Les perspectives de la maintenance de l'assurance sécurité

Camille de Sagazan

Silicomp-AQL
Rue de la Châtaigneraie
CS 51766
35517 Cesson-Sévigné Cédex

Résumé La maintenance de l'assurance sécurité couvre les problématiques liées à la prorogation des résultats d'une expertise sécurité menée sur un Système d'Information – par ex : audit technique de sécurité conduit de manière objective et répétable, évaluation de la sécurité des TI selon les Critères Communs – sur une période de temps donnée, en prenant en compte les *changements* susceptibles d'affecter la valeur du résultat établi par l'expertise sécurité du SI. Les changements peuvent être de nature *interne* : changement de configuration (paramètres ou version logicielle) de composants du système, ou *externe* : apparition de vulnérabilités, nouvelles techniques d'attaque, occurrence d'intrusions (qu'elles aient été contrées ou non par les mesures de sécurité ayant fait l'objet de l'expertise).

Cet article présente un certain nombre de notions issues de notre compréhension de ce sujet, dans le but de permettre aux RSSI de mieux appréhender et planifier les activités et les mesures de maintenance de l'assurance sécurité. Les démarches de maintenance de l'assurance sécurité spécifiques au Schéma Français d'Évaluation et de Certification sont présentées à titre d'exemple.

1 Introduction

1.1 Lutte informatique et maintenance de l'assurance sécurité

La lutte informatique couvre toutes les mesures et les techniques relatives à la sécurité informatique qui sont spécifiquement dédiées à la prise en compte de l'occurrence effective d'une attaque, c'est-à-dire de tout événement susceptible de causer un préjudice ou de conduire à la réalisation d'une menace sur les biens gérés par un Système d'Information.

Quand une attaque dirigée contre un Système d'Information (SI) survient, tout un spectre de situations peuvent en découler, suivant l'existence et l'efficacité des mesures de sécurité mises en place :

1. Les mesures de sécurité sont complètement efficaces, l'attaque est contrée ;
2. Les mesures de sécurité sont partiellement efficaces, une récupération ou une compensation sont possibles (incluant la restauration des données et des

services attaqués, ou l'obtention d'une compensation financière suite à la collecte d'éléments de preuve recevables (assurance, indemnités, etc.)) ;

3. Les mesures de sécurité sont pratiquement inefficaces, la récupération ou la compensation est impossible mais l'identification du chemin d'attaque est possible, ainsi que l'identification précise de la portée du préjudice. Des mesures correctives ou palliatives peuvent être prises ;
4. Les mesures de sécurité sont complètement inefficaces ou inexistantes, l'identification du chemin d'attaque est impossible, l'identification précise de la portée du préjudice est impossible.

Nous nous plaçons, dans le cadre de cet article, dans les cas « favorables » : on a un SI dont les mesures de sécurité (techniques ou non techniques) ont été *dûment expertisées* et sont *au moins partiellement efficaces*.

- « Dûment expertisées » signifie qu'une prestation d'expertise *indépendante* a été conduite sur les mesures de sécurité du SI, à l'aide d'un *référentiel* (ad hoc ou standardisé) permettant de garantir une certaine répétabilité et une certaine objectivité dans la manière dont l'expertise a été menée ;
- « Au moins partiellement efficaces » signifie que, sur une période de temps donnée, les mesures de sécurité ont une contribution positive et mesurable à la réduction des préjudices liée à la *réalité* des attaques.

On comprend qu'une manière raisonnable d'assurer que les mesures de sécurité sont efficaces consiste à tenter de faire valider *a priori* leur pertinence par rapport aux menaces supposées ou plus généralement à une problématique de sécurité bien définie. Cependant, dans la pratique, l'expertise sécurité n'est pas toujours abordée sous cet angle, et consiste plutôt à contrôler la conformité des mesures de sécurité par rapport à des bonnes pratiques. Cet état de fait est lié à un certain nombre de raisons dont nous ne discuterons pas ici.

Nous supposerons donc également dans la suite que l'« efficacité au moins partielle » des mesures de sécurité est basée sur leur validation *a priori* vis-à-vis d'une problématique de sécurité et/ou sur le contrôle de leur efficacité, au fil du cycle de vie du SI, entraînant des actions correctrices si nécessaire.

1.2 L'érosion de l'assurance sécurité

Le problème principal est que ce genre d'expertise, tout en coûtant cher, correspond à une *photographie* de la sécurité du SI. Or, au fil du temps, au moins trois types d'événements sont susceptibles d'entraîner l'obsolescence des résultats de l'expertise sécurité :

1. L'*état de l'art* associé aux d'attaques évolue : de nouvelles vulnérabilités apparaissent ; de nouvelles techniques d'attaques sont inventées ; celles existantes au moment de l'expertise mais jugées peu vraisemblables à ce moment-là peuvent être divulguées à un plus large public ; ou bien des outils peuvent être développés afin de mettre en œuvre ces attaques en ayant une compréhension minimale des détails techniques qu'elles impliquent ;

2. L'*état du SI* évolue. Par exemple, des effets liés à l'évolution du volume des opérations traitées par le système informatique, de la nature de ces opérations, des procédures organisationnelles ou de la composition des équipes peuvent remettre en question des faits sur lesquels l'expertise initiale était basée ;
3. Enfin, la *politique de sécurité* à laquelle le système doit se conformer ou qu'il doit mettre en œuvre peut changer, en liaison avec une évolution de la réglementation applicable ou des missions du système.

En conséquence, le *niveau de confiance* construit à un moment donné, à travers une expertise sécurité, dans l'efficacité des mesures de sécurité d'un SI pour répondre à la problématique de sécurité qui lui est assignée, peut être considéré comme subissant une forme d'*érosion* au fil du temps. D'où la question : comment faire pour maintenir cette confiance à un niveau satisfaisant de la manière la moins coûteuse possible ?

1.3 La mesure de l'impact des changements

Il paraît clair que procéder à une nouvelle expertise systématique est exclu dans la majorité des cas. En fait, à part dans le cas de changements tellement significatifs qu'ils reviennent à remplacer le SI par un autre complètement nouveau, on peut avancer qu'il devrait être inutile de ré-expertiser des éléments du SI qui n'ont pas changé depuis l'expertise initiale. La solution pourrait donc être de tracer les changements sur les composants de l'architecture de sécurité du SI et de ne procéder, lorsque la *ré-expertise* est décidée, qu'au ré-examen des composants ayant évolué. En pratique, c'est un peu plus compliqué que cela :

1. De nouvelles vulnérabilités peuvent être apparues sur un élément par ailleurs inchangé ;
2. Un élément inchangé peut dépendre, pour remplir son rôle dans l'architecture de sécurité, d'éléments ayant changé. Il convient de contrôler que ce rôle dans l'architecture de sécurité est toujours correctement rempli, malgré le changement des éléments dont il dépend.

A contrario, on peut tolérer des changements relativement significatifs sur des composants ne jouant pas de rôle dans la sécurité, dans la mesure où on est sûr que ces changements ne vont pas invalider de manière directe ou indirecte des hypothèses garantissant le fonctionnement nominal des composants dédiés à la sécurité.

Une approche pertinente de la maintenance de l'assurance sécurité devrait donc, selon nous :

1. Définir les *éléments* de la configuration du SI au niveau desquels les changements seront suivis, ce qui implique de faire un choix dimensionnant[1] concernant le niveau de granularité de ces composants ;

[1] Dimensionnant ... pour le volume des activités de suivi de l'impact des changements.

2. *Typer* ces éléments en fonction de leur criticité vis-à-vis de la sécurité (ex : critique, dédié, touchant, non touchant) ;
3. Identifier les *dépendances* entre ces composants élémentaires ;
4. Proposer des règles précises pour *décider de la ré-expertise* et *qualifier sa portée* en fonction de la nature des composants et des dépendances impactés par les changements, ainsi que du volume des changements.

