# C&ESAR 2015

**C**omputer & **E**lectronics
**S**ecurity **A**pplications
**R**endez-vous

Résilience des systèmes numériques

23-25 novembre 2015
Rennes - France

http://www.cesar-conference.org

## C&ESAR 2015 : Résilience des systèmes numériques

Pour sa 22ème édition, le thème des journées C&ESAR est la résilience des systèmes numériques.

La résilience est un concept assez récent qui a identifié le besoin de comprendre comment un système complexe fait face à des perturbations. Cette notion est apparue dans plusieurs domaines : écologie, test des matériaux, ingénierie des systèmes complexes, etc.

Dans le domaine de l'écologie, la résilience dénote la capacité d'un système à absorber une perturbation tout en préservant la même fonction. Dans le domaine du test des matériaux la résilience est la durée requise pour qu'un système revienne à un état d'équilibre. Dans le domaine de l'ingénierie des systèmes complexes, la résilience est définie comme la capacité d'un système à adapter son fonctionnement avant, pendant ou après des changements ou des perturbations de façon à ce qu'il puisse supporter les opérations requises dans des conditions aussi bien attendues qu'inattendues.

Dans le domaine des systèmes de défense, la résilience des systèmes numériques peut être définie comme la *capacité à fonctionner, éventuellement en mode dégradé, même en présence d'agressions.*

Traditionnellement, la résilience des systèmes industriels est assurée par une combinaison de techniques issues du domaine de la sûreté de fonctionnement comme la redondance des équipements, la dissimilarité des logiciels ou encore les analyses quantitatives d'indisponibilité d'un système.

Dans le domaine militaire, la résilience a pour objectif de permettre au système de remplir sa mission opérationnelle. Une analyse permanente des dysfonctionnements intégrant aussi bien les pannes, que les agressions doit être menée. Dans ce domaine, les solutions issues de la sûreté de fonctionnement ne suffisent pas forcément.

Actuellement, le recours systématique aux technologies de l'information dans les systèmes complexes oblige à réviser les architectures et les principes d'assurance pour prendre en compte les menaces informatiques. La combinaison des techniques issues des domaines de la sûreté de fonctionnement et de la sécurité informatique devient nécessaire.

On sait que les exigences de sûreté de fonctionnement et de sécurité peuvent être contradictoires : par exemple, introduire dans un système des fonctions de chiffrement apporte une protection en confidentialité voire en intégrité des données échangées, mais se traduit par une baisse de la disponibilité et de résilience. Néanmoins des études récentes sur des sujets tels que les hyperviseurs ou les processus d'assurance ont fait apparaître des

convergences fortes entre les domaines de la sûreté de fonctionnement et de la sécurité informatique.

Le programme de C&ESAR 2015 reprend les points énoncés dans cette introduction, en parcourant les axes suivants :

— État des lieux et concepts de la résilience : une première série de présentations traite de la définition de la résilience numérique et du positionnement des concepts proches de sécurité et sûreté.

— Challenges et Architectures : une seconde série d'articles traite des défis posés par l'émergence de nouvelles architectures dans deux domaines industriels : les télécommunications et la génération et distribution d'énergie.

— Solutions : la troisième série d'articles présente des solutions existantes pour assurer la résilience des systèmes numériques.

— Perspectives : la quatrième série d'articles présente des approches qui contribueront dans le futur à la résilience numérique.

Les présidents du comité d'organisation et du comité de programme tiennent à remercier chaleureusement tous les acteurs qui ont encore une fois rendu possible notre rendez-vous annuel : les conférenciers, les membres des deux comités, les organisateurs, et tous nos fidèles partenaires sans qui cette manifestation ne pourrait avoir lieu. En leur nom, nous vous souhaitons une excellente conférence.

Pierre BIEBER (ONERA), Président du comité de programme
Benoît MARTIN (DGA-MI), Président du comité d'organisation
Olivier HEEN (Technicolor), Directeur de publication

## Comité d'organisation

| | |
|---|---|
| José ARAUJO | ANSSI |
| Boris BALACHEFF | HP Labs |
| Ludovic PIETRE-CAMBACEDES | EDF |
| Yves CORREC | ARCSI |
| Frédéric CUPPENS | Télécom Bretagne |
| Hervé DEBAR | Télécom SudParis |
| Olivier HEEN | Technicolor |
| Éric JAEGER | DGSIC, min. défense |
| Ludovic MÉ | CentraleSupélec |
| Benoit MARTIN (président) | DGA-MI, min. Défense |
| Éric WIATROWSKI | Orange |

## Partenaires

DGA, DGSIC, ANSSI, ARCSI, CentraleSupélec, Télécom Evolution, EDF, HP, Airbus D&S, DCNS, Technicolor, Pôle d'Excellence Cyber, Orange Cyberdéfense, Chaire Cyberdéfense des Systèmes Navals.

**Site officiel :** `http://www.cesar-conference.org`

## Comité de programme

| | |
|---|---|
| José ARAUJO | ANSSI |
| Philippe AYRAULT | Thales |
| Boris BALACHEFF | HP Labs |
| Pierre BIEBER Président | ONERA |
| Jean-Paul BLANQUART | AIRBUS D&S |
| Patrice BOCK | Sogeti |
| Marc BOUISSOU | EDF |
| Jeremy BUISSON | CREC – IRISA |
| Jean CAIRE | RATP |
| Ludovic PIETRE-CAMBACEDES | EDF |
| Yves CORREC | ARCSI |
| Frédéric CUPPENS | Télécom Bretagne |
| Hervé DEBAR | Télécom SudParis |
| Véronique DELEBARRE | Saferiver |
| Ivan FONTARENSKY | AIRBUS D&S |
| Patrick HEBRARD | DCNS |
| Olivier HEEN | Technicolor |
| Éric JAEGER | DGSIC, min. défense |
| Mohamed KAANICHE | LAAS |
| Thierry KESSLER-RACHEL | EMA, min. Défense |
| Jean-Pierre LEBEE | DGA, min. Défense |
| Benoît MARTIN | DGA, min. Défense |
| Ludovic MÉ | CentraleSupélec |
| Benjamin. MONATE | Trust In Soft |
| François PESSAUX | ENSTA |
| Eric TOTEL | CentraleSupélec |
| Frédérique VALLEE | All4Tec |
| Éric WIATROWSKI | Orange |

## Table des matières

# Cyber Résilience

François-Régis Vigneau

Etat-major des Armées - Centre de Cyberprotection des Armées

**Résumé** La cyber-résilience est aujourd'hui un concept à la mode mais sans que ses limites soient clairement définies ni complètement partagées par ceux qui l'emploient. Etre en mesure de continuer son activité malgré une agression est avant tout une responsabilité de celui qui conduit la mission de l'organisme, la technique n'est qu'un des volets qui permettra d'atteindre l'objectif. Contrairement à une pratique répandue il faut en effet penser réalisation de la mission avant de penser solution technique et de se décharger de la résilince sur l'opérateur. La résilience fait appel aux contributions de l'ensemble des acteurs de l'organisme coordonnés par les opérationnels qui imposent les priorités et le tempo.

La cyber-résilience est aujourd'hui un concept à la mode mais sans que ses limites soient clairement définies ni complètement partagées par ceux qui l'emploient. Un bref retour aux sources, le dictionnaire, nous propose différentes approches. Le Larousse définit la résilience comme une *caractéristique mécanique définissant la résistance aux chocs d'un matériau.* Cette définition très mécanique ne nous éclaire pas vraiment sur son application au domaine des systèmes d'information si ce n'est par la notion de capacité de résistance du système à une agression extérieure. Une recherche dans la littérature anglo-saxonne, dans le *dictionary of contemporary English*, nous propose la définition suivante : *the ability to become strong, happy, or successful again after a difficult situation or event*, soit la capacité à retrouver sa force, son bonheur ou son succès après une épreuve ou une situation difficile. Appliquée au périmètre d'une entreprise ou d'une organisation nous pouvons nous interroger sur *la capacité à retrouver sa force son bonheur ou son succès.* La question revient à se demander ce qui est central pour elle. Toute organisation est mise en place dans un but précis. Son existence et sa survie y sont intimement liés. Ce but est ce que nous appellerons sa mission. D'autre part les notions de force et de succès renvoient à la disponibilité des moyens. L'objectif étant de réaliser la mission au coût minimal (financier, humain ...) pour l'organisation. Nous pouvons donc en déduire la définition de la résilience suivante : la capacité à poursuivre sa mission et à revenir à un état nominal, correspondant à un fonctionnement courant, après une agression. Cette agression pouvant être le résultat d'une action délibérée et ciblée ou d'aléa environnementaux (climatiques, techniques ...).

L'élément pivot de cette approche est la notion de mission et de capacité à la remplir. L'acteur central de la résilience devient donc celui qui porte la mission, qui est responsable de sa réalisation, à savoir le métier ou l'opérationnel. Nous retiendrons l'appellation « opérationnel » pour la suite de cet article.

