est ainsi que cette partie a pour but d'éclairer le public sur cette réalité. Tout part du système informatique. Les terminologies que nous voulons découvrir ici sont toutes nocives au système. Mais c'est quoi un système? Pour Le Moigne (2006), le système est quelque chose, n'importe quoi, présumé identifiable qui, dans quelque chose, c'est-à-dire, un environnement pour quelque chose qui peut être une finalité, fait quelque chose par quelque chose et qui se transforme dans le temps. Le système est, en fait, un ensemble

d'objet aussi bien homogènes qu'hétérogènes capables d'entrer en interaction et d'influencer son évolution. D'où l'aspect communicationnel et évolutif des éléments du système qui peut influer sur l'organisation de ces éléments selon des règles ou normes établies pour son fonctionnement.

2.1. Du système informatique

Le système informatique est l'automatisation d'un système d'information en vue de traiter les informations et de les restituer aux utilisateurs de façon automatique. Il assure la communication, le traitement et la mémorisation des données afin de permettre le fonctionnement adéquat d'un système d'information. Ce dernier est la partie du réel constituée d'informations qui sont organisées, d'événements qui ont un effet sur ces informations et d'acteurs qui agissent sur ces informations ou à partir de ces informations, selon des processus visant une finalité de gestion et utilisant les technologies de l'information (Morley, 2008). En revanche, le système informatique, cible potentielle des menaces dans le réseau, est un ensemble organisé des matériels, des logiciels, des applications dont la mise en œuvre réalise l'infrastructure d'un système d'information (Morley, 2008).

2.2. Les menaces et les intrusions informatiques

Dans le domaine informatique, il y a une pléthore d'attaques; certaines sont connues des utilisateurs, d'autres tenues cachées par les experts. Toutes ces attaques visent à modifier le comportement d'un système informatique. À côté de ces attaques, nous rencontrons diverses actions ou manipulations des logiciels malicieux visant à atteindre le noyau du système d'exploitation. L'objectif de ces attaques et malwares est de compromettre le système. Une fois que l'intrus s'introduit dans le système, il peut entreprendre des actions profitant de vulnérabilités et faiblesses afin d'utiliser le système et dans la majorité des cas, de pérenniser son accès à l'insu des utilisateurs légitimes (Lacombe, 2009). Or les attaques existent selon les objectifs et les finalités. On trouve par exemple, dans le cas des attaques d'un cryptosystème, l'emploi de la méthode de la force brute ou la méthode des dictionnaires. Pour ce qui est d'interception des paquets des données, le pirate peut faire usage des attaques de Man In The Middle (MITM), les attaques par rejet voire les attaques par injections des codes malicieux en exploitant soit la vulnérabilité de Cross-Site Scripting pour ce qui est des sites web soit en développant des logiciels malveillants tels que vers informatiques, virus, cheval de Troie, rootkit, etc. Ce qui conduit à un grand risque du système. À proprement parler, les risques peuvent se réaliser si les systèmes menacés présentent des vulnérabilités et/ou failles. Il est possible de préciser la notion de risque en la décrivant comme le produit d'un préjudice par une probabilité d'occurrence : **risque** = *préjudice* x *probabilité d'occurrence* (Bloch & Wolfhugel, 2009). Ce risque peut être source de conflit soit dans l'entreprise, entre l'agent qui a favorisé la brèche et le manager, soit dans la société entre le pirate et les victimes.

2.2.1. Des attaques informatiques

L'attaque en informatique c'est l'exploitation d'une faille ou d'une vulnérabilité du système informatique pour des fins que seul l'attaquant connaît. C'est ainsi que toute attaque du système nécessite une phase de préparation qui correspond à la collecte des informations du système compromis. Cette phase est aussi appelée prise d'empreinte et elle permet à l'attaquant de prendre le maximum d'informations sur sa cible, de la connaissance afin de mener l'attaque de façon efficace, et d'attaquer les points sensibles (Hoglund & Butler, 2006). Cette collecte d'informations sur la cible peut se faire d'une manière directe ou indirecte. Au clair, pour réaliser une attaque ingénieuse, il y a cinq principales étapes à suivre. Il s'agit de la collecte des informations sur la cible, appelée aussi

reconnaissance, du scannage, de l'obtention d'accès, du maintien d'accès à travers le Root-kit et le Backdoor, et de l'effacement de traces (ACISSI, 2009).

2.2.2. Les malwares

Au cœur des attaques qu'un système informatique peut subir se trouvent les logiciels malveillants, appelés en anglais malware. Ces logiciels se propagent en majeure partie dans le réseau, soit par accès direct à l'ordinateur compromis, soit cachés dans un mail, soit dans un site hyper attrayant, soit dans une image pornographique. Le but premier des malwares est de s'installer sur l'ordinateur dont ils ont réussi à compromettre afin de commettre des délits voire se propager vers d'autres victimes. Citons-en certains:

a) Les virus informatiques

Le virus est un programme autoreproducteur, un automate ayant la capacité d'auto propagation qui lui confère une certaine indépendance (Hoglund & Butler, 2006). Ce programme peut infecter des milliers des machines en quelques minutes vu son efficacité de se propager dans le réseau. Il est très nuisible dans un système. Le virus infecte d'autres programmes en les modifiant pour y inclure une copie de lui-même qui soit différente (Filiol & Fizaine, 2007). Il s'avère qu'en dépit de leur finalité, tous les virus fonctionnent selon un modèle unifié qui les décrit tous: le modèle de virus générique. Filiol et Fizaine (2007) estiment que le virus est une sorte de machine de Turing se décomposant en sous autres machines de Turing connectées entre elles dont chacune représente un élément fonctionnel. Un virus ***V*** est un quadruplet ***Pr*** = (***Rech, Inf, Cf, Tr***). ***Rech*** c'est l'ensemble des fonctions de recherche de cibles; ***Inf*** est l'ensemble des fonctions d'infection tel que ***Inf={E, R, A, T}*** où ***E,R,A,T*** sont des ensembles de machines de Turing tels que ***E*** la classe des fonctions par écrasement, ***R*** la classe des fonctions par recouvrement, A la classe des fonctions par accompagnement, ***T*** est la classe des fonctions par entrelacement. Tandis que ***Cf*** est l'ensemble des fonctions des charges finales et ***Tr*** l'ensemble des fonctions de transfert d'exécution (Filiol & Fizaine, 2007).

b) Les vers informatiques

À la manière des virus, les vers sont des programmes autoreproducteurs qui se propagent à travers le réseau. La différence est que le ver se propage par recopie sans être nécessairement attaché à un fichier et il est capable de se déplacer et de se reproduire via un réseau informatique (Filiol & Fizaine, 2007). Les vers sont classés en trois grandes catégories. D'abord, la classe des vers dits simples qui exploitent les failles logicielles réseau pour se disséminer. Ensuite, la classe des vers macro dont le mode de dissémination se fait à travers les pièces jointes contenants des documents bureautiques infectés. Enfin, les vers d'emails qui ont comme champ de déploiement les pièces jointes qui contiennent des codes malicieux activés soit par l'utilisateur soit par une application de courriel électronique ayant une faille (Filiol & Fizaine, 2007).

c) Les Chevaux de Troie

Le terme Cheval de Troie prête souvent à une confusion. Certes, certains confondent le Troyan au Rootkit. Le Troyan fait référence à l'histoire des grecs qui avaient pris d'assaut la ville de Troie en utilisant un cheval. Le Troyan ou Cheval de Troie est un programme simple composé de deux parties: un module serveur et un module client qui facilite l'accès à tout ou partie des ressources de la victime à l'attaquant qui en dispose par l'entremise du réseau. Toutefois, le Cheval de Troie se présente de plusieurs manières. À titre illustratif, les leurres (fausse bannière de connexion Unix), les espiogiciels, etc. (Filiol & Fizaine, 2007). Le cheval de Troie peut être une application malicieuse intégrée, par

exemple avec la technique d'Alternative Data Stream (ADS), dans une autre application de confiance. Une fois l'application légitime est exécutée dans la machine de la cible, l'application malicieuse s'exécute aussi en background. Cela peut être un simple espiogiciel ou un rootkit.

d) Le Rootkit

C'est ce concept qui constitue la clé de voute de notre recherche. Source de conflit et d'insécurité informatique, dans le monde actuel, le rootkit existe il y a environ vingt-cinq ans. Il désigne, comme le disent Hoglund et Butler (2006), un kit de petits programmes qui permettent à un attaquant de maintenir un accès de niveau "root", i.e. administrateur sur un système informatique ou un ordinateur, c'est un ensemble de programmes et de code qui donnent à son utilisateur une présence permanente ou cohérente et indétectable sur un ordinateur afin de mener des actions malveillantes ou pas (Hoglund & Butler, 2006). Aujourd'hui, la plupart des entreprises, pour se maintenir dans la concurrence managériale et entrepreneuriale, se dotent des rootkits afin d'espionner leurs conquérants et ainsi d'anticiper leur action sur le marché. Signalons que le rootkit est la technologie la plus dangereuse et la plus conflictuelle que le monde n'a pu connaître. Aujourd'hui, il est utilisé même dans les conflits entre Etats. ***Uroburos***, par exemple, est le rootkit russe espionnant plus d'une entreprise au monde. En fait, le gouvernement russe a été accusé d'être à l'origine d'une attaque numérique conçue pour viser les grands réseaux, les espionner, et en prendre le contrôle. Ainsi, on assiste actuellement à une nouvelle forme de conflit dans ce monde dit numérique.

e) Le Backdoor

Dans l'essence de tout système informatique, le système d'exploitation est aujourd'hui un abime présentant une simple interface matérielle ou logicielle derrière lequel est caché une façade complexe des logiciels qui fonctionnent dans l'arrière-plan. Un backdoor est une porte dérobée logicielle qui permet à un pirate de revenir ou d'élever ses privilèges au niveau root dans un système. Il est souvent dissimulé dans cette complexité fonctionnelle du système tout en s'exécutant en background. L'ouverture de cette porte dérobée a deux raisons majeures. D'une part, étant donné que l'accès à un système informatique est non moins fastidieux, le backdoor permet le maintien d'accès au système compromis pour une prochaine intrusion sans passer par toutes les étapes de l'attaque. D'autre part, elle a pour but de collecter les informations afin de surveiller le comportement de l'utilisateur au fil du temps, intercepter des paquets sur le réseau de la victime voire l'enregistrement des frappes du clavier (Hoglund & Butler, 2006). D'où, la mise en place d'un programme actif qui peut assurer ces besoins. Pour y parvenir, l'on fait recours à la programmation réseau.

