Sécurité informatique : peut-on se fier au numérique ? par Hélène Kirchner et Claude Kirchner. Le sens des mots, entretien avec Isabelle de Lamberterie, propos recueillis par Dominique Chouchan.

Hélène Kirchner, Claude Kirchner, Isabelle de Lamberterie

les cahiers de L'INRIA

Rendez-vous avec les sciences et technologies de l'information et de la communication

SÉCURITÉ INFORMATIQUE

Peut-on se fier au numérique?

Dans le monde de l'immatériel, la prévention contre les indésirables passe par des outils logiciels. Leur conception par des méthodes formelles apporte des garanties de plus en plus sérieuses pour anticiper la virtuosité des intrus.

Téléphones cellulaires, cartes de crédit, internet... les dernières décennies ont connu une diffusion spectaculaire des dispositifs capables de communiquer des données parfois très confidentielles. Le cap des 2 milliards d'utilisateurs de téléphones cellulaires dans le monde a par exemple été franchi dès 2005. Or ces derniers gèrent de plus en plus de services. Au Japon, ils servent même de porte-monnaie électronique, de carte de transport... Comment sécuriser les informations que l'utilisateur manipule ou transmet? La réponse à cette question impose de mettre en place des procédures cryptographiques et surtout des politiques de sécurité propres à s'assurer que chacun accède aux bonnes informations et à elles seulement. Un défi scientifique et technique mais aussi un enjeu sociétal majeur[1].

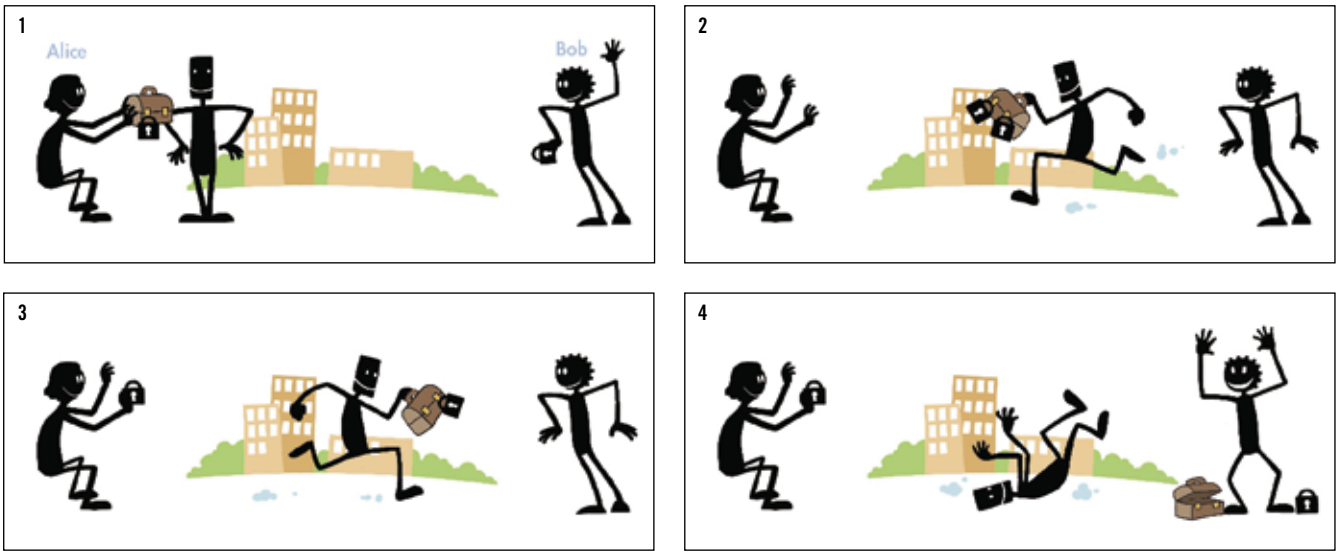
Certains se souviennent peut-être de l'affaire Serge Humpich, qui a éclaté publiquement en 1999. Le jeune Français avait prouvé... par l'exemple! que la carte à puce était falsifiable. Il fut alors poursuivi en justice par le GIE cartes bancaires et condamné. Il avait tout simplement mis en évidence la faiblesse du codage. Le codage cryptographique des cartes à puce est dit à clé publique*. Humpich avait compris qu'il fallait décomposer cette clé en facteurs premiers pour casser le code, ce qu'il fit assez aisément avec un logiciel adéquat car la clé comportait à l'époque un nombre relativement faible de chiffres. Cette histoire aura au moins permis de mettre l'accent sur les points de fragilité de ce type de chiffrage.