L'étude présentée dans la suite est globale, elle s'adresse donc à priori à un organisme de taille importante qui dispose en interne de l'ensemble des ressources (un ou plusieurs opérateur(s) de ses systèmes d'information, direction des ressources humaines, service logistique ...). Néanmoins la logique du processus peut s'appliquer à n'importe quelle organisation, en prenant en compte les sous traitants, via le levier contractuel. Il est en effet nécessaire de s'assurer que les moyens ont été mis en place pour garantir que les sous traitants sont en mesure d'assurer une résilience acceptable et de mettre en place les processus nécessaires en cas d'agression de notre propre organisation.

La première étape pour clarifier la notion de résilience et partir sur des bases communes est donc de bien définir ce que recouvrent les notions de résilience et de cyber-résilience, ainsi que leur environnement (leur périmètre, leurs acteurs ...). Les notions étant partagées nous pourrons nous intéresser aux différentes phases du processus : la préparation puis la réponse et le retour d'expérience.

# 1 L'Environnement

## 1.1 Définition

Partant de la définition précédente la notion centrale est celle de mission. C'est donc à ce niveau que doit se situer la prise en compte de la résilience. L'acteur naturel devient donc l'entité ou l'organisme qui en est responsable. La mission est sa raison d'être, c'est elle qui sous-tend son organisation, ses processus et ses moyens. En effet la mission est déclinée en processus et sous processus qui sont appuyés par un certain nombre de ressources qui peuvent être humaines ou techniques. C'est à ce niveau que nous trouvons les systèmes d'information qui ne sont que des moyens au service de l'objectif final. C'est également à ce niveau que la notion de cyber résilience commence à prendre du sens, dans le contexte de la participation des systèmes d'information à la mission. En effet, les mesures appliquées aux systèmes d'information permettant d'augmenter leur résistance aux agressions ne sont que des moyens, parfois techniques mais pas nécessairement, d'atteindre l'objectif plus global de poursuite de l'activité opérationnelle. La cyber résilience n'est donc qu'un volet de la résilience qui vise à minimiser l'impact sur la mission d'une agression sur l'un des systèmes d'information de l'organisation. Son objectif est de prendre en amont l'ensemble des mesures techniques et organisationnelles qui permettront aux systèmes d'informations mis en oeuvre de continuer à délivrer les services pour lesquels ils ont été conçu. Lorsque l'agression est trop forte les mesures mises en places doivent permettre aux systèmes d'information de dégrader de manière progressive les services offerts. Enfin, après avoir subi l'attaque la cyber réilience doit permettre de rétablir dans des conditions acceptables les services initiaux. Sous ensemble de la résilience, s'inscrivant dans la continuité de la mission plus que du système d'information en tant que tel, elle est de la responsabilité de la chaine opérationnelle et ne doit en aucun cas être considérée comme une problématique purement technique qui serait donc laissée aux seules mains des services d'information et de communication dont la raison d'être est la mise à disposition d'un service et

donc focalisés sur le fonctionnement technique. De cette première approche il apparait donc que le terme résilience recouvre deux aspects : c'est d'abord une capacité opérationnelle, mais c'est également le processus qui la sous-tend. Il comporte deux grands volets : un volet temps réfléchi qui vise à mettre en place une posture organisationnelle et technique qui permet de résister, de ralentir et de détecter une aggression, et un volet temps court qui vise après la détection à caractériser l'aggression et à y faire face pour conserver ou restaurer sa capacité à réaliser la mission. Par rapport à l'approche traditionnelle de gestion des risques, notamment pour les systèmes critiques, c'est ce second volet qui est fondamentalement différent. Lorsque l'approche traditionnelle fait l'inventaire des risques, cherche à les minimiser puis accepte l'occurrence d'un certain nombre d'évènements ou en partage l'impact avec des tiers, le concept d résilience suppose une résistance active. Je pense que le changement de paradigme est lié à la prise en compte, avec la généralisation des systèmes d'information largements interconnectés d'une menace humaine, dotée d'une volonté de nuire, qui est par nature pensante et adaptative. Il est donc nécessairede dépasser la notion de sureté de fonctionnement. Les évènements redoutés que sont la catastrophe naturelle, la défaillance technique, le vol, la malveillance (mais qui nécessitait une présence physique sur le site) restent pertinents. Néanmoins, aujourd'hui, l'apparition d'une menace humaine plus diffuse au sein des réseaux, distante, qui s'est professionnalisée, et est également capable de s'adapter en temps réel aux contre mesures mises en place, avec une capacité de nuisance beaucoup plus importante puisque les systèmes d'information sont au centre de tout (fini le cloisonnement physique entre alimentation électrique, système incendie, systèmes de production...) nécessite de tenir une posture beaucoup plus réactive et adaptative qui s'apparente plus à une partie d'échec.

### 1.2 Les Différentes Phases

La résilience est un processus qui s'inscrit dans la durée. Il comporte différentes phases qui constituent dans une boucle vertueuse d'amélioration continue :

- La préparation qui nécessite l'élaboration d'une cartographie des moyens concourant à la mission, l'analyse de leur criticité afin de déterminer le périmètre minimal en deçà duquel la mission ne peut plus continuer. Ce périmètre sera dénommé le coeur critique. Il est également nécessaire de définir les modes dégradés acceptables et les processus associés. Enfin ces réflexions devront être documentées. C'est ce que nous expliciterons dans la prochaine partie.
- La phase de continuité qui comporte l'application d'actions reflexes prédéfinies dans les plans de continuité et de reprise d'activité, visant à minimiser les impacts sur la conduite de la mission et à reprendre le contrôle de la situation.
- La situation étant stabilisée et la mission se poursuivant en mode dégradé, arrive une phase d'arbitrage majeure entre plusieurs objectifs concurrents : restaurer les fonctionnalités au plus vite pour reprendre une activité normale, collecter des informations sur l'origine de l'agression pour

limiter autant que possible la probabilité de nouvelle occurrence, et poursuivre des objectifs prioritaires à court terme de la mission quitte à les réaliser en mode dégradé. C'est toute la problématique de la décision de lancer les opérations de retour à la normale.

— La suite naturelle est le séquencement continu ou non des différentes phases de retour au mode nominal en fonction des plans préétablis. Chaque situation étant unique les plans ne doivent pas être vus comme un cadre rigide mais identifier et mettre en lumière les étapes clés et les dépendances diverses. Néanmoins chaque opération nouvelle doit être soigneusement étudiée car elle se fera sous contrainte temporelle, alors que les plans ont été élaborés à froids et qu'il a été possible de prendre du recul. Ils permettent donc de dégager des ressources pour se concentrer sur la situation atypique qui n'avait pas été prévue.

— La capitalisation par le retour d'expérience ou retex qui permet de tirer les enseignements de l'incident et de faire évoluer si besoin les processus, les mesures techniques et organisationnelles. Cette phase peut dépasser le cadre de l'entité. En effet le principe de base du retour d'expérience, *la vie est trop courte pour pouvoir s'offrir le luxe de refaire les expériences des autres*, s'applique. Le partage des expériences au sein d'une communauté de confiance permet d'avancer plus rapidement en tirant profit des enseignement élaborés par d'autres sur des événements pour lesquels nous n'avons pas été impactés, ce qui ne signifie pas que nous n'aurions pas pu être touchés.

### 1.3 Acteurs

Au cours des différentes phases que nous venons de décrire succinctement un nombre non négligeable d'acteurs différents vont intervenir. La résilience est donc un processus qui mobilise l'ensemble des forces de l'organisme. A des degrés divers, en fonction des évènements et des différentes phases seront sollicités les opérationnels responsables de la mission, les opérateurs de systèmes d'informations chargés de la mise à disposition des services, les acteurs de cyberprotection, les centres de supervision et de sécurité des systèmes d'informations, le CERT noyau d'expertise et de synthèse de la réponse à un incident informatique, les gestionnaires de ressources humaines chargés du recrutement des spécialistes identifiés et/ou de leur formation, les acteurs de la protection défense, les spécialistes de l'infrastructure, de l'énergie, de la climatisation, de la restauration, des services juridiques, de la logistique, des achats ... Les acteurs centraux sont ceux qui portent la mission, donc les opérationnels. Ils sont les seuls à même de définir si les impacts sur l'activité sont acceptables ou non. Ils s'appuieront en fonction des phases sur d'autres intervenants et seront ceux qui définissent les priorités et le tempo des actions. Les opérateurs, les spécialistes de l'énergie seront en soutien pour maintenir ou rétablir les services en fonction des priorités qui leur auront été fixées.

Lors de la préparation les acteurs de la cyberprotection, les opérateurs, les services de ressources humaines, les services juridiques, d'infrastructure, d'énergie,

sous la coordination des opérationnels, seront sollicités pour décrire les différents processus, identifier les points faibles, définir les modes dégradés et les processus associés. Ils seront également sollicités, ainsi que les centres de supervision et de sécurité et le CERT pour valider que les modes de fonctionnement imaginés sont effectivement réalisables et pertinents.

