Université Bordeaux 1
Master Cryptologie et Sécurité Informatique

Quel avenir pour les Trusted Platform Module ?

FLOISSAC Noémie

Talence, le 20 mars 2009

Table des matières

Introduction

Qui ne rêve pas d'avoir un ordinateur qui s'assure être intègre et digne de confiance ? C'est le défi que tente de relever le groupe Trusted Computing Group en élaborant, entre autre, une plateforme[1] dont la sécurité repose sur un composant physique, la puce Trusted Platform Module. Cette puce a pour vocation d'être une zone de stockage "inviolable" permettant de conserver des données sensibles telles que des clés cryptographiques. Après avoir exposé dans une première partie les principes de *l'ordinateur de confiance* selon le groupe TCG et son utilisation, nous approfondirons les mécanismes de sécurité sur lesquels repose la "confiance" de l'ordinateur.

Comme nous le détaillerons dans une troisième partie, la mise en place d'une telle plateforme sécurisée est simple et à présent à la portée de tous grâce à une large diffusion de puces TPM intégrées dans les ordinateurs portables. Cependant, et malgré la compatibilité des spécifications proposées par le groupe TCG avec les systèmes d'exploitation usuels tels que Windows ou Linux, la plateforme reste peu utilisée.

Nous pouvons nous demander quelle sera l'évolution de cet ordinateur de confiance : a-t'on enfin à notre disposition un ordinateur sûr ou bien cela reste-t'il encore une utopie ? En effet, la plateforme sécurisée telle qu'elle se présente à l'heure actuelle ne semble pas inviolable. Il existe, en plus des attaques physiques sur le composant TPM, des attaques logicielles permettant de compromettre la sécurité de la plateforme. Nous reviendrons dans une dernière partie sur ces attaques et leurs contre-mesures éventuelles.

[1] ensemble de composants matériels et logiciels, système d'exploitation et applications.

Chapitre 1

Présentation générale

Le goupe TCG a proposé, selon ses principes, un ordinateur de *confiance*. Nous verrons dans ce chapitre les critères de confiance qu'a souhaité développer le groupe, par quels moyens il y est parvenu et quelles en sont les utilisations possibles.

1.1 Le groupe TCG

Le groupe Trusted Computing Group[1], formé en 2003, est un consortium à but non lucratif. Son objectif est de sécuriser différentes plateformes, par le biais d'une sécurité physique et logicielle. Le groupe est composé de nombreuses entreprises. Pour les ordinateurs, il regroupe par exemple plus de 140 membres parmi lesquels AMD, IBM, Intel, Lenovo, Microsoft, Sun ou encore des universités.

Les entreprises qui composent le groupe étant parfois concurrentes entre elles sur leur marché, le groupe TCG n'a pas imposé une solution unique d'ordinateur de confiance. Il a cependant développé les grandes lignes à respecter pour les constructeurs et développeurs, à travers des spécifications. Le groupe TCG s'est divisé en 14 groupes de travail qui développent des spécifications correspondant à différentes plateformes ou sous-ensembles de plateformes sécurisées parmi lesquels :

- Storage
- Mobile
- Authentification (par biométrie et cartes à puce)
- Server
- Network
- **PC client**
- **Trusted Plateform Module**
- **Software Stack**

[1] https ://www.trustedcomputinggroup.org

Nous nous intéresserons ici aux spécifications développées par les groupes de travail *PC client*, *TPM* et *Software Stack* qui spécifient un "ordinateur de confiance". Le premier, PC Client[1], définit l'architecture de la plateforme sécurisée et les principes de confiance qui l'accompagnent, tandis que la spécification TPM[2] se focalise sur la puce en elle-même et les composants qu'elle doit comporter. Enfin, le groupe de travail Software Stack[3] a développé les spécifications correspondant à la pile logicielle qui fait le lien entre la plateforme et l'utilisateur.

1.2 Les grandes lignes

Malgré leurs diversités, les membres du groupe TCG doivent poursuivre un même objectif, à savoir la mise en place d'un ordinateur de confiance. Pour cela, il leur a fallu dans un premier temps définir les grandes lignes communes. La notion de "confiance" par exemple, qui est une notion assez subjective, a dû être plus précisément explicitée. Ils se sont de plus accordés sur le choix d'une sécurité logicielle ou matérielle, la sécurité matérielle ayant prévalu avec l'introduction de la puce TPM. Une architecture particulière a donc été élaborée à laquelle la description des différentes catégories d'utilisateurs de la puce s'est ajoutée.

1.2.1 La confiance

Les spécifications proposées décrivent les deux approches complémentaires du groupe TCG concernant un ordinateur de confiance. La première consiste à assurer à un utilisateur (mais aussi à n'importe quelle entité, par exemple une banque dont l'utilisateur est client), que l'ordinateur fonctionne sous une certaine configuration. Pour cela, des mesures sont réalisées à chaque démarrage de l'ordinateur et peuvent être rapportées à toute entité qui le désire.

La seconde approche assure une utilisation sûre des outils cryptographiques permettant de communiquer de façon sécurisée. Ces outils sont en particulier implémentés dans la puce TPM. Il s'agit de programmes de chiffrement/déchiffrement ou de signatures mais aussi de génération de clés RSA ou de nombres aléatoires.

La confiance provient ainsi, à la fois, du bon fonctionnement de l'ordinateur et de la sécurité fournie par les outils cryptographiques.

1.2.2 Architecture

La confiance souhaitée par le groupe TCG a été matérialisée au travers d'une architecture de référence d'une plateforme pour le PC [1]. Elle est

illustrée Figure 1.1. À une architecture classique, s'ajoute la puce TPM, composant physique intégré à la carte mère, sur laquelle repose toute la sécurité de la plateforme. Des routines Core Root of Trust for Measurement (CRTM), routines spécifiques à la plateforme, sont intégrées au BIOS.

Le système d'exploitation est le lien entre la partie physique et les applications. Une pile logicielle, appelée Software Stack, est chargée de faire le lien entre l'utilisateur et la plateforme. Viennent ensuite les applications qui offrent des services à l'utilisateur comme des outils cryptographiques par exemple.

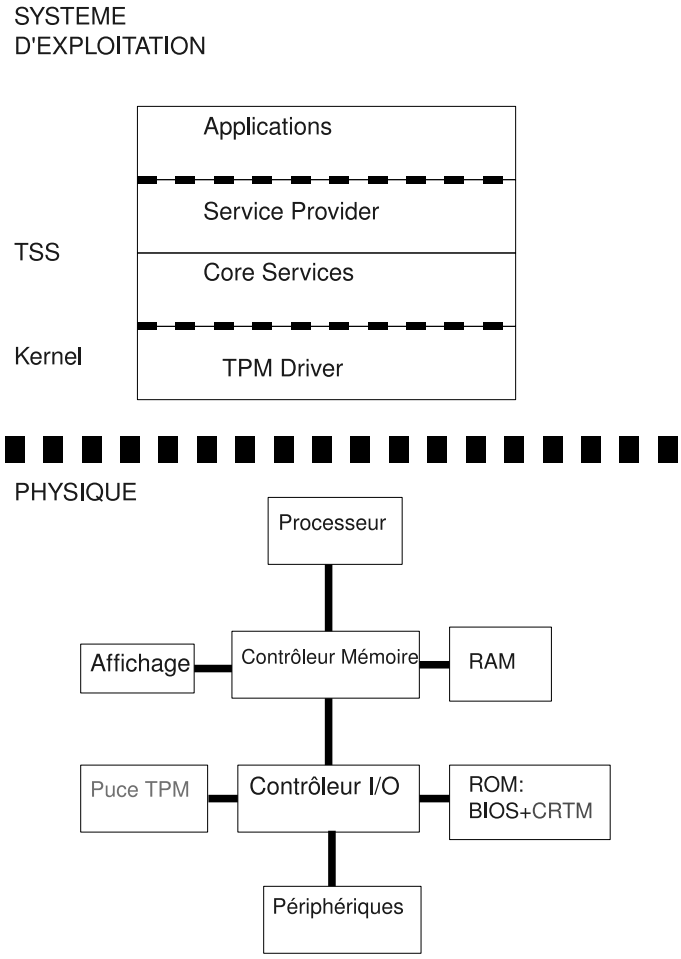

FIG. 1.1 – Architecture d'une plateforme sécurisée pour le PC

1.2.3 Les acteurs de la plateforme

Cette architecture sécurisée se doit de fournir des accès, limités ou non, aux zones sensibles de la plateforme selon le statut de l'utilisateur qui requiert ses services. Pour cela, le groupe TCG a défini trois status d'utilisateur :

- présent physiquement sur la plateforme
- propriétaire de la plateforme
- ni l'un, ni l'autre

Les deux premiers statuts peuvent être cumulables. On peut en effet être simultanément propriétaire de la plateforme et présent physiquement sur celle-ci.

Présence physique sur la plateforme

La puce TPM permet de détecter une présence physique d'un utilisateur par le biais d'un *détecteur de présence*. Une spécification de ce détecteur est d'ailleurs fournie par le groupe TCG [4].
Cela peut être un composant physique relié à la puce, comme un bouton sur le devant de la plateforme, et sur lequel une personne physiquement présente a la possibilité d'appuyer et donc d'informer la puce de sa présence. Cependant, la solution recommandée par le groupe est que les commandes nécessitant la présence physique d'un utilisateur sur la plateforme ne puissent être lancées qu'avant le chargement du système d'exploitation.

Une personne connectée physiquement a le droit d'allouer ou désallouer la puce à un propriétaire. Elle n'a cependant pas accès aux fonctions cryptographiques de la puce.

Allocation de la plateforme

Le second statut est d'être propriétaire de la plateforme. Pour le devenir, il faut exécuter le commande tpm_takeownership, le tout en étant physiquement connecté à la plateforme. Cette commande génère un secret partagé de 160 bits entre la puce et le propriétaire. Certaines opérations dans la puce requièrent une authentification du propriétaire, qui doit alors prouver qu'il connaît le secret partagé, après l'authentification, une session est établie. Deux protocoles sont spécifiés par le groupe TCG pour établir une session, basés sur le protocole HMAC[2].

[2]Hash Message Authentication Code

Lorsqu'un propriétaire est authentifié, une clé de session issue du secret partagé haché avec un nombre alatoire est utilisée pour chiffrer les communications entre l'utilisateur et la puce. Il a alors accès à des fonctionnalités qui requièrent le statut de propriétaire. Il peut par exemple stocker de façon sécurisée des clés en dehors de la puce.

Il peut ne peut pas y avoir qu'un propriétaire de la plateforme. La création d'un nouveau propriétaire supprime le propriétaire précedent. En revanche, si plusieurs personnes se mettent d'accord sur le secret partagé, alors ils sont plusieurs à être un même propriétaire.