2.3. Des notions du réseau informatique

Nous avons le devoir de rappeler que les réseaux sont nés du besoin profond de transporter une information d'un point à un autre. À l'origine, cette communication s'est faite directement par l'homme, comme dans le réseau postal, ou par des moyens sonores ou visuels. Il y a un peu plus d'un siècle, la première révolution des réseaux a consisté à automatiser le transport des données. Et, aujourd'hui, la communication réseau s'effectue à travers les couches d'abstraction réseau, les protocoles (que ce soit du type TCP ou UDP) et des sockets (Pujolle, 2001). Soient deux utilisateurs ***X*** et ***Y*** utilisant chacun son ordinateur, les deux PC, celui de ***X*** et celui de ***Y***, établiront une communication en réseau si et seulement si le premier connaît l'adresse IP du second et vice versa, et à condition que les deux utilisent un port ouvert ou libre et qu'ils exploitent ensemble un même protocole de transmission des données soit UDP soit TCP.

2.3.1. Adresse IP: Identifier un terminal sur le réseau

L'identification de chaque machine en réseau est faisable grâce à l'adresse logique. Cette adresse représente l'ordinateur dans le réseau et utilise le protocole IP qui est le protocole de base du réseau internet qui signifie Internet Protocole. Ce protocole sert d'interconnexion des réseaux (Pujolle, 2001). Il est de deux types ou générations. La génération d'IPV4 (IP version 4) et celle d'IPV6 (IP version 6). En revanche, l'adresse IP, en général, se subdivise en trois catégories: la première c'est l'IP interne, appelé loopback qui permet à la machine de communiquer à soi-même et qui sert à la pratique des tests. C'est l'adresse ***127.0.0.1***. La deuxième c'est l'IP du réseau local. En fait, dans un réseau, sans passer par Internet, les ordinateurs peuvent communiquer entre eux à travers les adresses locales de chaque PC. La dernière catégorie, c'est l'IP Internet. Pour communiquer avec n'importe quelle machine dans le monde, on doit exploiter l'IP Internet, cette adresse est souvent offerte par le Fournisseur d'Accès à Internet. Néanmoins, la connaissance de l'adresse IP d'un ordinateur dans un réseau ne suffit pas pour lancer la communication, car dans chaque machine, il y a plusieurs portes d'entrée. Pour éviter l'errance dans un PC, il faut savoir comment accéder à ces portes d'entrée qui sont appelées en langage informatique les ports logiques (Pujolle, 2014).

2.3.2. Les Ports logiques: Différents moyens d'accès à une machine

Dans un réseau informatique, les ordinateurs connectés reçoivent divers messages simultanément. Un utilisateur, en consultant son compte de messagerie peut en même temps downloader des fichiers dans un serveur. Pour éviter le désordre et assurer l'organisation des données, la notion des ports est de mise. En effet, le port assure l'accès à chaque service de la machine. Le port est identifié par un entier qui varie de 0 à 65536. On trouve généralement deux types de port. Les ports réservés et les ports libres. Les premiers sont ceux identifiés par les entiers inférieurs à 1024. Ces ports sont réservés à certains services de chaque machine. Par exemple, le port 80 réservé à la navigation sur le web, le port 110 à la réception des mails, 21 à l'envoi et la réception des fichiers dans un serveur des fichiers, etc. Ainsi, si l'on veut développer un logiciel qui communique en réseau, il faut utiliser un port qui soit compris entre 1024 et 65536. Certes, le choix de ce port permettra la communication entre machines. Toutefois, cette communication ne sera établie que dans la mesure où ces machines utilisent un langage commun de communication, c'est-à-dire, un même protocole (Pujolle, 2001).

3. Méthodes et matériels

Cette recherche a utilisé l'étude de cas (case study) comme approche. L'étude de cas est une approche empirique qui examine un phénomène dans son contexte réel, en particulier lorsque les frontières entre le phénomène et le contexte ne sont pas clairement évidentes. En ce sens, les études de cas sont la stratégie privilégiée pour répondre aux questions fréquemment posées sur le comment et le pourquoi (Kilani & Kobziev, 2016). En général, il y a trois raisons de choisir l'étude de cas comme approche lors d'une recherche dans le domaine des systèmes d'information: a) l'étude de cas permet au chercheur d'étudier les systèmes d'information dans leur cadre naturel et de générer des théories à partir de la pratique ; b) l'étude de cas permet au chercheur de répondre aux questions comment et pourquoi pour obtenir des informations plus explicites ; c) l'étude de cas permet au chercheur de se rendre compte de la nature et de la complexité du processus qui se déroule (Onatu, 2013).

Sur ce nous avons examiné la façon dont le système d'exploitation à noyau NT fonctionne et avons identifié certaines de ses failles afin de proposer des solutions comment se protéger contre les attaques qui peuvent compromettre ces vulnérabilités.

4. Résultats et discussion

4.1. Principe du fonctionnement d'un Rootkit

L'utilité du rootkit est d'une grande importance dans la situation où une brèche du système touché doit être maintenue (Hoglund & Butler, 2006). Dans ce sens, le rootkit offre deux principales fonctions à l'attaquant le contrôle total à distance et l'écoute. Dans son fonctionnement intrinsèque, il effectue des modifications. Alors que dans la plupart des cas, un logiciel est conçu pour prendre des décisions d'après des données très spécifiques; un rootkit peut modifier ces décisions pour qu'il fasse des choix incorrects. Ces modifications sont multiples telles que le patching, les œufs de pâques, le spyware, la modification de code source (Hoglund & Butler, 2006), etc. En général, le déploiement du rootkit s'effectue une fois qu'il y a eu exploit d'une faille logicielle. Pour que ce dernier ne soit pas détecté, il requiert l'accès au noyau car le rootkit contient un ou plusieurs programmes lancés au démarrage du système (Hoglund & Butler, 2006). Lorsqu'il est bien conçu et inséré dans le noyau, le rootkit est capable de contourner n'importe quelle mesure de sécurité tel qu'un système firewall ou un IDS en s'appuyant sur les deux méthodes de contournement: la méthode passive et la méthode active. Dans les attaques basées sur les méthodes actives, le rootkit apporte des modifications au matériel et au noyau tout en visant l'infiltration du système et en trompant le logiciel de détection d'intrusion. Ces types d'attaques prévoient certaines mesures pour désactiver le logiciel HIPS. Par contre, les attaques passives concernent la dissimulation des données stockées et transférées. L'exemple parlant est la technique de chiffrement des données avant leur stockage sur le système de fichier (Hoglund & Butler, 2006).