Lors de la phase de continuité les opérationnels font appliquer les processus définis pour les modes dégradés afin de garantir la réalisation de la mission dans les meilleures conditions. Le CERT et les centres de supervision et de sécurité et les opérateurs sont les interlocuteurs privilégiés. Il s'agit en effet avant tout de détecter (rôle de la supervision), de caractériser l'agression et de protéger le coeur critique. Les services juridiques peuvent également être sollicités. Lors d'une attaque délibérée que ce soit sur les systèmes d'information ou lors d'une agression physique, il y a un arbitrage permanent entre la volonté de repousser au plus tôt l'agresseur pour minimiser son impact sur le système et revenir à un régime nominal rapidement et le laisser se dévoiler en lui laissant une certaine marge de manoeuvre pour recueillir le plus de renseignements possible sur son identité, ses moyens, ses objectifs en lui laissant croire qu'il est non détecté. De cette seconde approche nous pourrons retirer de nombreux enseignements qui nous permettrons de mieux lui interdire l'accès à notre périmètre dans le futur. La ligne rouge ultime est sans conteste lorsque l'agresseur s'approche du coeur critique. Il existe un risque de perdre l'avantage que nous pensions avoir gagné. C'est pourquoi cet arbitrage, qui repose en grande partie sur les éléments fournis par les centres de supervision et de sécurité, le CERT et les opérateurs est une prérogative opérationnelle exclusive. Les autres acteurs (les services administratifs, les ressources humaines, les achats, la logistique...) appliquent les processus prévus pour les différents modes dégradés en fonction de l'évolution de la situation.

La décision de retour à la normale est également déterminée par un autre arbitrage entre les impacts liés aux opérations de restauration de service qui peuvent nécessiter des coupures ponctuelles de services, y compris du coeur critique, qui comportent intrinsèquement un risque de régression aussi borné soit-il, et la conduite d'opérations prioritaires liées à la mission qui ne sauraient être reportées et ne peuvent souffrir aucune dégradation même ponctuelle de la situation. Le choix de privilégier une augmentation des services disponibles à moyen terme ou la réalisation d'opérations à court terme est clairement une prérogative opérationnelle exclusive. Cette décision s'appuie notamment sur les éléments fournis par les centres de sécurité et les opérateurs (notamment durée de chaque étape, services restaurés...) mais surtout sur le tempo des opérations en cours et des priorités relatives des objectifs opérationnels à court et moyen terme ainsi que des moyens nécessaires à leur réalisation.

Enfin la capitalisation fait intervenir l'ensemble des acteurs toujours avec une coordination opérationnelle. Une contribution importante est celle des acteurs de la cyberprotection. Ils sont en effet responsables de faire évoluer si besoin la posture permanente de protection et en particulier le cadre réglementaire, ou les mesures techniques et organisationnelles.

Le cadre général de la résilience étant ainsi défini, il convient de détailler plus avant les différentes phases de ce processus.

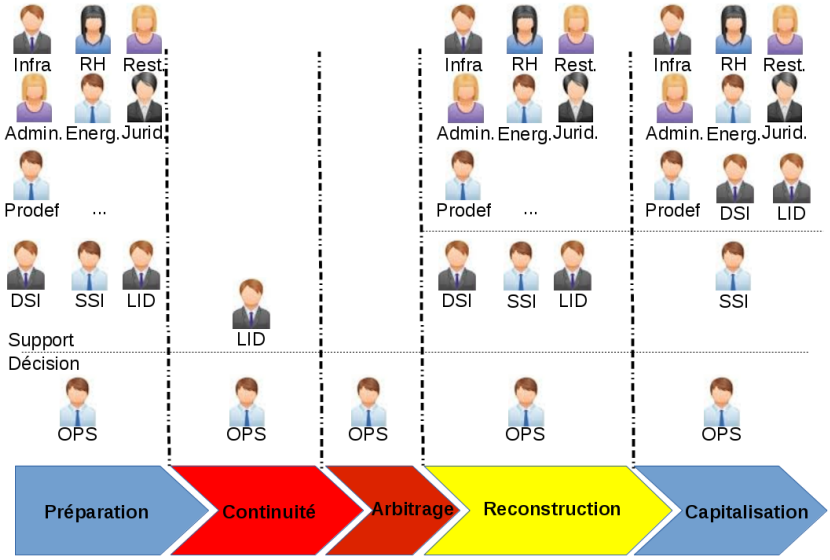



**Figure 1.** Les différentes phases et les principaux acteurs associés

## 2 Préparation

La première étape consiste à préparer les plans qui seront appliqués en cas d'incident. Elle peut paraître chronophage et complexe mais est néanmoins primordiale. Elle permet de réfléchir aux éléments primordiaux pour la mission qui devront être particulièrement protégés. Il est en effet illusoire et générateur de coûts exorbitants de chercher à garantir une forte disponibilité sur l'ensemble des systèmes, une intégrité parfaite de tous les processus et de toutes les données ainsi que la confidentialité absolue de l'ensemble des données et traitements. Il faut donc s'attacher à décrire précisément l'environnement de la mission et de réfléchir aux meilleures options pour pallier aux différents incidents ou accidents qui auront pu être identifiés.

### 2.1 Cartographie

Avant toute chose il est nécessaire de réaliser une cartographie de l'environnement de la mission. Il s'agit d'identifier l'ensemble des processus nécessaires

à la mission ainsi que les moyens et les données qui y sont rattachés. Il faut en particulier identifier les systèmes d'information et les données qui concourent à l'activité. Cet inventaire doit être réalisé en sollicitant les contributions de l'ensemble des services de l'organisme de manière à ne pas négliger de processus supports qui pourraient se révéler critiques. Par exemple si le personnel ne peut plus être payé ou nourri, il est fort probable que l'activité s'en trouve rapidement impactée. Les systèmes d'information et les données identifiés, il faut préciser la description en listant les systèmes et données sous-jacents nécessaires à leur fonctionnement ou traitement. Il est notamment important de bien prendre en compte les systèmes de soutien et d'environnement, trop souvent négligés comme la climatisation, l'alimentation électrique, le soutien des ressources humaines ...

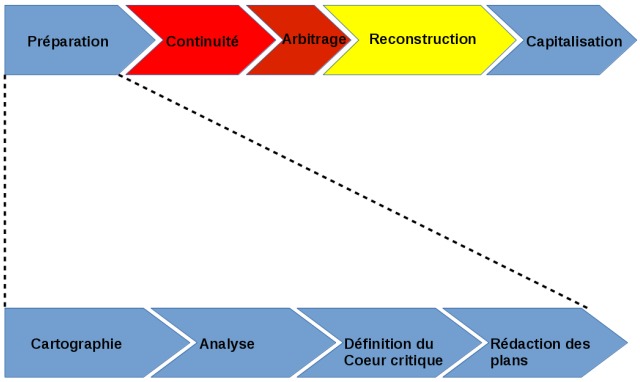



**Figure 2.** Les différentes étapes de la préparation

## 2.2 Analyse

La cartographie étant réalisée il est nécessaire de prioriser les différents éléments ainsi identifiés, c'est le rôle de la phase d'analyse. Elle consiste initialement à répondre à un certain nombre de questions dans le but de préciser notre dépendance vis à vis de tel ou tel système ou donnée. La première information qui conditionne largement notre marge d'action est le contrôle réel que nous pouvons avoir sur le système : quelles sont nos capacités d'action sur le système ? Cela consiste à lister les différents acteurs du périmètre et les verrous ou les leviers existants. Par exemple, quel est l'opérateur, qui est en charge du maintien en condition opérationnelle, en conditions de sécurité, sont-ils internes ou externalisés (industriel), le système est-il soumis à une réglementation particulière qui nous impose des contraintes particulières (enregistrement de certaines activités,

traitements particuliers lors des maintenances ...) ? Dans le cas d'une externalisation de l'opérateur ou du maintien en conditions opérationnelles, en conditions de sécurité, il convient d'identifier les dispositions contractuelles comme les délais d'intervention (en et hors heures ouvrables), les périmètres d'intervention de chaque intervenant, les limites éventuelles susceptibles de remettre en cause les garanties ou les contrats de maintien en conditions opérationnelles, en conditions de sécurité, les limitations sur le nombre ou le volume financier des actions possibles, les différentes prestations prévues ... Il faut également vérifier les possibilités et les délais d'accès pour les intervenants extérieurs (accès hors heures ouvrables, durée des préavis ...).

L'environnement du système étant ainsi précisé il faut définir le temps d'indisponibilité maximum supportable, le niveau d'intégrité nécessaire ainsi que les besoins en confidentailité dans le cadre de la mission. Ces données sont très dimensionnantes en termes de finances et de ressources humaines. Il est donc primordial de confronter ce résultat avec les éléments déterminés précédemment relatifs aux conditions de maintien en conditions opérationnelles et de sécurité et de revoir le cas échéant soit l'ambition soit les contrats. Cette étude doit également permettre de déterminer quels seraient les impacts en disponibilité, en intégrité et en confidentialité d'une compromission des données ou du système sur la conduite de l'activité.