1.4 Une boîte à outils conceptuelle

L'objet de cet article est de proposer des notions permettant aux exploitants de Systèmes d'Information, en charge de la sécurité, d'analyser, de planifier et de maîtriser de manière optimale la maintenance de l'assurance sécurité construite par des prestations d'expertise de type « audit de sécurité ». Ce jeu de notions doit donc être vu comme une « boîte à outils » conceptuelle permettant d'appréhender des situations concrètes plutôt que comme une proposition de norme. Nous examinerons les différentes approches de la maintenance de l'assurance sécurité qui ont été historiquement proposées dans le cadre du Schéma Français d'Évaluation et de Certification mis en œuvre par la DCSSI, nous en déduirons un jeu de notions pertinentes et nous les utiliserons comme une grille de lecture pour comparer les différences approches et proposer des applications à la maintenance de l'assurance sécurité des Systèmes d'Information.

2 La maintenance de l'assurance sécurité dans le cadre du Schéma Français d'Évaluation et de Certification

2.1 Généralités

Avant d'aborder les différentes approches de la maintenance de l'assurance sécurité, nous fournissons quelques rappels sur ce qu'est une évaluation de la sécurité des Technologies de l'Information (TI) dans le cadre du Schéma Français d'Évaluation et de Certification.

Une *évaluation de la sécurité des TI* est une expertise indépendante visant à confirmer qu'un ensemble de mesures de sécurité satisfont une problématique de sécurité spécifiée sous la forme d'un ensemble d'exigences.

Ces exigences portent sur un système informatique donné, ou bien sur un produit générique donné destiné à remplir un rôle défini dans l'architecture de sécurité des systèmes informatiques dans lesquels il sera intégré. Elles s'expriment typiquement comme le fait de contrer des menaces ou d'appliquer des règles sur les biens sensibles du système informatique.

Le système informatique ou le produit générique dont les mesures de sécurité font l'objet de l'évaluation est nommé « Cible d'Évaluation » (acronyme : TOE, *Target of Evaluation*). Une évaluation étant une prestation d'expertise *informatique*, elle met surtout l'accent sur les mesures de sécurité *informatiques* de la TOE, généralement exprimées sous la forme de comportements de sécurité observables de la TOE ; c'est ce qu'on appelle les *fonctions de sécurité*. On peut

donc reformuler la définition ci-dessus en exprimant qu'une évaluation est une expertise indépendante visant à confirmer que les fonctions de sécurité d'une TOE contrent les menaces et remplissent les règles spécifiées pour cette TOE. Le document dans lequel cette problématique et ces fonctions de sécurité sont spécifiées et justifiées s'appelle une « Cible de sécurité » (acronyme : ST, *Security Target*) ; elle constitue le document d'entrée de l'évaluation.

Dans le but de valider les annonces faites dans la ST, l'évaluation consiste à appliquer un certain nombre de *mesures d'assurance sécurité* sur la TOE, qui sont de deux types :

1. Des mesures correspondant à des bonnes pratiques en matière de développement logiciel (assurance qualité), mais orientées spécifiquement sous l'angle de la réalisation conforme des fonctions de sécurité ;
2. Des mesures d'*estimation de la vulnérabilité*, qui sont spécifiques à l'assurance sécurité, et par lesquelles on cherche à démontrer que les moyens nécessaires pour mettre en défaut les fonctions de sécurité, en réalisant des menaces ou en enfreignant des règles spécifiées, sont supérieurs à un certain niveau, voire ne sont pas atteignables dans la pratique.

Les jeux de critères ITSEC et CC [1,3,4,5] spécifient des mesures d'assurance sécurité sous la forme de niveaux d'évaluation (E1 à E6 et EAL1 à EAL7 respectivement).

En pratique, une évaluation fait intervenir quatre types d'acteurs :

1. Un commanditaire, *maître d'ouvrage de la sécurité*, qui a la responsabilité de l'application des mesures d'assurance sécurité et désire se prévaloir d'une attestation confirmant qu'elles ont été appliquées conformément aux standards en vigueur. L'application des mesures d'assurance sécurité par le commanditaire consiste à imposer les bonnes pratiques à la maîtrise d'œuvre de la TOE, et à produire certains argumentaires exigés par les critères d'évaluation ;
2. Un centre d'évaluation (CESTI), centre d'expertise sécurité indépendant, qui confirme que les bonnes pratiques sont appliquées, valide les argumentaires, et produit certaines études indépendantes sur la TOE. Les résultats de l'expertise du CESTI sont consignés dans un *Rapport Technique d'Évaluation* (RTE) ;
3. Les développeurs : l'ensemble de la maîtrise d'œuvre de la TOE ;
4. L'organisme de certification qui produira l'attestation de la bonne application des mesures d'assurance sécurité sur la base du contenu du RTE (le *certificat de la sécurité des TI*), supervise les travaux d'évaluation et agrée plus généralement les CESTI pour mener les travaux d'évaluation de manière conforme. La justification technique de la délivrance du certificat est consignée dans un *Rapport de Certification*, qui reprend les conclusions du RTE et peut ajouter des éléments supplémentaires, à la discrétion de l'organisme de certification.

Sans rentrer dans les détails, il s'avère que ce processus d'évaluation/certification est souvent jugé lourd, d'autant plus que le certificat de la sécurité des

TI ne porte que sur une version précise de la TOE (celle qui a été soumise à l'évaluation). La problématique de la reconduction des résultats de l'évaluation d'une TOE ses versions successives s'est donc posé très tôt, dès la parution des critères ITSEC.

2.2 Les dispositions de l'ITSEM

Le Manuel d'Évaluation de la sécurité des Technologies de l'Information (ITSEM) est la méthodologie applicable pour mener les évaluations suivant les critères ITSEC. L'annexe 6.D de l'ITSEM [2] couvre le sujet de l'« analyse d'impact pour la ré-évaluation ». Le processus d'analyse d'impact consiste à *typer les changements* puis à *déterminer leur résultat* sous la forme d'activités de ré-évaluation à effectuer par le CESTI ré-évaluateur.

Le typage de chaque changement est basé sur sa *cause*. Les causes couvertes sont :

- la modification de la problématique de sécurité (nouvelle menace) de la fonctionnalité de sécurité (nouvelle fonction), ou de la résistance annoncée des mécanismes,
- la découverte de vulnérabilités exploitables,
- un changement dans les documents de développement, incluant la représentation de l'implémentation (code source, schémas descriptifs du matériel, etc.),
- un changement dans les procédures de développement,
- un changement dans les outils de développement,
- un changement dans la documentation d'exploitation,
- un changement dans les procédures de livraison ou d'installation,

La détermination du résultat de l'analyse d'impact fait intervenir le *type des composants* ayant subi des modifications. Les *composants* sont les parties de la TOE qui apparaissent aux différents niveaux de développement (sous-systèmes, modules, etc.) Les types de composants possibles sont les suivants : « Dédié à la sécurité », « Touchant à la sécurité » ou « Non touchant à la sécurité ».

En résumé : si les changements ne portent que sur des composants Non touchants à la sécurité ou Touchants à la sécurité mais à un niveau de description qui n'a pas fait l'objet de l'évaluation (ex : représentation de l'implémentation par rapport au niveau E1), le commanditaire produit les éléments de preuve de la maintenance de l'assurance sécurité (notification des changements, éléments de preuve en appui, et résultats des tests de non-régression). À partir du moment où des composants Dédiés à la sécurité, des documents ou des procédures ayant fait l'objet de l'évaluation ont changé, un CESTI doit ré-appliquer certaines tâches d'évaluation. Les tâches d'évaluation peuvent être de simples revues des nouveaux documents, le repassage de tests de conformité ou de pénétration, ou un ré-examen global de la sécurité de la TOE. C'est à ce moment-là qu'on parle de *ré-évaluation*

L'intérêt de ces dispositions est que de règles très précises sont fournies concernant la nature des activités de ré-évaluation à entreprendre afin de prendre

en compte les changements. Le fait de lier la portée des changements au type des composants est une tendance qu'on retrouve par la suite.