4.1.1. Mode de fonctionnement intrinsèque du rootkit

Etant donné que le rootkit est un logiciel pour l'attaque dans un réseau et qu'il exige l'établissement de connexion entre le terminal de l'attaquant et celui de la victime, sa mise en place nécessite l'acquisition des connaissances relatives à la programmation réseau. Certes, pour que deux programmes se communiquent en réseau, il faut un travail complexe qui va de la connaissance d'un langage de programmation aux connaissances des fonctionnalités réseau. Et, les programmes malveillants se servent des accès réseau vulnérables pour infecter les ordinateurs. Ces programmes exploitent des failles de sécurité sur les systèmes d'exploitation (Annicette, et al., 2007). Et, son fonctionnement, hormis l'aspect topologique, c'est-à-dire du réseau, recours à la technologie multithreading. En effet, la programmation multithreading permet d'exécuter plusieurs tâches simultanément dans un même processus en vue de pouvoir s'affranchir de la simple file d'exécution à la création d'une nouvelle file d'exécution concourante de la première. Le multithread permet d'optimiser le rendement global du système, sans accélérer nécessairement le temps d'exécution propre à chaque tâche (Magoules & Roux, 2013).

En outre, le Rootkit est aussi appelé Remote Access Trojan. C'est une technologie qui n'est pas nécessairement malveillante car elle peut être exploitée d'une manière légale notamment pour faire la télémaintenance ou le monitoring. Et, pour qu'il soit fonctionnel, le Rootkit doit comporter deux parties: un Client et un Serveur. Entre ces deux programmes, il y a une connexion qui doit être établie à partir des adresses IP en passant par un port d'écoute. Les types de connexion à établir peuvent être directes ou reverse connexion. Le premier type consiste à créer une connexion à partir d'un client vers le serveur. Tandis que le second crée une connexion qui part du serveur vers le client (Magoules & Roux, 2013).

4.1.2. Reverse connexion

La configuration d'un pare-feu consiste à rédiger des règles propres à déterminer les paquets autorisés et les paquets non autorisés; chaque paquet est caractérisé par quelques paramètres. Ces paramètres du pare-feu permettent d'identifier le type de communication auquel appartient le paquet (Bloch & Wolfhugel, 2009); ceci pour éviter le problème qui peut surgir lors de la demande de connexion par le client. En ce sens, lorsque l'attaquant peut vouloir lancer l'attaque à travers le rootkit, le pare-feu peut bloquer la communication entrante. Pour bypasser les pare-feu, les attaquants utilisent souvent la technique de reverse connexion. Le principe de ce type de connexion est que l'adresse IP du client doit être déterminée à l'avance dans le serveur se trouvant chez la victime. On parle ainsi de l'utilisation d'une IP statique ou des services no-ip.com. Dès que le serveur se lance, il cherchera à se connecter au client dont l'adresse IP a été paramétrée. Du coup, si le client est actif, il accepte la communication. L'avantage est que si le serveur se trouve dans un local area network (LAN) derrière un routeur muni d'un firewall, les communications peuvent ne pas être bloquées, car les firewalls sont parfois configurés pour autoriser les communications sortantes et filtrer celles entrantes. En sus, bien que le rootkit soit une technologie que certains alignent dans la catégorie des malwares, il n'est pas fonctionnellement malveillant, mais il peut être employé par des programmes qui le sont (Hoglund & Butler, 2006). Il est utilisé lorsque l'attaquant veut demeurer dans un système lors de l'attaque. Le rootkit, nous l'avons dit ci-haut, opère d'une façon dissimulée et est une modification frauduleuse des composantes d'un système, il vise à compromettre son intégrité et à changer son fonctionnement normal et habituel.

4.1.3. De l'infiltration du kernel Windows

De prime abord, nous pouvons dire que l'ordinateur est constitué du matériel composé de fils, circuits, transistor, etc., et qu'il est inutilisable sans un logiciel constitué de programmes. Son âme serait donc le système d'exploitation qui est logiciel ayant pour finalité: facilité l'utilisation de l'ordinateur (Bloch, 2013). Au cœur de ce software se trouve un noyau, appelé en anglais kernel. Il existe plusieurs types de système d'exploitation, chacun avec son noyau. Chaque noyau a ses avantages et ses inconvénients. Dans le cas échéant, nous partons du modèle Windows, avec son noyau New Technology (NT). En fait, Windows NT est une version de système d'exploitation à destination des stations de travail et des serveurs qui a sa structure propre bien que reprenant la majeure fonction des systèmes d'exploitation (Microsoft, 2014).

a) Le noyau du système Windows

Le noyau ou Kernel en anglais est la couche la plus stratégique qui a l'accès total du Système d'exploitation. Il joue un rôle intermédiaire entre le matériel et les logiciels applicatifs. Il a pour fonction la gestion des processus, l'accès aux fichiers, la gestion de la mémoire et la sécurité du système. L'on peut dire que le noyau est l'ultime gérant du système qui se charge d'imposer des restrictions entre les différents processus en cours et différentes tâches. Sous Windows, le noyau accorde des permissions et parvient à isoler les plages mémoires des processus (Hoglund & Butler, 2006). En revanche, l'appel système est une fonction du noyau utilisée par les programmes exécutant dans l'espace utilisateur. Les appels systèmes servent aux manipulations des fichiers au niveau du système des fichiers avec des commandes telles que Open, Read, Write, Close, etc. et aux allocations et désallocations. Un exemple d'appel système pour la suppression d'un fichier en VB .NET c'est ***My.Computer.FileSystem.DeleteFile( )***.