Il est également nécessaire de s'interroger sur les sources de menace à prendre en compte. Il faut donc référencer l'ensemble des sources de menaces pesant sur la mission de l'organisation. Elles peuvent être de diverses origines :

- naturelles : inondation, raz de marée, tsunami, tremblement de terre, glissement de terrain, orages...
- technologiques : accident industriel, crash d'un avion, accident routier, défaillance de matériels, incendie ...
- humaines : maladresse, négligence, malveillance, espionnage ...

Pour chaque source de menace ainsi identifiée il est nécessaire de définir son niveau attendu compte tenu de l'organisation et de sa localisation : hauteur d'eau pour une inondation, force pour un tremblement de terre, niveau pour une menace humaine (qui couvre à la fois un niveau de technicité, de moyens financiers et un niveau de motivation). Pour une menace humaine il est également possible de synthétiser ces trois données en qualifiant le type d'acteur (organisation gouvernementale, terroriste, crime organisé, individu avec un niveau technique fort, faible ou expert, avec une motivation financière, idéologique, de revanche, ludique ...). Disposant d'une cartographie précise du système et des sources de menaces qui pèsent sur lui, il devient possible d'imaginer des scénarios d'aggression ainsi que leur vraisemblance et l'impact qui pourrait en résulter pour l'organisation et sa mission[1]. Cette étude permet de déterminer les évènements redoutés contre lesquels il est nécessaire de se protéger, ainsi que le niveau de ressources à y consacrer.

1. Cet impact est souvent qualifié de brut car il est avant l'application de toute mesure visant à le limiter

Les évènements redoutés étant maintenant identifiés ainsi que les impacts bruts associés il est devient possible de s'interroger sur les mesures à même de limiter les impacts ou la probabilité d'occurence de l'évènement redouté. Il faut également identifier les modes dégradés qu'il serait possible de mettre en place en cas de perte ou de compromission d'un ou plusieurs sous-systèmes afin de poursuivre la mission. Une fois identifiés, il est nécessaire d'évaluer leur durée maximale d'utilisation, ainsi que les contraintes qu'ils génèrent, à court et moyen termes sur la conduite des opérations. L'ensemble des modes dégradés possibles ayant ainsi été identifiés il faut déterminer ceux qui sont adaptés ou acceptables. Une solution satisfaisante à court terme peut se révéler pénalisante à moyen ou long terme en raison de la charge de travail générée lors du retour à l'état nominal (par exemple compte tenu des ressaisies engendrées, du volume de données à restaurer dans une fenêtre temporelle réduite). De l'existence ou non de modes dégradés acceptables découle la détermination des points de fragilité unique. Cela revient à identifier les éléments qu'il n'est pas possible de perdre, que ce soit en terme de disponibilité, d'intégrité ou de confidntialité sans mettre en péril la mission. Ils peuvent être techniques (alimentation électrique, matériels couteux ou rares pour lesquels la mise en place de lots de rechange sera très difficile, matériels obsolescents ...) ou organisationnels (compétences rares, point de décision unique, locaux, données vitales pour l'organisation, confiance dans l'intégrité d'un processus ou d'une donnée ... ). Pour obtenir la vision la plus pertinente possible il est nécessaire d'interroger tous les métiers concourants à la mission, chacun ayant généralement sa propre vision des impacts, de leur importance, de leur acceptabilité. Typiquement un évènement peut être jugé mineur par les opérationnels au regard de leur processus métier mais impacter très fortement un processus de soutien remettant en cause la mission la réalisation de la mission (par exemple la perte du lien Internet négligeable pour les opérationnels parce que le système métier est sur un intranet dédié mais nécessaire au service achat et donc au maintien en conditions opérationnelles et de sécurité).

### 2.3 Définition du Coeur Critique

La finalité de cette analyse est de définir la criticité de chaque sous-système ou donnée identifié au regard de la mission en fonction de l'existence de modes dégradés possibles, des temps d'indisponibilités maximum affichés, et de l'importance des impacts recensés. Ces criticités étant déterminées il faut imaginer différents modes de fonctionnement, en situation plus ou moins dégradée qui permettent de continuer à assurer la mission. Il est donc nécessaire d'identifier plusieurs périmètres (comportant des données, des systèmes, des ressources humaines ...) correspondants à des capacités de réalisation de la mission en situation plus ou moins dégradée. Le périmètre le plus restreint, en dessous duquel l'activité n'est plus possible, correspond au mode dégradé ultime admissible et forme le coeur critique ou le centre de gravité (cf. figure 3). S'il est impacté la mission ne peut plus continuer, tous les efforts doivent donc se concentrer sur sa protection. Le périmètre le plus large correspond au mode nominal lorsque tous les moyens sont disponibles.

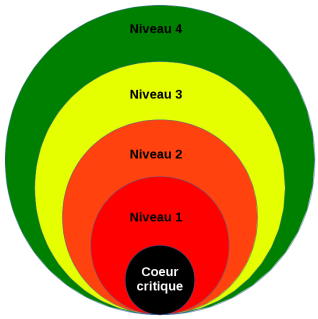



**Figure 3.** Définition des différents périmètres

Pour chaque périmètre il faut alors décrire les processus mis en place, les contraintes résultantes sur l'activité, les délais maximum pendant lesquels la mission pourra se réaliser sans amélioration de la situation.

### 2.4 Rédaction des plans

Arrivé à ce stade, cette étude a déjà nécessité un travail conséquent qu'il est primordial de capitaliser. C'est la phase de synthèse et de validation qui peut commencer. Elle se concrétisera par la fourniture des livrables que sont les plans de continuité et de reprise d'activité. Ces derniers pourront également être déclinés en plan de continuité et de reprise informatique pour leurs aspects techniques concernant plus particulièrement les systèmes d'information.

La première étape est de déterminer la liste des systèmes et des données qui représentent les actifs primordiaux de l'organisation ainsi que les actifs supports retenus. Les actifs primordiaux sont les éléments fonctionnels et organisationnels qui sont absolument nécessaires à la bonne réalisation de la mission. Ce sont donc les processus et les données manipulées qui concourent directement à la mission et dont une atteinte à la disponibilité, l'intégrité ou la confidentialité remet fortement en cause sa réalisation. Ils sont à rechercher en priorité dans les éléments du coeur critique. Les actifs supports sont ceux qui servent les actifs primordiaux. Ainsi nous y trouvons les matériels, logiciels, réseaux, personnes, sites, organisations, etc ...

La liste des actifs étant ainsi arrêtée il faut s'appuyer sur la cartographie et l'analyse réalisée précédemment de manière à les décrire le plus précisément possible. En particulier il est important de bien expliciter les différentes dépendances des actifs avec les différents sous-systèmes (matériels, protocoles, organismes extérieurs à l'unité ...). Chaque élément doit être décrit en termes de périmètre, de criticité, de durée maximale d'indisponibilité acceptable (RTO : Recovery Time Objective), de durée maximale d'enregistrement des données qu'il est admissible de perdre (RPO : Recovery Point Objective) ...

Les processus de fonctionnement dans chacun des modes dégradés retenus doivent être décrits avec précision ainsi que les limitations et impacts qui y sont associés à court et plus long terme. La formalisation de l'ensemble de ces éléments constitue les plans de continuité et de reprise d'activité. Les plans de continuité et de reprise informatique devront être déclinés de ces plans en fonction des priorités qu'ils ont déterminé. Ces plans étant écrits ils n'en constituent pas moins une oeuvre théorique qu'il va être nécessaire de valider et de confronter à la réalité. C'est l'objectif de la phase de validation.

Avant toute chose il s'agit de procéder à une validation formelle. Elle fait intervenir différents acteurs. L'opérateur sera sollicité pour confirmer sa capacité à mettre en oeuvre les solutions techniques préconisées ainsi qu'à tenir les délais associés. Les autorités d'emploi des différents systèmes d'information, ainsi que leurs responsables en sécurité de systèmes d'information seront également approchés afin de s'assurer que les procédures locales envisagées n'impactent pas de manière inattendue le fonctionnement global du système. Par exemple un composant du système en local peut apparaitre non critique mais être un élément important pour une autre unité assurant une autre mission et s'appuyant sur le même système. Enfin la validation finale est de la responsabilité de l'organisme, au plus haut niveau, afin de s'assurer que les processus retenus sont bien réalisables et permettent de réaliser sa mission dans une démarche de gestion du risque assumée. Il faut notamment s'assurer que les contraintes générées à court et plus long termes sont acceptables.