Une autre faiblesse éventuelle des systèmes de sécurité, moins connue, est de nature logique. Les échanges d'information nécessitent en effet la mise en œuvre d'un protocole composé d'une succession logique de plusieurs étapes: ouverture de session, authentification, transmission du code... (voir le dessin page suivante). Cet enchaînement d'opérations élémentaires doit n'être entaché d'aucune faille. Dans le cas contraire, même un code cryptographique inattaquable sera impuissant à empêcher les attaques. De nombreuses escroqueries résultent, à la base, d'une usurpation d'identité rendue possible par une telle faille.

Détecter une faille logique est un problème qui ne peut être résolu en toute généralité automatiquement: il est dit «indécidable»*. Des outils sont en revanche mis au point pour analyser les protocoles sous certaines hypothèses (qualité du code cryptographique, qualité du matériel...). Objectifs: repérer les attaques de manière

La machine de chiffrement Enigma a été utilisée par l'Allemagne nazie pendant la Seconde Guerre mondiale. C'est au Britannique Alan Turing, l'un des pères de l'ordinateur (entre autres) que l'on doit d'en avoir déchiffré le code.

© THE WWW.WITPCOM/ENIGMA ENIGMA MUSEUM



Pour être sûr qu'un message secret atteigne son destinataire, le protocole doit être sans faille. Ce n'est pas le cas ici. Alice veut transmettre une mallette (le message) à Bob. Elle place son cadenas (codage) et la confie au messager (le réseau) (1). Bob accepte en y mettant son propre cadenas (2). Voyant cela, Alice enlève le sien (3). Bob peut alors enlever le sien et ouvrir la mallette. Problème : rien ne permet d'identifier les interlocuteurs ! La preuve dans la suite : Alice met son cadenas et confie la mallette au messager (5)… un intrus ! Celui-ci met son propre cadenas (6). Alice croit donc pouvoir enlever le sien (7). L'intrus fait de même et ouvre la mallette et accède ainsi au message secret (8). Pour réparer cette faille, il faut ajouter sur chaque cadenas le nom de celui qui le pose.

systématique ou démontrer leur absence, dans un contexte précis spécifié au départ (réseau non fiable, nombre de sessions lancées simultanément…). Ces outils s'appuient sur des méthodes formelles qui consistent à vérifier systématiquement les propriétés des algorithmes (par opposition aux méthodes empiriques qui consistent à tester sur des cas de figure sans garantie d'exhaustivité).

L'informatique française fait figure de leader en la matière, comme en témoignent, entre autres, deux logiciels développés au cours des cinq dernières années. Le premier, ProVerif, produit conjointement par l'Inria, le Laboratoire d'informatique de l'École normale supérieure et l'Institut Max-Planck pour l'informatique (Sarrebruck, Allemagne), a par exemple servi à certifier un protocole de courrier électronique[2]. Le second, AVISPA (pour *Automated Validation of Internet Security Protocols and Applications*), est le fruit d'un projet européen auquel une équipe du Centre Inria Nancy-Grand Est a fortement contribué[3]. Il a déjà permis d'analyser plus de 80 protocoles connus sur Internet et de détecter des failles sur certains d'entre eux.

Mais une troisième question s'ajoute aux précédentes : comment garantir que, sur une puce comprenant plusieurs applications (retraits bancaires, abonnements de tous ordres…), celles-ci sont mutuellement étanches ? Autrement dit, comment éviter qu'un loueur de voitures, chez qui un utilisateur jouit d'un abonnement, ait accès au compte bancaire de cet utilisateur ? Cette situation se présente avec la JavaCard* : il faut prouver que chaque application est complètement protégée des autres. C'est ce que vient de réaliser en première mondiale, la société française Gemalto (leader de la production de cartes à puce). Elle a certifié ce produit commercial uniquement par des méthodes formelles, en s'appuyant sur le logiciel de preuves CoQ* de l'Inria, et ce au niveau le plus élevé des critères de sécurité. Cette preuve de séparation a ensuite été contrôlée et validée par la Direction centrale de la sécurité des systèmes d'information* (DCSSI).