Autres cas

On peut utiliser une plateforme TPM sans être physiquement connecté à la machine ni être propriétaire. Un utilisateur peut par exemple demander à une plateforme distante dans quelle configuration elle se trouve ou bien encore un certificat de la puce. Ces requêtes ne nécessitent aucun droit particulier.

Il n'y a pas de "super utilisateur". Tout dépend de l'utilisation que l'on souhaite faire de la plateforme. Par exemple, une entreprise, dont seule la direction possède un accès physique à la machine, peut distribuer le mot de passe partagé à certains employés sans pour autant le connaître. Ces derniers employés sont alors propriétaires de la plateforme mais n'auront pas accès à la plateforme. Ni la direction, ni les employés, ne sont super utilisateurs.

Il y a de plus des commandes accessibles à la fois à une personne présente physiquement et à un propriétaire comme par exemple l'action de réinitialiser la puce.

1.3 Utilisations d'une plateforme sécurisée

Nous avons vu que le principe de l'ordinateur de confiance n'est pas d'être un "super bouclier" contre les menaces informatiques comme on pourrait le penser. Cependant, les utilisations que l'on peut en faire sont diverses, aussi bien celles d'une plateforme personnelle et locale, que celles d'un correspondant à distance.

1.3.1 Utilisation d'une plateforme personnelle

Intégrité de la plateforme

Un utilisateur peut vouloir s'assurer de l'intégrité de sa plateforme. À chaque boot du système, des mesures sont effectuées et comparées aux an-

ciennes, ce qui permet de déterminer si une modification importante de la plateforme a eu lieu. Par exemple, tout malware modifiant le système d'exploitation est détecté. Ces mesures sont stockées dans la puce de manière sécurisée dans des registres appelés registres *Platform Configuration Register* (PCR).

Utilisation de la puce TPM

Que ce soit dans les protocoles SSL[3] employés pour des paiements en ligne, pour la signature de mises à jour de logiciels ou encore simplement pour le chiffrement de fichiers, les clés cryptographiques interviennent de plus en plus dans l'informatique du quotidien. Il est donc nécessaire de pouvoir les stocker de manière sécurisée et la puce TPM répond parfaitement à cette attente. Les clés peuvent être stockées ou non dans la puce TPM. Celles qui ne le sont pas sont tout de même chiffrées à l'aide d'une clé qui réside dans la puce TPM. La puce TPM assure donc le stockage protégé des clés de chiffrement de l'utilisateur.

La puce comporte, en plus de zones de mémoire de stockage, une implémentation de générateur de paires de clés cryptographiques RSA, la fonction de hachage SHA-1, ou encore un générateur pseudo-alatoire. Des applications telles que BitLocker[4], un logiciel propriétaire de Windows de protection du disque dur, reposent fortement sur l'emploi de puces TPM.

1.3.2 Utilisation d'une plateforme distance

On vient de voir qu'il est avantageux de posséder une plateforme sécurisée, il l'est tout autant que notre interlocuteur en possède une à son tour.

Grâce à la plateforme, on a accès à des clés sécurisées et donc on peut pratiquer le chiffrement et déchiffrement mais aussi la signature et la vérification de signature. Cependant, on peut faire encore mieux ; lorsque deux plateformes sécurisées communiquent, elles peuvent envoyer des *messages scellés*. Un message scellé contient :

- le message chiffré avec une clé symétrique
- la clé symétrique chiffrée à partir de quelques registres PCR (correspondant aux valeurs mesurées de la plateforme)
- le tout est chiffré de manière asymétrique avec la partie privée d'une clé asymétrique.

La plateforme destinatrice ne peut déchiffrer la clé symétrique que si elle est dans la configuration correspondant aux valeurs des mesures de la plateforme reçues.

[3] Secure Socket Layer
[4] http ://www.bitlocker.com/

Les messages scellés peuvent par exemple intervenir pour la vérification d'un serveur de paiement. Si le serveur n'est pas dans une configuration particulière, alors le déchiffrement des données bancaires ne peut s'effectuer.

Cette utilisation pourrait s'avérer être une solution au problème de piratage de logiciels ou de données propriétaires. Ainsi, les fichiers de musique et vidéo peuvent être scellés par l'industrie correspondante. Alors, seule une configuration particulière de la plateforme cliente, par exemple la présence d'un logiciel propriétaire payant, permet de déchiffrer (et donc lire) le fichier.

On peut aussi ajouter des valeurs mesurées à une signature pour informer le destinataire de l'état de la plateforme au moment de l'envoi du message. Ces valeurs sont alors ajoutées au message avant le hash et le chiffrement.

1.3.3 Quelques chiffres

Le groupe TCG a mis en place un ordinateur de confiance dont, nous l'avons vu, les utilisations sont diverses. Cependant, si on interroge notre entourage s'il connaît et s'il utilise la technologie des TPM, il est fort probable que le nombre de réponses positives soit très réduit, en dehors peut-être de personnes spécialisées en sécurité informatique. Pour ce qui est des entreprises, cette technologie semble cependant se mettre doucement en place.

Une étude a été réalisée en 2008 par le groupe Aberdeen Group[5] afin de déterminer quelle utilistation est faite des plateformes TPM[5]. Il en résulte que :

- 53% des ordinateurs en 2008 sont équipés d'une technologie TPM ce qui représente environ 200 millions d'ordinateurs.
- 250 millions d'ordinateurs équipés sont attendus en 2010.

Si les puces sont présentes dans les ordinateurs, elles ne sont pas nécessairement utilisées. Il n'y a pas à l'heure actuelle de chiffres concernant le pourcentage de la population qui utilise cette technologie mais l'on peut considérer qu'il est assez faible. Cependant, un sondage sur l'utilisation de la technologie des TPM par les entreprises a été réalisé. Environ 30% des entreprises sondées déclarent utiliser une politique de sécurité basée sur les TPM. Les résultats de ce sondage sont illustrés par la Figure 1.2. Il en ressort que les utilisations qui sont faites sont essentiellement l'authentification avec près de 90% des utilisations et l'attestation (environ 80%).

[5]http ://www.aberdeen.com/

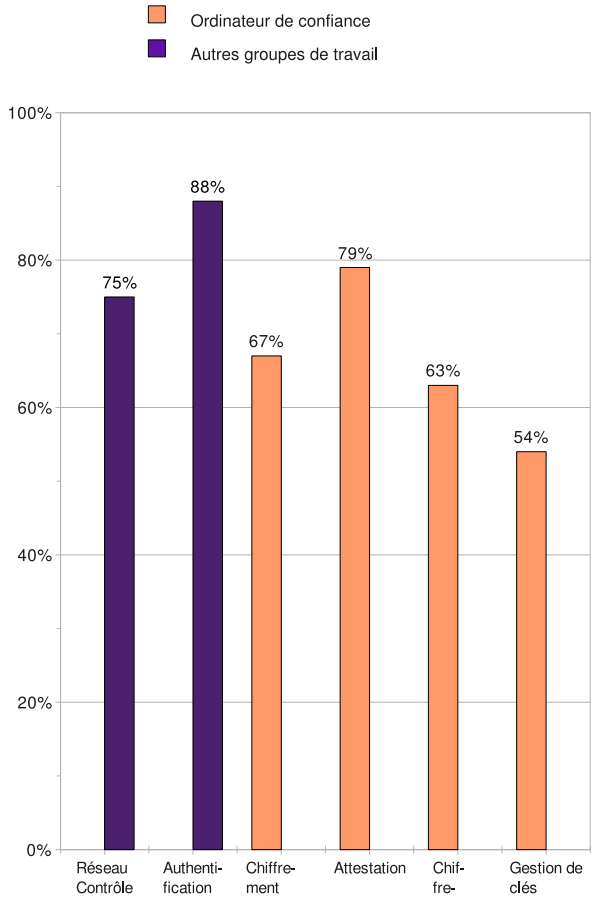

FIG. 1.2 – Domaines d'utilisation par les entreprises qui emploient la technologie TPM en 2008 (source :Aberdeen Group)

Les entreprises qui ont déclaré utiliser les TPM sont essentiellement des entreprises de finance ou de haute technologies. Si, comme le prédit le groupe Aberdeen, la technologie des TPM est destinée à s'étendre, alors les secteurs d'activités d'entreprises utilisant les TPM se diversifieront.

1.4 Les évolutions de la plateforme

A l'heure actuelle, la technologie TPM n'est pas encore très employée et reste peu connue. Le groupe TCG est pourtant confiant quant à l'avenir de sa technologie.

Dans cette section, nous nous arrêterons sur l'histoire des TPM : avec un passé mouvementé et un présent qui semble prendre forme, nous étudierons

quel avenir se dessine pour cette technologie.

1.4.1 Un passé mouvementé

Le fait que très peu de personnes utilisent aujourd'hui les capacités TPM de son ordinateur est peut-être lié au passé de *l'ordinateur de confiance*. En effet, le groupe TCG est le successeur du groupe Trusted Computing Platform Alliance(TCPA), associé à la polémique "Palladium"[6].

Le projet TCPA a vu le jour en 1999. Il a été conçu majoritairement par Intel et IBM. Les principes de l'ordinateur de confiance sont les mêmes que ceux du groupe TCG. Celui-ci, qui a simplement changé de nom et qui s'est agrandi, a en effet repris intégralement les spécifications développées par TCPA comme spécifications pour la plateforme TPM.

Microsoft a voulu incorporer la technologie TCPA à ses machines dans un projet "Palladium" mais cela a déclanché une polémique sans précédent. Les raisons de cette polémique sont :

- Une communication discrète : il n'y a pas eu, ou peu, de spécifications sur le "Palladium".
- Des retournements de situation : du jour au lendemain "Palladium" devient NGSCB.
- Des objectifs flous qui ont déclenché des rumeurs : un désir de maîtrise des configurations des postes de travail ?
- Un sentiment de monopôle : obligation d'exploiter des informations avec des logiciels certifiés comme Microsoft Word par exemple.

Parce que les liens entre TCPA et "Palladium" étaient peu clairs, la polémique a affecté le projet du groupe TCPA, qui a dû changer de nom. L'affaire semble avoir été oubliée depuis, et "Palladium" n'est plus.

1.4.2 A ce jour

L'actualité du groupe TCG concerne essentiellement le groupe de travail *Storage*. Les spécifications sont depuis peu disponibles(janvier 2009). On peut donc envisager que dans un avenir proche, des technologies basées sur ces spécifications voient le jour.