La ré-évaluation doit être planifiée dès l'évaluation initiale, et le Rapport Technique d'Évaluation doit contenir un chapitre spécial de « Conseils pour la ré-évaluation et l'analyse d'impact » contenant notamment une identification et un typage des items qui devront être suivis pour la maintenance (y compris les outils de développement).

Le §6.3 de l'ITSEM précise que c'est le commanditaire qui a la responsabilité d'identifier les changements, d'en déterminer les conséquences sur le rapport de certification et d'en informer l'organisme de certification. Par contre l'ITSEM ne fournit d'indication ni sur la manière d'organiser le suivi des changements chez le développeur, ni sur des audits de ces procédures de suivi par l'organisme de certification ou un CESTI, afin de s'assurer que le commanditaire se donne des bons moyens d'assurer ce contrôle, ni sur la fréquence des contrôles du type des changements par le commanditaire.

D'autres points restent assez flous dans l'ITSEM : en cas de décision de ré-évaluation, comment décider de la portée précise de cette prestation ? Au contraire, si le commanditaire annonce qu'aucun des changements opérés ne nécessite de ré-évaluation, sur quelle base l'organisme de certification valide-t-il cette estimation ? Le commanditaire est censé, à cette occasion, fournir de la documentation de test, mais que doit-il tester exactement ?

2.3 La classe AMA des Critères Communs

La maintenance de l'assurance sécurité spécifiée par la classe AMA des Critères Communs v. 2.1[2] couvre explicitement « la découverte de menaces ou de vulnérabilités nouvelles, les changements dans les exigences de l'utilisateur, la correction des bogues trouvés dans la TOE qui a été certifiée, et les autres mises à jour des fonctionnalités fournies. » [5, §15.1, p. 183].

À la différence de l'ITSEM, un modèle de processus de maintenance est défini, comprenant :

- une phase initiale d'*acceptation de la maintenance de l'assurance sécurité* formalisée par le commanditaire dans un *plan de maintenance de l'assurance sécurité*,
- et une phase de *surveillance* pendant laquelle le développeur suit les changements opérés sur la TOE et analyse leur impact sur la sécurité, sous la surveillance d'un *CESTI mainteneur* (audit). (cf. fig. 1)

« Le plan de maintenance de l'assurance sécurité définit le champ d'application des changements qui peuvent être effectués sur la TOE *sans déclencher une ré-évaluation.* » [5, §15.3.1, p. 188]. Autrement dit, un cycle de maintenance lié à une version initiale et à une évaluation associée de la TOE se termine lorsque la portée des changements nécessite d'effectuer une ré-évaluation. La nouveauté importante du concept de maintenance de l'assurance sécurité mis en

[2] Cette classe a été retirée de la version 2.2 des Critères Communs, qui est celle actuellement applicable dans le cadre du Schéma Français d'Évaluation et de Certification.

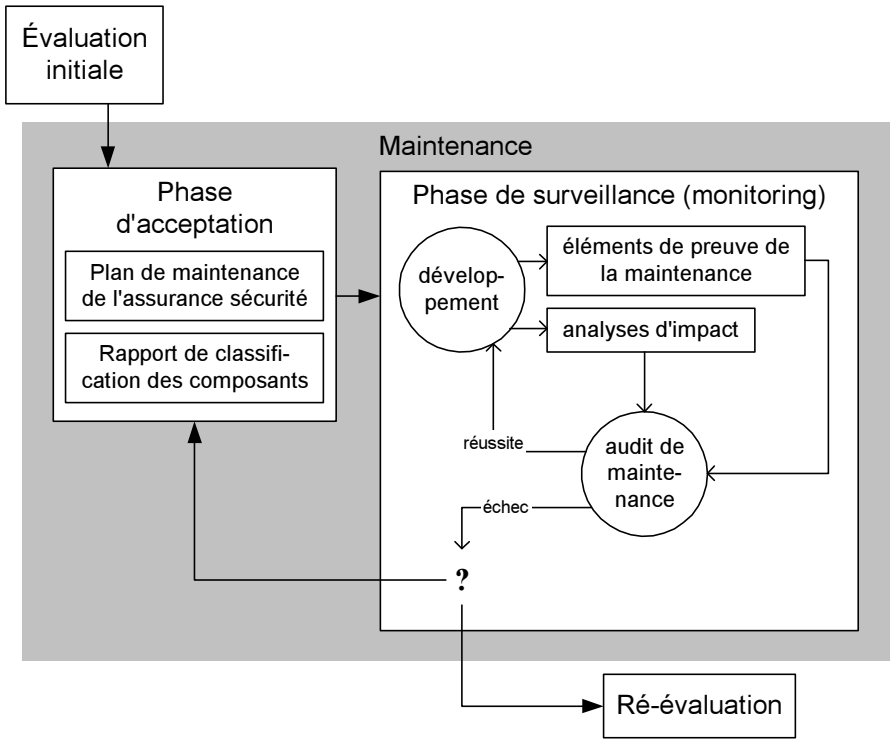

Fig. 1. *Modèle de processus de maintenance selon la classe AMA des Critères Communs*

œuvre par la classe AMA est donc de permettre de « procurer la confiance que l'assurance établie dans une TOE est maintenue, *sans nécessiter à chaque fois une ré-évaluation formelle des nouvelles versions de la TOE* » [5, §15.1, p. 183]. Les CC v. 2.1 opposent la maintenance de l'assurance sécurité à la ré-évaluation, alors que dans l'ITSEM, la ré-évaluation est une des modalités possibles de la maintenance.

Afin d'obtenir l'*acceptation* du cycle de maintenance, le commanditaire doit fournir à l'organisme de certification :

- le plan de maintenance de l'assurance sécurité (dont le contenu est spécifié par le composant[3] AMA_AMP.1),
- un *rapport de classification* des composants (composant AMA_CAT.1).

[3] Un composant d'assurance sécurité Critères Communs est une ensemble d'exigences d'assurance sécurité élémentaires qui spécifient des tâches d'assurance sécurité du commanditaire et de l'évaluateur. Les composants sont groupés en *familles* puis en *classes* (par ex. : la classe AMA).

La classification des composants correspond au typage des composants de l'ITSEM, mais le commanditaire est libre de proposer sa propre typologie – il faut seulement que cette typologie permette de distinguer les composants Dédiés à la mise en œuvre de la politique de sécurité de ceux Non dédiés – ainsi que des méthodes de classification des composants éventuellement ajoutés en cours de cycle de maintenance.

Le plan de maintenance de l'assurance sécurité permet d'encadrer de manière rigoureuse le processus de maintenance à travers la définition :

- De la *portée* des changements couverts par le cycle de maintenance ;
- De l'*échéancier* du cycle de maintenance (et notamment de sa durée maximum) ;
- Des *procédures de maintenance* devant notamment permettre la production d'*éléments de preuve de la maintenance de l'assurance sécurité* ;
- De l'identité des *analystes de sécurité du développeur*, et de leurs responsabilités organisationnelles chez ce dernier, d'une manière qui mette en évidence que leurs attributions leur donnent autorité pour mettre en œuvre les procédures de maintenance de l'assurance sécurité et leur fournissent les moyens d'effectuer correctement les analyses d'impact.

Le CESTI mainteneur évalue le rapport de catégorisation et le plan de maintenance de l'assurance sécurité selon les familles AMA_CAT et AMA_AMP. Si le cycle de maintenance est accepté par l'organisme de certification, on passe en *surveillance*.