Cette phase de validation étant terminée, il reste à confronter ces plans à la réalité du terrain. C'est l'objectif des tests qui seront réalisés pour confirmer la pertinence de chacun des processus définis. La validation initiale effectuée il faudra ensuite jouer ces plans régulièrement au cours d'exercices afin de confirmer leur validité, leur compatibilité avec les RTO et RPO retenus, la capacité à poursuivre la mission et d'entraîner les personnels à leur mise en oeuvre.

# 3 Réaction et Retex

## 3.1 Continuité

Lors de la détection d'un incident, la phase de continuité est déclenchée, sa conduite est dévolue aux opérationnels, seuls à même de réaliser les arbitrages entre conduite de la mission, sécurisation du système d'information. Les acteurs des chaines cyberdefense (centres techniques de lutte informatique défensive, le CERT et son réseau, les SOC des opérateurs ...), cyberprotection, les opérateurs interviennent en appui pour le conseil et la réalisation d'actions techniques de leur domaine. Ces acteurs doivent agir de concert avec d'autres intervenants comme les experts juridiques, les responsables de l'énergie, de l'infrastructure ... Outre une première ligne d'opérations qui consiste à garantir la continuité de la mission à court terme, une seconde ligne concerne la collecte d'informations sur l'attaquant dans le cas d'une attaque ciblée pour l'identifier, déterminer ses modes d'actions et ses outils, ce qui permettra de mettre en place les contremesures à même de limiter sa capacité d'action à moyen terme : recueil

des indicateurs de compromission (IOC) pour établir les différentes signatures à destination des outils de sécurité et le priver de l'utilisation future des outils développés, détermination de l'étendue de son intrusion sur le système... En effet mettre en place une attaque ciblée (renseignement, développement d'outils spécifiques, intrusions initiale, découverte du système d'information, ciblage des ressources intéressantes ...) nécessite un investissement conséquent (en temps mais éventuellement également financier). Lorsque ces éléments sont connus ils ne sont plus réutilisables en l'état. L'objectif est donc, outre de lui interdire de futur accès aux systèmes avec les mêmes outils, de minimiser les possibilités de retour sur investissement pour l'attaquant. Il est donc nécessaire d'arbitrer entre l'éradication rapide de la menace et la collecte d'informations qui permet d'affaiblir l'attaquant mais nécessite du temps et de laisser se dérouler les évènements. La ligne rouge est l'approche du coeur critique par l'attaquant. Ces interactions entre de multiples acteurs doivent donc être coordonnées sous la conduite des opérationnels responsables de la réalisation de la mission. Cette situation plaide pour une intégration rapide au cycle de planification et de conduite des opérations.

### 3.2 Retour à la Normale

La situation étant stabilisée, l'initiative étant revenue à la défense, le début des opérations de retour à la normale est du ressort exclusif des opérationnels en fonction du déroulement des opérations en cours et des impératifs de la mission. Il peut en effet être plus préjudiciable d'interrompre une action en cours pour garantir un retour au mode nominal que de la poursuivre en mode dégradé. Le compromis risques/bénéfices doit être systématiquement évalué. Une matrice bénéfice/risque spécifiant à quel niveau hiérarchique la décision doit être prise peut être définie lors de la préparation. Les opérations de retour à la normale, décomposées en différentes phases, peuvent nécessiter du temps. Lorsque la décision est prise les opérations planifiées peuvent se dérouler sous la conduite de la chaine cyberdéfense, les opérationnels validant systématiquement les phases critiques ayant un impact potentiellement négatif sur les opérations en cours (arrêt de services, remise en ligne de service ...), ainsi que les changements de phase. Le rôle des opérateurs dans cette phase est bien entendu primordial.

### 3.3 Retex et Boucle Qualité

Enfin la dernière étape, après le retour à la normale, est l'activation du processus de retour d'expérience ou retex, destiné à dégager un maximum d'enseignements de l'incident de manière à s'en prémunir au mieux dans l'avenir. Sa finalité n'est en aucun cas de rechercher des coupables, ce qui serait une responsabilité de la chaine hiérarchique, mais bien d'identifier les causes et les problèmes associés, pour améliorer l'organisation en place que ce soit sur le plan des facteurs humains, de la technique, de l'organisation ... A cet égard un exemple intéressant est celui développé par le domaine aéronautique dans le domaine de la sécurité des vols qui favorise la remontée d'information sur toutes les situations atypiques

en introduisant la notion de dépénalisation de l'erreur. Le retex se décompose en deux étapes : le retex à chaud qui vise à prendre les mesures immédiates qui s'imposent, et un volet à froid, après analyse et exploitation de l'ensemble des éléments, qui doit s'affranchir du contexte émotionnel de l'évènement. Le Retex ne vise pas à rechercher les *coupables* mais à reboucler un processus qualité d'amélioration continue qui peut être illustré par la roue de Deeming (cf figure 4 :

**Plan :** c'est l'établissement et la validation des plans de continuité et de reprise d'activité ;

**Do :** consiste à mettre en place les solutions techniques et organisationnelles prévues dans les plans ;

**Check :** vérification de la pertinence des solutions retenues lors de tests réguliers, d'exercices ou de traitements réels d'incidents. Cette vérification doit être également coordonnée avec une veille de l'état de l'art afin de détecter l'apparition de vulnérabilités nouvelles ou l'obsolescence de certaines solutions retenues ;

**Act :** vise à corriger les écarts détectés en reprenant les plans et en faisant évoluer la cartographie, les contraintes et leur acceptabilité ou les solutions retenues.

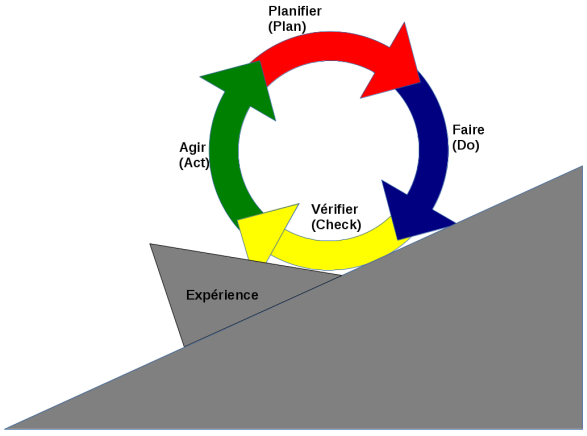



**Figure 4.** La roue de Deeming

Le volet « check » est notamment alimenté par les exercices dont l'objectif, outre l'entrainement du personnel, est également de détecter les évolutions de l'environnement (modification ou évolution technique, obsolescence de moyens palliatifs envisagés, obsolescence des matériels entrainant une indisponibilité des rechanges ...) susceptibles de remettre en cause les solutions retenues. D'une manière plus générale, le suivi des processus d'homologation pour les systèmes

d'information fournit des jalons pour le cycle de vie des plans de continuité et de reprise d'activité et donc pour l'élaboration et le suivi des plans de reprise et de continuité informatique afférents et leur inscription dans ce cercle vertueux de qualité.

Chaque système d'information est réalisé en fonction d'un objectif qui représente sa mission propre. Une fois déployé cette mission du système participera à lamission de l'organisme. Lors de la phase programme d'un système d'information il est nécessaire au plus tôt d'identifier les objectifs de sécurité du système au regard des contraintes de la mission qu'il supporte. Il est donc nécessaire de définir les besoins du système en termes de disponibilité, d'intégrité des données et des services et de confidentialité. Ces éléments sont clés pour définir la résilience au juste besoin. S'ils sont surspécifiés ils entraineront des coûts importants qui pourront préjudiciables à la survie de l'organisation, allant donc à l'opposé de la capacité de résilience recherchée. Inversement s'ils sont sous-évalués l'organisation ne sera pas en mesure d'affronter efficacement les agressions lorsqu'elles se présenteront, mettant en danger la pérénité de l'organisation qui ne pourra pas afficher une capacité de résilience suffisante. Les besoins de sécurité ainsi exprimés préfigurent donc le niveau de résilience à atteindre. Il est ensuite nécessaire de réaliser la cartographie fonctionnelle (quelles sont les processus supportés et avec quelles fonctionnalités). La prise en compte au plus tôt de ce besoin de résilience permet d'influer sur les choix techniques très en amont qui garantiront le niveau de résilience recherché. Il est ainsi possible de faire des choix en terme d'architecture (redondance, chiffrement, cloisonnement, maillage, répartition géographique...) ou de technologie (authentification forte, chiffrement de surface, sauvegarde/restauration de données...) qui ont un impact financier acceptable et une cohérence globale permettant d'obtenir les résultats attendus. En revanche lorsque ces problématiques sont abordées en fin de programme les choix dimensionnants d'architecture et de technologie sont déjà figés ce qui limite de manière importante les options possibles pour atteindre le niveau de résilience défini et implique des surcoûts importants qui sont généralement auxquels il est généralement très difficile de faire face. La conséquence est généralement un système dont la cohérence n'est pas optimale et pour lequel l'efficacité des mesures de résilience mise en place n'est pas à la hauteur des investissements consentis, ou encore pour lequel il est nécessaire de revoir les objectifs de résilience à la baisse faisant planer un risque sur la mission. Il est alors possible de réaliser la cartographie technique (quelles sont les composants matériels et logiciels, sur quels services tiers le SI s'appuie t'il ?) du système. Cette étape est nécessaire pour la définiton du maintien en conditions opérationnelles et du maintien en conditions de sécurité du système. Elle est egalement pariculièrement importante pour les organismes utilisateurs, porteurs de leur mission propre, lorsqu'ils devront préparer leur cartoraphie.

