Cours de Sécurité et Surveillance des Réseaux :

IUT de Villetaneuse — Licence Professionnelle ASUR

Laure Petrucci

6 mars 2011

Table des matières

Chapitre 1

Principes généraux et historique

1.1 Introduction

L'*objectif de la sécurité des systèmes et réseaux informatiques* est d'éviter que des acteurs malveillants puissent accéder à des données ou au système lui-même. Dans ce cadre, on distingue trois étapes :

- la *prévention* consiste à prendre des mesures rendant l'accès difficile pour un acteur malveillant ;
- les mesures de *détection* permettent de s'apercevoir qu'une action anormale a eu lieu, par qui et quoi elle consiste ;
- la *réaction* vise à réparer du mieux possible les dommages causés et mettre en place des mesures préventives pour que cela ne se reproduise pas.

1.2 Terminologie

Un système informatique peut être sujet à des attaques de la part d'entités malveillantes. Celles-ci peuvent profiter de failles pour s'introduire dans le système visé, ou surveiller les échanges de messages sur le réseau pour apprendre, détourner, falsifier des informations qui y transitent.

Définition 1 (Vulnérabilité) *Une* vulnérabilité *est un faiblesse ou une faille du système pouvant être exploitée pour mettre en danger sa sécurité.*

Définition 2 (Intrusion) *L'*intrusion *dans un système informatique est un accès illégal (non prévu) à ce système.*

Pour éviter que des messages soient divulgués, on leur applique des procédures de codage secret.

Définition 3 (Cryptographie) *La* cryptographie *a pour but de protéger des messages par des codages secrets.*

A l'inverse, lorsqu'un message secret est intercepté, des techniques permettent d'essayer de le décoder.

Définition 4 (Cryptanalyse) *La* cryptanalyse *vise à déchiffrer des messages secrets sans posséder les clés nécessaires.*

Les deux notions que sont la cryptographie et la cryptanalyse sont deux aspects complémentaires de la crypologie.

Définition 5 (Cryptologie) *La* cryptologie *est la science du secret.*

Contrairement aux techniques précédentes, la stéganographie consiste en la dissimulation sans pour autant rendre le message caché inintelligible pour quelqu'un qui saurait comment le trouver.

Définition 6 (Stéganographie) *La* stéganographie *vise à cacher le contenu d'un message pour qu'il soit difficilement visible.*

1.3 Historique

Dans cette section, nous relatons quelques anecdotes ou faits historiques ayant trait à la sécurité ou aux intrusions. De nombreuses cas similaires ont émaillé le cours de l'histoire.

Guerres médiques, 480 av. JC La préparation de troupes pour la guerre est inscrite sur le bois d'une tablette de cire qui est ensuite recouverte d'une nouvelle couche de cire, divulguant ainsi les intentions de l'attaquant.

Grèce antique, 5ème siècle av. JC Après avoir rasé le crâne d'un esclave, les grecs tatouaient un message sur le cuir chevelu, puis attendaient que les cheveux aient suffisamment repoussé pour envoyer l'esclave délivrer le message.

Rome antique, 1er siècle av. JC Pline l'ancien décrit la fabrication d'encre sympatique. De nos jours, le jus de citron est souvent utilisé par les enfants pour écrire des « messages secrets ».

XVIème siècle, grille de Cardan Largement utilisée par Richelieu, la grille de Cardan est un carton percé de trous qui laissent apparaître les lettres appartenant à un message caché dans un texte, et en masque le reste.

Seconde guerre mondiale La technique du micropoint est utilisée par les allemands : une photo d'une page est réduite en l'image d'un point sur une lettre d'un texte.

1.4 Propriétés à garantir

Pour assurer le bon fonctionnement d'un système, il faut garantir au mieux certaines propriétés. Celles-ci sont explicitées dans cette section.

1.4.1 Intégrité

Définition 7 (Intégrité) *L'*intégrité *assure que les données manipulées ou mises en œuvre ne sont pas altérées au cours du fonctionnement du système.*

La notion d'intégrité intervient à plusieurs niveaux :
- Dans des *communications*, via un réseau par exemple, on doit garantir que le message reçu est bien celui qui a été envoyé. En effet, un attaquant peut altérer le message, et ainsi transmettre des informations erronées au destinataire.
- Au sein du *système d'exploitation*, l'exécution d'un processus peut être détournée par une attaque, conduisant à l'exécution d'un code différent.