Pendant la surveillance :

- Pour chaque version succédant à la version certifiée de la TOE, les analystes de sécurité du développeur produisent une *analyse d'impact* sur la base de laquelle les résultats de l'évaluation initiale sont maintenus. L'analyse d'impact identifie et décrit :
  - les changements opérés sur les composants Dédiés à la mise en œuvre de la politique de sécurité, correspondant à chaque version successive de la TOE,
  - les changements associés sur les fournitures soumises à l'évaluation initiale,
  - les tests de non-régression de la sécurité ;
- À chacun des moments spécifiés par le plan de maintenance de l'assurance sécurité, le CESTI mainteneur procède à un *audit* visant à établir la confiance dans le fait que l'assurance dans la sécurité de la TOE est maintenue correctement par le développeur. Cet audit consiste à vérifier les analyses d'impact et les éléments de preuve de la maintenance de l'assurance sécurité, produits par les analystes de sécurité du développeur, sous l'angle des critères des familles AMA_SIA et AMA_EVD, ce qui implique de :
  - Contrôler la bonne application des procédures, et notamment la correction des analyses d'impact ;
  - Vérifier, sur la base des tests du développeur, la non-régression vis-à-vis de la sécurité.

Il est important de noter que les CC v. 2.1 n'explicitent pas le fait que l'organisme de certification puisse certifier les versions successives de la TOE, sur lesquelles les résultats de l'évaluation initiale ont été maintenus. L'adoption de règles spécifiques de re-certification est du ressort des schémas nationaux d'évaluation et de certification.

Par ailleurs, il apparaît que deux types d'échec peuvent survenir suite à un audit :

- Soit une non-conformité dans l'application des procédures de maintenance, ce qui remet en cause la confiance exprimée par le commanditaire, mais pas obligatoirement la confiance intrinsèque dans la version maintenue de la TOE ;
- Soit des problèmes mis en évidence par l'examen des éléments de preuve eux-mêmes, qui obèrent formellement la confiance dans la version maintenue de la TOE.

Les CC v. 2.1 ne font pas du tout cette distinction n'explicitent pas les conséquences que devrait avoir chaque type d'échec sur la prolongation du processus de maintenance de l'assurance sécurité (cf. le point d'interrogation dans la figure 1).

Enfin, les exigences de la classe AMA n'ont pas bénéficié des éclaircissements apportés par la parution de la Méthodologie d'Évaluation Commune (CEM) [6] car ils ne sont pas couverts par ce document. En décembre 2001, le CCIMB publia un supplément à la CEM [8], qui visait à décrire les activités d'évaluation auxquelles le CESTI devait se livrer dans le cadre de la maintenance de l'assurance sécurité. Des exigences minimum sur la gestion de la maintenance dans les schémas d'évaluation et de certification furent formulées et la classe AMA fut refondue.

Précisons que cette refonte de la classe AMA n'a jamais été applicable dans le cadre du Schéma Français d'Évaluation et de Certification.

On peut caractériser les changements entre la classe AMA des CC v. 2.1 et le document de refonte de la classe AMA [8] comme suit :

1. *La prise en compte la re-certification.* Le document introduit la notion *d'étape de maintenance* (*maintenance step*) qui correspond à une présentation des différentes analyses d'impact, et qui sont validées *a posteriori* suite aux audits de maintenance. En cas d'échec à l'audit, on révoque les certificats associées à toutes les étapes de maintenance survenues depuis le dernier audit ;
2. Les analystes de sécurité du développeur voient leur champ de compétence accru : ils ont maintenant également la charge de *mettre à jour les éléments de preuve produits par l'évaluateur* au cours de l'évaluation initiale. Tout en étant rattachés au développeur, ils doivent présenter une indépendance suffisante par rapport au processus de développement, ainsi que par rapport à l'évaluateur. Le développeur n'a plus qu'à gérer la mise à jour des fournitures de développement et à les transmettre aux analystes de sécurité.

À part cela, les re-formulations opérées, entre la v. 2.1 des CC [5] et le document de refonte de la classe AMA [8], sur le texte des composants consistent essentiellement à rajouter des précisions sur certains points (ex : mécanisme de

contrôle de l'application des procédures de maintenance) et à en gommer certains autres (ex : portée minimum de ces procédures).

Une note au début du document précise que certaines des notions présentées pourraient être transférées dans des exigences minimales sur des schémas d'évaluation et de certification, dans le cadre de la reconnaissance mutuelle internationale (CCRA). Cela suggère que les rédacteurs ont dû être confrontés à l'interrogation suivante : est-il approprié de coder la définition du processus de maintenance d'assurance sécurité lui-même comme des exigences d'assurance sécurité CC ?

Notre réponse est non, dans la mesure où les exigences sur la maintenance sont avant tout une affaire de définition des rôles et des responsabilités des différents acteurs de la maintenance, et que le contrôle du respect de ces exigences doivent sous-tendre le processus d'évaluation, plutôt qu'en faire partie. Les CC n'ont pas été rédigés avec l'intention de spécifier, dans leur contenu, des exigences sur les procédures des schémas d'évaluation et de certification. Ils spécifient de manière générale comment obtenir un résultat d'expertise sécurité, mais ne couvrent ni les détails spécifiques de l'expertise, ni le contexte réglementaire et organisationnel qui permet de passer de ces résultats à un certificat de la sécurité des TI (le schéma d'évaluation et de certification), surtout quand ces résultats sont produits sur la base d'activités dirigées par le commanditaire ou le développeur (à travers l'analyse d'impact). Vu sous cet angle, la forme du document de refonte de la classe AMA était inappropriée.

Faut-il pour autant abandonner la maintenance de l'assurance sécurité, telle qu'elle est spécifiée dans le document [8] ? À tout le moins, on ne peut que comprendre que la multiplication des rôles et des exigences spécifiées dans ce document ait rebuté les commanditaires.

2.4 Le guide ECF 12

Le guide ECF 12 [7] (maintenant retiré) est la partie de l'ancien Schéma Français d'Évaluation et de Certification qui traitait des programmes de maintenance des certificats. Il constituait l'implémentation dans le Schéma Français des directives de l'ITSEM et de la classe AMA des Critères Communs v. 2.1.

Le commanditaire doit préparer un dossier de maintenance qui contient principalement le plan de maintenance de l'assurance sécurité et le rapport de classification des composants. Un CESTI mainteneur est choisi et il effectue obligatoirement un audit initial des procédures de maintenance du commanditaire préalablement à l'acceptation de la maintenance par l'organisme de certification. Ensuite, le cycle est globalement conforme au modèle de la classe AMA, à une nuance terminologique près, détaillée ci-dessous. Le guide précise certains paramètres du cycle de maintenance laissés ouverts par les CC v. 2.1, par exemple :

- Le cycle de maintenance s'achève automatiquement si le volume des modifications dépasse 20% ;
- Pour une TOE initialement évaluée à un niveau EAL3 ou plus, le CESTI mainteneur doit vérifier que l'analyse d'impact couvre bien tous les chan-

gements, alors qu'avec un niveau EAL2 ou moins, il peut procéder par échantillonnage sur les changements documentés ;
– La portée du contenu de l'analyse d'impact, des modifications à opérer sur les fournitures, et des travaux d'évaluation dépend de la nature des changements (modifications de la cible de sécurité, de la conception, etc.)

Le guide ECF 12 présente en fait deux types de cycles : un *cycle de maintenance* et un *cycle de surveillance*. Le cycle de maintenance donne lieu à une revue de l'analyse d'impact puis, si elle est jugée formellement conforme et que les changements qu'elle décrit rentrent dans le cadre du plan de maintenance de l'assurance sécurité, à une évaluation des éléments de preuve de la maintenance de l'assurance sécurité, permettant de maintenir le certificat. Le cycle de surveillance vise surtout à vérifier par des audits périodiques que les procédures de maintenance sont toujours correctement appliquées par le commanditaire, sans forcément émettre un avis explicite sur la sécurité de la version courante de la TOE. Cela rend compte de la nuance, que la classe AMA ne fait pas, entre l'incapacité du commanditaire à appliquer les procédures de maintenance de la TOE, et la non-conformité intrinsèque d'une nouvelle version de la TOE vis-à-vis de la sécurité.

Le guide ECF 12 a été applicable de décembre 2000 à décembre 2003. Vers la fin de 2002, les retours d'expérience présentés par la DCSSI faisaient état de doutes sur l'intérêt des processus de maintenance. Leur valeur ajoutée dans le cas où ils auraient précédé des ré-évaluations n'était pas claire. Dans le cas où le commanditaire envisageait uniquement de la veille technologique vis-à-vis des vulnérabilités des versions successives de la TOE, sans finalité de ré-évaluation, ils paraissaient surdimensionnés.