Concernant l'ordinateur de confiance, le groupe TCG a peu fait évoluer ses spécifications depuis 2003. En effet, depuis la version 1.1b, qui, comme nous l'avons vu précédemment, est issue du travail du groupe TCPA, il n'y a eu que quelques fonctionnalités qui ont été ajoutées dans une version 1.2 [7]. Cette dernière version date de 2003 et n'a pas subi de grands changements depuis, si ce n'est quelques précisions sur l'implémentation de

certaines fonctionnalités, cryptographiques par exemple. La plateforme de confiance du groupe TCG semble donc stable. De plus, depuis juillet 2008, la version 1.2 des TPM est devenue une norme ISO/IEC 11889.

1.4.3 Et demain ?

Si l'évolution des spécifications de la plateforme TPM est ralentie, la technologie TPM n'est toutefois pas à son point mort. Elle est reprise par un grand nombre de recherches aussi diverses que variées, et suscite un grand intéret dans la communauté des chercheurs :

- La possibilité de stopper les rootkits[6][8, 9]
- Le projet Hight Assurance Platform (HAP) de la NSA[7], qui a pour objectif de définir une structure pour la prochaine génération de plateformes de confiance, envisage d'utiliser la puce TPM[8]
- La conférence *Future of Trust in Computing* qui s'est déroulée à Berlin en juillet dernier a regroupé de nombreux chercheurs.

L'attrait pour les *Trusted Computers* ne cesse donc de croître, ce qui laisse présager un avenir prospère pour la technologie TPM.

[6] ensemble de programmes destinés à prendre et à garder le contrôle d'une machine sans que l'utilisateur légitime ne s'en aper coive

[7] National Security Agency

[8] http ://www.nsa.gov/ia/programs/h_a_p/index.shtml

Chapitre 2

Une plateforme sûre

Nous avons vu dans le chapitre précédent que la plateforme TPM, telle qu'elle a été conçue par le groupe TCG, permet des utilisations multiples, et ce grâce à sa principale qualité : le sécurité. Mais sur quoi repose cette sécurité ?

La "confiance" en la plateforme se divise en deux entités complémentaires. On ne peut faire du chiffrement que si on a la certitude que les ordinateurs, aux deux extrémités de la communication, ne sont pas compromis ; sinon le chiffrement est inutile, d'où le mécanisme d'attestation de la plateforme. Réciproquement, deux ordinateurs aussi sains soient-ils ne peuvent pas communiquer des données sensibles en clair, la cryptographie est alors nécessaire. Le groupe TCG s'appuie sur ces deux mécanismes pour son ordinateur de confiance.

2.1 Une puce cryptographique

Une très large partie de la sécurité de la plateforme de confiance repose sur un composant physique : la puce TPM. Celle-ci sert de zone de stockage sécurisée mais c'est aussi un composant dédié à la cryptographie et à la gestion des clés qu'elle suggère.

2.1.1 Architecture de la puce

La puce TPM est un microprocesseur qui peut être intégré à la carte mère d'un ordinateur. Elle est placée entre le contrôleur de mémoire IO et le super IO (voir Figure 1.1).

Il existe plusieurs fabricants de puce TPM parmi lesquels : Atmel, Infineon, Broadcom ou encore Stmicroelectronic. Quelque soit ce fabricant, toute puce TPM doit répondre aux spécifications définies par le groupe

TCG pour des questions d'interopérabilité [2]. Elles doivent notamment comporter les algorithmes cryptographiques choisis par le groupe. La figure 2.1 présente l'architecture d'une puce TPM telle qu'elle est spécifiée par le groupe TCG.

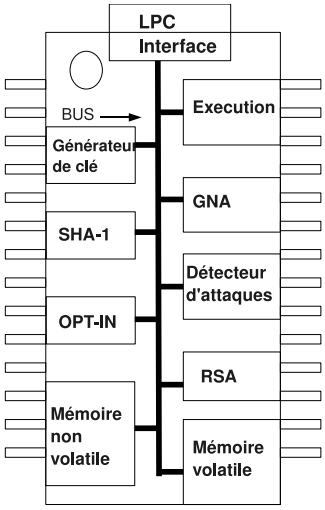

FIG. 2.1 – Architecture de la puce TPM

Communication

La puce, en tant que composant de la carte mère communique avec cette dernière. C'est le *Low Pin Count interface :* [10] qui fait le lien entre les deux parties. On peut considérer l'interface comme un I/O sécurisé. Les messages pour le bus externe ou interne sont respectivement encodés ou décodés.

Un *bus LPC* (33 MHz) est la voie de communication des messages. Associé avec le Opt-In pour réaliser un contrôle d'accès, il achemine au composant destinataire tout message qui lui est destiné.

Comme son nom l'indique, l'*Exécution* exécute les commandes reçues de l'interface LPC. Il est associé à un processeur 8 ou 16 bits.

Le *Opt-In* est employé pour déterminer l'état de la puce TPM. Il maintient les flags associés aux différents états de la puce ce qui permet une politique de contrôle d'accès aux différentes fonctionnalités de la puce. En effet, une puce peut être dans différents états. Pour décrire l'état d'une puce TPM, on emploie trois caractéristiques : enable/disable, activated/deactivated, owned/unowned. Ce qui donne huit états possibles pour la puce. A

chaque état est associée une politique de sécurité. Par défaut, la puce est dans l'état (disable ;deactivated ;unowned).

- owned/unowned : Une puce qui a authentifié son propriétaire est dans l'état "owned". Dans le cas contraire, elle reste dans l'état "unowned".
- enable/disable : Pour que la puce puisse passer dans l'état "owned" après authentification, la puce doit au préalable être dans l'êtat "enable". Quand la puce est disabled, elle restreint toutes les oprations sauf la capacité à atteindre les registres PCR.
- activated/deactivated : On peut associer cet état à la possibilité ou non d'accéder aux mesures de la plateforme. Dans l'état "activated", l'accès aux registres PCR est autorisé contrairement à l'état "deactivated".

Mémoire

La puce TPM possède deux sortes de mémoire. Une *mémoire volatile* stocke des données qui peuvent être effacées. La *mémoire non-volatile*, au contraire, conserve les données même lorsque l'ordinateur redémarre.

Les zones de mémoire servent en partie pour le stockage des clés. Comme nous le verrons dans la section 2.1.2, les différents types de clés ne sont pas nécessairement stockées dans la même zone de mémoire. La mémoire sert aussi à stocker les valeurs de mesures de la plateforme dans les registres PCR. Il y a au minimum 16 registres PCR, numérotés de 0 à 15 qui sont dans la mémoire non-volatile.

Outils de sécurité

Une puce TPM, pour une question d'interopérabilité doit comporter des *fonctions cryptographique*s prédéfinies, spécifiées par le groupe TCG mais peut supporter d'autres algorithmes cryptographiques (symétriques par exemple). Le choix du standard asymétrique RSA fait partie des spécifications. L'implémentation doit s'accorder avec la norme PKCS #1 [11]. RSA est utilisé pour le chiffrement ou la signature. La puce fournit également un générateur de clé RSA en accord avec la norme P1363 [12].

De plus, la puce TPM doit supporter une *fonction de hachage cryptographique*. Dans la dernière spécification, qui date de juillet 2007, le "standard" retenu est SHA-1 [13], bien que cette fonction de hachage ne soit plus considérée comme "sûre" à l'heure actuelle. Les hachés mesurent 160 bits.

Enfin, un *générateur de nombres aléatoires* (GNA), est intégré à la puce TPM. Il doit être capable de produire des nombres "aléatoires" de 32 bits. Il est par exemple utilisé pour la génération de clés. L'obtention d'un nombre aléatoire suit ce procédé :

1. Le GNA possède une zone de mémoire non-volatile dans laquelle est stocké un nombre aléatoire généré lors de la fabrication du GNA.

2. Le GNA doit comporter une source d'entropie. Il peut s'agir par exemple d'un générateur de bruit ou bien encore l'entropie peut être produite en fonction des mouvements de la souris,...

3. Si l'entropie de la source est biaisée (ie : la probabilité d'avoir un 1 est différente de celle d'avoir un 0) alors le biais est corrigé.

4. Le nombre associé au GNA est combiné à la production de la source d'entropie avant d'être haché par la fonction de hachage SHA-1.

5. Le résultat du haché est le nombre aléatoire produit par le GNA.

Sécurité de la puce

En tant que composant physique, la puce se doit de résister à des attaques physiques. Pour cela, des composants sont intgrés à la puce selon le constructeur. La puce AT97SC3202 d'Atmel[14] comporte par exemple :

- Un Real-Time-Clock (RTC) intgré à la puce TPM. Il utilise une batterie externe qui permet d'indiquer si la puce a été otée d'un PC et peut agir en conséquence.
- Circuit de prévention d'attaques qui détecte toute tentative de lecture de la puce.
- Une couche de métal de protection au dessus du circuit
- Un bus interne chiffré
- Des déffenses contre les *timming attacks* et les *power attacks*

Les puces TPM doivent être évaluées par des organismes accrédités selon les Critères Communs [15], c'est à dire toute une série d'attaques sont effectuées sur le composant pour déterminer sa résistance. La puce TPM obtient ainsi un certificat d'évaluation déterminant son niveau de sécurité testé. Il existe différents niveaux d'évaluation Evaluation Assurance Level (EAL) qui vont de 1 à 7 selon un ordre croissant de résistance aux attaques. Le groupe TCG recommande au moins le niveau EAL3 pour les puces TPM.

Une autre certification que la puce doit pouvoir obtenir est la certification FIPS 140-2 [16] qui définit un niveau de sécurité pour les modules cryptographiques.

2.1.2 Les clés cryptographiques

La gestion des clés dans la puce est très importante pour la sécurité de la puce. Il existe une hiérarchie des clés. La Figure 2.2 représente l'organisation des clés dans la puce.