Les attaques portant atteinte à l'intégrité des données ou du systèmes peuvent par conséquent avoir des conséquences très graves.

1.4.2 Confidentialité

Définition 8 (Confidentialité) *La* confidentialité *garantit que seules les personnes ou entités habilitées à prendre connaissance du contenu d'un message peuvent le faire.*

Cette propriété est particulièrement importante dans le cadre de communications réseau. En effet, imaginons que deux personnes communiquent pour échanger une information confidentielle. Il est clair qu'elles finiront toutes deux par connaître le contenu du message. Si ce message est

intercepté par un tiers, et que la confidentialité n'est pas assurée, ce tiers peut prendre connaissance du contenu du message et l'utiliser à des fins malveillantes (par exemple du chantage).

De même, une donnée stockée sur une machine peut être confidentielle. Le système d'exploitation doit donc être configuré au mieux pour garantir que les données ne puissent être consultées que *par les utilisateurs et les processus* autorisés.

1.4.3 Authenticité

Définition 9 (Authenticité) *L'*authenticité *d'un message garantit l'identité de l'émetteur de ce message.*

Si la propriété d'authenticité est satisfaite, un intrus ne peut pas envoyer un message en se faisant passer pour quelqu'un d'autre, que ce soit un utilisateur ou une entité matérielle.

1.4.4 Non-répudiation

Définition 10 (Non-répudiation) *La* non-répudiation *assure à la fois que le destinataire d'un message ne peux pas prétendre ne pas l'avoir reçu, et que l'émetteur ne peux pas prétendre ne pas l'avoir envoyé.*

Cette notion est cruciale dans les systèmes de transactions bancaires.

1.4.5 Disponibilité

Définition 11 (Disponibilité) *La* disponibilité *assure que le service est toujours fonctionnel.*

Parmi les attaques classiques, certaines visent à rendre le système non-opérationnel, et donc non disponible. Ceci peut conduire par exemple à des pertes financières (clients mécontents de ne pas être servis).

Chapitre 2

Protocoles et chiffrement

2.1 L'échange sécurisé de données

Pour représenter l'échange de données entre deux processus, ou utilisateurs, on utilise le paradigme de deux personnes, *Alice* et *Bob*, où Alice souhaite envoyer un message à Bob. Ce message est secret et ne doit pas pouvoir être lu par une tierce personne. Par conséquent, un intrus, communément appelé *Charlie*, ne doit pas être en mesure de lire le message. De plus, Charlie ne doit pas avoir la possibilité d'altérer le message d'Alice, de le remplacer par le sien propre, ou de se faire passer pour Alice auprès de Bob.

Imaginons cet exemple avec un message transmis dans un colis postal sur lequel on peut poser des cadenas, et un facteur malveillant. Initialement, chaque personne dispose de son propre cadenas et de la clé associée. Une communication entre ces différents acteurs peut être schématisée comme dans la figure 2.1.

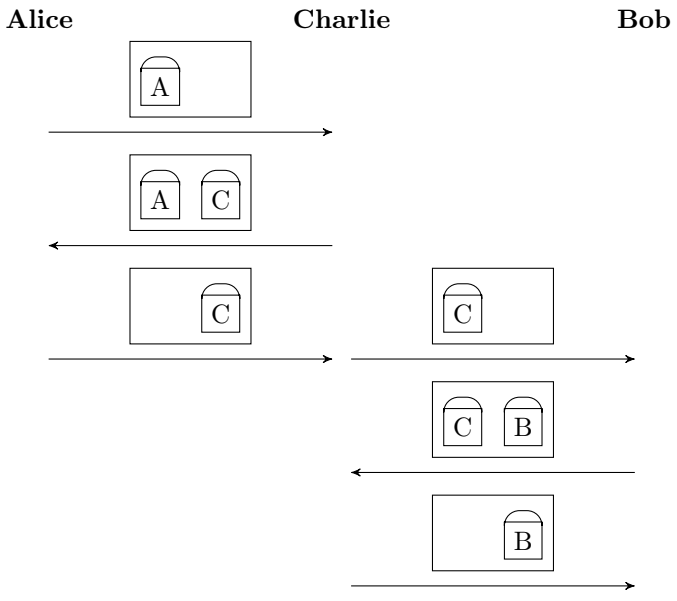

FIGURE 2.1 – Communication entre Alice et Bob, espionnée par Charlie

Dans ce scénario, Alice envoie son paquet avec son cadenas. Le facteur, Charlie, l'intercepte et pose également son propre cadenas, avant de renvoyer le paquet à Alice. Celle-ci croyant voir le cadenas de Bob, enlève le sien et renvoie le paquet. Charlie peut alors, en enlevant son cadenas,

regarder le message, puis le transmettre à Bob comme l'aurait fait Alice.