L'instruction ci-dessus est une méthode qui accède au système de gestion des fichiers pour supprimer le fichier dont le nom est passé en paramètre de la méthode sus-évoquée. Signalons que l'actuel noyau Windows NT garantit la sécurité du système d'exploitation

par la notion des privilèges et des Access Control List (ACL). Ce noyau a des composants relatifs à l'accès du système d'exploitation et les Ring sont le niveau de privilège. Le fichier ntoskrnl.exe contient le noyau de Windows NT. Le fichier ***win32k.sys*** contient le noyau graphique, c'est-à-dire le gestionnaire de fenêtrage. Dans le mode utilisateur, le fichier ***ntdll.dll*** est une librairie qui contient tous les appels systèmes non graphiques et, l'ensemble de ces appels systèmes s'appelle l'API native. En effet, les programmes qui fonctionnent uniquement avec la librairie ntdll.dll sont dits natifs (Microsoft, 2014).

Le Ring 0 appelé aussi mode noyau ou superviseur est exploité seulement par le Kernel et par les pilotes de périphériques. C'est ainsi qu'un programme s'exécutant en Ring 0 s'octroie l'avantage d'utiliser toutes les instructions fournies par le processeur. Tandis que les autres applications s'exécutent au niveau du Ring 3 appelé aussi mode utilisateur. Certes, une application s'exécutant à ce niveau ne peut pas accéder à certains registres ou instructions du processeur. Certes, l'on croirait que cette labyrinthe architecturale de Windows NT serait une forte garantie sécuritaire. Toutefois, en apportant quelques changements au code dans cette partie du noyau du système, un attaquant peut éliminer tous les mécanismes de sécurité (Hoglund & Butler, 2006) afin d'accéder au Ring 0 et de s'en prendre au système d'exploitation voire à tout le système d'information. Cela étant, nous pouvons affirmer que le Noyau Windows, bien que sécurisé, est une source potentielle de risque puisqu'il est en quelque sorte vulnérable en ce sens qu'il suffit simplement d'un détournement des drivers pour constater que Windows NT est vulnérable. Le risque encouru par un système est lié de manière étroite à la me vulnérabilité et aux failles qui le touchent, mais également aux contre-mesures mises en œuvre (ACISSI, 2009).

Les ingénieurs informaticiens ont remarqué que Microsoft, avec des classes ***.Net*** qu'il offre aux développeurs, a rendu le Noyau Windows fragile et que tout geek en informatique peut être à mesure d'implémenter ces classes réseaux telles que la classe ***TcpClient*** qui permet de se connecter au Socket en écoute, et envoyer et recevoir des données du Socket au moyen d'un des constructeurs de ladite classe ***[Constructeurs: TcpClient(), TcpClient(AdressFamily), TcpClient(IpEndPoint), TcpClient(String, Int32)]***, la transmission des données, au moyen d'un objet de type ***NetworkStream*** et la fermeture de la connexion, au moyen de la méthode Close (Lebrun, 2013). Aussi est-il qu'avec la classe ***TcpListener*** qui gère l'attente de connexion TCP, un hacker peut être doté d'une balise nocive que peut lui fournir cette classe à travers des méthodes d'écoute et d'acceptation des demandes de connexions entrante en mode blocage synchrone (Lebrun, 2013).

b) De la vulnérabilité de Windows NT

Aujourd'hui et même demain, des nombreuses vies humaines dépendent de l'ordinateur. Assurer la sécurité de l'ordinateur devient l'effort primordial de celui qui est soucieux de se protéger contre les menaces extérieures. Etant donné qu'il est possible de modifier le code du noyau et d'effectuer certains appels systèmes, le système d'exploitation présente une série de vulnérabilités exploitables. Bien que l'accès aux registres du processeur soit l'unique apanage du noyau et des pilotes des périphériques, le contournement des modules du noyau ou des drivers peut servir aux opérations de dissimulation des fichiers et de dissimulation des clés de registre. D'où, la répartition de code malveillant entre divers drivers pour faciliter l'attaque. Cette dissimulation s'effectue grâce aux techniques DKOM, techniques dont se sert le noyau pour sa propre gestion interne. Il est question dans cette technique de connaître la manière dont le noyau utilise les objets, comment l'objet change d'aspect après un changement majeur de la version d'un système d'exploitation et de savoir quand l'objet est utilisé à l'état du système. Pour cela, on utilise la structure OSVERSIONINFO ou OSVERSIONINFOEX pour obtenir les informations relatives au SE dont on veut modifier le DKOM. Après avoir identifié la version, on ajuste par la

suite les Offsets de ces structures. On peut aussi connaître la version d'un système d'exploitation à travers les clés de registre (Hoglund & Butler, 2006).