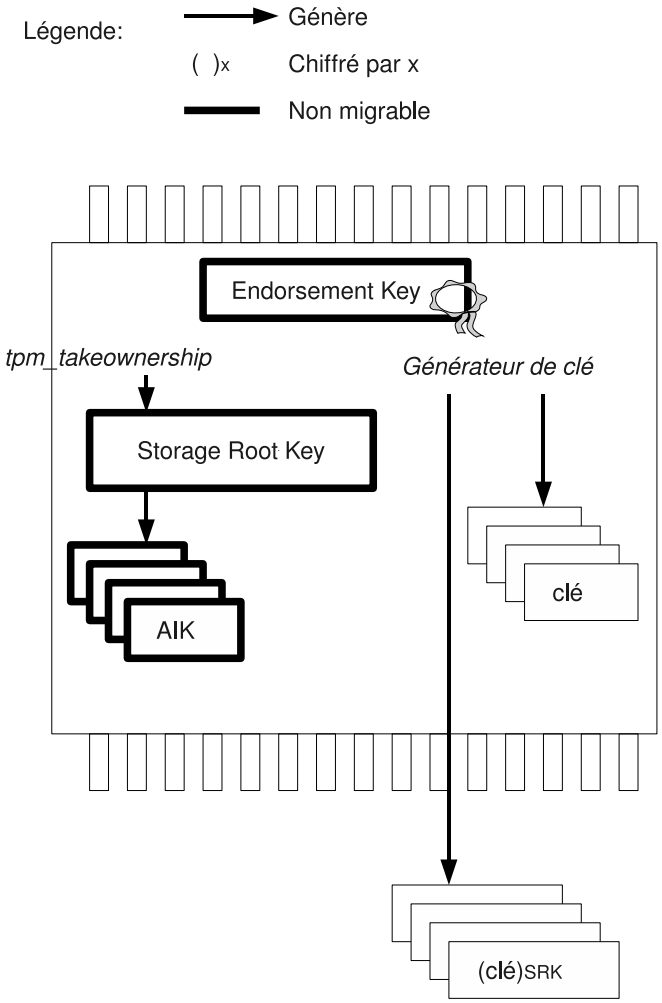

FIG. 2.2 – Gestion des clés

Endorsement Key La clé EK ou encore *Endorsement Key* est un couple de clés RSA de 2048 bits. Elle est générée par le fabricant de la puce lors de sa création. La partie publique de la clé, PUBEK, est diffusée dans le certificat de la puce et de la plateforme. La partie privée, PRIVEK, est stockée dans une zone de mémoire non-volatile de la puce TPM et n'en sort jamais. La clé PRIVEK peut servir pour déchiffrer des messages qui auraient été chiffrés

par la clé PUBEK mais **n'est en aucun cas utilisée pour signer ou chiffrer des messages**. Cette assertion est très importante car c'est sur elle que repose toute la sécurité cryptographique de la puce et donc de la plateforme. En effet, il y a une hiérarchie dans les clés et la clé EK en est au sommet.

Storage Root Key Une autre clé importante est la clé *Storage Root Key* (SRK). Elle est aussi stockée dans la mémoire non-volatile de la puce mais est générée par la puce elle-même. La clé SRK est générée lors de la commande tpm_takeownership et est liée à un propriétaire de la plateforme. A chaque changement de propriétaire, l'ancienne clé SRK est supprimée et une nouvelle est générée.

La clé SRK sert à gérer les clés stockées en dehors de la puce. En effet, la capacité de mémoire de la puce TPM étant limitée, des clés cryptographiques peuvent être stockées en dehors de la puce après avoir été préalablement chiffrées par la SRK. Pour pouvoir être transférées et utilisées par une autre TPM, les clés doivent avoir l'attribut *migrable*. Celles qui n'ont pas cet attribut, donc *non-migrable*, ne peuvent pas être transférées. C'est le cas de la SRK et de la EK bien sûr, mais aussi des clés AIK.

Attribut Identity Key Les *Attribut Identity Keys* (AIK) sont des clés utilisées uniquement pour signer des données originaires de la puce comme des registres PCR par exemple. Ces clés asymétriques ont une taille fixée à 2048 bits.

À chaque clé AIK est associé un certificat délivré par une Autorité de Certification dans lequel est enregistrée la partie publique de la clé. Ainsi, toute personne qui reçoit un message signé par une clé AIK peut s'assurer de l'authenticité de son expéditeur. Lors de l'attestation de la plateforme à distance par exemple, le certificat de la clé AIK ayant servi à signer les registres PCR est envoyé au challenger.

Pour obtenir un certificat de clé AIK, la puce TPM, après avoir généré un couple clé publique/privée demande à la CA de certifier la clé publique. Elle lui envoie une requête contenant la clé publique ainsi que son Endorsement Certificate. La requête est chiffrée avec la clé publique de la CA. Le certificat généré par la CA, contenant la clé publique AIK, est renvoyé à la puce TPM, chiffré avec la cl PUBEK de la puce. Ainsi, seule la puce TPM peut déchiffrer le certificat fourni par la CA grâce à sa clé PRIVEK.

Autres clés En plus de ces clés, il existe différentes clés qui peuvent être générées par le générateur de clé de la puce ou importées depuis l'extérieur.

Ces clés peuvent servir à du chiffrement/déchiffrement ou encore de la signature. Elles peuvent être symétriques ou asymétriques, migrables ou non.

2.2 Attestation de la plateforme

Le second principe de confiance défini par le groupe TCG consiste en la capacité de la plateforme à attester de la configuration dans laquelle elle se trouve, localement mais aussi à distance. Pour cela, elle mesure toutes les parties de l'architecture qui doivent être attestées, stocke les informations et les reporte à l'entité qui le désire, le tout de manière sécurisée. Cette section présente les mécanismes employés afin d'assurer cette "confiance".

Racines de confiance

On appelle *racine de confiance* (en anglais : *root of trust*) de la plateforme, un élément de la plateforme qui ne change pas au cours du temps. Ces parties sont certifiées par des organismes spécialisés ou par le fabricant lui-même et on peut donc être sûr de leur intégrité.

Il existe trois racines de confiance dans la spécification proposée par le groupe TCG :
- RTM (Root of Trust for Measurement) : racine de confiance pour la mesure
- RTS (Root of Trust for Storage) : racine de confiance pour le stockage
- RTR (Root of Trust for Reporting) : racine de confiance pour le report

2.2.1 Mesures de la plateforme

A chaque (re)démarrage de l'ordinateur, la plateforme est intégralement mesurée. Le point de départ des mesures est une RTM, une partie du BIOS qu'il est impossible de modifier. Elle comporte les routines Core Root of Trust for Measurement (CTRM) qui sont intégrées dans cette racine. Elles sont chargées de démarrer une chaîne de vérification à partir du BIOS sécurisé.

Un ordinateur démarre de la façon suivante :
- le BIOS
- le boot loader
- l'os loader
- le système d'exploitation

Les mesures de la plateforme suivent la chaîne de démarrage de l'ordinateur, le point de départ de la chaîne étant la racine de confiance pour la mesure.

Chaque élément de la chaîne mesure le maillon suivant (voir Figure 2.3). La mesure consiste en un hachage cryptographique, par le biais de la fonction SHA-1, du code d'exécution du prochain élément de la chaîne. Cette valeur est stockée dans la puce TPM et, seulement alors, l'exécution du maillon suivant est possible. Par exemple, le BIOS sécurisé, considéré comme RTM et donc comme premier maillon de la chaîne, mesure le reste du BIOS. Les mesures sont sauvées et le BIOS est exécuté.

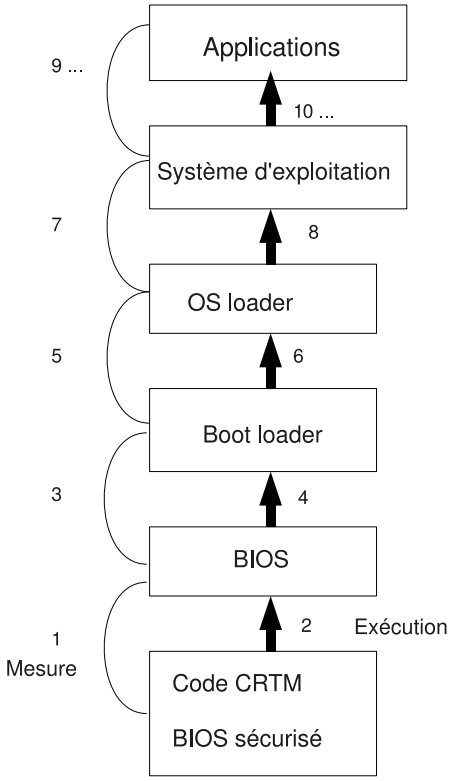

FIG. 2.3 – Chaîne de mesure au démarrage de l'ordinateur

2.2.2 Stockage des mesures

Les mesures effectuées sont stockées de deux façons différentes.

Elles sont tout d'abord stockées de manière protégée dans la puce TPM, dans des registres spéciaux appelés Platform Configuration Register (PCR). La puce TPM est en effet considérée comme une racine de confiance pour le stockage, ce qui assure donc la protection des données. Aucun élément de la chaîne ne peut modifier le registre PCR qui contient la mesure de sa valeur. Les valeurs sont stockées suivant le schéma :
PCR[n+1] := SHA1 (PCR[n] + nouvelles mesures) avec + représentant la concaténation.

Ces mêmes valeurs, non hachées, ainsi les anciennes valeurs de registres PCR sont appelées *Stored Measurment Log* (SML). Elles peuvent être stockées dans l'espace de mémoire virtuel mais ce n'est pas obligatoire. Elles ne sont pas sécurisées mais servent pour l'attestation de la plateforme ou de logiciels comme élément de comparaison avec les regitres PCR.

2.2.3 Report des mesures et attestation

Une fois que le système d'exploitation est lancé, l'utilisateur peut attester sa plateforme et n'importe quel programme chargé sur sa machine. Pour cela, il utilise les valeurs mesurées et stockées dans la puce TPM.

La plateforme fait la différence entre le report d'information et l'attestation. Le *report* consiste simplement à envoyer l'information à un tiers. Il peut s'agir par exemple de l'envoi d'un certificat à un challenger. L'*attestation*, qui peut être à distance, est utilisée dans le cas où une entité, distante ou non, veut s'assurer de la configuration actuelle et de l'identité de la plateforme.

Le protocole d'attestation est le suivant :

1. La machine distante ou l'utilisateur de la plateforme, qui est appelé aussi challenger, envoie une requête à la plateforme. Cette requête contient les numéros de registres PCR qu'il souhaite obtenir ainsi qu'un nonce[1].

2. La plateforme récupère les valeurs SML.

3. Les registres PCR demandés sont signés avec le nonce, à l'aide d'une clé privée RSA Attestation Identity Key générée par la puce TPM. Un certificat, fourni par une autorité de certification, comporte la partie publique de la clé AIK.
   Le nonce sert à éviter le rejeu. En effet, si le registre PCR signé était envoyé ainsi au challenger, toute personne interceptant le paquet de

[1] nombre aléatoire avec une validité qui porte sur une courte période de temps

réponse au challenger pourrait alors l'envoyer à n'importe quel autre challenger.

4. La plateforme envoie au challenger les valeurs SML, le certificat de la clé AIK et le message signé (les registres PCR et le nonce).

5. Pour vérifier la plateforme, le challenger déchiffre les registres PCR à l'aide de la clé publique contenue dans le certificat. La signature assure que les valeurs PCR sont bien issues de la puce TPM.