Cet exemple met en évidence plusieurs problèmes :

- Charlie connaît le contenu du message secret ;
- lorsque Charlie lit le message, rien ne l'empêche de le modifier ;
- Alice et Bob croient communiquer entre eux (les adresses des expéditeurs et destinataires sont correctes sur le colis) et ne soupçonnent pas leur facteur de malveillance.

Les techniques mises en place dans cet exemple simple donnent l'impression d'assurer une certaine sécurité, et pourtant ... Le célèbre protocole de Needham-Schröder [NS78] (voir section 2.3.2) a largement été utilisé avant qu'une faille ne soit trouvée [Low95]. Les techniques efficaces sont donc assez évoluées et complexes.

2.2 Utilisation de clés

Pour pouvoir garantir au mieux les propriétés de sûreté et de sécurité des communications, les messages circulant sur le réseau sont chiffrés.

Définition 12 (Texte clair) *Le message que l'on souhaite transmettre est qualifié de* texte clair. *Il est donc compréhensible par tout acteur souhaitant le lire.*

Pour éviter la compréhension du message, celui-ci est chiffré.

Définition 13 (Texte chiffré) *Le* texte chiffré *est obtenu à partir du texte clair par une opération de chiffrement.*

Comme expliqué dans les sections 2.3 et 2.4, différentes opérations de chiffrement sont disponibles. Elles se répartissent en deux principales catégories, utilisant des *clés symétriques* ou *asymétriques*. L'objectif est de faire transiter le *texte chiffré*, incompréhensible si l'on ne dispose pas d'outils ou de données particulières. Ces dernières sont par ailleurs fournies au récepteur du message qui peut ainsi le *déchiffrer* pour obtenir le *texte clair* d'origine.

2.3 Clés symétriques

2.3.1 Principe

Les cryptosystèmes à *clés symétriques* partagent la même clé pour le chiffrement et le déchiffrement. Celle-ci est donc connue à la fois de l'émetteur du texte chiffré et et de son récepteur. Lors de l'utilisation d'un tel mécanisme, encore appelé *à clé secrète*, les deux interlocuteurs doivent pouvoir se mettre d'accord sur la clé au préalable.

2.3.2 L'algorithme de Needam-Schröder

Le *(*protocole de Needhan-Schröder) est un protocole à clés symétriques introduit en 1978 par Roger Needham et Michael Schröder, longtemps utilisé avant qu'une faille ne soit découverte en 1995.

Ce protocole utilise une entité tierce, un serveur, S, en lequel les deux entités voulant communiquer, A (Alice) et B (Bob), ont confiance. Les différentes étapes successives du protocole sont les suivantes :

1. $A \longrightarrow S : A, B, N_A$: Alice indique au serveur qu'elle veut communiquer avec Bob, et lui transmet un *nonce*[1] N_A.

1. Un *nonce*, raccourci pour *number used once*, est un nombre aléatoire ou pseudo-aléatoire, généré pour éviter des attaques par rejeu d'anciens messages.

2. $S \longrightarrow A : \{N_A, K_{AB}, B, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$: le serveur partage une clé K_{AS} avec Alice, clé qu'il utilise pour communiquer avec elle. Le serveur et Alice sont les deux seules entités à connaître cette clé. Il's l'utilisent non seulement pour coder, mais aussi pour décoder les messages qu'ils échangent. Le serveur renvoie à Alice un message contenant le nonce qui vient de lui être transmis, montrant ainsi qu'il répond à son nouveau message, une clé K_{AB} qu'il a générée pour qu'Alice et Bob puissent communiquer, l'identité de Bob, et un message codé avec la clé K_{BS} partagée par le serveur et Bob. Ce message codé contient la clé K_{AB} que Bob devra utiliser pour communiquer avec Alice ainsi que l'identité de celle-ci.
3. $A \longrightarrow B : \{K_{AB}, A\}_{K_{BS}}$: Alice envoie à Bob le message codé avec la clé de Bob que le serveur lui a transmis.
4. $B \longrightarrow A : \{N_B\}_{K_{AB}}$: Bob envoie un nonce qu'il a généré, encodé avec la clé partagée entre Alice et Bob, montrant qu'il l'a bien reçue.
5. $A \longrightarrow B : \{N_B - 1\}_{K_{AB}}$: Alice effectue une opération simple sur le nonce que Bob lui a envoyé, et le revoie encodé avec la clé qu'elle partage avec Bob.