Il est possible que les commanditaires aient perçu que ce genre de démarche présentait l'inconvénient suivant : ce que l'on gagne en n'effectuant pas de ré-évaluation, on peut le perdre en produisant les documents spécifiques de maintenance de l'assurance sécurité.

À notre connaissance, aucune maintenance de certificat n'a été menée sur des produits logiciels, pour des raisons financières. Dans le domaine de la carte à puce, plusieurs programmes ont été lancés, mais ils n'ont pas permis de suivre le rythme d'apparition des nouvelles versions. Par contre, cette expérience a permis de mettre en place des actions de mutualisation des audits de sites de fabrication dans le cadre de cycles de maintenance sur des produits apparentés.

2.5 La famille ALC_FLR des Critères Communs

La famille ALC_FLR décrit des exigences sur un processus de suivi et de correction des failles de sécurité et de distribution aux utilisateurs finaux. L'évaluation porte uniquement sur la documentation des procédures de suivi fournie lors de l'évaluation initiale, et on ne va pas vérifier que les procédures sont effectivement appliquées, ni pendant l'évaluation, ni après (par ex. : par un audit).

Par rapport à sa formulation initiale dans les CC v. 2.1, cette classe a été refondue à l'occasion de la parution d'une méthodologie d'évaluation dédiée [9], à peu près vers la même période que les travaux correspondants sur la classe

AMA. Mais cette re-formulation et la méthodologie d'évaluation associée ont été intégrées dans les CC v. 2.2 et sont actuellement applicables.

En résumé, les exigences de la famille ALC_FLR spécifient l'existence, au sein du processus de développement, de *procédures de correction des failles de sécurité* (*security flaws*) ayant pour effet de permettre la publication d'informations générales sur les failles à destination des *utilisateurs*, et la distribution d'informations plus spécifiques, accompagnées de correctifs, à des *utilisateurs enregistrés*.

Les procédures de corrections des failles doivent couvrir le *cycle de vie* des anomalies : détection, qualification en tant que faille de sécurité, gestion de l'état (ex : suspectée, confirmée, corrigée), identification d'actions correctives, test de non-régression, etc. Elles doivent également couvrir le traitement des signalements par les utilisateurs et de leurs requêtes concernant l'état courant des failles détectées.

La famille ALC_FLR implique également l'existence et l'évaluation de *guides de soumission des anomalies* à destination des utilisateurs.

C'est actuellement ce type d'assurance sur la maintenance qui est le plus mis en œuvre dans le Schéma Français d'Évaluation et de Certification à travers la procédure de qualification mise en œuvre par la sous-direction régulation de la DCSSI [11], qui impose une évaluation selon un paquet d'assurance sécurité contenant le composant ALC_FLR.3). Nous avons pu constater qu'il présente l'avantage de pouvoir s'adapter à l'existant en matière de procédures de suivi et de correction de bugs chez les éditeurs de solutions logicielles de sécurité. Par ailleurs, dans le cadre d'une maintenance de l'assurance sécurité appliquée à un système intégré (q.v.), le fait de disposer de ce genre de processus de correction des failles de la part des fournisseurs des composant du système paraît être le minimum requis.

Nous pensons que, complété par un audit régulier du processus de correction des failles, ces dispositions pourraient constituer un niveau élémentaire, acceptable et pratique, de maintenance de l'assurance sécurité.

2.6 La procédure SUR/P/01.1

Suite au retrait de l'ECF 12, les trois types de besoins suivants ont été distingués vis-à-vis de la maintenance de l'assurance sécurité :

1. **La veille sur la résistance d'un produit certifié** : il s'agit de suivre dans le temps l'érosion de la sécurité d'une version donnée d'un produit. Ce besoin est principalement remonté par les banques, qui diffusent des cartes pour une durée de quelques années. Par contre, pour les systèmes informatiques dont les composants logiciels sont facilement et fréquemment mis à jour, ce besoin n'est pas applicable ;
2. **La re-certification suite à l'application de correctifs** : concerne au contraire les produits fortement évolutifs. Le certificat portant sur une version unique de la TOE, l'application de patches par l'utilisateur obère formellement la confiance établie par l'évaluation et attestée par le certificat, or, paradoxalement, une version ayant subi des correctifs de sécurité devrait

a fortiori être jugée plus sûre que la version de base non patchée, moyennant une application correcte des procédures de test de non-régression. *L'absence de solution à cette contradiction est de nature à dissuader les éditeurs de logiciels d'investir dans des projets d'évaluation et de certification de leurs produits* ;

3. **La maintenance d'environnement de développement** : parmi les activités d'évaluation, on trouve des audits des procédures de développement des produits : gestion de configuration, sécurité de l'environnement de développement, procédures de livraison sécurisée. Plutôt que de mener ces activités à l'occasion de chaque projet d'évaluation, l'idée est de mener ces audits indépendamment des projets et avec une périodicité régulière, puis, lors de chaque évaluation, de vérifier que les sites de développement et de production annoncés pour la TOE présentent des résultats d'audit récents et d'une portée applicable à cette TOE.

La procédure SUR/P/01.1 [10] répond au premier type de besoin. Cette maintenance mise en œuvre par la surveillance ne concerne, par définition, que la version certifiée de la TOE et porte essentiellement sur les *analyses de vulnérabilités* qui ont été produites à l'occasion de l'évaluation.

Le processus consiste à faire réaliser tous les six mois, ou à la demande de l'organisme de certification, une *analyse de résistance* par le CESTI en charge de la surveillance. Il s'agit d'une mise à jour de l'analyse de vulnérabilité et de l'analyse de résistance des fonctions établies initialement, suite à l'évolution de l'état de l'art. Le CESTI émet également un rapport présentant le résultat de l'analyse. En fonction du résultat, la surveillance peut être poursuivie ou arrêtée. Il n'y a donc pas de reconduction formelle du certificat.

Ce genre de processus paraît applicable lorsque le commanditaire possède un intérêt direct à retirer un produit qui présente des failles de sécurité, c'est-à-dire lorsque le produit protège des biens qui sont sous sa responsabilité ou sous celle d'entités auxquelles le commanditaire doit fournir des moyens de protection, de par les missions qui lui incombent. Dans ces contextes, le commanditaire est distinct du développeur, et il a un intérêt autre que marketing dans l'obtention du certificat. En d'autres termes, il est *Maître d'Ouvrage* de la sécurité et non *Maître d'Œuvre*.

À notre connaissance, cette séparation commanditaire/développeur constitue la règle dans le cas des évaluations de cartes à puce, alors qu'elle est exceptionnelle dans le domaine du logiciel sur plate-forme générique.

Par contre, l'obligation de lancer un programme de surveillance dans le cadre de la *qualification* des produits « susceptibles de protéger les informations de l'administration » [11] va probablement changer la donne en obligeant des commanditaires-développeurs industriels qui n'auraient pas *a priori* lancé ce genre de programme à le faire.

2.7 L'accord de reconnaissance mutuelle sur la continuité de l'assurance sécurité

L'abandon de la classe AMA a été principalement motivé par le manque d'accroche des commanditaires pour un processus leur attribuant la maintenance des éléments de preuve du développeur et des traces de l'évaluation ; processus ayant été jugé consommateur de ressources, et apportant peu de valeur ajoutée dans le cadre d'une ré-évaluation.

De surcroît, ces dispositions ne permettaient pas d'établir une reconnaissance mutuelle de leur transpositions dans les différents schémas nationaux d'évaluation et de certification. Le problème principal, actuellement, c'est qu'il n'y a ni certitudes, ni consensus sur la manière d'émettre un certificat de la sécurité des TI autrement que suite à une évaluation, éventuellement conduite « en delta » (ré-évaluation). Une approche commune ne pouvait donc pas relever de la re-certification, mais devait cibler une forme allégée de maintenance de l'assurance sécurité.

