C&ESAR 2007

Cryptographie: Nouveaux usages, Nouveaux défis

14èmes Journées SSI

6-7-8 novembre 2007

Rennes-Chateaugiron

Avant-propos

Pour cette quatorzième édition, les journées SSI du CELAR (appellation devenue quelque peu réductrice en raison du soutien croissant de multiples bonnes volontés pour l'organisation de cette conférence) vont étrenner leur nouvelle dénomination, C&ESAR (Computer & Electronics Security Applications Rendez-vous), qui concrétise d'une certaine manière un début d'ouverture internationale de cette manifestation ainsi que sa spécificité, diverses journées SSI ayant depuis quelque temps fleuri dans le paysage français de la sécurité des systèmes d'information…

Ces journées SSI, cuvée C&ESAR 2007, seront donc une nouvelle fois le point de rendez-vous de la communauté du domaine, toutes tendances confondues (chercheurs, praticiens, ou décideurs), autour d'un thème technique, cette fois ci la cryptographie. Ce thème avait déjà été abordé lors de la quatrième édition de 1999, mais l'essor de ses nouvelles applications justifiait bien cette nouvelle édition, et son titre: Cryptographie, nouveaux usages, nouveaux défis.

Les points abordés lors de ces trois journées seront:

- Construction (mardi 6): point sur les algorithmes et protocoles, avec un regard sur les aspects évaluation et législation.
- Utilisation (mercredi 7): les applications aux cartes à puce et variantes, les attaques, d'une part, et la problématique du déploiement d'autre part au travers de plusieurs exemples.
- Perspectives (jeudi 8): développements récents en matière de protection des contenus, des logiciels, et des communications.

Il convient enfin de remercier tous ceux qui, avec les organismes qui soutiennent ces journées, ou à titre individuel, apportent leur pierre à cet édifice, pour faire progresser le domaine SSI, et faire en sorte qu'un jour peut-être la sécurité ira naturellement de soi et que les problèmes actuels ne seront plus qu'un lointain et mauvais souvenir (on peut toujours rêver!)…

Yves Correc, pour le comité de pilotage

Organisation

Pilotage:

Pascal Chour (Services du Premier Ministre, SGDN/DCSSI)
Yves Correc (Ministère de la Défense, DGA/CELAR)
Olivier Heen (Thomson)
Ludovic Mé (Supélec)

Logistique:

Guy Appéré (Ministère de la Défense, DGA/CELAR)
Yves Correc (Ministère de la Défense, DGA/CELAR)

Programme:

Hervé Chabanne (Sagem Défense Sécurité)
Florent Chabaud (Services du Premier Ministre, SGDN/DCSSI), président
Yves Correc (Ministère de la Défense, DGA/CELAR), secrétaire
Henri Gilbert (France Télécom R&D)
Louis Granboulan (EADS)
Marc Joye (Thomson)
David Lubicz (Ministère de la Défense, DGA/CELAR)
David Pointcheval (Ecole Normale Supérieure)
Guillaume Poupard (Ministère de la Défense)
Nicolas Sendrier (INRIA)
Frédéric Valette (Ministère de la Défense, DGA/CELAR)

Programme

C&ESAR 2007

Cryptographie: nouveaux usages, nouveaux défis

Cryptography: new stakes, new challenges

Rennes, 6-7-8 novembre 2007

Mardi 6 novembre / *Tuesday, november 6*

Chateaugiron, salle Zéphyr

09:30	***Enregistrement*** *Registration*			*01:00*
	Café */ Coffee*			
10:30				*00:10*
10:40	Thierry Duquesne	DGA / CELAR	Introduction	*00:10*
10:50	Yves Correc	DGA / CELAR	Programme CESAR 2007	*00:10*
11:00	Henri Serres	DGSIC	Ouverture des Journées SSI */ Opening speech*	*00:30*
	Construction	*Problématique cryptographique* *Overview of cryptographic issues*		
			chairman: David Pointcheval	
11:30	Adi Shamir	Weizmann Institute, ENS	New side channel attacks on cryptographic schemes	*01:00*
12:30	***Déjeuner*** */ Lunch*			*01:30*
	Construction	*Algorithmes et protocoles, évaluation, législation* *Algorithms & protocols, evaluation, legal issues*		
			chairman: Louis Granboulan	
14:00	Antoine Joux	DGA - Univ Versailles Saint Quentin	Quelques avancées récentes en Cryptographie	*00:45*
14:45	Reynald Lercier	DGA / CELAR	Primitives cryptographiques asymétriques	*00:45*
15:30	Mathieu Baudet	DCSSI	Les référentiels du schéma de certification et leur usage dans l'évaluation de la cryptographie	*00:45*
16:15	***Pause*** */ Break*			*00:30*
16:45	Eric Jaeger	DCSSI	Méthodes formelles et systèmes cryptographiques	*00:45*
17:30	Marion Videau Stéphanie Lacour	LORIA CNRS	L'exemple du dossier médical personnel (DMP)	*00:45*
18:15				

Mercredi 7 novembre / *Wednesday, november 7*

Chateaugiron, salle Zéphyr

	Utilisation	*Applications, cartes à puces* *Applications, smart cards*		
			chairman: Frédéric Valette	
09:00	Susan Thompson	Mastercard	Security for an international card payment system	*01:00*
10:00	Hubert Pujol	GIE-CB	Les terminaux de paiement et la sécurité	*00:45*
10:45	***Pause*** */ Break*			*00:30*
11:15	Cécile Canovas Jessy Clédière Thanh Ha Le	CEA-LETI CEA-LETI Univ Luxembourg	Attaques par canaux auxiliaires	*01:00*
12:15	Nathalie Feyt Christophe Mourtel	Thalès Gemalto	Cartes sans contact, dual mode et combinatoires: nouvelles problématiques sécuritaires	*00:45*
13:00	***Déjeuner*** */ Lunch*			*01:30*
	Utilisation	*Applications, infrastructures de gestion de clés* *Applications, key management infrastructures*		
			chairman: Florent Chabaud	
14:30	François Morris	CNRS	Déploiement du chiffrement au CNRS	*00:45*
15:15	Eric Frémeaux Jean-Renault Meyer	Astrium CNES	Une application spatiale de la cryptographie: le système d'observation de la Terre PLEIADES	*00:45*
16:00	***Pause*** */ Break*			*00:30*
16:30	Peter Sylvester	Edelweb	Identity and authorization in multi-organisation contexts	*00:45*
17:15	Philippe Painchault	Thalès	Sécurité des systèmes utilisant la cryptographie quantique; le projet européen Secoqc	*00:45*
18:00				
20:45	**Dîner (Rennes) Salons Lecocq-Gadby** */ Reception at Lecoq-Gadby*			

Jeudi 8 novembre / *Thursday, november 8*

Chateaugiron, salle Zéphyr

	Perspectives	*Protection des contenus, des logiciels et des communications* *Content, software and communications security*		
			chairman: David Lubicz	
09:00	Nicolas Prigent et al.	Thomson R&D	AACS, nouveau standard de protection des contenus pré-enregistrés haute-définition	*00:45*
09:45	Sébastien Josse Guillaume Dabosville	Silicomp-AQL	Protection des logiciels contre la rétro-ingénierie	*00:45*
10:30	***Pause*** */ Break*			*00:30*
11:00	Hartmut Seifert	IABG	Security concept for the IT-system of the German Armed Forces	*00:45*
11:45	Francis Dupont	ISC	SEND: la découverte des voisins IPv6 sécurisée	*00:45*
12:30	***Déjeuner*** */ Lunch*			*01:30*
	Perspectives	*Nouvelles problématiques, évolution des standards* *New problems, new standards*		
			chairman: Marc Joye	
14:00	Bart Preneel	Katholieke Universiteit Leuven	Standardization of cryptographic algorithms and the role of EU-funded research on cryptology	*01:00*
15:00	Johann Barbier	DGA / CELAR	La stéganographie moderne: d'Hérodote à nos jours	*00:45*
15:45	Mireille Campana	MINEFI	Clôture des journées SSI */ Closing speech*	*00:30*
16:15	***Cocktail***			*01:00*
17:15				

Mardi 6 novembre 2007

Cryptographie:

Construction

Quelques avancées récentes en Cryptographie

Antoine Joux[1,2]

[1] DGA
[2] Université de Versailles St-Quentin-en-Yvelines
PRISM
45, avenue des Etats-Unis
78035 Versailles Cedex
FRANCE
antoine.joux@m4x.org

Abstract. Modern cryptography is a scientific field of research which evolves at a fast pace. To illustrate this, we study here three recent examples of unexpected and surprising discoveries. The first example is the discovery of pairing based cryptography together with its application to identity based cryptosystems. The second example is of cryptanalytic nature and presents a recently discovered weakness of hash functions based on Merkle-Damgård's paradigm. The third example presents a theoretical discovery on the relationship between the hardness of factoring large integers and the difficulty of inverting RSA encryption.

Résumé. La cryptographie moderne est une discipline scientifique dont l'évolution actuelle est très rapide. A titre d'illustration, nous allons étudier trois exemples récents d'avancées inattendues et étonnantes dans ce domaine. Le premier exemple concerne la découverte de cryptographie à base de couplage et les applications qui en découlent. Le deuxième exemple présente une cryptanalyse sur les fonctions de hachage construites sur le modèle de Merkle-Damgård. Le troisième exemple, plus théorique, illustre le fait que, au moins dans un certain sens, la sécurité du cryptosystème RSA n'est pas équivalente à la difficulté de la factorisation.

1 Cryptographie à base de couplages

Les concepteurs de systèmes de cryptographie à clef publique utilisent une grande variété d'outils mathématiques, généralement issus de l'algèbre dans leurs constructions. En particulier, ces dernières années ont vu le développement de la cryptographie basée sur les courbes elliptiques. Ces systèmes s'appuient en réalité sur un échange de clef de Diffie et Hellman opérant sur une structure de groupe issue de la géométrie algébrique : les courbes elliptiques. En utilisant un

groupe noté multiplicativement, l'échange de clef de Diffie et Hellman est un protocole simple. Les deux participants de l'échange, traditionnellement appelés Alice et Bob, choisissent chacun un nombre en secret, disons a et b et s'échange sur un canal de communication non protégé deux valeurs g^a et g^b appartenant au groupe utilisé. L'idée maîtresse de cet échange de clef est qu'Alice et Bob peuvent tous deux calculer la valeur g^{ab}. Pour Alice, ce nombre s'obtient comme $(g^b)^a$. Pour Bob, c'est $(g^a)^b$. De plus, une tierce personne ne peut pas retrouver g^{ab} à partir des seules valeurs g^a et g^b, sauf si le groupe est particulièrement mal choisi.

Avec des courbes elliptiques, l'idée est la même, mais quelques petits changements de notation s'imposent. Tout d'abord, les éléments du groupe sont des points typiquement appelés P ou Q et non plus g. D'autre part, le groupe est noté additivement et les valeurs rencontrées dans le protocole sont aP, bP et abP. Contrairement aux autres groupes usuellement rencontrés en cryptographie, avec les courbes elliptiques les seules méthodes connues dans le cas général pour calculer abP pour un tiers sont des méthodes génériques, très coûteuses en temps de calcul et de toute façon applicables sur tous les groupes. A l'opposé, pour le Diffie-Hellman usuel, utilisant comme groupe la multiplication dans un corps fini, des méthodes plus efficaces existent, ce qui impose d'utiliser des nombres de très grandes tailles, de l'ordre de 2048 bits environ. Toutefois, depuis de nombreuses années déjà, certaines courbes elliptiques sont connues pour être plus faibles que les autres. Sur ces courbes particulières, il existe des applications permettant de transporter la sécurité du protocole de Diffie-Hellman de la courbe elliptique vers un certain corps fini. Ces applications appelées couplages agissent sur deux points de la courbe et retournent un nombre. La propriété essentielle des couplages est leur bilinéarité, c'est à dire la relation :

$$e(aP, bQ) = e(P, Q)^{ab},$$

pour toute paire de points et tout couple d'entier (a, b).

Sans rentrer dans les détails, notons qu'il existe une grande variété de couplages : couplage de Tate, de Weil, couplage Eta, ... D'un point de vue cryptographique, les couplages les plus utiles sont ceux pour lesquels il existe un point P avec $e(P, P) \neq 1$. Dans cette configuration, on notera $g = e(P, P)$. Historiquement, les couplages ont

d'abord été utilisés pour mettre en évidence la faiblesse relative des courbes qui en sont munies. Ce n'est qu'en 2000 que la première application constructive des couplages est apparue : l'échange de Diffie-Hellman tripartite. Pour illustrer l'importance de la bilinéarité des couplages, examinons rapidement ce protocole tripartite entre Alice, Bob et Charlie. Tout comme dans le Diffie-Hellman classique, chacun choisit un secret, respectivement a, b et c, et publie un point relié à ce secret aP, bP ou cP. Grâce à la bilinéarité, chacun peut déduire des informations publiques et de son secret une clef commune aux trois participants. Cette clef commune $e(P,P)^{abc}$ peut ainsi être calculée par Alice comme :

$$e(bP, cP)^a,$$

utilisant ainsi les objets publiés par Bob et Charlie et le secret propre à Alice.

En 2001, une application phare des couplages a fait son apparition. Cette application, la cryptographie basée sur l'identité, permet dans certains cas de réaliser des échanges à clef publique sans obtenir au préalable la clef publique de son correspondant.

2 Sécurité des fonctions de hachage de type Merkle-Damgård

En cryptographie, les fonctions de hachage sont des primitives essentielles qui doivent permettre de transformer des messages quelconques en une chaîne de bits de petite taille. Le résultat du hachage peut-être considéré comme une sorte d'identifiant unique du message initial. Pour cela, il est essentiel qu'il ne soit pas possible d'exhiber des messages différents ayant des hachés identiques. Lorsque deux tels messages sont trouvés, on parle alors de collisions. Notons que pour de simples raisons de dénombrement, les collisions existent toujours dans les fonctions de hachage. D'autre part, pour un haché de n bits, il existe une méthode générique permettant de trouver des collisions après hachage d'environ $2^{n/2}$ messages. Cette méthode repose sur l'application du paradoxe des anniversaires.

La technique de Merkle-Damgård est une méthode simple permettant de construire une fonction de hachage à partir d'une fonction dite de compression permettant de réduire $n+m$ bits à n bits. Pour

cela, il suffit de partir d'un valeur initiale fixe sur n bits, puis de découper le message en tranches de m bits en application à chaque la fonction de compression à la valeur précédente (au début la valeur initiale) et à la tranche de m bits en cours pour obtenir une nouvelle valeur hachée intermédiaire. La valeur finale est le haché du message. Dans cette construction, il faut prendre quelques précautions afin de pouvoir traiter tous les messages même si leur longueur n'est pas une multiple de m, mais l'idée initiale reste simple.

Une question récurrente au sujet des fonctions de hachage concerne la possibilité de construire une fonction avec $2n$ bits de sortie à partir de deux fonctions sur n bits suffisamment solides. Bien évidemment, on souhaite qu'une telle construction améliore le niveau de sécurité et que la meilleure attaque possible sur la fonction obtenue ait un coût de l'ordre de 2^n appels au hachage. Pour cela, une construction ancienne consiste étant données deux fonctions de hachage G et H à simplement juxtaposer leurs sorties. On construit ainsi la fonction $(G\|H)$. Il est connu depuis longtemps que si G et H sont parfaites, ou plus précisément, que si G et H sont modélisées par des oracles aléatoires, cette construction offre le niveau de sécurité attendu. Une découverte récente, datant de 2004, montre que les fonctions de hachage de type Merkle-Damgård, il n'en est rien. En fait, même si G est parfaite, dès lors que H est de ce type, le niveau de sécurité obtenu par concaténation ne dépasse pas réellement $2^{n/2}$. Cette découverte s'appuie sur la construction de multicollisions pour H. En effet, si l'on est capable de construire $2^{n/2}$ messages différents ayant tous la même valeur par H, l'attaque générique par paradoxe des anniversaires permettra de trouver une paire de message parmi ceux-là collisionnant aussi par G.

L'idée de l'attaque est illustrée par la figure 1. Il suffit de rechercher une première collision sur des messages de 1 bloc, puis à partir du haché intermédiaire de construire une seconde collision en utilisant le deuxième bloc, et ainsi de suite. Au bout de t recherches de collisions, on dispose de t paires de blocs permettant d'obtenir 2^t messages différents de t blocs ayant tous le même haché. En effet, pour chaque bloc de message, deux choix sont offerts et toutes les combinaisons étant possibles, il y a 2^t messages en tout.

Les multicollisions ont également permis d'exhiber des attaques sur d'autres propriétés importantes des fonctions de hachage. A titre

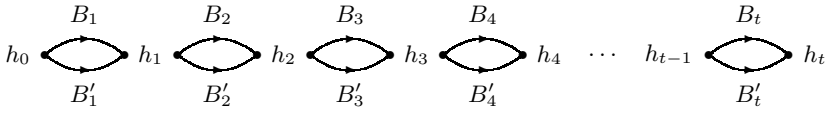


Fig. 1. Représentation schématique de la construction de multicollisions

d'exemple, citons l'attaque de Nostradamus présentée à Eurocrypt 2006 par Kelsey et Kohno.

3 Relations entre RSA et la factorisation d'entiers

Depuis l'invention du système RSA par Rivest, Shamir et Adleman en 1977, une question récurrente se pose sur ce cryptosystème : Quel est le lien exact entre la sécurité de RSA et la difficulté de factoriser des grands nombres ?

Tout d'abord, rappelons brièvement le principe du système RSA. La clef publique de ce système est formé de deux grands nombres, le premier est noté N et appelé module RSA, le second est noté e et appelé exposant public. La clef privée correspondante est formée de la clef publique et d'un nombre supplémentaire, l'exposant secret, noté d. Lors de la construction de paramètres RSA, N est obtenu comme produit de deux nombres premiers p et q et e et d sont choisis pour satisfaire la relation :

$$ed = 1 + \lambda(p-1)(q-1),$$

où λ est un nombre entier. Avec ces choix, on construit aisément deux fonctions E et D inverses l'une de l'autre en posant :

$$\begin{aligned} E(x) &= x^e \pmod N \quad \text{et} \\ D(x) &= x^d \pmod N. \end{aligned}$$

Ces deux fonctions opèrent sur les entiers modulo N, ou en fixant les représentants sur les entiers de l'intervalle $[0, N-1]$.

Mal utilisé, le système RSA souffre de quelques problèmes de sécurité, par exemple, le chiffré de 0 est toujours 0, celui de 1 toujours

1. De plus, si l'on chiffre un entier x d'un petit intervalle en calculant simplement $E(x)$, il est possible en essayant tous les éléments de l'intervalle de retrouver la bonne valeur. Un autre propriété de RSA pose parfois des problèmes de sécurité, c'est la multiplicativité, le chiffré par E d'un produit de nombres est toujours le produit de leurs chiffrés. De même, le déchiffrement par D d'un produit est le produit des déchiffrés. Cela peut conduire à ce que l'on appelle des attaques multiplicatives sur les signatures RSA. Pour remédier à ces éventuels problèmes, la pratique actuelle impose de toujours pré-traiter les messages à chiffrer ou signer par RSA par une transformation assez complexe pour les faire disparaître. En chiffrement, l'exemple typique est l'utilisation d'OAEP (Optimal Asymmetric Encryption Padding) de Bellare et Rogaway.

D'un autre côté, les propriétés multiplicatives de RSA sont parfois très utiles et elles permettent, en particulier, la réalisation de protocoles de signature en aveugle. Supposons que l'on souhaite obtenir la valeur $D(x)$, correspondant à une signature, sans pour autant révéler x à la personne réalisant le calcul. Il suffit de demander la valeur $D(z)$ avec $z = x \cdot y^e$, puis de retrouver $D(x)$ en divisant par y. Bien évidemment, ceci ne doit être réalisé que dans le cadre d'un protocole strict afin de ne pas signer n'importe quoi, mais la fonctionnalité offre de nombreuses applications.

Concernant les liens entre RSA et la factorisation, on sait que la connaissance de e et d permet de retrouver la factorisation de N. De plus, il existe une variante de RSA pour laquelle on choisit $e = 2$, alors que cette valeur n'est pas possible pour le RSA standard, dont la sécurité est équivalente à la difficulté de factoriser. Toutefois, pour le RSA standard, rien ne prouve qu'il ne soit pas possible de calculer des racines e-ièmes et donc d'appliquer D, sans connaître D. D'autant plus, que pour un cryptanalyste, il est généralement possible d'effectuer par ailleurs un certain nombre de requêtes à D. En fait, un résultat très récent, à paraître dans Asiacrypt'2007 vient montrer qu'au moins dans un certain contexte, il est plus facile d'inverser RSA que de factoriser.

Plus précisément, si un cryptanalyste à un accès direct à une boîte calculant D, il est capable en posant des questions bien choisies de construire une base de données qui lui permettra par la suite de calculer la racine e-ième $D(x)$ de n'importe quel nombre modulo

N. Cette méthode de cryptanalyse est complexe et coûteuse, mais en terme de calcul bien plus efficace que les meilleurs algorithmes de factorisation connus pour décomposer N. Pour donner un exemple, la factorisation d'un nombre de 512 bits a été effectuée en 1999 et a nécessité plus de trois mois de calcul sur une centaine de machines. Avec le même nombre, l'attaque que nous venons d'évoquer peut construire sa base de données en moins de deux jours sur un seul processeur puis calculer les racines e-ième au rythme d'une par heure. Pour la construction de la base de données, il est nécessaire d'obtenir 400 millions d'accès préalables à la fonction D. L'analyse de la complexité de calcul de cette attaque montre qu'asymptotiquement, attaquer par cette approche le système RSA coûte essentiellement le même temps que factoriser un nombre ayant deux fois moins de chiffres que le module RSA utilisé. Vu autrement, pour préserver le niveau de sécurité face à cette nouvelle attaque, il faut approximativement doubler la taille des clefs utilisées.

Primitives cryptographiques asymétriques

Asymmetric cryptographic primitives

Reynald Lercier
DGA / DET / CELAR
La Roche Marguerite
35170 Bruz
reynald.lercier@dga.defense.gouv.fr

Résumé

Les progrès accomplis depuis les années 90 en matière de sécurité concrète des algorithmes et protocoles, en particulier face à des attaquants adaptatifs, ont fourni un cadre formel solide à l'étude des schémas cryptographiques, nouveaux ou anciens. Passer à l'étape pratique reste cependant aussi délicat que par le passé. Une première difficulté est par exemple de rester dans le cadre strict de l'étude, une seconde est d'instancier avec des primitives concrètes les objets qui ont été modélisés comme idéaux dans la preuve de sécurité. Dans ce dernier cas, si l'on se restreint au domaine de la cryptographie asymétrique, il nous faut malheureusement constater que parmi les nombreux problèmes algorithmiques difficiles que l'on a vu fleurir depuis l'invention du RSA, peu aujourd'hui sont encore considérés comme viables.

Nous nous proposons d'illustrer cette situation, à partir de [1], avec le cas de la cryptographie Diffie-Hellman. Nous disposons sur le sujet de résultats de Shoup, d'abord en 1997 sur la difficulté prouvée des problèmes Diffie-Hellman considérés en toute généricité, ensuite en 1998 sur l'existence de schémas de chiffrement asymétriques efficaces. Il reste néanmoins à choisir un groupe précis et s'assurer, tant que possible, jusqu'à quel point les hypothèses théoriques y sont réalisées. Si, dans les cas nominaux (cas des corps finis premiers ou des courbes elliptiques), les quelques expériences menées grandeur nature sont rassurantes, nous mettons en évidence que des "accidents" inquiétants se produisent dès lors que l'on sort un tant soit peu des chemins balisés.

En conclusion, on dispose aujourd'hui de relativement peu de primitives cryptographiques asymétriques réputées solides. En développer de nouvelles, même si c'est une tâche ardue, serait appréciable.

[1] A. Joux, R. Lercier, "Algorithmes pour résoudre le problème du logarithme discret dans les corps finis", proceedings de la Journées Annuelles 2007 de la Société Mathématiques de France, consacrées aux Nouvelles Méthodes Mathématiques en Cryptographie"

Abstract

Progresses that have been accomplished since the 90's in terms of concrete security of algorithms and protocols, especially in front of adaptive attackers, yield a strong formal basis to study old or new cryptographic schemes. But, surprisingly, implementing them in practice remains as difficult as it used to be. A first difficulty is for instance to stay in the framework imposed by the study, a second one is to replace with concrete primitives ideal objects that are introduced in the security proofs. In the case of asymmetric cryptography, we must unfortunately notice that, among the numerous propositions that have been studied since the discovery of RSA, only few are still considered as viable.

We illustrate this situation, following [1], in the case of the Diffie-Hellman cryptography, where it is not difficult to catch the main ideas behind usual schemes. We furthermore have on the subject results from Shoup, first in 1997 on the difficulty of Diffie-Hellman problems in the generic setting, ans then in 1998 on the existence of efficient cryptographic cipher algorithms. But we have to choose a precise group and be sure, as much as possible, that theoretical hypothesis that have been done in the proofs, are still verified. If in usual cases (prime finite fields, elliptic curves), experiments are reassuring, we show that unexpected accidents can occur outside of this way.

In conclusion, we have today only few strong asymetric cryptographic primitives. To develop new ones, even if it is a very difficult task, would be nice.

[1] A. Joux, R. Lercier, "Algorithmes pour résoudre le problème du logarithme discret dans les corps finis", proceedings de la Journées Annuelles 2007 de la Société Mathématiques de France, consacrées aux Nouvelles Méthodes Mathématiques en Cryptographie"

Les référentiels du schéma de certification et leur usage dans l'évaluation de la cryptographie

The technical standards of the French certification scheme and their usage for cryptographic evaluations

Mathieu Baudet
SGDN / DCSSI
51, boulevard de La Tour Maubourg, 75007, Paris
mathieu.baudet@sgdn.pm.gouv.fr

Résumé

Le schéma de certification français prévoit que l'évaluation des produits de sécurité est supervisée in fine par la DCSSI. Pour la partie purement cryptographique, à l'heure actuelle, les évaluations sont conduites par la DCSSI elle-même, suivant un ensemble de règles et de principes regroupés dans plusieurs documents de référence. L'objet de cette présentation est de décrire les grandes lignes de ces documents et d'en illustrer le fonctionnement sur plusieurs exemples concrets.

Abstract

According to the French certification scheme, the evaluation of security products is supervised in fine by the DCSSI. As far as cryptography is concerned, at the present time, evaluations are conducted by DCSSI by itself, following a set of rules and principles described in several reference documents. The purpose of this talk is to outline these documents and illustrate their working on several concrete examples.

Méthodes formelles et cryptographie

Formal methods and cryptography

Eric Jaeger
SGDN / DCSSI
51, boulevard de La Tour Maubourg, 75007, Paris
eric.jaeger@sgdn.pm.gouv.fr

Résumé

L'intérêt des méthodes formelles pour augmenter le niveau d'assurance des produits de sécurité est largement reconnu et valorisé, notamment dans le cadre des Critères Communs. En pratique, toutefois, peu de produits disponibles bénéficient de l'apport des méthodes formelles. Dans cet exposé, nous tenterons de dresser un panorama des usages actuels et des perspectives d'application des méthodes formelles aux produits cryptographiques.
Nous rappellerons dans un premier temps que la terminologie de méthodes formelles peut s'appliquer aussi aux preuves de sécurité, avant d'introduire et de nous concentrer sur les méthodes formelles informatiques pour identifier les avantages qu'elles peuvent tout particulièrement apporter dans l'intégration de la cryptographie dans un système.

Abstract

The interest of formal methods for improving the assurance levels of security products is widely recognized and valued, notably in the framework of Common Criteria. In practice, though, few available products benefit from the contributions of formal methods. In this talk, we will attempt to outline the current usages and sketch some possible new applications of formal methods to cryptographic products. We will first recall that formal methods can be understood as formal proof of cryptographic security. Then, we will introduce and focus on formal methods in computer science in order to outline the advantages that they can naturally bring in the integration process of cryptography in a system.

Cryptographie: aspects légaux

Marion Videau
LORIA
615, rue du jardin botanique
BP101, 54602 Villers-lès-Nancy Cedex
marion.videau@loria.fr

Mercredi 7 novembre 2007

Cryptographie:

Utilisation

Les terminaux de paiement et la sécurité

Points of Sale and Security

Hubert Pujol
GIE Carte Bancaire
Washington Plazza, 75008 Paris
hubert-pujol@cartes-bancaires.com

Résumé

Les terminaux de paiement et plus généralement les systèmes d'acceptation sont des éléments sensibles d'un système de paiement et constituent donc des cibles potentielles pour les fraudeurs.
Les experts « sécurité » monétique procèdent donc régulièrement à une analyse de risque sur les systèmes d'acceptation qui vise à définir les exigences de sécurité de nature à contrer les menaces identifiées. Pour satisfaire ces exigences, différents mécanismes cryptographiques sont mis en place : authentification des supports et des utilisateurs, protection du code confidentiel, transactions stockées ou échangées avec les serveurs bancaires.
La certification des terminaux doit apporter la garantie que les matériels utilisés sont bien conforme aux exigences de sécurité définies.
L'exposé montrera comment le système « CB » procède dans le contexte actuel mais aussi comment les acteurs européens travaillent pour proposer une solution homogène pour les moyens de paiement en Europe.

Abstract

The Point of Sale and more generally accepting devices, are sensitive elements of a payment scheme. So they are seen as potential targets by fraudsters.
Banking security experts provide a risk assessment on a regular basis for accepting devices to define security requirements against identified threats. TO satisfy these security requirements, many cryptographic functions are in place: cards and user authentication, Pin protection, logged or transmitted transaction data for banking hosts.
Certification of Points of Sale must bring the guarantee that these devices are compliant with the initial security requirements.
The talk will show how « CB » manage its own system but also how European stakeholders are working together to provide a homogeneous solution for payment transaction in Europe.

Side Channel Attacks

Thanh Ha Le *
Université du Luxembourg
thanhha.le@uni.lu

Cécile Canovas
CEA-LETI Minatec
cecile.canovas@cea.fr

Jessy Clédière
CEA-LETI Minatec
jessy.clediere@cea.fr

Résumé

Depuis l'avènement des microcontrôleurs sécurisés dans le domaine des cartes bancaires françaises, de plus en plus d'applications sont implémentées sur ces plateformes souples d'utilisation, et par nature offrant un niveau de sécurité devant être élevé. Parallèlement, un engouement et une panoplie considérable d'attaques sur ces plateformes sont nés suite à la popularité des cartes à puce électroniques. Ces attaques peuvent être élaborées soit par des pirates, notamment pour la télévision à péage, soit par des laboratoires universitaires ou de tests de composants. Dans les deux cas, de part le dynamisme créé par ces attaques, la sécurité des applications implémentées sur ces plateformes électroniques est en constante progression. Ces attaques sont focalisées directement sur la nature électronique du support sur lequel est implémentée l'application. Du point de vue de la cryptographie, outre les faiblesses intrinsèques d'un algorithme choisi, son implémentation sur une plateforme donnée apporte des vulnérabilités supplémentaires.

L'objet de cette présentation est de présenter les différentes attaques cryptographiques qui exploitent les vulnérabilités connues des puces, en particulier les attaques par canaux auxilliares, ainsi que les contremesures classiques qui peuvent être implémentées.

Abstract

Since the introduction of secured microcontrollers in the French banking system, these electronic devices have become very popular in many applications, especially those requiring security features. At the same time, a strong community of "hackers", like individuals tampering with pay TV access, public or private integrated-circuit testing institutions, has progressively grown. With this large panel of non conventional testing, the security of IT is continuously improved. Attacks performed in those testings are essentially focalized on the electronic medium of the application. From a cryptographic point of view, besides the intrinsic vulnerabilities of a chosen algorithm, its implementation on a given platform adds subsequent weaknesses.

The presentation shows the different attacks that exploit some well-known vulnerability of the chips, in particular side channel attacks and classical implemented countermeasures.

Mots clés

Carte à puce, attaques, signaux compromettants, DPA, DEMA, CPA

*Travaux réalisés pendant sa thèse au CEA-LETI Minatec

1 Contexte

Aujourd'hui, à l'intérieur du portefeuille de chacun d'entre nous, il y a certainement au moins une carte bancaire, une carte vitale et peut-être d'autres cartes à puce. D'une manière transparente, les cartes à puce deviennent une partie non négligeable de la vie quotidienne des français. Elles permettent d'authentifier des utilisateurs, de stocker des données personnelles ou de réaliser des transactions d'une façon rapide et sécurisée.

Au-delà des performances du processeur et des applications que les cartes à puce peuvent embarquer, celles-ci sont réputées pour leur sécurité. Un des avantages principaux des cartes à puce en comparaison des autres types de carte, tels que les cartes à pistes magnétiques, est qu'elles peuvent stocker une grande quantité de données confidentielles et les transférer d'une façon protégée. De ce fait, elles sont la cible de plusieurs attaques. En principe, les attaques sur les cartes à puce peuvent être divisées en trois types : les attaques au niveau social, au niveau physique et au niveau logiciel. En pratique, ces types d'attaques peuvent être combinés pour effectuer une attaque à plusieurs niveaux. Par exemple, une attaque au niveau physique pourrait être considérée comme une préparation à une autre attaque au niveau logique, ce qui est par exemple le cas de l'attaque Differential Fault Analysis [6, 4].

2 La sécurité des cartes à puce

Les attaques au *niveau social* sont principalement reliées aux personnes qui travaillent avec la carte à puce. Ceux-ci peuvent être des concepteurs de puces travaillant pour des fabricants de semi-conducteurs, des concepteurs de logiciel ou, plus tard dans le cycle de vie de la carte, des utilisateurs de cartes. Un exemple typique de ce type d'attaque est l'acquisition du code PIN quand ce dernier est tapé sur le clavier. Les attaques au niveau social contre les programmeurs de cartes peuvent être restreintes si la sécurité ne dépend que des clés secrètes mais pas de la connaissance du code des développeurs.

Les attaques au *niveau physique* des cartes à puce demandent souvent des équipements techniques performants pour pouvoir accéder au matériel du microcontrôleur. Les attaques peuvent être statiques, c'est à dire qu'aucune alimentation n'est appliquée au microcontrôleur, ou dynamiques pendant le fonctionnement du microcontrôleur. Les attaques physiques statiques n'imposent aucune restriction au niveau du temps à l'attaquant, qui peut faire son travail à son propre rythme. Néanmoins, avec une attaque dynamique, les équipements de mesure doivent être disponibles et l'acquisition de données doit être suffisamment rapide.

Jusqu'à aujourd'hui, les attaques les plus réussies sur les cartes à puce ont été réalisées au *niveau logique*. Ces attaques ont surgi à partir de pures réflexions logiques. Cette catégorie inclut la cryptanalyse classique, ainsi que les attaques exploitant des défauts des systèmes d'exploitation ou des chevaux de Troie dans le code exécutable des applications. Les attaques au niveau logique peuvent être partagées en deux types : les attaques passives, dans lesquelles l'attaquant analyse le texte chiffré ou le protocole cryptographique sans les modifier, et les attaques actives dans lesquelles l'attaquant modifie les données au sein d'un protocole.

Dans la suite de cet article, nous nous focaliserons sur les attaques par observation de

signaux compromettants[1], un type d'attaque au niveau physique.

2.1 Les attaques

Si les attaques invasives demandent un accès au circuit et détruisent le paquetage de la puce, les ***attaques non invasives*** sont nettement plus transparentes. Elles sont particulièrement dangereuses dans quelques applications pour les raisons suivantes. Tout d'abord, le propriétaire de la carte peut ne pas savoir que la clé secrète a été volée et donc il l'utilise encore pendant un certain de temps. En plus, les attaques non invasives sont plus faciles à mettre à jour car les équipements nécessaires peuvent facilement être reproduits.

Cependant, les attaques non invasives demandent une bonne connaissance au niveau physique et logiciel de la puce alors que les attaques invasives peuvent être appliquées sur plusieurs types de produit sans connaître leurs technologies. Ainsi, les attaques commencent souvent avec des analyses invasives suivies par des attaques non invasives.

La première famille des attaques non invasives au niveau physique est constituée des attaques par injection de fautes. En 1996, Dan Boneh, Richard Demillo et Richard Lipton ont publié une étude [6] sur le modèle théorique pour déterminer les clés secrètes des algorithmes cryptographiques asymétriques en injectant des fautes au niveau matériel de la puce. Dans la même période, Eli Biham et Adi Shamir ont annoncé une extension appelée *Diffential Fault Analysis* (DFA) [4] pour les algorithmes cryptographiques symétriques comme le DES. Ce type d'attaque a été ensuite analysé par plusieurs auteurs [29, 36, 34]. Le principe de base de ces attaques est relativement simple. Dans le premier temps, un texte clair aléatoire est chiffré en utilisant la clé que l'on veut trouver et le texte chiffré est enregistré. Ensuite, la puce est perturbée en injectant des fautes pendant qu'elle effectue l'algorithme cryptographique. Cela rend un texte chiffré incorrect. Ce processus est répété plusieurs fois et tous les résultats sont sauvegardés et analysés. À partir des faux textes chiffrés, on peut appliquer des méthodes de cryptanalyse pour déduire la clé secrète. La dernière partie de l'attaque est donc purement cryptographique. Ainsi, l'attaque DFA est en même temps une attaque au niveau physique et au niveau logique.

La deuxième famille des attaques non invasives regroupe les attaques par observation de signaux compromettants. Le premier type de signaux analysé est la durée des opérations effectuées par la carte [22, 14, 19, 33]. Le fait que la puissance de consommation de la carte dépende des instructions exécutées et des données manipulées permet de déduire une certaine quantité d'informations secrètes. En 1999, Paul Kocher, Joshua Jaffe et Benjamin Jun ont publié un papier connu sur la *Simple Power Analysis* (SPA) et la *Differential Power Analysis* (DPA) [23]. L'analyse de la consommation a été ensuite validée sur plusieurs types de composants protégés par différents algorithmes cryptographiques [5, 10, 26, 11, 18, 24, 21, 25, 8, 3, 9, 30]. La méthode *High-order Differential Power Analysis* [28, 35, 20] est considérée comme une extension de la DPA pour attaquer les puces employant une simple contre-mesure anti-DPA. Ces trois types d'analyse de la consommation représentent aujourd'hui une attaque dangereuse aux niveaux matériel et logiciel. En plus, le coût et la complexité des équipements nécessaires pour réussir une attaque utilisant cette méthode sont relativement limités.

[1]Side Channel Attacks en anglais, ou autres noms en français : attaques par canaux auxiliaires, attaques par canaux cachés

Après la publication de Kocher et al., quelques auteurs ont publié simultanément en 2001 un autre type d'attaque en mesurant le rayonnement électromagnétique de la puce par différents types de capteurs [16, 31, 32]. Cette attaque est basée sur le même principe de l'analyse de la consommation car il existe aussi une relation entre les signaux électromagnétiques émis par la puce et les instructions et/ou les données manipulées, d'où viennent les termes *Simple ElectroMagnetic Analysis* (SEMA) et *Differential ElectroMagnetic Analysis* (DEMA).

2.2 Les protections

Concernant les contre-mesures des attaques par observation de signaux compromettants, les solutions proposées détruisent la dépendance entre la consommation électrique et les instructions ainsi que les données manipulées. On peut ajouter un bruit sur les signaux de consommation ou les signaux électromagnétiques pour diminuer l'efficacité de détection de la clé secrète. Une autre idée consiste à générer les signaux d'horloge d'une manière aléatoire pour que les signaux mesurés ne soient pas correctement synchronisés.

Les chiffrements des bus [15] du processeur sont également utilisés pour éviter les attaques à texte connu. La technique dual-rail et la logique asynchrone sont aussi des solutions contre les attaques par observation.

3 Attaques par observation de signaux compromettants

La section précédente donne une vue globale sur les attaques, les protections et les contre-mesures au niveau physique. Dans cette section, nous nous focalisons sur les attaques par observation de signaux compromettants. Nous rappelons que ces attaques sont au niveau physique, non invasives et effectuées quand les cartes sont en cours d'utilisation.

Une attaque par observation de signaux compromettants est une attaque basée sur les informations obtenues à partir de l'implémentation et de l'exécution physique d'un crypto système, plutôt que sur les faiblesses théoriques des algorithmes de chiffrement, par exemple la cryptanalyse. Le temps de calcul, la puissance de consommation et le rayonnement électromagnétique d'un crypto système, par exemple la carte à puce, sont des sources d'informations compromettantes qui peuvent être exploitées pour casser le système. Ce type d'attaque demande des appareils qui ne sont pas trop coûteux et le temps d'attaque est relativement faible par rapport aux attaques invasives de type microsondage.

L'attaque par temps d'exécution, ou *Timing Attack* en anglais, est basée sur le temps d'exécution des opérations d'un système. La première idée d'utiliser le temps d'exécution a été présentée par Paul Kocher en 1996 [22] pour attaquer l'implémentation de Diffie-Hellman, RSA et DES. Cette attaque a été ensuite mise en pratique par une analyse détaillée sur le temps de calcul de l'algorithme d'exponentiation modulaire du RSA [14] et développée en [19, 33]. Quelques contre-mesures de ce type d'attaque ont été proposées en [22, 14]. Aujourd'hui, en rendant le temps d'exécution des opérations constant et indépendant de la clé k, la plupart des cartes à puce sont protégées contre cette attaque.

3.1 Analyse de la consommation

La plupart des dispositifs cryptographiques modernes sont implémentés à l'aide de portes logiques de type semi-conducteur, qui sont construites à partir des transistors. Quand la charge est appliquée (ou enlevée) à une porte, les électrons s'écoulent à travers le substrat de silicone et par conséquence, une quantité d'électricité est consommée.

Pour mesurer la consommation électrique d'un circuit, on insère une petite résistance R, d'environ 10 Ohm, en série entre le connecteur de masse du composant et la masse de l'appareil de mesure. Le courant I circulant à travers R crée une tension variable V_{SS} en temps qui est proportionnelle à I et peut être échantillonnée avec l'aide d'un oscilloscope.

L'analyse simple de consommation **SPA** (Simple Power Analysis) a été initialement proposée par Kocher et al [23] et développée en [24, 25]. C'est une technique qui interprète directement des mesures de la consommation électrique d'un circuit pendant ses opérations cryptographiques. La SPA permet tout d'abord de détecter une exécution d'une opération. Un exemple typique de ce type de détection est l'observation des courbes de consommation d'une carte à puce exécutant une opération DES. La figure 1 représente une des courbes de consommation avec les 16 périodes correspondant aux 16 boucles de DES. En observant soigneusement les courbes de consommation, l'attaquant peut voir des détails concernant l'opération DES.

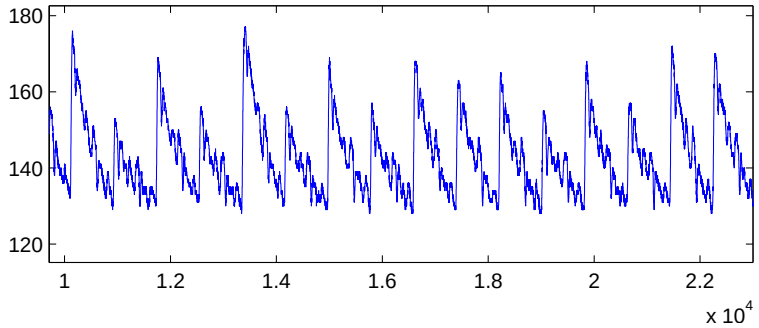


FIG. 1 – Un signal de consommation d'un ASIC

Comme la SPA peut indiquer l'ordre des instructions exécutées, elle peut être employée pour retrouver les opérandes d'une opération dans laquelle le chemin d'exécution dépend des données traitées. Quelques exemples de ce type d'opération sont la génération de la clé de DES (DES key schedule), les permutations de DES, les comparaisons, la multiplication et l'exponentiation [23]. Ainsi, pour pouvoir interpréter les mesures, l'attaquant doit avoir une connaissance sur l'implémentation des algorithmes du composant attaqué.

Les techniques pour empêcher la SPA sont généralement assez simples à mettre en place. Il faut tout d'abord éviter d'utiliser les secrets intermédiaires ou la clé dans les opérations conditionnelles. Une autre méthode utilisée dans plusieurs implémentations est de rendre les variations de consommation suffisamment faibles, ou indépendantes de données. L'ajout de paramètres aléatoires pour empêcher l'attaquant de réduire le bruit en faisant la moyenne de quelques signaux est également une bonne idée.

Comme indiqué avant, à côté des variations de consommation à grande échelle liées à la séquence des instructions, il y a des effets reliés aux données manipulées. Ces variations sont en général très faibles par rapport au niveau du bruit ajouté pendant les mesures.

Il faut donc employer une méthode statistique pour pouvoir observer ces variations. La technique **DPA** (Differential Power Analysis) [23] est une attaque exploitant la dépendance entre la consommation électrique et les données manipulées en utilisant un grand nombre de signaux de consommation. La DPA originale de Kocher [23] est basée sur la propriété intéressante suivante : pendant le chiffrement d'un texte, la consommation du composant quand un bit b du texte est mis à 1 sera différente de celle quand b est mis à 0. Si le nombre de messages est suffisamment grand, en utilisant la DPA, on peut distinguer la bonne clé parmi plusieurs hypothèses. Le plus grand avantage de la DPA par rapport à la SPA est qu'elle ne demande pas de connaître en détails l'implémentation du code.

Pour empêcher la DPA, on diminue la taille des variations et ajoute des bruits sur les signaux de consommation [23]. Goubin et al. [17] ont proposé une autre stratégie, appelée la méthode de "duplication", pour protéger l'algorithme DES en divisant la donnée secrète en deux parties aléatoires et opérant chaque partie séparément. Des techniques similaires peuvent être appliquées pour protéger l'algorithme AES (Advanced Encryption Standard) contre l'analyse de consommation [10]. Ces techniques sont considérées comme la première utilisation de masques, une contre-mesure efficace pour la DPA au premier ordre, qui est ensuite développée en [27, 12, 1].

Une extension de la DPA, la méthode **DPA d'ordre supérieur**[2] [23, 28] permet d'attaquer des composants qui emploient des contre-mesures de la DPA du premier ordre. Pendant cette dernière attaque, l'attaquant mesure les signaux de consommation et calcule des propriétés statistiques individuelles du signal à chaque instant. Dans une attaque DPA d'ordre supérieur, l'attaquant calcule des propriétés statistiques conjointes de la consommation à plusieurs instants du signal. Une attaque DPA de n-ième ordre est définie comme l'analyse qui utilise n échantillons différents du signal de consommation correspondant aux n valeurs intermédiaires différentes pendant l'exécution d'un algorithme. Des études approfondies sur la DPA du deuxième ordre, ainsi que des contre-mesures peuvent être trouvées dans [28, 35, 20, 2]

Depuis quelques années, l'analyse **CPA** (Correlation Power Analysis) [10, 13, 24, 7] est largement étudiée et considérée comme une attaque sur la consommation très efficace. Cette méthode exploite la corrélation entre la consommation électrique et le modèle de consommation du composant (par exemple le modèle de poids de Hamming, le modèle de distance de Hamming). Pour détecter la clé secrète, on calcule le facteur de corrélation entre la consommation réelle et le modèle théorique. On choisit ensuite l'hypothèse correspondant au plus grand facteur de corrélation pour révéler la clé secrète.

3.2 Analyse du rayonnement électromagnétique

Une variation brusque du courant dans un circuit CMOS cause une impulsion sur le champ électromagnétique émis par le circuit, qui peut être capturé par des capteurs inductifs. La relation entre le champ magnétique et le courant est donnée par la loi de Biot-Savart :

$$\overrightarrow{dB} = \frac{\mu I \overrightarrow{dl} \wedge \overrightarrow{r}}{4\pi r^3}$$

[2] *Higher Order DPA* en anglais

où $\overrightarrow{dl}$ est la longueur infinitésimale du conducteur portant le courant électrique I, μ est la perméabilité magnétique et $\overrightarrow{r}$ est le vecteur directionnel représentant la distance vectorielle entre le courant et un point dans le champ.

Par ailleurs, selon la loi de Faraday, n'importe quel changement dans l'environnement magnétique d'une bobine causera une tension :

$$V = \frac{d\phi}{dt}$$

où le flux magnétique ϕ est $\int_S \overrightarrow{B} \wedge \overrightarrow{dS}$. Par conséquent, si des informations utiles pour l'analyse de consommation sont contenues dans le courant I, elles peuvent être aussi détectées en mesurant la tension V aux bornes d'une bobine.

Les attaques utilisant des signaux électromagnétiques mesurés par différents types de capteurs, au lieu des signaux de consommation, ont été proposées simultanément par quelques auteurs [16, 31, 32]. Aux analyses SPA, DPA, correspondent les méthodes **SEMA** (*Simple ElectroMagnetique Analysis*) et **DEMA** (*Differential ElectroMagnetique Analysis*).

Le premier avantage des signaux électromagnétiques par rapport à des signaux de consommation est la possibilité de les acquérir sans accès direct au composant. En outre, pour chaque chiffrement d'un message, plusieurs signaux électromagnétiques peuvent être mesurés en plaçant des capteurs à différentes positions [31]. Par conséquent les signaux électromagnétiques permettent d'obtenir une information plus localisée. Notons également que les signaux électromagnétiques sont parfois plus bruités que ceux de consommation. Alors, des techniques de réduction de bruit sont nécessaires pour améliorer la qualité de signal avant d'effectuer des analyses.

4 Optimisations et améliorations de l'attaque

Les attaques par observation de signaux électromagnétiques ont déjà été testées sur plusieurs types de composant, par exemple les ASIC ou les FPGA, et sur différents types d'algorithmes cryptographiques (DES, AES, RC4, ECC, RSA). Les résultats montrent que ce type d'attaque permet de trouver la clé secrète dans de nombreux cas. Cependant, le succès de ces attaques est restreint par le niveau de bruit, le nombre de messages ou le modèle de la consommation du circuit considéré.

Pour attaquer efficacement une carte à puce en observant sa consommation ou son rayonnement électromagnétique, il est nécessaire de modéliser de façon réaliste le modèle de consommation de la carte. Aussi, afin d'améliorer l'efficacité de ces attaques, différents modèles doivent être étudiés afin de concevoir des méthodes plus puissantes. Un modèle complet, par exemple celui des 64 bits de calcul du DES, ne se rélève pas pertinent. Par conséquent, il est très souvent restreint à un sous ensemble du fonctionnement, typiquement la sortie d'une ou deux boîtes-S du DES. La consommation réelle de la carte contient donc une contribution venant du sous ensemble considéré et une contribution des autres parties du circuit. En théorie, ces deux contributions sont indépendantes. Néanmoins, en réalité, il peut exister une relation entre ces deux parties de consommation. La précision du modèle de consommation peut être réduite à cause de cette dépendance. De plus la modélisation peut tenir compte des déséquilibres entre les bits. Il s'agit de la

différence de comportements de consommation pour des bits différents. Ceci montre que la modélisation précise d'un modèle de consommation n'est pas une tâche aisée.

Les attaques par observation de signaux compromettants, comme leur nom l'indique, dépendent étroitement des signaux acquis. La performance peut être fortement réduite par différentes causes, les plus importantes sont :

- Le bruit : les différentes sources de bruit sont générées par le circuit lui même, par les mesures, par la quantification ou sont volontairement ajoutées par les producteurs de carte comme contre-mesure. Elles sont bien évidemment un obstacle qui empêche la détection de la clé secrète.
- Le désalignement des signaux : comme le bruit, le désalignement des signaux est provoqué par des sources involontaires et des sources volontaires. Les informations utiles pour la détection sont dispersées et par conséquent la performance de l'attaque peut être nettement réduite.

Il est donc utile d'étudier ces signaux et d'utiliser des techniques de traitement du signal, qui vont permettre de supprimer le bruit Gaussien, de diminuer les conséquences du désalignement ou bien d'exploiter des signaux provenant de capteurs différents.

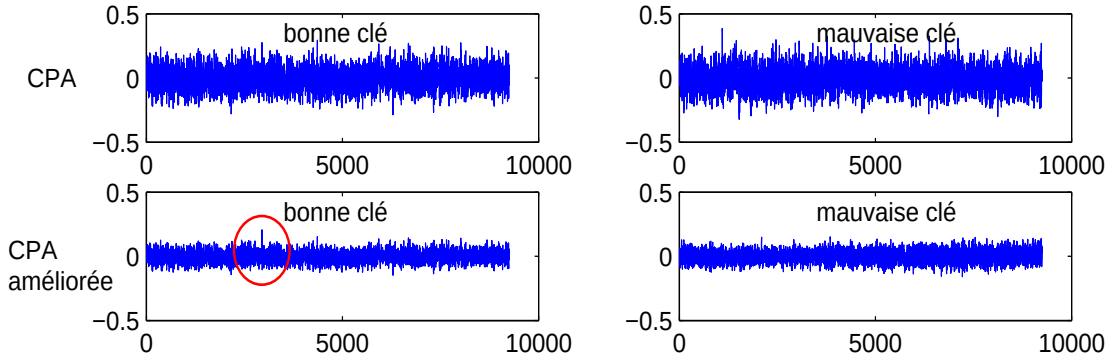


FIG. 2 – Signaux de la CPA et CPA améliorée obtenus avec 200 signaux électromagnétiques.

La figure 2 montre par exemple le gain obtenu au niveau du nombre de mesures utiles, grâce une technique de traitement du signal.

5 Conclusion

La carte à puce constitue un bon support pour stocker des secrets. Ses capacités de traitement sont de plus en plus grandes et permettent des calculs cryptographiques rapides avec des tailles de clés suffisantes. C'est de fait un produit fortement lié à la sécurité qui est utilisé massivement dans de nombreuses applications parmi lesquelles nous pouvons citer le bancaire, les transports, l'identité, la télévision à péage. Mais la carte à puce évolue aussi dans un environnement hostile, il est donc essentiel que les systèmes cryptographiques résistent aux attaques physiques.

Références

[1] M.L. Akkar and C. Giraud. An Implementation of DES and AES Secure Against Some Attacks. In *Proceedings of CHES 2001*, pages 309–318, Paris, France, 2001. LNCS 2162, Springer Verlag.

[2] M.L. Akkar and L. Goubin. A Generic Protection Against High-Order Differential Power Analysis. In *Proceeding of FSE 2003*, pages 192 – 205, Lund, Sweden, 2003. Springer Verlag.

[3] R. Bevan and E. Knudsen. Ways to Enhance DPA. In *Proceedings of ICISC 2002*, pages 327–342. LNCS 2587, Springer Verlag, 2002.

[4] D. Biham and A. Shamir. Differential fault analysis of secret key cryptosystems. In *Proceedings of EUROCRYPT*, pages 513–525, Konstanz, Germany, 1997. LNCS 1294,Springer Verlag.

[5] E. Biham and A. Shamir. Power Analysis of the Key Scheduling of the AES Candidates. In *Proceedings of the 2nd Advanced Encryption Standard Candidate Conference*, Rome, Italy, 1999.

[6] D. Boneh, R.A. Demillo, and R.J. Lipton. On the importance of checking cryptographic protocols for faults. In *Proceedings of CRYPTO*, pages 37–51, Santa Barbara, California, USA, 1997. LNCS 1233, Springer Verlag.

[7] E. Brier, C. Clavier, and F. Olivier. Correlation power analysis with a leakage model. In *Proceedings of CHES 2004*, pages 16–29, Boston, USA, 2004. LNCS 3156, Springer Verlag.

[8] E. Brier and M. Joye. Weierstrass elliptic curves and side channel attack. In *Proceedings of PKC*, pages 335–345, Paris, France, 2002. LNCS 2274.

[9] C. Canovas and J. Clediere. What do S-boxes Say in Differential Side Channel Attacks ? Cryptology ePrint Archive, Report 20085/311, 2005.

[10] S. Chari, C. Jutla, J. Rao, and P. Rohatgi. A Cautionary Note regarding Evaluation of AES Candidates on Smart-Cards. In *Proceedings of the 2nd Advanced Encryption Standard Candidate Conference*, Rome, Italy, 1999.

[11] J.S. Coron. Resistance against differential power analysis for elliptic curve cryptosystems. In *Proceedings of CHES*, Massachusetts, USA, 1999. LNCS 1717.

[12] J.S. Coron and L. Goubin. On Boolean and Arithmetic Masking Against Differential Power Analysis. In *Proceedings of CHES 2000*, pages 231–237, Massachusetts, USA, 2000. LNCS 1965 , Springer Verlag.

[13] J.S. Coron, P. Kocher, and D. Naccache. Statistics and Secret Leakage. In *Proceedings of Financial Cryptography*, pages 157–173. LNCS 1972, 2000.

[14] J.F. Dhem, F. Koeune, P.A. Leroux, P.Mestré, J.J. Quisquater, and J.L Willems. A practical implementation of the timing attack. In *Proceedings of CARDIS*, LNCS 1820, pages 167–182, Belgium, 1998. Springer.

[15] R. Elbaz, L. Torres, G. Sassatelli, P. Guillemin, C. Anguille, C. Buatois, and J.B. Rigaud. Hardware engines for bus encryption : a survey of existing techniques. In *Proceedings of Design, Automation and Test in Europe*, pages 40–45, Munich, March 2005.

[16] K. Gandolfi, C. Mourtel, and F. Olivier. Electromagnetic Attacks : Concrete Results. In *Proceedings of CHES 2001*, pages 252–261, Paris, France, 2001. LNCS 2162, Springer.

[17] L. Goubin and J. Patarin. DES and Differential Power Analysis : The Duplication Method. In *Proceedings of CHES 1999*, pages 158–172, Massachusetts, USA, 1999. LNCS 1717, Springer Verlag.

[18] G. Hachez and J.J Quisquater. Montgomery exponentiation with no final subtraction : Improved results. In *Proceedings of CHES*, pages 293–301, Massachusetts, USA, 2000. LNCS 1965.

[19] H. Handschuh and H.M. Heys. A Timing Attack on RC5. In *Proceedings of the Selected Areas in Cryptography*, pages 306–318, Kingston, Ontario, Canada, 1998. LNCS 1556.

[20] M. Joye, P. Paillier, and B. Schoenmarkers. On second-order differential power analysis. In *Proceedings of CHES 2005*, pages 293–308, Edinburgh, Scotland, 2005. LNCS 3659, Springer Verlag.

[21] M. Joye and J.J. Quisquater. Hessian elliptic curves and side channel attack. In *Proceedings of CHES*, Paris, France, 2001. LNCS 2162, Springer Verlag.