6. Le challenger hache les valeurs SML et les compare avec les valeurs PCR qu'il a précédemment déchiffrées et auxquelles il a été oté le nonce. Si elles sont identiques, alors la plateforme est attestée.

L'attestation à distance repose sur la confiance d'un tiers, l'autorité de certification, et sur la capacité de la puce TPM à créer des clés AIK robustes. La puce TPM est donc considérée comme une racine de confiance pour le report (RTR).

2.2.4 Les certificats

La sécurité dans la plateforme repose en grande partie sur des éléments inaltérables et dont on doit être sûr, les racines de confiance en font partie. La certification de ces éléments, délivrés par des "autorités de confiance" est indispensable à la sécurité de la plateforme. Il existe divers certificats, délivrés par diverses autorités de certification.

Certificat EK Une puce TPM est fabriquée avec une unique paire de clé RSA, appelée Endorsement Key (EK). La partie privée de la clé EK est contenue dans la puce TPM et n'en sort à aucun moment. C'est sur ce point important que toute la sécurité de la puce repose. C'est le fabricant de la puce qui génére le couple clé privée/clé publique et le certificat associé. Il contient, entre autre, le nom du fabricant, la version du TPM et la clé publique EK.

Certificat de conformité Il certifie que la puce TPM (ou plateforme) a été évaluée par quelqu'un de confiance. Il contient le nom de l'évaluateur, le fabricant de la puce (ou plateforme), la version de la puce (ou plateforme).

Certificat de la plateforme Il est généré par le fabricant de la plateforme. Il contient le nom du fabricant de la plateforme, le numéro de la plateforme et sa version, les certificats EK et de conformité.

Certificat de validation Le groupe TCG estime qu'il est préférable de pouvoir attester de l'intégrité de certains composants (disque dur, processeurs, logiciels,...). Pour cela, le fabricant du composant, une fois que le composant a fini d'être testé, réalise une mesure. Le certificat contient le nom de l'organisme qui a validé le certificat, le nom du fabricant, la valeur de mesure.

Certificat d'une Attestation d'Identité (AI) Les clés AIK, comme vu précédemment, servent à chiffrer les registres PCR lors d'une attestation à distance. Elles sont générées par la puce TPM. Pour que le challenger soit assuré que la clé utilisée pour la signature des registres PCR appartienne bien à la puce TPM souhaitée, un certificat associé à cette clé est délivré par une autorité de certification.

Lorsque la puce TPM génère un couple AIK, elle demande à la CA de la certifier. Pour cela, le certificat généré par la CA est renvoyé à la puce TPM mais chiffré avec la clé publique EK de la puce. Ainsi, seule la puce TPM peut déchiffrer le certificat fourni par la CA. Le certificat contient une référence du fabricant et le modèle de la puce et de la plateforme. Il est signé par la CA.

Chapitre 3

Une plateforme "clé en puce"

Nous venons de voir que la plateforme a été pensée pour être "robuste". Elle a de plus la particularité d'être facile à installer et ce, avec une large variété de systèmes d'exploitation. Dans ce chapitre, nous suivrons la mise en place d'une plateforme TPM sous le système d'exploitation Linux.

Aujourd'hui, une grande majorité des ordinateurs portables sont vendus avec une puce TPM intégrée à la carte mère. Cependant, par défaut, la puce n'est pas activée. Pour pouvoir s'en servir, il faut suivre les étapes suivantes :

1. S'assurer d'avoir une puce TPM et déterminer son type
2. Activer la puce dans le BIOS
3. Installer un driver approprié pour qu'il puisse parler à la puce
4. Installer un TSS pour faire le lien entre la puce et les applications
5. Faire les configurations nécessaires

3.1 Type de puce et driver

3.1.1 Détermination du type de puce

Avant de vouloir utiliser la puce, il faut tout d'abord s'assurer de la présence d'une telle puce dans l'ordinateur et connaître son type. Le nom de la puce TPM est stocké dans les tables ACPI[1]. Ces tables regroupent les configurations des composants matériels de l'ordinateur. Pour obtenir un dump de ces tables, il faut réaliser, en root, les commandes suivantes :

`cp /proc/acpi/dsdt ./dsdt`
`iasl -d dsdt` (après avoir installé le paquet iasl).

[1] Advanced Configuration and Power Interface

Après édition du fichier dsdt.dsl, il ne reste plus qu'à chercher le device TPM.

```
[...]
Device (TPM)
{
Name (_HID, EisaId ("...")
```

Ce qui suit définit quel type de puce est présent dans l'ordinateur :
`BCM___` -> Broadcom
`ATM___` -> Atmel
`IFX___` -> Infineon
`STM___` -> STMicroelectronic

Par exemple, pour une puce Infineon de version 1.1 on peut lire :

```
Device (TPM)
{
Name (_HID, EisaID ("IFX0101"))
Name (_UID, 0x01)
[...]
```

3.1.2 Activation dans le BIOS

La puce TPM est par défaut inactive or il faut qu'elle soit active pour pouvoir fonctionner. L'activation se passe lors du démarrage de l'ordinateur. On rentre dans le BIOS lorsque l'écran POST apparaît,en général grâce à la touche F2. Il existe une grande variété de BIOS différents, l'activation de la puce se fait dans un des items du menu, généralement en rapport avec la "sécurité". Il faut changer l'état `[disable]` en `[enable]`.

3.1.3 Chargement du driver

Les noyaux Linux ont depuis la version 2.6.12 des drivers pour les puces TPM. Ce sont des modules qui ne sont pas chargés initialement. Dans un premier temps, il faut charger les modules génériques `tpm` et `tpm_bios`. Pour charger le driver correspondant au type de puce présente dans la machine, dans le répertoire /lib/modules/2.6.xx/kernel/drivers/char/tpm, exécuter, pour le module correspondant à la puce, les commandes :

```
modprobe tpm_bios
modprobe tpm
modprobe nom_module
```

Le module `tpm_tis` est un module générique correspondant aux versions 1.2 des puces TPM. Pour les versions antérieures des puces, on peut utiliser les modules correspondant au fabricant de la puce. Par exemple, pour une puce Infineon 1.1, on utilisera le module `tpm_infineon`.

3.2 Installation de la pile

3.2.1 Projets Open Source

Tout composant physique doit être accompagné d'un fournisseur qui fait le lien entre ce composant et l'utilisateur. Le groupe TCG est une organisation regroupant de nombreuses entreprises et la puce TPM est implantée sur de nombreux ordinateurs. Il ne peut donc pas y avoir un fournisseur unique pour Windows par exemple. Différents projets Open Source ont vu le jour. La Figure 3.1 illustre ces projets.[17, 18, 19, 20, 21, 22]

Nous avons choisi d'utiliser la pile proposée par le projet TrouSerS. C'est en effet un projet compatible avec Linux, accessible et très bien documenté. Nous travaillons sur la version `0.3.1-7` [23].

3.2.2 La pile TrouSerS

Le projet TrouSerS se devait de suivre les spécifications proposées par le groupe TCG en matière de Software Stack [3]. Les acteurs du projet ont donc mis en place une architecture compatible avec ces spécifications. La figure 3.2 illustre cette architecture.

TSS Service Provider La plateforme peut être utilisée à distance ou localement. Quelque soit son type, chaque application utilise son propre TSP, chargé comme une bibliothèque. Il fournit des services de la plateforme aux applications qui se divisent en deux catégories.

- Le premier type de services, *Context manager* fournit les outils de gestion de la plateforme (réinitialisation ...).
- Les *Fonctions cryptographiques* qui fournit les fonctions de cryptographie de la puce telles que la fonction de hachage SHA-1 ou des mécanismes de signatures.

TSS Core Services C'est le démon qui tourne dans l'espace utilisateur afin de centraliser toutes les commandes de l'espace utilisateur à destination de la puce.

TSS Device Driver Library Cette bibliothèque est située dans le répertoire `/src/tddl`. Elle fait le lien entre le pilote et le TCS et ce, quelle que soit l'implémentation du pilote. Nous rappelons en effet que le pilote

	TrouSerS [17]	**Open TC** [18]	**Trusted JAVA project** [19]	**EMSCB** [20]	**TPM4JAVA** [21]
Quoi	- Software Stack - TPM keyring	- Système d'exploitation sécurisér	- Fonct. pour JAVA utilisant les TPM - Software Stack	- Appli. sécurisées utilisant la puce TPM - Turaya [17] (VPN sec. par exemple)	- Librairie JAVA d'accès à la plateforme
Qui	Surtout IBM	- Partenariat - Sponsorisé par l'UE	- U. Graz - partie de l' Open TC	- Sponsorisé par le gvt allemand	- U. Darmstadt
Langage	C	C, JAVA	JAVA	C	JAVA
OS	i386 GNU Linux	Linux (SuSE)	Linux 2.6 Windows Vista		Linux Windows
Licence	GPL		GNU GPL	GPL	LGPL
+	- Spec disponible - TSS 1.1 complet - Aide en ligne	- Système d'exploitation complet et sécurisé disponible		- En partenariat avec le projet Open TC	- Compatible avec les plateformes 1.1 et 1.2

FIG. 3.1 – Les projets Open Source pour une plateforme TPM

n'est pas implémenté par le projet TrouSerS mais est déjà présent dans le système d'exploitation Linux.

TPM Device Driver Situé à la frontière entre l'espace utilisateur et l'espace noyau, il permet le passage d'un espace à l'autre. C'est lui qui fait le lien direct entre la puce et l'espace utilisateur.