Le document « *Assurance Continuity : CCRA requirements* » [12] vise à proposer une approche minimale conforme à l'Accord de Reconnaissance Mutuelle des Certificats Critères Communs (CCRA). Ce document – applicable en l'état depuis mars 2005 dans le cadre du Schéma Français d'Évaluation et de Certification – répond au deuxième type de besoin identifié au début du §2.6 – la prise en compte de l'application de correctifs, ou de tout autre type de modification, sur la version certifiée – mais on introduit une distinction entre une *TOE certifiée* et une *TOE maintenue*.

Une version de la TOE est dite maintenue, par rapport à une version certifiée de référence, lorsque l'organisme de certification atteste que les changements opérés entre les deux versions sont *mineurs*. S'il y a un changement *majeur*, la re-évaluation est obligatoire.

Sinon, l'identifiant de la TOE maintenue est publié dans *l'addendum de maintenance* du certificat de la sécurité des TI initial, et de même un *rapport de maintenance* détaillant les changements est publié en tant qu'addendum du rapport de certification initial. Notez que cela ne signifie pas, selon la DCSSI, que la version maintenue est formellement certifiée. On ne se prononce pas non plus sur le fait que si une évaluation avait été effectuée sur la version maintenue plutôt que sur la version certifiée, le résultat aurait aussi été positif.

Des changements mineurs typiques sont :

- des changements de la plate-forme logicielle ou matérielle sous-jacente, sans changements de l'interface régissant les interactions entre la TOE et cette plate-forme ;
- des changements qui n'impactent pas les éléments de preuve ; par exemple, des changements portant uniquement sur le code source d'une TOE évaluée et certifiée selon le niveau EAL1 ;
- des changements éditoriaux qui n'impactent pas la problématique ou la fonctionnalité de sécurité sur le fond ;
- des changements sur les procédures ou les outils de l'environnement de développement (ex : gestion de configuration), qui n'ont pas d'effet sur le

résultat de l'application de ces procédures ou de ces outils (ex : établissement de la liste de configuration).

En entrée de chaque étape de maintenance, conduisant éventuellement à l'émission du rapport de maintenance par l'organisme de certification, le commanditaire fournit un *Rapport d'Analyse d'Impact* décrivant les changements opérés et leur impact sur les éléments de preuve du développeur.

On ne parle pas de maintenir les traces produites par l'évaluateur ; au contraire, il est dit qu'« un *changement mineur* est un changement dont l'impact est suffisamment réduit pour ne pas affecter la confiance dans un mesure telle que les tâches de l'évaluateur doivent être ré-appliquées de manière indépendante ».

Des exigences sur la présentation du Rapport d'Analyse d'Impact sont énoncées (chapitre 5), mais on ne précise pas si la validation de ce rapport est directement du ressort de l'organisme de certification, ou s'il délègue cette expertise à un CESTI. Au contraire, l'une ou l'autre formule sont autorisées par le document.

Nos critiques portent sur deux points :

- *La portée des changements couverts est vraiment réduite.* La ligne de démarcation entre changements mineurs majeurs est exactement la même que celle que l'ITSEM [2] établissait entre les changements ne portant que sur des composants Non touchants à la sécurité ou Touchants à la sécurité mais à un niveau de description qui n'a pas fait l'objet de l'évaluation, et les autres changements, ayant un impact sur les éléments de preuve de l'évaluation initiale (cf. §2.2)[4]. En conséquence, ce processus ne peut accompagner des « releases » successives d'une TOE certifiée que si le niveau de l'évaluation initiale ne couvrait pas le code source et la conception de bas niveau[5], ce qui est une incitation pour le commanditaire à ne pas faire évaluer la TOE initiale au dessus d'EAL2. Le moindre changement dans la fonctionnalité de sécurité observable est qualifié de majeur ;
- *La signification exacte de la maintenance est ambiguë.*
  - D'un côté, on fait bien la différence entre TOE certifiée et TOE maintenue.
  - De l'autre, on définit les changements mineurs, ouvrant la voie à la maintenance, d'une manière qui est quasiment synonyme de certification et qui interdit pratiquement toute possibilité de réfuter que les résultats de l'évaluation de la TOE certifiée pourraient aussi être utilisés pour certifier la version maintenue[6].

[4] Dans l'ITSEM, la « notification des changements » doit être accompagnée des résultats des tests de non-régression, ce qui n'est pas le cas dans le document [12].

[5] Et encore, pas trop de releases, car un nombre « significatif » de changements mineurs vaut pour un changement majeur.

[6]
- [12, §9 c] « Le certificat de la TOE certifiée s'applique également [à la TOE maintenue], la confiance acquise dans la TOE certifiée s'applique aussi à la TOE maintenue ».

3 La maintenance de l'assurance sécurité des Systèmes Informatiques

3.1 Le bilan des approches concernant les produits

Les péripéties de la maintenance de l'assurance sécurité donnent l'impression d'un trajet aller-retour :

- partant de l'ITSEM [2] qui impose une ré-évaluation, sauf pour les changements qui n'impactent pas les fournitures de l'évaluation ;
- puis allant jusqu'à l'extrémité opposée constituée par la classe AMA [5,8] et l'ECF 12 [7], dans lesquels on fait confiance à la compétence et à la motivation du commanditaire pour maintenir les éléments de preuve de l'évaluation ;
- et revenant finalement à une approche plus *conservatrice* où la ré-évaluation redevient nécessaire à partir du moment où l'impact des changements est majeur, c'est-à-dire observable au niveau des éléments de preuve qui ont fait l'objet de l'évaluation ; cette approche est transcrite dans l'accord de reconnaissance mutuelle sur la continuité de l'assurance sécurité [12].

Au passage, on a construit :

- une reconnaissance mutuelle internationale sur l'applicabilité de l'approche conservatrice ;
- des approches alternatives ou complémentaires de la maintenance de l'assurance sécurité : exigences sur les processus de suivi et de correction des failles de sécurité et de distribution des correctifs aux utilisateurs finaux, audits mutualisés de sites de développement, ... ;
- une compréhension plus fine des événements relevant de la maintenance de l'assurance sécurité des produits : d'un côté la succession des versions des produits présentant des évolutions rapides, de l'autre côté l'érosion de la résistance d'un produit qui évolue peu ou pour lequel la mise à jour fréquente ou l'application de correctifs n'est pas envisageable.

3.2 Un modèle de processus de gestion des événements de sécurité

Dans la suite, nous envisageons principalement la prise en compte des changements portant sur la facette *technique* du Système d'Information (le Système Informatique). La raison en est que les différents approches de la maintenance de l'assurance sécurité que nous avons analysées au §2 ne couvrent pas les aspects organisationnels.

- [12, §42] « Un changement mineur est un changement dont l'impact est suffisamment réduit pour ne pas affecter la confiance dans une mesure telle que les tâches de l'évaluateur doivent être ré-appliquées de manière indépendante ».
- [12, §45] « Les changements mineurs consistent typiquement en des changements sur la TOE qui n'ont pas d'effet sur la validité des annonces faites dans la cible de sécurité. »

Dans le domaine de l'évaluation ITSEC ou CC, on définit traditionnellement un *produit* comme le résultat d'un développement informatique destiné à être intégré dans de nombreux Systèmes Informatiques et dont le mode d'exploitation est décrit par des hypothèses génériques, par opposition à un *système* informatiques intégré dans un environnement d'exploitation bien défini, existant ou dont la création est planifiée. Les systèmes sont intégrés à partir de développements spécifiques et, de plus en plus, à partir de produits.

Les approches du §2 sont avant tout orientées produit. En effet, elles visent à maintenir l'assurance sécurité produite par des évaluations selon les critères ITSEC ou CC, dont le biais vis-à-vis des produits et l'inadaptabilité vis-à-vis des systèmes sont des caractéristiques connues.