[22] P. Kocher. Timing Attack on Implementation of Diffe-Hellman, RSA, DSS and other Systems. In *Advances in Cryptology - Crypto'96*, pages 104–113, New York, 1996. LNCS 1109, Springer Verlag.

[23] P. Kocher, J. Jaffe, and B. Jun. Differential power analysis. In *Proceedings of CRYPTO 1999*, pages 388–397, Santa Barbara, California, USA, 1999. LNCS 1666, Springer Verlag.

[24] R. Mayer-Sommer. Smartly analysing the simplicity and the power of simple power analysis on smartcards. In *Proceedings of CHES 2000*, pages 78–92, Massachusetts, USA, 2000. LNCS 1965, Springer Verlag.

[25] T.S. Messerges, E. A. Dabbish, and R. H. Sloan. Examining smart-card security under the threat of power analysis attacks. volume 51, pages 541–552, May 2002.

[26] T.S. Messerges, E.A. Dabbish, and R.H. Sloan. Power analysis of modular exponentiation in smart-cards. In *Proceedings of CHES*, pages 144–157, Massachusetts, USA, 1999. LNCS 1717.

[27] T.S. Messerges, E.A. Dabbish, and R.H. Sloan. Securing the AES Finalists Against Power Analysis Attacks. In *Proceedings of FSE 2000*, New York, USA, 2000. Springer Verlag.

[28] T.S. Messerges, C.K. Koc, and P. Christof. Using Second-Order Power Analysis to Attack DPA Resistance Software. In *In proceedings of CHES 2000*, pages 238–251, Massachusetts, USA, 2000. LNCS 1965, Springer Verlag.

[29] P. Paillier. Evaluating Differential Fault Analysis of Unknown Cryptosystems. In *Proceedings of PKC*, pages 235–244, Kamakura, Japan, 1999. LNCS 1560.

[30] E. Peeters, F.X. Standaert, and J.J. Quisquater. Power and electromagnetic analysis : Improved model, consequences and comparisons. In *INTEGRATION, the VLSI Journal*, volume 40. Elsevier Science.

[31] J.J. Quisquater and D. Samyde. Electromagnetic Analysis (EMA) : Measures and Countermeasures for Smart Cards. In *Proceedings of e-Smart 2001*, Sophia, Antipolis, France, 2001. LNCS 2140, Springer.

[32] J.R. Rao and P. Rohatgi. EMpowering Side-Channel Attacks. Cryptology ePrint Archive, Report 2001 037, 2001.

[33] W. Schindler. A Timing Attack against RSA with the Chinese Reminder Theorem. In *Proceedings of CHES*, pages 109–124, Massachusetts, USA, 2000. LNCS 1965.

[34] S. Skorobogatov and R. Anderson. Optical Fault Induction Attack. In *Proceedings of CHES*, pages 2 – 12, San Francisco Bay, USA, 2002. LNCS 2523.

[35] J. Waddle and D. Wagner. Towards efficient second-order power analysis. In *Proceedings of CHES 2004*, pages 1–15, Boston, USA, 2004. LNCS 3156, Springer Verlag.

[36] S.M. Yen, S. Kim, S. Lim, and S. Moon. RSA Speedup with Residue Number System Immune against Hardware Fault Cryptanalysis. In *Proceedings of ICISC*, pages 397–413, Seoul, South Korea, 2001. LNCS 2288.

Cartes sans contact, dual mode et combinatoires, Nouvelles problématiques sécuritaires

Nathalie Feyt [1] ,Christophe Mourtel [2]

[1]Thales CEACI
TSS Toulouse
18 Avenue Edouard Belin BPI1414
31401 Toulouse Cedex 9
nathalie.feyt@thalesgroup.com

[2] Gemalto
La Vigie, Avenue du Jujubier, ZI Athelia IV
13705 La Ciotat Cedex
christophe.mourtel@gemalto.com

Résumé : Quatre industriels français, Inside Contactless, Raisonance, Thales CEACI et Gemalto se sont associés il y a deux ans autour d'un projet RNRT (le projet DRASTIC) pour étudier d'un point de vue sécuritaire les spécificités des produits sans contact et combinatoires. Leur but était d'identifier les problématiques sécuritaires de ces composants afin de définir les outils et les méthodes qui permettent de garantir la sécurité des systèmes d'informations utilisant ces produits. L'article qui suit illustre les résultats obtenus sur ce projet.
Abstract : Four French companies - Inside Contactless, Raisonance, Thales CEACI et Gemalto joined their expertise's two years ago to conduct a security analysis on combined contact and contactless products. This project labeled DRASTIC was funded by the French Ministry of Industry. First objective is to clarify the potential new threats coming from the combination of those technologies, then as a second objective, guidelines were produced to demonstrate that state-of-the-art security can be also reached. Finally, a set of tools and methodology has been developed to be capable of asserting the security level on combined contact and contactless products.

1. Introduction

La dénomination « produit sans contact » regroupe communément des catégories très différentes de produits. Citons notamment l'existence de plusieurs principes de fonctionnement (infrarouge, liaison optique, liaison radio- ou hyperfréquences), de mode d'alimentation (télé- ou auto-alimentés), de distances ou de modes de fonctionnement. Nous souhaitons dans cette présentation nous concentrer sur l'analyse des produits cartes à puces sans contact fonctionnant selon la norme ISO 14443.

L'apparition de la norme en 2000 a permis une unification des produits sans contact, limitant le développement de produits aux modes de fonctionnement propriétaires.

Si l'utilisation de carte à puce à contact s'est banalisée, celle de cartes sans contact commence juste à se développer. Des notions comme l'interopérabilité et les contraintes de fonctionnement sont sans doute des limitations à l'expansion de cette technologie, mais d'autres notions comme la sécurité ou l'intimité du porteur sont souvent décriées. Nous essayerons de mettre en évidence au travers de nos résultats ce qui relève du fantasme et ce à quoi il faut réellement veiller lors de l'usage de produits sans contact. Nous démontrerons ensuite qu'une utilisation et une sécurité adaptée à ces produits ne remet pas en cause la sécurité d'une application.

2. Les principes de fonctionnement

a. Energie de fonctionnement de la carte

Il faut distinguer deux catégories de produits, ceux qui comportent une batterie embarquée et ceux qui reçoivent leur énergie du champ électromagnétique dans lequel ils sont placés. Les premiers sont dits auto-alimentés ou actifs, les seconds télé-alimentés ou passifs. Les cartes à puce sans contact auxquelles nous nous intéressons font partie de la deuxième catégorie de produit. Elles reçoivent leur énergie du champ électromagnétique alternatif, émis par l'antenne du lecteur devant lequel elles sont placées pour fonctionner. Ce champ électromagnétique rayonne à partir de l'antenne du lecteur dans une sphère (ou demi-sphère) autour de celle-ci. La carte à puce accordée sur la fréquence du champ électromagnétique, va ainsi récupérer grâce à sa propre antenne de communication l'énergie du champ électromagnétique et la transformer en tension pour son propre fonctionnement.

b. Transmission de données

La modulation du champ électromagnétique sert de principe de transmission des données du lecteur vers la carte et réciproquement. Pour des questions de robustesse de la communication, le schéma de modulation peut prendre différentes formes (modulation d'amplitude, de fréquence, saut de phase, modulation avec sous porteuse). La fréquence de l'onde du champ électromagnétique sert aussi d'horloge à la carte à puce afin de cadencer ses opérations.

c. Fréquences de fonctionnement

Plusieurs fréquences de fonctionnement (fréquence du champ électromagnétique) existent (125 KHz, UHF ou hyperfréquences). Certaines étant plus utilisées que d'autres. La fréquence de 13.56 MHz est celle qui a été retenue pour les applications de sécurité. Elle est l'objet du plusieurs normalisations (ISO 14443 ,ISO 15693 et plus récemment ISO 18092). Cette fréquence a été retenue pour plusieurs raisons. La première est que cette gamme de fréquences permet de réduire la taille (et la valeur) des éléments nécessaires à l'accord en fréquence. Une autre raison est que l'augmentation de la fréquence du champ électromagnétique autorise l'augmentation du débit d'information entre la carte et le lecteur permettant d'atteindre facilement un débit de l'ordre du Méga bits par seconde (à comparer à la dizaine de kilo bits par seconde dans le cas du 125 KHz). La contre partie de cette gamme de fréquences est la sensibilité aux conditions environnementales. La présence de métal aux environs de l'antenne du lecteur ou de la carte modifie fortement les performances de l'ensemble.

d. Classification des produits

L'ensemble des produits sans contact est communément regroupé sous l'appellation RFID, "Radio Frequency Identifier". C'est une classification qui ne permet pas de distinguer ces produits à faibles capacités fonctionnelles et aux protocoles de communications propriétaires, des cartes sans contact de proximité pour les produits à bases de microcontrôleurs et utilisés dans des applications de sécurité. Les cartes sans contact de proximité sont par ailleurs totalement conformes dans leur mode de fonctionnement à la norme ISO 14443 dédiée aux produits de proximité. Nous garderons ici le terme de RFID pour cette première catégorie de produit à faible capacité fonctionnelle et nous distinguerons les tags RFID des cartes de proximité sans contact.

RFID

Les produits RFID englobent 70 à 80 % des produits sans contact. Leur utilisation est destinée à des applications où les recommandations sont la faible consommation, la grande distance de fonctionnement et le bas coût des produits. Cela correspond notamment à des applications de traçabilité, d'étiquetage.

CARTE SANS CONTACT DE PROXIMITE

Nous retrouvons sous cette dénomination l'ensemble des produits au format carte à puce et plus récemment les produits dédiés aux passeports et carte d'identité électronique. Les points communs de ces produits sont la distance de fonctionnement inférieure à 10 cm, la capacité de procéder à des opérations de cryptographie à base d'algorithme à clé publique ou secrète les désignant comme les produits idéaux à utiliser dans des applications de sécurité et d'identité.

CARTE DUAL MODE

La définition d'un produit sans contact correspond aux produits les plus fréquemment utilisés, ils ne disposent pour communiquer qu'une interface de communication, l'interface sans contact. Plus récemment ont été introduits les produits dits dual mode. Il s'agit de produits disposant de deux interfaces de communication, une classique à contact (relative à l'ISO 7816) et une autre sans contact (relative à l'ISO 14443). Ces produits peuvent dialoguer et exécuter des commandes venant de l'une des ces deux interfaces de communication. Le produit au cours d'une même session ne peut dialoguer qu'au travers d'une des deux interfaces, l'autre étant désactivée le temps de la session. Trois scénarios de fonctionnement différents peuvent être définis, liés chacun à une règle de priorité :

Priorité à l'interface contact :

L'interface de communication à contact est prioritaire sur l'interface sans contact. Ce scénario impose que si l'interface à contact est la première à être activée par la montée de la ligne reset suivie de l'envoi d'une commande APDU, l'interface sans contact ne sera pas activable tant que l'interface à contact n'aura pas fini de traiter sa commande et/ou n'aura pas été désactivée. Dans le cas où l'interface de communication sans contact est activée alors qu'une activation de l'interface de communication à contact est sollicitée, il est mis fin de manière prématurée à la communication sur l'interface sans contact. Ce cas de figure impose le reset général du produit afin que son état logique au début du traitement de la commande de l'interface à contact soit identique à celui qui existerait si la carte était sollicitée uniquement par cette interface.

Priorité à l'interface sans contact :

L'interface de communication sans contact est prioritaire sur l'interface à contact. Ce scénario impose que si l'interface sans contact est la première à être activée, l'interface à contact ne sera pas activable tant que l'interface sans contact n'aura pas fini de traiter sa commande et/ou n'aura pas été désactivée.

Dans le cas où l'interface de communication à contact est activée alors qu'une activation de l'interface de communication sans contact est sollicitée, il est mis fin de manière prématurée à la communication sur l'interface à contact. Ce cas de figure impose le reset général du produit afin que son état logique au début du traitement de la commande de l'interface sans contact soit identique à celui qui existerait si la carte était sollicitée uniquement par cette interface.

Priorité à la première interface activée

Aucune des deux interfaces n'est prioritaire sur l'autre. Le produit ne pouvant fonctionner qu'au travers d'une interface à la fois, c'est la première interface à devenir active sur le produit qui est prioritaire sur l'autre. Tant qu'une interface est active sur le produit, l'autre n'est pas activable et toute sollicitation d'activation de celle-ci ne sera pas traitée/vue par le produit.

CARTE COMBI

Enfin nous distinguerons les produits combi ou combinatoires qui sont des produits de type dual mode ayant en plus la possibilité de dialoguer au travers des deux interfaces de communication durant une même session (commandes reçues alternativement ou concomitamment par l'interface contact ou sans contact).

Le produit ne disposant que d'un cœur de traitement (CPU+mémoires+périphériques cryptographiques) commun aux deux interfaces une seule commande est traitée à la fois. Il faut néanmoins définir plusieurs modes distincts de fonctionnement :

- *Mode 1: mode séquentiel*

Dans ce mode une commande APDU en cours de traitement n'est pas interrompue par l'arrivée d'une APDU sur la seconde interface. Cette seconde APDU sera traitée lorsque l'APDU en cours aura fini de l'être. Ce mode peut ressembler au mode de fonctionnement des produits dual mode, à la différence près que le traitement consécutifs de deux APDU ne provenant pas de la même interface, ne nécessite pas le reset général du produit. Ce reset peut exister si les applications le nécessitent mais il n'est pas obligatoire. Ce mode de fonctionnement est identique aux problématiques de sélections d'applications d'une carte multiapplicative sur des canaux logiques différents. Cependant la différence avec la problématique des canaux logiques, c'est que l'horloge et les I/O, voire le Vcc, seront différents entre les deux APDUs. Il est alors impératif de gérer un contexte logiciel associé aux interfaces de communication et donc à leur APDU.

- *Mode 2 : mode entrecoupé*

Ce mode de fonctionnement autorise l'interruption de traitement d'une APDU de l'interface non prioritaire si une APDU arrive pendant ce temps là sur l'interface prioritaire. Le traitement de l'APDU non prioritaire est repris, si cela est possible, à la fin du traitement de l'APDU de l'interface prioritaire. Ce mode de fonctionnement ne peut pas autoriser le reset du produit lors du changement de traitement des APDUs. Cette absence de reset pose une problématique de sauvegarde des contextes matériels et logiciels du produit. Cette sauvegarde des contextes doit assurer un cloisonnement strict entre les contextes des interfaces et de leur application respective.

- *Mode 3 : mode entrelacé*

Ce mode de fonctionnement ne définit pas d'interface prioritaire. L'APDU d'une interface peut être traitée simultanément avec celle de l'autre interface. Pour cela le CPU exécute ces traitements en temps partagé en entrelaçant les exécutions. Comme pour le mode 2 le reset du produit n'est pas envisageable lors du changement d'interface.

Chacun de ces modes peut se retrouver utilisé dans les différentes applications sans contact, toute mauvaise gestion de changement de contexte peut entrainer de nouvelles brèches de sécurité qui

s’ajouteront à celles déjà connues des produits contact et sans-contact. C’est ce que nous allons détailler dans le chapitre suivant.

3. Qu’avons-nous de plus à protéger ?

Les problématiques des produits combinatoires étant définies, il nous faut maintenant identifier les biens à protéger associés à ces produits. Nous avons réparti ces biens en quatre grandes catégories, le composant, le code, les données et la perception du porteur face à l’utilisation de sa carte. Nous n’allons détailler pour chacune de ces catégories l’ensemble des biens à protéger qui y sont associés. Nous nous attacherons surtout au point nouveau amené par la technologie sans-contact : la perception du porteur.

On retrouve classiquement comme pour les produits contact tous les mécanismes de stockage de données et de sécurité d’exécution pour les puces et leurs logiciels embarqués, bien sûr on y associe tous leurs mécanismes de protection propres (tamper-evident, tamper-resistant).

a. Biens usuels

Pour assurer une authentification, il faut utiliser toutes les données propres à identifier un individu. Plusieurs types de données sont présentes nous les dénommerons de la manière suivante :
ce que l’on est : données biométriques, ce que l’on sait : valeur de PIN, ou mot de passe et ce que l’on a : valeur de clefs stockées dans un module portable, personnel, sécurisé.

DONNEES BIOMETRIQUES

Mises de plus en plus en avant dans des applications, ces données bien que publiques (type empreintes digitales), sont tout de même à considérer comme sensibles. Leur encodage dans une application est parfois secret, et c’est la connaissance de l’encodage ajouté à la donnée elle-même qui doit être retrouvé par un attaquant, certaines sont sensibles car pas accessibles sans démarche volontaire du porteur (données de scan d’iris, données de reconnaissance faciales, réseau veineux d’une main)

VALEUR DE PIN

Nombres d’applications utilisent le PIN pour faire le lien entre le module sécurisé et le porteur. C’est le seul moyen de s’assurer de l’identité du porteur.

VALEUR DE CLEFS

Toutes les applications qui utilisent de la cryptographie comme moyen d’obtenir des services d’authentification ou de confidentialité, s’appuient sur des clefs partagées dans un système ou propres à un porteur de cartes. On conçoit que la perte, la duplication, ou la compromission de ces clefs peut entraîner l’écroulement de ces applications.

b. Perception du porteur

Cette rubrique est introduite car le fonctionnement des produits sans contact introduit la possibilité de communiquer avec un produit sans qu’il existe une matérialisation évidente de celle-ci pour le porteur du produit. Cette absence de matérialisation est une nouveauté pour les utilisateurs de produits carte à puce. Face à cette nouveauté il convient de rassurer le porteur sur les risques liés à l’utilisation

de son produit à son insu. Quatre notions sont introduites afin d'assurer le porteur qu'aucune transaction ne peut avoir lieu à son insu :

- Perception d'une possible communication
  Cette notion informe le porteur qu'il est dans une zone où des lecteurs peuvent être amenés à communiquer avec son produit. L'idée est d'avertir le porteur qu'il entre dans une zone équipée de lecteur sans contact afin qu'il rende son produit inopérant s'il ne souhaite pas s'en servir ou risquer que l'on adresse son produit à son insu.

- Acceptation d'une transaction
  On introduit ici la notion qui à contact consiste à insérer sa carte dans un lecteur. Par cet acte le porteur accepte l'établissement d'une transaction. Il est fondamental que les produits capables de communiquer via une interface sans contact possèdent un acte de la part du porteur qui recrée cette acceptation.

- Validation d'une transaction
  La validation d'une transaction introduit une connaissance ou un acte du porteur que lui seul peut donner ou faire. Il s'agit ici d'empêcher qu'une transaction complète puisse avoir lieu sans que le porteur ait à la valider. Cela peut par exemple consister en la présentation d'un code secret qui n'autorise pas la validation d'une transaction s'il n'est pas présenté.

- Authentification de la validité d'un lecteur sur l'ensemble d'une transaction
  Enfin cette dernière notion est introduite pour s'assurer qu'une communication a lieu du début à la fin par l'intermédiaire d'un même et unique lecteur. En effet l'absence de lien physique permet d'envisager qu'une transaction soit initiée par un lecteur licite mais qu'elle soit augmentée (en terme de commandes envoyées) par un lecteur pirate qui se servirait de l'établissement initial d'une communication pour récupérer des données auxquelles il n'aurait pas le droit d'accéder sans cela. Cette authentification du lecteur impose que la notion soit testée et valide tout au long de la transaction.

L'introduction de ces notions dans les transactions sans contact, assurera la confiance du porteur. S'il paraît à première vue quelque peu étonnant d'identifier ce ressenti comme un bien, cette notion conditionnera l'acceptation de ce nouveau moyen technique que constitue la communication sans contact. Sans cette acceptation, l'utilisation et le déploiement des produits sans contact et combinatoires ne pourra avoir lieu. Bien plus que les sécurités embarquées dans le produit, c'est cette notion qui convaincra le porteur d'utiliser son produit par l'interface sans contact – notamment dans des applications de paiement.

Nous avons commencé à l'évoquer mais l'identification des biens et des menaces n'aurait pas d'intérêt s'il n'y avait pas d'attaques réalisables permettant de concrétiser les menaces sur les biens identifiés.

4. Rapide présentations des menaces connues sur les produits sans-contact

La communication entre un produit sans contact et un lecteur possède des particularités qu'il faut correctement analyser afin de les prendre en compte dans le développement d'une application sans contact. Ces spécificités peuvent se répartir en trois grandes familles : la communication, la sécurité physique, la sécurité logicielle. Dans ces deux dernières catégories nous trouverons beaucoup de menaces et donc de contre-mesures qui sont aussi présentes sur les produits classiques à contact.

a. Vitesse et distance de fonctionnement

Ces deux notions antagonistes pourraient résumer à elles seules ce qui fait la particularité du sans contact. Rappelons que l'énergie dont dispose la carte dépend notamment de la géométrie de l'antenne du lecteur, de la puissance émise par celui-ci et de la distance qui sépare la carte de l'antenne. Des liens complexes existent également entre vitesse et distance de fonctionnement. Ainsi les notions de vitesse et de puissance de calcul sont antagonistes de celle de distance de fonctionnement. Quelle en sont les répercutions sur la sécurité ? Si on observe la retro-modulation du champ par la carte à puce durant le temps où elle utilise de la puissance de calcul, cette retro-modulation sera plus marquée sur le signal global de la porteuse à certaines distances qui constituent l'optimum du ratio retro-modulation carte/ porteuse. Ainsi, à ces distances de fonctionnement, l'activité de la carte est plus visible. C'est donc un canal caché -side-channel- nouveau.

Ce canal caché peut être magnifié selon les conditions environnementales et permettre d'être observé à distance, sans altérer le produit.

Comme tous les produits cartes à puce qui veulent se prémunir d'attaques par side-channel type SPA, DPA, CPA,.., des contre-mesures qui diminuent la fuite d'information ou qui brouillent l'interprétation du comportement de la carte peuvent être suffisantes.

b. L'absence de lien physique

Cet avantage par rapport à une communication à contact, permet une communication plus aisée grâce à l'absence de liaison physique point à point entre les deux parties. Cela veut dire qu'il faut s'assurer en permanence que le lecteur ou la carte communique avec la tierce partie qu'il a identifiée auparavant. Il faut aussi tenir compte du fait que plusieurs cartes peuvent être en même temps dans le champ du lecteur, qu'une carte peut entrer ou sortir du champ à tout moment. Enfin il est nécessaire de considérer que l'on peut dans une volonté de nuire substituer une des parties.

Ces points critiques se résolvent en instaurant une communication où toutes les parties se reconnaissent et s'appairent par un échange de secrets qui permettront à tout moment de vérifier que l'on dialogue avec la bonne tierce partie. Ce schéma existe déjà et peut être réalisé par une authentification mutuelle entre les parties au début de chaque nouvelle communication.

c. L'espionnage

On vient de l'évoquer, la communication entre la carte et le lecteur se faisant par l'intermédiaire d'un champ magnétique, il est possible de capter celui-ci et de l'analyser afin de collecter les informations échangées entre la carte et le lecteur. Afin que cet espionnage de la communication, qu'il n'est pas possible d'empêcher, soit sans conséquence sur la sécurité de

l'application, il faut que l'ensemble des échanges de données se fasse de manière sécurisée. Des méthodes existent déjà qui consistent à chiffrer la communication entre la carte et lecteur avec échange préliminaire de clés afin d'empêcher le rejeu.

d. Utilisation à l'insu du porteur

La communication sans contact permet d'envisager qu'un lecteur non autorisé, initie un dialogue avec une carte à l'insu du porteur. Cette menace est ce qui freine aujourd'hui le plus l'acceptation de la technologie par les utilisateurs. Des moyens simples permettent d'annuler ce risque. Ces moyens dépendent de l'application, mais on peut citer la présentation d'un code secret par l'utilisateur avant que toute transaction ne soit possible. L'instauration comme nous l'avons déjà dit de schéma de sécurité (authentification mutuelle, échange de clés…) est aussi une réponse au problème car il impose au lecteur d'avoir les bons secrets liés à l'application.

e. Intimité du porteur

Si la protection d'une application par des méthodes de sécurité que nous venons de décrire résout un certain nombre de problèmes, certains demeurent. Revenons en effet aux commandes de bases définies dans la norme qui permettent de sélectionner un produit. Ces commandes sont partagées par l'ensemble des applications utilisant des produits normés. Or, la norme impose à une carte de répondre aux interrogations du lecteur. Cela peut poser un problème de non-respect de l'intimité car on peut à l'insu du porteur obtenir des informations permettant de le tracer et de l'identifier.

La compatibilité des cartes avec l'ISO 14443, permet de limiter ces informations au numéro de série de la carte. Heureusement la norme permet de couper le lien entre le support physique et la puce en autorisant la carte à renvoyer un numéro de série aléatoire en réponse aux interrogations du lecteur.

f. Déni de service

Ce problème est lié à des actions de piratage sur une application. On peut envisager qu'un groupe mal intentionné puisse, dans un lieu de fortes concentrations de personnes (métro, aéroport…), à l'aide de lecteurs pirates bloquer les cartes sans contact. Imaginons que toutes les cartes de transport dans le cas d'un métro, d'embarquement dans celui d'un aéroport soient bloquées ou endommagées à l'aide de lecteurs pirates, le gain pour les attaquants est faible par contre il peut mettre à mal tout un système de transport entraînant notamment des retards… Afin de se prémunir de ce type d'actions malveillantes, les solutions proposées comme l'étui de blindage ou la désactivation de la carte hors utilisation voulue peuvent être utilisées.

A titre d'illustration, voici quelques exemples de mise en œuvre des principes d'attaques énoncés précédemment.

- Jouer des commandes publiques à l'insu du porteur pour
  - Faire la liste de toutes les cartes d'un porteur
  - Faire la liste de toutes les applications d'une carte
  - Tracer une carte par son numéro de série
- Jouer des transactions à l'insu du porteur
  - Attaques en relais
  - Transaction opérée sur plusieurs cartes du porteur au lieu d'une
  - Transaction opérée sur une carte au lieu d'une autre
- Jouer sur les conditions environnementales

        - Exploitation d’une mauvaise gestion du reset
        - Variation intentionnelle des conditions de fonctionnement entraînant une mauvaise initialisation, un mauvais reset,…
    - Jouer sur le lien distant
        - Remplacement d’une carte par un émulateur à l’insu du marchand

Nous avons vu ici que pour chacune des menaces présentées, une grande variété de solutions existent et peuvent être proposées par les industriels. Les biens à protéger restent les mêmes et beaucoup du savoir faire sécurité du monde contact peut-être directement réutilisé dans le monde sans contact.

5. Nouvelles problématiques sécuritaires des produits dual mode et combinatoire

Les attaques réalisables sur les produits contact, sans contact, dual mode ou combinatoires seront pour partie communes et pour partie spécifiques ou adaptées à la nature de(s) l’interface(s) de communication. Ici, nous nous attachons à celles qui sont spécifiques.

Nous distinguerons dans la suite deux groupes d’attaques les attaques des produits dual mode et celles des produits combinatoires. Précisons que les attaques des produits dual mode seront valables pour les produits combinatoires puisque les produits combinatoires peuvent fonctionner selon le mode d’utilisation des produits dual mode, l’inverse n’étant pas vrai.

a. Initialisation et de reset des produits combinatoires

Les produits combinatoires introduisent dans l’utilisation des cartes à puce une complexité nouvelle d’utilisation et de gestion. Cette complexité liée à l’utilisation simultanée de deux interfaces de communication impose une étude approfondie des menaces et des risques associés.

Le fonctionnement des produits combinatoires nécessitent que l’on s’intéresse également à aux conditions d’initialisation et de reset du produit et des interfaces. En effet un produit mono interface ou dual mode ne fonctionne qu’à travers une seule interface à la fois. Le contexte applicatif, logiciel et matériel n’a pas à être sauvegardé entre chaque session ou traitement d’APDU car les scénarios d’utilisation prennent en compte le reset complet du produit entre chaque session ou lors du changement d’interface dans le cadre des produits dual interface. Ce reset général permettant de démarrer une session sur la nouvelle interface sans qu’il subsiste dans le produit de « traces » de contexte logiciel et matériel de la session précédente. Il en va autrement dans le cas des produits combinatoires. En effet les deux interfaces étant actives, il faut envisager les scénarios d’initialisation et de reset du produit en conséquence.

On peut citer pour plus de détails :

- Exploitation d'une mauvaise gestion des reset et des actions associées (nettoyage des mémoires, clefs de chiffrement des mémoires...)
- Exploitation d'une mauvaise gestion d'interruption de communication sur une interface, l'autre restant active.
- Exploitation de la mauvaise initialisation lors de l'ouverture de la deuxième interface (exploitation d'actions incorrectement effectuées telles que nettoyage des mémoires, clefs de chiffrement des mémoires...)
- Exploitation d'une mauvaise restauration de contexte hardware lors d'un switch entre interfaces
- Exploitation d'une mauvaise gestion des accès concurrents

Ces exploitations nécessitent évidemment une phase d'apprentissage sur le produit pour en connaitre ses faiblesses et voir si elles sont potentiellement exploitables. Ces problèmes sont plus proches d'une exploitation de bugs ou d'erreurs de conception que de vulnérabilité propre à la technologie. Nous pourrions tout à fait avoir les mêmes types de problèmes sur un produit contact ayant de multiples interfaces (USB, MMC…). On comprend qu'il est facile de s'en prémunir en n'oubliant pas de qualifier ces passages fonctionnels d'une interface à l'autre et ce dans tous les modes possibles.

b. Cohabitation des modes contact et sans contact sur le même silicium

L'idée est ici de jouer sur la possible incohérence des protections choisies par le développeur entre les deux interfaces : ces protections peuvent être différentes car elles répondent à des contraintes de vitesse et de consommation dissociées.

- Utilisation de l'interface la plus vulnérable

De la même manière on peut essayer de perturber une des interfaces alors que l'autre est sollicitée pour traiter des opérations.

- Stress de l'interface contact pendant l'utilisation de l'interface sans contact ou vice-versa en vue d'injection de fautes
- Maintien d'un Vcc sur une des interfaces afin d'empêcher un RESET de la carte

Concernant cette analyse, on peut juste conseiller au développeur d'éviter de supprimer des contre-mesures existantes sur une interface notamment à contact, dans un autre mode sans-contact que ce soit pour des contraintes de vitesse ou de consommation. S'il devait le faire, il est évidemment conseillé de mener une étude sécuritaire complète sur l'interface la plus dégradée d'un point de vue sécurité pour s'assurer qu'elle garde un niveau de résistance suffisant.

c. Attaques propres aux modes entrecoupé et entrelacé (crosstalk)

Le problème vient toujours du fait que l'on n'a pas pour des produits combinatoires de ressources propres pour chaque interface. Ainsi, des zones mémoires ou des moyens de calcul du produit risquent d'être partagées entre les deux interfaces. Si le développeur ne prend pas garde à qualifier correctement son produit surtout dans des modes entrecoupé et/ou entrelacés, alors des problèmes à l'utilisation peuvent s'avérer utilisables comme moyen d'attaque.

Par exemple :

- Exploitation d'une mauvaise restauration de contexte logiciel, notamment lors d'une opération cryptographique lors d'un switch entre interfaces, pour bénéficier d'un calcul de clef de session.

Ces problèmes peuvent être trouvés et écartés par une qualification du produit qui prend en compte le fonctionnement simultané des deux interfaces

d. Attaques propres au mode entrecoupé (handover)

On peut aussi imaginer prendre le relais d'un ensemble atomique de commandes dont le but est de dérouler une transaction sur une interface en envoyant des commandes avec l'autre interface. Si le produit ne distingue pas ce changement de canal de communication, on peut se retrouver à bénéficier de privilèges sur cette interface que l'on aurait pas sur l'autre.

- Exploitation d'une mauvaise définition des priorités entre les interfaces

Là encore, ces problèmes peuvent être trouvés et écartés par une qualification du produit qui prend en compte le fonctionnement simultané des deux interfaces.

6. Conclusion

Cette analyse de vulnérabilité fait apparaître que l'ensemble de biens à protéger ne varie pas entre les applications contact/sans-contact/dual interface et combi. La présence d'une interface sans contact sur un produit contact, amène à considérer tous les moyens nouveaux d'attaques propres aux applications sans contact, en même temps que ceux connus dans le monde du contact, mais la nouveauté des produits combi, vient de l'influence croisée que peuvent avoir les applications dédiées à chaque interface.

Des solutions existent. Il suffit de connaître en détails les nouvelles menaces que la cohabitation des ces deux interfaces amènent, additionnées de celles connues aujourd'hui sur les deux modes de communication (contact et sans-contact).

Des méthodes d'évaluation sécuritaires mises en place dans les laboratoires accrédités, sont constamment remises à jour pour suivre les évolutions technologiques - nouvellement NFC, par exemple-. Elles fournissent ainsi un outil d'analyse de risque très précis aux industriels confrontés aux choix toujours présents entre sécurité, coûts et performance.

L'industrie a ainsi tous les moyens pour proposer des produits sans-contact, contact et combi adaptés à leurs marchés et construits à l'optimum.

Déploiement du chiffrement au CNRS

François Morris
IMPMC , CNRS
140 rue de Lourmel
75015 PARIS

Francois.Morris@impmc.jussieu.fr

Mots clés
Chiffrement, organisation du chiffrement, déploiement, recouvrement, certificat

Résumé
Depuis quelque temps le CNRS a mené une analyse des risques présentés par les données sensibles. Une cartographie des risques fait apparaitre des besoins de protection dans différents domaines : portables, mobiles, supports amovibles ; échanges de données, transferts de fichiers ; opérations de maintenance, mise au rebut de matériel ; données particulièrement sensibles ; partage de données sensibles.
La réponse logique à ces besoins est le chiffrement, il existe pléthore de solutions qui différent par la nature de l'objet chiffré (fichier, répertoire, disque), les algorithmes, les mécanismes de gestion des clés, les méthodes de recouvrement et point capital qui conditionne l'acceptation du produit, l'interface utilisateur.
Une population naturellement ouverte vers l'extérieur, voire à la culture libertaire, une organisation complexe et très décentralisée, une diversité des façons de travailler, une multiplicité des matériels et systèmes d'exploitation, rendent impossible l'adoption d'une solution unique pour tous.
Dans ce contexte difficile, une démarche pragmatique et privilégiant le possible au souhaitable a permis d'aboutir à des recommandations en matière de chiffrement.

Abstract
For some time CNRS has conducted an analysis of the risks related to sensitive data. A map of the risks show needs for protection in various fields: laptops, PDA, removable media; data exchanges, file transfers; maintenance, hardware disposal; very sensitive data; sensitive data sharing.
The natural answer is to cipher the data. There are many products differing by what is ciphered (file, directory, disk), the algorithms, the key management, the recovering method sand a very important point the interface with the user.
An open minded population with a libertarian culture, diversity in the ways of working, a complex and decentralized organization, an extreme variety of hardware and operating systems make impossible the choice of a unique solution for everyone.
In this difficulty context, a pragmatic approach preferring what is possible rather than is wanted leads to directives for ciphering data.

Si la nécessité de mettre en œuvre du chiffrement est aujourd'hui communément admise, cela doit procéder d'une réelle analyse des risques et non pas seulement du simple désir d'utiliser les possibilités fournies par la technique actuelle. Il ne faut pas oublier que le chiffrement n'est pas un objectif en soi mais une réponse technique à un besoin de protection des données.

Cartographie des risques et besoins de chiffrement

Le CNRS a conduit, à titre expérimental, dans quelques laboratoires une analyse de risques selon la méthode EBIOS (CAPSEC[1]).

La présentation des risques et des besoins de protection repose d'une part sur cette étude et d'autre par sur le fruit de l'expérience des membres du groupe de travail ayant participé à la réflexion sur le chiffrement.

Périmètre

Un premier critère est la distinction entre intérieur et extérieur. On suppose qu'à l'intérieur du périmètre, il existe des moyens de protection, des contrôles d'accès qui assurent un certain niveau sécurité, tandis que l'extérieur est un milieu a priori hostile.

La règle est que chaque fois qu'une information sort du périmètre, elle doit être protégée spécifiquement. Généralement cette protection résulte du chiffrement des données, les autres solutions comme la protection physique (transport en véhicule blindé) étant rarement réalistes. Cependant le rangement d'un petit support (clé USB mince ou mini CD par exemple) dans son portefeuille peut dans certaines situations s'avérer une solution acceptable.

Un deuxième critère est le niveau de sensibilité des données. Il s'évalue par les conséquences qu'aurait la divulgation de ces informations. Certaines informations sont suffisamment sensibles pour que même à l'intérieur, il soit nécessaire de mettre en œuvre des moyens assurant que seules les personnes autorisées puissent y accéder. Les solutions reposant sur l'utilisation de machines dédiées, de supports que l'on s'échange en mains propres montrent très vite leurs limites. Le chiffrement des données avec une distribution des clés de déchiffrement à ceux qui ont à en connaître est une bonne réponse technique.

Selon ces critères on peut établir deux catégories de risques :

- Risque élevé, chiffrement indispensable
    - Protection des mobiles (ordinateurs portables, assistants personnels, etc.) et des supports amovibles (CD, DVD, clés USB, bandes, disques, etc.). Le risque contre lequel on cherche à se prémunir est évidemment celui du vol ou de la perte qui conduirait à la divulgation d'informations sensibles. La solution technique consiste à chiffrer les supports.

    - Protection des communications. Il s'agit de se prémunir contres les écoutes ou la falsification des données transférées. La solution technique repose sur des algorithmes de chiffrement et de contrôles d'intégrité.
    - Protection des messages ou fichiers transmis. Elle se rapproche de celle des communications à la différence que la protection va se faire au niveau de l'objet transmis et non au niveau du canal de transmission.
    - Protection des données particulièrement sensibles (données à caractère personnel, brevets en cours de dépôt, etc.). Il s'agit de se protéger contre des attaques plus ou moins ciblées provenant de l'extérieur (vol, piratage) mais aussi de curiosités de l'intérieur. Il n'est pas nécessaire de chiffrer l'ensemble des données, du disque. Il est préférable de ne chiffrer que les données réellement confidentielles, pratiquement il va s'agir de fichiers ou répertoires. Cette méthode possède l'avantage de ne pas exposer les données chiffrées lorsque l'on n'y accède pas. Avec un chiffrement de l'ensemble du disque, tous les fichiers sont, dès que l'on a fourni la clé de déchiffrement, potentiellement accessibles en clair à un programme, tandis avec un chiffrement de répertoire il faut attendre que l'utilisateur ait déverrouillé l'accès à ce répertoire. Même alors, l'accès n'est possible que sous le compte de celui qui a effectué le déverrouillage (ou à son groupe, suivant le paramétrage) et surtout, les données ne sont pas copiables en clair sur un support informatique
- Risque moins élevés, chiffrement utile
    - Poste de travail à l'intérieur des locaux
    - Données occasionnelles
    - Données de moindre valeur
    - Protection des informations lors d'une réparation, d'une cession ou d'une mise au rebut de matériels. La solution de la destruction physique des supports de données n'est pas facile à mettre en œuvre, difficilement acceptable au niveau économique (comment faire admettre qu'il faut acheter un nouveau disque pour remplacer un disque défectueux qui est encore garanti ?). L'effacement efficace (n passes de réécriture) des données stockées sur les disques est une procédure contraignante, coûteuse en temps, peu valorisante (on demande d'abord au personnel en charge du parc informatique d'installer du nouveau matériel, de dépanner, pas de nettoyer des machines qui ne servent plus) voire impossible pour un disque en panne. Une solution de chiffrement comme celle appliquée pour les mobiles peut alors s'avérer très intéressante.

Recouvrement

Le chiffrement introduit en lui-même un nouveau risque, celui de ne pouvoir relire les données si à la suite de l'oubli d'un mot passe, de la perte ou de la défaillance du dispositif matériel (carte à puce, puce TPM sur la carte mère, etc.) qui stocke le secret on ne peut plus rétablir la clé de déchiffrement. Ce risque doit être évalué et des solutions (recouvrement, séquestre de clés) pour y parer être mises en œuvre.

Pour les communications la question ne se pose pas. Une fois la communication terminée on n'a nul besoin de la réécouter, ce qui d'ailleurs aurait supposé qu'on l'ait enregistrée. La seule contrainte pourrait être une obligation légale de permettre un déchiffrement a posteriori.

Pour des données temporaires comme l'espace de pagination (swap) le recouvrement est évidemment inutile.

La situation pour l'envoi de messages se rapproche de celle des communications, à la différence que c'est le contenu du message qui est chiffré alors que la transmission ne l'est pas obligatoirement. Le recouvrement n'est pas utile car il est toujours possible de demander à l'expéditeur de retransmettre son message.

Il faut toujours faire le bilan entre le coût du recouvrement ou du séquestre et celui de reconstituer les informations perdues. S'il s'agit de transporter l'information sur un support amovible d'un point à un autre, il vaut probablement mieux recommencer l'opération plutôt que procéder à un recouvrement. Pour un mobile qui serait synchronisé avec des informations stockées sur le réseau interne et sur lequel on ne créerait pas ou quasiment pas d'information hors connexion le besoin de recouvrement n'est pas manifeste.

Pour toute l'informatique mobile, le risque de perte, de vol (c'est justement pour cela que l'on chiffre) ou tout simplement de panne du matériel est très certainement supérieur à celui de l'oubli d'un secret. L'impossibilité de recouvrement n'est pas forcément une catastrophe pire que les autres. Il est donc impératif d'avoir des sauvegardes. Celles-ci ne dispensent pas d'avoir une procédure de recouvrement mais un échec dans le recouvrement ne sera pas alors forcément une catastrophe.

Il faut aussi envisager le cas où pour assurer la continuité de l'activité, il faut pouvoir relire l'information d'un portable ou d'un support amovible dont le possesseur a disparu. Cela veut dire que la conservation du secret servant au recouvrement ne doit pas être exclusivement du ressort de l'utilisateur mais que ce secret doit être conservé dans un secrétariat par exemple.

Parmi les méthodes de recouvrement il faut en distinguer deux :

- Le séquestre qui consiste à ranger en lieu sûr (coffre fort par exemple) une copie du secret ou de la clé permettant de déchiffrer les données. Cela peut être un simple (mais suffisamment robuste, pas « toto ») mot de passe écrit sur une feuille de papier et rangé dans une enveloppe. Certains objecteront qu'il va falloir acquérir des coffres forts. Ce n'est pas forcément nécessaire, il suffit que le stockage soit sécurisé, vis-à-vis des personnes contre lesquelles on veut se protéger (souvent, toutes personnes étrangères à l'unité) : un secrétariat suffit la plupart du temps.
- Le recouvrement par différents agents ce qui signifie que si le propriétaire des données est dans l'impossibilité de les déchiffrer, une autre personne sera à même de le faire.

Ces deux méthodes n'ont pas les mêmes implications en matière d'organisation et de déploiement. Dans bon nombre de cas, la première est la plus simple. C'est probablement aussi celle qui est aujourd'hui la plus répandue. On pourra objecter que si à la suite d'un sinistre le secret est détruit, il n'y a plus de recouvrement possible. Ce à quoi on peut répondre qu'il est plutôt improbable de perdre simultanément le mot de passe rangé dans le

coffre et d'oublier son mot de passe. De plus il est toujours possible d'avoir une copie du mot de passe en plusieurs endroits sûrs.

La deuxième est beaucoup plus riche en possibilités mais beaucoup plus contraignante en matière d'organisation. Le fait de pouvoir définir plusieurs agents de recouvrement situés en différents endroits permet d'assurer qu'il y aura toujours quelqu'un pour déchiffrer les données.

De toute façon la mise en œuvre de chiffrement impose d'être encore plus rigoureux sur les procédures de sauvegardes.

Typologie des solutions de chiffrement

Sans entrer dans la multitude de produits de chiffrement on peur considérer trois niveaux définis par ce qui est chiffré :

- Fichier. Permet un contrôle par l'utilisateur de qui a le droit d'accéder aux informations (DRM). Doit être appliqué à toutes les données sensibles, il ne faut pas oublier de fichiers.
- Répertoire. Emplacement pour ranger les données sensibles. Problèmes pour les caches, les fichiers temporaires.
- Disque entier (ou partition). Tout est chiffré, aucun risque d'oubli. Plus d'outil d'analyse ou de récupération en cas de problème. Impossible de protéger certaines données de la curiosité d'un administrateur. Les sauvegardes régulières sont encore plus importantes. Deux endroits pour l'authentification :
    - Pre-boot
    - Post-boot

Chacune possède ses avantages et inconvénients et ne sont pas forcément exclusives l'une de l'autre. Par exemple on peut vouloir chiffrer le disque pour protéger un serveur de fichiers contre les risques de vol et effectuer un chiffrement de répertoire pour protéger les utilisateurs entre eux.

Un autre critère est la méthode utilisée pour stocker le secret utilisé pour le chiffrement.

- Mot de passe
- Certificat
- Dispositif matériel
    - Carte à puce
    - Token
    - TPM

Certificats

Le CNRS a mis en œuvre depuis quelques années une IGC. Si jusqu'ici les certificats ont surtout servi à l'authentification, il est tentant de les utiliser aussi pour du chiffrement. Il faut cependant rester prudent.

Tout d'abord précisons que pour chiffrer une communication les certificats servent essentiellement à l'authentification des interlocuteurs. Si c'est pour utiliser à la place d'un mot de passe, un certificat stocké dans un fichier PKC#12 chiffré avec un mot de passe, le gain en matière de sécurité ne semble pas évident. Par contre si on utilise pour accroître la sécurité un dispositif matériel comme une carte à puce les certificats s'imposent. Si c'est pour partager à plusieurs des données chiffrées l'utilisation de certificats et d'une IGC facilite grandement les choses. On connaît les difficultés et les limites d'un mot de passe partagé.

Un usage correct des certificats exige de séparer le certificat servant à signer et à authentifier de celui à servant à chiffrer des données. Les contraintes ne sont pas les mêmes pour ces deux types de certificat. La clé privée dans le premier cas ne doit en aucun cas quitter son détenteur (l'idéal est la génération sur une carte à puce où par construction elle est illisible) tandis que pour le chiffrement un séquestre de la clé privée est possible voire souhaitable. La révocation d'un certificat servant à signer ou authentifier doit conduire à rejeter l'opération tandis que la révocation d'un certificat de chiffrement ne devrait pas interdire un déchiffrement, seulement de chiffrer de nouveaux documents.

L'idéal pour stocker un certificat reste un dispositif cryptographique comme une carte à puce ou un « token » USB ou un TPM. Si on a besoin de la sécurité offerte par la carte à puce, il est tout naturel d'utiliser des certificats. Si on doit se contenter de certificats stockés sur le disque et protégés par un mot de passe, il faut évaluer leur utilisation face à un simple mot de passe. La réponse n'est pas univoque. Il y a des cas où les certificats s'intègrent tout naturellement avec les produits utilisés et de fait facilitent les choses mais il y en a d'autres où cela ne fait qu'ajouter un niveau de complexité.

Questions ouvertes

Sauvegardes

Les sauvegardes doivent-elles être chiffrées ? La question est éminemment complexe. D'une part la sauvegarde, surtout si elle se trouve sur un support amovible constitue une cible évidente pour des personnes malveillantes. Pourquoi mettre en œuvre des moyens compliqués pour intercepter l'information s'il suffit de voler une bande qui contient tout en clair ? D'autre part la sauvegarde est ce qui sert pour récupérer l'information en cas de désastre. Il faut alors être sûr de pouvoir, en toutes circonstances, non seulement la relire mais aussi la déchiffrer. Ce qui impose de fortes contraintes sur la conservation des clés de déchiffrement.

Parmi les méthodes de chiffrement celles reposant sur l'usage de cryptographie asymétrique semblent préférables. On chiffre avec une clé publique. C'est uniquement pour le

déchiffrement qui ne doit être qu'une opération exceptionnelle - on ne restaure pas tous les jours son disque – que l'on a besoin de la clé privée.

Lorsque des fichiers ou plutôt des répertoires sont chiffrés sur une machine, les copier tels quels sur les sauvegardes est une bonne solution mais il faut avoir des procédures de recouvrement particulièrement fiables car il ne sera alors plus possible de compter sur des sauvegardes en clair.

Archives

Les archives posent un problème particulier. Par définition si on archive des données c'est qu'on souhaite les conserver longtemps. La pérennité des outils de chiffrement est loin d'être assuré, la conservation des clés est encore plus problématique. Dans ces conditions le chiffrement des archives ne peut être recommandé en général. Il est plutôt conseillé de stocker les archives numériques en lieux sûr comme on le ferait pour des archives sur papier. Le chiffrement ne peut être envisagé et probablement alors souhaitable que dans le cadre de la mise en œuvre d'un système d'archivage prenant en compte toutes les contraintes comme la régénération périodique des informations.

Contournement du chiffrement

Un chiffrement aussi robuste soit-il reste vulnérable à des attaques par contournement. Pourquoi essayer de casser une clé de chiffrement s'il est possible d'introduire un code malicieux sur la machine ? Un « key logger » sur une machine permettra de récupérer directement le mot de passe de l'utilisateur. Un cheval de Troie pourra intercepter la clé de chiffrement ou tout simplement le contenu en clair pour les envoyer à l'espion. Le chiffrement si nécessaire qu'il soit, ne peut être considéré comme une panacée et ne dispense pas d'autres mesures de sécurité comme la sécurisation du poste de travail.

Les produits de chiffrement eux-mêmes ne sont pas toujours aussi robustes qu'ils le prétendent. Par exemple les algorithmes de chiffrement de certaines suites bureautiques ne résistent pas plus de quelques secondes (XOR). Il ne faut pas non plus exclure l'existence de portes dérobées.

Expérience pilote de déploiement du chiffrement

Tests et évaluations de produits de chiffrement

Le besoin de chiffrement ayant été reconnu, une étude à été conduite au sein de l'Unité Réseau du CNRS (UREC[2]) afin d'évaluer des produits pouvant y répondre[3].

La première difficulté rencontrée réside dans le fait qu'il n'existe aucun produit qui soit disponible simultanément sur les différentes plateformes. Dans le contexte du CNRS où les postes de travail se répartissent sous les trois systèmes Windows, MacOS et Linux et où de plus certains fonctionnent alternativement avec l'un ou l'autre des systèmes (« dual boot ») c'est un gros problème. Peut-être que les disques récemment sortis comme ceux de Seagate[4] qui intègrent le chiffrement pourrait être une solution. Le choix a été fait de

s'intéresser d'abord à Windows qui étant le système le plus répandu a suscité le plus de développement de solutions de chiffrement. L'idée étant que l'expérience acquise dans le déploiement d'une solution Windows pourrait être transposée à d'autres systèmes.

Il a été privilégié une solution pouvant répondre aux différents besoins identifiés qui vont de la problématique de la fuite d'informations en cas de perte ou de vol du support à la protection des données partagées sur un réseau afin que seules les personnes autorisées puissent y accéder.

Le CNRS ayant mis en place une IGC, il a semblé assez logique d'utiliser des certificats comme mécanisme de gestion des secrets de chiffrement. L'idée étant que l'organisation mise en place pour l'IGC puisse être reprise pour gérer les recouvrements.

La solution retenue est le produit ZoneCentral de la société Prim'x.[5] Ce produit possède un certain nombre d'atouts :

- Une granularité assez fine permet de ne chiffrer que uniquement le contenu d'un répertoire aussi bien que l'ensemble des répertoires d'une partition.
- Possibilité d'utiliser des certificats pour stocker les clés de chiffrement.
- Contrôles d'accès fin par personne et par répertoire pour dire qui a le droit de déchiffrer le contenu d'un répertoire.
- Conteneur chiffré pour échanger des informations.

Premiers tests

Une première expérience a été conduite auprès de quelques volontaires dans la région de Toulouse. Le but était de procéder à une analyse approfondie des fonctionnalités du produit et surtout de réfléchir aux conditions de sa mise en œuvre dans un contexte réel.

Bien qu'il n'y ait pas eu de difficultés techniques de déploiement, le nombre de machines (une trentaine) mises en production lors de cette opération a été trop faible pour pouvoir tirer des conclusions quant à l'aspect déploiement. Une des raisons de ce faible engagement des utilisateurs tient au fait que de nombreux ordinateurs utilisent des systèmes d'exploitation autres que Windows (Mac OS, Linux).

De manière générale, les administrateurs ont signalé des réticences voire des refus à la mise en place d'un produit de chiffrement. Il y a plusieurs raisons à cela. Concrètement un produit de chiffrement n'apporte rien de tangible à l'utilisation quotidienne d'un ordinateur. Par contre les objections sont évidentes. Tout d'abord il y a la contrainte supplémentaire de fournir un mot de passe pour déverrouiller l'accès. Il y a la crainte du ralentissement de la machine lié au chiffrement même si l'expérience a montré que la perte de performance n'était pas réellement sensible. La peur, probablement plus fantasmée que réelle, de ne pouvoir récupérer ses données en cas d'oubli de son mot de passe ou à la suite de problèmes logiciels ou matériels est un frein à l'acceptation du chiffrement. Enfin les procédures de recouvrement qui accompagnent le chiffrement lèvent l'objection d'un « big

brother » qui pourrait tout connaître. Certes le bon sens voudrait que les informations non chiffrées soient encore plus facile d'accès à quiconque mais nous sommes dans un milieu à la culture volontiers libertaire où le fait qu'un tiers, l'agent de recouvrement, puisse avoir du pouvoir est mal vu.

Si le produit a globalement répondu aux objectifs fixés, son utilisation et sa mise en œuvre ne sont pas si simple que cela. La formation des utilisateurs et encore plus des administrateurs est un préalable important. La principale difficulté provient du paramétrage du produit qui offre de très, voire trop, nombreuses options. Bien que ce ne soit pas obligatoire l'utilisation des outils de déploiement Windows (Active Directory, GPO) facilitent grandement les choses. Or si les systèmes Windows sont relativement répandus au CNRS, il n'y a que peu d'endroits, principalement dans les services de l'administration mais rarement dans les laboratoires de recherche, où est mise en œuvre une administration centralisée des postes de travail sous Windows.

Le faible nombre de demandes alors que les besoins sont patents montrent qu'il faut sensibiliser les différents acteurs et en particulier les directeurs d'unité pour permettre d'avoir une réelle adhésion des utilisateurs au chiffrement. La politique de sécurité des systèmes d'information (PSSI[6]) du CNRS qui vient d'être adoptée devrait aider à améliorer ce point.

Mise en place d'une opération pilote

Le test sur quelques volontaires ne suffit pas à déterminer tous les écueils que peut rencontrer le déploiement à grande échelle d'une solution de chiffrement. Il a été donc prévu de conduire une opération pilote sur un échantillon réaliste (de quelques centaines à un millier de postes semblent un chiffre raisonnable) afin d'évaluer précisément :

- Les ressources financières et humaines liées à l'installation du produit.
- Les besoins de formation et d'assistance des administrateurs et des utilisateurs.
- L'acceptation de la solution par les utilisateurs
- La vérification en situation réelle des procédures de recouvrement.

Pour le choix de l'échantillon plusieurs possibilités s'offrent à nous. La première consiste à cibler l'administration (siège et délégations régionales). Il s'agit sûrement de la situation la plus favorable en matière de déploiement de ZoneCentral au CNRS. La population, les usages, les façons de travailler sont relativement homogènes. Le parc des machines est administré de façon centralisée et déployer un produit de chiffrement n'est qu'une tâche supplémentaire qui ne devrait pas amener à revoir l'organisation. Comme Windows est quasiment le seul système d'exploitation utilisé sur les postes de travail, le choix de la solution ZoneCentral (exclusivement Windows) ne pose aucune difficulté. Etant données ces particularités, il sera difficile d'extrapoler l'expérience à l'ensemble du CNRS qui présente des situations très variées.

L'autre optique consiste à s'intéresser aux laboratoires les plus sensibles en matière de confidentialités des données. A priori le personnel y est sensibilisé à la nécessité de protéger l'information ou du moins on peut l'espérer. Ces laboratoires sont prioritaires pour la mise en place de la PSSI (politique de sécurité des systèmes d'information) du CNRS et sa déclinaison au niveau local. Leur diversité les rend assez représentatifs de l'ensemble du CNRS. Ils sont donc tout naturellement désignés pour une opération pilote en matière de chiffrement. La principale objection provient du fait que la solution ne s'applique qu'aux postes de travail Windows et le parc comprend une part importante d'autres systèmes (Linux, MacOS). Bien que plus difficile à mettre en œuvre ce choix est celui qui nous apportera le plus d'informations pour un déploiement à l'ensemble du CNRS.

Cette opération pilote devrait être lancée prochainement.

Démarche pour un déploiement

Au-delà des opérations pilotes, il faut réfléchir à ce que pourrait être le déploiement du chiffrement qui rappelons le est une nécessité comme il est spécifié dans la PSSI du CNRS.

Etant donnée la diversité des situations au CNRS, la multiplicité des matériels et systèmes d'exploitation, les moyens limités (financier et humains) dont disposent certaines unités, les besoins de sécurité (le niveau de sensibilités des données est très variable), il n'est pas souhaitable ni simplement possible d'imposer une solution unique en matière de chiffrement.

L'utilisation de produits de chiffrement libres ou bien fournis en standard avec le système d'exploitation (EFS ou Bitlocker sous Windows, FileVault sous MacOS, dm-crypt sous Linux) peut s'avérer moins coûteuse en termes d'achat et de déploiement qu'un produit spécifique. Cependant il faut bien étudier la situation car par exemple le chiffrement du disque dur Bitlocker n'est disponible que sur les versions les plus coûteuses (intégrale ou entreprise) de Vista. La décision en matière d'achat informatiques se fait généralement au niveau local, celui du laboratoire. Il difficile d'imposer un produit payant, certes aux nombreux avantages, à des personnes qui n'en voit pas nécessairement l'intérêt alors qu'il existe des solutions gratuites et qui de plus sont, par nature, rebelles à toute décision imposée de l'extérieur.

Comme il n'est pas possible de tout faire tout de suite, il faut se définir des objectifs suffisamment modestes pour être réalistes.

- Chiffrer les communications qui transitent par Internet. Les solutions techniques existent et sont éprouvées : SSL/TLS, versions sécurisées des protocoles (https, imaps, pop3s, smtps, etc.), SSH, VPN (openVPN), … Généralement l'authentification se fait par certificat au moins pour le serveur, le CNRS possède une IGC, il n'y a aucune raison de tergiverser. De fait le chiffrement est déjà effectif en pas mal d'endroits, il faut encourager les retardataires (et peut être même un peu plus) mais nous avons l'expérience des précurseurs.