3.2.3 Installation de la pile

L'installation se fait à partir des paquets synaptic. Les paquets `tpm-tools` et `trousers` doivent être installés :

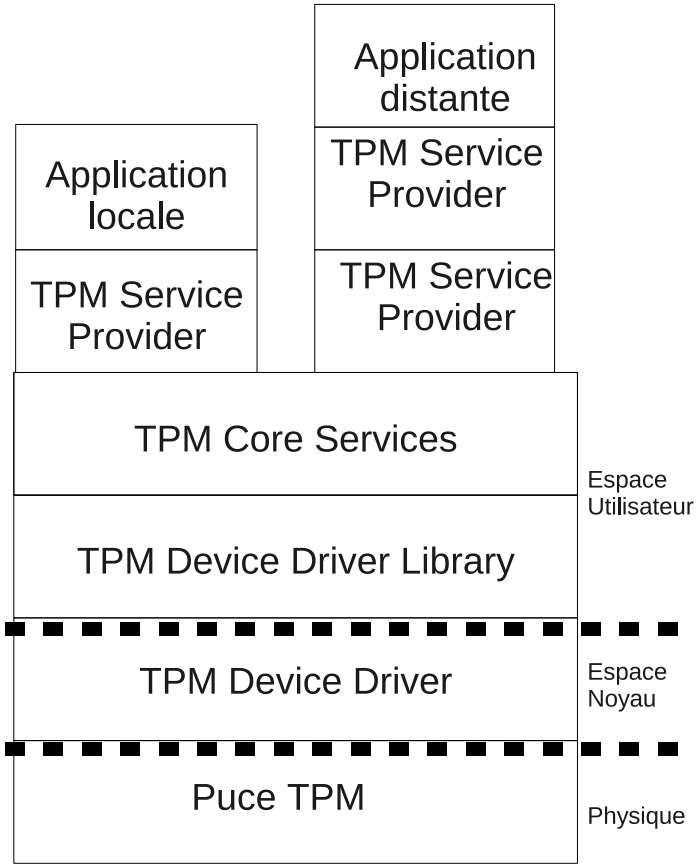

FIG. 3.2 – La pile logicielle du projet TrouSerS

```
apt-get install tpm-tools
apt-get install trousers
```

Il faut enfin lancer le démon TrouSerS (TCSD) :

```
/etc/init.d/trousers start
```

On peut à présent vérifier si la puce est accessible. Si la commande `tpm_version` renvoi :

```
TPM Version: xxxxxxxx
Manufacturer Info: xxxxxxxx
```

Alors tout va bien, la puce est accessible, tout est installé, il ne reste plus qu'à utiliser la plateforme maintenant !

3.3 Utilisations

Nous avons à présent à notre disposition une plateforme configurée et prête à l'utilisation. TrouSerS a pour objectif de fournir la pile logicielle permettant de faire le lien entre les applications et la puce, à travers notamment du TCS. Le projet TrouSerS a cependant développé quelques applications regroupéees dans le paquet *tpm-tools*. Dans cette section, nous présenterons quelques exemples d'utilisation de ces commandes.

3.3.1 Devenir propriétaire

Avant toute chose, il faut devenir propriétaire de la plateforme, afin de pouvoir utiliser les fonctions cryptographiques qu'elle permet. Pour cela, il suffit de lancer la commande `tpm_takeownership`

```
tpm_takeownership
Enter owner password:
Confirm password:
Enter SRK password:
Confirm password:
```

Ces mots de passe sont importants car il vous seront demandés pour certaines opérations sur la puce comme par exemple pour la commande `tpm_clear` qui replace la puce dans son état de manufacture Ces mots de passe peuvent être modifiés à tout instant grâce à la commande `tpm_changeownerauth` avec les options :

- `-o` pour changer le mot de passe du propriétaire
- `-s` pour changer le mot de passe SRK

Il ne peut y avoir qu'un seul propriétaire de la plateforme. Ainsi, la commande `tpm_takeownership` lancée une seconde fois provoque l'erreur suivante :
`Tspi_TPM_TakeOwnership failed: 0x00000014 - layer=tpm, code=0014 (20), Owner already set.`

Si l'on souhaite malgré tout prendre la place du propriétaire actuel, il faut effacer la puce

3.3.2 Effacer la puce

Pour remettre la puce dans son état de manufacture, on peut le faire de deux façons :

- physiquement à partir du BIOS

– à partir de la commande `tpm_clear`. Par défaut, cette commande demande le mot de passe du propriétaire mais, avec l'option `tpm_clear -f`, on peut le faire en étant physiquement connecté à la plateforme sans connaître le mot de passe du propriétaire.

```
tpm_clear
Enter owner password:
TPM Successfuly Cleared.
You need to reboot to complete this operation.
After reboot the TPM will be in the default state:
unowned, disabled and inactive.
```

La coomande tpm_clear désalloue le propriétaire de la plateforme, efface toutes les clés stockées dans la puce, sauf la clé EK. La clé SRK faisant partie des clés supprimées, toutes les clés stockées en dehors de la puce et chiffrées à partir de la clé SRK deviennent inutiles.

La puce retourne dans son état d'origine, à savoir : *disable*, *deactivated* et *unowned*. Il faut donc la réactiver dans le BIOS. Il peut arriver que le BIOS ne se rende pas compte de la désactivation de la puce par la commande et garde l'attribut `[enable]` pour la puce. Il faut alors placer manuellement la puce en `[disable]` dans le BIOS, relancer le système puis remettre la puce dans l'état `[enable]`. Il faut de plus réallouer la puce à un propriétaire car celle-ci est `[unowned]`.

3.3.3 Obtention des registres PCR

Les registres PCR sont des zones de stockages non-volatiles dans lesquelles sont stockées les valeurs hachées des mesures de la plateforme. Il peut y avoir entre 15 et 23 registres selon les constructeurs. Chaque numéro de registre correspond à une mesure :

Numéro PCR	**Description**
de 0 à 7	**Mesures des étapes de lancement**
0	CRTM et BIOS
1	Carte mère
2	Code ROM
3	Configuration ROM
4	Master Boot Record (MBR) Code
5	MBR table de partitions
6	Changements d'états
7	Fabricant
de 8 à 14	**Mesures du système d'exploitation**
8 et 9	NTFS boot
10 et 11	Boot manager
12	Arguments du menu.lst
13	Fichiers "checked"
14	Fichiers chargés (ex : Linux Kernel, modules...)
15 à 23	**Ne sont pas utilisés**

Les valeurs de ces registres sont disponibles dans le fichier : `/sys/class/misc/tpm0/device/pcrs`

Le resultat de la commande : `cat /sys/class/misc/tpm0/device/pcrs` permet de voir les registres PCR.

```
PCR-00: F6 CB F6 A8 2A 01 8B 1A AC 68 B2 83 17 6C D1 05 2A 4B F6 C5
PCR-01: 67 39 EC 4D 62 21 DE C9 58 6B 6F C1 F3 43 4D 92 48 8D 87 C4
PCR-02: 25 B0 A3 6A 75 1A 05 94 0E 4E 79 68 CF 49 75 2F 3D 1B EF 78
PCR-03: 04 FD EC DD 50 1D AF 0F 62 4C 1F 99 60 12 CF 30 44 FF 46 10
PCR-04: 0F 76 F4 EC 13 43 B1 BB 8E 40 7D 5C 42 20 7C 41 5A F7 C4 16
PCR-05: 27 66 4B 64 05 D3 70 CE FB BC 39 C0 78 BB 6A 79 85 CD 51 41
PCR-06: 04 FD EC DD 50 1D AF 0F 62 4C 1F 99 60 12 CF 30 44 FF 46 10
PCR-07: 04 FD EC DD 50 1D AF 0F 62 4C 1F 99 60 12 CF 30 44 FF 46 10
PCR-08: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-09: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-11: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-12: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-13: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-14: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
PCR-15: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Les registres numérotés à partir de 8 sont nuls. Ils correspondent aux mesures sur le système d'exploitation ce qui signifie que celui-ci n'est pas

mesuré. Pour obtenir ces mesures, il faut installer un Trusted GRUB[2]

3.3.4 Sceller un message

Sceller un message consiste à chiffrer le message avec une clé symétrique, ajouter à ce message les valeurs des registres PCR souhaités et chiffrer le tout avec la partie publique d'une clé asymétrique. La commande qui permet de créer ce type de messages est la commande `tpm_sealdata` avec les options :

- `-i` suivie de l'empacement du message à chiffrer.
- `-o` suivie du fichier de destination.
- `-p` suivie du registre PCR à ajouter au message, on peut mettre autant de `-p` que de registres PCR que l'on souhaite ajouter.

La commande `tpm_sealdata -i a_coder -o destination -p 2 -p 3` scelle le message contenu dans le fichier `a_coder`. Elle nécessite d'être propriétaire sur la plateforme. Le message est chiffré avec les registres PCR numéros 2 et 3 et le résultat est envoyé dans le fichier de destination `destination`. Ce message scellé est à destination de la plateforme qui a servi à sceller le message. Seule cette plateforme, et uniquement lorsqu'elle se trouve dans la même configuration que lors du sceau, peut desceller le message.

Le fichier de destination comporte :

```
-----BEGIN TSS-----
-----TSS KEY-----
AQEBBwARAAAABAEAAAABAAMAAQAAAAwAAAgAAAAAgAAAAAAAAAAAAABAIxDAfbE
ST5pcv6kNODfKscMAAWHG/NENjESRuiDpcgAwvRmJj2URAt7zWSQwGE0xzwGi3Gz
nlqKcj0iPky+kG3mGXNVphzp9eKNvV5Pmv4diYAtvIFpM4yL+WNaxcNcjuMrxK+h
DQ9yikdXV9jVsHvQ8NcEo3aJ9IlXfo5U3/n1KnriLM/K4jdRuL/fJNNlVs7tZXkr
sKSUcpp0JtgZVhRrbpsaQWbtbGzl2esx8+1XDgnyPnEgaOeWlAvuP8pGmzVYSX3r
BxyzThBu3Cyjg7nkfya+wkPWVCxKdx0jw9PHk55lr1qCi30Ectmiqjlp2aXoYuzO
yDJryVeIrTMeRYEAAAEALkDsr1E7U0uUtSOfW7Jvynxh9t0vjCB16ynvwY1aP4Vy
mgzVzAlA+jKmmdqCZ0++DyHbtvOUyTwMi1B1fMxIea5ZhsCpGh+SpUmhyY4hnC/M
jGgLxuP0aRPAYs2gf0+55Aj/uwJLU6++g7H4gQgavCgd2/3hG2JYjHQhutsBGO5x
hRWyge9GLblE2bokzsNBczswIQXFYc+e0yMLUrctC8rjZXMjwMllzu3L1GiBO8wb
YewCIqwIMMdT6+kOMTGexVXag6ZOpEInUDdRKt/fXiDiIGk+MP9f/3FHDD1E2Xxm
LNagAyb/AiVPCcLP8y+58w0+8nHEH0Dd/7YecQvknA==
-----ENC KEY-----
Symmetric Key: AES-256-CBC
AQEBBwAAACwAAgwAcd6q9jm77329JOC40rr3rQFaMgBx3qr2Obvvfb0k4LjSuvet
AVoyAAAAAQAmkTRDDQE2DJvm78sSUoi8yiWWYnib/tETVsLIdVerGrdlwxU8oRJ7
wbWz03rbfSKAzAl/0XB6S7jHLozQzuNSZMfzYaaJFtocGxDVsQNaudILxbL6LEYm
```

[2]http ://sourceforge.net/project/showfiles.php ?group_id=165379