La phase programme permet également de définir certaines responsabilités comme l'autorité d'emploi, le responsable SSI qui seront les correspondants naturels pour ce système des responsables de l'unité lorsqu'ils auront à réaliser la cartographie et l'analyse de leurs systèmes. Les objectifs étant définis et le

système étant décrit il convient ensuite de déterminer les points de faiblesse éventuels au regard des objectifs de sécurité retenus. Combinées à une probabilité d'occurrence ces vulnérabilités permettent de déterminer les risques pesant sur la capacité du système à soutenir sa mission.

La phase suivante vise à limiter ces risques par la définition de mesures techniques et/ou organisationnelles acceptables, à les accepter en l'état, à les refuser ou à en partager la responsabilité (typiquement assurance ou sous-traitance) en fonction du contexte physico-financier du moment. En fonction de ce qu'il a été possible de définir sont déduits les risques résiduels qui seront soumis à la validation. Les conclusions de ce processus sont tracées dans le dossier d'acceptation des vulnérabilités résiduelles. Cette étude s'intègre parfaitement au processus de cartographie et d'analyse décrit précédemment pour l'élaboration des PCA-PRA. Le résultat de ces études est présenté à l'autorité d'emploi, responsable de la mission, pour validation. L'autorisation d'emploi qui est alors décrétée pour le système générique valide la pertinence des solutions retenues au regard de l'état de l'art, des objectifs de sécurité retenus par les opérationnels, et des contraintes physico-financières. Lors de chaque déploiement une autorisation d'emploi sera alors recherchée. Elle visera à garantir que ce déploiement ne remet pas en cause la sécurité globale du système d'information, mais également qu'il est compatible avec les besoins de sécurité exprimés en regard de sa participation à la mission de l'organisme.

Suite à un incident ou un exercice si les conclusions du PCA-PRA sont amenées à évoluer il est probable que les risques résiduels d'au moins un des systèmes d'information support soient amenés à évoluer. D'autre part les décisions d'homologation d'un système ayant une durée déterminée, elles sont appelées à être renouvelées périodiquement. Ce renouvellement est un moment privilégié pour revoir les risques résiduels, ce qui implique de revoir les PCA et PRA en cas d'évolution. Le processus d'homologation de sécurité s'inscrit donc parfaitement dans le cycle qualité de la vie des plans de continuité et de reprise d'activité.

## 4 Conclusion

La résilience est une problématique opérationnelle avant d'être une problématique technique. Elle doit être pensée au niveau de la mission. C'est donc l'emploi qui donne le cadencement et les priorités des actions. Compte tenu de la contrainte temporelle qui pèse lors d'un incident il est crucial de bien s'y préparer plutôt que d'improviser le jour J. C'est le propre de la conduite de tous les systèmes critiques. La ressource cognitive étant par essence comptée pour chacun, vu la volatilité de ces situations il importe de les avoir préparées par la rédaction de plan de continuité et de reprise d'activité réalistes et coordonnés par l'emploi pour pouvoir dégager les ressources nécessaires à l'adaptation face à un développement inattendu de la situation. Cette réalité impose que les opérationnels se réapproprient ce volet de leur activité trop souvent considéré comme une problématique purement technique.

# Sûreté et sécurité : différences et complémentarités

Jean-René Ruault*, Christophe Kolski*, Frédéric Vanderhaegen*,
Dominique Luzeaux**

* LAMIH-UMR CNRS 8201, Université de Valenciennes et du Hainaut-Cambrésis, Le Mont Houy, 59313 Valenciennes CEDEX 9
{prenom.com}@univ-valenciennes.fr
** rattaché à la Chaire Ingénierie des Systèmes Complexes de l'Ecole Polytechnique,
dominique.luzeaux@polytechnique.org

**Abstract.** Le monde actuel voit les technologies de l'information prendre une place croissante. La sécurité élargit les menaces affectant la sûreté des systèmes critiques mettant en œuvre ces technologies de l'information. Dans la perspective de la résilience des systèmes, qui est leur capacité à faire face à des événements imprévisibles, sans précédent, les enjeux de l'architecture système consistent à tisser les aspects relevant de la sûreté et ceux de la sécurité en construisant sur leur complémentarité. Nous proposons un modèle d'architecture orientée service intégrant la sécurité et la sûreté.

**Keywords:** Résilience, sécurité, sûreté, architecture système

## 1 Introduction

Les quinze dernières années ont vu le domaine de la sécurité des systèmes d'information évoluer radicalement. L'interconnexion puis la numérisation de la société et de l'ensemble des secteurs de l'économie, *via* notamment l'intégration des technologies de l'information potentiellement à l'ensemble des produits manufacturés, ont (re)placé l'information, davantage que le système d'information, au cœur de notre société. Même si le mouvement avait été amorcé au cours des années 90, la question centrale aujourd'hui est donc la sécurité de l'information comme le décrivent les normes ISO 27xxx (11, 12). La sensibilité à ces questions a été renforcée par la publication répétée d'attaques informatiques de toutes natures.

La résilience devient donc un enjeu tant économique qu'opérationnel, et les notions habituelles autour de la sécurité ne suffisent plus à la garantir. Dans cet article, nous nous intéresserons donc à la notion de sûreté, habituellement employée dans d'autres contextes, et nous décrirons l'apport mutuel de ces différentes notions, en les articulant au sein d'une architecture de contrôle de haut niveau par rapport aux architectures organique, fonctionnelle et physique du système.

Après avoir présenté la sécurité (protection des systèmes vis-à-vis d'attaques malveillantes) et la sûreté (protection des systèmes vis-à-vis du danger) (18), nous analysons leur complémentarité ainsi que leurs impacts sur l'architecture du système.

Nous proposons un modèle s'appuyant sur l'architecture orientée service et intégrant la sécurité et la sûreté.

La conclusion met en évidence des perspectives d'approfondissement, en particulier pour concevoir une architecture cohérente et qui soit évolutive.

## 2 Sécurité et sûreté : une nécessaire complémentarité

Sûreté et sécurité sont des termes employés dans de nombreuses disciplines, souvent avec des significations différentes, voire diamétralement opposées, d'une discipline à une autre (18). Dans ce contexte, nous reprenons à notre compte les définitions de sûreté et de sécurité (18), respectivement protection contre des accidents et protection contre des comportements malveillants.

### 2.1 La sécurité

Issue du chiffre[1], c'est-à-dire de l'échange d'information sécurisé dans le domaine militaire, la sécurité des systèmes d'information (SSI) est la démarche qui consiste à obtenir une confiance jugée suffisante dans la capacité d'un système d'information à respecter ses critères de sécurité face à des menaces intentionnelles.

Les critères formulés dans une expression de besoin de sécurité sont relatifs aux propriétés suivantes :

- Confidentialité : propriété d'un système d'information qui interdit l'accès à une information à quiconque n'est pas autorisé à en prendre connaissance.
- Intégrité : propriété d'un système d'information qui interdit qu'une information ou que le traitement d'une information soit indûment modifié.
- Disponibilité : propriété d'un système d'information qui permet qu'une information ou un traitement soit toujours accessible à quiconque est autorisé.

A ces critères de base sont parfois ajoutés d'autres critères comme par exemple ;

- Non répudiation : propriété d'un système d'information qui rend impossible à un utilisateur de nier avoir lu, modifié ou transmis une information.

La sécurité des systèmes d'information s'attache à faire respecter ces critères face à des attaques. La SSI n'est pas une performance obtenue au moment de l'acquisition du système et qui restera pérenne pendant toute son existence. C'est une démarche continue mise en œuvre tout au long du cycle de vie du système de l'expression de besoin au démantèlement. La sécurité d'un système d'information ne se prouve pas. Il s'agit d'acquérir un niveau de confiance jugé satisfaisant dans la capacité du système à résister aux attaques à faire respecter les critères de sécurité demandés. Ce niveau de confiance est atteint par le choix judicieux des intervenants, par une architecture accompagnée de certains produits correctement administrés, par une infrastructure et

[1] Service d'une armée ou d'une ambassade chargé de coder les messages secrets afin qu'ils ne puissent pas être dévoilés.

une organisation adaptée, par une couverture de test, une évaluation, un audit, une maintenance, …

Elargissant le périmètre de la SSI, perçue comme statique, la cybersécurité, perçue comme plus dynamique, peut être définie comme étant un ensemble des moyens organisationnels, techniques et humains assurant la confidentialité, l'intégrité et la disponibilité de l'information. La cybersécurité va au-delà du système d'information lui-même (11).