- Chiffrer les courriers contenant des informations confidentielles. Etant donné l'existence d'une IGC au CNRS, l'utilisation de S/MIME semble une bonne solution.
- Chiffrer les mobiles (ordinateurs portables) en commençant par ceux qui contiennent les informations les plus sensibles.
- Les supports amovibles (disques, clé USB, CD, DVD, etc.) qui sortent du périmètre doivent être chiffrés. On pourra utiliser les mêmes méthodes que pour les mobiles, surtout si ces supports sont montés et vus comme des disques. Dans le cas où le support amovible servirait uniquement à transférer des informations, l'utilisation d'un container chiffré dont l'usage est analogue à une archive ZIP paraît le plus simple.
- Chiffrer les fichiers et répertoires contenant des données sensibles. Là où l'environnement tant matériel qu'humain s'y prête et où le besoin s'en fait sentir, on mettra en œuvre sur les machines connectées sur le réseau interne un chiffrement de répertoire. Le produit utilisé pourra être ZoneCentral qui a été très favorablement évalué.
- La protection de données fortement sensibles nécessite une étude particulière. On a peu de garanties sur la robustesse des produits de chiffrement utilisés mais la principale faiblesse est sans doute celle du poste de travail très vulnérable à des attaques par des logiciels malveillants.

Conclusion

Faute de pouvoir imposer un produit de chiffrement unique au sein du CNRS, il faut s'attacher d'abord à l'objectif qui reste la sécurisation des données sensibles. Une démarche pragmatique conduit à laisser une certaine liberté de choix aux différents acteurs qui se décideront en fonction des contextes, des plateformes, de la sensibilité des données, tout en préconisant quelques solutions de chiffrements qui auront été évaluées et reconnues comme répondant aux besoins.

Il ne s'agit pas non plus de laisser tout un chacun se débrouiller dans son coin pour mettre en œuvre le chiffrement. Il faut mutualiser les efforts. Le travail en réseau, le partage des connaissances et des expériences est quelque chose qui fonctionne plutôt bien au CNRS. Aboutir à des préconisations sur le choix de solutions de chiffrement, des directives et des conseils sur la façon de les déployer, des documents d'accompagnement à la mise en œuvre et l'utilisation est un objectif tout à fait raisonnable.

L'expérience en matière d'organisation acquise lors du déploiement de l'IGC au CNRS peut certainement être mise à profit pour la mise en place des procédures de recouvrement ou de séquestre liées au chiffrement.

[1] http://www.urec.cnrs.fr/IMG/pdf/secu.articles.Article-Jres2005-112.pdf

[2] http://www.urec.cnrs.fr

[3] http://www.urec.cnrs.fr/rubrique116.html (rubrique chiffrement)

[4] http://www.seagate.com/www/en-us/products/laptops/momentus/momentus_5400_fde.2/

[5] http://www.primx.eu/

[6] http://www.sg.cnrs.fr/FSD/securite-systemes/documentations_pdf/securite_systemes/PSSI-V1.pdf

Présentation d'une application spatiale de la cryptographie : le système d'Observation de la Terre PLEIADES

Eric Frémeaux, Astrium (eric.fremeaux@astrium.eads.net)
Jean-Renaud Meyer, CNES (jean-renaud.meyer@cnes.fr)

Résumé

Le système spatial d'Observation de la Terre PLEIADES, développé sous maîtrise d'ouvrage CNES et maîtrise d'œuvre satellite Astrium, avec un lancement prévisionnel pour fin 2009, a pour vocation de fournir des images de haute résolution (0,7 mètre) pour des clients civils institutionnels et commerciaux (via la société Spotimage) comme pour des utilisateurs de la Défense française et d'autres pays européens associés au programme.

La sensibilité de ce système (confidentialité des images et des demandes de programmation, disponibilité du service) a induit la nécessité de mettre en œuvre des moyens cryptographiques afin de protéger les accès entre les satellites et leur segment sol (centre de contrôle et stations réceptrices des images), sur la voie montante (accès télécommande) comme sur la voie descendante (télémesure image).

Le CNES, en tant que maître d'ouvrage du système PLEIADES, a établi la définition et les spécifications de besoin en terme de sécurité de ces liaisons, en collaboration avec la Direction de Programme de la DGA, avec le laboratoire de cryptographie du CELAR, ainsi qu'avec la DCSSI et le maître d'œuvre satellite Astrium. Des analyses de risque ont été menées à bien sur les accès bord/sol, conduisant à proposer des schémas d'implémentation de mécanismes de sécurité (algorithmes, taille des marquants et des clés) et des scenarii de génération/distribution de clés adaptés aux besoins spécifiques du projet et de ses utilisateurs.

Il est présenté un aperçu de la définition des sous-système cryptographiques bord et sol, arrêtée et validée par le projet et toutes les instances associées, en tenant compte des différentes contraintes spécifiques au monde du spatial ; il en ressort la nécessité d'anticiper au plus tôt ces besoins de sécurité dans le planning de développement, afin d'obtenir efficacement l'agrément de chaque composant, et enfin l'homologation de l'ensemble du système PLEIADES.

Presentation of a cryptographic space application : The Earth Observation System PLEIADES

Abstract

The Earth Observation space system PLEIADES, developed by CNES as System Prime and Astrium as satellite Prime Contractor, with an estimated launch date by the end of 2009, has the mission to provide high resolution images (0,7 meter) to civilian customers (via the commercial company Spotimage) as to French Defence users and other European countries associated to the program.

The sensitivity of this system (confidentiality of the images and the programming requests, service availability) induced the need to implement cryptographic means in order to protect the accesses between the satellites and their ground segment (Control Center and images receiving stations), on the uplink (Telecommand access) as on the downlink (image telemetry).

CNES, as PLEIADES System Prime, established the definition and requirement specifications in term of security for these links, in cooperation with DGA Programs Directorate, with CELAR cryptographic laboratory, with the DCSSI and the satellite Prime Contractor Astrium. Risk analyses have been performed on the onboard/ground interfaces, to propose implementations of security mechanisms (algorithms, counters and keys size) and scenarios of keys generation / distribution, taking into consideration specific project and users needs.

The presentation will establish an outline of the agreed definition validated by the project and associated authorities, of ciphering/deciphering units on ground and onboard, describing the several specific constraints.

This highlights the need to anticipate as soon as possible sizing of security architecture, in the aim of obtaining the agreement to each cryptographic component, and finally the accreditation of the whole PLEIADES system.

1. PRESENTATION DU SYSTÈME PLÉIADES

Le satellite PLEIADES

Pléiades HR est la nouvelle génération de satellite haute résolution submétrique, développé par le CNES pour un système dual civil et Défense, en complément du système Cosmo-Skymed en coopération avec l'Italie, dans un objectif de coopération Défense au niveau européen.

Le système est composé de 2 satellites identiques, déphasés de 180° sur une orbite polaire héliosynchrone calée à 10h30 d'heure locale, et à 700 km d'altitude.

Le CNES (Centre National d'Etudes Spatiales) est le maître d'ouvrage du système Pléiades pour le compte du Ministère de la Recherche, Astrium est le maître d'oeuvre du satellite et du bus (servitudes : gestion bord, alimentation, contrôle d'attitude et d'orbite et du sous système de télémesure de charge utile incluant la compression et la gestion des prises de vue) ; enfin Thalès Alenia Space (Alcatel) est le maître d'oeuvre de l'instrument (télescope optique, détecteur).

Description du satellite Pléiades

Chaque satellite embarque un instrument de prise de vues à haute résolution. Cet instrument permet d'acquérir simultanément une image d'une largeur de fauchée de 20km, en bande Panchromatique avec une résolution de 0.7 mètre au nadir, et une image en bandes multispectrales (bandes bleue, verte, rouge et proche infrarouge) avec 2.8 m de résolution au nadir.

Le satellite est conçu autour de l'instrument de prise de vue. Les panneaux solaires sont fixés sur le corps du satellite sans mécanisme d'entraînement de façon à assurer une grande rigidité au satellite permettant ainsi une excellente agilité autour de ses axes.

Le système de contrôle d'attitude et d'orbite utilise 4 actionneurs gyroscopiques qui permettent d'obtenir des performances de dépointage de 10 degrés en 10 sec et de 60 degrés en 25 sec sur les axes roulis et tangage. Ces performances permettent d'obtenir des images stéréo le long de la trace et de mosaïquer des images.

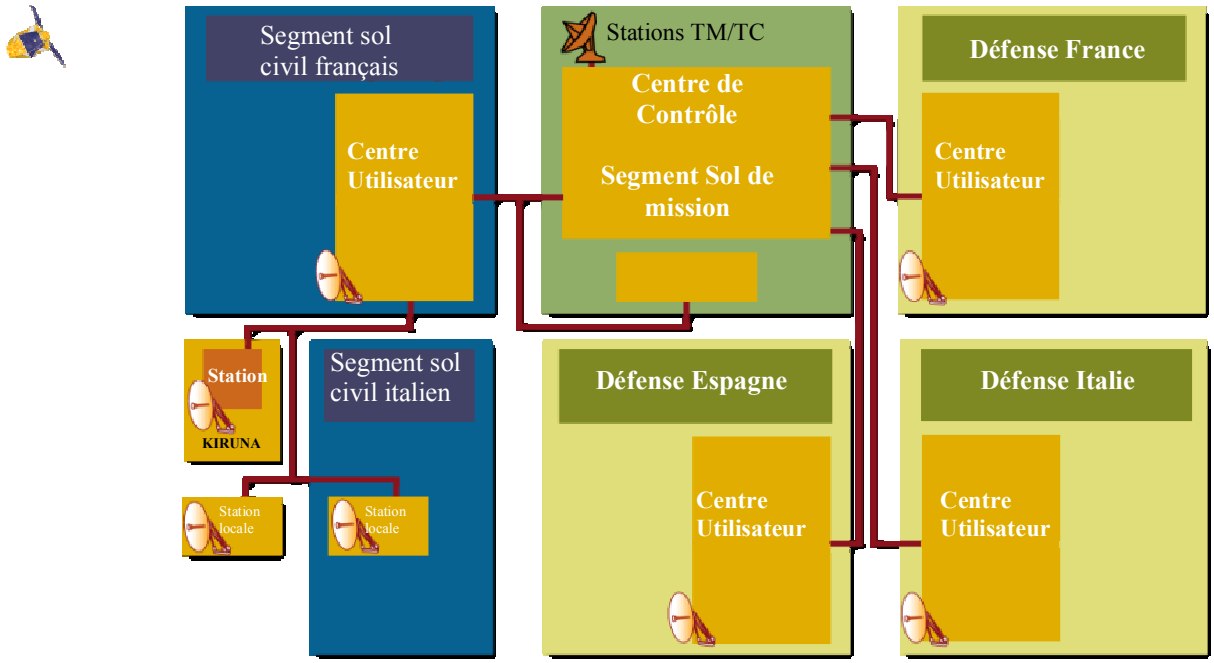
Le choix de l'orbite permet un temps de revisite de 2 jours pour un seul satellite ; lorsque le système sera complètement déployé, l'accessibilité quotidienne sera possible pour n'importe quel point du globe grâce à la combinaison de 2 satellites déphasés de 180° dans le plan d'orbite. Chaque satellite pourra produire plus de 250 images par jour.

Le rythme de bits de la télémesure image est de 450 Mbits/sec en utilisant 3 canaux de 150 Mbits/sec chacun. Cette télémesure est transmise au sol à travers une antenne fixe fournissant une couverture omni directionnelle avec une ouverture de 64°.

Le stockage des images à bord utilise une mémoire d'une capacité de 600 GBits en fin de vie.

Le segment sol Pléiades

Le système PHR est conçu comme un système dual, ce qui permettra de fournir des services aux utilisateurs civils et de défense qui auront chacun une capacité de réception, de traitement et de programmation.

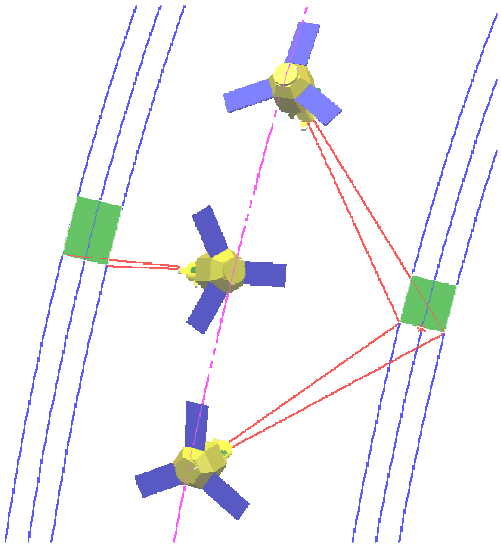


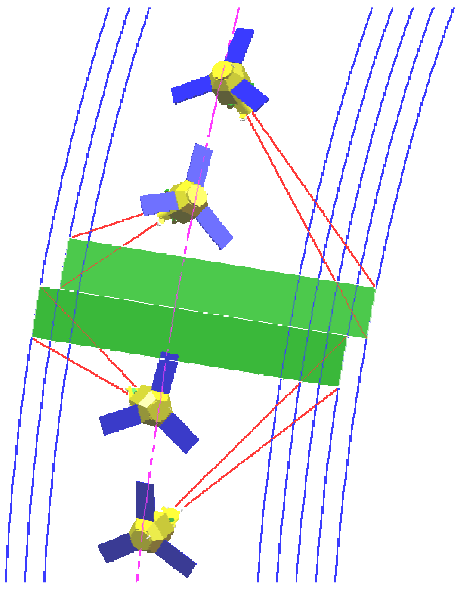
Le segment sol comprend les éléments suivants :

- le segment sol de contrôle et de commande (CCC) associé à un réseau de stations, incluant une unité responsable de la programmation duale des satellites, intégrant les demandes issues du Canal Défense (France et partenaires) et du Canal Civil (Spotimage) ;
- les segments sol utilisateurs civil et de Défense en charge de la gestion des requêtes mission, de la réception, de l'archivage et du traitement des données ;
- le centre de qualité image qui permet le suivi de la performance des instruments.

Modes d'acquisition

Une innovation majeure par rapport à la précédente filière d'observation de la Terre (SPOT) est l'agilité obtenue grâce à des actionneurs gyroscopiques (qui remplacent les roues à réactions, beaucoup moins dynamiques).





Mode multispot : accessibilité est-ouest, capacité stéréoscopique **Mode mosaïque :** augmentation de la fauchée globale

Cette agilité permet des dépointages rapides de l'axe de visée, depuis une position quasi géocentrique jusqu'à des débattements de 60° en 40s, favorisant la capture de scènes dans des délais rapides, ainsi que le mosaïquage (assemblage de scènes contiguës en un seul passage).

Exemple de Scénario d'enchaînement de scènes

Acquisitions multi-Spots + Bandes le long des côtes + Prises stéréo et tri-stéréo + Mosaïques

Produit Panchromatique (Haute résolution : 0.7 m)

Produit «vraies couleurs» 0.7 m

Produit «fausses couleurs» 0.7 m

2. ANALYSE DE RISQUE

Dans le secteur spatial, les autorités de conception des systèmes à satellite se sentent de plus en plus concernés par les vulnérabilités de leurs composantes bord et sol, qui peuvent être exploitées bien plus que dans les décennies antérieures. Des politiques de protection, déjà établies pour les systèmes spatiaux de Défense, le sont désormais pour les systèmes civils, dans la mesure où ils deviennent stratégiques comme Pléiades.

A titre d'exemple illustratif, le rapport du General Accounting Office (Commission d'Etudes du Sénat américain) d'août 2002, intitulé « Critical infrastructure protection : commercial satellites security should be more addressed » (GAO-52-781) souligne la nécessité de protéger davantage les infrastructures étatiques et y compris commerciales de réseau de communications par satellite, et ce à la lumière de profils d'attaques potentielles ou avérées comme sur les systèmes d'information.

Les services fédéraux (NASA, NOAA, FAA, Secret Services,…) s'appuyant de plus en plus sur de tels systèmes, il devenait impératif de définir et faire appliquer une stratégie de protection, reposant pour l'essentiel sur le chiffrement des communications entre les satellites et le sol, qu'il s'agisse du segment sol de contrôle ou des terminaux des clients utilisateurs.

Entre autres types d'agressions (tir laser, missiles guidés, brouillage), il se développe un potentiel de plus en plus élevé de risque d'attaque des liaisons bord/sol des systèmes à satellite par :

- écoute des données télécommande et télémesures (états du satellite, données clients),
- brouillage (jamming) en cours de passage TM/TC opéré,
- usurpation (spoofing, masquerade) de la liaison de télécommande ou télémesure par interception / modification / rejeu des données lors d'un transfert sur un réseau Internet, ou par émission depuis une station non agréée.

Les techniques pour contrer de telles attaques sont :

- chiffrement des liaisons (télécommande pour assurer en particulier l'authentification afin d'éviter une prise en main du satellite par un attaquant),
- usage d'un émetteur TC voie montante maîtrisé et suffisamment puissant pour éviter un déni de service par brouillage.

Les satellites CNES scientifiques et d'Observation de la Terre civils (Spot) utilisent jusqu'alors des systèmes simples de protections par authentification (algorithme CCSDS) pour la voie montante de télécommande, et de chiffrement (agrément gouvernemental) des données pour la protection commerciale des données image, sans politique forte de sécurité.

La robustesse est liée au besoin de protection des informations, les mécanismes de protection reposant sur la confidentialité, l'intégrité et l'authentification des données transmises et reçues.

Les autres catégories d'attaques s'adressent aux stations sol et au Centre de Contrôle, qu'il y a lieu de redonder et de protéger vis-à-vis des réseaux externes.

Le programme PLEIADES a mené il y a quelques années une étude de vulnérabilité reprenant, entre autres, la liste des menaces mentionnée ci-dessus, afin de déterminer le poids des risques et les contre-mesures à appliquer dans la mesure du possible. Ainsi, si pour les événements majeurs il est difficile d'apporter des contre-mesures (destruction d'un centre de contrôle ou d'une station par un séisme, destruction du satellite par une météorite, une éruption solaire majeure, une attaque balistique ou un tir laser), tout l'effort a été porté sur la protection des liaisons spatiales sol/bord/sol afin d'apporter des assurances de confidentialité, de contrôle d'intégrité, d'authentification, et de dispositif de contrôle antirejeu sur la liaison montante de télécommande comme sur la liaison descendante de la télémesure image.

Il est important pour les systèmes à satellite civils de disposer d'algorithme et d'équipements de chiffrement/déchiffrement agréés par le NSA de leur pays, même dans le cas d'un algorithme en version commerciale.

Appliquer le renforcement de la protection des liaisons bord/sol se fait au détriment de la fiabilité et disponibilité de ces liaisons spatiales, et en premier de l'opérabilité et de l'observabilité du satellite (le commande/contrôle dans le jargon du monde spatial). L'insertion de tels dispositifs Chiffre dans l'architecture bord doit se faire au regard de ces paramètres (fiabilité, redondance, cross-strapping, modes de pannes et conséquences, opérations de rétablissement…) sachant qu'il est pour ainsi dire impossible d'intervenir sur du matériel en orbite pour le réparer, du moins à de telles altitudes. L'environnement radiatif et de particules lourdes (ions, protons…) sont d'autres dangers à prendre en compte dans les modes de pannes de l'électronique embarquée.

Enfin, l'introduction d'équipements cryptographiques spécifiques, donc nouveaux, dans le planning de développement du satellite et de son segment sol associé, exige la prise en compte d'un processus d'évaluation de sécurité et d'agrément de chacun de ces équipements bord et sol, qui n'ont pas les mêmes contraintes de conception ni d'environnement, ce qui représente des développements complètement distincts, et par conséquent un investissement financier important.

On compte en général, dans le monde du spatial, 2 ans de développement pour obtenir un prototype d'équipement chiffre à soumettre à évaluation, avant de lancer en production les équipements de vol.

3. CONCEPT DE DUALITE ET EXIGENCES DE SECURITE

Le système PLEIADES a ceci d'innovant qu'il fédère les besoins de clients du monde civil (opérateur commercial Spotimage) et du monde de la Défense française et de la Défense de pays partenaires (Espagne, Italie…), et ce avec une composante bord (le satellite) commune, un centre de contrôle commun, et des centres utilisateurs spécifiques conçus toutefois sur une architecture et des briques communes (clé en main).

Le défi majeur a été de construire un tel système dual avec des garanties de confiance entre les clients partenaires (civil, Défense France, et Défenses étrangères), avec une stratégie d'homologation pilotée par l'Autorité d'Homologation PLEIADES nommée SAP-FR (Security Accreditation Panel France, présidé par le Haut Fonctionnaire de Défense et de Sécurité du Ministère de la Recherche auprès du CNES).

Le système HELIOS était déjà un exemple de partenariat entre Défenses de pays européens, mais il était fondamentalement dissocié du système Spot, bien que certains éléments (plateforme, centre de contrôle, stations de réception, centres utilisateurs) aient été développés sur une base commune.

Dans le cas de PLEIADES, le satellite étant dual, chaque utilisateur est séparé du périmètre de ses voisins par des systèmes de cloisonnement reposant sur le Chiffre, et ce à plusieurs niveaux :

- chiffre sur les produits image spécifiques à chaque client,
- chiffre sur la liaison de télécommande de la composante bord (le satellite),
- chiffre entre les centres utilisateurs et le centre de contrôle pour la protection des demandes.

Les systèmes de chiffre des différentes catégories de liaisons ont été développés pour être utilisés par les différents clients, pour une fonction donnée, sur la base d'équipements identiques, avec des algorithmes distincts développés par le CELAR pour les mondes civil et Défense. Enfin, le cloisonnement dans la répartition et la distribution des clés utilisateurs assure la confidentialité des liaisons de chacun des centres utilisateurs clients.

Nous ne parlerons ici que du chiffrement des liaisons sol/bord/sol entre le satellite et les stations sol (commande/contrôle et produits d'imagerie spatiale).

Contraintes et exigences générales de sécurité

En phases de lancement et d'utilisation, le satellite doit recevoir, mémoriser, traiter et émettre de l'information pouvant être classifiée aux niveaux suivants :

- les ordres de programmation transmis au Satellite via la TC bande S qui sont classifiées,
- les images transmises par la TMI bande X qui sont classifiées pour les utilisateurs Défense, et libres ou sensibles pour les clients civils, tout en devant assurer un cloisonnement pour chaque partenaire.

Il découle des exigences majeures ci-dessus les spécifications de besoin en terme de protection des liaisons sol/bord/sol qui suivent.

A partir de la phase de lancement, le satellite doit rester sous contrôle exclusif de la France. Cela implique que seuls des ordres et des informations authentiques et intègres émis par une autorité autorisée doivent être acceptés par le satellite.

Les exigences de sécurité relatives à la composante spatiale PLEIADES s'appliquent au satellite, ainsi qu'à ses interfaces sol/bord/sol incluant :

- les équipements de gestion de données bord (calculateur, chaîne image, chiffreur TMI et déchiffreur TC),
- les équipements de chiffrement/déchiffrement sol (centres de traitement des images et centre de contrôle).

Niveau et moyens de protection

Les dispositifs de protection ont été définis à partir des besoins de protection des données stockées à bord et transitant entre le bord et le sol, et par une analyse de risque conduisant à une expression de besoin de type SSRS (System Security Requirements Statement), qui a permis de définir les services de protection à implémenter pour répondre aux besoins PLEIADES. Cette SSRS a été soumise à l'Autorité d'Homologation PLEIADES SAP-FR.

Les liaisons TC de servitude et de données image TMI entre le satellite et le segment sol sont opérées en liaison RF entre le satellite et les stations sol, et au travers du réseau public entre les stations sol et le CCC. Ces 2 types de liaisons doivent être protégées contre tout type d'intrusion défini dans la SSRS.

Les informations transitant sur la voie Télémesure de servitude (voie descendante bande S) ne sont pas classifiées, par conséquent la liaison descendante TM de servitude n'est pas protégée (les informations seront transmises en clair sur cette liaison).

Le cloisonnement

Le cloisonnement entre les différents périmètres Chiffre TC et TMI (civil et multinationale Défense) sera assuré sur des moyens dédiés comme suit :

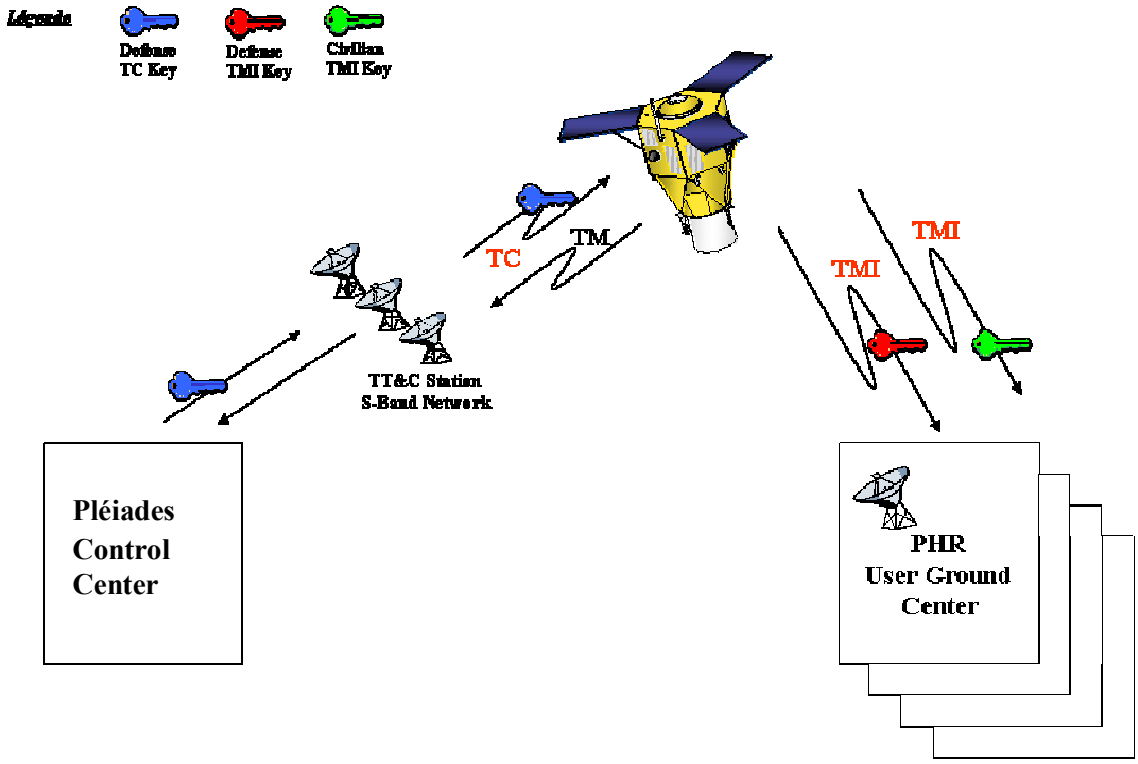
- la gestion des clés TC (de niveau gouvernemental français) sera assurée par la CGS (Cellule Gouvernementale de Surveillance de la DCSSI);
- la gestion des clés TMI (civil et multinationale Défense) sera assurée comme suit :
  - les clés civiles seront produites et remises par la CGS à l'opérateur civil Spotimage, sur un poste dédié ;
  - les clés Défense France et Défenses partenaires seront produites et remises par la CSC (Cellule multinationale de Surveillance des Clés de la DCSSI) aux différents ODC (Organisme de Distribution des Clés) des différents partenaires Défense.

Cependant, l'équipement bord de chiffrement TMI sera commun aux deux services civil et Défense, au titre de la dualité (une seule charge utile commune). De même, l'équipement de déchiffrement TC et le calculateur bord traitent en commun des plans de commande de clients civils et des Défense.

La CGS et la CSC seront en charge de mettre à la clé les équipements bord de déchiffrement TC et chiffrement TMI lors de la campagne de lancement.

4. PRESENTATION DE L'ARCHITECTURE DE SECURITE SOL/BORD/SOL

Les liaisons sol bord télécommande et bord/sol télémesure image de PLEIADES HR sont protégées par un chiffrement des données. La liaison descendante télémesure bande S ne véhicule que des données de servitude non sensibles. Elle n'est pas chiffrée.



Liaisons TC de servitude

Les plans de mission de prise de vue des différents clients civils et Défense sont regroupés dans un seul et même message de programmation au CCC, et transmis au satellite via la liaison montante avec une protection cryptographique unique et unifiée, gérée par une seule entité, l'opérateur CCC.

La liaison montante servitude (bande S) doit être protégée contre tout type d'intrusion défini dans la SSRS et devra donc assurer la confidentialité, le contrôle d'intégrité et d'antirejeu des données réceptionnées à bord.

La protection du canal TC s'effectue par un chiffrement avec calcul de signature des TC au sol (Chiffreur TC placé dans le centre de contrôle opérationnel sol), contrôlé en authentification et antirejeu à bord.

Les Télécommandes reçues par le bord sont déchiffrées dans l'équipement Déchiffreur TC, équipement en coupure, avant traitement par le logiciel de vol. Toute TC incorrecte est immédiatement rejetée.

Par ailleurs, il existe un service de TC de sécurité destinées exclusivement au déchiffreur, et déchiffrées / authentifiées au sein de son périmètre cryptographique, pour la gestion des clés (effacement et renouvellement par téléchargement).

Liaison TMI de données image

La liaison descendante de données image contient les informations images brutes générées à bord. Elle est descendue sur trois canaux différents en bande X (8GHz) vers les stations sol des différents opérateurs :

- la Défense France et les clients Défense étrangers,
- opérateur civil français (Spotimage),
- clients Opérateur Civil avec service local de réception.

Cette liaison doit être protégée contre tout type d'intrusion défini selon la SSRS, et devra donc assurer la confidentialité des données, ainsi que le contrôle d'intégrité et antirejeu des données.

La protection (intégrité et confidentialité) des canaux TMI s'effectue par un chiffrement des données à bord par un équipement en coupure (Chiffreur TMI) vers le segment sol (centres utilisateurs) disposant d'un déchiffreur TMI démarqué selon le client.

De même que pour le déchiffreur TC bord, le chiffreur TMI dispose de TC de sécurité pour la gestion des clés (effacement et renouvellement par téléchargement).

5. IMPLANTATION D'UNE FONCTION DE CHIFFREMENT SUR LE SATELLITE

Généralités

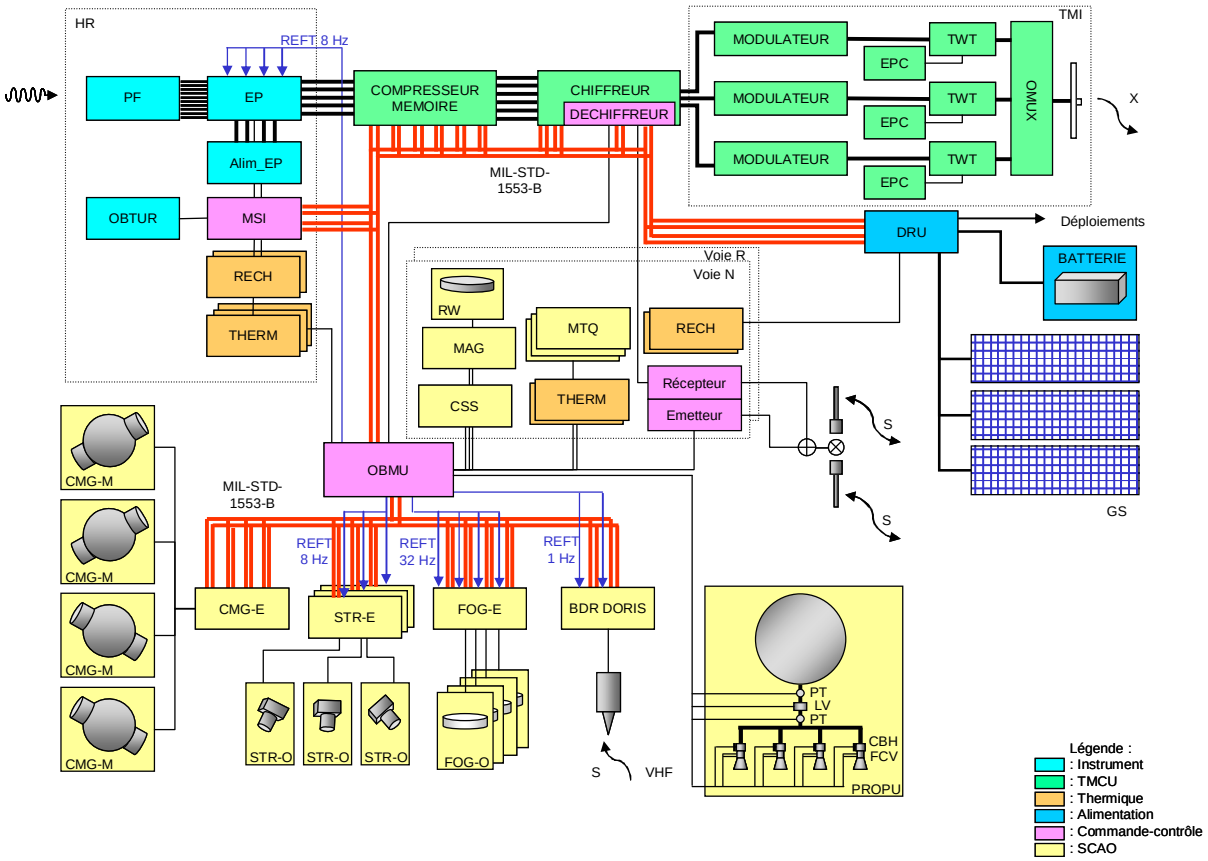
Au-delà de protéger l'accès du satellite contre des intrusions par une fonction d'authentification des commands émises du sol, la mission de défense implique directement des fonctions de confidentialité, d'intégrité, l'utilisation d'un algorithme gouvernemental et l'évaluation de tout ou partie du satellite par un organisme habilité.

Les données image et d'attitude du satellite sont classifiées.

Les analyses de risques sur les liaisons sol/bord/sol ont naturellement montré le besoin d'implanter une fonction chiffre en coupure sur les liaisons entre le satellite et le sol.

La figure ci-dessous montre l'architecture du satellite Pléiades.

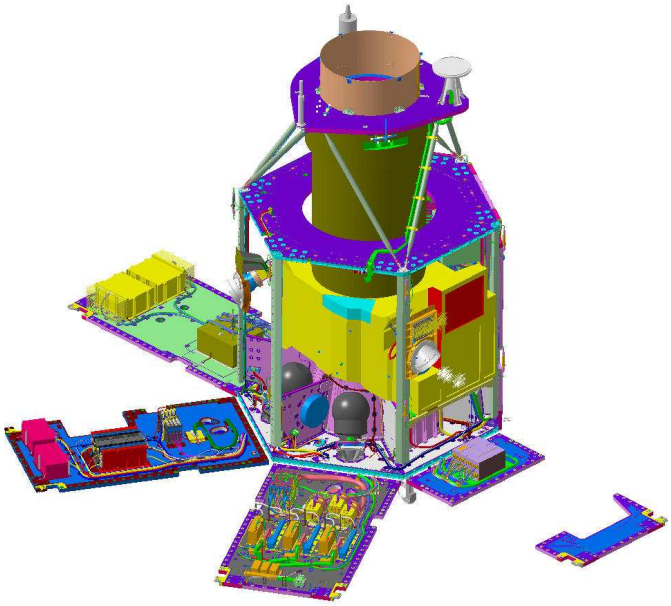
Architecture système électrique HR



On peut décomposer les fonctions du satellite Pléiades en plusieurs sous-systèmes :

- Alimentation, qui comprend une batterie, trois panneaux solaires permettant de maintenir la charge adéquate de la batterie et une électronique de régulation et de distribution des alimentations vers les autres sous systèmes.
- Gestion bord, qui intègre les équipements permettant le commande /contrôle du satellite (récepteur TC, émetteur de TM de servitude, déchiffreur TC, calculateur, antenne et autres éléments RF). A ce sous système il faut associer le :
- Logiciel de vol, implanté évidemment dans le calculateur, qui permet la gestion du satellite comme la mise en œuvre des plans de vol téléchargés à partir du sol, la détection des anomalies, la reconfiguration après anomalies et l'interface avec le sol. Le logiciel de vol comprend aussi les logiciels nécessaires aux autres sous systèmes du satellite (contrôle d'attitude, thermique,..).
- Contrôle d'attitude et d'orbite, qui contrôle l'attitude du satellite lors des prises de vue et permet le maintien du satellite sur son orbite (au travers du sol) grâce à un système de propulsion, des actionneurs gyroscopiques, des magnétocoupleurs et des senseurs stellaires (reconnaissance des étoiles), des gyromètres à fibres optiques, des senseurs solaires et des magnétomètres accompagnés de l'instrument de navigation DORIS.
- Contrôle thermique, qui distribue et régule la puissance de réchauffage dans le satellite pour maintenir les équipements dans leur plage de fonctionnement.
- L'instrument, responsable de la prise de vue et la transmission des données vers le sous-système : Télémesure Charge Utile (TMCU), qui assure la compression, le stockage, le chiffrement et la transmission vers le sol des données images issues de l'instrument.

La figure ci-dessous montre le satellite ouvert. En effet le satellite a été conçu à partir d'une structure hexagonale pouvant s'ouvrir en pétales facilitant les accès pour les phases d'intégration et d'essais. Une fois en configuration de vol (murs fermés), l'accès au satellite en particulier pour le chargement des clefs s'effectue au travers de prises de peau.



Les contraintes majeures du développement du satellite PLEIADES

La définition, le développement, le test et le lancement de deux satellites nécessitent la réutilisation d'équipements existants et une coopération internationale pour les équipements nouveaux dans le but de réduire les coûts de la maîtrise d'ouvrage par un financement issu d'autres pays (Calculateur en Suède, Structure, Amplificateur à tube pour la bande X et électronique de puissance en Belgique, Electronique des actuateurs gyroscopiques et Transpondeur Bande S en Espagne,…..).

L'architecture ainsi que les critères et contraintes de développement ont permis de réduire le périmètre à l'équipement chiffre.

La réutilisation d'infrastructures sol existantes dans plusieurs pays ainsi que des équipements de vol éprouvés conduit à maintenir des standards de communication existants et non compatibles dans certain cas avec une fonction de chiffrement gouvernemental telle que définie pour Pléiades.

Cette deuxième contrainte a conduit à créer un couple équipement chiffre sol / équipement chiffre bord qui modifie le standard entre les deux équipements.

Définition de la cible de sécurité

Les analyses de risques ont montré que les équipements et le logiciel de vol du satellite peuvent, en cas de dysfonctionnements « volontaires » ou non, créer un déni de service, et dans un cas particulier un risque de perte de niveau de confidentialité. Le deuxième risque a été pris en compte par l'ajout d'un mécanisme de protection géré par l'équipement de chiffrement et donc entièrement maîtrisé.

Sans parler de piégeage des équipements, le fait qu'une société, en dehors du territoire français, fournisse un équipement récurrent ou un équipement nouveau dans le cadre d'une coopération internationale rend difficile un contrôle étatique sur le développement d'un tel équipement.

Prenons en exemple le développement du calculateur. Le calculateur est livré avec son BIOS et l'ensemble des interfaces permettant de commander tous les organes du satellite. Il est facile d'imaginer des quantités de déni de service que cet équipement peut générer. Cependant cet équipement a été développé dans un cadre strict défini par des règles de qualité, de développement et de tests d'un programme spatial. Ces règles conduisent à une grande couverture de test et permettent de garantir in fine l'absence de dysfonctionnement hors panne. Le respect des ces règles est garantie par l'équipementier d'une part et par le suivi du maître d'œuvre et de la maîtrise d'ouvrage, le CNES.

On peut prendre un autre exemple, celui du logiciel de vol où une description exhaustive du processus qualité, de la gestion de configuration, de la définition des métriques utilisées et des tests effectuées tant au niveau du maître d'œuvre que du maître d'ouvrage a permis de démontrer que les risques de malversation du logiciel étaient négligeable.

Si on considère que l'ensemble des équipements du satellite peut contribuer à un déni de service, il parait illusoire de réaliser une évaluation sur l'ensemble des équipements du point de vue des moyens humains, des aspects calendaires et des coûts. Comme rapidement montré par les deux exemples ci-dessus, le niveau de qualité de développement des équipements spatiaux garantit de base une bonne confiance sur le fonctionnement. Il a donc été décidé de limiter les cibles de sécurité aux portes d'entrée et de sortie du satellite :

- La liaison télécommande en bande S,
- La liaison de télémesure de servitude en bande S,
- La liaison de télémesure image en bande X,

Pour des raisons de non confidentialité des données, la liaison de télémesure de servitude n'est pas chiffrée et n'est pas soumise à des contrôles cryptographiques. Cette liaison n'a pas été incluse dans le périmètre des cibles de sécurité.

Les deux autres liaisons ont été concentrées sur une seul et même équipement qui a fait l'objet de deux cibles de sécurité (voie montante et voie descendante) et donc d'une évaluation en respectant les critères communs. L'aspect chargement des clefs est couvert par les deux cibles.

Architecture de la liaison montante

La fonction de déchiffrement de la liaison TC est constituée de 2 déchiffreurs en redondance chaude, entièrement indépendants.

La liaison TC est une liaison RF en bande S. Le satellite comporte deux antennes de réception opposées, permettant une couverture de 4 π stéradians de l'espace. Le signal RF ainsi collecté est envoyé à deux transpondeurs bande S en redondance chaude au travers d'un coupleur 3 dB.

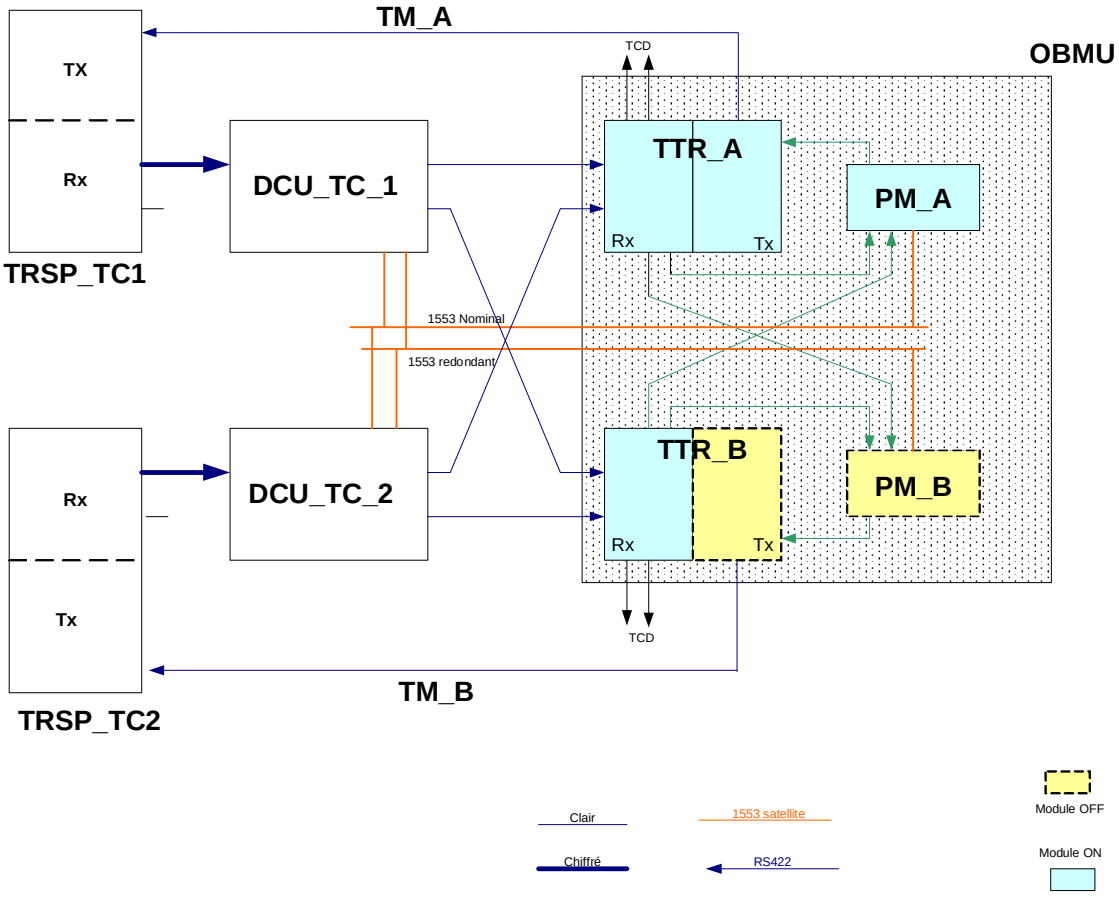
La sortie de chaque récepteur est connectée sur l'entrée de chaque déchiffreur, lui-même connecté aux deux calculateurs bord (OBMU A et OBMU B). Les déchiffreurs sont en coupure totale.

Chaque calculateur comprend une carte TTR responsable de la décommutation des TC, de l'exécution des TCD (TéléCommande Directe) et du formatage de la liaison TM de servitude pour envoi sur une liaison descendante en bande S.

Les TC décommutées sont, soit envoyées à une carte PM (Processor Module) pour être prises en compte par le logiciel, soit exécutées en local dans le cas des TCD (commandes de reconfiguration hardware). Il existe des interconnexions entre les deux cartes TTR (A ou B) et les deux cartes PM (N ou R). Ces interconnexions sont activées de façon à obtenir une configuration complète après une panne. Les deux cartes TTR (au moins la fonction TC) sont en redondance chaude. Les cartes PM sont en redondance froide.

Après décommutation des trames TC, la carte TTR est sélectionnée par identification de canal virtuel (VCID) de la trame TC.

La figure ci-dessous montre l'insertion du déchiffreur en coupure entre les transpondeurs et les OBMU.



Architecture de la liaison descendante

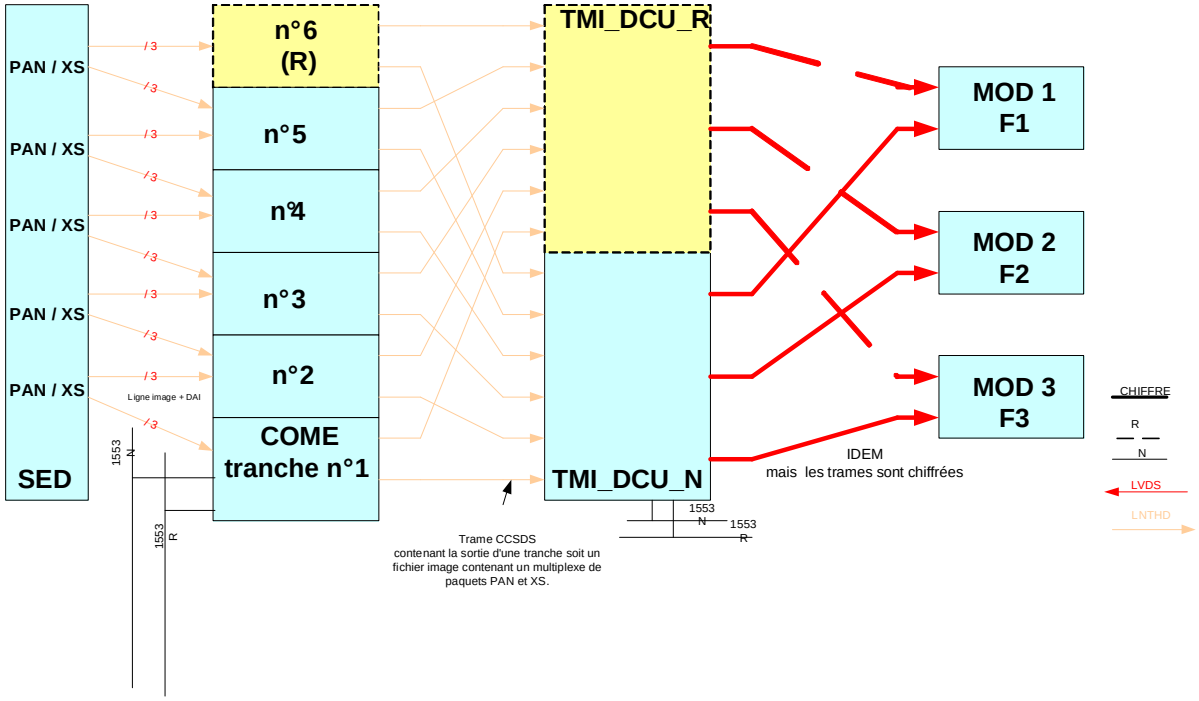
La fonction de chiffrement TMI se compose de deux chiffreurs en redondance froide, entièrement indépendants.

Les données images produites par l'instrument de prise de vue sont transmises dans l'espace mémoire de l'équipement COME par des liaisons LNTHD (Liaison numérique Très Haut Débit) pour y être stockées, avant leur transfert vers le sol.

Le COME (équipement Compresseur Mémoire) est composé de 6 tranches indépendantes réalisant la compression et la mémorisation des données d'un détecteur PAN et d'un détecteur XS. Comme il y a cinq détecteurs, cinq tranches sont actives simultanément, la sixième étant en redondance froide. Chaque tranche possède deux interfaces de lecture ; une pour chaque déchiffreur.

Un chiffreur TMI est composé de trois canaux de chiffrement, un par modulateur. Le chiffreur nominal est connecté sur les entrées nominales des trois modulateurs le chiffreur redondant sur les entrées redondantes. Chaque chiffreur est en coupure entre le COME et les modulateurs.

La figure ci-dessous montre le positionnement des deux chiffreurs en coupure sur la liaison descendante bande X.



Les cycles de vie du satellite PLEIADES

Les contraintes de sécurité sont supportées par un seul équipement sur le satellite PLEIADES, il est donc important de considérer les cycles de vie de ce satellite :

- La phase d'intégration et de test du satellite qui est réalisée avec des éléments secrets placebo non classifiées.
- La phase de chargement des éléments secrets de vol sur le site de lancement jusqu'au lancement lui-même.
- La phase opérationnelle en vol.

A l'ensemble de cycle de vie du satellite incluant l'équipement de chiffre, il faut rajouter le cycle de développement de l'équipement lui-même.

La contrainte principale concerne la confidentialité de l'équipement de chiffre. En ce qui concerne l'équipement, sa confidentialité repose sur les informations qu'il contient :

- Les clefs opérationnelles sont classifiées, alors que les clefs de tests ne le sont pas.
- L'algorithme de chiffrement fait l'objet d'une close de non-divulgation entre les différents partenaires.

Les conséquences principales sont les suivantes :

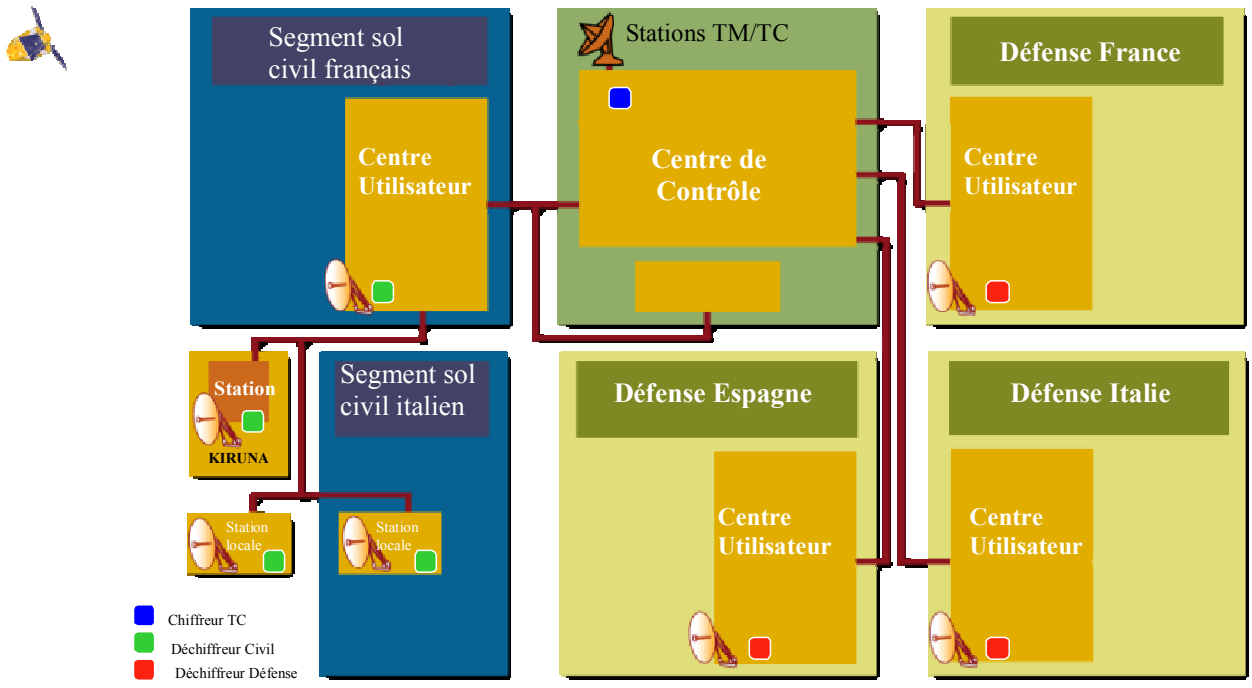
- Il est important de tester l'équipement de chiffre avec les éléments secrets, cependant une fois ces éléments chargés, l'équipement et donc le satellite (en raison de l'ACSSI qu'il contient) deviennent classifiés. Il a donc été validé que le rechargement d'éléments placebo, après effacement, permet de « déclassifier » le satellite une fois les essais effectués. Cette déclassification autorise le partage des moyens (salle blanche de classe de propreté 10000 par exemple) avec d'autres projets.
- Les contraintes de sécurité, de l'environnement spatial et les débits demandés sur les canaux TMI requièrent l'utilisation d'un ASIC pour le cœur cryptographique. Pour cela un ensemble de mesures qui satisfont les critères de sûreté au niveau gouvernemental a été mis en place pour garantir l'intégrité de l'ASIC.
- Les équipements chiffre de vol doivent être évalués par un organisme certifié, au même titre que les équipements sol, ce qui peut poser des problèmes dans le planning de développement très contraignants pour un satellite, en cas de faits relevés lors de l'évaluation (retard au lancement, reprise tardive de tests au niveau satellite…)

- Les clefs opérationnelles seront chargées au dernier moment que l'on peut situer avant le remplissage des ergols du satellite. Après ce moment, des mesures organisationnelles seront mises en œuvre pour surveiller le satellite.
- La phase de chargement de clefs sur le pas de tir fait l'objet de mesures organisationnelles, de connaissances de mesures TEMPEST et aura lieu dans des locaux sécurisés au centre spatial guyanais à KOUROU.
- Pendant la phase opérationnelle, le satellite est en orbite et ne craint pas grand-chose hormis des collisions avec des objets divers. Cependant nous avons fourni un ensemble de mesures permettant une analyse sur les signaux compromettants telle qu'une surmodulation des porteuses des émetteurs par leurs alimentations qui pourraient contenir un résidu d'activité sur des données confidentielles non cryptées.
- La phase opérationnelle se caractérise par un principe simple : « il n'est pas possible de réparer matériellement le satellite ». En effet, le satellite PLEIADES n'est pas accessible même avec la navette par exemple. Les équipements sont donc redondés pour contrer une panne simple. Le chiffreur TMI est en redondance « froide ». Il faut donc une action extérieure pour détecter la panne et décider d'utiliser le deuxième équipement de chiffre. L'équipement de déchiffrement de la liaison montante TC est lui en redondance chaude qui permet, lors d'une panne de l'un des deux équipements de déchiffrement TC, la prise en compte des télécommandes par le satellite. La redondance chaude correspond à une mise on simultanée de deux voies TC de l'équipement de déchiffrement. Ces deux voies sont indépendantes et par construction il ne peut exister de propagation de panne d'une voie sur l'autre. Le corollaire à cette gestion en redondance chaude est qu'il est impossible de couper un équipement même en panne (sauf en cas de court circuit après un certain nombre de réarmement d'un circuit de protection). Les cas de panne interne, n'induisant pas une perte franche de la fonction, ont donc été étudiés et analysés dans le cadre de la vulnérabilité du satellite.

6. EQUIPEMENTS CHIFFREURS ET DECHIFFREURS SOL

Les équipements de chiffre sol assurent, au même titre que les équipements bord, les services de confidentialité, intégrité, authentification et antirejeu, autant pour la liaison de télécommande que de télémesure image.

Leurs spécifications ont été établies à partir des besoins projets de sécurité des données satellite, et consolidés avec les SSRS des entités sol destinataires. Des cibles de sécurité ont été par ailleurs élaborées pour la définition du programme de tests à l'évaluation.



Le chiffreur TC est exclusivement installé au Centre de Contrôle (SDGC), alors que les déchiffreurs TMI sont installés dans chaque centre de réception image utilisateur, qu'il soit Défense ou opérateur civil

(Spotimage ou local). A noter que les déchiffreurs civils et Défense ont fait l'objet d'un démarquage d'algorithme et de design, afin d'éviter la rétroingéniérie.

Aucun TC ne peut sortir du CCC sans avoir été préalablement chiffrée, ce qui assure la confidentialité des plans de commande. Le système de signature basé sur les données et le compteur sol permet d'assurer la résistance aux attaques intrusives non authentifiées ou rejouées (par enregistrement et relecture de trames TC précédemment envoyées).

A noter que pour la réception des données TMI, le contrôle antirejeu n'induira pas de blocage, il servira à remonter des anomalies à des fins d'investigation).

Ces équipements passeront une évaluation cryptographique et TEMPEST en vu de leur agrément d'utilisation dans le système PLEIADES.

7. GESTION DES CLES BORD ET SOL

Le besoin du projet Pléiades est de disposer de jeux de clés permettant le chiffrement des liaisons TC et TMI entre le satellite et le segment sol, les clés ayant une nécessité d'usage par plusieurs clients et entités (clients imagerie, centre de contrôle, Cellule Gouvernementale), avec nécessité de renouvellement périodique.

Ces clés seront distribuées aux entités suivantes :

- les équipements bord (déchiffreur TC et chiffreur TMI),
- les Chiffreurs TC localisés au Centre de Contrôle (CCC),
- les déchiffreurs TMI Défense (France et partenaires),
- les déchiffreurs TMI civils (Spotimage et stations de réception locales).

Toutes ces clés (hormis les clés TMI civiles) doivent être protégées en fonction de la sensibilité (gestion ACSSI pour la Défense France), ces clés devant effectuer les opérations de chiffrement/déchiffrement d'informations protégées.