Ce protocole est sujet à une *attaque par rejeu* : un intrus peut intercepter le message $\{K_{AB}, A\}_{K_{BS}}$ envoyé par Alice à Bob, et ce dernier ne s'apercevant pas que la clé n'est pas fraîche, l'acceptera.

2.3.3 Principaux algorithmes

Les algorithmes a clés symétriques se fondent sur un principe d'itération de chiffrement par blocs des bits de message.

DES (Data Encryption Standard)

L'algorithme DES (*Data Encryption Standard*) a été développé par IBM au milieu des années 1970 et adopté par le gouvernement américain en 1977 comme standard de chiffrement. Il a été reconduit depuis tous les cinq ans jusqu'en 1999.

Les principales caractéristiques de DES sont les suivantes :
- messages clairs et chiffrés de 64 bits ;
- clé de 64 bits décomposée en 8 blocs de 8 bits comprenant un bit de parité pair ;
- le chiffrement d'un message est obtenu en 16 tours.

IDEA (International Data Encryption Algorithm)

L'algorithme IDEA (*International Data Encryption Algorithm*) a été développé en 1999, est breveté et commercialisé par la société suisse MediaCrypt. Les tours s'appuient sur les opérations suivantes :
- la multiplication d'entiers modulo un nombre premier p ;
- l'addition modulo n d'entiers compris entre 0 en $n - 1$;
- l'addition modulo 2 bit à bit de blocs de n bits.

AES (Advanced Encryption Standard)

L'algorithme AES (*Advanced Encryption Standard*) est issu d'un appel d'offres du 2 septembre 1997 visant à remplacer le standard DES. Le 20 août 1998, 15 algorithmes étaient déposés, dont seulement 5 restaient en lice un an plus tard. Des tests visant à mettre en défaut ces 5 algorithmes au travers de diverses attaques n'ayant pas abouti, ils furent départagés en 2000 par d'autres critères tels que la complexité algorithmique ou des critères d'implantation. L'algorithme retenu, Rinjdael, a été officiellement publié le 26 novembre 2001 et approuvé comme moyen cryptographique de protection de données sensibles non classifiées pour le gouvernement américain.

Les principales caractéristiques de l'algorithme AES sont les suivantes :
- codage et décodage de textes de 128 bits ;
- utilisation de clés de 128, 192 ou 256 bits ;

– le nombre de tours dépend de la clé : 10, 12 ou 14 tours pour des clés de 128, 192 ou 256 bits, respectivement ;
– opérations sur les octets et non sur les bits.

2.4 Clés asymétriques

Les cryptosystèmes à *clés asymétriques*, encore appelés à *clé publique*, utilisent des fonctions de cryptage et de décryptage différentes. La clé de chiffrement est connue de tous, donc *publique* tandis que celle de déchiffrement n'est connue que du récepteur, donc *privée*.

Avec un tel mécanisme, l'émetteur chiffre le texte avec la clé publique, mais seul le récepteur a la clé privée pour le déchiffrer.

Le cryptosystème RSA

Introduit en 1978 par Rivest, Shamir et Adleman, le cryptosystème RSA se fonde sur les propriétés mathématiques des nombres premiers. Il est en effet difficile de trouver les entiers p et q factorisant un nombre $n = pq$ lorsque p et q sont des nombres premiers de grande taille. Différents calculs à partir de ces entiers p et q permettent d'obtenir un couple (clé publique, clé secrète) ainsi que les fonctions de chiffrement et de déchiffrement associées.