On procède de la manière suivante: on va dans exécuter puis on tape la commande «Regedit» en passant par le chemin: ***HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\CurrentVersion***. Il s'ensuit l'établissement de la communication avec un driver au travers le mode utilisateur pour dissimuler le processus qu'on veut cacher. Les processus dans Windows NT sont décrits dans les objets ***Executive*** et ces objets sont référencés par ***taskmgr.exe*** et par d'autres outils qui listent les différents processus s'exécutant sur la machine. Parvenir à cacher un processus demande la connaissance du fonctionnement de la structure EPROCESS afin de changer la valeur du pointeur FLINK du bloc EPROCESS qui le précède et le pointeur BLINK de celui qui le suit pour maintenir la cohérence de la liste chaînée (Hoglund & Butler, 2006). De plus, la technique de chaînage de drivers est de mise pour intercepter les périphériques du système puisque tous les périphériques matériels ont des chaînes des drivers. Sur ce, la connaissance de la façon dont le noyau gère les drivers est la clé de voute pour pouvoir chaîner un driver. La preuve éloquente de cette technique est le ***sniffing*** du clavier, appelée aussi Keylogger. Le Keylogger intercepte les frappes du clavier et les écrit soit dans un fichier texte soit le transfert par mail ou messagerie selon les besoins du sniffeur. Le chaînage de drivers représente un moyen fiable et robuste d'intercepter et de modifier des données dans un système (Hoglund & Butler, 2006). L'exploitation de ces diverses techniques constitue une grande menace pour le noyau Windows NT.

4.2. Se protéger contre l'infiltration des rootkits

Comment se protéger contre les infiltrations des rootkits ? Cette question a trouvé sa réponse dans cette section des résultats. Le déploiement massif des réseaux informatiques est à l'origine des nombreuses inquiétudes dans le monde du computer. Ces inquiétudes s'insèrent dans un cadre de complexité qui va du déploiement du réseau à sa maintenance tout en passant par des voies efficaces de la sécurisation de ce dernier. Cette complexité a poussé les entreprises et les ingénieurs IT à développer des outils qui permettent d'assurer la protection et la disponibilité de l'information dans les réseaux. Certes, les raisons d'intrusion dans un système informatique sont multiples. Toutefois, les mesures de sécurité élevées dans ces dernières décennies n'offrent aucune chance d'accéder aux systèmes d'information sans autorisation. Par contre, certains internautes voire nombreux informaticiens non avisés se soucient peu de la confidentialité des données. Alors qu'il est facile d'éradiquer un virus qui est plus visible par les antivirus, il existe des dangers beaucoup plus discrets et insidieux qui se dissimulent facilement dans un système à l'insu des utilisateurs voire des administrateurs (Edigo, 2010).

Une des solutions pouvant pallier ce problème d'intrusion indue est l'utilisation des logiciels de détection d'intrusion tels que Snort, Tiger, Logcheck et les antivirus heuristiquement puissants afin de préserver la sécurité des données dans les réseaux informatiques. Mais, cela ne suffit pas. Car rien ne sert à se munir des moyens techniques puissants si l'utilisateur de ces moyens est ignorant de l'existence des rootkits et de techniques pour les contrer puisque rien ne sert d'avoir des protections informatiques infaillibles si une personne interne à votre système permet à son insu à un attaquant de déjouer toutes ces protections. L'humain est une faille qu'il faut surveiller (ACISSI, 2009). En plus, la meilleure façon serait de contrôler régulièrement toutes les tâches qui tournent dans sa machine pour identifier les processus anormaux. Avec la commande ***taskmgr***, on peut afficher son gestionnaire de tâches pour vérifier. L'image ci-dessous illustre les différentes tâches en exécution dans un PC:

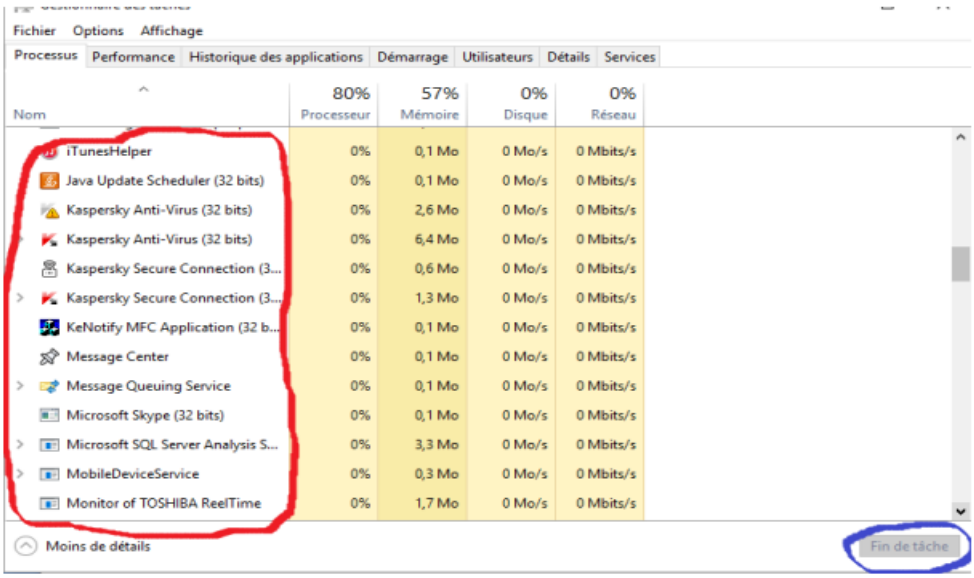

Figure 1. Illustration des tâches du système en exécution

L'image ci-dessus illustre les applications qui démarrent ensemble avec le système d'exploitation du PC dont le chemin du dossier **Démarrer** est repris ci-haut. Avec ce répertoire, chaque utilisateur de machine saura quelle application n'est pas crédible et qui s'auto lance. En plus de la technique de vérification du dossier **Démarrer** et du contrôle du gestionnaire de tâches, on peut se protéger en vérifiant régulièrement sa base de registre. En fait, la base de registre est un fichier qui contient une arborescence dans laquelle sont classés tous les paramètres par critère. Chaque critère correspond à une branche de l'arborescence et chaque élément de configuration est situé tout au bout d'une branche dans ce qu'on appelle clé de registre. Pour s'apercevoir des clés des applications s'exécutant dans son PC, on tape d'abord la commande ***regedit*** sur son invite de commande. Puis, dans l'interface qui va apparaître, on navigue sur cette arborescence: ***HKEY_CURRENT_USER/SOFTWARE/Microsoft/Windows/CurrentVersion/Run***. Ce qui donne cette image avec des clés en exécution:

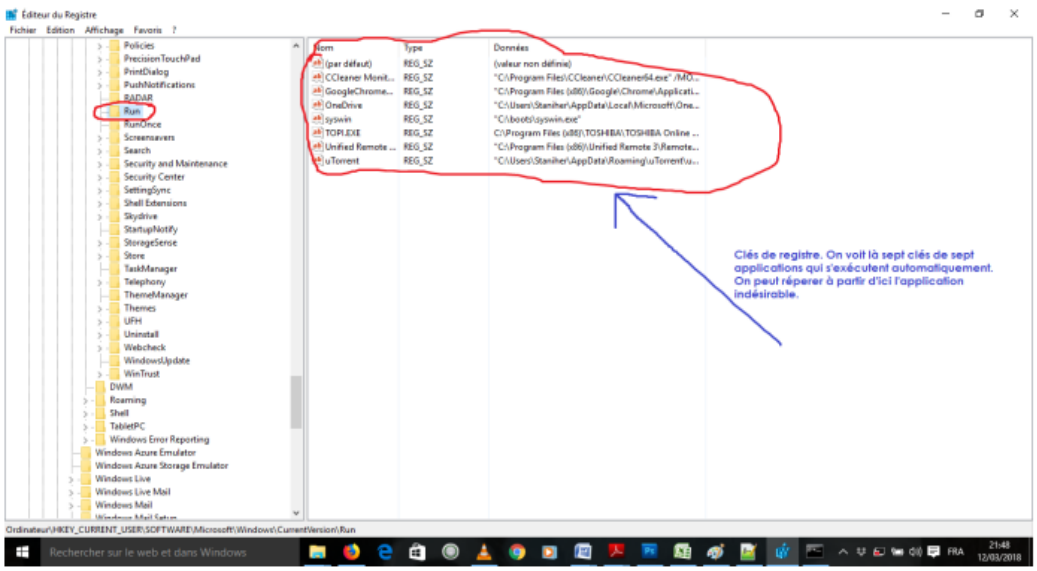

Figure 2. Illustration d'un virus parmi les clés des applications en exécution

Parmi ces sept clés, on voit bien l'existence d'un virus nommé ***syswin***. Ce virus rend la taille tous les exécutables d'un PC à zéro octet, à partir d'ici, on peut alors le supprimer.

En revanche, disons qu'il existe un nombre important des méthodes de protection contre ce danger, source de conflits actuels qu'est le rootkit. Mais, pour des raisons de simplicité et d'accessibilité à d'autres formes de sécurité et de prévention sécuritaire, nous avons retenu ces trois façons de protection: vérification du dossier **Démarrer** de son PC, contrôle des tâches et processus s'exécutant dans son gestionnaire de tâches (pour accéder à ce gestionnaire, il suffit de taper, sur son invite de commande: ***taskmgr***) et de la consultation de la base de registre (avec la commande ***regedit***).

Pour des entreprises, elles doivent savoir que si la sécurité des systèmes d'information a été absolument bouleversée durant ces dernières décennies par l'évolution exponentielle de l'Internet et ses applications, cette question de sécurité ne saurait s'y réduire. En effet, c'est un problème épineux dont les aspects techniques ne sont qu'une infime partie. Les aspects sociaux, juridiques, psychologiques et organisationnels sont aussi des facteurs on ne peut plus importants. À elles, les entreprises, nous pouvons réitérer cette recommandation de Bloch et Wolfhugel (2009) qu'il serait inutile de se préoccuper de sécurité sans avoir défini ce qui est à protéger. En effet, toute organisation désireuse de protéger ses systèmes et ses réseaux doit déterminer son périmètre de sécurité. Voilà une des pistes de solution qui peut prévenir les conflits qui peuvent découler des vols des informations, d'espionnage et de contrôle illicite des systèmes dans les réseaux informatiques.