L'expression des besoins de protection des liaisons sol/bord/sol PLEIADES a abouti à l'établissement de spécifications d'architecture de gestion des clés destinées à l'exploitation du système PLEIADES, définissant :

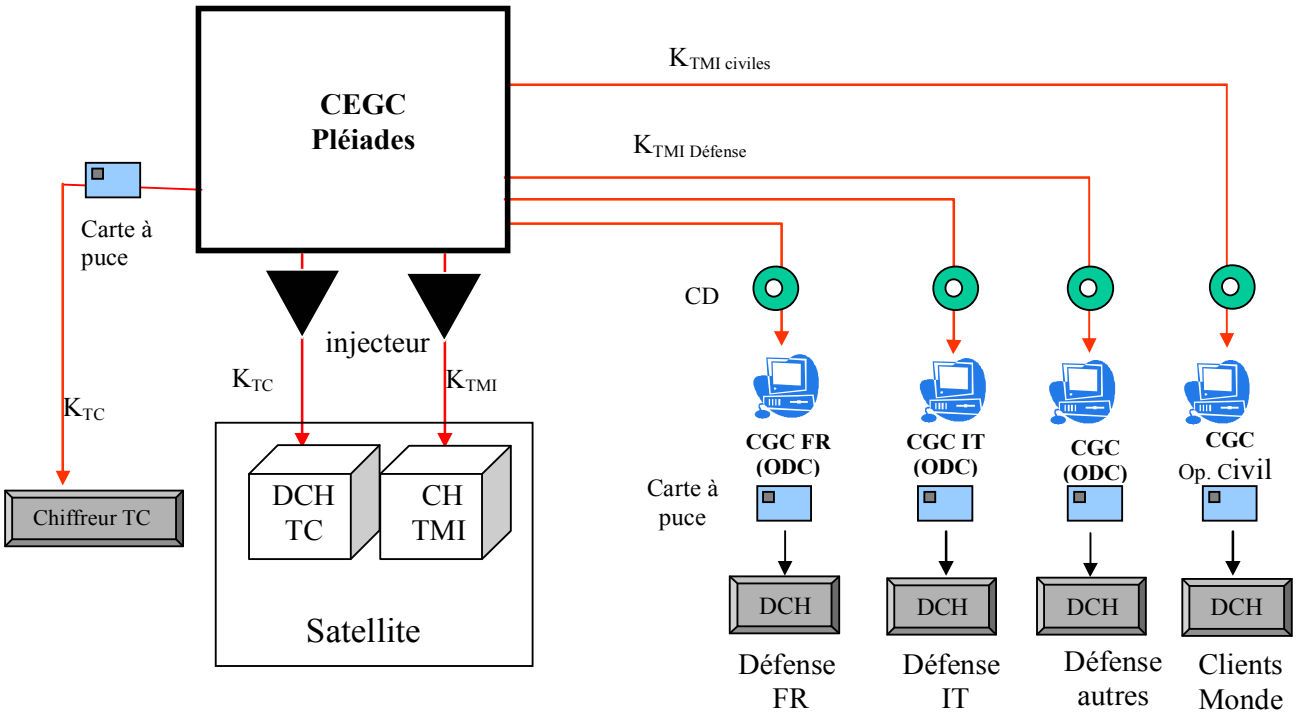
- le Centre d'Elaboration et de Gestion des Clés (CEGC) sous responsabilité de la CGS (Cellule Gouvernementale de Surveillance de la DCSSI) en charge de la création et de la distribution de l'ensemble des clés TC et TMI PLEIADES,
- des Centres de Gestion de Clés (CGC) destinés aux ODC (Organisme de Distribution de Clés) des différents clients TMI.

Le CEGC est en charge de :

- générer les clés de vol TC et TMI du satellite afin de mettre les équipements chiffre bord à la clé lors de la campagne de lancement, au moyen d'un dispositif d'injection de clés agréé,
- générer les supports physiques (cartes à puce) pour mettre les équipements sol chiffreur TC ;
- délivre des partitions de clés sous forme de CD-ROM vers les utilisateurs Défense France et étrangers qui disposeront chacun, au sein d'un ODC (Organisme de Distribution de Clés), d'une station nommée CGC; ces ODC seront chargés de recevoir les lots de clés destinés aux utilisateurs Défense par pays, de les stocker et de les redistribuer au fur et à mesure des besoins des utilisateurs finaux (stations fixes ou de théâtre) vers les déchiffreurs TMI Défense au moyen de cartes à puces.
- délivre un jeu de clés sous forme de CD-ROM vers le CGC civil de l'opérateur civil qui disposera d'un CGC, pour les stocker et les redistribuer au fur et à mesure de ses besoins vers les déchiffreurs TMI civils des utilisateurs finaux au moyen de cartes à puces.

Il existera par ailleurs des clés de gestion à usage spécifique (invalidation de clés, commandes de sécurité) sous contrôle des cellules en charge de la gestion centralisée du Chiffre Pléiades (CGS et CSC).

La figure ci-dessous illustre le principe général de génération et de transfert de ces clés depuis les CEGC des clés TC et des clés TMI.



Afin d'assurer le cloisonnement entre les 3 services, le CEGC a été segmenté par domaine de responsabilité (gouvernemental, civil, Défense).

Les CGC (Centre de Gestion de Clés) seront en charge de distribuer et gérer les clés de chaque entité Défense locale ou civile.

Les CEGC TC et TMI seront en charge de mettre les clés de vol dans des injecteurs de clés ségrégués pour un chargement dans les équipements bord de déchiffrement TC et chiffrement TMI lors de la campagne de lancement.

8. SYNTHESE

Les contraintes de sécurité ont été pleinement prises en compte dans les spécifications de besoin du système PLEIADES, et déclinées sur les composantes bord et sol, en s'appuyant particulièrement sur la définition détaillées des protections à mettre en œuvre sur les liaisons sol/bord/sol.

Cet exercice, basé notamment sur des analyses de risque menées suffisamment en amont dans le planning de développement PLEIADES, a permis d'aboutir assez tôt à la définition précise des besoins en terme de sécurité, pour l'élaboration des équipements chiffre bord et sol.

Par ailleurs, cette anticipation a permis de mettre en place un processus d'évaluation, planifié avec la DCSSI et le CELAR, de chacun des équipements chiffre dans son contexte d'application clairement défini, et ce afin de se donner la meilleure assurance possible de la prononciation de l'homologation du système par le Ministère de la recherche. Les travaux d'élaboration de la documentation (SSRS, cibles de sécurité) ainsi que les travaux d'évaluation se déroulent dans un bon esprit grâce à l'anticipation des besoins, et grâce au financement à sa juste valeur de ces travaux auprès des industriels, avec une maîtrise appuyée du processus par les responsables CNES du projet PLEIADES, avec le soutien du Maître d'Oeuvre Satellite Astrium.

Identity and Authorization in multi-organisation contexts

Peter Sylvester

EdelWeb, France

peter.sylvester@edelweb.fr

Abstract

Identity and authorisation control are deeply related together with impacts on their respective management. The topic gets complicated in contexts where authorisation has to be managed across boundaries of organisations. As an example, this applies in particular to public administrations or to semi-public organisations where agents/employees from one organisation want to
access resources of another organisation. Using a concrete use case, the topic and a solution will be explained and compared with other techniques. We describe an approach for cooperation that does not involve the installation of common infrastructures for end-to-end authentication and authorisation (including the management). Instead, it uses a delegated approach using organisation level proxies and gateway services that map local authentication and authorisation decisions from a client organisation to the local authentication and authorisation environment of the server organisation.
The work is proposed and currently developed as an element of the general referential of interoperability and security for the French Administration.

Les contrôles d'identités et d'autorisations sont liés entre eux avec des impacts sur la gestion respective, en particulier dans des contextes multi-organismes où des droits doivent être géres pour des entités d'autres organismes. C'est par exemple le cas pour la communication entre différents adminstrations et services publiques ou semi-publiques quand un agent d'un service doit accéder à des informations gérées par une autre adminstration.
Nous allons décrire une approche qui ne nécessite pas une harmonisation des infrastructures spécifiques de chaque organisme pour obtenir une solution de bout-en-bout, mais une solution par délégation et d'utilisation de passerelles qui transforment le résultats du controle d'identité et d'autorisation en visa utilisé par l'organisme fournisseur pout établir un contexte de sécurité selon les besoins de l'organisme qui fournit le service.
Le specification font partie d'une activitité de standardisation pour les Référentiel Général d'Interopérabilité et Securité (RGI/RGS) de la DGME.

Catégorie: spécialisée
Keywords: RGI, RGS, SAML, Identity Management, Privilege Management,, Liberty Alliance, Single Sign On, Authentication, Authorisation

Introduction

The article describes the findings and results of work that has been done by EdelWeb for the organisations of the French Social Security together with the Direction Securité Sociale (DSS) of the French Ministry of Health and the

Caisse Nationale d'Assurance Viellesse des Travailleurs salariés (CNAVTS).

The objectives of the work were to elaborate a standardised architecture[1] to allow client server applications of information systems of different organisations interoperate in conformance to different requirements. Furthermore, a study concerning the feasibility of a smart card based solution for all agents of the social security area.

The project associated all important actors of the social security area, and, in order to address a larger audience the work has been done in coordination with the French Direction Générale de la Modernisation de l'Etat (DGME) because the work is related to a more global activity of the DGME concerning general reference specifications for IT systems of the French administration to ensure interoperability [2] and security. On the European level, there is the European Interoperability Framework EIF [3].

Contexts and backgrounds

The ever growing penetration of information systems into practically all areas of the society has not yet created a radical change in the area of paperless activities known as de-materialisation. The difficulties are less technical but related to aspects like resistance to organisational changes, management of proofs, and also, a little bit of a paradox, the need to re-materialise. I

The problem of identities and privileges is a much larger than the scope of the described work. Therefore we present here some general remarks.

Management and control of identities and privileges are linked together but they have different characteristics, which prohibits the use of one single solution. For example, in the case of a smart card using for agents containing X.509 certificates, they should contain minimal information stable in time which should not depend on functions that an agent is authorised at a some moment.

Concerning privileges, in complex environments it is not appropriate for each application to manage privileges directly related with identities, rather it is useful to use an intermediate level of privilege or role attributes which are managed and accessed using a secure directory or a privilege service.

X.509 attribute certificates are in theory a good approach to transport privileges through non-secure environment in a similar way as X.509 certificates do this for identities. Unfortunately, there are not too many applications that understand them. Therefore, most applications use some sort of privilege directory.

In a context of multi-organisations, there is a need of federation, for example: At a global level, this can be simply the mutual recognition of certification authorities; at an individual level, persons can federate their identities using technologies like Liberty Alliance.

Concerning the authority that controls and manages the federation of identities and authorisations, we can distinguish at least three different contexts, one of them being the topic of this article. We give a short overview of the others in order to establish a clear scope.

One well known environment is where a single organisation tries to centralise the definition of identities and their management under a single sign on (SSO) approach. IT solution providers have been addressing this

since long time.

Today we can see here a large attempt of standardisation in order to simplify implementation and to permit interoperability of independently developed components; this is manifested in the OASIS SAML activities. [4] The SAML technologies permits a clear separation of applications and security infrastructures dealing with control of identities. Nevertheless, there are important limits, in particular, when organisations are very large or very heterogeneous with different security requirements for sub-populations. Furthermore, when the organisation structures change to more local responsibility, this approach can become difficult to implement.

We note here that an SSO implementation in an organisation can help implementing the architecture we propose in our standard since the external applications are always accessible through local gateways, i.e., representable as local applications.

A different scenario is addressed by the Liberty Alliance [5]. Here, an individual has many different identities. but there is no governing organisation that can create a global identity, this is not even desirable. It is the individual that wants to share information between different organisations. Again, the SAML frameworks and products seems to work. Besides the commercial sector, there are also examples in the public sector. In France, a particular application is the context of e-administration relation with citizens. The Liberty Alliance approach is used in order to permit a citizen to use a one-stop approach in various administrations' activities, e.g., when changing a postal address. In France, for historical reasons, administrations (and any other organisation) cannot easily share information about citizens since there is not single shareable identifier. Thus, to permit a citizen to use a one-stop service, a citizen can federates several identities of different administrations, only authenticate to one, and can easily propagate the changes to other administrations [6].

The third scenario is the one that we treat in this article: On one side organisations permit qualified persons to access some information of another organisation, and, on the other side, organisations propose access to some information to qualified entities of another organisation.

This problem occurs in private and public contexts, in particular it applies to cooperation among different part of public administrations or of para-public organisations. In particular, in Europe with its number of highly independent national structures. The need for such a communication exist as well for the public sector (A2A) as well as for B2B or A2B contexts, anywhere, where employees act on behalf of the organisation or enterprise, i.e., where the employer is responsible for the acts of the persons.

Requirements, constraints, consequences

In this sub-chapter we list some requirement and constraints and mention approaches which are difficult to implement, and thus excluded.

What needs to be developed is a solution allowing qualified persons determined by one organisation to access to an application of another organisation through the use of a local application, in other words, to implement client/server applications between consenting but independent organisations.

There is a need for many-to-many relationships with obvious scalability

problems.

Organisations have different local and incompatible infrastructures concerning not only the operational information system but also concerning authentication and access and authorisation control. In the past, ad hoc approaches using PKI together with simple access control directories and SSO approaches have been partially implemented to address needs for a more unified security infrastructure. These approaches are not been extremely successful in the context of multiple organisations for various reasons.

The proposed PKIs for end-to-end authentication are rather cost intensive and not easy to deploy, and do not solve the problem of authorisation management. Such solutions also require a very homogeneous use of safe cryptographic algorithms. The difficulties for a global migration in case of weakness are unknown.

In several cases, a solution involving end-to-end authentication is not even possible. The reason may be organisational, e.g. confidentiality of the requesting entities, but also technical, e.g. when client and server institutions don't share a common set of algorithms. Although this is likely not the case inside the French administration, it happens if systems of the Russian Federation participate. One can imagine a control at a Russian Airport to verify a health insurance of a person.

The impact of the new communication to the existing security infrastructure of each organisation must be as small as possible, i.e., to the largest possible degree, the existing techniques for authentications, roles and access management should be used, and, the required changes should not have an impact to existing usages, and the number of local modifications of the authorisation schemes should not depend of the number of producer organisations that need to be accessed.

Each organisation is responsible for the attribution of rights and duties to its employees or agents. Thus, authentication and authorisation management need to be controllable by the client organisation.

The distance (physical, organisational, legal) between organisations is often very large and makes management performed by another organisation (even a neutral third party) very difficult and expensive, in particular treatment of changes, and sometimes any delegation is simply impossible. In other words, the overall management of entities and rights must be done in a decentralized, and as a consequence, be done inside the client organisation or under its control..

Naming, roles, rights, attributes or application profiles are specific in each organisation and not necessarily compatible. For example, in one organisation, there may be a hierarchy of global rights, in another and another access rights are based on geographical distribution. In addition, the size of the organisation is an important factor. A global authorisation scheme that tries to combine all the differences is complex and unrealistic to implement.

Another aspect is that some organisations do not want to or must not reveal the identities of employees or agents, this does not mean that anonymous accesses are required, but rather there is need for depersonalisation, i.e., the only the client organisation can (and must be able to) determine from some opaque identifier who has performed a particular transaction.

As a consequence, approaches that would require first some kind of global harmonisation or unification of the identity management is most likely doomed to fail (not only) in our context.

When a member of a client organisation accesses to another organisation's information system, the client organisation takes the responsibility for proper authorisation and they are accountable for misuse. Organisations must respect many constraints regarding the information to be shared, often the concrete possibilities to share information need to be formally agreed and documented, in particular since the information are related to physical persons. The solution must permit the establishment of a service contract, allow easy implementation through a defined set of technologies permitting each partner to remain « master at home » and to assume his responsibilities in a controllable way.

A sufficient level of a security must be guaranteed using proper local and strong authentication and authorisation and by a priori trust combined with the possibility of a posteriori control through traces. Both organisations specify how traces of transactions are performed. It is important that both organisations handle this independently in order to allow each organisation to implement its own mechanisms for analyse which permits proper confrontation in case of problems.

Functional decomposition

The core idea of a transactions are that the consumer organisation manages its users and access rights and for each request it provides an assertion or attestation which is propagated to the producing organisation together with the request.

The intended interactions between organisations can be characterized as follows:

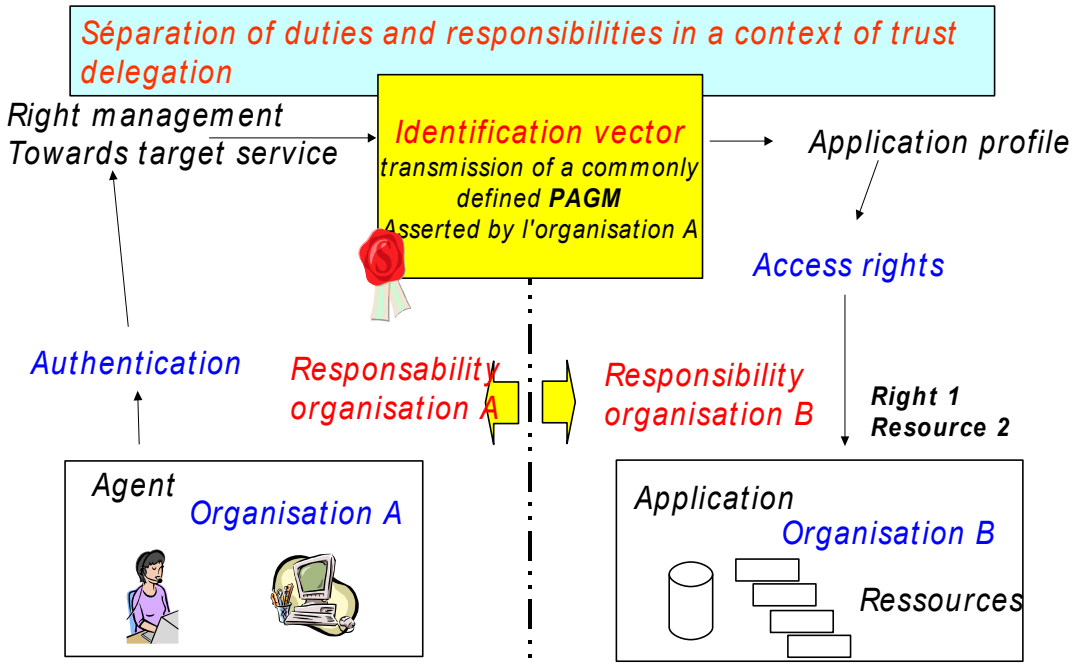


Illustration 1: Global data flow

There is a producer – consumer relationship between the information

systems of the organisations, the data produced and handled by the information system of one organisation are of interest to another organisation and is accessed and treated by some application in that consuming organisation.

This motivates the first type of an interaction protocol which is based on web services. But for certain (less critical) applications the producer is a simple web interface or portal accessed by a standard browser. The organisations establish an explicit trust through a contract that fixed the rights and duties of each organisation. Depending on the degree of sensibility the contract description may include precise definitions about how the participating perform the identity management. In other words, and to compare it with some better techniques, at least, the contract resembles the engagement of a PKI policy statement, but in this case, it is an agreement. The equivalence of a practice statement may be almost anything between some minimal provisions concerning the communication between the organisation, e.g., addresses of services and certificates, or totally contractual describing even the details of attribution of authorisations.

The overall architecture has three levels:

- The first level consists of all components of preparation (including legal aspects) and a semi-automatic parameter provisioning.,
- the second level is the operational system that treat the transaction
- The third level are all supporting services including the local authentication and authorisation infrastructures, a secure journal infrastructure, dedicated networks and PKIs allow a secure network infrastructure among the organisations.

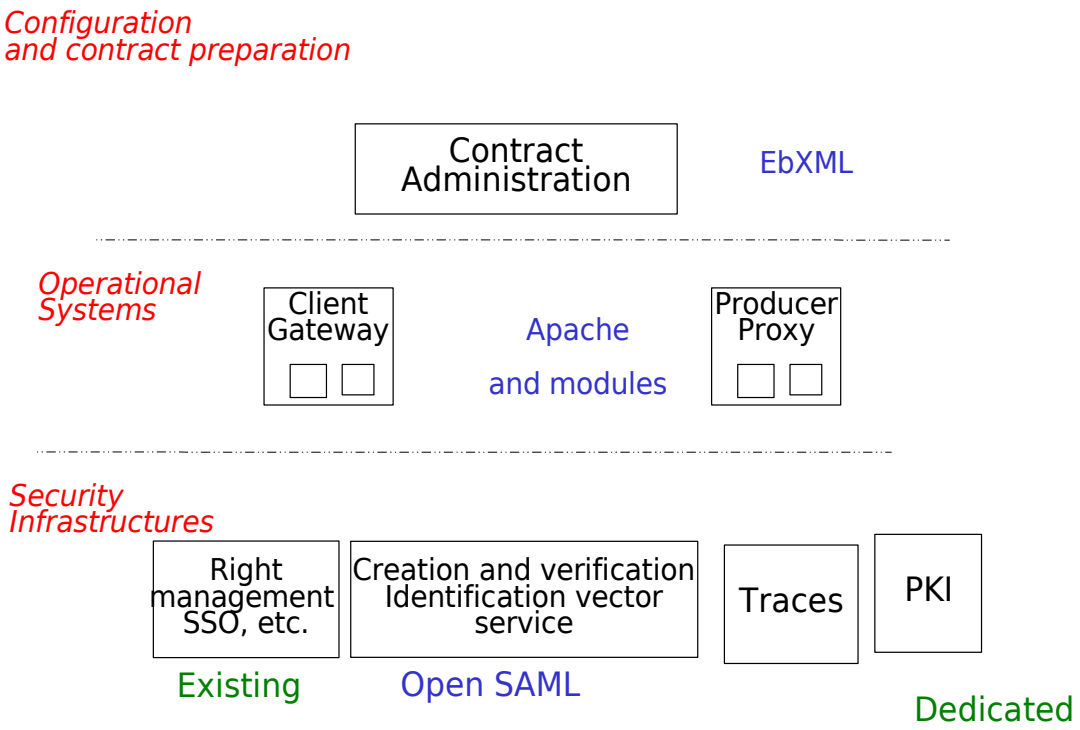


Illustration 2: Functional layers and modules

The configuration level

The cooperation of each pair of organisation is governed by a formal contract which is also the expression of the mutual trust between the organisations. This contract not only set the legal rules, but also has a formal technical part which can be used without manual interpretation to parametrize the systems. In this way all non-local configuration parameters are fully described in the collaboration agreement, and can therefore be processed automatically. This technical annex is therefore a machine readable description of the provided and desired services. We have proposed to evaluate the possibilities the ebXML [7] collaboration establishment procedure and formats, i.e. the establishment of collaboration protocol profiles and agreements to provide all technical parameters of a collaboration. In particular, the usage of an existing standard or proposition allows to use tools for the establishment of the contract and, to some extent, the glue code to parametrize the systems. Although the complexity of the approach may be larger than required, there is the interest of interoperability and adoption.

The operational level

Two usages scenarios have been selected:

- Access via web browser to a portal service
- Web Services

The producer scenario where the service is provided via a web portal accessible to several client organisations has the following characteristics: The user (or its browser) does not directly access the service. Instead, an outgoing proxy is used that establishes or controls the required authorisations and forwards this together with the request to the target institution.

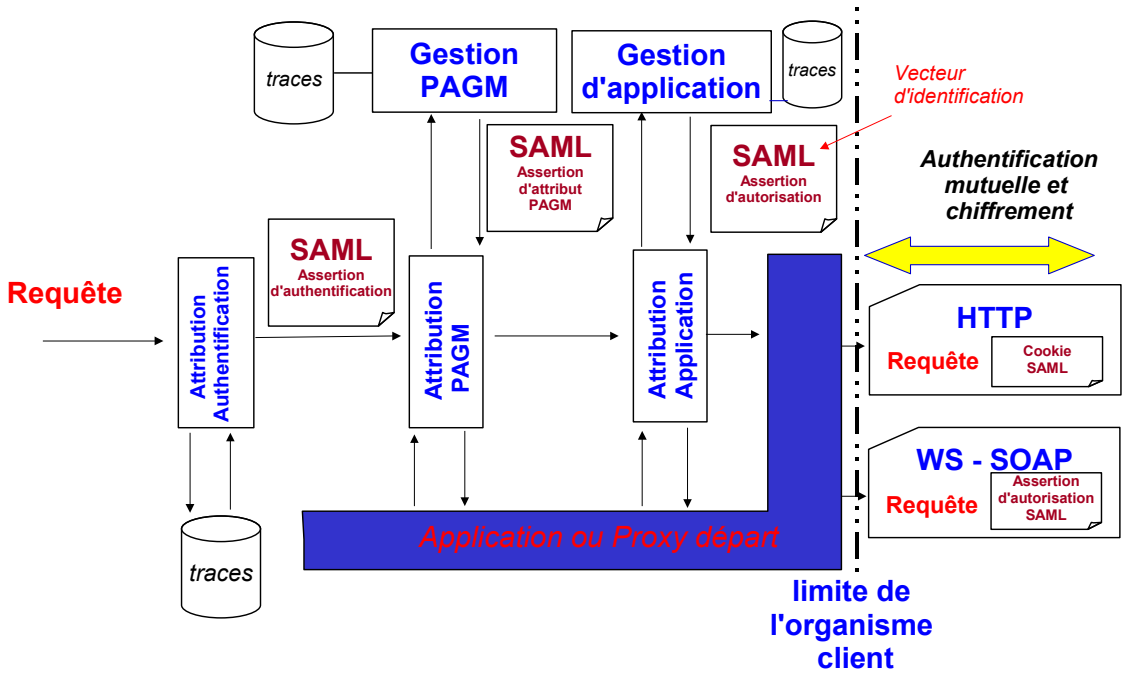


Illustration 3: Consumer architecture details

There is an obvious performance problem when a web page contains many

visual elements like icons an images. Since, conceptually, each request, i.e., a URL requires an independent and specific authorisation decision. It must therefore be possible to allow the proxy to bypass the strict authorisation rules. Also, a wild card logic for authorisation decisions are necessary to implement.

The outgoing proxy is a completely generic solution for each client organisation independent of any producer service and without any specific local application logic. It can therefore be regarded as a complete part of the client organisations infrastructure.

The producer scenario for web services is very different since there is always a local application that accesses a producer application via a web service. The client application architecture and related security components must ensure proper authorisations for the users. It is highly desirable for this context to have a multi-tiers architecture, where the (potentially insecure) application determines authentication and authorisation via specialized services similar as with some SSO implementations, and, in addition, have a generic web service proxy which controls whether the outgoing requests are accompanied by correct authorisation statements

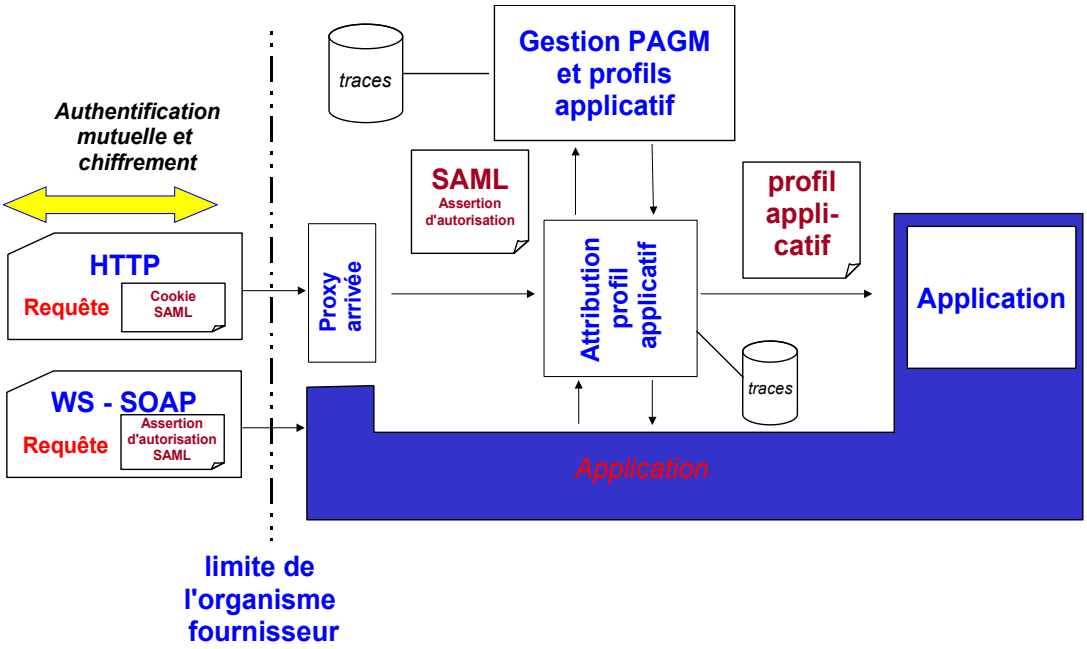


Illustration 4: Producer architecture details

For this environment, the is also the question about how asynchronous responses in work flows are to be returned. The underlying problem is that some requests cannot be treated immediately, they may for example need manual interaction in the work flow of the provider institution. The two following approaches have been discussed:

- Symmetric communication: For each client/server web service, there is a corresponding service in the opposite direction for which the roles of client and server are exchanged. This approach was not recommended for various reasons. One important concerns operational issues, since the client organisation must always be ready to receive answers.
- Asymmetric communication: In order to receive answers a web service similar to a mail box is provided. The client accesses in

regular intervals or upon explicit request. This solution is much simpler for a client organisation and for the definition of he communication agreement. E;g. it allows to use the same credentials for the services, connections are established in the same direction. Furthermore, the server has no responsibility to deliver the result. This approach was recommended.

In both cases a reverse proxy that implements the journal and verifies the authorisation decision and establishes a security context required for the producer environment, i.e. The authorisation statements are mapped to appropriate capabilities of a security context in the producer environment.

The transactions can be this summarised as follows: The client accesses to a service via a local mirror or gateway (in REST terminology). The client is authenticated using whatever means are used in his organisation. The mirror service (aka gateway or proxy) takes local authentication information to establishes an SAML authentication statement. Next, an attribute assertion is created that asserts a generic role to calling entity, and finally, an authorisation statement permitting the client user to access the target application or service. The assertion and the user requests are transmitted via a gateway to the target server application. On the server side, the authorisation statement is transformed into whatever the target organisation needs to access the service.

For the infrastructure level we have several components.

- The solution interfaces to existing authentication and identity management infrastructure.
- In both scenarios the generic outgoing proxy aka local mirror or gateway, and the incoming (reverse) proxy are responsible for journalising the requests. Although, or, in fact, because the communication is based on mutual trust, the design of the communication standard includes the usage of a secure journal system in order to allow a posteriori control and/or auditing. The design of technical details of the implementation such a system are obviously out of scope, but such a system must meet some obvious service requirements implied by the contracts between the organisation. Technically it is expected that a standardized protocol for secure archiving will be used to interface with the trace system.
- A generic module for the creation and verification of the identification vector. SAML assertion formats are proposed, and thus, the solution can either use existing open source tool kit, or, in the case or local SSO systems be partially based on assertions of the SSO infrastructure (in case that SAML is used there).
- A small dedicated PKI to secure the communication between organisations (not the users).

Since the organisations only talk through proxies, there is no need for an end to end network communication. This not simplifies the network infrastructure between the organisations, but also the security infrastructure between the organisations. Since the services involve access to sensible information concerning persons, organisations are required to protect the communication (at least in France).

It has been decided that the gateways and proxies talk to each other using TLS which client and server authentication in order to have full control over

the communication channel at the gateway/proxy level. The alternative of using IPSEC is more difficult to use since the applications (the proxies) may not be easily aware about whether the connection is protected or not.

Roles, profiles and "PAGM"s

As already indicated, the authorisation schemes used in the participating environments are very different. Even in a simple example where all organisation use an RBAC approach, it is difficult to make a global solution, since the roles are defined independently in each organisation, and, in general there is no simple way to federate them, i.e. , to obtain a common set of roles. One has to solve the n*m problem, i.e. the mapping of roles of n client organisations to application profiles of m producers, i.e. It is desirable to have less then n*m different attributes. The proposed approach is to study carefully the role definitions and application profiles, and try to determine a common set of attributes whose definitions are influenced by both existing roles and application profiles, but also, and this is helpful, by legal requirements for qualifications of the actors, i.e. the employees or agents. It is under the responsibility to assign this qualification profiles for which we have invented the abbreviation PAGM (profil générique applicatif métier) in order not to reuse or misuse any other terminology). The client organisation has to map local roles or qualifications to such a PAGM, or, in an extreme case, assign PAGMs directly to agents, and it is the task of the producer to map this to application profiles. Very often, the consumer organisation simply assigns one simulated user representing the combination of PAGM and client organisation and give the appropriate rights to that user.

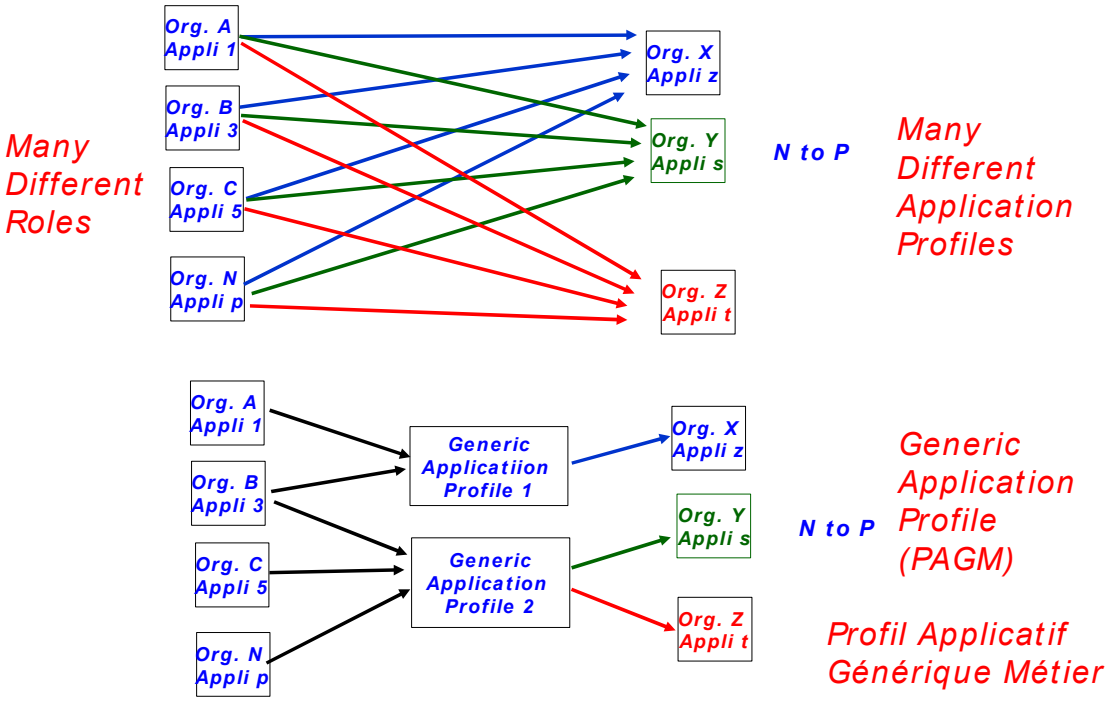


Illustration 5: PAGM architecture

The problem may be more complicated in the case when roles and rights depend on the status of the information, we have not treated this in depth. In some cases it is necessary that agent at the producing site has to validate a request, before it can be treated.

The identification vector

As already indicated, the authorisation decision made by the client organisation is transported to the producer organisation. This statement, called identification vector which is mapped to SAML assertions. The vector contains in an abstract way the following information:

- The client organisation's identification
- An identification of the client
- A validity period
- The producer organisation's identification
- The producer service (a prefix of an URL)
- A list of PAGMs that the client owns for the desired service
- A place holder for optional other attributes
- An indication about the strength of the authentication.

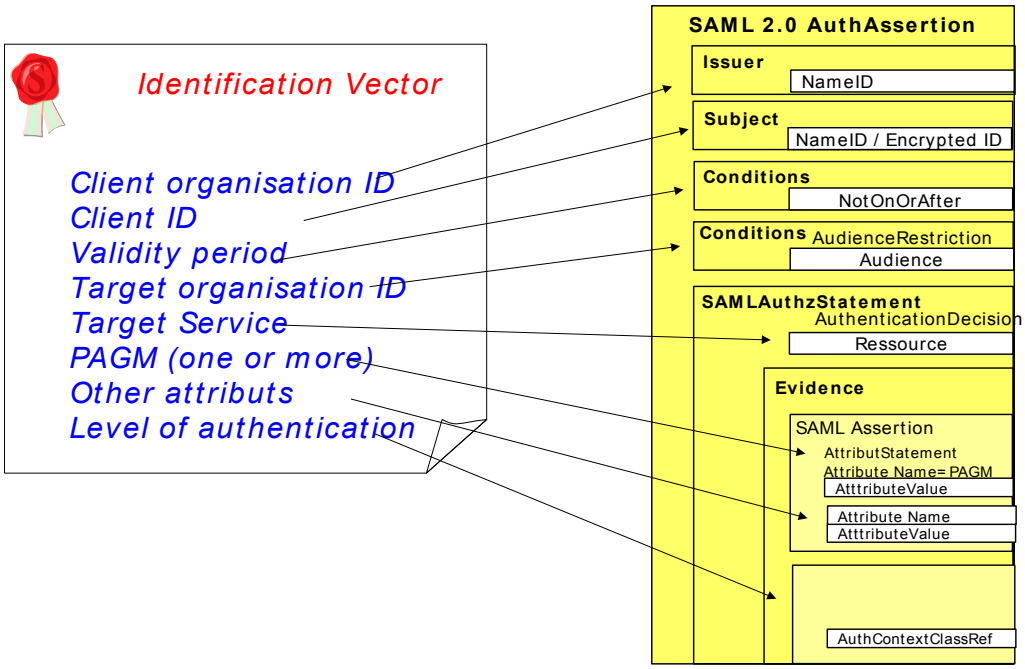


Illustration 6: Identification vector

These information are represented using three embedded SAML assertions, an authorisation statement for the service embedding an attribute statement for the PAGM embedding an authentication statement.

Additional remarks

The approach has been specified in detail for the group of all French Public Social Security Organisations, including the transformations necessary in the various local security infrastructures, and published to a wider audience for comments by the French Direction General pour la Modernisation de l'Etat.

Remarks and comments have been addressed in the specifications. The experimental implementations will concern real information, including access to information about 80 million persons treated by the French Social Security and Retirement services, in order to get feedback from real users. At the time of writing this article, several client organisations that use different local security infrastructures have been fully specified and are being test against the two producer scenarios. They and are expected to be operational end of 2007.

The study and the development of the standard was conducted in two steps by EdelWeb. During spring 2005, general specifications and principles were developed. During that phase, some alternative approaches and possibilities have been studies, like for example the usage of smart cards with single purpose or multi-purposes. Also, as an alternative to SAML, proxy certificates had been discussed, this approach was rejected mainly for psychological reasons.

In winter 2005/2006 detailed specifications as well as for the generic parts as for the specific parts for some initial applications and customer environments, and development and integration guidelines have been produced. The participating organisations are currently developing an experimental solution for real applications. The experiments cover first the outgoing gateway and the mapping to the local authentication environments and the incoming proxy and the mapping to application contexts. In a second step, the collaboration agreement management tools, i.e. the feasibility of the ebXML collaboration process, and the traces system will be studied.

The specifications use or reuse to a very large degree existing techniques and standards, it is therefore expected that that a large part of the necessary implementation will either be available using existing building blocks, in particular concerning the gateway to proxy communication and the transport of SAML assertions. For a web service context, this is essentially one of the approaches used by the Liberty Alliance project. For the direct web interface the logic is slightly different. Some SSO solution providers provide interfaces which are be easily used in the mirror gateway or in the reverse proxy.

A similar problem has been described in [8] concerning the verification of electronically signed documents that are exchanged between the Russian Federation and Poland. This is very close to what happens to signed web services. The proposed solution involves notaries that verify electronic signatures and then create formal attestations concerning the validity of the signed document, in the concrete case RFC 3029 [9] is used.

For the Web based environment, the identification vector is carried as an HTTP header. Shortly after the definition of the standard, an Internet draft was published to carry SAML assertions in an TLS extension. This approach seems to be a good separation of payload and security information. Since the authors are try to patent this text, and IETF community is reluctant to accept such activity as a standard. We note here that certain appliances of web servers which perform SSL as a proxy are transforming client certificates into an HTTP header.

Contrary to technologies proposed in the RGI [2] to ensure interoperability for the French administration which are mostly bottom-up building blocks, our proposal is completely complementary and defined as a top-down

application using some of the proposed lower layers of the RGI.

From a psychological standpoint it seemed useful to use to have all participants take fair parts in the development and the implementation of the standard, i.e., have both, clients and providers share the work.

References

[1] Direction de la Sécurité Sociale. Standard d'interopérabilité inter-organisme, V1.0, july 13, 2005.

http://www.edelweb.fr/iops/Standard_Interoperabilite-V1.0.pdf

[2] DGME, Référentiel Général d'Interopérabilité version 0.90, april 13, 2006

[3] European Communities, IDABC.. European Interoperability Framework, version 1, 2004,

http://ec.europa.eu/idabc/servlets/Doc?id=19528

[4] OASIS Security Services (SAML) TC

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security

[5] The liberty Alliance Project. www.projectliberty.org/

[6] Service-Public. http://www.service-public.gouv.fr/

[7] Ałła Stoliarowa Myć, Unizeto Technologies SA, Poland, The service of cross-border verification of electronic signatures for document exchange between Russia and European Union Countries

http://www.efpe.pl/upload_module/downloads/efpe/2006/prezentacje/3/07_DVCS-TopCross-Unizeto-EFPE.pdf, june 9,2006

[8] OASIS ebXML. http://www.ebxml.org/

[9] Carlisle Adams, et al., Data Validation and Certification Server Protocols, IETF RFC 3029, February 2001

Sécurité des systèmes utilisant la cryptographie quantique ; le projet européen Secoqc.

Philippe Painchault
Thales Communications
160 Boulevard de Valmy 92700 Colombes
Philippe.Painchault@fr.thalesgroup.com

Résumé

La cryptographie quantique a fait naître la promesse d'une très forte sécurité mais les limitations qui sont apparues ont entraîné de gros doutes sur l'apport en sécurité qu'un tel système pourrait apporter. Il s'agit ici de faire un point précis sur la sécurité des communications quantiques en fonction du système et des menaces envisagés. En particulier le projet européen Secoqc sera résumé afin de situer ce qu'un système quantique pourrait offrir actuellement ou dans un proche avenir.

Abstract

With quantum cryptography, the hope of an absolute security seemed reachable but several limitations have seriously compromised this vision. To have a clear image of this security, systems and threats are carefully studied in order to extract the possible benefits and drawbacks of a quantum solution. The European project Secoqc is summarized to show what a current quantum system could offer now or in a few years.

1. Introduction

L'évaluation de la sécurité des systèmes de cryptographie quantique est un sujet qui fait l'objet de positions souvent tranchés entre les tenants de systèmes quantiques dits inconditionnellement sûrs et les détracteurs reprochant à ceux-ci une complexité élevée pour un apport en sécurité marginal voire nul. L'objet de ce papier est de détailler précisément les systèmes quantiques envisageables, et les principales menaces à prendre en compte, pour obtenir dans chaque cas les bénéfices et/ou inconvénients de chaque système. En particulier, le projet européen Secoqc, qui a permis un développement important de la cryptographie quantique, sera exposé, afin d'en déterminer les enjeux.

Dans un premier temps sont rappelés les principaux niveaux ou preuves de sécurité qui peuvent exister en cryptographique classique. Ensuite, le principe de la cryptographie quantique est exposé. Cet exposé se veut le plus simple possible, notamment sans détailler les différents moyens de réalisations physiques.

Il est alors possible d'aborder l'apport potentiel de la cryptographie quantique. Cet apport est décrit en séparant les différents cas d'emploi. En effet, différents systèmes peuvent être construits à

partir des briques quantiques, et il faut étudier la résistance de ces systèmes vis-à-vis de chacune des menaces identifiables, depuis la compromission d'une clé d'authentification passée jusqu'à la cryptanalyse d'un algorithme de chiffrement.

Enfin, certains points marquants du projet Secoqc sont décrits, ce qui permettra de cerner son positionnement par rapport aux systèmes envisagés précédemment, et ce qu'il peut apporter concrètement.

2. Sécurité en cryptographie classique

2.1. Rappel sur la cryptographie

2.1.1. Fonctionnalités

Les principales fonctionnalités requises par un système de cryptographie sont celles de confidentialité et d'authentification : un système cryptographique doit assurer que des données ne puissent être lues que par des utilisateurs autorisés et / ou que ces données proviennent bien d'une certaine origine, sans avoir pu être modifiées.

A partir de ces fonctionnalités de base, d'autres services ont pu être développés. On peut citer par exemple le partage de secret, qui permet de diviser un secret en plusieurs parts de telle sorte qu'il faille réunir plusieurs parts pour obtenir une information utilisable, ou la preuve sans divulgation, permettant de prouver la connaissance d'un secret sans avoir à dévoiler aucune information secrète. De plus, afin d'assurer ces fonctionnalités principales, différentes fonctionnalités secondaires ont été développées : fonctions de compression cryptographiques, génération de nombres réellement aléatoires, ou protocoles de distribution de clés.

Différents outils ont ainsi été développés, dont la combinaison permet des applications variées, comme les transactions bancaires (carte bleue, transactions Internet), l'accès contrôlé à des services (pay-TV), la gestion des copyrights, ou le vote électronique, en plus des services traditionnels de protection des communications (communications militaires, grandes administrations, groupes bancaires ou autres, GSM).

2.1.2. Outils

Pour répondre à ces besoins, différents outils cryptographiques existent. Les principaux outils peuvent être classés en deux familles : la cryptographie symétrique et le cryptographie asymétrique.

La cryptographie symétrique suppose un secret commun partagé entre tous les utilisateurs désirant écrire ou lire un certain message. Elle repose sur des fonctions de chiffrement construites spécifiquement. Un exemple de telle fonction est la fonction de chiffrement par bloc AES.

La cryptographie asymétrique fait jouer un rôle différent aux utilisateurs. Elle repose sur une bi-clé comportant une partie secrète, appelée aussi clé secrète, et une partie publique, appelée aussi clé publique. Deux fonctionnements sont possibles, en chiffrement et en signature. En chiffrement, tout utilisateur peut chiffrer un message, à l'aide de la clé publique, et seul le destinataire possédant l'information clé secrète peut déchiffrer ce message. En signature, seul le signataire peut réaliser la signature d'un message, à l'aide de la clé secrète, mais tout utilisateur à l'aide de la clé publique peut vérifier que la signature provient bien de la personne possédant la clé secrète.

Toutes les fonctions de cryptographie asymétrique effectives reposent sur deux problèmes mathématiques fondamentaux : la difficulté de factoriser un nombre et la difficulté de trouver le logarithme d'un nombre dans des corps finis.

D'autres outils complètent ces fonctions de chiffrement ou de signature, comme les fonctions de hachage cryptographiques ou les générateurs d'aléa vrai.

Pour un système opérationnel, ces différents outils sont souvent combinés. Typiquement, un système peut utiliser une primitive asymétrique afin de construire pour chaque communication entre deux utilisateurs un secret commun qui sera ensuite utilisé pour protéger la communication avec une primitive symétrique. Un tel système combine ainsi la souplesse offerte par les techniques asymétriques avec la vitesse de traitement offerte par les techniques symétriques.

2.2. Sécurité des fonctions cryptographiques

La principale propriété désirée pour un système de chiffrement est qu'un attaquant disposant du message échangé (le chiffré) ne puisse acquérir aucune information significative sur le message réel (le clair).

La propriété que l'on souhaite pour un système d'authentification est que seul le ou les utilisateurs désignés peuvent réaliser le code permettant d'authentifier un message.

2.2.1. Sécurité inconditionnelle

Certains résultats de sécurité inconditionnelle existent mais ils sont malheureusement très limités.

Le premier résultat est celui d'existence de fonctions de partage de secret : il existe effectivement des fonctions permettant de remplir la fonctionnalité de partage de secret, ces fonctions étant en plus peu coûteuses en ressources. Cependant la fonctionnalité remplie est relativement marginale.

Le second résultat concerne le chiffrement. Il a été montré que le seul système de chiffrement inconditionnellement sûr nécessite une clé aussi longue que le message à chiffrer. C'est le système du One-time-pad.

Le troisième résultat concerne l'authentification. Il a été montré qu'une authentification inconditionnellement sûre est possible avec une taille de la clé ne dépendant pas linéairement de la taille du message, mais de l'ordre du logarithme de la taille du message. En pratique, la taille de la clé ne dépend donc pas de la longueur des messages à signer, mais du nombre de messages.

Ces résultats sont donc intéressants mais ne permettent pas en général de définir un système opérationnel.

2.2.2. Sécurité combinatoire / calculatoire

L'idée suivante est de relier la sécurité des primitives cryptographiques à la théorie de la complexité, l'idéal étant de montrer qu'il n'existe pas pour l'attaquant de stratégie meilleure que celle consistant à tester une à une toutes les clés possibles.

Les problèmes peuvent être classés en différentes familles, notamment les classes P ou NP, en particulier NP-complet.

Il a été mentionné que toute la cryptographie asymétrique repose sur la difficulté de deux problèmes, la factorisation et le logarithme discret. Ces problèmes sont tous deux dans une même classe de complexité (NP-intermédiaire). Ils sont a priori difficiles, mais deux restrictions doivent être notées. La première, théorique, est le fait que la preuve même que P soit différent de NP n'a pas encore été apporté. La seconde est due au fait que la sécurité considérée est seulement une sécurité asymptotique : elle ne fait qu'indiquer le comportement de la complexité quand la taille du paramètre de sécurité (taille des clés) augmente. En pratique, les systèmes utilisent une taille de clé fixe, et il peut très bien exister un algorithme de résolution efficace pour une taille donnée. Ainsi, les tailles des clés pour le problème de factorisation sont-elles beaucoup plus importantes que les tailles de clés symétriques.

Concernant les fonctions symétriques, la situation est moins claire : elles ne sont pas en général positionnées dans une classe précise.

2.2.3. Sécurité « pratique »

En pratique, l'évaluation de la sécurité des fonctions cryptographiques repose en fait sur les analyses effectives menées par la communauté cryptologique : pour un problème ou un algorithme donné, les meilleures attaques parues sont considérées, et la sécurité effective est déduite de ces études. Les meilleures attaques ne sont normalement applicables que sur des versions réduites des algorithmes, et en fonction de la différence entre la version réduite et la version effective, on peut estimer une sécurité de l'algorithme. Naturellement d'autres facteurs interviennent mais faisant intervenir dans tous les cas un caractère artisanal et subjectif à l'évaluation.

Néanmoins, cette sécurité pratique permet ainsi de dimensionner certains paramètres. Ainsi, la taille de la clé pour la factorisation est au moins 10 fois plus importante qu'une clé symétrique.

Les fonctions asymétriques, devant offrir une fonctionnalité beaucoup plus puissante que les fonctions symétriques, et ne reposant que sur deux problèmes, sont souvent estimées plus fragiles.

2.2.4. Sécurité prouvable

Un dernier volet important est celui de la sécurité prouvable. Ces dernières années ont en effet vu le développement d'un important champ de recherche aboutissant à de nombreux résultats permettant de prouver la sécurité de certains protocoles. Malheureusement, la sécurité de ces protocoles n'est pas prouvée dans l'absolu, mais seulement en se rapportant à la sécurité des primitives sous-jacentes.

Ces résultats sont donc particulièrement intéressants pour montrer que la construction d'un protocole ou d'un système complet est correcte, mais ne permet pas de fournir une assurance absolue sur la sécurité réelle du système.

2.2.5. Sécurité des systèmes réels

Les notions de sécurité précédentes s'intéressent à la sécurité des algorithmes en soi. Pour un système réel, d'autres éléments de sécurité sont à considérer.

Par exemple les algorithmes implémentés en logiciel peuvent être sujet à des attaques permettant d'inhiber simplement le chiffrement ou de récupérer la clé utilisée à cause de la présence de virus. L'implémentation elle-même peut comporter des erreurs.

De plus, il existe des attaques permettant, en fonction du temps mis par le processeur ou de sa consommation, d'obtenir des informations sur la clé.

Ces menaces doivent être prises en compte lors de l'implémentation d'un système réel, mais ne sont pas considérées ici.

2.2.6. Synthèse

Les principaux systèmes cryptographiques existant ont fait l'objet de tentatives de cryptanalyse par une communauté importante, sans qu'aucune attaque réelle n'ait été trouvée au niveau algorithmique. Cependant leur sécurité ne repose sur aucune preuve mathématique.

Par ailleurs, on distingue souvent la sécurité des algorithmes symétriques et celle des algorithmes asymétriques, ces derniers devant réaliser une fonctionnalité plus difficile à obtenir, et reposant seulement sur deux problèmes bien précis.

3. Principes de la cryptographie quantique

3.1. Dispositif physique

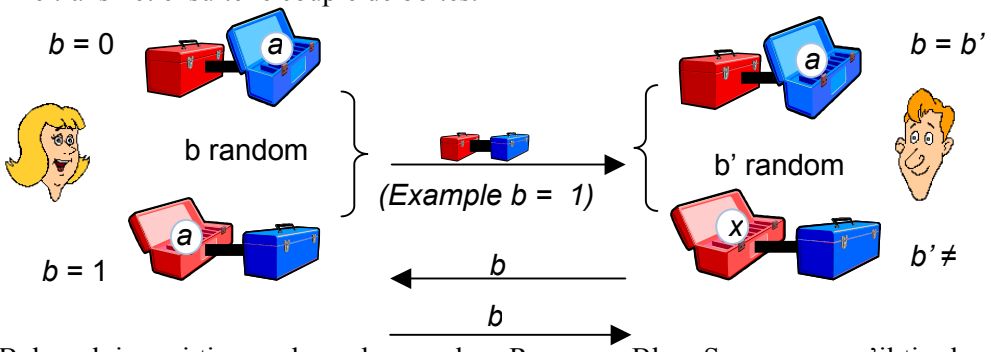
Quel que soit le dispositif matériel utilisé, les liaisons quantiques reposent sur un principe logique similaire. En effet, ces dispositifs matériels peuvent se modéliser par des couples de boîtes spéciales, vérifiant les propriétés suivantes. Elles peuvent contenir un bit d'information. Elles peuvent être ouvertes à volonté, ce qui permet de lire et / ou écrire le bit à l'intérieur puis refermées. Cependant, les deux boîtes ne peuvent être ouvertes simultanément. De plus, l'ouverture d'une des boîtes a comme conséquence que le bit à l'intérieur de l'autre boîte est effacé et remplacé par une nouvelle valeur aléatoire.

Ces boîtes sont la modélisation du dispositif historique de Bennett et Brassard de 1984. Ce dispositif repose sur la polarisation d'un photon. Deux repères orthogonaux, tournés de 45° sont utilisés. Dans un repère, il est possible de positionner l'orientation soit verticalement, soit horizontalement. Pour une base donnée, une mesure consiste à déterminer si la polarisation est verticale ou horizontale. Enfin, si le photon est positionné dans un état à 45° de la base utilisée pour la mesure, le résultat de la mesure sera une des deux possibilités de la base avec une probabilité ½ pour chacune.

3.2. Le protocole BB84

Alice veut communiquer avec Bob la valeur d'un bit. Elle dispose d'un couple de boîtes spéciale, une Rouge et une Bleue. Elle va tirer au sort entre la boîte Rouge et la boîte Bleue. Supposons qu'elle tombe sur le Rouge. Elle va alors ouvrir la boîte Rouge et placer le bit à transmettre dans cette boîte.

Elle transmet ensuite le couple de boîtes.



Bob va lui aussi tirer au hasard une valeur Rouge ou Bleu. Supposons qu'il tire la même valeur Rouge. Alors, il va ouvrir la boîte Rouge et lire la valeur à l'intérieur.

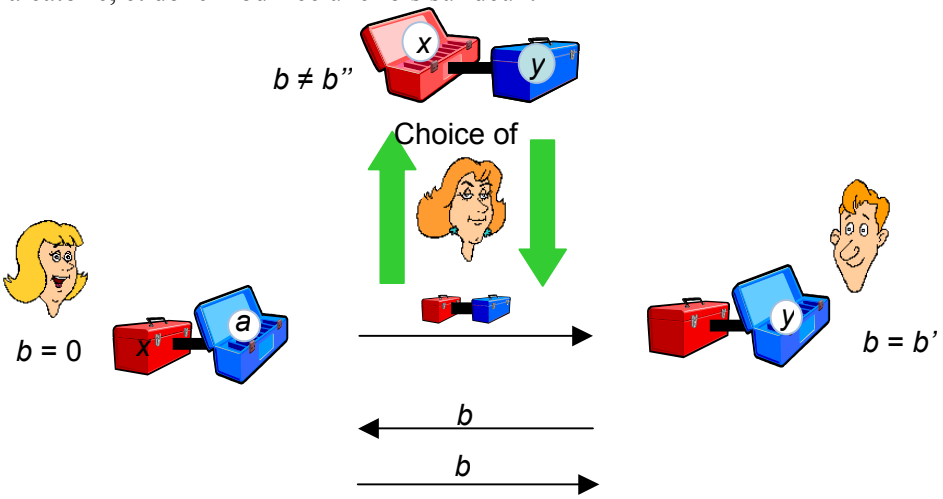
S'il avait tiré une valeur différente de celle d'Alice, Bleue ici, alors il aurait ouvert la boîte Bleue et observé une valeur qui était en fait aléatoire, du fait du placement initial de la valeur dans la boîte Rouge par Alice. Cependant, le protocole prévoit qu'après l'observation de Bob, Alice lui transmet, en clair, l'information de la boîte à considérer. Si Bob a tiré la même valeur Rouge ou Bleu, alors il conserve la valeur du bit observée, sinon, il jette cette observation. Statistiquement, la moitié des bits transmis est donc utilisable.

3.3. Sécurité vis-à-vis des écoutes

Supposons que Eve intercepte les boîtes et cherche à obtenir une information. La seule action qu'elle peut commencer à réaliser est d'ouvrir une des boîtes, c'est-à-dire d'en choisir une.

Si elle choisit la même boîte qu'Alice, par exemple ici la boîte Rouge, elle peut observer la valeur, puis refermer la boîte sans que cette observation ne change quoique ce soit dans les boîtes. Elle peut donc ensuite retransmettre le couple de boîte sans se faire repérer.