```
2jyECYEMf7UcZiY1/OxMGP9BBGE3vPQ7vcATtSwnSm8V8dg8DcB+N32dFBkxQ2Yg
krJG/e42t2Ahy5licATtE3rYc1MdaXRVCgN1LUjxkhmxn85g2oDD813JXBLFjjkk
4P70C7dlG8jb2v0sQ7ZmB3y5Wi2qOUJXvmJRBr5Ux3nJyo8M6w/MIAGJPn4Shqdg
whEVMaihkgeKivw3IWlEk1GvCtkbNLZ3
-----ENC DAT-----
BBd9tgaEsbV743j3BWPYoyU2Cg13nWT4HaTJHs9U32k=
-----END TSS-----
```

- TSS Key est la partie publique d'une clé asymétrique de la plateforme.
- Le message chiffré avec une clé symétrique
- La clé symétrique et les registres PCR choisis, le tout chiffrè avec la clé TSS Key

3.3.5 Autres commandes

Le tableau suivant illustre les autres commandes disponibles dans le paquet *tpm_tools*.

Nom de la commande	**Description**
`tpm_createek`	Crée un couple de clé EK
`tpm_getpubek`	Renvoi la clé PUBEK (partie publique de la clé EK)
`tpm_restrictpubek`	Restreint la capacité d'accès à la partie publique de la clé EK au propriétaire de la plateforme
`tpm_restrictsrk`	Restreint la capacité d'accès à la partie publique de la clé SRK au propriétaire de la plateforme
`tpm_revokeek`	Révoque la paire de clé EK
`tpm_selftest`	Demande à la puce TPM de s'auto-tester
`tpm_setactive`	Permet le passage à l'état `active` (option -a) ou `inactive` (option -i)
`tpm_setclearable`	Détermine si la puce peut être effacée à partir des droits du propriétaire (option -o) ou d'une personne présente physiquement (option -f)
`tpm_setenable`	Permet le passage à l'état `enable` (option -e) ou `desable` (option -d)
`tpm_setownable`	Détermine si la puce est dans l'état `owned` ou `unowned`
`tpm_setpresence`	Change l'état de la présence physique sur la plateforme
`tpmtoken_import`	Importe un certificat X509 ou une clé publique RSA
`tpmtoken_init`	Initialise la zone de stockage TPM PKCS#11 d'un utilisateur
`tpmtoken_objects`	Affiche les objets dans la zone de stockage TPM PKCS#11 d'un utilisateur
`tpmtoken_protect`	Chiffre ou déchiffre des données à partir d'une clé de la zone TPM PKCS#11 d'un utilisateur
`tpmtoken_setpasswd`	Change le mot de passe associé à la zone de stockage TPM PKCS#11 d'un utilisateur

Chapitre 4

Un système infaillible ?

L'avenir de la plateforme de confiance est fortement lié à la sécurité qu'elle propose. Si la plateforme de confiance n'est pas aussi sûre qu'elle le prétend, son avenir risque d'être fortement compromis. Dans cette section nous déterminerons quelles sont les menaces physiques et logicielles sur la plateforme et s'il existe des contremesures pour y remédier.

4.1 Attaque par Reset

Cette attaque physique a été élaborée en 2004 par Bernhard Kauer [24]. Elle a été reprise pour une thèse par l'Université de Darthmouth [25] et une vidéo de cette attaque est disponible sur You Tube[1].

La faille

L'attaque par Reset repose sur la manière dont sont stockées les mesures de la plateforne. Nous avons vu dans la section 2.2.2 que le stockage des mesures dans les registres PCR suit le schéma suivant :
$PCR_i(n+1) = SHA1(PCR_i(n) + nouvelles mesure)$ où la loi + correspond à la concaténation.

L'attaque

L'attaque consiste à remettre la puce dans son état d'origine. Les registres PCR deviennent alors nuls mais le reste de la plateforme n'est pas réinitialisée. Puis, à partir d'un driver TPM malicieux, et plus particulièrement de la commande tpm_startup, on peut réinitialiser les registres PCR.

[1] http ://www.youtube.com/watch ?v=f_Zx8sKCiTo

Parce que la puce TPM est positionnée sur le bus LPC, pour remettre les registres à zéro, on place un fil électrique entre les lignes RESET et GROUND. Tous les composants sur le bus sont alors réinitialisés, la puce TPM y compris. On peut aussi uniquement ne réinitialiser que la puce TPM.

L'attestation à distance et le sceau des messages sont alors faussés. Par exemple, l'attaque réalisée par Evan Sparks dans son rapport de thèse consiste à mémoriser les valeurs PCR du système, puis changer le disque dur de la machine par un autre disque comportant un système d'exploitation différent. Ensuite, un signal de Reset est envoyé à la puce qui se réinitialise. Enfin, les mesures PCR préalablement sauvées sont réinjectées dans la puce. La plateforme tourne ainsi dans une configuration non sécurisée mais continue de penser qu'elle l'est.

Contre-mesures

Cette attaque a été réalisée à partir d'une puce TPM sur la daughterboard ce qui facilite l'accès aux lignes de Reset. Une première contre-mesure consiste à intégrer la puce à la carte mère pour compliquer l'accès à cette ligne de Reset.

La faisabilité de cette attaque repose grandement sur le fait que la puce s'initialise à partir de la commande tpm_startup. Si seul le BIOS, puisque c'est lui qui démarre la chaîne de confiance, a la capacité de dire à la puce de se réinitialiser après un signal de reset, alors l'attaque devient obslolète.

4.2 Attaque Cold Boot

Les attaques de la classe Cold Boot reposent sur le fait que, contrairement aux idées reçues, la mémoire DRAM persiste quelques temps après avoir éteint l'ordinateur. Plus la température environnant la DRAM est basse et plus le temps de persistance est long d'où le nom "cold boot". On peut ensuite récupérer des informations sur les données stockées dans la DRAM.

Dans un papier[26] sur les attaques Cold Boot, des chercheurs, principalement de l'Université de Princeton, ont assuré que l'attaque Cold Boot est efficace sur les disques chiffrés par BitLocker et ce, malgré l'utilisation de la puce TPM comme zone de stockage de la clé de chiffrement.

Bien que cette attaque ne soit pas une attaque directe sur la puce TPM, elle a fait réagir le groupe TCG. À travers une spécification[27] récemment publiée (mai 2008), le groupe de travail PC Client empêche les attaques Cold Boot dans le cas où la mémoire DRAM n'est pas ôtée de l'ordinateur. Elle

décrit comment détecter si un système a mal été éteint et dans ce cas rendre les clés stockées invalides en nettoyant la mémoire.

4.3 Attaque Toctou

Cette attaque logicielle a été élaborée en 2007 par une équipe de l'université de Darthmouth[25, 28]. Ils ont par ailleurs proposé la contre-mesure associée.

La faille

Toute la confiance en la configuration de la plateforme repose sur la mesure qui en est faite. Cette mesure n'est réalisée qu'avant que le code soit chargé en mémoire dans la RAM et n'est plus effectuée ensuite. Le temps qui sépare le moment où le programme a été chargé et celui où il est utilisé peut varier. Plus le laps de temps est important et plus il y a de chance que le programme soit compromis au moment de son utilisation. Le nom de cette attaque, TOCTOU, provient de l'accronyme de Time Of Check Time Of Use.

L'attaque

Comme la mesure du code d'un programme n'est effectuée qu'une fois et avant son utilisation, si l'on parvient à changer le segment de code d'un programme dans la RAM, alors on peut modifier le comportement de ce programme sans que la puce ne s'en rende compte. Le but de l'attaque est donc de parvenir à écrire dans le segment de code d'un processus cible à partir de son adresse virtuelle.

Pour accéder à la RAM à partir de l'adresse virtuelle d'un processus, on passe par les tables :
- Page Global Directory (PGD)
- Page Middle Directory (PMD)
- Page Table Directory (PTD)
- RAM

Le moyen retenu pour réaliser l'attaque est un module noyau. Dans un premier temps, la page PTD associée à l'adresse virtuelle est copiée afin de modifier les droits d'accès en ajoutant le droit *write*. On peut ainsi écrire les données modifiant le programme dans la RAM.

Contre-mesure

Une première contre-mesure consisterait à empêcher l'écriture dans la RAM mais c'est plutôt contraignant. La contre-mesure retenue par l'équipe de chercheurs de l'Université de Darthmouth consiste à avertir d'une modification du segment de code d'un programme préalablement mesuré.

Dans leur contremesure proposée, la puce TPM est connectée au MMU[2]. Cette connection physique permet au MMU de détecter les opérations mémoire qui peuvent affecter les mesures effectuées par la puce. Dans le cas d'une modification quelconque du segment de code, la puce est informée par le MMU.

4.4 Attaque par rejeu

Cette attaque a été découverte et contrecarrée en 2005 par des chercheurs de l'Université degli Studi à Milan[29]. Elle révèle la possibilité, malgré les précautions prises par le groupe TCG de faire une attaque par rejeu dans un des protocoles d'authentification du propriétaire : le protocole OIAP. Après une analyse par des méthodes de *Model Checking*, ils ont découvert et réparé l'erreur commise par le groupe TCG.