Par delà la lacune que constitue le fait de ne pas auditer l'environnement d'exploitation du système évalué[7], le fait d'évaluer un système comme un « gros produit » se heurte :

1. à la complexité et à la volumétrie supérieure de l'architecture ; d'une petite dizaine à plusieurs milliers d'équipements pour les très gros systèmes contre cinq ou six composants au maximum pour les produits ;
2. ainsi qu'au manque relatif de liberté dans les choix de conception ; on doit en effet souvent faire avec ce qui est disponible sur le marché pour satisfaire tel ou tel besoin de l'architecture d'un système, alors que le concepteur d'un produit maîtrise tous les choix de conception jusqu'au développement.

En fonction de la nature des composants du système, du degré d'hétérogénéité de leurs processus de maintenance respectifs, ainsi que du rythme de parution, de la nature et de la qualité des descriptions des alertes de sécurité portant sur ces composants, une transposition naïve des dispositions décrites au §2 risque fort de provoquer une surcharge du processus de suivi des changements, qui s'avérera alors plus consommateur de ressources que les ré-expertises qu'elles visent à éviter.

Comment ces notions peuvent-elles être utilisées de manière pertinente dans le cadre des Systèmes Informatiques intégrés, ayant fait l'objet d'une expertise initiale ?

3.3 Une démarche de détection et de gestion des événements touchant à la sécurité

Nous proposons une démarche permettant d'établir des procédures de détection et de gestion des *événements touchant à la sécurité*, c'est-à-dire tous ceux qui pourraient mettre en cause le résultat de l'expertise de sécurité initiale, qu'ils soient liés à des changements sur le système, ou bien qu'ils correspondent à l'apparition de nouvelles vulnérabilités, de nouvelles techniques d'attaque ou à des changements dans les conditions d'exercice des attaques existantes.

[7] Les critères ne le demandent pas, mais les CESTI le font, en mettant ces activités d'audit technique et organisationnel en correspondance avec certains aspects liés aux mesures d'assurance sécurité relatives aux tests, à la conformité de la documentation d'exploitation, et à l'analyse de vulnérabilité.

Cette démarche est basée sur des hypothèses minimum concernant d'une part l'approche ayant abouti à la conduite de l'expertise initiale et d'autre part les traces produites lors de cette expertise.

Ce qui est décrit ci-dessous s'applique à un système de taille moyenne, présentant une unité des responsabilités vis-à-vis de la sécurité. Dans le cas de gros systèmes, répartis sur de multiples sites, et dont la responsabilité est déléguée à différentes entités organisationnelles, il importe de faire la part, dans ce qui est décrit ci-dessous, entre et les principes centralisateurs et généraux et les dispositions particulières qui doivent être instanciées au niveau de chaque site.

Les étapes de la démarche sont :

– Établir une *description de l'architecture* identifiant chacun des composants ou des types de composants présents dans le système. Il importe de donner une définition des composants qui soit précise et dont le niveau de granularité soit adapté au suivi des évolutions.

  Une définition précise permet notamment de faire la différence entre les *sous-systèmes* et les *équipements* intégrés dans le système et les *composants matériels et logiciels* dont ces sous-systèmes et ces équipements sont constitués. Par exemple, il convient de faire la différence entre « la station de supervision HP OpenView » et « le logiciel de supervision HP OpenView ».

  L'identification des composants de l'architecture doit être faite en liaison avec les procédures de gestion de configuration du système.

– Rédiger un *argumentaire de l'architecture de sécurité*. Le système ayant été conçu pour répondre à un certain nombre de *règles techniques de sécurité* (incluant le fait de contrer certaines menaces), l'argumentaire doit décrire le rôle de chaque composant et démontrer que la combinaison du comportement de sécurité de chacun satisfait ces règles. Cet argumentaire permet d'identifier la *criticité* des composants vis-à-vis de la sécurité et les *dépendances* qui existent entre eux. Identifier également les *éléments de configuration* (paramètres, fichiers de configuration) qui peuvent avoir une influence sur le comportement de sécurité de chaque composant.

  Dans le contexte des évaluations ITSEC ou CC, cet argumentaire correspond à l'ensemble formé par la cible de sécurité et les analyses de vulnérabilité. Il s'agit d'une démonstration, s'appuyant sur les caractéristiques techniques des niveaux de description successifs du système, du fait que l'architecture répond à la problématique de sécurité posée.

– Décrire le *cycle de vie* de chacun des composants. Identifier des dépendances éventuelles entre les cycles de vie des différents composants (ex : logiciel applicatif dépendant d'une version données du système d'exploitation). Estimer les échelles de temps associées aux cycles de vie des produits respectifs (combien de temps entre chaque version majeure ? combien de temps

entre chaque correctif ? le développeur produit-il des versions du produit spécifique au système ?), afin de repérer les composants qui vont solliciter le plus le processus de maintenance de l'assurance sécurité.

– Identifier les *événements gérés par le cycle de vie de chaque composant* et dont peut dépendre le rôle du composant dans l'architecture (son comportement de sécurité identifié par l'argumentaire). Si le cycle de vie ne traite pas explicitement ce genre d'événements, il faut partir de l'existant, anticiper ceux qui pourraient toucher en pratique à la sécurité et spécifier des règles qui vont permettre de qualifier les événements remontés par les procédures du cycle de vie sous l'angle de la sécurité. On peut par exemple adopter une définition des composants Dédiés/Non dédiés, comme au §2.3, et surveiller les événements qui affectent les composants Dédiés. On peut également s'inspirer de la famille ALC_FLR des critères Communs (cf. §2.5 et [9]).

  Cette étape permet notamment de repérer les sources d'information des événements de sécurité : changements planifiés et maîtrisés (gestion de configuration), signalement et correction des anomalies, veille sur les vulnérabilités, etc.

L'établissement de cette cartographie de l'architecture de sécurité est nécessaire aussi bien à l'organisation de l'expertise initiale qu'à celle des activités de maintenance de l'assurance sécurité. L'objectif est de concevoir des procédures de suivi des événements touchant à la sécurité du système qui permettent de juger, de la manière la plus efficace et la moins coûteuse possible, de l'impact de chaque événement sur l'assurance sécurité construite lors de l'expertise initiale.

Un *référentiel d'éléments de preuve* devra être identifié lors de la préparation de cette expertise ou au plus tard lors de la réalisation. Ce référentiel permet de repérer, parmi l'ensemble des événements touchant à la sécurité identifiés, ceux qui auront un impact observable sur les éléments preuve qui ont soutenu l'argumentaire de l'architecture lors de l'expertise initiale. À titre d'exemple, on peut citer :

1. Les dossiers de spécification et de conception des composants ayant fait l'objet d'un développement spécifique ;
2. Les dossiers justificatifs des COTS, dossiers d'architecture matérielle et autres dossiers d'intégration ;
3. La documentation des procédures d'exploitation et de maintenance ;
4. Les fichiers de configuration audités ;
5. Les traces des tests et des revues documentaires effectuées par l'expert.

La *liste de configuration* identifiant la version *précise* des composants ayant fait l'objet des tests, des fichiers de configuration audités, et des documents revus doit être établie. C'est en effet sur cette base que les changements constituant des événements touchant à la sécurité sont identifiés.

La mise en place d'un *processus de veille des vulnérabilités publiques* permet par ailleurs d'injecter dans le système les événements liés à l'apparition de vulnérabilités et à l'évolution de l'état de l'art des techniques d'attaque.

En reprenant la terminologie du CCRA (§2.7, [12]), on qualifiera de *mineurs* les événements n'ayant pas d'impact observable sur les éléments de preuve, et de *majeurs* ceux qui en ont un.

Les événements mineurs ne nécessitent pas de ré-expertise. Lorsque seuls des événements mineurs sont survenus depuis l'expertise initiale, on peut présumer que si l'expertise avait été conduite sur l'état courant du système, elle aurait donné les mêmes résultats.