Cependant, si elle choisit l'autre boîte, la Bleue ici, la valeur dans la boîte Rouge sera alors rendue aléatoire, et donc modifiée une fois sur deux.



Au final, environ le quart des bits écoutés par Eve sera donc modifié. Pour détecter cette écoute éventuelle, une certaine proportion de bits produits seront sacrifiés : ils seront transmis en clair et cette valeur transmise sera comparée à la valeur précédemment obtenue.

On peut donc arbitrairement diminuer la probabilité d'une écoute non détectée.

Le mécanisme décrit permet donc une communication dont l'écoute est en pratique impossible.

3.4. Authentification

Pour avoir une communication réellement sûre, il faut ajouter une fonction d'authentification. Sans cette étape, Eve pourrait ou bien tromper le protocole précédent en modifiant certains des messages en clair, ou bien réaliser une attaque par le milieu : ouvrir une session quantique avec Alice en se faisant passer pour Bob puis indépendamment une session avec Bob en se faisant passer pour Alice.

Or, il existe deux types d'outils permettant de réaliser cette fonction d'authentification. Le premier est celui des fonctions asymétriques ; le second celui utilisant une fonction d'authentification inconditionnellement sûre.

Le premier cas permet de conserver la souplesse offerte par les fonctions asymétriques. La sécurité en est cependant amoindrie, mais elle reste pourtant meilleure qu'un système classique, car la fonction asymétrique n'intervient que pour l'authentification et non pour protéger des données. Cela permet notamment de se prémunir contre les attaques passives, en particulier celles où l'attaquant enregistre des messages afin d'attendre de disposer de capacités d'analyse plus importante.

Dans le second cas, l'authentification inconditionnellement sûre nécessite en pratique un nombre de bits de taille fixe pour un message quelconque. Supposons qu'Alice et Bob partage un certain nombre de bits de secret. L'authentification de l'ensemble d'un échange va consommer une certaine quantité de bits secrets. Mais l'échange quantique va permettre de transmettre une grande quantité de nouveaux bits de secrets communs. Une partie de ces bits peut donc être utilisée pour compenser les bits d'authentification utilisés, ce qui permet de conserver une même capacité d'authentification.

Il est donc possible d'utiliser un système d'authentification inconditionnellement sûre, la contrainte étant de devoir disposer au départ d'un secret commun.

3.5. Protocole complet

La définition complète d'un lien nécessite encore certaines étapes techniques, pour tenir compte des dispositifs physiques non parfaits utilisés : chaque élément de la chaîne comporte des probabilités de perturbation, qui pourraient être considérées comme autant d'écoute effective. Il faut donc ajouter des algorithmes de traitement des chaînes de bits produits pour pouvoir s'assurer d'un secret réel (réconciliation, amplification de secret). Ces algorithmes sont bien maîtrisés et permettent en final d'obtenir une chaîne réellement secrète, avec une probabilité arbitrairement faible.

Il est donc possible d'obtenir un lien inconditionnellement sûr entre deux utilisateurs.

3.6. Limitations concrètes

Un lien quantique comporte des limitations pratiques importantes.

La première est la distance maximale d'une liaison. En effet, la distance maximale entre deux utilisateurs est limitée à un certain nombre de kilomètres, selon le médium utilisé. Les distances typiques vont de 10 à 100 kilomètres, avec comme exception notable les communications satellitaires. Pour obtenir un système de communication à longue distance, cette limitation se traduit par la nécessité d'utiliser des nœuds de communications afin d'acheminer l'information. Ces nœuds doivent être complètement de confiance, puisqu'il est indispensable de déchiffrer puis re-chiffrer les données à transmettre.

La deuxième est le débit effectif. Ce débit dépend de la distance entre les deux utilisateurs, mais est de toutes manière limité, de l'ordre de quelques kilobit/s actuellement.

Enfin, la dernière limitation est financière : le coût d'un équipement quantique est nettement plus important qu'un équipement de chiffrement classique, notamment qu'un équipement de basse sécurité. Il faut noter cependant que ces coûts pourraient être sensiblement diminués avec des quantités raisonnablement importantes, et d'éventuelles améliorations permettant notamment d'utiliser des composants optiques standard.

4. Apports de sécurité d'un système quantique

L'apport de sécurité d'un système quantique est souvent mal défini, pris entre l'annonce commerciale d'une sécurité inconditionnelle et le rejet dû à la présence de cryptographie classique dans les systèmes quantiques opérationnels.

Pour avoir une vue complète de l'apport en sécurité, il faut d'abord préciser le système exact qui est considéré c'est-à-dire d'une part le rôle précis des moyens quantiques et classiques, avec les contraintes opérationnelles acceptées, notamment au niveau de la distribution des éléments secrets, et d'autre part les menaces pertinentes pour le système en question.

4.1. Différents types de systèmes

Différents types

Deux principaux points peuvent différer d'un système à l'autre : d'une part l'authentification peut être inconditionnelle, ou reposer sur des algorithmes asymétriques, et d'autre part les bits secrets produits peuvent être utilisés soit directement pour chiffrer les données (one-time-pad), soit servir de clé pour un algorithme de chiffrement symétrique classique.

Si l'authentification est inconditionnelle, les deux intervenants sur une liaison doivent partager un secret initial commun. On est proche, de ce point de vue, du déploiement d'un système de chiffrement symétrique classique où un couple d'utilisateurs doit partager une clé commune.

Si l'authentification est asymétrique, la souplesse des systèmes asymétriques est obtenue : chaque utilisateur peut ne posséder que sa clé privée, et une information authentifiée, typiquement la clé publique authentifiée par une autorité de certification.

Si le chiffrement est direct, la principale limitation est alors le débit utile : seules les communications acceptant un débit très faible, actuellement, sont possibles. Dans le cas d'un chiffrement symétrique supplémentaire, il faut noter que la fréquence de renouvellement de la clé peut alors être très importante, de l'ordre de plusieurs fois par seconde.

Différents systèmes

Le premier système possible est donc le système « tout quantique ». Dans ce cas, l'authentification est inconditionnelle, et les bits produits servent directement à chiffrer les données.

Le deuxième système envisagé est celui où l'authentification reste inconditionnelle, mais où les bits produits servent de clé pour un algorithme classique.

Le troisième est celui où l'authentification est asymétrique, et où les bits produits servent de clé.

Le seul cas qui ne sera pas envisagé est donc celui où l'authentification est asymétrique et le chiffrement inconditionnel. Ce cas pourrait être considéré, mais en pratique, ne semble pas se rapprocher d'une utilisation opérationnelle.

4.2. Menaces

Les principales menaces sont

- Un secret est compromis. Différents sous cas peuvent être distingués : le secret peut être la clé symétrique de longue durée utilisé dans le cas d'un système de chiffrement purement symétrique, la clé asymétrique dans le cas d'un système utilisant une authentification ou un échange de clés asymétrique, ou encore le secret d'authentification inconditionnelle. Dans le cas d'un système quantique avec des nœuds intermédiaires de communication, ces compromissions peuvent avoir lieu à chaque nœud. On ne prend en compte ici que la compromission des clés de durée significative, et non celle des clés intermédiaires (clés de session ou données d'authentification évoluant).
- L'algorithme de chiffrement symétrique a été cryptanalysé par l'attaquant.
- L'algorithme asymétrique a été cryptanalysé par l'attaquant.
- Pour un système de communications quantiques avec des nœuds intermédiaires de communications, le trafic complet est écouté par un attaquant.

Il faut de plus distinguer les attaques actives des attaques passives : une attaque passive se contente d'observer un trafic, et peut donc rester indétectée. Une attaque passive suppose une interaction effective de l'attaquant avec le système, ce qui peut ou non être détecté selon les systèmes.

Enfin, on s'intéressera également à la sécurité des messages passés : dans certains cas, notamment en cryptanalyse, les informations peuvent être stockées par l'attaquant, en attente des

différents progrès possibles, ou de l'achèvement de l'exécution de certains algorithmes de cryptanalyse. Dans d'autres cas au contraire, l'attaquant doit disposer rapidement de la bonne information pour réussir son attaque.

4.3. Caractéristiques du système avec authentification asymétrique et chiffrement symétrique

Ce système utilise donc des techniques asymétriques pour l'authentification, et les bits produits sont utilisés comme clé pour un algorithme symétrique.

A première vue, ce système n'est pas plus résistant qu'un système classique, puisque s'appuyant sur des techniques asymétriques pour l'authentification et sur des techniques symétriques pour le chiffrement des données. Ce point de vue est incomplet.

Comparons ce système avec le système classique le plus proche : celui où une authentification asymétrique est utilisée combinée à une élaboration de clé de session (Diffie-Hellman ou autre) et où cette clé de session est utilisée par un algorithme de chiffrement symétrique.

Les contraintes opérationnelles sont identiques pour les deux systèmes, en-dehors du problème des distances de communication.

Le système classique est sujet notamment aux attaques suivantes :

- cryptanalyse de l'algorithme symétrique,
- cryptanalyse de l'algorithme asymétrique d'élaboration de clé,
- cryptanalyse de l'algorithme asymétrique d'authentification,
- compromission de la clé d'authentification.

En pratique, on peut prendre comme hypothèse qu'un attaquant réussissant à cryptanalyser l'un des algorithmes asymétriques peut cryptanalyser l'autre. Les deux menaces de cryptanalyse peuvent donc être regroupées.

Vis-à-vis des menaces de compromission de la clé d'authentification et de cryptanalyse de l'algorithme symétrique, les conséquences sont globalement identiques.

Vis-à-vis de la cryptanalyse de l'algorithme asymétrique, les conséquences sont différentes, car l'algorithme n'est utilisé que pour l'authentification dans le cas quantique, alors qu'il sert également à transmettre la clé, d'une manière ou d'une autre. Par exemple dans le cas d'une élaboration de clé de type Diffie-Hellman, casser l'algorithme asymétrique signifie que l'attaquant peut effectivement reconstruire lui aussi la clé de session.

Dans le cas quantique, un attaquant passif ne retire aucune information, alors qu'il peut retrouver toutes les données utiles dans le cas classique. Pour opérer une attaque entre deux utilisateurs dans le cas quantique, il doit retrouver les secrets des deux interlocuteurs et réaliser une attaque par le milieu, c'est-à-dire initier une communication avec le premier interlocuteur en se faisant passer pour le second, et une autre avec le second en se faisant passer pour le premier, puis en relayant ainsi les messages échangés.

Une telle attaque est donc nettement plus complexe à opérer qu'une simple écoute. De plus, les données échangées dans le passé reste de toutes manières protégées dans le cas quantique, alors que dans le cas classique, un attaquant peut stocker des données échangées en attendant les progrès de la cryptanalyse, ou la fin des traitements parfois très longs de cryptanalyse.

En contrepartie, un tel système est plus onéreux et complexe à mettre en oeuvre. Surtout, si des distances importantes sont nécessaires, ce système nécessite la présence de nœuds de confiance, ce qui ajoute une faiblesse de sécurité. Il faut noter qu'un système réel combinerait vraisemblablement les deux solutions : un chiffrement de bout en bout classique et un système quantique complémentaire.

En résumé, cette solution quantique offre l'avantage, dans le cas d'une cryptanalyse des algorithmes asymétriques, de garantir la sécurité des communications vis-à-vis d'un attaquant passif. Autrement dit, un attaquant voulant intercepter une communication devra réaliser une attaque par le milieu, plus lourde à réaliser. En particulier, les messages passés restent protégés quels que soient les progrès de cryptanalyse réalisés. Cette solution présente l'inconvénient pour les longues distances de nécessiter des nœuds de confiance.

4.4. Caractéristiques du système avec authentification inconditionnelle et chiffrement symétrique

Il est possible de comparer cette solution avec deux systèmes classiques : celui identique au précédent (authentification et élaboration de clé de session par des techniques asymétriques) et celui où les couples d'utilisateurs partagent une clé symétrique. Cette clé partagée peut ainsi servir à dériver ou transmettre une clé de session.

4.4.1. Comparaison avec un système asymétrique

Par rapport au cas précédent, aucun algorithme asymétrique n'intervient plus dans le système quantique, et donc la cryptanalyse d'un tel algorithme n'a plus aucun conséquence.

De plus, la compromission du secret d'authentification a des conséquences bien moindre : dans le cas quantique, il est sans cesse renouvelé, et donc la compromission doit s'opérer juste avant l'attaque effective, alors que dans le cas classique, l'attaquant a tout le temps pour d'une part acquérir le secret, puis ensuite l'utiliser pour réaliser son attaque active.

Le gain en sécurité est donc net.

4.4.2. Comparaison avec un système symétrique

Un tel système est envisageable par exemple dans le cadre d'un système de distribution de type Secoqc dans la mesure où les centres de confiance sont relativement peu nombreux et ont des communications entre eux relativement stables.

Les menaces à prendre en compte sont

- cryptanalyse de l'algorithme symétrique,
- compromission de la clé symétrique / compromission du secret d'authentification.

Dans le cas de compromission du secret, les conséquences sont différentes. En effet, dans le cas classique, la simple écoute des communications permet de retrouver toutes les données échangées. Dans le cas quantique, l'attaquant doit mener une attaque active. De plus, le secret d'authentification est renouvelé avec les données échangées : un attaquant doit acquérir le secret d'authentification à un instant donné et s'en servir juste à ce moment pour monter une attaque par le milieu.

Dans le cas de cryptanalyse de l'algorithme symétrique, les conséquences sont grossièrement similaires. Il faut cependant noter que l'algorithme symétrique utilisé pour élaborer les clés de session dans le cas symétrique utilise une clé dont la validité est relativement longue. Or cryptanalyser un algorithme est en général un processus long et coûteux : une cryptanalyse réussie signifie trouver un algorithme de reconstitution meilleur que l'essai systématique de toutes les clés, mais il peut rester très long à opérer pour chaque clé à retrouver. La rentabilité d'une telle attaque est donc nettement moins importante. Typiquement, il peut s'agir d'une journée de communications, voire beaucoup plus dans le cas classique, et une fraction de seconde dans le cas quantique.

En résumé, le système quantique permet une résistance bien plus grande face à la menace de compromission du secret, l'attaquant devant réaliser une attaque active en temps réel, et ajoute un peu de résistance vis-à-vis de la cryptanalyse de l'algorithme symétrique.

Là aussi, il ajoute au contraire une vulnérabilité du fait des nœuds de confiance dans le cas de systèmes à longue distance.

4.5. Caractéristiques du système « tout quantique »

Dans le cas d'une simple liaison, le système peut résister pratiquement à toutes les attaques. En effet, le seul cas d'attaque possible est celui où l'attaquant peut prendre connaissance à un instant du secret d'un des participants, et se fait passer pour celui-ci vis-à-vis de l'autre juste à cet instant.

Il faut noter que cette attaque ne fournit aucune information sur les messages passés, et ne permet également d'obtenir aucune information sur les messages futurs, puisque si les deux utilisateurs autorisés tentent une communication ultérieurement, la tentative d'intrusion sera détectée.

Dans le cas d'un réseau de communications avec plusieurs nœuds intermédiaires, la menace est la compromission d'un des nœuds. Dans ce cas, si l'attaquant prend le contrôle complet (écoute de toutes les communications), il peut effectivement accéder aux différentes communications.

En résumé, pour un système point à point, un tel système permet effectivement de réaliser une sécurité inconditionnelle. Le problème vient des limitations pratiques énoncées : débit faible, distance limitée (s'il n'existe pas de nœuds intermédiaires), coût important. On peut ajouter aussi la nécessité de disposer au départ d'une valeur secrète commune.

4.6. Synthèse

Généralités

Dans tous les cas de figure, le système quantique apporte un certain niveau de sécurité complémentaire. Pour les systèmes à longue distance, il entraîne cependant une vulnérabilité supplémentaire, aux niveaux de nœuds intermédiaires. Cette vulnérabilité existe pour tous les systèmes ; on ne la répètera pas dans chaque cas.

La sécurité supplémentaire provient en général du fait que pour beaucoup d'attaques comparables, l'attaquant peut se contenter d'une écoute dans le cas classique, alors qu'il doit réaliser une attaque active (attaque par le milieu) pour intercepter une communication. Certaines autres attaques restent valables dans les deux cas.

Situation pour chaque système

Pour le système « tout quantique », la sécurité inconditionnelle est à peu près réalisée. Les inconvénients majeurs proviennent des limitations intrinsèques (débit, souplesse d'utilisation).

Pour le système quantique avec chiffrement symétrique, la compromission des éléments secrets est beaucoup plus difficilement utilisable par l'attaquant (nécessité de réaliser une attaque active au moment même où l'information est disponible) ; la cryptanalyse de l'algorithme symétrique entraîne par contre une vraie faiblesse du système comme pour un système symétrique, même si l'impact est un peu limité par la fréquence de renouvellement des clés.

Pour le système quantique avec authentification asymétrique et chiffrement symétrique, les deux menaces de compromission de la clé d'authentification et de cryptanalyse de l'algorithme symétrique ont les mêmes conséquences que dans un système classique. Vis-à-vis de la menace de cryptanalyse de l'algorithme asymétrique, le système quantique oblige l'attaquant à réaliser une attaque active.

5. Le projet européen Secoqc

5.1. Systèmes quantiques existants

Actuellement, des systèmes quantiques opérationnels commencent à être proposés. Ces systèmes sont uniquement des systèmes de sécurisation de lien.

Au niveau matériel, différentes techniques sont en compétition, chacune présentant actuellement certains avantages et inconvénients.

Le support peut être en fibre optique, mais aussi en air libre, ce qui inclut également les communications satellitaires.

Sur fibre optique, plusieurs expérimentations dépassent les 100 Kms, et en air libre, une distance de 24 Kms a été atteinte.

De plus, d'autres essais mettent l'accent sur le débit, avec des objectifs de l'ordre du million de bits / seconde.

Parmi les premiers produits commerciaux proposés, on peut citer par exemple IdQuantique qui propose un produit de chiffrement de lien couplant un système quantique pour la gestion des clés et un système classique. La limitation en débit est ici comblée au prix de l'utilisation d'un chiffrement symétrique complémentaire. Ces produits cherchent à cibler des applications où la limitation en distance n'est pas un obstacle. On peut citer par exemple le cas d'un centre de secours d'un système de données, situé typiquement à quelques dizaines de kilomètres du centre principal.

5.2. L'objectif du projet Secoqc

Le projet européen Secoqc a pour ambition de développer les systèmes quantiques au sein de l'Europe. Son but est le développement d'un réseau de communication à longue distance permettant

de distribuer des secrets symétriques à des nœuds à des distances quelconques. Cet objectif se décline en :

- la réalisation d'une technologie de communication point-à-point de distribution de clés quantiques complètement fonctionnelle, « ready-to-market » ;
- le développement d'un niveau abstrait d'architecture permettant des communications à longue distance par la combinaison des technologies quantiques et cryptologiques classiques ;
- la définition effective d'un réseau de communication quantique orienté vers l'utilisateur.

Ces objectifs se déclinent en différents objectifs scientifiques : amélioration des techniques physiques de distribution de clés quantiques, définition d'interfaces au niveau utilisateur, et définition d'interfaces au niveau réseau vis-à-vis des fournisseurs de services quantiques, et développement du concept de réseau quantique.

Ce projet a commencé en 2004 et finira en 2008. Il fait intervenir 41 participants appartenant à 11 pays européens, soit un grand nombre des acteurs majeurs de la cryptographie quantique en Europe.

Il a été décomposé en différents sous-projets :

- un sous-projet quantique,
- un sous-projet infrastructure,
- un sous-projet implémentation.

Le sous-projet quantique regroupe tous les développements de la partie physique. En effet, différentes technologies sont possibles, sans qu'une de celles-ci se dégage de manière indiscutable. Ces différentes possibilités sont donc étudiées et améliorées, et l'implémentation finale réalisée combine ces différentes technologies.

Le sous-projet Infrastructure regroupe la définition des différents algorithmes nécessaires à la sécurité, la définition du réseau, ainsi qu'une partie intégration et une partie certification.

Enfin, le sous-projet implémentation regroupe les réalisations concrètes permettant d'obtenir un démonstrateur de réseau quantique.

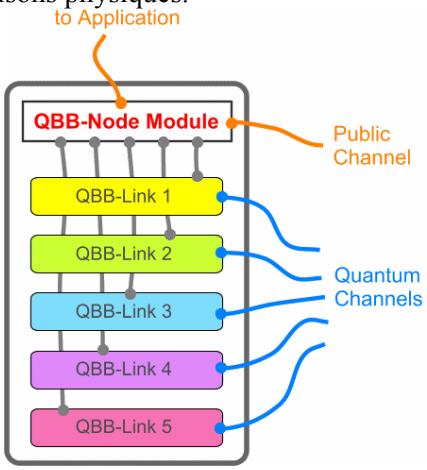
Les moyens physiques ne seront pas développés ici. A l'inverse, on s'intéressera plus particulièrement à la partie Infrastructure, notamment à la définition du réseau et de ses interfaces.

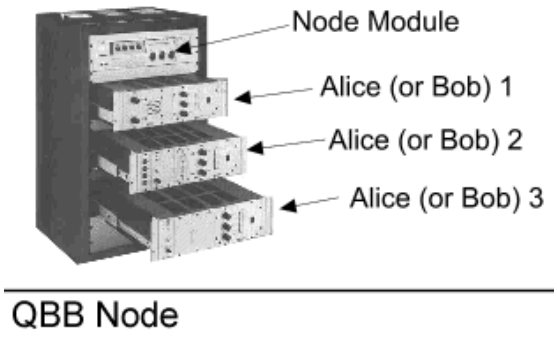
5.3. Un réseau quantique

Le réseau quantique peut être divisé en une partie constituant le cœur du réseau lui-même (« Quantum Backbone »), et une partie d'accès au réseau (« Quantum Access Network »).

Le cœur du réseau est constitué d'un ensemble de nœuds reliés entre eux. Chaque couple de nœuds en relation construit un secret commun qui sera utilisé pour calculer le secret final de bout en bout.

Un noeud est constitué de deux niveaux : un niveau de gestion du nœud, et un niveau de gestion des liaisons physiques.

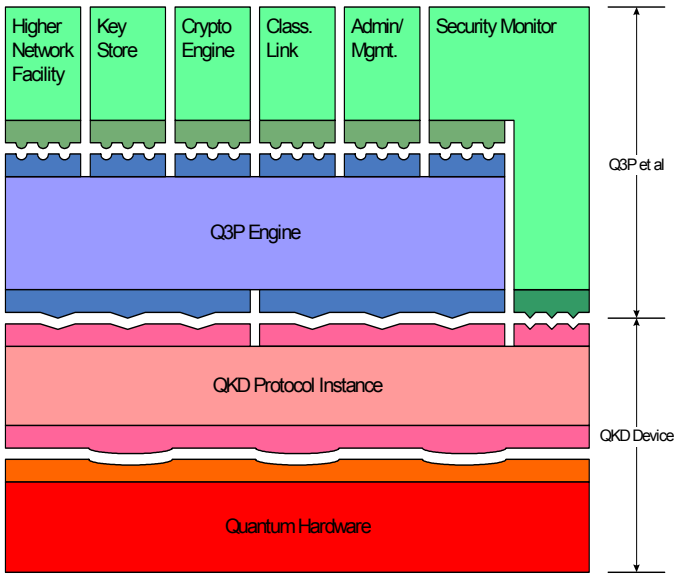




Au niveau d'un module de gestion d'une liaison physique sont opérés tous les traitements spécifiques à une implémentation physique particulière. Une interface est définie avec le module de gestion de nœud, lequel sera chargé de réaliser tous les traitements et toutes les communications

permettant de réaliser l'ensemble du protocole de communication quantique. Un protocole de communication entre 2 noeuds a également été défini : le protocole Point à Point Quantique (« Quantum Point to Point Protocol » - Q3P)

Ce protocole régit donc les échanges entre 2 nœuds du réseau. Il inclut l'utilisation des primitives cryptologiques classiques et des algorithmes de réconciliation et d'amplification de secret permettant de réaliser l'ensemble des traitements nécessaires à l'élaboration d'un secret commun.



Ce schéma montre notamment la présence d'un moteur crypto pour les différents algorithmes cryptologiques utilisés, mais aussi celle d'un magasin de clés, afin de gérer la génération et la fourniture des différentes clés produites.

Par ailleurs, il faut noter la possibilité de définir des fonctions permettant, selon le réseau, de partager le secret et de définir des chemins différents indépendants qui obligent un attaquant à prendre simultanément le contrôle de plusieurs nœuds. Si la topologie du réseau le permet, ces techniques permettent donc de renforcer la sécurité de la transmission.

5.4. Autres avancées

Certification, standardisation

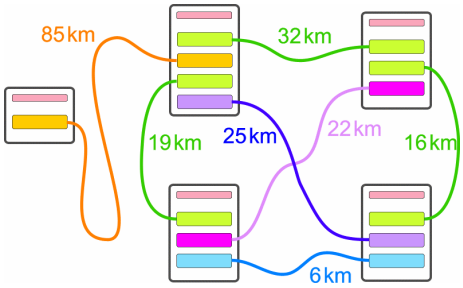
Une partie importante du projet a également été de préparer une formalisation accrue de la sécurité et des standards quantiques.

Plus particulièrement, la question des spécifications de sécurité (profil de protection et cible de sécurité) d'un réseau quantique a été étudiée. Les approches existantes pour les systèmes classiques ont été ainsi étendues pour les appliquer aux spécificités quantiques. Globalement, la partie purement réseau reste classique dans une très large part, mais des menaces spécifiques au niveau quantique ont été définies, avec en correspondance des critères standardisés, et des exigences de conformité à ces critères.

La standardisation se situera à différents niveaux, notamment au niveau des composants d'optique quantique, au niveau des algorithmes de distillation de clés, au niveau des preuves de sécurité ainsi qu'aux interfaces nœud / liaison.

Démonstrateur

Un des résultats visibles du projet est le démonstrateur de réseau quantique. Ce démonstrateur comportera 5 nœuds et mettra en œuvre les différentes techniques développées dans le cadre de Secoqc. En effet au niveau de chaque nœud, les différents modules physiques développés seront utilisés et le maillage permettra de faire fonctionner effectivement les différents algorithmes réseau.



Les distances typiques sont donc de quelques dizaines de kilomètres, avec un débit de l'ordre de plusieurs kilobit/s.

Ce démonstrateur sera opérationnel autour de Vienne, en Septembre 2008.

Points divers

Le projet Secoqc a donné lieu à différents développements dans des directions très diverses. On décrit ici juste l'exemple du calcul réparti à titre d'illustration.

Une fonctionnalité cryptographique particulière est celle où un résultat doit être obtenu à partir d'un secret d'un participant A et d'un autre secret d'un participant B, sans que ces secrets ne soient divulgués. Ce résultat peut être par exemple l'information « ces deux secrets sont-ils identiques ? » ou peut être la sortie d'une fonction beaucoup plus compliquée. Une telle fonctionnalité peut être obtenue avec des outils classiques, mais sa sécurité repose sur l'existence de fonctions non-inversibles à trappe. L'utilisation d'outils quantiques permet de diminuer l'exigence de sécurité, puisqu'il est alors possible de réaliser un système n'exigeant que l'existence de fonctions non-inversibles.

La sécurité obtenue n'est donc pas inconditionnelle, mais l'exigence de sécurité est diminuée.

De manière générale, une autre voie de progrès est d'assurer la bonne synchronisation des magasins de clé, de telle sorte qu'un adversaire ne puisse désynchroniser deux participants autorisés. Avec ces techniques, il est ainsi possible de renforcer la résistance aux attaques par déni de service.

5.5. Perspectives

Il existe des techniques permettant de s'affranchir de la limitation en distance. Ces techniques ne sont pas encore opérationnelles actuellement, mais ne présentent pas de difficultés théoriques importantes.

En effet, il est possible de définir des répéteurs quantiques. Ces répéteurs permettent de lier des éléments quantiques entre deux nœuds proches dans un premier temps, puis petit à petit entre des nœuds distants. Les éléments quantiques finaux ainsi construits entre deux nœuds distants ne pourront pas être déduits des données utilisées au niveau des nœuds intermédiaires. Ce fonctionnement est notamment envisageable car il existe des techniques de purification permettant de combiner plusieurs ensembles d'éléments quantiques afin d'obtenir un seul ensemble quantique mais de meilleure qualité. Il est ainsi envisageable de conserver une qualité constante le long de la chaîne de communication.

Ces techniques nécessitent cependant la maîtrise de la conservation d'états quantiques sans perte d'intrication, sur une certaine durée. Ces techniques ne sont actuellement pas maîtrisées, mais elles ne semblent pas poser de problème théorique fondamental. Il n'est donc pas exclu de voir apparaître d'ici quelques dizaines d'années de tels éléments prêts à l'emploi. Un tel système serait similaire au système actuel, mais sans aucune limitation en distance, ou autrement dit sans la nécessité de disposer de centres de confiance.

5.6. Synthèse

Le projet Secoqc a donc permis de passer du stade d'une simple liaison quantique à un système complet de distribution de clés.

Pour cela, des développements ont eu lieu dans un grand nombre de directions afin de couvrir tous les niveaux nécessaires à un tel système. Notamment, les principaux algorithmes cryptologiques et réseau à utiliser ont été précisés, avec une définition claire des interfaces des différents modules.

Au final, les résultats sur l'intérêt des outils quantique indiqués au chapitre 4 ne sont pas remis en question, mais la réalisation effective d'un réseau de distribution de clés quantique a été rendue tout à fait envisageable en pratique, ce qui était bien le principal objectif de ce projet.

Toute information complémentaire peut être trouvée à l'adresse suivante : www.secoqc.net.

6. Conclusion

Les différents apports potentiels d'un système quantique ont été exposés. Ceux-ci dépendent étroitement du système précis considéré, et des menaces envisagées. Il est donc trop réducteur de vouloir discuter de LA sécurité apportée par un système quantique, mais il vaut mieux discuter des différentes caractéristiques d'un tel système, en comparaison avec le système classique le plus proche du point de vue opérationnel.

Par exemple dans le cas de systèmes utilisant la cryptologie asymétrique, un système quantique par rapport à un système classique présente notamment l'avantage d'obliger un attaquant à réaliser une attaque active. Dans le cadre d'un réseau de type Secoqc en supposant des nœuds relativement fixes utilisant une authentification inconditionnelle, la sécurité entre nœuds est réelle ; cependant, dans ce cas et dans tous les cas de systèmes quantiques à longue distance, les nœuds eux-mêmes deviennent des points de vulnérabilité, les données de clés transitant en clair à l'intérieur d'un nœud.

Par ailleurs, les systèmes quantiques actuellement sont essentiellement utilisés pour distribuer des clés ; dans ce cadre-là, la menace de cryptanalyse de l'algorithme de chiffrement symétrique reste globalement identique dans les cas classique et quantique.

La sécurité doit donc être étudiée précisément pour chaque système, en prenant en compte les avantages et inconvénients précis d'une solution quantique. Le fait que la sécurité inconditionnelle complète n'est en général pas obtenue ne doit pas masquer qu'un tel système peut apporter certains éléments de sécurité complémentaires ; cependant ceux-ci doivent être appréciés également en fonction du coût encore important induit par ces équipements actuellement.

Jeudi 8 novembre 2007

Cryptographie:

Perspectives

AACS, nouveau standard de protection des contenus pré-enregistrés haute-définition

Nicolas Prigent, Mohamed Karroumi,
Mathieu Chauvin, Marc Eluard, Yves Maetz

Thomson R&D France,
Technology Group, Corporate Research, Security Lab
1, Avenue de Belle Fontaine,
35 576 Cesson-Sévigné Cedex, France

Résumé

Suite au fiasco du CSS (*Content Scramble System*) qui devait protéger les contenus diffusés sur DVD, l'industrie des contenus audiovisuels de loisir a décidé de profiter de l'avènement des standards HD-DVD et Blu-Ray Disc en 2006 pour promouvoir un nouveau système de protection des contenus pré-enregistrés, l'AACS (*Advanced Access Content System*). Bien plus abouti techniquement et scientifiquement, il a été conçu en tirant parti de l'expérience issue de CSS ainsi que de résultats de recherche en cryptographie. Pourtant, il a été rapidement attaqué avec succès. Dans cet article, nous présentons AACS et évoquons les attaques dont il a été récemment victime.

Following the failure of CSS (Content Scramble System) that was supposed to protect pre-recorded DVD contents, the Media & Entertainment industry decided to take benefit from the new generations of DVDs (HD-DVD and Blu-Ray) to promote a new content protection system called AACS (Advanced Access Content System). Both technically and scientifically better, AACS was designed using the experience taken from the failure of CSS and from research results in cryptography. Nevertheless, it has been quickly successfully attacked. In this article, we describe AACS and present the attacks it was a victim of.

1 Introduction

L'industrie du contenu audiovisuel de loisir fait face à un très important phénomène de piratage : à peine un contenu est-il disponible sur CD ou sur DVD que des copies pirates sont échangées, notamment en *peer-to-peer*. Ce piratage est l'œuvre non seulement d'organisations criminelles, mais aussi du grand public. Ainsi, les ayants droits ont décidé de profiter de l'avènement des standards HD-DVD et Blu-Ray Discs pour promouvoir un nouveau système de protection des contenus pré-enregistrés, l'AACS (*Advanced Access Content System*). Ce standard est promu et placé sous l'autorité du consortium AACS LA (*AACS Licencing Administrator*) qui regroupe IBM, Intel, Microsoft, Panasonic, Sony, Toshiba, The Walt Disney Company et les studios Warner Bros. Sa mise en œuvre est obligatoire sur tout disque optique pré-enregistré de contenus vidéo suivant les standards HD-DVD ou Blu-Ray.

Dans cet article, nous présentons AACS. Dans la section 2, nous introduisons ses objectifs de sécurité. Dans la section 3, nous décrivons la préparation d'un disque protégé par AACS. Nous traitons de sa lecture par un appareil compatible dans la section 4. Les fonctions avancées d'AACS, à savoir la révocation d'appareils et le traçage des traîtres (*traitor tracing*) sont respectivement détaillés en sections 5 et 6. Enfin, dans la section 7, nous présentons brièvement les attaques dont AACS a été victime quelques mois à peine après son déploiement.

2 Objectifs d'AACS

Afin de fournir une méthode robuste et renouvelable de protection des contenus audiovisuels pré-enregistrés, AACS entend tirer parti de l'expérience de l'échec de CSS. Il s'en distingue notamment en étant un standard aux spécifications publiques et en utilisant des algorithmes cryptographiques réputés sûrs après une étude intensive par la communauté, à savoir AES (*Advanced Encryption Standard*) et ECDSA (*Elliptic Curve Digital Signature Algorithm*).

De manière détaillée, les objectifs d'AACS pour la protection des disques pré-enregistrés sont les suivants :

- protection contre la copie bit-à-bit de disques ;
- gestion des règles d'usage (*usage rules*) s'appliquant au contenu ;
- révocation des lecteurs compromis[1] ;
- traçage des traîtres.

Tous ces objectifs se traduisent par des mesures cryptographiques qui seront présentées dans les sections suivantes.

3 Création d'un DVD protégé par AACS

La création d'un DVD protégé par AACS fait intervenir trois entités :

- le propriétaire du contenu, qui fournit les contenus vidéos ;
- le réplicateur, qui produit les disques enregistrés ;
- l'AACS LA, qui gère la distribution des clés.

Le propriétaire du contenu n'est impliqué qu'au tout début et à la toute fin du processus. Il fournit le contenu en clair sous la forme d'un ou plusieurs titre(s) (un même disque pouvant éventuellement contenir plusieurs titres) accompagné(s) d'un ensemble de règles d'usage. À la fin du processus, il récupère les disques protégés par AACS. Entre ces deux étapes, le réplicateur et l'AACS LA se chargent de toutes les opérations relatives à la sécurité. Dans cette section, nous présentons les quatre opérations intermédiaires du processus de création d'un disque compatible AACS en soulignant le rôle de chacune des entités.

3.1 Chiffrement du contenu par la ou les *title key(s)*

Après avoir reçu le contenu et les règles d'usage, le réplicateur assigne à chaque titre une *title key* tirée aléatoirement et qu'il garde secrète. Il chiffre chacun des titres avec la *title key* qui lui est associée. C'est cette version chiffrée qui sera écrite sur le disque. Toute la sécurité de l'accès au titre reposera par la suite sur l'accès à la *title key* : quiconque la connaît sera capable de déchiffrer le titre. De ce fait, les opérations présentées plus avant dans cette section visent à chiffrer la *title key* pour que seuls les appareils compatibles AACS et non-revoqués puissent la déchiffrer, et donc accéder au contenu, et ce uniquement sur des disques originaux, donc non reproduits sur des graveurs grand-public.

3.2 Protection contre la copie bit-à-bit par la *Volume ID*

AACS a pour objectif de protéger les disques contre la copie bit-à-bit. À cette fin, chaque disque pré-enregistré compatible AACS dispose d'un identifiant de volume (*Volume Identifier* ou *Volume ID*) qui est partiellement placé dans une zone du disque inaccessible en écriture aux graveurs grand-public. Il est de plus inscrit d'une manière spécifique sur le disque pour que seuls des *racks* optiques respectant la norme AACS puissent accéder au *Volume ID*.

Un lecteur utilise le *Volume ID* pour déchiffrer la ou les *title key(s)*. Si un attaquant copie servilement un HD-DVD ou un Blu-Ray Disc avec un graveur grand-public, il ne sera pas en mesure d'écrire la partie du *Volume ID* à placer sur la zone du disque non-atteignable par celui-ci, et le disque résultant ne permettra donc pas l'accès au contenu.

Lors de la production, le réplicateur assigne à la ligne de disque qu'il va produire un *Volume ID* unique tiré aléatoirement. Remarquons que selon la norme AACS, à un titre (ou ensemble de titres) peut correspondre une ou plusieurs lignes de disques, à la discrétion du réplicateur. En d'autres termes, c'est le réplicateur qui

[1] Notons que la révocation dans AACS empêche les appareils révoqués d'accéder aux disques produits après leur révocation. Toutefois, ils peuvent lire les disques qui ont été produits avant leur révocation.

choisit si tous les disques comportant le ou les même titres partageront le même *Volume ID* (et seront donc totalement identiques), ou si il en produira différentes séries ayant un *Volume ID* différent.

3.3 Mise en oeuvre de la révocation par le *Media Key Block*

Pour créer un DVD protégé par AACS, le réplicateur obtient ensuite auprès de l'AACS LA :
- une clé de 128 bits appelée *Media Key* générée aléatoirement par l'AACS LA qui est conservée secrète par le diffuseur et n'apparaîtra notamment pas en clair sur le disque.
- le bloc de clé de média (*Media Key Block* ou *MKB*) associé qui est inscrit sur le disque pour permettre aux lecteurs autorisés d'obtenir la *Media Key*. Le fonctionnement du *MKB* est détaillé dans la section 5.

L'intervention de l'AACS LA dans le processus a plusieurs buts. Elle permet tout d'abord de bénéficier de la gestion centralisée par l'AACS LA des appareils compatibles et non-révoqués. Elle permet aussi de s'assurer qu'un réplicateur créant un disque AACS est lié contractuellement à l'AACS LA, et donc qu'il se conformera par contrat aux règles de bonnes conduites définies par l'AACS LA. En effet, il sera impossible à un réplicateur de créer par lui même un *MKB* et donc de créer un disque compatible AACS contenant par exemple des contenus pirates, et ceci même s'il possède un outil de réplication pouvant écrire dans la zone non-accessible aux graveurs grand-public. Remarquons toutefois que rien n'empêche de réutiliser une *Media Key* et un *MKB* déjà utilisés pour un autre contenu.

3.4 Contrôle fin des usages

À chaque disque protégé par AACS est associé un ensemble de règles décrivant les usages autorisés pour les contenus stockés. Ces règles traitent par exemple de l'extraction du contenu pour des appareils portables, de la résolution maximale pour la restitution du contenu si l'affichage n'est pas compatible HDCP, etc.

Ces règles d'usage sont signées et leur résumé cryptographique calculé par le réplicateur.

3.5 Chiffrement de la ou des *title key(s)* et protection du disque

À cet instant, le réplicateur dispose de toutes les données cryptographiques nécessaires pour terminer la conception du disque. Rappelons que le but est de contrôler l'accès à la *title key* pour que seuls les appareils compatibles et non-révoqués puissent lire des disques originaux en respectant les règles d'usage. Ainsi, le réplicateur calcule le haché cryptographique de la *media key* combinée avec les hachés cryptographiques des règles d'usage signées et le *Volume ID*. Le résultat est la Clé Unique de Volume (*Volume Unique Key*) utilisée pour chiffrer les *title keys*.

Le réplicateur conçoit finalement un *master* du disque qui contiendra, en plus bien sûr du ou des titres chiffrés, toutes les informations nécessaires pour déchiffrer la ou les *title key(s)* (et subséquemment le(s) titre(s) associé(s)), à savoir :
- le *Volume ID*,
- les règles d'usage signées,
- le *Media Key Block* contenant la *media key*,
- la ou les *title key(s)* chiffrée(s).

Dans la section suivante, nous présentons comment ces informations seront utilisées par le lecteur pour accéder au contenu.

4 Lecture d'un DVD protégé par AACS

Le déchiffrement d'un disque protégé par AACS se fait en plusieurs étapes, chacune étant dédiée à un objectif de sécurité. À chaque étape, le lecteur obtient une ou plusieurs clés qui ne peuvent *a priori* être obtenues que si l'objectif de sécurité associé est atteint.

Remarquons que dans le cas des lecteurs sur PC, le standard AACS requiert que la communication entre le *rack* optique AACS et le logiciel de rendu soit sécurisée de manière spécifique. En effet, contrairement aux lecteurs de salon, les PC sont des machines multi-fonctions et ne peuvent donc pas être considérés comme des environnements dignes de confiance[2]. Ainsi, le logiciel et le *rack* doivent s'authentifier mutuellement grâce

[2]Les attaques actuelles sont d'ailleurs majoritairement réalisées sur des PCs.

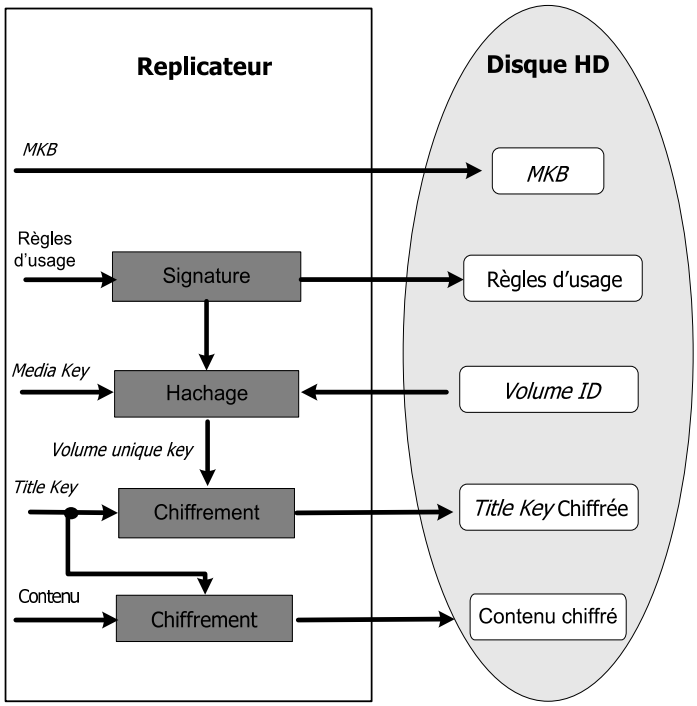


FIG. 1 – Organisation d'un disque protégé par AACS

à un mécanisme de certificats/listes de révocation et mettre en place une clé partagée pour communiquer de manière sécurisée.

4.1 Recupération du *Volume ID*

Le lecteur obtient tout d'abord le *Volume ID* du disque. Comme indiqué précédemment, celui-ci est en partie placé sur le disque dans une zone qui n'a pu être inscrite que par du matériel professionnel, ce qui empêche donc la copie bit-à-bit sur des graveurs grand-public.

4.2 Extraction de la *Media Key* du *Media Key Block*

Chaque lecteur compatible AACS dispose d'un ensemble de clés d'appareil (*Device Keys*) dont il se sert pour obtenir la *Media Key* du *Media Key Block*. Cet ensemble de *Device Keys* peut appartenir à un seul appareil ou être partagé par plusieurs d'entre eux. La manière dont le *MKB* est conçu et dont les *Device Keys* sont distribuées sera décrite dans la section 5. Considérons pour l'instant ce système comme une boîte noire qui permet à l'AACS LA de gérer la révocation d'appareils. Ainsi, si un appareil est compatible et n'a pas été révoqué, il dispose d'au moins une *Device Key* permettant de déchiffrer la *Media Key* incluse dans le *MKB*. Si, par contre, un appareil a été révoqué, ses *Device Keys* ne lui permettront pas d'obtenir cette *Media Key*.

4.3 Vérification des règles d'usage

Le lecteur accède ensuite aux règles d'usage. Si leur signature est authentique, le lecteur en calcule le résumé cryptographique. Sinon il arrête le processus de déchiffrement. Ainsi, le disque doit présenter les règles d'usage originales, faute de quoi la signature ne pourra être vérifiée et/ou le résumé cryptographique ne permettra pas de calculer la ou les bonnes *title key(s)*.

4.4 Déchiffrement de la ou des *title key(s)*

Une fois que le lecteur dispose du *Volume ID*, des règles d'usage et de la *Media Key*, il est capable d'obtenir la ou les *title key(s)*. Ainsi, il peut accéder au contenu. La figure 2 résume l'ensemble de ces opérations.

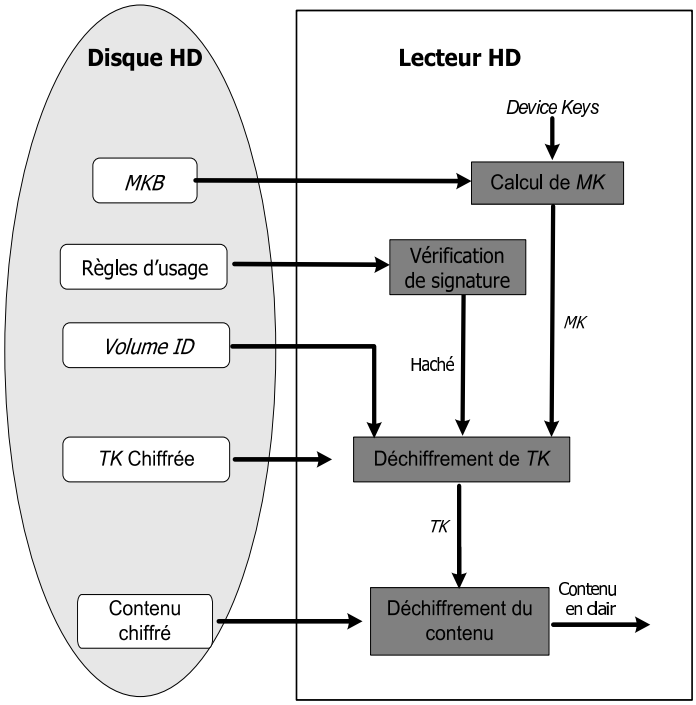


FIG. 2 – Processus de déchiffrement d'un disque protégé par AACS

Après avoir présenté l'accès conditionnel par le chiffrement des *title keys* dans cette section, nous présentons dans les sections suivantes la révocation d'appareils et le traçage des traîtres tels qu'AACS les définit.

5 *MKB* et révocation d'appareils

La sécurité d'AACS repose presque entièrement sur le *MKB*. En se basant sur les principes de la diffusion chiffrée (*Broadcast Encryption*) [6], celui-ci permet de restreindre l'accès au contenu aux seuls appareils compatibles, tout comme il permet la révocation d'appareils.

5.1 *Broadcast Encryption*

Dans les mécanismes de *broadcast encryption*, les contenus chiffrés, identiques pour tous les destinataires, ne sont déchiffrables que par les appareils qui ont préalablement reçu les clés nécessaires et qui n'ont pas été révoqués.

Une hypothèse importante dans le contexte de la *broadcast encryption* est que les clés sont distribuées une fois pour toutes, et qu'on ne peut pas redistribuer de nouvelles clés dans les anciens appareils par la suite. En accord avec cette hypothèse, l'idée générale de la *broadcast encryption* est que chaque appareil ne reçoit pas strictement les mêmes clés de déchiffrement que les autres. Généralement, les appareils sont organisés en groupes. Chaque appareil appartient à plusieurs groupes, et à chaque groupe correspond une clé. Seuls les appareils qui appartiennent à un groupe connaissent la clé de ce groupe.

AACS utilise la méthode par "différence de sous-ensembles" (*subset difference*) [11]. Selon cette méthode, les appareils sont organisés au sein d'un arbre binaire, chaque appareil étant une feuille de cet arbre. Les

sous-ensembles[3] sont définis à partir d'un nœud interne donné : deux appareils appartiennent au même sous-ensemble dit "sous-ensemble des descendants de X" si tous deux descendent du nœud X dans l'arbre binaire.

Pour illustrer notre propos, considérons la figure 3. Il y a 16 feuilles (le système considéré peut donc avoir au maximum 16 appareils différents, par contraste avec le système AACS qui peut en avoir 4 millards), de a à p. Les appareils i, j, k, l, m, n, o et p par exemple appartiennent au même sous-ensemble des descendants de 3, mais aussi au sous-ensemble des descendants de 1.

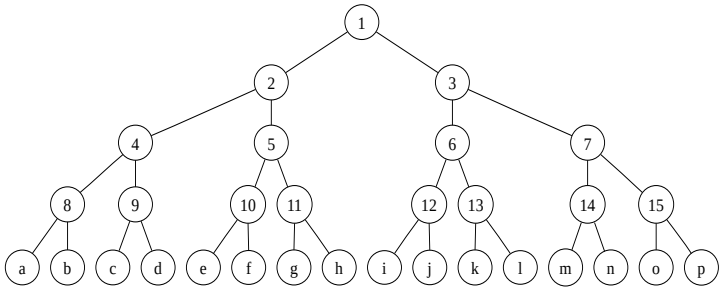


FIG. 3 – Organisation arborescente des 16 appareils de a à p

La méthode de *subset difference* va créer les groupes d'appareils de la manière suivante : pour chacun des sous-ensembles (en d'autres termes, des nœuds non-feuille X), elle crée un groupe pour chacun de ses descendants Y tel que sont membres du groupe tous les appareils qui sont descendants de X et *ne* sont *pas* descendants de Y. Ainsi, chaque groupe est la différence entre deux sous-ensembles. Un groupe ainsi formé se note $S(X, Y)$, qui peut se lire "groupe des descendants de X ôté des descendants de Y". Par exemple (cf. figure 4), $S(1, 5)$ est l'ensemble des appareils qui descendent du nœud *1* mais pas du nœud *5*, à savoir tous les appareils sauf e, f, g et h. À chaque groupe d'appareils correspondra une clé qui sera connue de tous les membres de ce groupe.

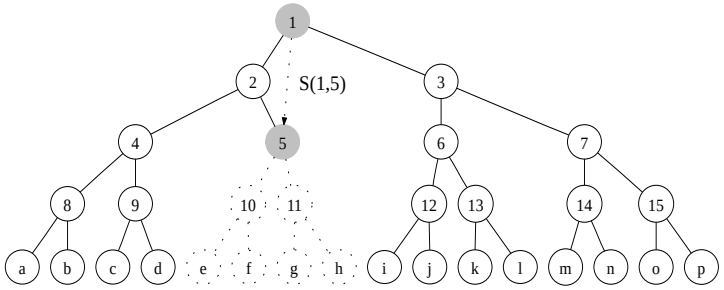


FIG. 4 – Exemple : le groupe $S(1, 5)$

La mise en œuvre de la méthode de *subset difference* dans AACS est constitué de 4 étapes :
- la construction des arbres de clés par l'AACS LA ;
- la distribution des clés aux appareils ;
- le choix des groupes autorisés et la création du *MKB* ;
- le déchiffrement du *MKB* par les lecteurs autorisés.

Ces étapes sont décrites dans les sections suivantes.

5.2 Contruction des arbres de clés

L'opération initiale de la *broadcast encryption* par *subset difference* est la construction des arbres de clés. Dans le cas d'AACS, cette opération a été réalisée une fois pour toutes par l'AACS LA.

[3]Soulignons que les sous-ensembles considérés ici servent à la construction des groupes, mais ne sont pas les groupes eux-mêmes.

En suivant les principes de la *subset difference*, la clé associée à chaque groupe est calculée de la manière suivante. Tout d'abord, pour chacun des nœuds non-feuilles X, une graine $Seed_X$ est tirée aléatoirement. Cette graine sera utilisée pour calculer toutes les clés des groupes $S(X, Y)$ pour tous les Y descendant de X. Ainsi, il y aura autant d'arborescences de clés calculées que de nœuds internes.

L'arborescence des clés est calculée de la même manière pour chacune des graines. Sans perte de généralité, nous présenterons la génération des clés à partir de $Seed_1$. Tout d'abord, le haché cryptographique $Label_1$ de la graine $Seed_1$ est calculé grâce à la fonction de hachage AES-G3[4]. $Label_1$ est ensuite divisé en trois parties égales de 128 bits chacune : $Gauche(1)$, 128 bits qui ne seront plus utilisés et $Droite(1)$.

$Gauche(1)$ est utilisée comme entrée de la fonction AES-G3 pour calculer le *label* du nœud fils de gauche $Label_{1,2}$. Celui-ci fait 384 bits, séparable en trois parties : $Gauche(1, 2)$, $ProcKey(S(1, 2))$ (la clé du groupe - aussi appelée *processing key* - $S(1, 2)$), et $Droite(1, 2)$. $Droite(1)$ est pour sa part utilisée comme entrée de la fonction AES-G3 pour le *label* $Label_{1,3}$ du nœud fils de droite. $Label_{1,3}$ est aussi séparable en trois parties de 128 bits chacune : $Gauche(1, 3)$, $ProcKey(S(1, 3))$ la *processing key* du groupe $S(1, 3)$, et $Droite(1, 3)$.

Cette méthode est itérée récursivement jusqu'en bas de l'arbre : ainsi, pour chaque nœud Z, on calcule le *label* de son fils de gauche en appliquant AES-G3 à $Gauche(Z)$, et le *label* de son fils de droite en appliquant AES-G3 à $Droite(Z)$. A l'issue de la procédure, à chacun des nœuds Y (descendant du nœud *1*) sera associé $Label_{1,Y}$ dont la partie centrale est la *processing key* du groupe $S(1, Y)$ notée $ProcKey(S(1, Y)$.

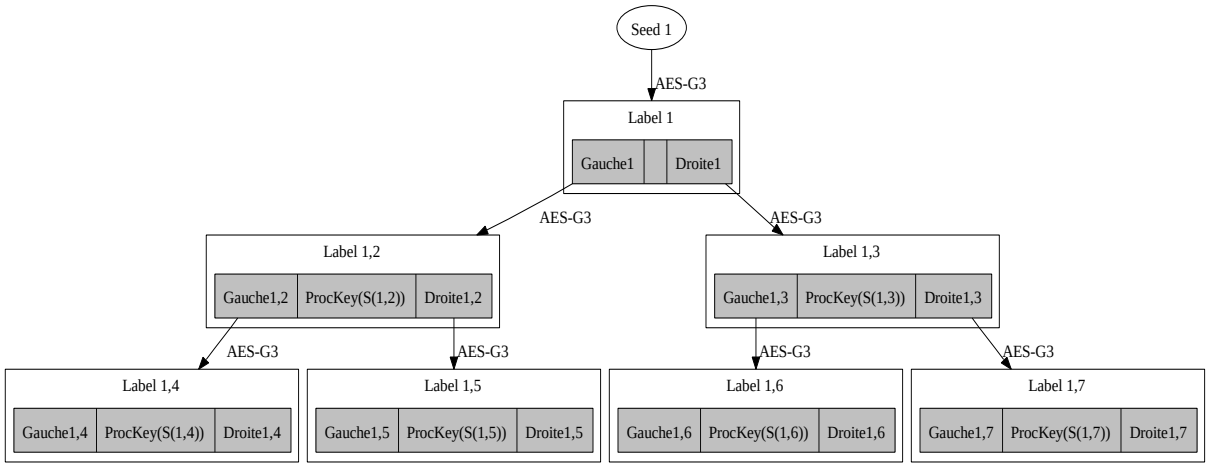


FIG. 5 – Construction de l'arbre de clé pour $S(1, Y)$

Au total, et à la fin de toutes ces opérations pour chaque $Seed_X$, nous disposons d'autant d'arborescences de *Labels* que de nœuds internes. L'une d'entre elles (celle des groupes $S(1, Y)$) a une profondeur égale à celle de l'arbre des appareils, deux d'entre elles (celles des groupes $S(2, Y)$ et $S(3, Y)$) ont une profondeur égale celle de l'arbre des appareils moins 1, et ainsi de suite.

5.3 Distribution des clés aux appareils

L'AACS LA distribue sous licence les clés d'appareils aux fabricants de lecteurs. Chaque lecteur se voit attribué les *Labels* des groupes auxquels il appartient. L'algorithme de distribution des clés pour l'appareil α est le suivant : tout d'abord, on ne considère que les nœuds situés sur le chemin entre l'appareil et la racine X dans l'arbre des appareils (et donc dont α est descendant). Puis, on parcourt chacun de ces nœuds X, en ajoutant à chaque fois les $Label_{X,Y}$ des nœuds fils directs Y qui ne sont pas dans le chemin, autrement dit les clés des groupe d'appareils qui n'excluent pas α et dont on peut déduire toutes les autres clés. Remarquons qu'étant donné que certains *Labels* se déduisent d'autres grâce à la fonction AES-G3, les appareils n'en recevront que le nombre minimum nécessaire. En effet, la connaissance de $Label_{X,Y}$ permet grâce à la fonction AES-G3 de générer tous les autres $Label_{X,Z}$ lorsque le nœud Z descend du nœud Y.

[4] AES-G3 est une fonction de hachage cryptographique basée sur AES qui pour toute entrée de 128 bits retourne $3 \cdot 128$ bits, soit 384 bits.