En conformité à ces normes, entre autres, des méthodes ont été élaborées et sont mises en œuvre pour assurer la sécurité des systèmes d'information. Ainsi, la méthode EBIOS, pour « expression des besoins et identification des objectifs de sécurité » comprend des éléments de gestion des risques (3) et des propositions d'outillage adaptées (2). La fiche d'expression rationnelle des objectifs de sécurité (FEROS), quant à elle, est une méthode d'évaluation des risques adaptée à la sécurité (23). Ces risques se déclinent en termes d'espionnage, de sabotage, d'écoute, d'accès illégitime, d'abus de droit et s'appuient sur des dispositifs matériels et logiciels que sont, entre autres, les portes dérobées, les chevaux de Troie, les vers, les logiciels espions. Lutter contre ces menaces consiste à détecter les intrusions et à générer des alertes (10). Ces enjeux, initialement prégnants dans le domaine militaire, concernent dorénavant tout autant le domaine civil, à la hauteur des enjeux économiques, sociaux, politiques, sociétaux, des échanges d'information qu'autorisent les technologies de l'information.

La résilience du système, au sens de la sécurité du système d'information – ce dernier étant considéré sous son acception système large, y compris sa composante qu'est la connexion –, se décline en des plans de continuité d'activité (PCA) et de reprise d'activité (PRA), ou leur déclinaison aux spécificités du système considéré (7, 8).

Parmi les impacts possibles d'une attaque – qu'elle soit de type déni de service ou piratage avec usurpation d'identité ou de droit –, les dysfonctionnements induits au niveau des infrastructures critiques peuvent générer des accidents catastrophiques. En effet, les systèmes actuels ayant de plus en plus une composante informatique, à la fois capteurs et actionneurs (télémaintenance), la sécurité voit son périmètre et son domaine d'application évoluer, débordant le domaine d'application d'origine pour s'approcher de ceux de la sûreté, c'est-à-dire la protection vis-à-vis du danger.

### 2.2 La sûreté

La sûreté est l'état d'être sauf, c'est-à-dire être protégé contre les conséquences de défaillances, d'erreurs, d'accidents, et de tout événement indésirable. Elle peut être aussi définie comme le contrôle de dangers identifiés pour maintenir un niveau de risque acceptable (16). Elle est alors vue comme un sous-objectif de la sûreté de fonctionnement, notion issue des activités humaines et industrielles à risque et formalisée par la cyndinique qui est la science du danger (9, 15). Elle s'appuie sur une démarche de type gestion des risques, qui consiste à identifier les événements redoutés et à évaluer tant la gravité de leurs conséquences que la probabilité d'occurrence de tels événements redoutés. La démarche consiste à éviter les accidents et, si des accidents surviennent, à en réduire les effets en mettant en œuvre des dispositifs de protection tels que des barrières.

Une telle démarche atteint ses limites dès lors que les dispositifs de sécurité sont désactivés, que les barrières sont franchies ou que les évolutions du contexte opérationnel engendrent de nouveaux événements redoutés qui ne peuvent pas être identifiés en phase amont des projets. Ces situations amènent à repenser la sécurité du point de vue de la résilience pour prendre en compte les situations réelles opérationnelles et donner aux opérateurs les moyens de piloter à vue.

Si, à l'origine, les événements redoutés de la sécurité relevaient de perturbations non prévues ou de défaillances, c'est-à-dire de comportement non volontaire, la généralisation des technologies de l'information et l'utilisation de ces technologies de l'information dans des dispositifs de commande et de contrôle d'infrastructures critiques, ainsi que dans les dispositifs de sécurité, ouvrent la voie à des comportements malveillants, c'est-à-dire des événements redoutés de nature intentionnelle. Ces comportements malveillants sont peu traités et pris en compte dans le domaine de la sûreté qui s'appuie principalement sur les modèles de défaillance des composants du système.

Par ailleurs, que ce soit dans le cadre de la sécurité ou dans celui de la sûreté, lorsqu'un incident se produit, qu'il soit intentionnel ou pas, il est nécessaire de le détecter et d'émettre une alerte pour traiter l'incident et sa cause. Le système d'alerte doit lui-même être sécurisé pour éviter qu'il ne soit leurré.

### 2.3 Complémentarité entre sûreté et sécurité

Notre article s'inscrit dans ce contexte de complémentarité entre sécurité et sûreté, et plus précisément, dans le cadre de la résilience des systèmes sociotechniques, c'est-à-dire : leur capacité à réagir et à récupérer après une perturbation, avec un minimum d'effet sur la stabilité de leur dynamique (17), en particulier pour faire face à des événements perturbateurs imprévisibles, sans précédent (14). La prise en compte de ces événements perturbateurs imprévisibles, sans précédent, ne peut être faite qu'en exploitation du système, en surveillant l'état du système et de son environnement afin d'alerter les opérateurs et leur permettre de naviguer à vue (22).

La sûreté et la sécurité ont pour point commun de préconiser de surveiller le système et d'alerter lorsqu'un événement redouté advient (10, 22). Elles ont aussi pour point commun que les barrières élaborées d'une part pour éviter qu'un événement redouté ne survienne, d'autre part pour réduire ses conséquences en cas de survenue, sont contournées. Les origines du contournement de ces barrières sont différentes dans les deux cas. Dans le domaine de la sûreté, il est principalement dû à l'accroissement des performances (19, 1) et aux évolutions de l'environnement opérationnel (21). Dans le domaine de la sécurité, il traduit un comportement intentionnel et malveillant, toute l'énergie étant consacrée à contourner ou détruire les barrières mises en place.

Dans tous les cas, il est nécessaire que les opérateurs sachent quel est l'état des barrières et qu'ils soient informés des attaques dont elles peuvent être la cible. Les opérateurs doivent aussi être informés du niveau de danger et de la présence, ainsi que de la contamination du système par un agent malveillant. Ils doivent alors avoir la capacité à naviguer à vue (i.e. assurer la viabilité du système sans disposer de modèle *a priori*

fiable de l'interaction de l'environnement avec le système), sur le plan de la sûreté, mais aussi à découpler ou arrêter des composants contaminés afin de limiter la contamination d'autres composants du système, et décontaminer ces composants avant de les remettre en fonctionnement, sur le plan de la sécurité.

Les analyses préliminaires de risques ainsi que la conception et la réalisation des dispositifs de sûreté, y compris les dispositifs de surveillance et d'alerte (22), doivent prendre en compte les exigences de la sécurité. En effet, dès lors que des composants d'un système échangent des informations, *a fortiori* lorsque des systèmes différents échangent des alertes, les informations échangées ne doivent pas être accessibles à ceux qui ont des comportements malveillants (confidentialité). Les informations échangées ne doivent pas être modifiées, corrompues, pour générer des fausses alarmes ou pour cacher des alarmes (intégrité). Et les dispositifs de sûreté ne doivent pas être détournés de leur usage à des fins de nuisance (non-répudiation). Enfin, les défauts de sécurité doivent être pris en compte comme étant des sources potentielles de danger. Ainsi l'analyse des menaces de la sécurité rejoint et complète l'analyse des modes de défaillance de la sûreté.

Le tableau 1 met en perspective les caractéristiques communes et les caractéristiques spécifiques de la sûreté et de la sécurité.

| | **Sûreté** | **Sécurité** |
|---|---|---|
| **Démarche d'analyse des risques** | Identification des événements redoutés, de leur probabilité d'occurrence et de leurs conséquences | Identification des menaces, de leur vraisemblance, des vulnérabilités du système cible et de leurs conséquences |
| **Caractéristique** | Accident ou défaillance d'un composant du système (événement non intentionnel) | Attaque (événement intentionnel, malveillant) |
| **Anticipation** | Simulation pour mesurer la performance du système face à des événements redoutés prévisibles | Absence de mesure de performance du système |
| **Méthode à mettre en œuvre** | Méthodes de la sûreté de fonctionnement (AMDEC…) | EBIOS, FEROS |
| **Dispositif de sécurité** | Barrières, redondance, réduction des modes communs … | Pare-feu, accès par diode, système de détection d'intrus … |

**Tableau 1.** Analyse comparée de la sûreté et de la sécurité.

## 3 Modèle d'architecture orientée service intégrant la sûreté et la sécurité

Nous proposons un modèle d'architecture orientée service intégrant la sûreté et la sécurité.

Ce modèle d'architecture vise à détecter les accidents, les dérives du point de la sûreté et les actes malveillants, les intrusions, du point de vue de la sécurité.

Nous présentons l'approche fonctionnelle de la complémentarité entre sûreté et sécurité puis nous regardons les impacts de la prise en compte de la sécurité et de la sûreté, avant de présenter le modèle d'architecture orientée service.