5. Conclusions

Les activités de l'homme d'aujourd'hui se réalisent généralement aux travers des outils des NTIC. Nombre de personnes font recours aux réseaux informatiques pour accomplir leurs tâches quotidiennes. En effet, les réseaux informatiques sont nés du besoin de relier des machines distantes à un site central puis des ordinateurs entre eux et enfin des terminaux, telles que stations de travail ou serveurs (Pujolle, 2004) afin d'assurer l'échange des informations. Néanmoins, dans les réseaux, il existe de nombreux risques qui peuvent amener un tiers à l'insécurité sans détour puisque l' internet qu'est le réseau des réseaux est le lieu où ouvert à tout le monde. Cette ouverture constitue le risque imminent de tout système qui se connecte aux réseaux. Les menaces dans les réseaux informatiques sont on ne peut plus nombreuses et cela varient selon des catégories. Il existe, de ce fait, des virus informatiques, des vers, des backdoor, des rootkit, ... Chacune de ces menaces a un objectif particulier selon son fonctionnement. Toutefois, de ces différentes menaces, l'on retiendra que le rootkit qui est une technologie en vogue dans ces dernières décennies s'avère dangereux pour les systèmes informatiques car il requiert l'accès au noyau et contient un ou nombreux programmes qui peuvent être lancés lorsque le système démarre (Hoglund & Butler, 2006). Le rootkit permet l'infiltration du noyau, dans le cas échéant le noyau Windows. En effet, ce noyau contient des vulnérabilités et l'on peut même partir des certaines classes de programmation réseau qu'offre le langage VB. Net aux développeurs pour mettre en place des outils d'attaque des systèmes.

Toutefois, il est possible de détecter les menaces présentes dans son PC et de les éradiquer par soit l'installation des intrusion detection systems (IDS), soit par le contrôle régulier de son gestionnaire de tâche, soit par la vérification des exécutables présents dans le dossier Démarrer de son ordinateur, soit par la consultation des clés de registres en passant par la commande regedit. Ceci est une façon de se prévenir contre les conflits qui surgissent lors d'une attaque cybercriminelle. En définitif, notre souci a été d'offrir un instrument d'éveil de conscience aux internautes et aux utilisateurs finaux des systèmes informatiques, voire aux informaticiens ignorants l'existence de la technologie de rootkit, technologie qui a mis en brèche la sécurité de plusieurs systèmes et qui constitue une véritable source de conflit interpersonnel, interentreprises et inter-états. La connaissance de ce type de logiciel, encore méconnu dans la sphère internet les aiderait à sortir de l'impasse de cybercriminalité qui gangrène le monde aujourd'hui. Puisqu'il faut d'abord commencer par identifier la menace potentiel. Il faut connaître son ennemi, ses motivations et prévoir la façon dont il procède pour s'en protéger et limiter les risques d'intrusion (ACISSI, 2009).

Contributions: Conceptualisation, M.H.N.; méthodologie, M.H.N.; validation, M.H.N.; investigation, M.H.N.; ressources, M.H.N.; traitement des données, M.H.N.; écrire le manuscrit, M.H.N.; visualisation, M.H.N.; supervision, M.H.N.; correction du manuscrit M.H.N. L'auteur a lu et approuvé la version publiée de ce manuscrit.

Sponsor financier: Cette recherche n'a reçu aucun soutien financier.

Disponibilité des données: Les données ne sont pas disponibles.

Remerciement: Non applicable.

Conflits d'intérêt: L'auteur déclare aucun conflit d'intérêt.

Références

1. ACISSI, Sécurité informatique. Ethical Hacking. Apprendre l'attaque pour mieux se défendre, Collection Expert IT, ENI, Paris, 2009.
2. Annicette D., et al., Sécurité Windows Vista. Le guide complet, Edition Micro Application, 1ere édition, Paris, 2007.
3. Bloch L. & Wolfhugel C., Sécurité informatique. Principes et méthode à l'usage des DSI, RSSI et administrateurs, 2 e édition, Eyrolles, Paris, 2009.
4. Bloch L., *Les Systèmes d'exploitation des ordinateurs. Histoire, fonctionnement, enjeux*, Edition Vuilbert et Laurent Bloch, Paris, 2013.
5. Edigo, *Le piratage de A à Z*, Edigo, 2010.
6. Filiol E. & Fizaine J-P., "Les codes malveillants sous Unix/Linux: La menace n'est pas fantôme", *GNU/Linux Magazine France*, 2007.
7. Hoglund G. & Butler J., *Rootkit. Infiltration du noyau Windows*, Edition Campus Press, Paris, 2006.
8. Kilani M.A. & Kobziev V., "An Overview of Research Methodology in Information System (IS)", *Open Access Library Journal*, Vol. 3, e3126, 2016. http://dx.doi.org/10.4236/oalib.1103126.
9. Lacombe E., *Sécurité des noyaux des systèmes d'exploitation. Thèse en vue de l'obtention du doctorat de l'Université de Toulouse*, INSA de Toulouse, 2009.
10. Lebrun T., "La programmation réseau en VB.Net", *Developpez.com*, 20 décembre 2013.
11. Magoules F. & Roux F-X., *Calcul scientifique parallèle. Cours, exemples avec OpenMP et MPI, exercices corrigés*, Dunod, Paris, 2013.
12. Microsoft, *Microsoft Windows. Présentation du noyau de Windows NT*, Version 1.0, 06 février 2014.
13. Moigne L.J-L., *La théorie du système général. Théorie de la modélisation*, Collection les classiques du Réseau Intelligence de la complexité, 2006.
14. Morley C., *Management d'un projet système d'information. Principes, techniques, mise en œuvre et outils*, 6e édition, Dunod, Paris, 2008.
15. Onatu G.O., "Building Theory from Case Study Research : The Unanswered Question in Social Sciences", *1st Global Virtual Conference*, Vol. 1, 2013.
16. Pujolle G., *Initiation aux réseaux*, Eyrolles, Paris, 2001.
17. Pujolle G., *Les réseaux*, 5 e édition, Eyrolles, Paris, 2004.
18. Pujolle G., *Les réseaux*, 8e édition, Eyrolles, Paris, 2014.
19. Ury W., *Comment négocier la paix. Du conflit à la coopération chez soi, au travail et dans le monde*, Nouveaux horizons, Paris, 2001.