Pour illustrer ce mécanisme, prenons l'exemple de l'appareil g (cf. Fig. 3). Dans l'arbre des clés ayant 1 pour racine, g est situé sur le chemin nœud *1*, nœud *2*, nœud *5*, nœud *11*. Lors de la descente de l'arbre, g reçoit les *Labels* des nœud fils directs $Y \in \{3, 4, 10, \text{h}\}$ qui ne sont pas sur le chemin allant de 1 à lui, à savoir :

- $Label_{1,3}$, associé au groupe $S(1,3)$ auquel il appartient, étant descendant de 1 et pas de 3, et complémentaire de $S(1,2)$;
- $Label_{1,4}$, associé au groupe $S(1,4)$ auquel il appartient, étant descendant de 1 et pas de 4, et complémentaire de $S(1,5)$;
- $Label_{1,10}$, associé au groupe $S(1,10)$ auquel il appartient, étant descendant de 1 et pas de 10, et complémentaire de $S(1,11)$;
- $Label_{1,h}$, associé au groupe $S(1,h)$, complément de $S(1,g)$.

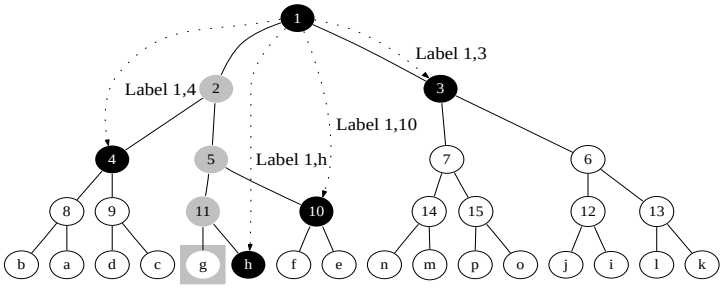


FIG. 6 – Distribution des clés pour le nœud *1*

Comme nous l'avons indiqué plus haut, en disposant du $Label_{1,3}$, l'appareil g est capable de dériver grâce à AES-G3 l'ensemble des *labels* associés à $S(1,6)$, $S(1,7)$, $S(1,12)$, $S(1,13)$, $S(1,14)$, $S(1,15)$, S(1,i), S(1,j), S(1,k), S(1,l), S(1,m), S(1,n), S(1,o), et S(1,p). En dérivant le $Label_{1,4}$, il est capable d'obtenir $Label_{1,8}$, $Label_{1,9}$, $Label_{1,a}$, $Label_{1,b}$, $Label_{1,c}$ et $Label_{1,d}$. Enfin, grâce à $Label_{1,10}$, il peut dériver les *labels* associés à S(1,e) et S(1,f).

En utilisant l'arbre des clés ayant pour racine 2 de la même manière (cf. Fig 7), g reçoit les *labels* des nœuds fils directs $Y \in \{4, 10, \text{h}\}$ qui ne sont pas sur le chemin, à savoir :

- $Label_{2,4}$;
- $Label_{2,10}$;
- $Label_{2,h}$.

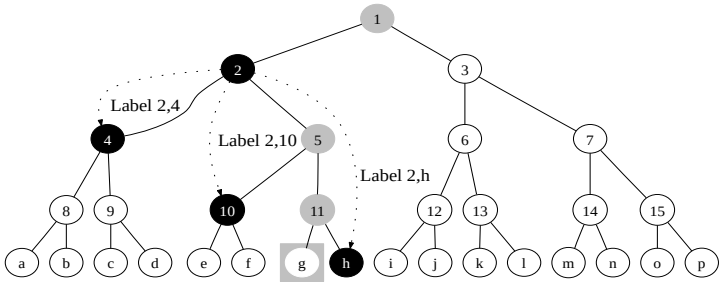


FIG. 7 – Distribution des clés pour le nœud *2*

Puis g reçoit de l'arbre de clés ayant le nœud *5* pour racine $Label_{5,10}$ et $Label_{5,h}$, et enfin $Label_{11,h}$ pour de l'arbre de clés ayant le nœud *11* pour racine.

5.4 Choix des groupes autorisés et création du *MKB*

Pour calculer le *MKB*, l'AACS LA détermine tout d'abord la liste des lecteurs autorisés à accéder au contenu (par construction, il s'agit de l'ensemble des lecteurs AACS ôté des lecteurs révoqués). Une fois cet

ensemble déterminé, l'AACS LA choisit l'ensemble minimal de *processing keys* tel que :

- chaque lecteur non-revoqué dispose d'au moins une de ces clés ou puisse la calculer en appliquant AES-G3 aux *labels* dont il dispose ;
- chaque lecteur révoqué n'en connaisse aucune et ne puisse en calculer à partir des labels dont il dispose.

Une fois les groupes d'appareils choisis, l'AACS LA chiffre la *Media Key* avec chacune des *processing keys* de ces groupes. Le *MKB* est obtenu en regroupant les différents chiffrements de la *Media Key*.

Considérons tout d'abord le cas où aucun appareil n'est révoqué. Dans ce cas, l'AACS LA peut par exemple choisir les deux groupes $S(1,2)$ et $S(1,3)$. En effet, tous les appareils appartiennent à l'un de ces deux groupes, et chacun d'entre eux dispose donc soit de la clé *ProcKey(*$S(1,2)$*)* dans le $Label_{1,2}$, soit de la clé *ProcKey(*$S(1,3)$*)* dans le $Label_{1,3}$. Le *MKB* produit, destiné à être déchiffré par tous les appareils contiendra donc deux entrées : la *Media Key* chiffrée avec *ProcKey(*$S(1,2)$*)* et la *Media Key* chiffrée avec *ProcKey(*$S(1,3)$*)*.

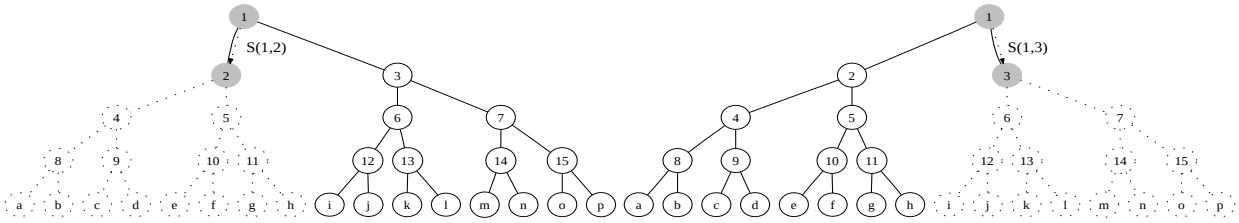


FIG. 8 – Groupes lorsque aucun appareil n'est révoqué

Supposons maintenant le cas où g et h sont révoqués. L'AACS LA doit donc choisir les groupes de telle manière que tous les autres appareils connaissent ou puissent calculer une *processing key* utile à partir des *labels* qu'ils ont déjà et de la fonction AES-G3, mais que g et h en soient incapables. Une solution très simple est de chiffrer la *Media Key* par la *processing key* associée à l'unique groupe $S(1,11)$, *ProcKey(*$S(1,11)$*)* (cf. Fig. 9).

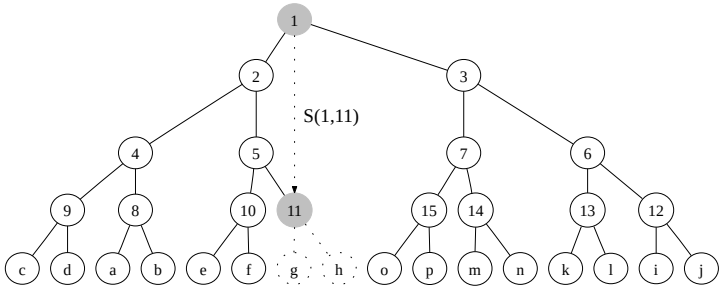


FIG. 9 – Groupe unique choisi lorsque g et h sont révoqués

Mais l'AACS LA peut aussi par exemple (cf. Fig. 10)choisir de chiffrer la *Media Key* pour les groupes $S(1,2)$ (incluant tous les appareils de i à p), $S(2,5)$ (incluant tous les appareils de a à d), et $S(5,11)$ (incluant e et f). Le *MKB* produit contient donc 3 entrées : l'une est le chiffrement de la *Media Key* avec *ProcKey(*$S(1,2)$*)*, une autre le chiffrement de la *Media Key* utilisant *ProcKey(*$S(2,5)$*)*, et la troisième utilisant *ProcKey(*$S(5,11)$*)*.

Remarquons qu'avec les *labels* en leur possession, g et h sont incapables d'obtenir ne serait-ce qu'une des clés permettant de déchiffrer la *Media Key*. Par exemple g a en sa possession $Label_{1,3}$, $Label_{1,4}$, $Label_{1,10}$, $Label_{1,h}$, $Label_{2,4}$, $Label_{2,10}$, $Label_{2,h}$, $Label_{5,10}$, $Label_{5,h}$ et $Label_{11,h}$. On vérifie aisément qu'aucun d'eux ne permet d'obtenir une clé valide.

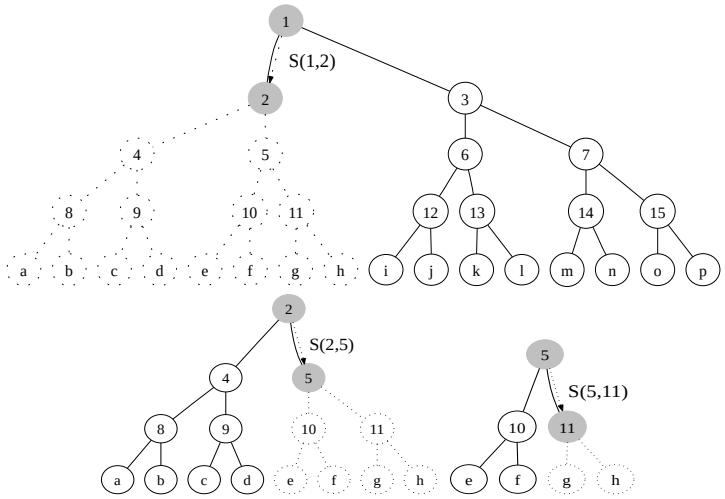


FIG. 10 – Groupes choisis lorsque g et h sont révoqués

5.5 Mise en œuvre de la *broadcast encryption* sur un disque AACS

Après avoir présenté les principes le régissant, considérons maintenant l'organisation du *MKB*. Celui-ci est formé de cinq champs. Deux d'entre eux sont dédiés à la révocation explicite des appareils :

- La liste de révocation des hôtes (*Host Revocation List*), qui est utilisée par le *rack optique* et contient la liste des logiciels révoqués ;
- La liste de révocation des *racks* optiques (*Drive Revocation List*), qui est utilisée par le logiciel et contient la liste des *racks* révoqués.

AACS prévoit donc la possibilité de révoquer indépendamment les *racks* et les logiciels corrompus. Comme expliqué précédemment, un PC est considéré dans AACS comme deux modules séparés (*rack* et logiciel). Dans ce type de configurations, il est possible qu'un seul des modules présente une faille de sécurité, mais que celle-ci mette en péril tout le système de protection. Ainsi, si le logiciel (respectivement, le *rack*) est dans la liste de révocation reçue par le *rack* (respectivement, le logiciel), celui-ci refusera de communiquer avec lui.

Les trois autres champs sont directement dédiés à la *broadcast encryption* :

- Les chiffrements de la *Media Key* (*Media Key Data*), qui consistent en plusieurs enregistrements, chacun contenant la *Media Key* chiffrée avec une clé de groupe différente, selon la méthode décrite dans cette section ;
- Les données de différence explicite de sous-ensembles (*Explicit Subset Difference*), qui indiquent au lecteur les groupes correspondant et lui permettent de choisir correctement quel enregistrement utiliser dans le champ *Media Key Data* ;
- L'information de vérification de *Media Key* (*Verify Media Key*), qui permet au lecteur de vérifier qu'il a correctement déchiffré la *Media Key*.

Ainsi, un lecteur compatible AACS va tout d'abord chercher dans le champ *Explicit Subset Difference* un groupe auquel il appartient et pour lequel il possède ou peut calculer le *label*. À partir de cette information, il déchiffre l'enregistrement adéquat du champ *Media Key Data* avec la *Processing Key* qu'il détient ou qu'il a calculé. Il dispose ainsi de la *Media Key*. Enfin, il déchiffre le contenu du champ *Verify Media Key* avec la *Media Key* obtenue. Si les 8 premiers octets sont `0x0123456789ABCDEF`, il considère qu'il a obtenu la bonne *Media Key* et poursuit l'opération d'obtention des *Title Keys* comme indiqué dans la section 4.

6 Bloc de clés de séquence et traçage des traîtres

Avant de procéder à la révocation d'appareils, il faut tout d'abord identifier les appareils qui ont permis la fuite des contenus. À cette fin, le standard AACS propose un mécanisme de traçage des traîtres (*traitor tracing*) [5] basé sur la méthode des variantes. Remarquons qu'à notre connaissance, aucun disque ne met

en œuvre le traçage des traitres au moment de la rédaction de cet article.

6.1 Le traçage des traîtres par la méthode des variantes

Le traçage des traîtres par la méthode des variantes est basé sur le principe de la diversification de séquences : au sein d'un contenu donné (et donc sur un disque), plusieurs séquences (d'une durée de 2 à 5 secondes) existent sous plusieurs variantes. Ces variantes d'une même séquence paraissent visuellement identiques mais sont tatouées de manière différente[5]. Par exemple, sur la figure 11, deux séquences (2 et 4) existent sous la forme de 6 variantes chacune, permettant un total de $6 \cdot 6 = 36$ versions différentes du contenu.

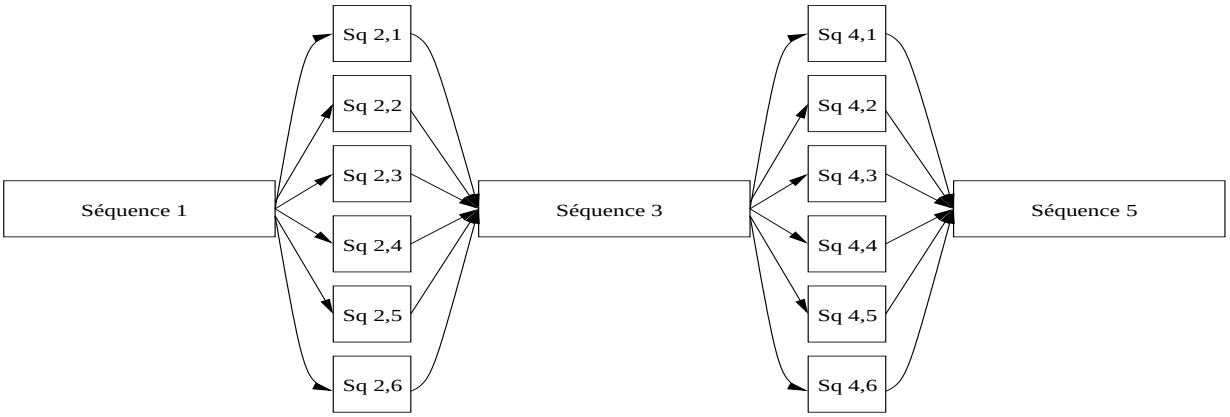


FIG. 11 – Un contenu fait de 5 séquences dont 2 sont diversifiées

En outre, chaque variante d'une même séquence est chiffrée en utilisant une clé différente. Les clés ont été préalablement distribuées aux lecteurs de telle manière que chacun d'entre eux ne puisse déchiffrer et donc produire qu'une variante de chaque séquence. Ainsi, pour chaque contenu, un lecteur restitue une unique version qui le caractérise car elle contient les variantes auxquelles ce lecteur a pu accéder.

Si aucune contre-mesure n'est mise en œuvre, la méthode des variantes est vulnérable aux attaques par coalition : plusieurs attaquants mettent en commun des variantes de leurs versions respectives afin de créer une nouvelle version différente qui ne permet *a priori* plus de remonter jusqu'à eux. L'utilisation d'un code correcteur d'erreur sur le choix des variantes codant les versions permet dans une certaine mesure de contrer les attaques par coalition. De cette manière, il existe pour un lecteur donné un ensemble de versions différentes de films qui incrimineront ce lecteur, et ce même si certaines variantes n'ont pas été fournies par le lecteur en question.

6.2 Compromis dans le cadre d'AACS

L'environnement d'utilisation du traçage de traîtres dans AACS est très exigeant. En effet, la méthode choisie est supposée permettre d'identifier un lecteur compromis parmi l'ensemble des lecteurs compatibles AACS (et donc permettre qu'un disque génère un nombre très important de versions, se comptant éventuellement en milliards si l'on considère les hypothèses de déploiements futurs). Comme nous venons de le voir, elle doit aussi être résistante aux attaques par coalition. Si l'utilisation d'un code correcteur d'erreur rend la détection plus fiable, il demande aussi que plus de variantes soient utilisées pour un même nombre d'appareils déployés. Or, la place disponible sur un disque est bornée. Les concepteurs de l'AACS ont donc du prendre en compte l'augmentation de la taille globale du contenu induite par l'ajout des variantes. Il a été admis que 10% de l'espace disque total pouvait être alloué aux variantes.

De par ces contraintes, il n'était plus possible d'identifier le ou les appareils ayant fait fuir un contenu retrouvé à partir des variantes d'un seul disque. Les concepteurs d'AACS ont donc consenti un compromis :

[5]Remarquons que la méthode des variantes fait l'hypothèse d'un tatouage résistant. Celui-ci ne peut notamment pas être retiré, et le fait de disposer de deux variantes de la même séquence tatouées différemment ne permet pas d'en construire une troisième.

ils ont opté pour un traçage des traîtres à deux niveaux, utilisant un code "interne" (*inner code*), et un code "externe" (*outer code*), présentés dans la section suivante.

6.3 Versions de contenus et code interne

Le code "interne" porte sur les versions qui peuvent être obtenues à partir des variantes du même contenu. À notre connaissance, ni le nombre de séquences qui seront diversifiées, ni le nombre de variantes pour chaque séquence n'a été officiellement publié. De plus, comme nous l'avons déjà indiqué, le traçage des traîtres n'est à notre connaissance actuellement pas mis en œuvre. Nous ne pouvons donc pas fournir de manière certaine les valeurs numériques. Cependant, le rapport de recherche [9] mentionne la diversification de 15 séquences selon 16 variantes. Pour résister aux attaques par coalition, un code de Reed-Solomon est utilisé, conduisant à un total de 256 versions par disque. Ainsi, un lecteur restituera une version parmi 256 possibles pour un contenu donné.

6.4 *Sequence key block* et code externe

La diversification d'un contenu donné en un maximum de 256 versions étant largement insuffisant pour le déploiement prévu, AACS met en œuvre un second niveau de code, le code "externe", qui permet d'incriminer avec certitude un traître à partir d'un ensemble de contenus pirates qu'il a produit.

Pour ce faire, les contenus sont répartis en 256 catégories, chaque contenu appartenant à une catégorie donnée. Pour chaque catégorie, l'AACS LA a généré un ensemble appelé colonne (*columns*) de clés appelées clés de séquence (*Sequence Keys*). Il existe donc 256 colonnes de 256 *Sequence Keys* chacune.

Lors de sa fabrication, chaque lecteur reçoit une clé de chaque colonne, dont il se sert pour obtenir les variantes des contenus appartenant à la catégorie correspondante. Il dispose donc au total d'un ensemble de 256 *Sequence Keys* qui le caractérise de manière unique. Il est en effet très peu probable ($p = 1/256^{256}$) que deux lecteurs possèdent exactement le même ensemble de clés.

À l'échelle d'un contenu, les *Sequence Keys* permettent d'attribuer une version de ce contenu à un lecteur donné. Chaque disque permettant le traçage des traîtres possède un bloc de clés de séquences (*Sequence Key Block*). Ce *Sequence Key Block* contient au moins un enregistrement dit "inconditionnel" permettant de calculer les informations utiles à la lecture des variantes (*Calculate Variant Data Record* ou *CVDR*). Il comporte notamment les champs suivants :

- L'identifiant de la colonne de clés à utiliser (parmi les 256 possibles) pour déchiffrer les informations dans l'enregistrement.
- Autant d'enregistrements chiffrés de données de variantes (*Encrypted Variant Data* ou *EVD*) que de clés existant dans la colonne. Chaque enregistrement contient les informations nécessaires au déchiffrement de toutes les variantes de la version assignée (parmi les 256 possibles), elles-même chiffrées grâce à une des *Sequence Keys* de la colonne.

Ainsi, lorsqu'un lecteur cherche à obtenir ses variantes et à les déchiffrer, il lit le *CVDR* et récupère l'identifiant de la colonne à utiliser. Puis il se sert de l'unique clé dont il dispose pour cette colonne afin de déchiffrer l'enregistrement adéquat, et subséquemment déchiffrer correctement les variantes de séquences qui lui sont affectées. Insistons sur le fait que les informations sur les variantes étant différentes pour chaque enregistrement (et donc pour chaque clé de la colonne), chaque lecteur accédera à une version différente du film en fonction de la clé dont il dispose dans la colonne. En pratique, la détection d'un contenu pirate permet donc *a priori* de réduire la population des suspects à 1/256 de la population totale des lecteurs.

L'existence de plusieurs colonnes permet le traçage d'un lecteur à l'origine de plusieurs contenus pirates différents découverts. En effet, tous les contenus n'utilisent pas la même colonne, et la détection de n contenus pirates produits par un même traître et utilisant des colonnes différentes permet d'incriminer un lecteur parmi 256^n. Il faut 4 contenus pirates provenant du même traître et utilisant des colonnes différentes pour pouvoir incriminer un lecteur parmi $4 \cdot 10^9$, qui est l'ordre de grandeur du déploiement envisagé à long terme pour AACS.

6.5 Révocation partielle dans les *Sequence Key Blocks*

En plus de permettre le traçage des traîtres, les *Sequence Key Blocks* permettent aussi une révocation partielle des lecteurs incriminés : lorsqu'une *Sequence Key* a été identifiée par l'AACS LA comme étant

corrompue, son utilisation pour déchiffrer les *EVDs* ne retournera plus les informations permettant d'accéder aux variantes.

Une *Sequence Key* compromise par un pirate étant aussi utilisée par des lecteurs honnêtes, il n'est pas possible de la révoquer purement et simplement. Ainsi, lorsqu'au moins une *Sequence Key* a été compromise dans la colonne utilisée pour un contenu donné, le *Sequence Key Block* du disque contiendra en plus de l'enregistrement "inconditionnel" un ou plusieurs enregistrements "conditionnels". Ces enregistrements "conditionnels" ne seront utilisés que par les lecteurs dont la *Sequence Key* pour la colonne utilisée dans l'enregistrement "inconditionnel" a été révoquée. Ils leur permettront d'utiliser une de leurs *Sequence Keys* non-révoquées pour accéder aux variantes.

La structure des enregistrements "conditionnels" est identique à celle de l'enregistrement "inconditionnel". Toutefois, à la différence des enregistrements"conditionnels", il est nécessaire de s'assurer qu'ils ne seront utilisés que par les lecteurs qui ont échoués dans le déchiffrement de l'enregistrement "inconditionnel" parce que leur clé était revoquée. Ainsi, pour déchiffrer les *EVDs* des enregistrements conditionnels, un lecteur doit non seulement connaître la bonne *Sequence Key*, mais aussi une clé de lien (*link key*) qu'il a obtenu en déchiffrant l'*EVD* de l'enregistrement pour lequel il disposait d'une clé compromise.

La figure 12 présente un exemple de *Sequence Key Block*. Supposons que 'AACS LA ait retrouvé des copies pirates générées par un seul traître qui a utilisé les *Sequence Keys* X5, Y2 et Z3. Ces clés ont donc été révoquées. Parallèlement, l'AACS LA sait qu'il n'existe aucun autre appareil qui possède ce jeu de clé. Le *Sequence Key Block* de la figure 12 a donc pour but de permettre à tous les appareils, à l'exclusion du pirate, d'accéder aux variantes. Ainsi, un appareil qui possède pour au moins une des colonnes X, Y et Z une clé qui n'est pas X5, Y2 ou Z3 doit pouvoir obtenir des informations de variantes.

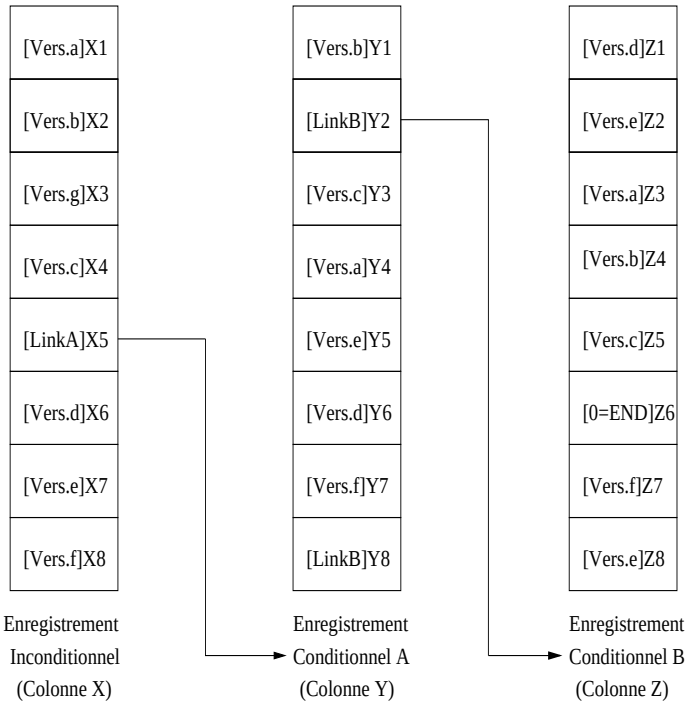


FIG. 12 – Organisation des *Sequence Key Blocks*

Un lecteur tentant d'obtenir les informations de variantes commence par tenter de déchiffrer l'*EVD* de l'enregistrement "inconditionnel" qui correspond à la clé qu'il possède dans la colonne X. S'il connaît X1, X2, X3, X4, X6, X7 ou X8, il obtient directement les informations codant les versions associées . S'il connaît X5, le résultat du déchiffrement lui indique d'accéder à l'enregistrement "conditionnel" A avec la *link key* A.

L'accès aux enregistrements conditionnels se fait de la même manière, à l'exception du fait que le lecteur doit tenter de déchiffrer l'*EVD* à la fois avec la *link key* associée et avec la *Sequence Key* dont il dispose pour la colonne. Par conséquent, si le lecteur connaît X5 et l'une des clés Y1, Y3, Y4, Y5, Y6, Y7, Y8, il pourra accéder aux informations de variantes. S'il connaît X5 et Y2, le résultat du déchiffrement de l'*EVD*

avec Y2 et la *link key* obtenue du déchiffrement de l'*EVD* par X5 lui indique d'accéder à l'enregistrement conditionnel B avec la *link key* B.

L'appareil possédant la clé Z6 pour la colonne Z (et, puisqu'il en est arrivé à cet enregistrement "conditionnel", les clés X5 et Y2 respectivement pour les colonnes X et Y) obtiendra par le déchiffrement de son *EVD* une information spéciale lui indiquant qu'il a été révoqué. Les autres obtiendront leurs informations de variantes.

7 Attaques et perspectives

Dans cet article, nous avons présenté AACS. Ce système semble, au moins d'un point de vue théorique, proposer plusieurs méthodes efficaces pour protéger les contenus distribués sur supports optiques pré-enregistrés.

Pourtant, quelques mois à peine après la sortie des premiers disques compatibles, les premières attaques réussies furent annoncées. Muslix64 lança l'offensive en annoncant sur le forum *doom 9* [7] la disponibilité d'un logiciel permettant, sous reserve que l'utilisateur fournisse la *Title Key*, de déchiffrer les titres d'un HD-DVD ou d'un Blu-Ray. Ce logiciel était principalement une mise en œuvre du déchiffrement des titres conforme aux spécifications AACS. Il représentait *a priori* une faible menace étant donné que ces spécifications sont publiques et que l'obtention d'une *Title Key* est supposée très difficile.

Peu de temps après pourtant, Muslix64 annonça avoir trouvé une *Title Key* par analyse de la mémoire lors de la lecture d'un HD-DVD par lecteur compatible. Il prouva cette annonce en diffusant cette clé. Plusieurs personnes se mirent alors à diffuser les *Title Key* qu'ils étaient parvenus à extraire.

Après ce premier succès, d'autres sur le forum s'entendirent sur le fait qu'il était nécessaire (et probablement possible en utilisant les mêmes méthodes d'analyse de la mémoire) de trouver d'autres clés "plus importantes" dans la hierarchie de clés AACS. Le 11 février 2007, Arnezami annonça avoir trouvé une *processing key*, ouvrant la voie à un déchiffrement générique des disques. En effet, dans la mise en œuvre actuelle, tous les disques d'une même génération utilisent la même MKB. Par conséquent, la découverte d'une *processing* permettant de déchiffrer un disque d'une génération donnée permet en fait de déchiffrer tous les disques de la même génération.

Cette découverte, bien qu'importante, n'était cependant pas suffisante. En effet, pour obtenir les *Title Keys*, il faut aussi disposer du *Volume ID* (cf. section 3). Rapidement, plusieurs méthodes sont apparues. La première tirait partie de la non-mise en œuvre du chiffrement des communications entre certains racks optiques et le logiciel, pourtant requise par la norme. Ainsi, il suffisait de *sniffer* les communications USB pour obtenir le *Volume ID* en clair. La deuxième reposait sur une modification du *firmware* du lecteur HD-DVD de la X-Box 360, qui permettait au logiciel de demander le Volume ID sans authentification. La troisième enfin consistait en l'utilisation d'une clé privée de logiciel compromise (encore une fois par analyse de la mémoire) pour réaliser l'authentification et subséquement obtenir le *Volume ID*.

Remarquons tout d'abord que toutes ces attaques sont des attaques d'implémentations et ne sont pas dues à des failles de l'architecture AACS en tant que telles. Ceci étant, elles démontrent une fois de plus que la mise en œuvre est un point très important de la sécurité, et qu'une architecture qui semble très sûre peut être contournée si son implémentation est partielle ou défaillante.

Pour répondre aux attaques, L'AACS LA a d'ailleurs mis en œuvre les mécanismes de révocation dont elle dispose. Tout d'abord, une seconde MKB n'utilisant plus la clé d'appareil compromise a été diffusée aux réplicateurs, conduisant à la seconde génération de disques, en activité au moment de la rédaction de ces lignes. Pour répondre à la fuite des *Volume ID*, les racks optiques et les lecteurs ont été mis à jour et la clé privée corrompue a été révoquée.

Malgré cela, une nouvelle *processing key* utilisée dans les nouvelles MKB a été découverte. Ainsi, il semblerait que l'AACS LA soit entrée avec les *hackers* dans un jeu du chat et de la souris dont l'issue n'est cependant pas si évidente. En effet, il reste aujourd'hui de nombreux mécanismes de protection qui n'ont pour l'instant pas été mis en œuvre, tel que les *Sequence Keys* par exemple.

Références

[1] Advanced Access Content System (AACS), Technical Overview, Juillet 2004.

[2] Advanced Access Content System (AACS), HD DVD and DVD Pre-recorded Book v 0.93, Aout 2007.

[3] Advanced Access Content System (AACS), Introduction and Common Cryptographic Elements v 0.93, Aout 2007.

[4] Advanced Access Content System (AACS), Pre-recorded Video Book v 0.93, Aout 2007.

[5] B. Chor, A. Fiat, and M. Naor. Tracing traitors. *Lecture Notes in Computer Science*, 839 :257–270, 1994.

[6] A. Fiat and M. Naor. Broadcast encryption. *Lecture Notes in Computer Science*, 773, 1994.

[7] Forums Doom 9. http ://forum.doom9.org/, 2006-2007.

[8] H. Jin and J. Lotspiech. Renewable traitor tracing : A broadcast, tracing and revoke system for anonymous attacks. Technical report, IBM, 2006.

[9] H. Jin, J. Lotspiech, and N. Megiddo. Efficient traitor tracing. Technical report, IBM, 2006.

[10] H. Jin, J. Lotspiech, and S. Nusser. Traitor tracing for prerecorded and recordable media. In *DRM '04 : Proceedings of the 4th ACM workshop on Digital rights management*, pages 83–90, New York, NY, USA, 2004. ACM Press.

[11] D. Naor, M. Naor, and J. Lotspiech. Revocation and tracing schemes for stateless receivers. *Lecture Notes in Computer Science*, 2139, 2001.

Protection des logiciels contre la rétro-ingénierie

Sébastien Josse*, Guillaume Dabosville
Silicomp-AQL
{sebastien.josse,guillaume.dabosville}@aql.fr

17 octobre 2007

Résumé *La conception et l'implémentation de logiciels résistants à la rétro-ingénierie est un problème à la fois crucial pour de nombreuses applications et présentant de nombreuses difficultés, tout particulièrement lorsque la possibilité d'utilisation d'un composant matériel de sécurité est exclue.*
L'objectif de ce papier est tout d'abord de présenter quelques unes des briques fondamentales (cryptographiques[1] et non-cryptographiques) constituant un système de protection logicielle et une manière de les agencer, à travers l'étude d'une solution phare du marché civil. Nous présentons ensuite un état de l'art des méthodes de rétro-ingénierie, puis une démarche, des critères et des outils pour évaluer la robustesse d'une solution de protection logicielle.

Abstract *Design and implementation of software that are resilient against reverse engineering is both a crucial and difficult problem, for a lot of applications, particularly when use of hardware components is forbidden.*
At first, this paper presents some of fundamental parts of a software protection system and several ways of assembling them together, trough the survey of some commercial solutions. Next, a state-of-art of reverse engineering methods is presented. At last, criteria and tools to evaluate the robustness of software protection systems are presented.

Mots clés Cryptographie boîte blanche, Rétro-ingénierie, Protection, Mécanismes, Efficacité, Robustesse

*Doctorant au Laboratoire de virologie et cryptologie de l'Ecole Supérieure et d'Application des Transmissions, sebastien.josse@esat.terre.defense.gouv.fr.

[1] Cet exposé concerne pour partie la conception cryptographique, dans un contexte boîte blanche, à travers la présentation de modes de gestion des clés et de primitives de chiffrement adaptés.

1 Introduction

La conception et l'implémentation de logiciels résistants à la rétro-ingénierie est un problème crucial pour de nombreuses applications, particulièrement lorsqu'il s'agit de protéger les algorithmes propriétaires implémentés par le logiciel et/ou de protéger la fonction de contrôle de droits conditionnant l'accès à tout ou partie de ses fonctionnalités.
Lorsque l'application à protéger ne peut pas asseoir sa sécurité sur l'utilisation d'un composant matériel de sécurité, ou d'un serveur du réseau, nous devons faire l'hypothèse d'un attaquant capable d'exécuter l'application dans un environnement qu'il contrôle parfaitement. Le modèle de l'attaquant correspondant à ce contexte, que nous appelerons WBAC (*White Box Attack Context*) dans la suite de ce papier, impose une implémentation logicielle particulière des primitives cryptographiques classiques.
Dans ce contexte, la protection d'un programme repose sur des mécanismes couvrant différents objectifs de sécurité, dont la capacité à contrôler, en plusieurs points de son exécution, l'intégrité de son code, de ses données critiques et de son contexte d'exécution ; la capacité à garantir la confidentialité des algorithmes propriétaires, la diversification de ses instances, l'ancrage du logiciel à une plate-forme cible d'exécution personnalisée, etc.
Le marché civil offre plusieurs systèmes de protection logicielle *sur étagère*, permettant en principe de couvrir ces objectifs de sécurité. Ces systèmes permettent l'application de mécanismes de protection élémentaires au niveau du code source ou au niveau du code exécutable. Dans le premier cas, il s'agit de chaînes de compilation spécialisées (autorisant l'application des protections et des transformations d'obfuscation au niveau du code source) ; dans le second cas, il s'agit d'un loader de protection (qui ne va déchiffrer tout ou partie de l'exécutable qu'après avoir contrôlé les droits conditionnant l'accès à tout ou partie des fonctionnalités de l'exécutable protégé). Nous voyons donc que nos objectifs de sécurité peuvent théoriquement être couverts par l'utilisation conjointe d'une *chaîne de compilation spécialisée* et d'un *loader de protection*.
Cependant, l'utilisation et l'évaluation de ces solutions de protection logicielle n'est pas aisée. Dans quelle mesure la protection mise en oeuvre est-elle effectivement résistante à la rétro-ingénierie ?

Le papier est organisé de la manière suivante : nous présentons l'architecture générale d'une solution de protection logicielle du marché civil représentative de l'état de l'art dans ce domaine et quelques un des mécanismes sur lesquels elle s'appuie dans la section 2. Nous présentons les méthodes d'attaque dans la section 3. Nous présentons enfin une démarche, des critères et des outils pour évaluer la robustesse d'un système de protection logicielle dans la section 4.

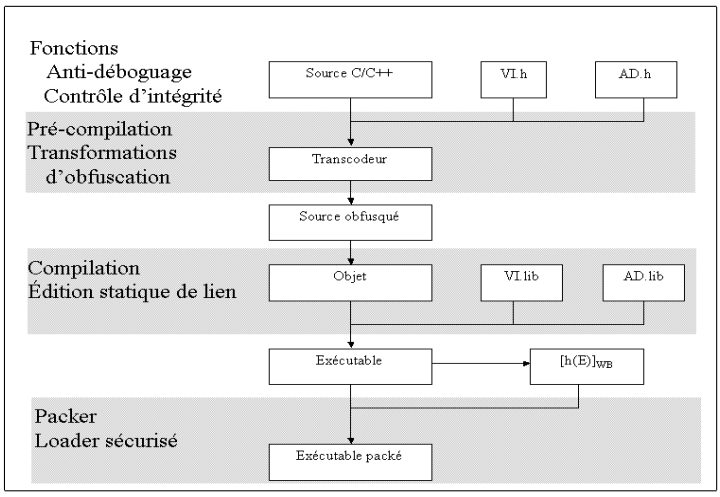


FIG. 1 – Synoptique de la chaîne de compilation CSS

2 Solution et mécanismes de protection logicielle

2.1 Architecture d'une solution

Le marché civil offre différents types de solutions de protection purement logicielle. Parmi celles-ci, les chaînes de compilation spécialisées paraissent les plus pertinentes, dans la mesure où elles autorisent l'intégration des mécanismes de protection à différentes étapes de la compilation du logiciel, assurant ainsi une protection en profondeur. On trouve dans cette catégorie de produits la solution CSS (Cloakware Security Suite), qui est à plusieurs égards représentative de l'état de l'art dans ce domaine, notamment en raison de la richesse des mécanismes qu'elle met en oeuvre. Elle propose en particulier des primitives cryptographiques adaptées au contexte WBAC, décrites au paragraphe 2.2.1. La figure 1 représente la synoptique de fonctionnement de cette chaîne de compilation, qui se compose pour l'essentiel d'un pré-compilateur, d'un packer et de bibliothèques spécialisées.

Le développeur dispose de deux bibliothèques pour protéger son application : la première regroupe les fonctions de contrôle d'intégrité, et la seconde regroupe les fonctions anti-déboguage. L'appel à ces fonctions est à la discrétion du développeur.

Un pré-compilateur, nommé transcodeur, applique plusieurs transformations d'obfuscation se fondant principalement sur le codage/décodage dynamique des variables et sur des transformations d'obfuscation intra-procédurales. Ce transcodeur implémente également un mécanisme de diversification des transformations paramétrable par le développeur via une mise à la clé.

Le code source obfusqué est ensuite compilé. L'édition de liens entre l'objet et les bibliothèques du produit est une édition statique de liens. Une ou plu-

sieurs empreintes numériques portant sur tout ou une partie de l'exécutable résultant sont calculées. Ces empreintes sont ensuite chiffrées par utilisation d'un algorithme de chiffrement boîte blanche, le résultat étant concaténé avec l'exécutable. L'exécutable est ensuite empaqueté au moyen d'une application appelée *Secure Loader*. Cette ultime protection implémente effectivement une partie des fonctionnalités d'un loader d'exécutable. Elle est chargée de sécuriser le chargement en mémoire de l'application protégée. Elle procède en particulier à son déchiffrement en boîte blanche. Nous présentons dans la section suivante quelques uns des mécanismes implémentés dans les solutions de protection. Nous étudions donc plus en détail les rouages internes de ces solutions de protection.

2.2 Mécanismes de protection

Nous avons vu dans la section précédente que la robustesse des fonctions de sécurité proposées par une solution de protection logicielle repose sur l'agencement savant de mécanismes élémentaires de protection logicielle. Nous présentons maintenant une sélection de mécanismes de protection représentatifs des approches suivantes :

- la protection en confidentialité des clés et en intégrité des hachés par utilisation de la cryptographie boîte blanche (section 2.2.1),
- la protection en confidentialité des clés et l'ancrage logiciel, par génération environnementale de clé et de code (section 2.2.2),
- la protection contre l'analyse statique et dynamique, par application de transformations d'obfuscation et diversification du code (section 2.2.3),
- la protection contre l'instrumentation du contexte d'exécution, par contrôle de l'intégrité de l'environnement d'exécution (section 2.2.4),
- la protection contre l'analyse dynamique par utilisation de code automodifiable (section 2.2.5).

2.2.1 La cryptographie boîte blanche

Le modèle de menace traditionnellement utilisé en cryptographie symétrique boîte noire est le modèle d'attaque à clair choisi adaptatif. Ce modèle suppose que l'attaquant connaît l'algorithme, contrôle le nombre et le contenu des textes clairs, et qu'il a accès au chiffré. Il ne connaît pas la clé et la dynamique des opérations effectuées par l'algorithme est cachée, opaque. L'attaquant est supposé n'avoir aucune visibilité sur l'exécution du programme.
Le modèle de menace de la cryptographie boîte blanche paraît plus réaliste dans notre contexte : dans ce modèle, l'attaquant à un accès complet au logiciel de chiffrement et contrôle totalement l'environnement d'exécution. Il est donc en mesure de tracer l'exécution, examiner les résultats intermédiaires et les clés en mémoire, effectuer une analyse statique du logiciel, altérer le résultats des calculs et analyser dynamiquement les perturbations induites.
Pour le modèle boîte noire, des algorithmes de chiffrement itératifs par blocs comme le DES, puis son successeur l'AES, ont été proposés. Des attaques ont été proposées sur le DES et sur des versions affaiblies de l'AES, comportant

un nombre réduit de tour. Dans le modèle WBAC, cette manière de concevoir un algorithme de chiffrement et les méthodes de cryptanalyse correspondantes perdent un peu de leur sens. Dans ce modèle, un attaquant a accès aux fonctions de tour, et peut donc effectuer la cryptanalyse d'une partie choisie de l'implémentation représentant un nombre réduit de fonction de tour.

La solution de protection CSS propose une implémentation particulière du DES et de l'AES permettant de rendre plus difficile l'extraction de la clé en boîte blanche. Le principe consiste à implémenter une version spécialisée de ces algorithmes, c'est-à-dire une version qui embarque la clé K, et qui n'est capable d'effectuer qu'une seule des deux opérations chiffrer ou déchiffrer. Cette implémentation est résistante en boîte blanche dans la mesure où il est difficile d'extraire la clé K en observant les opérations effectuées par le programme et qu'il est difficile de forger la fonction de chiffrement à partir de l'implémentation de la fonction de déchiffrement.
L'idée consiste à exprimer l'algorithme comme une séquence (ou un réseau) de *lookup tables*, et d'obfusquer ces tables en codant leurs entrées et sorties. Toutes les opérations du chiffrement par blocs, comme l'addition modulo 2 avec la clé de tour, sont embarquées dans ces *lookup tables*[2]. La représentation de l'algorithme sous forme de lookup tables nécessite de découper les transformations réalisées durant les r tours d'une manière différente. La figure 2 illustre ce redécoupage pour le DES. Chaque tour du DES est découpé en deux couches, l'une qualifiée de non-linéaire contient les S-Box, tandis que la seconde, qualifiée de linéaire rassemble les transformations linéaires telles que l'expansion, l'opération XOR et la permutation. Ces tables sont randomisées, de manière à obfusquer leur fonctionnement : les entrées/sorties de ces *lookup tables* sont codées par des bijections aléatoires. L'utilisation de cet encodage assure une sécurité locale, c'est-à-dire que la *lookup table* $g \circ T \circ f^{-1}$ codée par utilisation des bijections f et g ne fournit pas d'information sur la *lookup table* T d'origine. Soit T' une bijection quelconque. Il existe toujours deux bijections f' et g' telles que $g' \circ T' \circ f'^{-1} = g \circ T \circ f^{-1}$ (prendre par exemple $f' = f \circ T^{-1}$ et $g' = g \circ T'^{-1}$).
Cette sécurité locale est évaluée par une mesure dite "d'ambiguïté", qui exprime la difficulté à laquelle un attaquant cherchant à lever les protections doit faire face (cf. paragraphe 4.3 pour une définition de la mesure d'ambiguïté).

2.2.2 La génération environnementale de code et de clé, ancrage logiciel

La *génération environnementale de clé* [18, 19, 33] est un mécanisme qui permet de ne pas stocker la clé dans l'exécutable. Celle-ci est générée par application d'une fonction de hachage à des données d'activation présentes dans l'environnement d'exécution du logiciel.
Inspirée de ce principe, la *génération environnementale de code* [4] est un mécanisme qui permet de générer du code dynamiquement, en fonction de données

[2] Remarquons que le stockage en mémoire de ces tables impacte les performances de manière non négligeable.

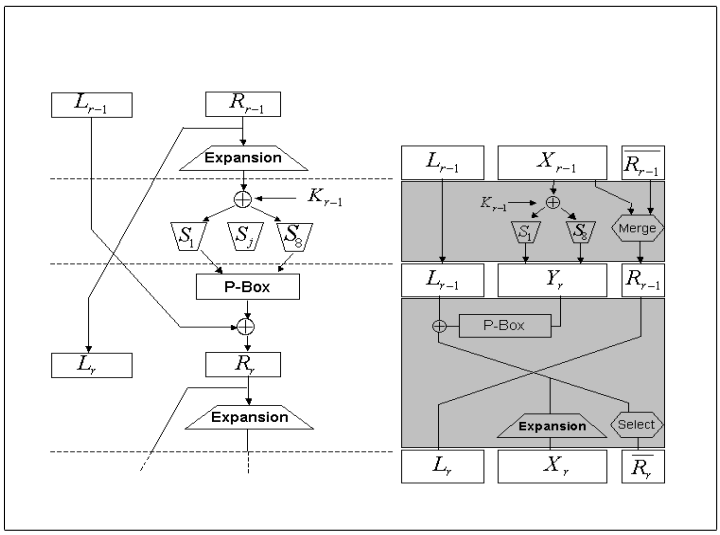


FIG. 2 – Un tour du DES et son équivalent en boîte blanche

d'activations présentes dans l'environnement d'exécution du logiciel.
Lors de la phase de protection du logiciel, un groupe d'instructions I est supprimé. Etant donnée une clé K, et une fonction de hachage h, une valeur S est recherchée en force brute de manière à satisfaire à l'équation $h(K||S) = I$.
Lors de la phase d'exécution du logiciel, la clé K est générée par application d'une fonction de hachage à des données d'activation présentes dans l'environnement d'exécution du logiciel. Le groupe d'instructions I est alors généré par $h(K||S) = I$.
Lors d'une analyse statique de code (ou d'une analyse dynamique dans un environnement ne possédant pas les mêmes propriétés que l'environnement cible), l'analyste connaît S, le domaine de valeurs de K. Il ne connaît pas le morceau de code généré. Pour retrouver le morceau de code, l'attaquant doit parcourir l'espace de K et pour chaque valeur, tester le code généré. Selon la nature (sémantique) du code généré, cette attaque par force brute peut être très difficile à mener. Une analyse critique de ces mécanismes est développée dans [19].
Remarquons également que lors d'une analyse dynamique du code dans l'environnement cible, l'attaquant peut cependant retrouver la clé K et le code associé.

2.2.3 Techniques d'obfuscation de code résistantes à l'analyse statique et dynamique

Les techniques d'obfuscation de code visent à augmenter la complexité du programme pour ralentir la compréhension de celui-ci par un attaquant. Nous proposerons une définition plus formelle de cette notion au paragraphe 4.1. Il

existe une grande similarité entre ces techniques, destinées à dégrader l'intelligibilité d'un programme, et les méthodes de diversification de code, visant à générer des versions polymorphes d'un programme, c'est à dire des programmes possédant les mêmes fonctionnalités que le programme d'origine, mais dont le code est sensiblement différent.
L'analyse de ces protections se fonde également sur des méthodes similaires, visant à défaire les transformations d'obfuscation ou à normaliser le programme protégé, par application de transformations successives sur son graphe PCG (*Program Control Graph*), de manière à converger vers une forme *canonique* du programme, proche du programme d'origine.
Les techniques d'obfuscation doivent être résistantes à l'analyse statique, et en particulier à toutes les transformations d'optimisation visant à réduire la taille du programme tout en conservant sa sémantique. Les transformations implémentées par un moteur de polymorphisme doivent de plus être résistante aux méthodes d'analyse différentielle de binaire, lorsque l'attaquant dispose de plusieurs instances polymorphes du programme protégé.

Parmi les techniques usuelles d'obfuscation, on trouve notamment : l'utilisation de *junk code*, la destructuration du découpage fonctionnel du code, la sur-utilisation d'instructions de saut à des adresses calculées lors de l'exécution et l'utilisation de prédicats opaques.
Un générateur de junk code transforme le code initial en un code final équivalent, en ajoutant des instructions supplémentaires dans le code initial. Ces instructions supplémentaires forment le junk code. Elles ne doivent pas changer la sémantique du programme. Le junk code peut être de différentes natures : il est soit mort (il n'est jamais exécuté), soit vivant. Le junk code vivant peut être composé d'instructions neutres (par exemple `add eax, ebx` suivit de `sub eax, ebx`) ou d'instructions utilisant des variables mortes (par exemple `add eax, ebx` sachant que le registre `eax` n'est pas utilisé).
Un code non obfusqué possède une certaine structure : un découpage en procédures et fonctions. Cette structure est très importante pour la compréhension du programme. La déstructuration du découpage fonctionnel du code vise à masquer la structure du code d'origine. L'ensemble des fonctions et des appels à ces fonctions est caché en le remplaçant par un autre ensemble de fonctions fabriquées aléatoirement. Les appels à ces nouvelles fonctions sont mis à jour de manière à ne pas modifier la sémantique du programme. Parmis les transformations qu'il est possible d'appliquer pour masquer la structure du programme, on trouve notamment : l'*inlining* (remplacement de l'appel à une fonction par le corps de la fonction), la fusion de fonctions (transformation permettant d'obtenir une fonction à partir de plusieurs) et la distribution de fonctions (transformation inverse permettant d'éclater une fonction en plusieurs).
La sur-utilisation d'instructions de saut à des adresses variables vise à entraver l'analyse statique du code. En effet, pour chaque `jmp eax` l'attaquant devra trouver la valeur contenue dans le registre `eax` lors de l'exécution du saut.
Un prédicat opaque est une séquence d'instructions ajoutées au code initial. Il n'en modifie pas la sémantique et permet de lier artificiellement des sections

de code indépendantes. Un prédicat opaque est un prédicat dont la valeur est connue lors de la compilation, mais qui est difficile à établir après compilation, par analyse du programme. Un prédicat opaque peut être soit toujours vrai, soit toujours faux, soit indéterminé. Afin de ne pas être facilement supprimé, un prédicat opaque est basés sur une propriété que l'on appellera son fondement. Ce fondement peut être par exemple une comparaison de pointeurs de l'application ou de pointeurs ajoutés, une comparaison de variables de l'application (sémantique) ou une propriété mathématique, dont la valeur de vérité est difficile à évaluer par une analyse du flot de données.

Ces techniques élémentaires d'obfuscation sont le plus souvent utilisées conjointement par les chaînes de compilation spécialisées. En particulier, le transcodeur de la solution CSS permet d'appliquer des transformations d'obfuscation mettant en oeuvre la plupart de ces techniques.

2.2.4 Contrôle de l'intégrité de l'environnement d'exécution

Ces techniques ont pour objectif de détecter les modifications apportées à l'environnement système d'exécution du logiciel, par rapport à l'environnement cible.
Un premier jeu de techniques vise à détecter les modifications effectuées par l'attaquant au niveau des API du système, afin de pouvoir récupérer de l'information sur un programme. Ces techniques d'instrumentation des API (*API Hooking*) peuvent être détectées en contrôlant l'intégrité des fonctions d'API critiques utilisées par le logiciel pour implémenter ses fonctions de sécurité.
Un second jeu de techniques vise à détecter l'exécution dans un mode correspondant à celui d'une analyse dynamique. Une analyse dynamique peut être menée au moyen d'un débogueur ou au moyen d'un émulateur.
Une analyse dynamique au moyen d'un débogueur peut être détectée par contrôle d'intégrité du code de l'exécutable ou par examen des modifications induites par le déboguage sur le contenu des registres du CPU.
Une analyse dynamique au moyen d'un émulateur est plus difficile à détecter. En particulier, si l'émulateur développé par l'attaquant est de bonne qualité, le contrôle d'intégrité du code ou l'examen des changements d'états du CPU virtuel ne seront d'aucune utilité. L'idée est alors d'appliquer des stimuli complexes au matériel émulé, et de contrôler la valeur de retour.

2.2.5 Code auto-modifiable

On parle de code auto-modifiable pour désigner un programme qui modifie dynamiquement sa section de code en mémoire. Les packers d'exécutable (par exemple le Secure Loader de la solution CSS) implémentent ce type de mécanisme pour supprimer ou déplacer une partie de l'information nécessaire à l'exécution d'un programme ou pour ne déchiffrer à chaque instant que certaines portions du programme protégé en mémoire. Ce mécanisme permet d'entraver à la fois l'analyse statique et la reconstruction de l'exécutable à partir de son

image mémoire, lors d'une analyse dynamique.

Nous avons présenté dans cette section une solution de protection logicielle représentative de l'état de l'art dans ce domaine. Nous avons également examiné quelques-uns des rouages fondamentaux de ce type de solution. Nous avons donc présenté le bouclier. La section suivante se propose d'étudier le glaive.

3 Techniques et méthodes de rétro-ingénierie

Nous présentons dans cette section les principales techniques et méthodes pouvant être mises en oeuvre par un attaquant pour analyser un exécutable protégé et lever la protection.

3.1 Instrumentation du contexte d'exécution

Lorsque l'on a levé la protection d'un programme, le programme déprotégé peut être selon les cas exécuté dans n'importe quel environnement, ou au contraire nécessiter un environnement d'exécution modifié (modifications au niveau du système d'exploitation, programme d'instrumentation dynamique devant être attaché au programme à chaque fois qu'il est lancé).
Dans tous les cas, il est nécessaire, lors de la phase d'analyse, d'instrumenter l'environnement d'exécution du programme observé, de manière à récupérer de l'information ou contourner sélectivement les protections.
Les méthodes d'instrumentation de l'environnement doivent être furtives, de manière à ne pas trahir la présence d'un environnement d'analyse aux yeux du programme protégé. Techniquement, cela impose la mise en place des codes d'instrumentation au plus bas niveau possible dans le système hôte : au niveau des API du système d'exploitation, au niveau des pilotes de périphériques, au niveau du matériel dans le cas de l'utilisation d'un composant matériel spécialisé ou dans celui de l'utilisation d'une machine virtuelle. Notons que la machine virtuelle utilisée doit être spécialisée, de manière à rendre inopérantes les fonctions de détection implémentées par la protection [24].

3.2 Méthodes d'analyse statique, dynamique, hybride

3.2.1 Analyse statique

L'analyse statique d'un exécutable vise à en obtenir le maximum d'information. Cette analyse comporte plusieurs étapes, chacune pouvant être naturellement entravée, compte tenu de la structure du jeu d'instruction CPU (non alignement des instructions) ou des limites intrinsèques de l'analyse statique (séparation du code et des données).
La première étape consiste à décoder les instructions, afin de localiser les instructions de branchement du programme. Les instructions de branchement interprocédurales sont utilisées pour construire le graphe PCG (*Program Control*

Graph) du programme, tandis que les instructions de branchement intra-procédurales sont utilisées pour forger les graphes CFG (*Control Flow Graph*) des procédures. Cette analyse peut être séquentielle ou récursive. Aucune de ces deux approches n'abouti à un désassemblage et à une représentation exacte du programme : la première approche abouti généralement à une sur-approximation conservative du graphe PCG d'origine, tandis que la seconde abouti à une sous-approximation du graphe PCG d'origine.
Des heuristiques sont donc employées pour localiser le point d'entrée des procédures. Ensuite, pour chaque procédure, une analyse récursive peut être conjuguée à une analyse séquentielle afin d'identifier les instructions de branchement intra-procédurales permettant de construire le graphe CFG. Enfin, le graphe CFG obtenu est élagué par application d'heuristiques appropriées.

On conçoit aisément dans ce contexte que l'introduction de sauts inutiles et l'obfuscation des prologues et épilogues des procédures sont autant de moyens naturels d'entraver l'analyse statique.
Cependant, la limitation la plus importante de l'analyse statique réside probablement dans l'utilisation intempestive de sauts à des adresses variables. La figure 3, page 11 illustre le problème de l'utilisation de sauts à des adresses variables, à travers l'utilisation des moteurs d'analyse statique d'OllyDbg (plugin OllyGraph) et d'IDA Pro.

- Dans le cas d'OllyGraph, l'instruction `call eax` n'est pas considérée comme une instruction de branchement. L'analyse s'arrête au basic bloc contenant l'instruction de saut.
- Dans le cas d'IDA Pro, la granularité des basic blocs prend en compte l'instruction de saut mais occulte la partie du programme invoquée par le saut.

Ce dernier problème est résolu par l'analyse dynamique.