De plus en plus, les méthodes empiriques sont ainsi supplantées par des approches formelles. Pour ce faire, il a d'abord fallu formaliser les propriétés recherchées, c'est-à-dire exprimer en langage mathématique et informatique des définitions auparavant assez intuitives (intégrité des données, confidentialité, authentification…). Aujourd'hui, comme dans le cas de Gemalto, les recherches s'orientent vers la conception de modèles formels de systèmes (matériel et logiciel) dont la certification est automatisée et vérifiable par l'être humain. C'est ainsi que le projet SSURF (cofinancé par l'ANR), auquel nous participons, s'est fixé pour objectif de mettre au point un environnement d'outils formels permettant à la fois de certifier des logiciels et de faciliter la tâche des organismes de certification en leur apportant toutes les informations nécessaires.

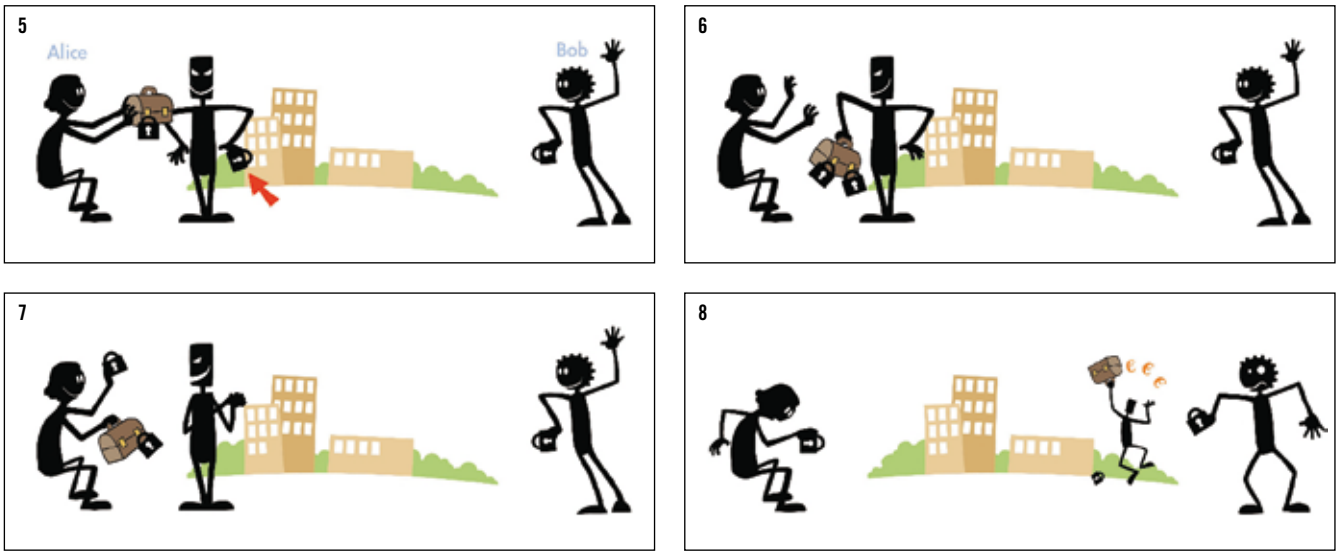
C'est dans ce contexte que nous cherchons à élaborer des politiques de sécurité, elles aussi entièrement fondées sur des méthodes formelles[4]. Pour un système informatique qui traite des flux d'informations d'origines diverses et qui

(1) Claude Kirchner, Chapitre Sécurité informatique, *Encyclopédie des systèmes d'information,* Vuibert, 2006.

(2) Martin Abadi et Bruno Blanchet, Computer-Assisted Verification of a Protocol for Certified Email, *Science of Computer Programming 58*, Elsevier Science, 2005.

(3) Alessandro Armando and al., The AVISPA Tool for the automated validation of internet security protocols and applications, *Lecture Notes in Computer Science vol. 3576*, Springer, 2005.

(4) D. J. Dougherty, C. Kirchner, H. Kirchner, and A. S. de Oliveira, Modular access control via strategic rewriting, *Lecture Notes in Computer Science vol. 4734*, Springer, 2007.