Les événements majeurs peuvent impacter le comportement de sécurité d'au moins un composant. Il peut s'agir d'événements directement liés à un composant, de changements sur des composants ne faisant pas partie du référentiel d'expertise mais sur lesquels un dépendance a été identifiée, ou encore de changements dans des paramètres de sécurité ayant fait l'objet de l'expertise, ou de l'apparition de nouvelles techniques d'attaque.

3.4 La prise en compte des événements majeurs

Même s'il faudrait, en théorie, procéder à une ré-expertise partielle du système dès la détection d'un événement majeur, il peut y avoir intérêt à regrouper ces activités et à distinguer :

- les événements dont le traitement est prioritaire,
- ceux qui, tout en remettant formellement en cause le résultat de l'expertise, peuvent être traités sans faire appel à un expert indépendant.

En suivant l'esprit de la classe AMA des Critères Communs [5,8] et de l'ECF 12 [7], on peut envisager de laisser, dans le deuxième cas, aux responsables de la sécurité du système – qui « subissent » normalement l'expertise – la possibilité de démontrer, sur la base du référentiel d'expertise, que les changements n'impactent pas l'argumentaire de l'architecture de sécurité. À notre avis, cette capacité doit être limitée par les considérations suivantes :

1. Il importe de bien borner le type d'événements majeurs qui peuvent faire l'objet de ce genre de procédure. C'est normalement le but de la définition de la portée du plan de maintenance de l'assurance sécurité. En particulier, les événements qui impactent l'argumentaire de l'architecture de sécurité, ou ceux qui correspondant à de nouvelles techniques d'attaque devraient être exclus de cette portée et nécessiter une ré-expertise indépendante systématique.
2. Pour pouvoir justifier que les changements du référentiel d'expertise ne remettent pas en cause l'argumentaire de l'architecture de sécurité, encore faut-il avoir observé et compris le détail des activités de l'expertise initiale, et en quoi leurs résultats soutiennent l'argumentaire. Il faut aussi être capable de concevoir des activités de recette des changements et d'analyse des vulnérabilités publiques récentes de telle sorte qu'elles démontrent de manière pertinente que les résultats de l'expertise initiale peuvent être maintenus. Il faut pour cela, comme le suggèrent la classe AMA des CC et

l'ECF 12, auditer spécifiquement, et de manière indépendante les procédures et les équipes chargées du suivi des événements touchant à la sécurité.

En fonction des contraintes opérationnelles (compétence et dimension des équipes, dynamique des événements de sécurité, complexité du systèmes, etc.), cette démarche peut en fait s'avérer aussi coûteuse que de ré-expertiser systématiquement tous les changements liés à des événements majeurs.

On notera que ce n'est pas parce qu'une ré-expertise est jugée nécessaire qu'il faut ré-examiner le système informatique de fond en comble. Il est cependant évident que l'adoption de mesures de suivi des événements touchant à la sécurité devrait permettre, lorsque le recours à un expert est devenu nécessaire, de focaliser l'intervention de ce dernier sur les points impactés de la manière la plus critique par les changements.

4 Conclusions

La conception et l'implémentation des procédures de maintenance de la sécurité des Systèmes Informatiques constitue un problème complexe qui nécessite à la fois une connaissance des vulnérabilités, de l'expertise des architectures de sécurité, et une maîtrise des contraintes opérationnelles de suivi et de gestion des systèmes au jour le jour.

Cet article a tenté de faire le point sur la manière dont on peut s'inspirer des différentes approches de la maintenance de l'assurance de sécurité historiquement adoptées dans le cadre du Schéma Français d'Évaluation et de Certification pour les produits certifiés. Ces approches n'ont initialement pas été construites sur la base d'expériences pratiques, mais ont plutôt été proposées comme des extensions des méthodes d'évaluation et des schémas de certification existants, puis soumises à un retour d'expérience, qui a souvent été négatif et a occasionné d'importants revirements. Or, ce n'est pas parce que certaines de ces approches ont été jugées inapplicables en général, qu'elles ne pourraient pas être précisément adaptées à des contextes particuliers. Mais sur quelle base opérer le choix ?

Adoptons le raisonnement suivant : en l'absence d'un processus de surveillance des événements de sécurité, des ré-expertises sécurité systématiques doivent être conduites sur le SI à intervalles réguliers, disons tous les ans, et cela en n'ayant aucun moyen objectif de justifier, à chaque fois, que tel composant n'a pas changé depuis la dernière fois.

Une première étape est donc d'identifier les composants ayant changé entre chaque expertise.

Ensuite, on peut essayer de restreindre les activités d'expertise à ce qui est strictement nécessaire, eu égard à l'impact des changements. Beaucoup de changements opérés sur un composant donné seront des « améliorations » de ce composant ou de sa configuration, sans remise en cause de son rôle dans l'architecture. D'autres porteront sur des composants qui ne jouent pas de rôle dans la sécurité. Mais pour identifier ces changements, encore faut-il connaître et avoir formalisé ce rôle, d'où la nécessité d'établir l'argumentaire de l'architecture de sécurité.

L'étape suivante consiste à tracer la portée de l'expertise initiale pour distinguer les changements mineurs des changements majeurs.

La dernière étape consiste à distinguer, parmi les changements majeurs,ceux qui sont prioritaires vis-à-vis de la ré-expertise indépendante, et ceux qui peuvent être ré-expertisés en interne à l'organisation, ce qui nécessite d'acquérir une expertise équivalente à celle de l'expert dans le domaine de l'architecture.

Les éléments déterminant le choix des approches de la maintenance de l'assurance sécurité résident dans des caractéristiques identifiable à chacune de ces étapes de la mise en place du processus de surveillance. C'est principalement la nature et le niveau de détail des arguments développés dans l'argumentaire de l'architecture de sécurité qui conditionnent la nature des événements de sécurité et la nécessité d'une ré-expertise indépendante.

Références

1. *Information Technology Security Evaluation Criteria* (ITSEC), version 1.2, Office for Official Publications of the European Communities, June 1991.
2. *Information Technology Security Evaluation Manual* (ITSEM), version 1.0, Office for Official Publications of the European Communities, Luxembourg, September 1993, ISBN 92-826-7087-2.
3. *Common Criteria for Information Technology Security Evaluation*, Version 2.1, Part 1 : Introduction and general model, CCIMB-99-031, August 1999.
4. *Common Criteria for Information Technology Security Evaluation*, Version 2.1, Part 2 : Security functional requirements, CCIMB-99-032, August 1999.
5. *Common Criteria for Information Technology Security Evaluation*, Version 2.1, Part 3 : Security assurance requirements, CCIMB-99-033, August 1999.
6. *Common Methodology for Information Technology Security Evaluation*, Version 1.0, Part 2 : Evaluation Methodology, CEM-99/045, August 1999.
7. *Schéma Français d'Évaluation et de Certification de la Sécurité des Technologies de l'Information*, ECF 12 - Programme de maintenance - version 1.0, Service Central de la Sécurité des Systèmes d'Information (SCSSI), Décembre 2000 (Ce document n'est plus applicable dans le cadre actuel du Schéma Français d'Évaluation et de Certification).
8. *Evaluation Methodology – Supplement : AMA - Assurance Maintenance*, Version 0.6 Draft, CCIMB-2001/0032R, December 2001.
9. *Common Methodology for Information Technology Security Evaluation*, Part 2 : Evaluation Methodology, Supplement : ALC_FLR - Flaw Remediation – Version 1.1, CCIMB-2001/0015R, February 2002.
10. *Surveillance des produits certifiés*, 000310/SGDN/DCSSI/SDR - SUR/P/01, Secrétariat général de la défense nationale/ Direction centrale de la sécurité des systèmes d'information, février 2004.
11. *Processus de qualification d'un produit de sécurité - niveau standard - version 1.0*, 001591/SGDN/DCSSI/SDR, Secrétariat général de la défense nationale/ Direction centrale de la sécurité des systèmes d'information, juillet 2003.
12. *Assurance Continuity : CCRA Requirements – Version 1.0*, CCIMB-2004-02-009, February 2004.