3.2.2 Analyse dynamique

Nous avons vu au paragraphe précédent que l'analyse statique souffre de certaines limitations, dont celle d'aboutir à une sur-approximation (conservative) du graphe PCG et l'incapacité à résoudre certains problèmes liés à l'utilisation de variables dynamiques. L'analyse dynamique résouds naturellement le second problème. La figure 4, page 12 illustre la résolution du problème de l'utilisation de sauts à des adresses variables, par utilisation de l'outil d'analyse dynamique VxStripper. Une analyse dynamique peut donc permettre d'analyser correctement le code en présence de sauts à des variables.
Cependant, la limitation majeure de l'analyse purement dynamique provient du fait que seule une partie du graphe PCG est susceptible d'être découverte (lors de l'exploration d'un ou plusieurs chemins d'exécution), aboutissant donc à une sous-approximation du graphe PCG. Pour augmenter la taille du graphe PCG, et obtenir ainsi une représentation suffisamment complète pour permettre la mise en oeuvre d'une attaque efficace, l'idée consiste à forcer l'exploration de plusieurs chemins d'exécution [26]. Cette exploration peut être :

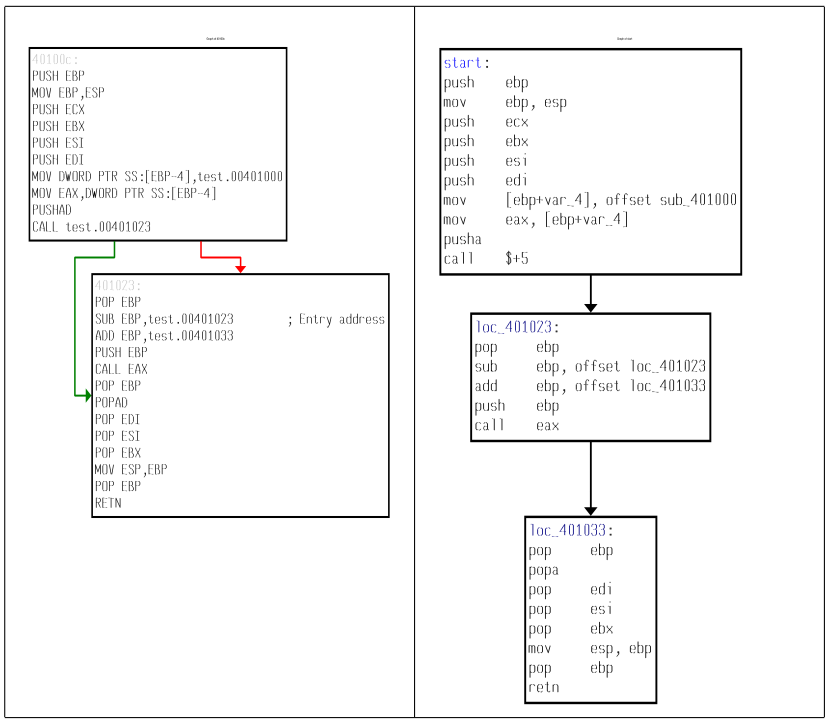


FIG. 3 – Graphes VCG de `call eax` sous Ollygraph et IDA Pro

1. Systématique, c'est-à-dire que l'on localise tous les branchements conditionnels par lesquels le flux d'exécution passe et que l'on inverse la condition de manière à explorer une nouvelle branche.
2. Gouvernée par une analyse du flux de données, en traçant dynamiquement certaines données utilisées par le programme comme entrée et en identifiant les endroits où cette entrée est utilisée pour prendre une décision qui va modifier le flux de contrôle du programme.

L'analyse dynamique permet de lever facilement certaines transformations d'obfuscation, lorsqu'elles ne sont pas suffisament robustes (comme par exemple l'utilisation de sauts inconditionnels ou de sauts conditionnés par un prédicat opaque déterminé pour réordonner les instructions d'un programme, ou l'utilisation de junk code mort).
Nous illustrons cette position avec le cas de la suppression des sauts inconditionnels intempestifs. Il est possible de définir un invariant caractérisant un noeud comme destination d'un saut inconditionnel ne pouvant pas être supprimé [13, 29]. Ce résultat classique de l'optimisation statique des programmes est facilement utilisable lors d'une analyse dynamique.
La figure 5, page 13 présente l'adaptation de cette méthode statique à l'analyse

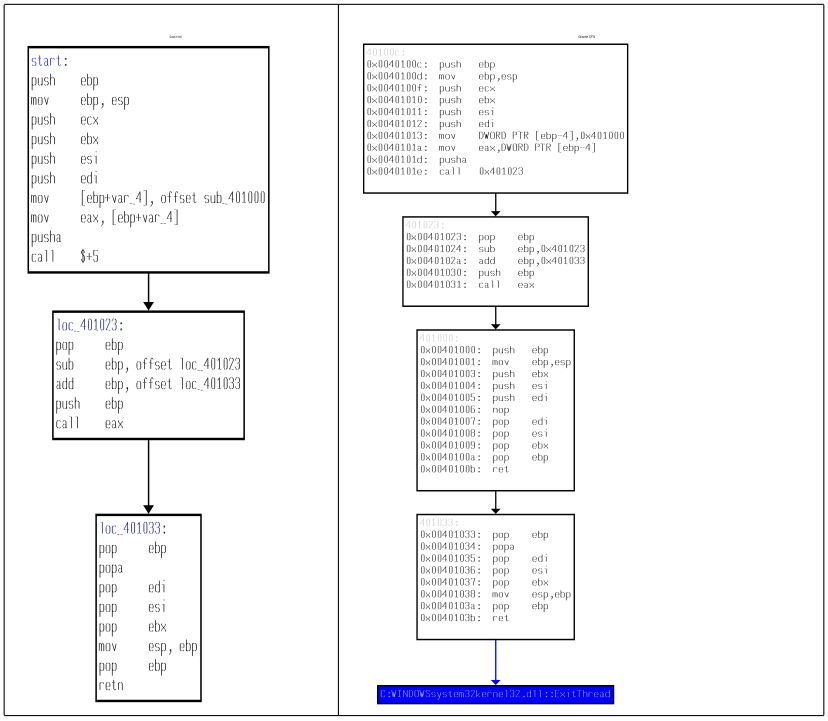


FIG. 4 – Graphe VCG de `call eax` sous IDA Pro et VxStripper

dynamique. La partie gauche de la figure montre le programme obfusqué. Dans la partie centrale, les noeuds violants l'invariant sont indiqués en rouge. La partie droite montre le graphe de flux du programme après suppression des sauts inconditionnels superflus.

Remarquons que l'analyse dynamique permet de supprimer naturellement le junk code mort et de *casser* les prédicats opaques déterminés. L'analyse dynamique permet également de supprimer aisément le loader de protection d'un exécutable, dès lors que celui-ci n'est pas suffisamment évolué [23].

3.2.3 Analyse hybride statique-dynamique

Les transformations d'obfuscation sont conçues pour entraver l'analyse statique. Elle visent le plus souvent à complexifier le graphe des chemins d'exécutions.

Les méthodes d'analyse statique (propagation de la constante, par exemple) propagent généralement l'information sur un sur-ensemble des chemins d'exécution effectivement parcourus lors d'une exécution du programme. Ces méthodes d'analyse sont, pour cette raison, dite conservatives.

Les méthodes d'analyse dynamique (tracing, profiling) ne peuvent généralement

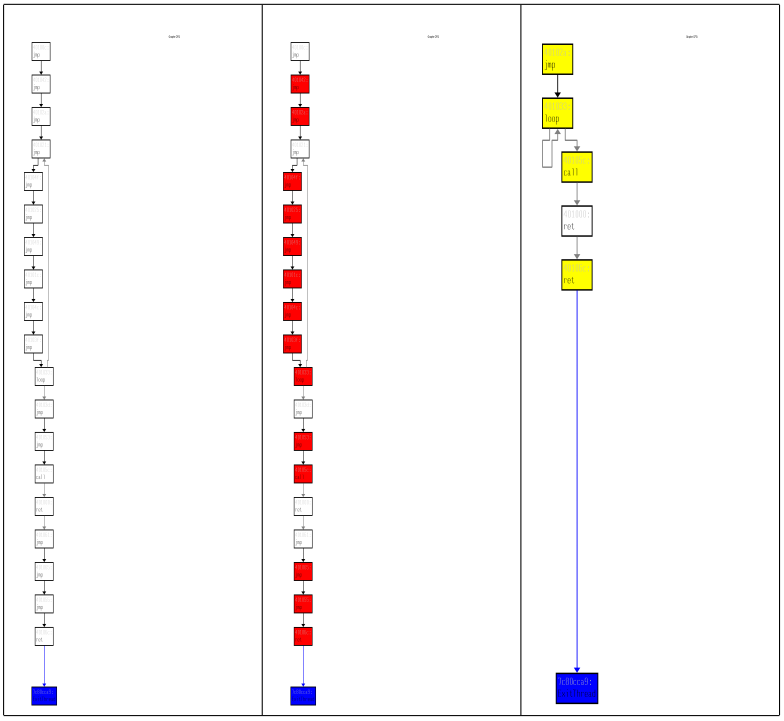


FIG. 5 – Suppression des sauts inutiles sous `VxStripper`

pas prendre en compte toutes les valeurs d'entrées ou toutes les interactions environnementales possibles d'un programme, et ne peuvent donc observer qu'un sous-ensemble des chemins d'exécution possibles. L'information est obtenue dynamiquement en exécutant le programme et en observant les chemins empruntés lors de l'exécution. Parce qu'il n'est pas possible d'observer le programme pour toutes les entrées et interactions possibles, ces méthodes d'analyse sont dites approximatives.
Ces deux approches, duales par nature, peuvent être utilisées conjointement. Les méthodes d'analyse hybrides [3] conjuguent une analyse dynamique, permettant de reconstruire une partie du graphe, avec une analyse statique, permettant d'ajouter des noeuds au graphe. Il est possible également de conjuguer d'abord une analyse statique, permettant de construire une sur-approximation du graphe, avec une analyse dynamique permettant d'élaguer le graphe, c'est-à-dire de supprimer les noeuds ou les chemins qui ne sont jamais parcourus lors d'une exécution du programme.

Nous avons présenté dans cette section les problèmes théoriques et pratiques rencontrés lors d'une tentative de lever la protection d'une application, ainsi que les principales méthodes d'analyse pouvant être mises en oeuvre par un at-

taquant. Nous en déduisons des critères théoriques et empiriques de résistance des mécanismes.

4 Evaluation d'une solution de protection

L'évaluation d'une solution de protection logicielle impose de définir des critères objectifs de qualité théoriques et pratiques sur les mécanismes de protection et sur la manière dont ils sont agencés. Rapellons que lors d'une analyse de robustesse des mécanismes, l'analyste adopte le point de vue de l'attaquant. Une partie des mécanismes de protection logicielle sont des mécanismes cryptographiques. Leur analyse impose donc a priori une cotation des mécanismes cryptographiques[3]. Or, la propriété de résistance face à la rétro-ingénierie impose une étude spécifique de la résistance de l'architecture cryptographique vis à vis d'attaques non cryptographiques. Une telle analyse ne fait généralement pas partie du périmètre d'une cotation cryptographique, dans la mesure où la démonstration de résistance face à la rétro-ingénierie n'est pas explicitement mentionnée dans la portée de l'analyse. L'analyse d'efficacité des mécanismes cryptographique est principalement fondée sur l'analyse cryptographique du schéma de conception et l'utilisation de tests statistiques. Dans le cadre d'une évaluation fondée sur les critères communs, une étude de l'efficacité des mécanismes vis à vis d'attaques non cryptographiques et donc notamment de résistance face à la rétro-ingénierie, est plus importante.

Pour adresser la problématique de la résistance à la rétro-ingénierie dans le contexte WBAC, plusieurs modèles concurents ont été proposés dans la littérature, pour formaliser les notions d'obfuscation et de désobfuscation, et définir un modèle de l'attaquant. Nous présentons dans la section 4.1 les modèles issus de la théorie de la complexité, hérités pour nombre d'entre eux des résultats théoriques de l'optimisation des programmes et rappelons dans la section 4.2 les résultats utiles à l'analyse de primitives cryptographiques en contexte boîte blanche. Ces modèles ont donné naissance à des critères de sécurité théoriques et empiriques sur la plupart des mécanismes de protection logicielle que nous avons présentés, et sont utilisés par les concepteurs de nouveaux mécanismes de protection pour justifier de leur pertinence. Nous présentons certains de ces critères dans la section 4.3. Nous proposons enfin une démarche (section 4.4) et présentons des outils (section 4.5) permettant d'assister l'évaluateur dans son analyse d'efficacité des mécanismes.

4.1 Modèles issus de la théorie de la complexité

Soit Π un ensemble de programmes, et PPT l'ensemble des machines de Turing probabiliste dont le temps de calcul est polynomial relativement à la taille des entrées. Un obfuscateur peut être défini formellement [9] comme un algorithme probabiliste $\mathcal{O}$ permettant à partir d'un programme P, d'en obtenir

[3] dans le sens adopté dans le cadre des qualifications mises en oeuvre par la DCSSI.

une version fonctionnellement équivalente $\mathcal{O}(P)$, qui est inintelligible au sens suivant : $\forall A \in PPT$, $\exists S \in PPT$ tel que

$$\forall P \in \Pi,\ p[A(\mathcal{O}(P)] \backsimeq p[S^P(1^{|P|})].$$

Cette propriété, dite propriété de la *boîte noire virtuelle*, stipule que la version obfusquée $\mathcal{O}(P)$ est parfaitement inexpugniable, dans la mesure où on ne peut pas espérer apprendre plus de la rétro-ingénierie du programme $\mathcal{O}(P)$ que de la simple observation de ses entrées/sorties.
Un tel compilateur générique n'existe pas. Ce résultat d'impossibilité a naturellement des conséquences importantes pour les concepteurs de mécanismes d'obfuscation adaptés au contexte WBAC.
Ce résultat peut être perçu comme très négatif pour le praticien. Il est cependant très dépendant du choix des définitions, d'un modèle et des questions posées. Il reste donc de peu d'utilité en pratique et ne répond pas à la question de l'existence d'un programme capable de dissimuler certaines des informations critiques qu'il embarque, quitte à autoriser la fuite d'informations moins importantes (ou sans utilité pour l'attaquant) lors de l'exécution du programme protégé.
Il conviendrait donc de définir un modèle plus adapté à la réalité, utilisable par exemple dans le contexte définit par la cryptographie boîte blanche.

Concernant les mécanismes d'obfuscation non cryptographiques, la sécurité de ces mécanismes est souvent justifiée par un argument de complexité, comme par exemple le caractère difficile de l'analyse du résultat produit par application d'une transformation du graphe PCG. L'efficacité de ces mécanismes est le plus souvent justifiée par le fait d'embarquer un ou plusieurs problèmes difficiles de l'optimisation de programme.
Les compilateurs éludent ce problème en résolvant des approximations conservatives de ces problèmes. Cependant, ces approximations conservatives ne peuvent jamais être complètes, et les compilateurs choisissent dans ce cas de laisser le programme partiellement non-optimisé.

Cependant, même si un problème difficile est embarqué, il n'est pas évident que la désobfuscation soit difficile. De manière générale, concernant les techniques d'obfuscation dont la sécurité repose sur la difficulté à résoudre des problèmes difficiles, au regard de la théorie de la complexité, il n'est pas immédiat qu'un attaquant doive nécessairement résoudre le problème difficile sous-jacent pour atteindre ses objectifs.

Concernant l'utilisation d'arguments issus de la théorie de la complexité pour établir une preuve de la sécurité d'un mécanisme cryptographique, nous pouvons également faire l'observation suivante : même si les problèmes NP-complets sont généralement considérés intraitables, cette difficulté ne rend finalement compte que du pire cas. En effet, certains problèmes sont faciles en moyenne et sont résolubles en temps polynomial avec des algorithmes d'approximation fournissant une solution très proche de l'optimale. Des réductions plus complexes du type connection cas moyen - pire cas « à la Ajtai » [1] ou encore des réductions de

problèmes réputés difficile en moyenne au mécanisme cryptographique considéré doivent alors être envisagées.

La définition de la notion de désobfuscation est tout aussi controversée : De manière informelle, on peut définir simplement un désobfuscateur comme un algorithme capable, partant d'un programme $\mathcal{O}(P)$, de trouver un programme moins complexe et fonctionnellement équivalent. Les transformations de désobfuscation implémentées par un tel programme sont alors des transformations d'optimisation, visant à réduire la complexité générale du code au regard de métriques appropriées. Il est possible d'imposer une propriété plus forte sur le programme résultant, en stipulant par exemple que les transformations de désobfuscation doivent converger vers une version normalisée du programme, unique à un isomorphisme de graphe PCG près. Dans ce modèle, le problème de l'arrêt est clairement réductible au problème général de la désobfuscation et est donc indécidable.

Nous rappelons dans la section suivante les résultats utiles à l'analyse de primitives cryptographiques en contexte boîte blanche.

4.2 Analyse des primitives cryptographiques

Il est tentant d'essayer de caractériser des notions de diffusion et de confusion, introduites par Shanon [34], par des critères sur les fonctions booléennes vectorielles. L'analyse en boîte noire du DES a conduit à la définition de critères de sécurité généraux sur les boîtes de diffusion et de confusion d'un système de chiffrement itératif par bloc vis à vis des cryptanalyses linéaire et différentielle, et de manière plus générale aux attaques par oracle aléatoire sur le dernier tour. Les fonctions vectorielles f les plus robustes vis à vis de ces attaques sont les fonctions courbes ou parfaitement non linéaires, dans le cas où f a autant de variables que de composantes, et les fonctions presques courbes (qui sont aussi presque parfaitement non linéaires, la réciproque étant fausse) dans le cas contraire. Ces fonctions sont caractérisées par une corrélation minimale avec les fonctions affines et les décalées de f
Cependant, nous pouvons remarquer qu'il n'y a pas de consensus réel sur la caractérisation des notions de diffusion et de confusion. La notion de confusion est le plus souvent caractérisées par une non linéarité optimale. Cependant, une autre approche consiste à interpréter la notion de confusion comme une décorrélation parfaite des composantes de f prises deux à deux.
La notion de diffusion est parfois confondue avec celle de critère de propagation d'ordre élevé, qui est maximal pour les fonctions courbes, ou avec la notion de multipermutation. Une autre approche consiste à caractériser la propriété de diffusion d'une fonction booléenne vectorielle f par une corrélation uniformément petite avec les fonctions affines et les décalées de f.

Concernant l'efficacité des primitives de chiffrement itératif par blocs dans le contexte WBAC, nous pouvons faire les observations suivantes : tout d'abord

il est assez naturel d'avoir essayé de porter des algorithmes connus pour leur robustesse en boîte noire, dans la mesure où un algorithme de chiffrement résistant dans le contexte WBAC doit également être résistant en boîte noire. Nous pouvons ensuite remarquer que les premières spécifications sont assez recentes : Spécification WB-DES (2002, [11]), spécifications WB-AES (2002, [12]). Les attaques sont principalement des cryptanalyses différentielles : Cryptanalyse WB-DES par injection de fautes (2002, [10]), Cryptanalyse WB-AES (2004, [8]), Amélioration WB-DES (2005, [28]), Cryptanalyse WB-DES (2005, [32]), Cryptanalyses WB-DES (2007, [21] et [22]). L'efficacité de ces attaques est bien plus importante que les meilleures attaques menées dans un contexte boîte noire sur le DES ou l'AES.

Nous sommes donc à même de nous demander si une implémentation boîte-blanche forte du chiffrement par bloc est réalisable. Malgré l'efficacité des cryptanalyses et les résultats généraux d'impossibilité concernant l'obfuscation, il n'est pas prouvé qu'il n'est pas possible de réaliser une implémentation boîte-blanche forte d'un algorithme de chiffrement par bloc. Nous savons seulement que la tâche n'est pas aisée. Une première piste de recherche pourrait être de développer un algorithme de chiffrement par bloc spécifique, conçu spécialement pour une implémentation boîte blanche, plutôt que d'utiliser des algorithmes de chiffrement itératif par blocs connus pour être résistants en boîte noire et dont l'implémentation est simplement adaptée au contexte boîte blanche.
La grande force de ces algorithmes repose en effet sur l'utilisation répétée d'une fonction de tour. Dans le contexte WBAC, les attaques peuvent être menées sur un nombre réduit de tour, voyant leur efficacité décuplée.
Il est peut être possible de concevoir un algorithme d'obfuscation de clé dans un algorithme dont la connaissance ne fournit pas suffisamment d'information pour permettre la mise en oeuvre d'attaque par cryptanalyse linéaire ou différentielle, visant à reconstruire les primitives cryptographiques de confusion et de diffusion, les bijections aléatoires ou à retrouver la valeur des clés de tour.
Les techniques actuelles d'obfuscation des algorithmes de chiffrement par bloc consistent, en quelque sorte à reproduire un contexte boîte noire à un niveau de maillage plus fin : celui du tour. En effet, chaque tour r est grosso modo implémenté par une lookup table « noircie » grâce à l'adjonction de bijections aléatoires en entrée et sortie. Cette boîte transmet une donnée (le texte en cours de chiffrement) à la lookup table du tour suivant de manière « codée », là encore grâce à l'insertion de bijections aléatoires. La sécurité de cette implémentation boîte blanche semble alors réduite à la robustesse des lookup tables « noircies », c'est-à-dire sur l'incapacité d'extraire de l'information de ces dernières, ainsi que sur celle du codage des données transitant entre les lookup tables. L'utilisation d'un schéma de type « confusion / diffusion » apparaît clairement (surtout dans l'implémentation boîte blanche de l'AES) pour coder les données transitant entre ces lookup tables noircies. Intuitivement, il n'est donc pas étonnant que ces implémentations soit « cassées » puisque, si l'on considère les lookup tables comme des boîtes noires (on suppose qu'il n'y a aucune fuite d'information), ces boîtes communiquent finalement entre elles à l'aide d'un schéma de chiffrement

de type « confusion / diffusion » ne comportant qu'un unique tour, là où dans la cryptographie boîte noire classique, dix tours pour l'AES-128 et seize pour le DES sont préconisés.

4.3 Critères d'analyse

Evaluer la robustesse d'une transformation d'obfuscation n'est pas une tâche aisée. Il n'y a pas de métrique unifiée qui apporte une réponse univoque à la question : quelle est la robustesse de cette transformation d'obfuscation ? De nombreuses transformations d'obfuscation sont très utiles, même s'il n'est pas toujours possible de prouver leur pertinence au regard de la théorie de la complexité. Des critères plus souples permettent de décrire et de classifier ces transformations. Ces critères, même s'ils reposent sur des mesures de complexité, sont empiriques.
La notion de mesure de complexité est un concept général qui recouvre des notions plus intuitives comme le temps nécessaire à l'exécution d'un programme, la place mémoire nécessaire à cette exécution et la taille du programme. Plus formellement, on appelle mesure de complexité [7] toute application μ de $\mathbb{N} \times \mathbb{N}$ dans $\mathcal{F}^*$ telle que $\forall (k, z) \in \mathbb{N} \times \mathbb{N}$, $\mu(k, z) = \mu_z^k \in \mathcal{F}^{k+1}$ et le prédicat $\mu_z^k(x) = y$ est décidable.
On trouve dans la littérature un certain nombre de critères fondés sur une ou plusieurs mesures de complexité, permettant de justifier de l'efficacité d'une transformation d'obfuscation ou plus prosaïquement d'étalonner un obfuscateur. Nous donnons ici quelques uns de ces critères, tirés de [14] Il est ainsi possible de quantifier la gêne imposée à une analyse humaine par la quantité :

$$\mathcal{T}_{pot}(P) = \frac{c_\mu(\mathcal{O}(P))}{c_\mu(O)} - 1,$$

mesurer l'effort requis pour défaire une transformation d'obfuscation, en fonction de l'effort $\mathcal{T}_1$ à produire pour forger un outil de désobfuscation et de la classe de complexité $\mathcal{T}_2$ de l'analyse :

$$\mathcal{T}_{res}(P) = R(\mathcal{T}_1(P), \mathcal{T}_2(P)),$$

où R est une matrice de cotation définie par le concepteur. On peut mesurer le coût en terme de ressources (temps CPU/espace mémoire) induit par la protection : $\mathcal{T}_{cost}(P)$. Il est possible de définir un autre critère, sur la base de l'observation suivante : si une transformation introduit du code nouveau dans le programme protégé $\mathcal{T}(P)$, un attaquant pourra facilement le détecter et exploiter cette faiblesse pour lever la protection. Définissons donc un ensemble de caractéristiques du langage utilisé par un programme, $\mathcal{L}(P)$. Notons $\mathcal{L}(\mathcal{O})$ l'ensemble des caractéristiques du langage introduites par la transformation d'obfuscation $\mathcal{O}$. La furtivité de la transformation $\mathcal{O}$ relativement au programme P peut être définie par :

$$\mathcal{T}_{ste}(P) = 1 - \frac{|\mathcal{L}(\mathcal{O}) - \mathcal{L}(P)|}{|\mathcal{L}(\mathcal{O})|}$$

Si la transformation n'enrichit pas le langage des programmes qu'elle protège, la furtivité $\mathcal{T}_{ste}(P)$ est maximale et égale à *1*. Il est finalement possible de considérer l'ensemble de ces critères simultanément, en définissant la qualité d'une transformation par :

$$\mathcal{T}_{app}(P) = \frac{\omega_1 \mathcal{T}_{pot}(P) + \omega_2 \mathcal{T}_{res}(P) + \omega_3 \mathcal{T}_{ste}(P)}{\mathcal{T}_{cost}(P)}$$

où $\omega_i,\ i = 1, 2, 3$ sont des constantes fixées en fonction du contexte opérationnel d'utilisation des transformations d'obfuscation.
Ces critères, même s'ils reposent sur des mesures de complexité, sont empiriques, et sont plus utiles à l'étalonnage d'un obfuscateur qu'à la preuve de son efficacité face à la rétro-ingénierie.

Nous avons vu dans la section 4.2 des critères précis pour évaluer la sécurité d'une primitive cryptographique en contexte boîte noire. Dans le contexte WBAC, on peut envisager d'autres critères, tels que la diversité ou l'ambiguïté, permettant de rendre compte de la qualité cryptographique des bijections aléatoires, etc.
La diversité et l'ambiguïté sont des mesures permettant de qualifier a priori la robustesse de l'implémentation boîte blanche. La mesure de diversité consiste à compter le nombre d'implémentations différentes qu'il est possible de générer (incluant la variation des clés embarquées). Cette mesure est importante car elle qualifie la capacité de l'obfuscateur à parer, a priori, les attaques à grande échelle. Les attaques spécifiques à une instance n'ont alors qu'un impact limité. Néanmoins, cette mesure ne rend pas compte de la robustesse d'une implémentation particulière face à une attaque cherchant à extraire la clé embarquée. Afin de mieux qualifier la robustesse d'une implémentation il est plus intéressant de compter le nombre de constructions, c'est-à-dire le nombre de clés et bijections aléatoires, aboutissant à une même lookup table. Plus ce nombre est grand, plus l'obfuscateur introduit de l'ambiguïté. La mesure de l'ambiguïté permet de rendre compte de l'espace des possibilités auquel l'attaquant fait face pour trouver les bonnes combinaisons clé / bijection utilisées lors de la génération de l'instance boîte blanche qu'il détient.

4.4 Démarche d'analyse

Avant d'exposer notre démarche d'analyse, nous pouvons faire l'observation suivante relative à la définition d'une attaque effective et au modèle de l'attaquant : Le fait pour un système de protection de reposer sur des mécanismes dont la sécurité est prouvée vis à vis d'une attaque statique exacte ou conservative n'est pas suffisant. Si certaines attaques approximatives sont effectives, la sécurité en confidentialité des données critiques peut être compromise complètement.
Un système de protection logicielle est un produit de sécurité comme un autre. Sa sécurité peut donc être évaluée suivant la même démarche. Les mécanismes de sécurité mis en oeuvre ont cependant été moins étudiés que les mécanismes

cryptographiques traditionnels. Les règles concernant le dimensionnement des mécanismes et la propagation de la confiance au sein de telles architectures reste donc à définir[4].

Cette problématique a été éprouvée dans le cadre d'une analyse de sécurité de produits DRM du marché civil et dans le cadre d'une étude de robustesse de mécanismes de protection logicielles [25]. Quelques critères et repères méthodologiques semblent émerger.
En se basant sur le fait que toute solution de protection logicielle se décompose en un ensemble de mécanismes élémentaires, nous avons élaboré une démarche permettant de déduire d'une façon rigoureuse les objectifs de sécurité, vulnérabilités et hypothèses de sécurité d'une solution à partir des objectifs de sécurité, vulnérabilités et hypothèses de sécurité des mécanismes élémentaires constituant cette solution. Nous présentons succintement son principe ici. Pour une large gamme de mécanismes élémentaires (anti-dump, anti-déboguage, obfuscation, contrôle d'intégrité, chiffrement) nous décrivons leurs spécifications, contraintes d'intégration et d'exploitation, objectifs de sécurité, vulnérabilités, objectifs de sécurité et degrés de liberté.
Sur la base de cette description, nous pouvons représenter les objectifs et hypothèses de sécurité d'un mécanisme à l'aide des caractéristiques suivantes : résistance à l'analyse statique, à l'analyse dynamique, à la modification de données (ou de code), diversification et couplage entre une section de code et l'ensemble de l'application protégée.

Partant de l'analyse des hypothèses de sécurité d'une solution, nous pouvons alors évaluer le niveau de sécurité de cette solution. Cette démarche s'adapte par ailleurs facilement à l'introduction de mécanismes supplémentaires dans une solution de protection logicielle sur étagère et permet de maîtriser l'apport en terme de sécurité de cette introduction dans la solution existante.

4.5 Outils d'analyse

Une analyse de sécurité ne peut pas se dispenser de tests intrusifs, visant à éprouver la sécurité d'une application en exploitant ses failles pour lever la protection, c'est à dire dans notre contexte parvenir à exhiber les algorithmes propriétaires d'un programme ou parvenir à contourner une fonction de contrôle des droits. Nous avons vu que l'analyse hybride statique-dynamique est probablement la méthode la plus efficace en pratique. L'analyse manuelle d'un programme protégé nécessite la maîtrise de plusieurs types d'outils spécialisés :

- instrumentation du contexte d'exécution,
- analyse statique,

[4] A titre d'exemple, les contraintes liées à l'implémentation de primitives cryptographiques dans un contexte boîte blanche ne sont pas prises en compte dans le référentiel documentaire de la DCSSI (règles et recommandations concernant la gestion des clés et le dimensionnement des primitives cryptographiques).

– analyse dynamique.

4.5.1 Instrumentation du contexte d'exécution

Concernant l'instrumentation du contexte d'exécution, de très nombreux outils sont disponibles. Les kits de développement en proposent pour chaque plate-forme d'exécution. Les outils de diagnostic système peuvent également être utilisés. En outre, certains outils d'analyse dynamique spécialisés proposent des mécanismes d'instrumentation furtive de l'environnement d'exécution.

4.5.2 Analyse statique

De très nombreux outils d'analyse statique dédiés à la vérification ou à l'optimisation de programme ont été développés ces vingt dernières années, et peuvent potentiellement servir de support à la rétro-ingénierie. L'ambition de ce papier n'est pas de les présenter. Citons simplement une suite logicielle développée récemment et permettant d'éprouver par la pratique les quelques concepts que nous avons évoqués dans ce papier. Loco [17], est un outil permettant d'implémenter à la fois des transformations d'obfuscation et de désobfuscation. Il s'appuie sur l'outil Diablo [16], un éditeur de lien (édition statique de lien uniquement) particulier, permettant de réécrire un exécutable ELF après avoir effectué des modifications sur son graphe PCG (*link-time binary rewriting framework*). Ces deux outils s'appuient sur une interface graphique appelée Lancet [27].

4.5.3 Analyse dynamique

Les outils d'analyse dynamique spécialisés pour la rétro-ingénierie logicielle sont plus rare. Certains débogueurs proposent des fonctionnalités intéressantes (script d'OllyDbg ou d'IDAPro, par exemple), mais imposent à l'attaquant de laborieux efforts pour surmonter certaines protections logicielles. Les outils les plus efficaces pour récupérer de l'information sur un exécutable sans être détecté (et parfois lever automatiquement les premières protections) sont probablement les machines virtuelles spécialisées. Les outils CWSandbox [15], Norman SandBox [30], TTAnalyze [6], Cobra [35] et VxStripper [31] permettent de charger un exécutable dans une machine virtuelle et de l'analyser automatiquement.
L'outil d'analyse VxStripper permet en outre de supprimer automatiquement le loader de protection mis en place par la pluspart des packers du marché et de défaire automatiquement certaines transformations d'obfuscation. Il propose également des fonctionnalités avancées d'instrumentation du contexte d'exécution et d'analyse forensique du système d'exploitation, permettant de surveiller les interactions du programme cible avec celui-ci.

5 Conclusion

Nous avons présenté dans ce papier les caractéristiques, en terme d'architecture, des solutions de protection logicielle du marché civil et décrit les mécanismes fondamentaux qu'elles implémentent.
Nous avons présenté les problèmes théoriques et pratiques rencontrés lors d'une tentative de lever la protection d'une application, ainsi que les principales méthodes d'analyse pouvant être mises en oeuvre par un attaquant.
Nous avons proposé une démarche et des outils permettant de mener à bien une analyse de robustesse des mécanismes, sur la base de critères théoriques et empiriques. Une telle approche peut être utilisée par un évaluateur, pour mesurer le niveau de confiance qu'il peut porter à une solution ; par le concepteur de solution, afin d'en améliorer l'efficacité ; ou par l'utilisateur, afin d'intégrer une ou plusieurs solutions et éventuellement développer des protections supplémentaires.

Remerciements Cet état de l'art repose pour partie sur une étude effectuée pour un opérateur, dans le cadre de l'analyse des solutions DRM du marché civil, et sur une étude sur la protection logicielle réalisée pour un industriel de défense. Nous tenons donc à remercier les personnes avec qui nous avons pu collaborer lors de ces études. Nous remercions également les personnes du laboratoire de virologie et cryptologie et de l'école doctorale de l'école polytechnique qui encadrent les travaux de thèse d'un des auteurs.

Références

[1] Ajtai, M. & Dwork, C. (1996), A public key cryptosystem with worst-case / average-case equivalence. In : proceedings of 29th STOC, Texas, 1997, pp. 284-293.

[2] Anckaert, B., Cappaert, J., De Bus, B., De Bosschere, K., Madou, M., Preneel, B. (2006). On the Effectiveness of Source Code Transformations for Binary Obfuscation. In Proc. of the International Conference on Software Engineering Research and Practice (SERP06), June. 2006.

[3] Anckaert, B., De Bosschere, K., De Sutter, B., Madou, M. (2005). Hybrid Static-Dynamic Attacks Against Software Protection Mechanisms. In Proc. 5th ACM Workshop on Digital Rights Management (DRM05).

[4] J. Aycock, R. deGraaf, M. Jacobson (2005), Anti-Disassembly using Cryptographic Hash Functions. University of Calgary, Canada. Available at : http://pages.cpsc.ucalgary.ca/~aycock/.

[5] Anckaert, M., De Bosschere, K., B., Madou (2006). A Model for Self-Modifying Code. In Proc. 8th Information Hiding (IH06), July. 2006.

[6] Bayer, U. (2005). TTAnalyze : A tool for analyzing malware. Master's Thesis, Technical University of Vienna.

[7] Blum, M. (1967). A machine independent theory of the complexity of recursive functions. In : J. Assoc. Comput. Mach. 14 (1967), No. 2, pp. 322-336.

[8] Billet, O., Gilbert, H., & Ech-Chatbi, C. (2004). Cryptanalysis of a white box AES implementation. In : Helena Handschuh and M. Anwar Hasan, editors, Selected Areas in Cryptography, volume 3357 of Lecture Notes in Computer Science, pages 227-240. Springer, 2004.

[9] Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S., & Yang, K. (2001), *On the (Im)possibility of Obfuscating Programs*. Available at : http://www.math.ias.edu/~boaz/Papers/obfuscate.html

[10] Jacob, M., Boneh, D. & Felten, E. (2002). Attacking an obfuscated cipher by injecting faults. In Digital Rights Management Workshop, pages 16-31, 2002. Available at : http://www.cs.princeton.edu/~mjacob/papers/drm1.pdf

[11] Chow, S., Eisen, P. A., Johnson, H., & van Oorschot, P. C. (2002). A white-box DES implementation for drm applications. In Security and Privacy in Digital Rights Management, ACM CCS-9 Workshop, DRM 2002, Washington, DC, USA, November 18, 2002, Revised Papers, volume 2696 of Lecture Notes in Computer Science, pages 1-15. Springer, 2002.

[12] Chow, S., Eisen, P. A., Johnson, H., & van Oorschot, P. C. (2002). White-Box Cryptography and an AES Implementation. In Kaisa Nyberg and Howard M. Heys, editors, Selected Areas in Cryptography, volume 2595 of Lecture Notes in Computer Science, pages 250-270. Springer, 2002.

[13] Christodorescu, M., Jha, S., Kinder, J., Katzenbeisser, S., & Veith, H. (2007). Software Transformations to Improve Malware Detection. In : proceedings of Journal in Computer Virology.

[14] Collberg, C., Thomborson, C., & Low, D. (1997), A Taxonomy of Obfuscation Transformation. Department of Computer Science, University of Auckland (New Zealand).

[15] Sunbelt CWSandbox (2007). CWSandbox : Behaviour-based Malware Analysis. Available at : http ://www.cwsandbox.org/, September 2007.

[16] Diablo, a better link-time optimizer. Available at : http ://diablo.elis.ugent.be/, September 2007.

[17] De Bosschere, K., Madou, M., Van Put, L. (2006). Loco : An Interactive Code (De)Obfuscation tool. In Proc. of ACM SIGPLAN 2006 Workshop on Partial Evaluation and Program Manipulation (PEPM06).

[18] Filiol, E. (2004), Strong Cryptography Armoured Computer Viruses Forbidding Code Analysis : the BRADLEY virus. INRIA ISSN 0249-6399. In proceedings of EICAR 2005 Conference, StJuliens/Valletta - Malte.

[19] Filiol, E. (2006), Techniques virales avancées. Springer, Collection IRIS, XXI, 283 p., ISBN 978-2-287-33887-8.

[20] Filiol, E. & Fontaine, C. (1998), Highly nonlinear balanced boolean functions with a good correlation immunity. In : Advanced in Cryptology, EuroCrypt'98, Lecture notes in computer science 1403, pp. 475-488, Springer Verlag.

[21] Michiels, W., Gorissen, P., Preneel, B., & Wyseur, B. (2007). Cryptanalysis of White-Box DES Implementations with Arbitrary External Encodings.

[22] Goubin, L., Masereel, J.-M., & Quisquater, M. (2007). Cryptanalysis of white box DES implementations. Cryptology ePrint Archive, Report 2007/035, 2007. http ://eprint.iacr.org/

[23] Josse, S. (2006). Secure and advanced unpacking using computer emulation. In : proceedings of the AVAR Conference, Auckland, New Zealand, December 3-5, 2006.

[24] Josse, S. (2007). Rootkit detection from outside the Matrix. In : proceedings of the 16th EICAR Conference, Budapest, Hungary, May 5-8, 2007.

[25] Josse, S. & Monsifrot, A. (2005). Etude de l'architecture des solutions de protection logicielle, France Telecom division R&D. Document de diffusion limitée pouvant être commniqué, au cas par cas, sur demande.

[26] Moser, A., Kruegel, C., & Engin Kirda, E. (2007). Exploring Multiple Execution Paths for Malware Analysis.

[27] Lancet, a nifty code editing tool. Available at : http ://diablo.elis.ugent.be/lancet_main, September 2007.

[28] Link, H. E., & Neumann, W. D. (2005). Clarifying obfuscation : Improving the security of white-box DES. In ITCC (1), pages 679-684, 2005.

[29] Muchnick, S. S. (1997). Advanced compiler design and implementation, Morgan Kaufmann ed., ISBN 1558603204, 856 p.

[30] Norman SandBox Malware analyzer. Available at : http ://www.norman.com/microsites/malwareanalyzer/, September 2007.

[31] VxStripper, Virus (and other armored software) reverse engineering tool. http ://monsite.orange.fr/vxstripper/

[32] Preneel, B., &. Wyseur, B. (2005). Condensed white-box implementations. In Proceedings of the 26th Symposium on Information Theory in the Benelux, pages 296-301, Brussels,Belgium, 2005.

[33] James Riordan, Bruce Schneier, Environmental Key Generation towards Clueless Agents. School of Mathematics Counterpane Systems, University of Minnesota, Minneapolis, USA. Available at : http://www.schneier.com/paper-clueless-agents.pdf.

[34] Shannon, C. E. (1949), Communication theory of secrecy systems. Bell systems technical journal, 28, pp. 656-715.

[35] Vasudevan, A., & Yerraballi, R. (2006). Cobra : Fine-grained Malware Analysis using Stealth Localized-Executions. In : proceedings of IEEE Symposium on Security and Privacy, pp. 264-279, 2006.

[36] Wang, J. (1997). Average-case computational complexity theory, In : Complexity Theory Retrospective II, pp. 295-328, Springer, 1997.

[37] Wroblewsky, G. (2002). General Method of code obfuscation.

Security Concept for the IT-System of the German Armed Forces

Hartmut Seifert
Industrieanlagen-Betriebsgesellschaft mbH (IABG) Ottobrunn, Germany
seifert@iabg.de

ABSTRACT

The "Security Concept for the IT-System of the German Armed Forces" has been revised fundamentally.

The main focus is now based towards a global network orientation, which is normally known under the term "Network Enabled Capabilities" and used in Germany under the term "Vernetzte Operationsführung, NetOpFü"

To support network enabled capabilities, a transparent network technology is necessary (IP-Protocol-Suite), which allows the network-wide operation of IPSec-based VPNs.

To avoid the formally used separation of the IT-System in different classification-zones, a new object-oriented approach is used to allow a parallel usage of different classified information (and services) within the same network-domain.

OVERVIEW

The "Security Concept for the IT-System of the German Armed Forces", actually available as a draft, is based on a Requirements Document from the German MoD, called „System Capabilities Requirements – Protected Information Provision" (Systemfähigkeitsforderungen – Sichere Informationsversorgung, SFF SIV).

Both documents are actually under preparation. Drafts are available.

This concept itself is structured in 5 parts:

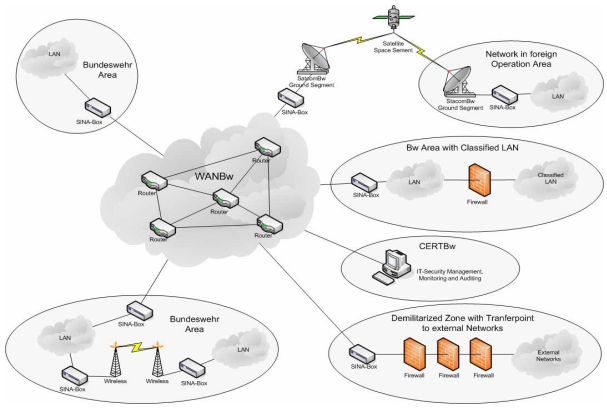
- Part A: Introduction
- Part B: Starting Position
- Part C: Conceptional Security Elements
- **Part D: Conceptional Solutions**
- Part E: Further Procedures

This presentation focuses on the Conceptual Solutions from this draft.

STRUCTURE OF THE SECURITY CONCEPT

In the Basic protection or enhanced basic protection is made no difference between information with different classification levels; in other words information with a higher classification level has to be protected within the object itself.

- persistent secure network infrastructure
- wide area networks are handled as unsecure (black); the IPsecurity is established at the network borders
- red/black separation at the edge-Routers
- accessing single information-objects via separate protection and appropriate labeling
- required is a persistent PKI-concept (also deployed) with a unique credential mapping to users/roles



ARCHITECTURAL SECURITY ELEMENTS

In principle, all users are placed within LANs that are interconnected through VPNs across the WANBw. External (nomadic) users

are connected via the WANBw to LANs using SCIP. If higher classification is required, hierarchical VPNs will be used (VPN-tunnel in VPN-tunnel).

The root element is the basic security („Grundschutz“) for elements of the ITSysBw. All elements that are basically placed in user domains are separated from the WAN interconnections. The access to the ITSysBw is centrally and consistent administrated: the user (and respectively his roles) has to authorize once at the system and he will receive the credentials for services and information („Single Sign On“). The access to services and information is controlled locally respectively when the access occurs.

MEASURES AGAINST ATTACKS WITHIN THE WANBW

To protect the WANBw against external and internal attacks, the following capabilities are necessary:

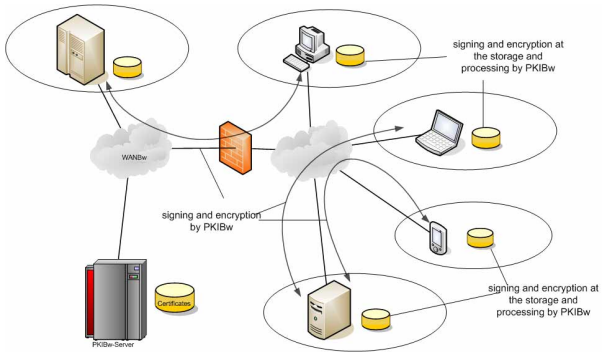
- protection
- detection
- reaction
- recovery

Protection is mainly realized by firewall and intrusion prevention systems, detection by intrusion detection systems (IDS). The efficiency of the detection is measured. The detection and localization of an attack(er) is realized and persons concerned are automatically informed or alarmed. To enable an automatic reaction WANBw should implement IRS (intrusion response systems). The WANBw implements a CERT (Computer Emergency Response) and features an CIR (Computer Incident Response) if necessary.

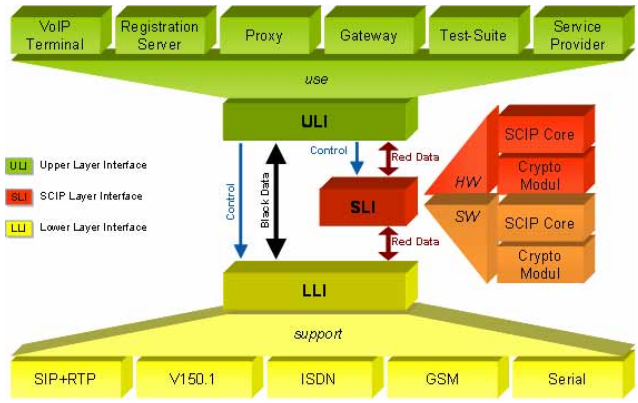
To ensure secure information flow in WANBw there is a need for homogenous encryption-means and ~devices which can also be deployed in multinational environments. There is also a need of device authentication to the network.

Conceptual methods are:

- PKI Bw (certificate management)



- SINA (IPSec VPN management)
- SCIP (application layer encryption)



- link layer encryption (ELCRODAT, SITLink …)

MEASURES AGAINST ATTACKS WITHIN LOCAL NETWORKS

To protect the LANBw against external and internal attacks, the same capabilities as in WANBw are necessary:

In local networks authentication is required for:

- devices
- users
- services

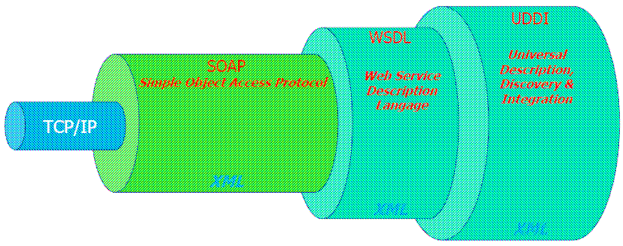
To realize the capability-requirement for transparent and interoperable IT-security functions there is the need of standard, coordinated and universal IT-security functionalities and ~services, to ensure confidentiality, liability and integrity of the transferred information.

These features can be enabled with method like:

- PKI enabled applications
- certificate-based authentication services
- online-key management
- e.g. single sign on, digital signature, controlled data access, …

CONCEPTIONAL SOLUTIONS

Conceptional Solutions to realize the security concept are a persistent PKI based on a bridge CA, the use of the SCIP Framework and the integration of an enhanced object security.



- any kind of information (as far as possible) is available via information objects
- bases on the protection requirement for a single information object each object is separately encrypted
- access rights for users/roles are realized via separate labels

used protocol e.g.:

- https; SOAP/SAML; XML wrapping/signature

FURTHER PROCEEDING

- aggregation of IT-Solutions to achieve the goal of a secure ITSysBw
- testing of new technologies
- implementing new technologies under real-life environments
- recommendations and selection of appropriate technologies
- updating of specifications and documents

REFERENCES

[1] Systemfähigkeitsforderung (SFF) für Sichere Informationsversorgung (SIV), BMVg Fü S VI 3, Az 79-10-30, Draft Version, dated Aug. 22, 2007

[2] Konzept IT-SysBw – IT-Sicherheit im IT-SysBw, IT-AmtBw A6, Draft Version, dated Aug. 31, 2007

SEND : la découverte de voisins IPv6 sécurisée

Francis Dupont
Internet Systems Consortium
Francis.Dupont@fdupont.fr

7 novembre 2007

Résumé

SEND est la version sécurisée du protocole de découverte des voisins d'IPv6 qui gère entre autres la correspondance entre les adresses niveaux réseau et liaison, et l'auto-configuration.

Il permet en particulier de contrer les usurpations d'adresses et l'injection de préfixes ou de routeurs pirates.

Il utilise la cryptographie d'une manière non immédiate.

Abstract

SEND is the secure version of the IPv6 neighbor discovery protocol which manages among other things the mapping between network layer and link layer addresses, and the auto-configuration.

In particular it provides a defense against address, prefix or router spoofing.

Its usage of the cryptography is quite sophisticated.

Très courte introduction à IPv6

IPv6 [1] est la nouvelle version du protocole réseau de l'Internet. Sa principale caractéristique, mais pas la seule, est d'offrir un adressage sur 128 bits au lieu des 32 bits d'IPv4.

Les adresses IPv6 ont une structure : les adresses unicast commencent par 64 bits de préfixe utilisés par le routage, suivis par 64 bits d'identifiant d'interface (IID) qui est par défaut l'adresse IEEE MAC sur 64 bits avec le bit U/L inversé. Elles ont aussi une portée, par exemple les adresses `fe80::<iid>` sont les adresses locales au lien (*link-local*).

Pour finir la réponse à la question que tout le monde pose est oui, IPv6 va être déployé :

- il ne reste que pour quelques années d'adresses IPv4 à allouer ;
- le logo "Vista Ready" exige le support d'IPv6, quoique Teredo soit présenté par Microsoft plutôt comme un "peer-to-peer"...
- quelques très gros utilisateurs doivent disposer d'un espace d'adresses privées en dizaines de millions d'adresses, donc trop gros pour les plages d'adressage réservées (10.0.0.0/8, 192.168.0.0/16, etc) ;
- tout ceux qui savent vraiment ce qu'est un NAT (*Network Address Translator*) n'en veulent plus, surtout pour eux !

La découverte des voisins

Le protocole de découverte des voisins d'IPv6 [2] fait bien plus que remplacer le protocole ARP [3] dont en passant la sécurisation est totalement impossible.

En effet les fonctions assurées par le protocole de découverte des voisins (ND) sont :

- la découverte des routeurs par défaut sur le lien ;
- la découverte des préfixes du lien ;
- la découverte des paramètres du lien (*timers, hop limit, MTU, ...*) ;
- l'auto-configuration des adresses y compris la vérification qu'une adresse n'est pas déjà utilisée (*Duplicate Address Detection*, DAD) [4] ;
- la résolution d'adresse (la détermination de l'adresse lien en fonction de l'adresse IPv6) ;
- le routage sur le lien (la détermination du *next-hop*) ;
- la détection des voisins inaccessibles (*Neighbor Unreachability Detection*, NUD) ;
- la redirection (sur le lien, vers un autre routeur, ...).

Le protocole utilise cinq types de messages :
- la sollicitation du routeur (RS) ;
- l'annonce du routeur (RA) ;
- la sollicitation du voisin (NS) ;
- l'annonce du voisin (NA) ;
- la redirection (RED).

Des options extensibles transmettent les adresses niveau liaison, les informations sur un préfixe, etc.

L'analyse de la sécurité de ND [5] démontre que la plupart des fonctions de ND sont facilement attaquables. L'objectif de SEND est de protéger les points les plus critiques, en particulier les adresses et la découverte des préfixes.

Exemples de découverte des voisins

Pour les trois premières fonctions (découverte des routeurs, des préfixes et des paramètres), ND utilise un échange RS/RA :

```
sendr# tcpdump -e -p -v -n -s 1500 -i lnc1
tcpdump: listening on lnc1, link-type EN10MB (Ethernet), capture size 1500 bytes
14:07:43.208230 00:0c:29:cd:45:03 > 33:33:00:00:00:02, ethertype IPv6:
 fe80::20c:29ff:fecd:4503 > ff02::2: icmp6: router solicitation
 (src lladdr: 00:0c:29:cd:45:03)
14:07:43.264477 00:0c:29:e8:f8:76 > 33:33:00:00:00:01, ethertype IPv6:
 fe80::20c:29ff:fee8:f876 > ff02::1: icmp6: router advertisement
 (chlim=64, pref=medium, router_ltime=1800, reachable_time=0, retrans_time=0)
 (src lladdr: 00:0c:29:e8:f8:76)
 (prefix info: LA valid_ltime=2592000,preferred_ltime=604800,prefix=2001:fd::/64)
```

Mais l'échange le plus fréquent est le NS/NA pour le NUD (vérification de la présence des voisins) :

```
14:17:28.944505 00:0c:29:cd:45:03 > 00:0c:29:e8:f8:76, ethertype IPv6:
 2001:fd::20c:29ff:fecd:4503 > 2001:fd::20c:29ff:fee8:f876: icmp6: neighbor sol:
 who has 2001:fd::20c:29ff:fee8:f876
 (src lladdr: 00:0c:29:cd:45:03)
14:17:28.944654 00:0c:29:e8:f8:76 > 00:0c:29:cd:45:03, ethertype IPv6:
 2001:fd::20c:29ff:fee8:f876 > 2001:fd::20c:29ff:fecd:4503: icmp6: neighbor adv:
 tgt is 2001:fd::20c:29ff:fee8:f876 (RS)
```

Le problème des adresses

Un des principaux problèmes est le vol d'adresse dans toutes ses variantes. Pour le contrer il faut lier une adresse, ou ses 64 derniers bits après le préfixe

(l'identifiant d'interface, *IID*), à une clé afin de signer les messages critiques.

Deux grandes méthodes sont possibles :

- dériver la clé de l'adresse (*Address Based Key*) en utilisant un protocole cryptographique de type IBE (*Identity Based Encryption*, [6]). Un mécanisme basé sur le *pairing* de Weil dans des courbes elliptiques a été proposé mais n'a pas été retenu, peut-être parce qu'il n'était pas "assez mûr" ? Un autre problème de ce type de solutions est qu'il nécessite un tiers de confiance, l'*Identity-based Private Key Generator*, alors que les concepteurs de la protection des adresses dans SEND recherchaient un mécanisme strictement sans état.
- dériver l'adresse de la clé (*Key Based Address*). C'est la solution qui a été retenue sous le nom de CGA (*Cryptographically Generated Address*, [7]) et qui est décrite dans la section suivante.

Les CGA

Dans une CGA, l'IID est dérivée d'une clé publique RSA et d'un jeu de paramètres. La clé privée associée sert à signer les messages. La validation des messages est effectuée d'abord sur les règles de dérivation et ensuite sur la signature qui est une opération coûteuse.

Du point de vue cryptographique le problème est de fournir une preuve de possession de l'adresse tout en ayant par défaut pas d'état préalable dans les récepteurs.

Les paramètres sont :

- un modificateur sur 128 bits ;
- le préfixe sur 64 bits ;
- un compteur de collision sur 8 bits prenant les valeurs 0, 1 ou 2 ;
- la clé publique (encodage DER de la structure ASN.1 d'X.509) ;
- d'éventuelles extensions.

Ils servent à calculer deux valeurs de hachage :

- $hash_1$: les 64 premiers bits de l'application de SHA-1 aux paramètres ;
- $hash_2$: les 112 premiers bits avec le préfixe et le compteur de collision mis à zéro.

$hash_1$ fournit l'IID à l'exception des 3 premiers bits qui forment le paramètre de sécurité *sec*, du bit U (*universal*) forcé à un et du bit G (*group*) forcé à zéro.

La création d'une CGA est simple :

1. tirage d'un couple clé publique/clé privée ;
2. tirage d'une valeur aléatoire pour le modificateur ;
3. calcul de $hash_2$ et incrémentation du modificateur tant que $hash_2$ ne convient pas ;
4. remplissage du jeu de paramètres avec le compteur de collision à 0 ;
5. calcul de $hash_1$ et dérivation de l'adresse ;

6. vérification de l'unicité de l'adresse sur le lien, sinon incrémentation du compteur de collision et reprise à la phase précédente.

$hash_2$, qui ne dépend pas du préfixe afin de faciliter la renumérotation, doit avoir ses $16sec$ premiers bits à zéro. L'idée est d'augmenter le coût d'une attaque contre $hash_1$, c'est-à-dire trouver un jeu de paramètres donnant le même IID, du même facteur : la complexité du problème est en $O(2^{59+16sec})$ si SHA-1 reste robuste pour les problèmes de pré-images. Trouver une bonne valeur du modificateur, c'est-à-dire créer une nouvelle CGA, est aussi en $O(2^{16sec})$ ce qui est délibérément très difficile pour les grandes valeurs de *sec*.

Ce mécanisme de *puzzle* a été rendu nécessaire par le relatif petit nombre de bits disponible dans l'IID, la preuve de possession d'une adresse étant basée sur la difficulté à "casser" le hachage ou la clé privée.

L'utilisation de clé nue est due à la contrainte sans état mais n'est pas incompatible avec un système à base de certificats, il faut seulement qu'il ne soit pas "par défaut".

SEND

La version sécurisée de ND, SEND (*SEcure Neighbor Discovery*, [8] et [9]), est basée sur plusieurs mécanismes : des estampilles temporelles, des nonces, des preuves de possession des adresses et des délégations autorisées de préfixe avec des ancres de confiance.

Sauf pour le dernier mécanisme, de nouvelles options ont été définies :
- estampille temporelle (*timestamp*)
- nonce
- jeu de paramètres CGA
- signature RSA (avec le hachage de la clé publique et portant sur les adresses, les en-têtes ICMPv6 et ND, et toutes les options jusqu'à l'option signature/dernière option non comprise)

Les estampilles temporelles

Les estampilles permettent de rejeter les messages trop dans le passé ou le futur, ou pour une source connue, c'est-à-dire déjà présente dans le cache des voisins, trop décalés par rapport à la différence des horloges relevée au message précédent (les paramètres standards sont un delta maximal de 5 minutes, une imprécision d'horloge d'une seconde et un décalage maximal de 1 pour cent).

Plus précisément :

$$-Delta < (RDnew - TSnew) < +Delta$$

$$TSnew + fuzz > TSlast + (RDnew - RDlast) \times (1 - drift) - fuzz$$

Les nonces

Les échanges sollicitation/annonce sont protégés par des nonces ce qui assure un anti-rejeu : les réponses qui ne correspondent pas à une requête en cours sont rejetées.

Comme pour les estampilles temporelles, l'état nécessaire pour pouvoir valider les messages n'est qu'ajouté aux entrées du cache des voisins, c'est-à-dire il n'est pas créé et géré spécifiquement.