Chapitre 3

Vulnérabilités, attaques et intrusions

3.1 Définitions

Les failles ou faiblesses présentes dans un système peuvent avoir diverses sources :
- une *erreur de conception ou de réalisation* (la plus fréquente) ;
- l'*environnement d'exécution* ;
- le *matériel*.

Définition 14 (Vulnérabilité) *Une* vlnérabilité *est une faille ou une faiblesse pouvant être exploitée pour compromettre la sûreté d'un système d'informations*

En général les faiblesses ou vulnérabilités peuvent être corrigées en appliquant des correctifs (*patchs*). Toutefois, la connaissance de vulnérabilités n'implique pas nécessairement la disponibilité de correctifs appropriés.

Définition 15 (Attaque) *Une* attaque *est une activité dirigée contre un système cible, visant à compromettre son intégrité, sa fiabilité ou sa confidentialité.*

Définition 16 (Intrusion) *Une* intrusion *est le résultat d'une attaque réussie et la compromission de la sûreté du système.*

Pour éviter les intrusions, il faut donc parer aux attaques en combinant plusieurs solutions de sécurité traitant des attaques de différentes natures.

3.2 Taxinomie des attaques

Une classification permet de définir le paysage des différents types d'attaques. Elle comporte plusieurs axes.

3.2.1 Cible de l'attaque

La *cible* de l'attaque peut tout aussi bien être matérielle (composants électroniques, mémoire, disques, etc.) que logicielle (systèmes d'exploitation, applications).

3.2.2 Vecteur de l'attaque

Le *vecteur* d'une attaque est la méthode utilisée pour pouvoir atteindre la cible. Parmi les différents vecteurs d'attaques, on trouve :

– l'utilisation d'un *logiciel malveillant* ou *malware*. Ceux-ci s'installent sur le système, en utilisent les ressources pour se dupliquer et se propager. Les *virus* profite de l'exécution d'un programme légitime pour s'exécuter et se reproduire. Les *vers* ne s'attachent pas à un programme, contrairement aux virus, mais profitent des faiblesses ou défauts de configuration pour espionner les activités du système, ouvrir de nouvelles failles ou attaquer une autre cible ;
– la *surcharge* du système ou *déni de service — DoS* exploite les limites matérielles et logicielles en submergeant le systèmes de requêtes, monopolisant ainsi toutes les ressources ;
– l'*exploitation d'une faille* du système est utiisée pour prendre le contrôle du système, usurper des identités ou rendre le service inopérant ;
– les *attaques protocolaires* (écoute de paquets, modifications de trames).

3.2.3 Exploitation de vulnérabilités

L'*exploitation de faiblesses* d'un système concerne différents types de vulnérabilités :
– de *configuration* : ouverture de ports, mots de passe par défaut, etc. ;
– d'*architecture* ou de *conception* : manque de contrôle d'accès, données non protégées, etc. ;
– d'*implémentation*.

Parmi les vulnérabilités d'implémentation, sont fréquentes :
– les *dépassements de tampon* ou *buffer overflow* qui écrivent dans la mémoire hors des limites allouées au processus, par exemple en modifiant l'adresse de retour d'une fonction licite ;
– les *situations de compétition* ou *race conditions* qui sont des situations dans lesquelles un attaquant modifie les données manipulées par le programme, par exemple pour *usurper une identité* ou *abuser de privilèges*.

Chapitre 4

Systèmes de détection d'intrusion

4.1 Généralités

Les *systèmes de détection d'intrusions* (IDS — Intrusion Detection System) ont été introduits en 1980, en partant du constat que les données d'audit contiennent de nombreuses informations sur le comportement et l'utilisation d'un système. Les *logs* permettent en effet de repérer *a posteriori* des comportements anormaux, qui peuvent être dûs à des bugs ou des intrusions.

Une exploitation des données des *mainframes* du gouvernement américain a conduit ensuite à créer des profils d'utilisateurs en fonction de leur utilisation des machines. L'adéquation entre les opérations effectuées et le profil permet ensuite de vérifier que ces opérations sont licites dans le sens où elles correspondent au comportement attendu de l'utilisateur. Cette approche a été implémentée dans le premier système de détection d'intrusions, IDES (Intrusion Detection Expert System).

En 1990, NSM (Network Security Monitor) utilise les événements circulant sur le réseau pour détecter des intrusions.