© DESSIN FRANZ KIRCHNER

doit autoriser ou refuser l'accès à telles ou telles données, cela se traduit par la formulation de règles. Ces règles sont souvent nombreuses et à appliquer dans un ordre déterminé. Comment gérer un système aussi complexe et en particulier les interactions entre toutes les règles ? Telle est l'une des questions sur laquelle notre équipe travaille actuellement.

Exemple simple : une entreprise E possède deux filiales, A et B. Supposons que la politique de sécurité s'appuie sur trois règles : l'une permet aux employés de A d'écrire dans les fichiers de E, la seconde aux employés de B de communiquer avec ceux de A, et la troisième interdit à ceux de B d'écrire dans les fichiers de E. Cette politique est-elle cohérente ? Visiblement, non : un employé de B peut communiquer une donnée à un collègue de A qui pourra l'écrire chez E, ce qui enfreint la troisième règle (interdiction à un employé de B d'écrire chez E). Ce type d'incohérence est fréquent et constitue, avec d'autres propriétés, l'objet de nombreuses recherches au plan international. L'un des défis à relever est de combiner de façon sûre des politiques développées indépendamment (soit dans des entreprises en train de fusionner ou dotées de filiales, comme précédemment, soit dans l'écriture d'une politique complexe à partir de « briques » développées séparément).

Pour notre part, nous travaillons sur des techniques dites de réécriture. L'idée est de transformer une arborescence d'opérations (décrivant par exemple l'état d'un système) en une arborescence plus simple. Pour effectuer cette transformation, nous mettons en œuvre des règles de transformation syntaxique (c'est la réécriture), qui transforment ces objets formels en objets plus simples. Un peu comme l'on simplifierait la syntaxe de phrases bourrées de propositions relatives et conjonctives. En matière de sécurité, des règles sont par exemple utilisées pour définir une politique de pare-feu destiné à filtrer les accès dans un réseau. Bien entendu, une bonne politique doit au minimum garantir que les règles ne sont pas contradictoires entre elles.

Il ne fait aucun doute que l'intérêt de tous ces travaux n'échappera pas aux industriels. Les méthodes formelles sont par exemple applicables aux modules de confiance, ou *Trusted Platform Module* (TPM), qui pourraient à moyen terme remplacer les cartes à puce (cartes SIM, de paiement...). Les TPM sont développés sous la houlette de grands industriels tels que Intel, Microsoft, Infineon..., dans le cadre du consortium *Trust Computing Group* (TCG). Au lieu d'être extérieurs à la machine, comme les cartes à puce, les TPM seront intégrés dans les processeurs. Mais en la circonstance, l'enjeu dépasse largement le cadre scientifique et technique, Il est économique et politico-éthique : nul ne peut prédire les systèmes de contrôle voire de cœrcition susceptibles d'être mis en place au sein de tels systèmes. Quant à l'application effective de nos méthodes, elle sera conditionnée par le niveau de protection industrielle de ces technologies. **H. K. et Cl. K.**

Hélène Kirchner est directeur scientifique adjoint à l'INRIA. Elle a dirigé le Centre INRIA-Lorraine et le LORIA de 2001 à 2006 et a assuré la direction scientifique de l'équipe-projet Protheo de 1997 à 2000.

Claude Kirchner, directeur du Centre INRIA Bordeaux-Sud-Ouest, a présidé de 2003 à 2007 les comités scientifiques des actions de recherche sur sécurité et informatique du ministère de la Recherche, puis de l'ANR, et a assuré la responsabilité scientifique de l'équipe-projet Protheo.

* Les codes à clé publique sont fondés sur le caractère public du code de cryptage des messages. Leur décodage nécessite en revanche un code secret que seul possède le destinataire (comme un cadenas à combinaison que chacun peut fermer mais une seule personne ouvrir).

* Un problème est indécidable si aucune procédure mécanique ne permet, en un nombre fini d'étapes, de répondre (oui ou non) à la question posée.

* La JavaCard est une carte à puce permettant d'exécuter des applications en Java, un langage de programmation objet.

* L'assistant à la preuve CoQ permet de concevoir et mettre en œuvre des preuves de programmes ou de théorèmes mathématiques.

* La DCSSI est placée sous l'autorité du secrétaire général de la Défense nationale.