Les adresses

Les nœuds supportant SEND utilisent des CGA et transmettent le jeu de paramètres dans une option. Les routeurs utilisent en outre des certificats plus riches que des clés nues.

Les signatures

Tous les messages, à l'exception des RS de source indéfinie, doivent être signés avec une clé privée associée à l'adresse (CGA et/ou certificat). Il est donc impossible d'usurper les adresses car ce mécanisme fournit une preuve de possession (*ownership*) de l'adresse.

Les certificats

Les routeurs utilisent des certificats de clé publiques X.509v3 avec des attributs décrivant les préfixes gérés [10]. Ces attributs ont été définis pour le support de S-BGP, une version sécurisée du protocole de routage inter-domaine de l'Internet.

Par exemple :

```
sendr# openssl x509 -in cert.pem -text
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 0 (0x0)
        .......
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            .......
            sbgp-ipAddrBlock: critical
                IPv6 (Unicast):
                  2001:fd:/64
            .......
```

La chaîne de délégation des préfixes et la chaîne de signature des certificats sont parallèles : une délégation est donc reconnue comme sûre dès qu'une ancre de confiance figure dans le chemin de certification.

La découverte des délégations autorisées

SEND comprend un protocole sollicitation/annonce pour transmettre le chemin de certification en fonction d'ancres de confiance (désignées par leur sujets X.501 `Subject` ou FQDN `SubjectAltName`).

Il suffit donc de configurer sur les machines une ancre de confiance compatible pour sécuriser dans l'auto-configuration la partie découverte des préfixes du lien.

Analyse critique de SEND

SEND ne protège pas des attaques contre une couche liaison non sécurisée, par exemple il n'assure pas que les paquets proviennent du même nœud que les messages ND, ni lie cryptographiquement les adresses liaison et réseau.

De même le service multicast est géré par MLD (*Multicast Listener Discovery*, [11]) qui n'est pas (encore) sécurisé, alors que ND et SEND reposent fortement sur ce service.

SEND protège contre la création de fausses entrées dans le cache des voisins, la non-détection des voisins inaccessibles, les dénis de service utilisant la détection de la duplication des adresses, les fausses annonces provenant d'un routeur pirate (mais pas totalement d'un routeur piraté) et les rejeux.

De plus, SEND a été conçu pour être raisonnablement robuste.

SEND a quand même quelques défauts de conception notables :

- la cryptographie utilisée n'est pas "agile" : RSA et surtout SHA-1 ne sont pas remplaçables sans redéfinir le protocole ;
- le mélange de SEND avec un ND standard et donc non sécurisé sur le même lien est défini mais fait perdre le plus gros des avantages de SEND ;
- l'utilisation intensive des signatures rend SEND sensible aux dénis de service quand certains nœuds sont limités en performances de calcul ;
- le mécanisme de délégation autorisée des préfixes est complexe à déployer. Sans lui, il suffit de pré-configurer les nœuds quite à perdre toute flexibilité.

Ces défauts restent relatifs : SHA-1 est faible sur les collisions, pas sur les pré-images ; le coût des signatures n'est que le coût de la sécurité ; l'APNIC (le RIR Asie-Pacifique) expérimente les attributs X.509 du RFC 3779 [10].

SEND semble incontournable pour maintenir un niveau de sécurité acceptable dans de nouveaux protocoles comme NETLMM (*Network-based Localized Mobility Management*, [12]) ; et pour finir SEND est au minimum recommandé pour les infrastructures avec de hautes exigences de sécurité.

Retour d'expérience sur des implémentations de SEND

Nous disposons de deux implémentations expérimentales de SEND, une écrite à l'ENST Bretage fin 2005 mais non diffusée, l'autre par le laboratoire de DoCoMo aux USA.

Les deux sont entièrement en mode utilisateur : les paquets sont détournés dans les couches basses pour être traités à l'extérieur du noyau. Non seulement l'impact sur les performances est important, mais certains détails ne peuvent pas être correctement supportés, par exemple une requête avec une estampille temporelle inacceptable est simplement ignorée alors que dans certains cas il faudrait y répondre. En général, la gestion du mélange de SEND avec le ND standard est très partielle car elle demande une action directe sur le cache des voisins maintenu par et dans le noyau.

Enfin la partie certificat reste complexe à gérer quoique la dernière version d'OpenSSL (0.9.8e) supporte le RFC3779 (quand elle compilée avec la configuration idoine).

Conclusion

Le protocole de découverte des voisins est par conception protégé de l'activité en dehors du lien : il utilise des adresses dont la portée est le lien et vérifie que les messages n'ont pas été routés.

Avec un peu de configuration sur toutes les machines, il est facile de filtrer les mauvaises annonces venant de nœuds se prenant par "stupidité" pour des routeurs. Malheureusement ce type de problèmes est surtout fréquent dans les réseaux peu ou pas gérés, ce qui contredit l'aspect configuration.

En prenant en compte la contrainte d'éviter au maximum d'entretenir de l'état, en particulier sur les routeurs, afin de gérer la sécurité, la sécurité offerte par SEND semble raisonnable. Une solution plus directe comme une association de sécurité IPsec par couple de voisins serait probablement plus sûre et sûrement bien plus coûteuse, sans parler de la difficulté de baser l'identification des nœuds sur des adresses dont la gestion, y compris l'auto-configuration, est un des rôles du protocole...

Pour finir SEND a clairement besoin de plus d'expérimentation et de déploiement. En effet les contraintes de conception et les compromis qui en résultent n'ont pas vraiment été validés dans le monde réel, du moins dans le monde réel civil.

Références

[1] S. Deering, R. Hinden, *Internet Protocol Version 6 (IPv6) Specification*, RFC 2460, IETF, Décembre 1998

[2] T. Narten, E. Nordmark, W. Simpson, *Neighbor Discovery for IP Version 6 (IPv6)*, RFC 2461, IETF, Décembre 1998

[3] D. Plummer, *An Ethernet Address Resolution Protocol*, RFC 826, IETF, Novembre 1982

[4] S. Thomson, T. Narten, *IPv6 Stateless Address Autoconfiguration*, RFC 2462, IETF, Décembre 1998

[5] P. Nikander (ed.), J. Kempf, E. Nordmark, *IPv6 Neighbor Discovery (ND) Trust Models and Threats*, RFC 3756, IETF, Mai 2004

[6] A. Shamir, *Identity-Based Cryptosystems and Signature Schemes*, Crypto 84, Springer-Verlag LNCS 196, 1984

[7] T. Aura, *Cryptographically Generated Addresses (CGA)*, RFC 3972, IETF, Mars 2005

[8] J. Arkko (ed.), J. Kempf, B. Zill, P. Nikander, *SEcure Neighbor Discovery (SEND)*, RFC 3971, IETF, Mars 2005

[9] J. Arkko, T. Aura, J. Kempf, V.-M. Mäntylä, P. Nikander, M. Roe, *Securing IPv6 Neighbor and Router Discovery*, 1st ACM Workshop on Wireless Security, Atlanta, Septembre 2002

[10] C. Lynn, S. Kent, K. Seo, *X.509 Extensions for IP Addresses and AS Identifiers*, RFC 3779, IETF, Juin 2004

[11] S. Deering, W. Fenner, B. Haberman. *Multicast Listener Discovery (MLD) for IPv6*, RFC 2710, IETF, Octobre 1999

[12] V. Narayanan, J. Soininen,
`http://www.ietf.org/html.charters/netlmm-charter.html`

Standardization of Cryptographic Algorithms and the Role of EU-Funded Research on Cryptology

Bart Preneel

Katholieke Universiteit Leuven, Dept. Electrical Engineering-ESAT/COSIC,
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
bart.preneel@esat.kuleuven.be

During the last twenty years cryptographic algorithms have made a transition from limited use by governments and financial institutions towards massive use by every citizen. This includes applications on the Internet, mobile phones and credit cards. Before this transition, cryptographic was restricted to hardware; during the transition it moved to software and in the next decade cryptography will move to tiny devices and be "everywhere". This is only possible with a substantial effort in standardization. In this lecture we present an overview of the standardization bodies that are active in the area of security: the ISO world (ISO/IEC JTC1/SC27, TC68,...), the IETF, ETSI, CEN, NIST and IEEE (P1363, 802). Next we discuss the most important cryptographic standards. Subsequently we highlight the importance of independent evaluation before the publication of security standards. As examples, we will discuss the RIPE, AES, NESSIE and eSTREAM competitions as well as the upcoming NIST hash function competition. Finally we attempt to address the question why developing and maintaining security standards is so difficult.

La Stéganographie Moderne: d'Hérodote à nos Jours

Johann Barbier

Résumé — Nous présentons une introduction à un domaine jeune et qui offre de nouvelles perspectives en matière de protection de l'information, la stéganographie. Après un bref rappel historique, nous détaillons les définitions modernes de la stéganographie et de la stéganalyse et mettons en lumière les services de sécurité offerts par l'utilisation conjointe de schémas de cryptographie et stéganographie. Nous balayons enfin quelques unes des techniques classiques de stéganographie pour le texte, les images fixes et le son.

Mots clés — stéganographie, stéganalyse, TRANSEC.

Abstract — We present an introduction to steganography, a young research area which is a pomising mean to protect information. After a brief recall of the history, we detail the definitions of modern steganography and steganalysis and point out the new security services provided by the use of joint cryptography and steganography schemes. Finally, we describe some of the most popular steganography algorithms to hide data into texts, images and noise.

Keywords — steganography, steganalysis, TRANSEC.

I. Introduction

AUSSI ancienne que la cryptographie, la stéganographie moderne prend ses racines dans l'antiquité. Tandis que la première permet de communiquer secrètement, la seconde offre en plus la discrétion de la communication. Réservée jusqu'alors aux érudits, la stéganographie se démocratise avec l'avènement d'Internet et généralise alors la notion de furtivité des transmissions dans des canaux physiques, aux documents numériques. Parallèlement au développement des outils de stéganographie, la communauté scientifique s'est organisée depuis le milieu des années 90 et propose depuis des schémas de plus en plus évolués d'une part, mais aussi des détecteurs tout aussi efficaces. De plus, Internet offre un terrain de jeu inégalable pour les utilisateurs de logiciels stéganographiques. En effet, la quantité de supports présents sur la toile dilue l'information dissimulée dans un flux inexploitable par un analyste. Dans ce jeu un peu particulier *du glaive et du bouclier*, il semble que l'avantage soit définitivement du coté de celui qui met en œuvre la stéganographie.

Nous présentons tout d'abord un historique des techniques de stéganographie pour bien cerner la philosophie du domaine et les concepts d'emploi. Nous posons ensuite les bases de la stéganographie moderne et mettons en évidence les propriétés intrinsèques des schémas de stéganographie. Nous en déduisons ainsi les services de sécurité offerts par de telles techniques ainsi que les règles fondamentales de leur mise en œuvre. Puis, nous prenons la place de l'attaquant, ou stéganalyste, et modélisons le problème qui consiste à décider si un médium est stéganographié ou non en un problème de discrimination statistique. Pour ce faire, nous distinguons stéganalyse spécifique, *i.e.* dédiée à la détection d'un algorithme de stéganographie particulier, et stéganalyse universelle, c'est-à-dire consistant à essayer de détecter des algorithmes inconnus. Enfin, nous passons en revue quelques unes des techniques classiques de stéganographie.

J. Barbier appartient au département de Cryptologie du CELAR (France). Email : johann.barbier@dga.defense.gouv.fr

II. D'Hérodote à nos jours

A. Un art ancien

S'il est une discipline connue du grand public émulant l'imagination et la curiosité, c'est bien la cryptographie. Du code de César au « Da Vinci code », la cryptographie fascine ; tantôt elle est l'apanage des militaires et des espions, au secours de l'histoire ou d'amours impossibles, tantôt elle préoccupe les mathématiciens par les énigmes qu'elle offre et tantôt elle alimente les romans grands public [11], [10], [37]. Peut être plus ancienne et souvent amalgamée à la cryptographie, la *stéganographie* évolue dans l'ombre des « codes secrets », dissimulée derrière un objectif et un formalisme à la fois proches et différents de ceux de la cryptographie. Son étymologie grecque « *stego* », le secret, et « *graphia* », l'écriture, l'enracine dans l'antiquité. La stéganographie est donc l'art de l'écriture secrète. Tout au long de l'histoire, elle tient au même titre que la cryptographie, une place importante dans des évènements marquants. Ainsi, Hérodote relate, dans son œuvre *l'Enquête*, comment en 480 av. J.-C., Démarate réussit à prévenir les Grecques d'une invasion imminente du roi de Perse Xerxès 1er en envoyant un message gravé dans le bois d'une tablette d'écriture recouverte de cire, d'apparence vierge. En 300 av. J.-C., Énée le tacticien dans ses *Mémoires sur la stratégie*, décrit le premier système stéganographique qui consiste à marquer d'un trou les lettres d'un texte constitutives d'un message. En Chine, la coutume veut que le signal de la révolte des chinois contre la dynastie mongole Yuan lors de la *fête de la lune*, a été donné par des messages cachés dans des *gâteaux de lune*. Plus technique, l'invention de l'encre sympathique est attribuée au naturaliste Pline l'Ancien, romain du 1er siècle av. J.-C. et est encore utilisée de nos jours. Les techniques de stéganographie devenant de plus en plus savantes, très tôt les premiers ouvrages traitant du sujet voient le jour à partir du XVI^{e} siècle. En 1499, l'abbé Jean Trithème (1462-1516) publie le premier traité de stéganographie, intitulé *Steganographia* et composé de trois livres qui ne livrèrent tous leurs secrets que récemment. Le troisième livre n'a été finalement « décodé » par Thomas Ernst qu'en 1996 et indépendamment par Jim Reeds [34] en 1998.

Un scientifique allemand, Gaspart Schott (1608-1666) explique dans son livre *Schola Steganographica* comment dissimuler des messages en utilisant des notes de musique. Souvent taxés d'ésotérisme, certains de ces ouvrages, à l'instar de

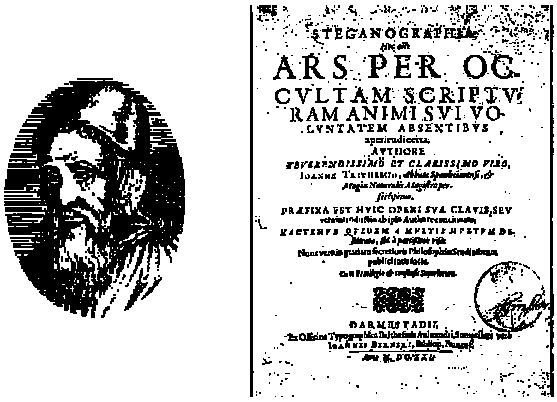


Fig. 1. Jean Trithème et *Steganographia*

Steganographia ont été interdits en leur temps. Néanmoins, l'intérêt vif du public pour les sciences du secret a rendu possible la diffusion de ces livres sous le manteau. La littérature en tant que vecteur de diffusion d'information est elle-même support servant à dissimuler des messages. Qu'elle soit médiévale ou moderne, elle foisonne de figures de styles telles *l'acrostiche* [1] mais aussi d'exemples plus croustillants les uns que les autres. Les plus célèbres d'entre eux sont notamment le poème de Boccacio (1313-1375), *Amorosa visione*, long d'environ 1500 vers et la correspondance privée entre George Sand et Alfred de Musset en 1883.

Alfred de Musset à George Sand
Quand je mets à vos pieds un éternel hommage
Voulez-vous qu'un instant je change de visage ?
Vous avez capturé les sentiments d'un cour
Que pour vous adorer forma le Créateur.
Je vous chéris, amour, et ma plume en délire
Couche sur le papier ce que je n'ose dire.
Avec soin, de mes vers lisez les premiers mots
Vous saurez quel remède apporter à mes maux.
Bien à vous, Éric Jarrigeon

La réponse de George Sand
Cette insigne faveur que votre cour réclame
Nuit à ma renommée et répugne mon âme.

Fig. 2. Correspondance entre George Sand et Alfred de Musset

Les techniques se sont complexifiées avec le temps et l'invention du micro-film en 1857 par Sir Brewster, puis du micro-point a redonné un nouveau souffle à la stéganographie. Elles permettent ainsi de réduire des photos à la taille d'un point sur un i et de les dissimuler dans un texte. Ces techniques ont été largement employées par les militaires pendant les différentes guerres franco-allemandes mais aussi les services de renseignement. La stéganographie a aussi marqué de son empreinte l'histoire contemporaine, notamment celle de la France. Le « *message de Verlaine* », diffusé en deux parties sur les ondes de la BBC le 5 juin 1944 à 21h15, « *Les sanglots longs des violons de l'automne* » et « *Blessent mon cœur d'une langueur monotone* », annonce le débarquement imminent des alliés. Plus tard, dans les années 80, Margaret Tatcher réussit à identifier la source de nombreuses fuites de documents en traçant ceux-ci à l'aide de techniques de dissimulation d'information. Enfin, plus récemment, de nombreux spécialistes relayés par les média [8], [25], [26], [36] avancent l'hypothèse selon laquelle Bin Laden aurait coordonné les attentats du 11 septembre 2001 en utilisant des messages cachés dans des images de sites à caractère pornographique. Le lecteur féru d'épistémologie trouvera son bonheur dans [21], [22], [23], [29], [33].

[1] L'acrostiche est un poème dont les premiers mots, lettres ou syllabes de chaque vers forment un message.

B. Un domaine de recherche jeune

Paradoxalement, la stéganographie dite *moderne*, c'est-à-dire adaptée aux données numériques, est relativement jeune. En pleine expansion, elle suit depuis le milieu des années 90 un essor corrélé à celui d'Internet ; le nombre de conférences scientifiques proposant des sessions dédiées à la dissimulation d'information augmentant chaque année. Si on se reconnaît dans une communauté par les points communs que l'on partage avec ses membres, parler le même langage est un point de passage obligé. De ce fait, on peut situer avec bonne approximation la naissance de la communauté des *stéganographes* en 1996, lors de la première édition d'Information Hiding et l'adoption d'un corpus relatif à la dissimulation d'information [31]. C'est d'ailleurs en 1997 qu'est soutenue une des premières thèses [18] dans le domaine. Á la lumière de l'histoire de cette discipline, on s'aperçoit que très longtemps la stéganographie est restée l'exclusivité de gens cultivés voir instruits. De nos jours, cela ne semble plus être le cas. En effet, Internet a fait tomber les barrières et offre à qui le veut des outils très performants et « prêts à l'emploi » en quelques clics. Entre 2004 et 2006, nous avons répertorié environ 120 outils de stéganographie facilement disponibles. Indépendamment de l'intérêt scientifique, l'étude de techniques de *stéganalyse*, c'est-à-dire des techniques visant à détecter la présence d'information cachée, possède un impact certain dans le domaine de la recherche de preuves informatiques [28], dans la lutte contre la pornographie infantile [3], [4], [35] et le terrorisme [1], [2]. Devant la prolifération avérée de tels outils, il devient alors important de développer des méthodes de stéganalyse pour dissuader l'usage de plus en plus répandu d'outils de stéganographie à des fins illégales.

III. La stéganographie moderne

A. Définition

La première étape est bien évidemment de définir précisément l'objet que l'on va étudier. Le lecteur pourra se référer à d'excellents ouvrages [20], [24], [39] traitant de dissimulation d'information. Bien que la communauté des stéganographes se soit constituée dans les années 90, G.J. Simmons pose en 1983 le socle de la stéganographie moderne en définissant la notion de *canal subliminal*. Pour illustrer son propos, il reprend le *problème du prisonnier*.

Le contexte général est le suivant. Soient Alice et Bob deux protagonistes partageant un secret commun et désirant communiquer ensemble de façon « sécurisée » ; Wendy une amie indiscrète qui voudrait bien avoir accès au contenu de leur correspondance. Un premier moyen pour Alice et Bob de protéger leurs communications est d'utiliser la cryptographie afin d'assurer notamment la confidentialité, l'intégrité, l'authenticité des messages qu'ils s'échangent. En employant la cryptographie, ils mettent ainsi en œuvre de la *sécurité de communication* (COMSEC). Dans de nombreux cas de figure, cette seule protection est suffisante. Prenons maintenant l'exemple d'un agent infiltré dans une organisation mafieuse qui doit rester en contact avec un agent de liaison de la police. Dans ce cas très précis, les deux agents doivent évidemment protéger leurs communications afin qu'un tiers interceptant le message ne puisse apprendre aucune information. De plus, l'existence même de leurs communications, indépendamment de leur contenu, peut compromettre la couverture de l'agent infiltré. En effet, un membre de la pègre avec le numéro d'un policier en mémoire sur son téléphone portable le désignerait rapidement comme suspect. Ils doivent alors rendre furtif leur canal de transmission, en mettant en œuvre de la *sécurité de transmission* (TRANSEC).

Dans le contexte du *problème du prisonier*, Alice et Bob sont deux détenus qui communiquent par l'intermédiaire de Wendy, le gardien. Si Wendy soupçonne qu'ils élaborent un plan pour s'échapper, celle-ci s'autorise à mettre fin à la communication entre les deux détenus. De plus, Wendy peut aussi modifier les messages si elle le désire. L'utilisation de messages chiffrés éveillerait les soupçons ; ils seraient de plus, contraints par les autorités à divulguer leur clé de chiffrement. La seule alternative d'Alice et Bob est donc de s'envoyer des messages innocents et de dissimuler l'information compromettante dans ceux-ci. De fait, ils mettent en place un canal de transmission (par l'intermédiaire des messages eux-mêmes) qui n'est pas visible pour Wendy ; ce canal est appelé *canal subliminal*. La stéganographie permet alors de généraliser les techniques classiques de TRANSEC, telles l'étalement de spectre ou l'évasion de fréquence, à tout type de données. Réciproquement, l'étalement et spectre et l'évasion de fréquence peuvent être vus comme des techniques de stéganographie, dissimulant un signal dans de la bande passant ou le spectre des fréquences. Ces techniques visent par ailleurs à rendre furtives les transmissions mais aussi à se protéger contre un attaquant actif qui brouillerait le canal.

Nous supposons tout d'abord qu'Alice et Bob se sont échangés au préalable une clé secrète cryptographique (ou ont accès à un serveur de clés publiques cryptographiques) ainsi qu'une clé secrète stéganographique (ou ont accès à un serveur de clés publiques stéganographiques). Nous appelons dans la suite *médium support* ou *support de couverture* le médium qui va contenir le message caché et *stégo médium* tout médium contenant de l'information cachée. Par abus de langage, nous utilisons aussi le terme de support et nous qualifions de *stégo* un stégo médium et de *non stégo* un médium non stéganographié. Le message à dissimuler est appelé *stégo message*. La mise en œuvre d'un schéma de stéganographie s'effectue alors en 2 étapes distinctes.
Pour envoyer un message à Bob, Alice effectue les opérations suivantes :

1) elle compresse son message et le chiffre avec la clé cryptographique,
2) elle génère un support de couverture,
3) l'algorithme de stéganographie sélectionne les sous-parties du support favorables à la dissimulation,
4) il insert ensuite aléatoirement, à l'aide de la clé stéganographique, le message chiffré dans les parties favorables,
5) Alice envoie le stégo médium par un canal classique.

Cette étape, appelée aussi *dissimulation*, est illustrée par la figure 3.

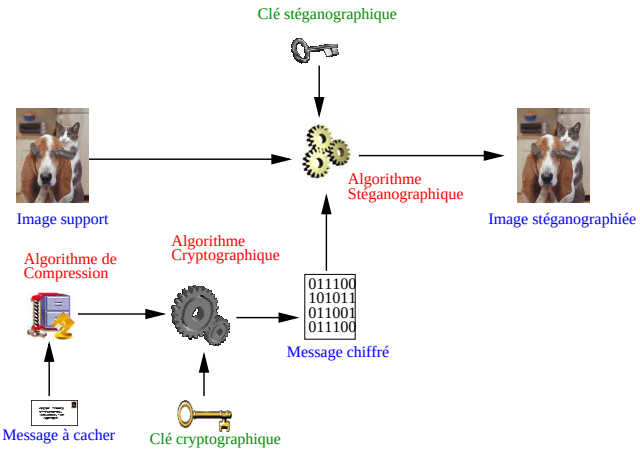


Fig. 3. Étape de dissimulation pour une image fixe

Pour lire le message d'Alice, Bob effectue les opérations suivantes :

1) Bob reçoit le stégo médium par le canal classique,
2) l'algorithme de stéganographie sélectionne les sous-parties du support favorables à la dissimulation,
3) il retrouve la position du message chiffré dans les parties favorables, à l'aide de la clé stéganographique,
4) Bob déchiffre le message à l'aide de la clé cryptographique et le décompresse,

Cette étape, appelée aussi *extraction*, est illustrée par la figure 4. Comme la cryptographie, la stéganographie peut être abordée sous l'angle de la théorie de l'information. Dans cet esprit, des définitions de schémas de stéganographie ont été proposées par C. Cachin [12], [13], [14], J. Zöllner *et al.* [41] et R. Chandramouli [15], [16], [17].

B. Du bon usage de la stéganographie

Quelques règles de base doivent être respectées pour éviter de mettre en défaut le schéma par des attaques triviales. Tout d'abord, c'est l'émetteur qui génère le support. Celui-ci doit n'être utilisé qu'une seule fois et détruit après utilisation, afin d'éviter les attaques par différence. En effet, tout attaquant

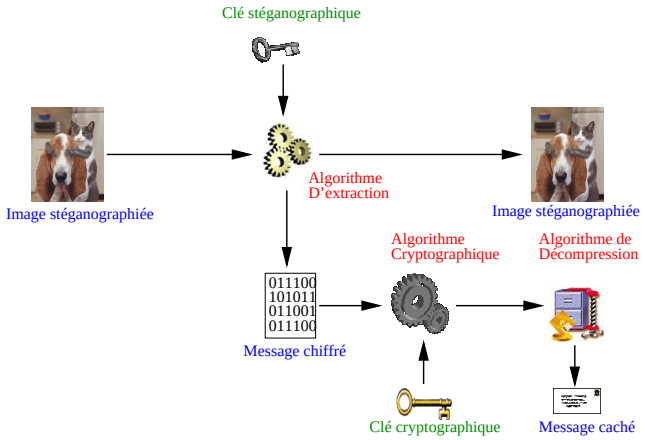


Fig. 4. Étape d'extraction pour une image fixe

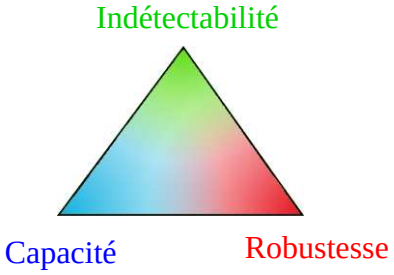


Fig. 5. Compromis capacité, détectabilité, robustesse

possédant le support original est capable avec une probabilité égale à 1 de détecter tout stégo médium issu du support. De même, pour éviter les attaques visuelles, l'algorithme de stéganographie ne doit pas détériorer visuellement le support. En général, les valeurs du support que l'algorithme de stéganographie modifie pour insérer l'information possèdent une distribution uniforme (par exemple les bits de poids faible (LSB)). Chiffrer permet d'une part d'assurer le COMSEC et d'autre part d'uniformiser la distribution des bits du message effectivement dissimulé. Le but étant d'obtenir une distribution des valeurs du stégo médium identique à celle des valeurs du support. De plus, afin de se prémunir contre des attaques classiques sur les moments d'ordre supérieur, comme un test du χ^2 par exemple, on demande aux algorithmes de stéganographie de conserver les statistiques des valeurs qu'il modifie, à l'ordre 1 et supérieur.

Enfin, la quantité d'information à dissimuler doit être petite. Le support peut être en effet considéré comme un canal au sens de Shannon [15], avec une capacité limitée. Intuitivement, plus on dissimule d'information dans un support, plus celui-ci subit de changements et plus le stégo médium risque d'être détecté. Dans le domaine de la dissimulation d'information, il faut composer avec un compromis entre la *capacité*, c'est-à-dire la quantité d'information que l'on peut insérer dans un support, *l'indétectabilité*, c'est-à-dire la probabilité que le stégo médium soit déclaré non stégo par un attaquant et la *robustesse*, c'est-à-dire la quantité d'information dissimulée résiduelle après un certain nombre de transformations sur le stégo médium. Ce compromis est traditionnellement représenté par un triangle comme illustré sur la figure 5. En stéganographie, le compromis qui nous intéresse est celui entre la capacité et l'indétectabilité. En effet, on considère que si le message est altéré, il sera ré-émis. L'objectif du stéganographe est bien d'envoyer le maximum d'information sans qu'un attaquant puisse le détecter. La notion de robustesse est plutôt importante pour le tatouage ou le marquage ; ceux-ci ne rentrant pas dans le cadre de notre étude. Le lecteur intéressé par le marquage d'image pourra trouver une bonne introduction dans les ouvrages [19], [20], [24].

C. Un nouveau service de sécurité

Nous prenons maintenant la place de l'analyste. Comme nous venons de le voir, les schémas modernes de stéganographie intègrent deux niveaux de sécurité indépendants et répondant à deux besoins de sécurité différents. Le Saint Graal du stéganalyste est bien évidemment d'avoir accès à l'information en clair échangée par Alice et Bob. Pour ce faire, il doit tout d'abord distinguer les stego média des autres, puis extraire l'information dissimulée et enfin cryptanalyser le message chiffré. Dans un contexte réaliste, et selon les principes énnoncés par A. Kerckhoffs [27], l'analyste ne dispose que des spécifications des schémas de stéganographie utilisés par Alice et Bob. Cryptanalyser le message chiffré est alors équivalent à une attaque à chiffré seul. Extraire l'information est équivalent à reproduire la suite de pseudo-aléa générée à l'aide de la clé stéganographique sans aucune connaissance ni de cette clé, ni même de la suite. Seule une attaque par recherche exhaustive sur la clé semble convenir. Enfin, distinguer les stégo média des autres est équivalent à trouver au moins une mesure statistique sur les média dont la distribution est différente suivant que le média est stégo ou non. Aux vues des étapes que doit franchir le stéganalyste, il semble que l'avantage soit définitivement acquis au stéganographe.

Supposons de plus, que celui-ci dissimule dans un même support de couvertue C deux messages m_1 et m_2 avec les clés stéganographiques respectives k_1 et k_2 pour obtenir le stégo médium S. Supposons de plus qu'il existe un distingueur stéganographique idéal ; c'est-à-dire capable de détecter les stégo média avec probabilité égale à 1. Une analyse de S avec ce distingueur indiquera qu'il contient de l'information cachée. Confondu, le stéganographe sera contraint de révéler une clé k_i et donc un message m_i, $i \in \{1, 2\}$. Or, l'extraction de m_i consiste en une lecture de S ; S étant inchangé après l'extraction, le distingueur classifiera toujours S comme stégo médium, qu'il contienne plus d'information dissimulée ou non. En d'autres termes, quelque soit le distingueur stéganographique, celui-ci ne peut intrinsèquement pas distinguer un stégo médium contenant exactement un stégo message, d'un autre en possédant plus d'un. Nous appelons cette propriété *l'indistingabilité indéniable*. Á la vue de cet exemple, il apparaît naturellement

une règle d'or du bon usage de la stéganographie : il faut dissimuler un message sans importance en plus du stégo message, et ce avec une clé stéganographique différente.

IV. L'analyse stéganographique

A. Stéganalyse spécifique et universelle

La première tache de l'attaquant, appelée aussi *stéganalyse*, est de distinguer les supports et les stégo média. Pour ce faire, il doit mettre en évidence des mesures qui donnent des résultats significativement différents selon qu'elles sont effectuées sur la population des supports ou sur celle des stégo média. Le problème de distinguer ces deux populations à partir de ces mesures est alors un problème de discrimination statistique. En effet, si l'on considère ces mesures comme des variables aléatoires définies sur un type de médium donné, alors pour pouvoir construire un distingueur stéganographique il suffit que ces variables aléatoires possèdent des distributions de probabilité différentes suivant qu'elles sont évaluées sur la population des supports ou sur la population des stégo média.

Selon le type de mesures effectuées, nous distinguons deux types de stéganalyse. Si les mesures dépendent des algorithmes que nous essayons de détecter, la stéganalyse est dite *spécifique*. Par exemple, J. Barbier *et al.* [5] proposent une stéganalyse dédiée aux algorithmes Outguess [32], F5 [40] et JPHide and JPSeek [30] en mesurant la variation d'entropie binaire d'une image JPEG après avoir stéganographié successivement plusieurs fois l'image avec le même algorithme. Cette méthode, appelée *stéganalyse par stéganographie multiple*, est clairement spécifique. La stéganalyse spécifique permet de répondre à la question : « *le médium a-t-il été stéganographié avec l'algorithme $\mathcal{A}$?* », où $\mathcal{A}$ est un algorithme de stéganographie donné.

A contrario, lorsque les mesures sont indépendantes de l'algorithme que l'on cherche à détecter, la stéganalyse est dite *universelle*, ou encore *aveugle*. Par exemple, J. Barbier *et al.* [6], [7] élaborent une stéganalyse dédiée au format JPEG en découpant le données compressées en blocs de taille N bits et en mesurant le poids de Hamming moyen de ces blocs sur l'ensemble de l'image. Ils remarquent notamment que ce poids moyen suit une loi de probabilité différente selon qu'elle est calculée sur la population des images JPEG naturelle ou sur celle des images JPEG stéganographiée. Cette stéganalyse est *universelle*. La stéganalyse universelle permet alors de répondre à la question « *le médium est-il stéganographié ?* ».

De façon immédiate, un attaquant mettant en œuvre une stéganalyse spécifique est alors plus « puissant » qu'un attaquant mettant en œuvre une stéganalyse universelle pour un même type de média. En effet, pour détecter des images stéganographiées avec un même algorithme, répondre à la question « *le médium a-t-il été stéganographié avec l'algorithme $\mathcal{A}$?* » permet de répondre à la question « *le médium est-il stéganographié ?* ». En revanche, un détecteur universel peut détecter plusieurs algorithmes à la fois. Puisque plus généraux, les détecteurs universels sont moins performants que les détecteurs spécifiques pour détecter un algorithme donné. De plus, en faisant travailler en parallèle tous les détecteurs spécifiques connus, on peut construire le meilleur détecteur universel qui répond « *stégo* » si et seulement si un des détecteurs spécifiques répond « *stégo* ». On voit donc clairement que l'intérêt principal d'un détecteur universel est de pouvoir détecter l'utilisation d'algorithmes pour lesquels on ne connaît pas de stéganalyse spécifique. J. Barbier *et al.* étendent alors la définition de l'universalité d'un schéma de stéganalyse à sa capacité à détecter l'utilisation d'algorithmes encore inconnus [6], [7]. Nous adoptons cette définition pour élaborer notre modèle d'attaquant.

Un attaquant, spécifique ou universel, procède en deux étapes qui correspondent aux deux étapes de la discrimination statistique. Tout d'abord, l'attaquant va essayer d'obtenir un maximum d'information sur un algorithme ou un type de médium donné. Ppour cela, il a en sa possession un ensemble d'algorithmes stéganographiques. L'attaquant va alors générer un ensemble de supports et de stégo média. Puis, à partir de cet ensemble, il va essayer de trouver les meilleurs critères pour distinguer les supports des stégo média de son ensemble. Cette étape est appelée *phase d'apprentissage*. Dans un deuxième temps, un ensemble de média va lui être soumis ; il devra alors essayer de deviner lesquels sont stéganographiés. Cette étape est appelée *phase de challenge*. La puissance d'un attaquant est mesurée avec trois indicateurs : la *probabilité de détection* $\mathcal{P}_{det}$, c'est-à-dire la probabilité que l'attaquant réponde correctement, la *probabilité de faux positifs*, $\mathcal{P}_{fa}$, c'est-à-dire la probabilité qu'il se trompe en répondant « *stégo* », et enfin, la *probabilité de faux négatifs*, $\mathcal{P}_{fn}$, c'est-à-dire la probabilité qu'il se trompe en répondant « *non stégo* ». La probabilité de détection se décompose en *probabilité de vrais positifs* $\mathcal{P}_{vp}$, *i.e.* la probabilité de répondre correctement lorsque le médium est un stégo médium, et *probabilité de vrais négagtifs* $\mathcal{P}_{vn}$, *i.e.* la probabilité de répondre correctement lorsque le médium est un support. Les performances d'un détecteur stéganographique sont alors représentées sur des courbes ROC (Receiver Operating Characteristic) qui mettent en évidence le compromis entre $\mathcal{P}_{vp}$ et $\mathcal{P}_{fp}$, comme l'illustre la figure 6. En pratique, on fixe *a posteriori* la probabilité de faux positifs maximum que l'on souhaite obtenir, ce qui fixe la probabilité de vrais positifs. Intuitivement, plus la courbe ROC est proche du coin supérieur gauche et meilleur est le détecteur. En stéganographie, on limite au maximum les faux positifs car ils sont plus coûteux que les faux négatifs. En effet, une fois qu'un stégo médium est détecté, on essaie d'extraire l'information cachée. D'autre part, le compromis illustré par la figure 5, montre que les performances des détecteurs stéganographiques dépendent de la quantité d'information dissimulée. Pour prendre en compte ce paramètre, les stéganalystes fixent le *taux stéganographique* (taille du message/taille support) et évaluent la courbe ROC pour ce taux. Pour un détecteur donné, nous obtenons ainsi une famille de courbes ROC.

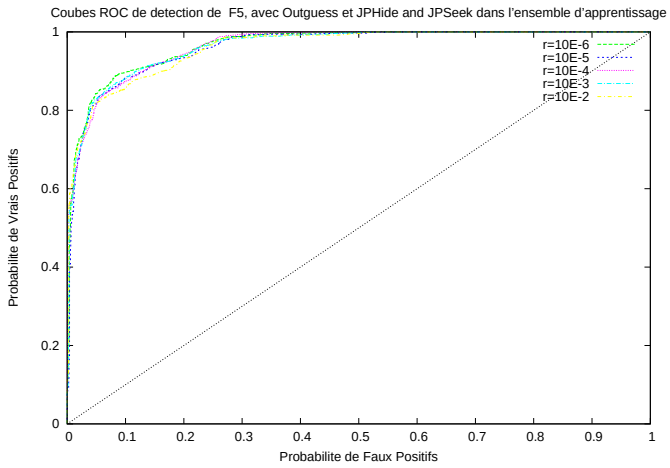


Fig. 6. Exemple de courbes ROC pour un détecteur universel [7] pour différents taux stéganographiques.

B. Un modèle d'attaquant spécifique

L'objectif d'un attaquant spécifique est de discriminer les supports des stégo média générés à l'aide d'un algorithme donné $\mathcal{A}$. Pour ce faire, durant la phase d'apprentissage, celui-ci génère un ensemble de supports puis il demande à un oracle de stéganographie de dissimuler un message aléatoire de taille choisie avec des clés aléatoires et l'algorithme $\mathcal{A}$ dans un support choisi. Il peut faire autant de requêtes à cet oracle qu'il le souhaite.

Dans la phase de challenge, l'oracle lui soumet des média qu'il a lui-même générés et éventuellement stéganographiés avec $\mathcal{A}$. L'attaquant peut alors demander à l'oracle de stéganographier le média qu'il doit analyser, et cela, autant de fois qu'il le souhaite. Enfin, il répond à la question « *est-ce que ce médium a été stéganographié avec l'algorithme $\mathcal{A}$?* ».

C. Un modèle d'attaquant universel

L'objectif d'un attaquant universel est de discriminer les supports des stégo média stéganographiés à l'aide d'un algorithme qu'il ne connaît pas. Pour ce faire, durant la phase d'apprentissage, celui-ci génère un ensemble de supports puis il demande à un oracle de stéganographie de dissimuler un message aléatoire de taille choisie avec des clés aléatoires et un algorithme, appartenant à un ensemble $\mathcal{E}_1$, dans un support choisi. Il peut faire autant de requêtes à cet oracle qu'il le souhaite.

Dans la phase de challenge, l'oracle lui soumet des média qu'il a lui-même générés et éventuellement stéganographiés avec un algorithme $\mathcal{A}$ n'appartenant pas à $\mathcal{E}_1$. L'attaquant répond alors à la question « *est-ce que ce médium a été stéganographié ?* ».

V. Quelques techniques classiques

L'ère du « tout numérique » a non seulement permis la dissémination des outils de stéganographie mais aussi la diversification des supports. Sans être exhaustif, il est possible de dissimuler, par exemple, de l'information dans le formatage d'une page html, dans des zones mortes de fichiers exécutables, dans des images, du son, de la vidéo ou encore des trames TCP. Comme nous l'ont montré nos anciens, chaque vecteur d'information est potentiellement support de schémas de stéganographie. Nous détaillons ici quelques techniques parmi les plus usitées.

A. Dissimuler de l'information dans du texte

Bien que très anciennes et très simples, les techniques consistant à dissimuler des stégo messages dans du texte sont toujours d'actualité. La quantité de mails échangés renforce de plus l'idée que cacher un stégo message dans du texte sera difficilement détectable. Une première méthode consiste à utiliser le formatage même du texte, en changeant, par exemple, la taille des espaces entre les mots. Pour lever tout ambiguïté, nous codons chaque bit du message par une transition sur le nombre d'espaces. Par exemple, ␣ ␣␣ → 1 et ␣␣ ␣ → 0.

Les␣parfums␣␣ne␣font␣␣pas␣␣frissonner␣sa␣narine ;
Il␣␣dort␣␣dans␣le␣␣soleil,␣la␣main␣␣sur␣␣sa␣poitrine,
Tranquille.␣␣Il␣a␣deux␣␣trous␣␣rouges␣au␣côté␣␣droit.

Fig. 7. « 110100100101 » dissimulé dans l'espacement inter mots (extrait du *dormeur du val*- Rimbault, 1870)

Malheureusement, cette méthode est très sensible au reformatage et facilement détectable. Une autre technique consiste à utiliser des dictionnaires de paires de synonymes. Un des mots de la paire codera 0 et l'autre 1.

1	0
dissimuler	cacher
information	donnée
équivalent	synonyme
simple	facile

Dissimuler des informations dans un texte avec des synonymes est très simple.

Cacher des informations dans un texte avec des équivalents est très facile.

Fig. 8. Dissimulation de « 0110 » par la méthode des synonymes

Enfin, une technique plus évoluée consiste à générer automatiquement un texte qui ressemble à du langage naturel, en choisissant des mots en fonction des bits du message à dissimuler. Pour cela, le texte généré doit être grammaticalement correcte et les mots utilisés doivent appartenir au même corpus pour donner un minimum de sens au texte. Le lecteur intéressé pourra se référer notamment à [18].

B. La stéganographie +/- k

La stéganographie +/-k est une généralisation de la stéganographie dite LSB (Least Significant Bit), technique

la plus simple mais aussi la plus répandue. Elle consiste à changer les bits de poids faible des valeurs du support pour qu'ils soient exactement égaux aux bits du message à dissimuler. Par exemple, dans le cas d'une image non compressée, les bits de poids faible de chaque triplet d'octets du codage (R,V,B) (Rouge, Vert, Bleu) d'un pixel contiendront chacun un bit du message comme l'illustre la figure 9.

Dans le cas d'un image non compressée codée sur 24 bits, chaque pixel est décrit par trois octets.

(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)
⋮

Pour dissimuler la chaîne de bits 10000011 on utilise le bit de poids faible de chaque octet.

(00100111 11101000 11001000)
(00100110 11001000 11101000)
(11001000 00100111 11101001)
⋮

Fig. 9. Stéganographie LSB pour une image non compressée

Soit un message m de taille l à dissimuler dans un support de couverture C. L'algorithme de stéganographie détermine tout d'abord les valeurs $c_1, \ldots, c_j$, qui peuvent être modifiées, puis dans un deuxième temps une permutation aléatoire σ de $[1, j]$ à l'aide d'un Générateur Pseudo Aléatoire (GPA) et de la clé stéganographique k_s. Enfin, pour chaque bit de message m_i, si $LSB(c_{\sigma(i)}) \neq m_i$ alors $c_{\sigma(i)}$ est incrémenté ou décrémenté de k (impaire).

C. La stéganographie adaptée au JPEG

Le format JPEG est sûrement l'un des formats d'échange d'images fixes les plus répandus et les plus usités. Tout naturellement, la majorité des algorithmes de stéganographie permettent de dissimuler de l'information dans des images JPEG. La compression JPEG peut se résumer à trois étapes majeures. Tout d'abord, l'image non compressée, située dans le *domaine spatial*, est représentée dans le *domaine fréquentiel* par l'application d'une transformée en cosinus discrète. Cette application est en fait la partie réelle d'une transformée de Fourrier discrète appliquée à l'image découpée en blocs de 8×8 pixels. Les coefficients DCT sont ensuite *quantifiés*, c'est-à-dire divisés et arrondis. Le passage de l'image non compressée aux coefficients DCT quantifiés est donc non réversible car elle implique une perte d'information. Les coefficients DCT quantifiés sont finalement compressés par un codage sans perte (Run Length Encoding suivi d'un code de Huffman). Pour plus de détails sur le format JPEG, le lecteur pourra se référer à [9], [38] et à la norme ISO 10918-1. La figure 10 résume la compression au format JPEG.

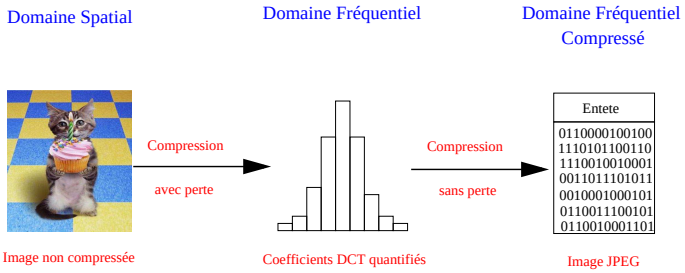


Fig. 10. Compression JPEG

La première partie étant une compression avec perte, dissimuler de l'information dans le domaine spatial n'est pas envisageable. De même, cacher de l'information dans le flux de données compressées entraînerait trop de distortions de l'image. Majoritairement, les algorithmes de stéganographie dédiés au format JPEG insèrent le message dans les coefficients DCT quantifiés. L'insertion peut se faire directement sur les LSB, comme le propose Outguess [32] ou en diminuant de 1 la valeur absolue des coefficients DCT non nuls, à la manière de F5 [40]. Dans ces deux algorithmes, les coefficients DCT sont choisis aléatoirement à l'aide d'un GPA. Dans le cas d'Outguess, la valeur du message est codée dans le LSB des coefficients DCT, dans le cas de F5, les valeurs paires négatives et impaires positives codent un 1 et les autres, un 0. Tous les deux conservent les statistiques d'ordre 1 des coefficients DCT quantifiés.

D. Dissimuler de l'information dans l'écho

Soit un signal discret $f(t)$, il est possible d'insérer un message m de longueur l en introduisant un écho dans le signal. Un écho correspond à l'ajout du signal décalé de Δt. Le signal stéganographié est alors

$$s(t) = f(t) + \alpha f(t - \Delta t) \ .$$

Le message à insérer est alors codé dans le choix de Δt. Un 0 est codé par Δt et 1 par $\Delta t'$. Le signal est alors découpé en blocs et l d'entre eux sont choisis aléatoirement pour contenir le stégo message. Enfin, les blocs sont concaténés pour donner le signal final.

Pour décoder l'information, le récepteur applique la *fonction d'autocorrélation* au *cepstrum du signal* donné par la formule

$$FT(\ln_{complexe} FT(f(t)) \ ,$$

où FT est la transformée de Fourrier. Un pique apparaît alors à la valeur Δt ou $\Delta t'$. Suivant la valeur des piques, le récepteur retrouve le stégo message.

E. Dissimuler l'information dans la phase

Une autre alternative pour dissimuler un message m de longueur l, consiste à décomposer le signal porteur en amplitude - phase et coder le message dans un décalage de

la phase. Pour ce faire, le signal $f(t)$ est découpé en n séquences de longueur l (la longueur du stégo message), $f_i(j)$ pour $i = 1 \dots n$ et $j = 1 \dots l$. Nous appliquons la transformée de Fourrier discrète et obtenons l vecteurs de phase $[\Phi_1(j) \dots \Phi_n(j)]^t$ pour $j = 1 \dots l$. Dans le cas d'un signal sonore, un décalage de phase entre deux segments est détectable facilement, il faut donc préserver la différence entre deux phases de segments consécutifs. Seul le premier d'entre eux contient donc le message m. Celui-ci est codé de la manière suivante. $\forall\, j = 1 \dots l$,

$$\Phi_0^{'}(j) = \begin{cases} \pi/2 & \text{si } m_k = 0 \\ -\pi/2 & \text{sinon.} \end{cases}$$

Afin de conserver la différence entre deux phases de segments consécutifs, on définit l nouveaux vecteurs de phase

$$\begin{bmatrix} \Phi_0^{'}(j) & & \\ \Phi_1^{'}(j) & = & \Phi_0^{'}(j) + [\Phi_1(j) - \Phi_0(j)] \\ \vdots & \vdots & \vdots \\ \Phi_n^{'}(j) & = & \Phi_{n-1}^{'}(j) + [\Phi_n(j) - \Phi_{n-1}(j)] \end{bmatrix}.$$

On reconstruit le signal par l'application de la transformée de Fourrier discrète inverse et concaténation des segments. L'extraction se fait en effectuant exactement les mêmes opérations et en évaluant les $\Phi_0^{'}(j)$. Il faut néanmoins noter qu'il faut que l'émetteur et le récepteur soient synchronisés pour pouvoir retrouver le stégo message.

VI. Conclusion

Outre l'évolution de la robustesse des techniques au cours du temps, les propriétés inhérentes à un schéma de stéganographie offrent un niveau de sécurité supplémentaire. En effet, l'analyste qui veut avoir accès illégalement à un stégo message doit d'abord détecter le stégo médium dans lequel il a été inséré, extraire l'information puis la déchiffrer. La sécurité liée à la détection est évaluée sous l'angle des courbes ROC à la lumière des stéganalyses connues. La sécurité liée à l'extraction est équivalente à une attaque par recherche exhaustive sur la clé stéganographique et la sécurité liée à la cryptanalyse est la sécurité traditionnelle en cryptographie de l'algorithme qui a servi à chiffrer le message clair. D'autre part, Internet fournit, par la quantité des média accessibles, un cadre favorable pour « noyer » des stégo média dans la masse d'information. De plus, à chaque format de données correspond une ou plusieurs techniques de stéganographie adaptées. La multitude de telles techniques contribue aussi à rendre la tache de l'analyste encore plus difficile. Forte de plus d'une trentaine d'années de recherche intensive, les cryptographes bénéficient de critères (taille des clés, propriétés de certains schémas ou fonctions, sécurité prouvable …) leur assurant un certain niveau de sécurité des schémas qu'ils proposent. *A contrario*, la stéganographie étant un domaine de recherche jeune, les stéganographes n'ont pas le même recul et leur objectif est un peu plus complexe dans la mesure où ils doivent se protéger contre la détection d'une part et l'extraction d'autre part. Néanmoins à l'heure actuelle, l'avantage semble dans leur camp ; un schéma de stéganographie bien conçu possède des propriétés intrinsèques fortes.

References

[1] Les efforts de la NSA vis-à-vis du Web : la stéganographie. *Le Monde du Renseignement*, 26 octobre 2000.

[2] Des messages cachés sur l'internet pour préparer les attentats. *Agence Française de Presse*, 12 octobre 2001.

[3] Anon : Child pornography on internet. http ://www.instant-essays.com.

[4] B.H. Astrowsky : " steganography "hidden images, a new challenge in the fight against child porn. *UPDATE*, 13(2), 2000.

[5] J. Barbier, É. Filiol et K. Mayoura : New features for specific JPEG steganalysis. *In* C. Ardil, éditeur : *Proc. 3rd International Conference on Computer, Information, and Systems Science, and Engineering, CISE 2006*, volume 16 de *Transactions on Engineering, Computing and Technology*, pages 72–77. World Enformatika Society, novembre 2006. ISBN : 975-00803-6-X.

[6] J. Barbier, É. Filiol et K. Mayoura : Universal JPEG steganalysis in the compressed frequency domain. *In* Y. Q. Shi et B. Jeon, éditeurs : *Proc. Digital Watermarking, 5th International Workshop, IWDW 2006*, volume 4283 de *Lecture Notes in Computer Science*, pages 253–267, Jeju Island, Korea, novembre 2006. Springer.

[7] J. Barbier, É. Filiol et K. Mayoura : Universal detection of JPEG steganography. *Journal of Multimedia*, 2(2):1–9, avril 2007. ISSN : 1796-2048.

[8] J.-P. Bay : Attention, une image peut en cacher une autre. *Lci*, septembre 2001.

[9] C. W. Brown et B. J. Shepherd : *Graphics File Formats, reference and guide*. Manning, 1995.

[10] D. Brown : *Da Vinci Code*. Jean-Claude Lattès, 2004. ISBN : 2709624931.

[11] D. Brown : *Forteresse Digitale*. Jean-Claude Lattès, février 2007. ISBN : 2709626306.

[12] C. Cachin : An information-theoretic model for steganography. *In* D. Aucsmith, éditeur : *Proc. Information Hiding, 2nd International Workshop*, volume 1525 de *Lecture Notes in Computer Science*, pages 306–318, Portland, Oregon, USA, avril 1998. Springer.

[13] C. Cachin : An information-theoretic model for steganography. *Information and Computation*, 192(1):41–56, juillet 2004.

[14] C. Cachin : Digital steganography. *In* H.C.A. van Tilborg, éditeur : *Encyclopedia of Cryptography and Security*. Springer, 2005. ISBN : 978-0-387-23473-1.

[15] R. Chandramouli : Data hiding capacity in the presence of an imperfectly known channel. *In Proc. SPIE Security and Watermarking of Multimedia Contents II*, volume 4314, 2001.

[16] R. Chandramouli : Mathematical theory for steganalysis. *In Proc. SPIE Security and Watermarking of Multimedia Contents IV*, 2002.

[17] R. Chandramouli, M. Kharrazi et N.D. Memon : Image steganography and steganalysis : Concepts and practice. *In* T. Kalker, I. J. Cox et Y. M. Ro, éditeurs : *Proc. Digital Watermarking, Second International Workshop, IWDW 2003*, volume 2939 de *Lecture Notes in Computer Science*, pages 35–49, Seoul, Korea, octobre 2003. Springer. ISBN : 3-540-21061-X.

[18] M. Chapman : *Hiding the Hidden : A Software System for Concealing Ciphertext in Innocuous Text*. Thèse de doctorat, The University of Wisconsin-Milwaukee, mai 1997.

[19] C. Fontaine : *Contribution à la recherche de fonctions booléennes hautement non linéaires, et au marquage d'images en vue de la protection des droits d'auteur*. Thèse de doctorat, Université Paris VI, novembre 1998.

[20] N.F. Johnson, Z. Duric et S. Jajodia : *Information Hinding - Steganography and watermarking - Attacks and countermeasures.* Adavances in Information Security. Kluwer Academic. ISBN : 0-7923-7204-2.

[21] N.F. Johnson et S. Jajodia : Exploring steganography : Seeing the unseen. *IEEE Computer*, 31(2):26–34, 1998.

[22] J.C. Judge : *Steganography : Past, Present and Future*. SANS, 2001.

[23] D. Kahn : *The Codebreakers*. MacMillan, New York, 1967.

[24] S. Katzenbeisser et F.A.P. Petitcolas : *Information Hiding. Techniques for steganography and digital watermarking.* Computer Science. Artech House. ISBN : 1-5853-035-4.

[25] J. Kelley : Terror groups hide behind Web encryption. *USA Today*, mai 2001.

[26] J. Kelley : Terrorist instructions hidden online. *USA Today*, mai 2001.

[27] A. KERCKHOFFS : La cryptographie militaire. *Journal des Sciences Militaires*, février 1883.

[28] G.C. KESSLER : Steganography : Implications for the prosecutor and computer forensics examiner. Rapport technique, American Prosecutors Research Institute, avril 2004.

[29] G. KIPPER : *Investigator's guide to steganography.* Information Security. Auerbach, 2004. ISBN : 0-8493-2433-5.

[30] A. LATHAM : Steganography : JPHIDE AND JPSEEK, 1999. http ://linux01.gwdg.de/∼alatham/stego.html.

[31] B. PFITZMANN : Information hiding terminology. *In Proceedings of the Workshop on Information Hiding*, numéro 1174, pages 347–350, Cambridge, England, mai 1996. Springer Verlag.

[32] N. PROVOS : Defending against statistical steganalysis. *In 10th USENIX Security Symposium*, Washington, DC, USA, 2001.

[33] F. RAYNAL, F. PETITCOLAS et C. FONTAINE : L'art de dissimuler les informations. *Pour la Science*, été 2002. Dossier " L'art du secret ".

[34] J.A. REEDS : Solved : The ciphers in book III of Trithemius's Steganographia. *Cryptologia*, (22):291–319, octobre 1998.

[35] E. RENOLD, S.J. CREIGHTON, C. ATKINSON et J. CARR : Images of abuse : A review of the evidence on child pornography. Rapport technique, National Society for the Prevention of Cruelty to Children (NSPCC), octobre 2003.

[36] D. SIEBERG : Bin Laden exploits technology to suit his needs. *CNN*, septembre 2001.

[37] N. STEPHENSON : *Le Cryptonomicon.* Payot, avril 2000.

[38] G.K. WALLACE : The JPEG still picture compression standard. *Commun. ACM*, 34(4):30–44, 1991.

[39] P. WAYNER : *Disappearing cryptography - Information Hiding : steganography & watermarking*. Morgan Kaufmann, 2002. ISBN : 1-55860-769-2.

[40] A. WESTFELD : F5-a steganographic algorithm. *In* I.S. MOSKOWITZ, éditeur : *Proc. Information Hiding, 4th International Workshop, IHW 2001*, volume 2137 de *Lecture Notes in Computer Science*, pages 289–302, Pittsburgh, PA, USA, avril 2001. Springer. ISBN : 3-540-42733-3.

[41] J. ZÖLLNER, H. FEDERRATH, H. KLIMANT, A. PFITZMANN, R. PIOTRASCHKE, A. WESTFELD, G. WICKE et Gritta WOLF : Modeling the security of steganographic systems. *In* D. AUCSMITH, éditeur : *Proc. Information Hiding. Second International Workshop, IH'98*, volume 1525 de *Lecture Notes in Computer Science*, pages 344–354, Portland, Oregon, USA, avril 1998. Springer-Verlag.