Définition 17 (Détection d'intrusion) *La* détection d'intrusion *consiste à* surveiller *et* analyser *les* événements *issus d'un système, d'un réseau, de données d'audit, dans le but de détecter des intrusions.*

Définition 18 (Système de détection d'intrusions) *Un* système de détection d'intrusions *est un logiciel ou un matériel automatisant la surveillance et l'analyse d'intrusions.*

4.2 Caractéristiques des IDS

4.2.1 Positionnement de l'IDS par rapport au système surveillé

Le *positionnement* de l'IDS par rapport au système surveillé est particulièrement important.

IDS réseau

Les *IDS réseau* (NIDS — Network IDS) sont positionnés sur le réseau et visent à protéger une grappe de machines en *analysant* les informations circulant sur le *réseau* à destination de ces machines. Ils sont en général présents sur un équipement dédié, et bien isolé des machines à protéger.

Les NIDS analysent les informations circulant sur le réseau à destination des machines protégées. Par exemple SNORT compare les paquets à un ensemble de règls contenues dans une base de données.

Un NIDS ayant une vision du trafic sur le réseau, il est particulièrement efficace contre des attaques de vers ou par surcharge conduisant à un déni de service. Ces attaques sont visibles sur

le réseau. Par contre, les attaques exploitant des vulnérabilités logicielles ne sont pas détectées par ce type d'IDS. il en est de même pour les attaques provenant de l'intérieur du réseau protégé.

IDS système

Les *IDS système* (HIDS — Host-based IDS) se situent directement sur une ressource à protéger. Leur champ d'action pour détecter les intrusions se limite par conséquent à cette ressource. Toutefois, le type d'attaque concerné n'est pas restreint au trafic réseau comme pour les NIDS, mais concerne également différents éléments présents à même la machine : appels système, processus en cours d'exécution, événements liés au système de fichiers, interactions avec l'utilisateur, etc.

4.2.2 Mécanismes de détection

Deux mécanismes principaux sont mis en œuvre pour détecter les intrusions.

Détection par scénarios

La *détection par scénario* se fonde sur une base de données de *signatures de scénarios d'attaques connus*. Par conséquent, le principe consiste à comparer le comportement à une attaque connue. Ce mécanisme de détection est efficace contre les attaques clairement répertoriées, mais pas contre de nouvelles attaques. Elle génère peu de faux-positifs, c'est-à-dire de fausses alertes, mais nécessite une description très précise des scénarios auxquels se comparer.

Les signatures sont principalement utilisées de deux manières différentes : un langage de haut niveau décrit les scénarios (expressions rationnelles, logique temporelle, etc.), ou on se concentre sur les changements d'états (automates, etc.).

Détection par anomalies de comportement

La *détection par anomalies de comportement* ou *détection comportementale* recherche non pas des attaques spécifiques, mais des *comportements suspects*. Dans cette approche, les opérations effectuées sont comparées à un *comportement de référence* du système. Ceci permet la détection de nouvelles attaques ou d'attaques trop subtiles pour pouvoir être décrites de manière détaillée.

Index

Bibliographie

[And08] R. Anderson : *Security Engineering.* John Wiley & Sons, 2008.

[Gol99] D. Gollmann : *Computer Security.* John Wiley & Sons, 1999.

[HP08] S. Harari et L. Poinsot : *Cryptographie et procédés de chiffrement*, chapitre 7, pages 117–151. *In* Kordon *et al.* [KPP08], 2008.

[KPP08] F. Kordon, L. Pautet et L. Petrucci : *Systèmes répartis en action : de l'embarqué aux systèmes large échelle.* Hermès, 2008.

[Low95] G. Lowe : An attack on the Needham-Schröder public key authentication protocol. *Information Processing Letters*, 56(3):131–136, novembre 1995.

[Mac10] K. Maciunas : *Computer security course.* University of Adelaide, 2010.

[NS78] R. Needham et M. Schröder : Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21(12):993–999, décembre 1978.

[Puj10] P. Pujas : *Cours de cryptologie.* IUT de Béziers, département R&T, 2010.

[Sch96] B. Schneier : *Applied Cryptography.* John Wiley & Sons, 1996.

[Vor09] J.-B. Voron : *Construction automatique et particularisée de systèmes de détection d'intrusion pour les systèmes parallèles à l'aide de réseaux de Petri.* Thèse de doctorat, Université Pierre et Marie Curie, décembre 2009.