© CNRS-N.TIGET

Entretien avec Isabelle de Lamberterie

Le sens des mots

Traduire une langue dans une autre est mission quasi impossible. C'est pourtant à cette tâche que se sont attelés juristes et informaticiens, à des fins de sécurité informatique.

Isabelle de Lamberterie, juriste et directrice de recherche CNRS, est actuellement directrice adjointe du département Sciences humaines et sociales de l'organisme. Elle a dirigé (1995 à 2001) l'antenne parisienne du Centre d'études sur la coopération juridique internationale (CECOJI) et enseigné dans diverses universités françaises, ainsi qu'à l'université de Montréal.

Juristes et informaticiens entendent-ils la même chose lorsqu'ils parlent de sécurité informatique ?

Isabelle de Lamberterie : Cette question, centrale, a notamment fait l'objet du programme pluridisciplinaire @sphales*, achevé en 2007. Par exemple, que signifie pour un informaticien ou pour un juriste le concept d'intégrité ? Sur quels critères se fonder pour garantir l'intégrité d'un document juridique ? Pour le juriste, c'est le contenu (en langage naturel) qui doit rester immuable. Aux yeux de l'informaticien, il suffit de modifier un bit* du code le représentant en machine pour que son intégrité soit contestée. Or si pour une raison ou pour une autre le document change de support informatique (modernisation de matériel...), sa représentation numérique peut changer sans que le contenu ne soit affecté. Cette double lecture du concept d'intégrité et l'ambiguïté qui en découle ont été relevées dans le cadre d'@sphales. En effet, ce terme a été introduit dans la loi française (2000) lors de la transposition de la Directive européenne sur la signature électronique. À l'époque, nous ignorions encore certaines conséquences de son utilisation : le transfert d'un document était susceptible d'invalider la signature.

Faut-il changer la loi ?

I. de L. : Ce n'est pas d'actualité. Il est important en revanche, dans la lecture juridique de ce texte, d'inciter ceux qui l'appliquent à tenir compte de ce décalage de langage. Au lieu de considérer isolément le concept d'intégrité, nous avons travaillé de concert (juristes et informaticiens) sur la question des « garanties » de l'intégrité. Quelles garanties donner que le contenu d'un document n'est pas atteint dans son intégrité ? Il a été suggéré de se placer sur le terrain des procédures admissibles, autrement dit des procédures certes susceptibles de modifier la représentation numérique du document mais préservant son contenu. Le document doit alors être accompagné de « métadonnées », sorte de traçage de « sa vie » antérieure, avec indication des personnes (éventuellement assermentées) ayant procédé aux transferts de supports. Si les juristes n'avaient pas travaillé avec des informaticiens, ils n'auraient peut-être pas ressenti la nécessité de cette réflexion. Or le droit doit prendre en considération la manière dont sont manipulés les concepts dans le contexte précis où ils sont mis en œuvre.

Vous notez même que si l'on n'y prend pas garde, la loi peut conduire à des absurdités...

I. de L. : En effet. Lors de la discussion de la loi sur la confiance dans l'économie numérique (juin 2004), le législateur a introduit un article qui sanctionne pénalement « le fait de détenir » un instrument, un programme informatique ou des données adaptées pour s'introduire dans les systèmes informatiques. Appliqué à la lettre, ce texte ôtait toute possibilité aux chercheurs de travailler sur des attaques à des fins scientifiques. Après maints débats, il a été convenu qu'aucune sanction ne devait être appliquée en cas de recherches pour un « motif légitime ». L'ANR vient d'ailleurs de lancer un appel à projets sur ce thème (Défi Sécurité Système d'Exploitation Cloisonné et Sécurisé pour l'Internaute). Quant à nous, juristes, le défi est d'être autant que possible neutre technologiquement, de ne pas s'enfermer dans un cadre technique déterminé qui, inévitablement, évolue.

Propos recueillis par Dominique Chouchan

* @sphales, financé dans le cadre de l'Action concertée incitative (ACI) « sécurité et informatique » du ministère délégué à l'Enseignement supérieur et à la Recherche, associaitt 7 équipes de juristes et d'informaticiens.

* Contraction de binary digit : quantité élémentaire d'information pouvant prendre les seules valeurs 0 ou 1.

Réalisation rédactionnelle : Dominique Chouchan - Réalisation graphique : E-Graphics