Le protocole OIAP

Il existe deux types de session pour le propriétaire :
- *Object Specific Authorisation Protocol*(OSAP) : une session pendant laquelle plusieurs commandes agissant sur une même ressource de la puce peuvent être utilisées.
- *Object Independant Authorisation Protocol* : une même commande agissant sur des ressources différentes de la puce peut être utilisée plusieurs fois pendant la session.

L'attaque concerne le protocole OIAP. La session s'établie ainsi :

1. Le propriétaire signale qu'il souhaite initialiser une session OIAP. Il possède un secret partagé avec la puce à savoir le mot de passe du propriétaire noté K.
2. La puce lui fournit les informations concernant la future session (`S1`) ainsi qu'un nonce (`N1`).
3. Le propriétaire crée un message `m` comme la concaténation des éléments :
   - La commande qu'il souhaite exécuter avec ses arguments : `CMD(arg)`
   - Le `S1` et le nonce `N1`

[2]Memory Management Unit

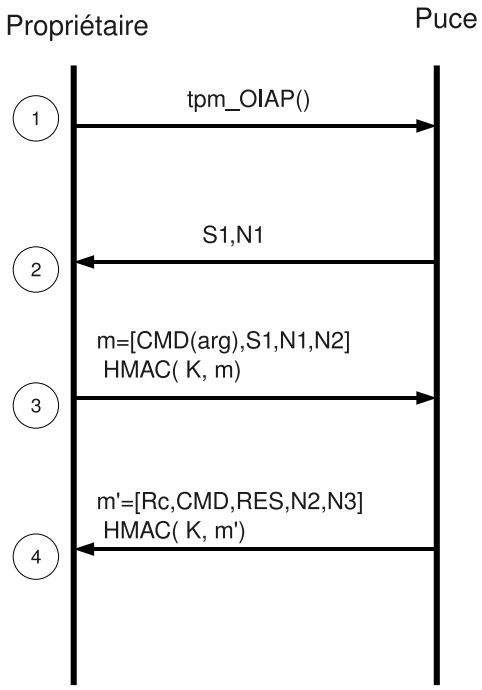

FIG. 4.1 – Le protocole OIAP

– Un nouveau nonce `N2`

Le propriétaire envoie (m,HMAC(m)) où $HMAC(M) = Hash(K.m)$ où . désigne la concaténation.

4. La puce vérifie le message en calculant de son côté HMAC(K.m) et, si la comparaison avec la valeur envoyée par le propriétaire est correcte, elle exécute la commande désirée . Elle envoie alors le résultat de cette exécution au propriétaire. Pour cela, elle crée à son tour un message `m'` comportant la concaténation des éléments suivants :
   – Le code de retour `Rc`
   – La commande `CMD`
   – Le résultat `RES`
   – Un nouveau nonce `N3` ainsi que `N2`

   La puce envoie (m',HMAC(m')).

Le groupe TCG a spécifié que la session reste ouverte jusqu'à ce que le propriétaire décide de la fermer ou qu'il envoie une commande qui provoque une erreur.

L'attaque

L'attaquant se place entre la puce et l'utilisateur. L'attaque comporte plusieurs phases.

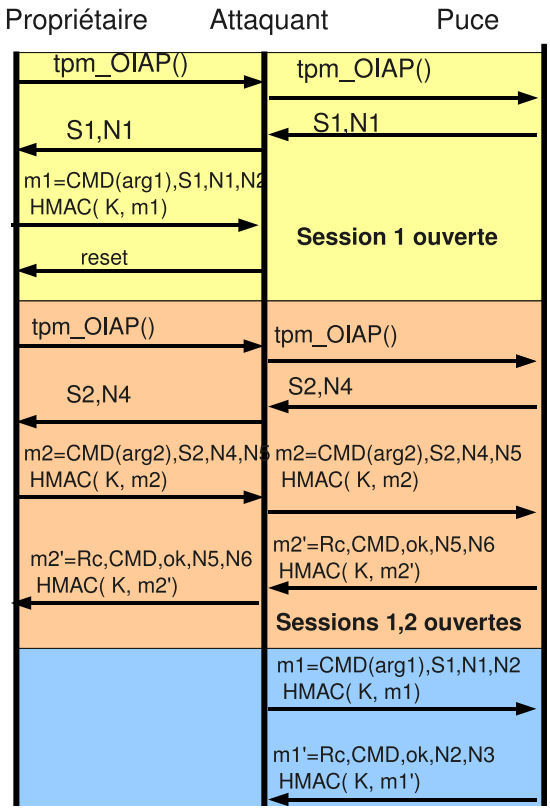

FIG. 4.2 – L'attaque par rejeu

phase 1 Le propriétaire de la puce lance une requête d'ouverture de session à la puce. L'attaquant se place au milieu et intercepte les messages 1 et 2 du protocole OIAP qu'il rejoue ensuite au destinataire légitime. Il intercepte ensuite le message 3, c'est à dire la commande à exécuter, mais ne la transmet pas à la puce. Pour ce qui est du message 4, à savoir la réponse de la puce après avoir exécuté la commande, l'attaquant intercepte la réponse. Il envoie au propriétaire un message d'erreur ce qui fait croire à ce dernier que la session OIAP s'est terminée alors qu'en réalité celle-ci reste ouverte.

phase 2 Le propriétaire pensant que la session précédente a échoué, il tente d'en ouvrir une nouvelle pour la même commande mais avec des données différentes. L'attaquant, toujours au milieu, intercepte et rejoue tous les mes-

sages qui circulent. Il laisse la session 2 s'ouvrir correctement et la commande du propriétaire est correctement effectuée par la puce.

phase 3 On a donc deux sessions ouvertes. Il s'agit pour l'attaquant de rejouer la commande lancée par le propriétaire lors de la phase 1. La commande correspond à la première session et vient remplacer l'action de la commande du propriétaire lors de la phase 2.

La contre-mesure

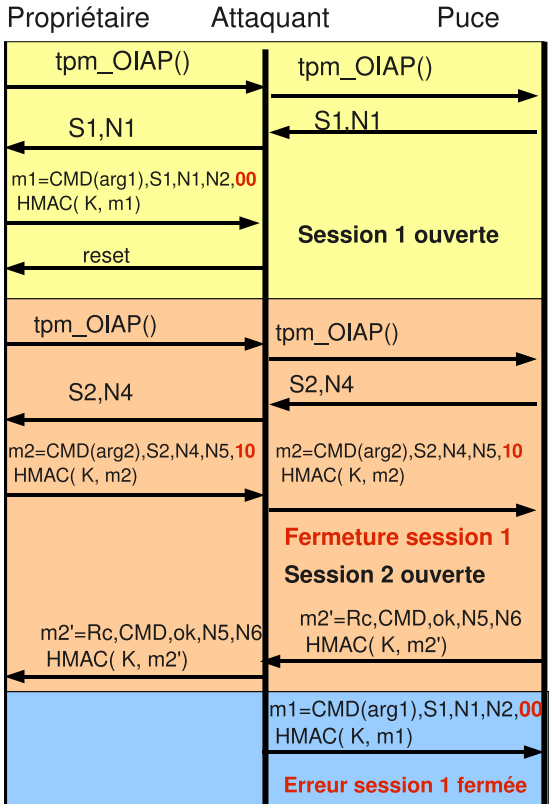

FIG. 4.3 – La contre-mesure

L'erreur du groupe TCG dans ce protocole a été trouvée grâce au logiciel de Model Checking SPIN[3]. Elle provient du fait que lors de plusieurs session, le propriétaire ne connait pas nécessairement l'état de chaque session ouverte. Les chercheurs de l'université de Milan ont alors rajouté un champ dans le message, n bits appelés *bitmask* qui représente la connaissance de

[3]spinroot.com

l'état des sessions par le propriétaire. Le n-ième bit du bimask correspond à la session numéro n. Si le bit est à 0, la session est considérée comme ouverte par le propriétaire sinon c'est qu'elle est fermée. Ainsi, si la puce remarque une différence entre le bitmask du propriétaire et l'état de la session alors elle peut fermer la session dont le bit diffère dans le bitmask.

Plusieurs sessions peuvent ainsi être ouvertes simultanément mais à condition que le propriétaire soit au courant. Le nombre de sessions pouvant être ouvertes en même temps est égal au nombre de bits du bitmask.

Conclusion

L'informatique de confiance est un espoir qu'ont tenté de matérialiser, à travers la plateforme TPM, de nombreuses entreprises regroupées sous le nom de groupe TCG. Cette plateforme a été conçue pour répondre à deux exigences en matière de sécurité : attester la configuration de l'ordinateur et proposer des fonctionnalités cryptographiques.

La renommée internationale de ses concepteurs en fait un produit crédible en matière de sécurité informatique. Le composant physique sur lequel repose cette sécurité, la puce TPM, est une zone de stockage protégée qui suscite un interêt grandissant pour la communauté de chercheurs. Anti-rootkits ou solution aux problèmes de téléchargement de musique, cette technologie promet de multiples utilisations.

A présent normalisée, la puce TPM reçoit l'appui du gouvernement américain qui la recommande. Elle est largement diffusée dans les ordinateurs portables et son utilisation est facile et compatible avec la majorité des systèmes d'exploitation. Malgré tous ces avantages, la technologie TPM est très peu utilisée.

Victime d'un passé tâché par la polémique Palladium, l'ordinateur de confiance souffre encore de quelques critiques à son égard. Cependant, concernant la sécurité assurée par la plateforme, les attaques qui ont été menées, en tant que preuves de concept essentiellement, ont toutes été contrecarrées par des contre-mesures simple.

En conclusion, l'ordinateur de confiance proposé par le groupe TCG n'est pas le "remède miracle" à tous les problèmes de sécurité informatique mais il semblerait qu'un ordinateur protégé par la technologie TPM soit plus sûr qu'un qui ne l'est pas. L'ordinateur de confiance a donc un avenir prometteur devant lui...

Bibliographie

[1] groupe TCG. TCG specification architecture overview, août 2007.

[2] groupe TCG. TCG specification design priciples, juillet 2007.

[3] groupe TCG. TCG software stack specifications, mars 2007.

[4] groupe TCG. TCG physical presence interface specification, avril 2007.

[5] Aberdeen Group. Trusted computing, turne in, turne on, 2008.

[6] J.P Lacombe and J.P Gras. TCPA/palladium : Big brother is watching you. Technical report, 2003.

[7] groupe TCG. TPM v1.2 specifications changes, 2003.

[8] groupe TCG. Stopping rootkits at the network edge, 2007.

[9] D. Winder. How to detect and remove rootkits with windows encryption, 2008.

[10] Intel. Low pin count specification. http ://www.intel.com/design/chipsets/industry/lpc.htm.

[11] Norme PKCS # 1. http ://www.rsa.com/rsalabs/node.asp ?id=2125.

[12] IEE 1363 home page. http ://grouper.ieee.org/groups/1363/.

[13] Description du standard SHA-1. http ://www.itl.nist.gov/fipspubs/fip180-1.htm.

[14] Atmel. La puce AT97SC3202 d'atmel, 2004. https ://www.trustedcomputinggroup.org/press/Atmel_TPM1.2_PR-Final2.pdf.

[15] Site officiel des Critères Communs. http ://www.commoncriteriaportal.org/.

[16] FIPS home page. http ://www.itl.nist.gov/fipspubs/index.htm.

[17] Trousers home page. http ://trousers.sourceforge.net/.

[18] Open TC home page. http ://www.opentc.net/.

[19] Trusted java project. http ://trustedjava.sourceforge.net/.

[20] EMSCB home page. http ://www.emscb.com/.

[21] EMSCB projet Turaya.
http ://www.emscb.com/content/pages/turaya.htm.

[22] TPM4JAVA. http ://tpm4java.datenzone.de/trac.

[23] Dernière version de TrouSerS téléchargeable.
http ://sourceforge.net/project/showfiles.php ?group_id=126012.

[24] B. Kauer. OSLO : Improving the security of trusted computing, 2007.

[25] E. R. Sparks. A security assessment of Trusted Platform Modules, 2007.

[26] J. Halderman and S. Schoen. Lest we remember : Cold Boot Attacks on Encryption Keys, 2008.

[27] Groupe TCG. TCG platform reset attack mitigation specification, 2008.

[28] S. Bratus, N. D Cunha, and S. W. Smith. TOCTOU, traps, and trusted computing, 2008.

[29] D. Bruschi, L. Cavallaro, and A. Lanzi. Replay attack in TCG specification and solution, 2008.