### 3.1 Approche fonctionnelle de la complémentarité entre sûreté et sécurité

Tant dans une démarche de sécurité (5, 6, 20), que dans une démarche de sûreté (22), il est recherché de surveiller l'état du système en fonctionnement et d'alerter lorsqu'une situation particulière (intrusion, proximité par rapport à une zone de danger) survient.

Le tableau 2 montre la comparaison des deux dimensions que sont sécurité et sûreté, vis-à-vis des trois fonctions que sont surveiller, évaluer les risques et alerter.

| | **Sûreté** | **Sécurité** |
|---|---|---|
| **Surveiller** | Obtenir une représentation de l'environnement du système<br>Obtenir une représentation de la dynamique de système | Surveillance de l'intégrité et gestion du changement<br>Détection d'attaque<br>Surveillance des équipements / automates<br>Surveillance de la communication |
| **Evaluer les risques** | Évaluer des dérives<br>Évaluer la proximité du danger | |
| **Alerter** | Alerter et conseiller les opérateurs | |

**Tableau 2.** Analyse comparée de la sûreté et de la sécurité.

Il y a de fortes similarités quant à la surveillance des systèmes. Au regard des documents analysés (5, 6, 20), l'évaluation des risques et l'alerte semblent moins formalisées, relevant peut-être d'une intervention manuelle. Pour autant, l'évaluation de la menace ainsi que l'alerte sont des activités clef de la sécurité.

Nous proposons d'appliquer ces trois fonctions à la sûreté et la sécurité pour contrôler l'une et l'autre lorsque le système est mis en œuvre, au stade d'utilisation.

Par ailleurs, nous complétons cette approche en identifiant, sans prétendre faire une analyse fonctionnelle systématique de la sécurité, ce qui nécessiterait plus d'un article, les fonctions de sécurité suivantes relatives à la détection et au traitement approprié d'actes malveillants et d'intrusion.

- surveiller les systèmes opérants et détecter les actes malveillants et les intrusions qu'ils subissent ;

- tant que les systèmes sont exempts d'agent contaminant, émettre des patentes nettes[2] attestant de leur état de santé ;
- en cas d'actes malveillants, d'intrusion, lancer une alerte pour évaluer le périmètre de la zone compromise, l'identité de l'agent compromettant, les fonctions atteintes, leur criticité, et les fonctions alternatives non compromises susceptibles de limiter la dégradation du service ;
- réévaluer le niveau de sûreté, en fonction des actes malveillants détectés et de leurs conséquences, en tenant compte de la criticité du système, de son niveau de sûreté actuel et des conséquences des actes malveillants ;
- confiner les systèmes atteints, retirer les patentes nettes des systèmes contaminés, les blanchir, les mettre en quarantaine jusqu'à retour d'une situation saine, exempte d'agent compromettant ;
- déconfiner les systèmes qui ont recouvré un fonctionnement sécurisé et leur délivrer les patentes nettes leur permettant de fournir les services à leurs clients et à consommer ceux de leurs fournisseurs.

Les patentes nettes peuvent être complétées par des patentes brutes[3] pour les systèmes modérément infectés ou dont le fonctionnement est potentiellement dangereux, hors de sa zone de fonctionnement sûr. Cette patente brute permet de conserver opérationnels des systèmes dès lors que leur fonctionnement n'est pas dangereux pour les autres. Pour des raisons de sûreté, les systèmes critiques ne peuvent faire appel qu'à des services de systèmes présentant des patentes nettes. En particulier, des systèmes non critiques peuvent faire appel aux services de systèmes présentant des patentes brutes pour limiter les conséquences de délestage.

Après avoir présenté les fonctions de sécurité et de sûreté, nous poursuivons en appréhendant les impacts de la sûreté et de la sécurité sur l'architecture du système.

### 3.2 Architecture du système : impacts de la sûreté et de la sécurité

L'intégration de la sûreté et de la sécurité a des conséquences sur l'architecture du système. En effet, l'architecture fonctionnelle résultant des exigences de sûreté et ses dispositifs adaptés (22), ainsi que l'architecture résultant de la prise en compte de la sécurité, doivent être tissées avec l'architecture fonctionnelle « traditionnelle » du système – nous nous plaçons ici dans le paradigme habituel séparant les exigences fonctionnelles et non fonctionnelles, où les problématiques qui nous intéressent ici sont souvent qualifiées de « ities »[4] (13). Ces deux architectures de sécurité et de sûreté sont en fait transverses aux fonctions du système, et donc à l'architecture fonctionnelle basée sur la décomposition des fonctions du système et évidemment à

---

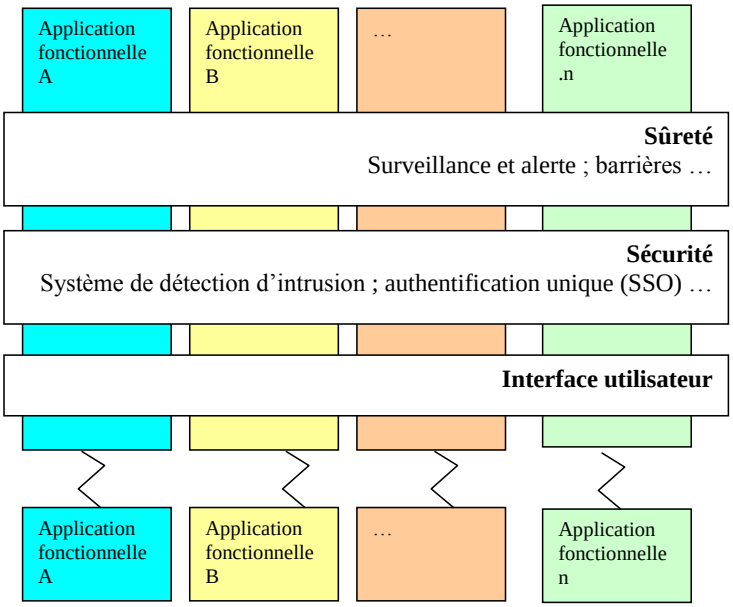
[2] Dans la marine marchande, une patente nette est une attestation légale qui constate qu'un navire est sorti d'un pays exempt de maladies contagieuses.

[3] Certificat de santé délivré dans les ports aux vaisseaux attestant qu'ils sont partis d'un pays infecté.

[4] Les « ities », suivant l'usage anglo-saxon, dénotent l'ensemble des propriétés non fonctionnelles d'un système et des disciplines afférentes : sécurité, utilisabilité, maintenabilité, portabilité, entre autres.

l'allocation de ces fonctions à des composants du système (cf. Figure 1). Ainsi, chaque fonction et chaque composant résultant dans l'allocation – sans présager d'une quelconque univocité entre les deux – sont affectés par ces deux dimensions (sûreté et sécurité). Pour que l'architecture fonctionnelle et l'architecture organique puissent évoluer facilement avec le minimum d'impacts sur ces deux dimensions, par exemple pour prendre en compte de nouveaux besoins, de nouvelles solutions technologiques, il est souhaitable que ces architectures soient découplées. Il en est de même pour pouvoir prendre en compte de nouvelles menaces de sécurité et de nouveaux événements redoutés de sûreté en minimisant les impacts sur l'architecture fonctionnelle et l'architecture physique du système. La conception orientée aspect (24) permet de séparer ce qui relève, d'une part des architectures fonctionnelle et organique du système, et d'autre part des dimensions de sécurité et de sûreté.

**Figure 1.** Architecture du système global, de la sûreté et de la cybersécurité




Si découpler les architectures fonctionnelle et organique du système des dimensions de la sécurité et de la sûreté est une première étape, il est aussi souhaitable de découpler, pour partie, ce qui relève de la sûreté et de la sécurité. En effet, ces deux dimensions évoluent à des rythmes différents. Si la stabilité est principalement recherchée pour la sûreté, avec souvent de fortes exigences de certification – comme dans le domaine aéronautique –, la prise en compte des menaces émergentes impose un rythme d'évolution plus élevé pour la sécurité afin de pouvoir déployer rapidement des solutions de sécurité adaptées à des menaces émergentes. Cela affecte directement les architectures des dispositifs de sûreté et de sécurité.

En contrepartie, découpler complètement la sûreté et la sécurité a pour conséquence de multiplier les capteurs et les dispositifs d'alerte, ce qui peut nuire à l'architecture globale du système et à ses performances, puisque chaque mesure, à un titre ou à un autre, perturbe le fonctionnement du système. De plus, certaines informations peuvent être pertinentes tant pour la sûreté que pour la sécurité. Il est donc nécessaire de s'assurer que des composants qui puissent être mutualisés (dispositifs de surveillance, dispositifs d'alerte) ne soient pas affectés par des rythmes d'évolution différents.

Dans ce contexte, les activités d'architecture du système globale, d'architecture de sûreté et d'architecture de sécurité doivent être menées de concert pour déterminer les dispositifs surveillés en fonction des risques identifiés (défaillance, attaque…), les capteurs nécessaires à la surveillance de ces dispositifs, les informations issues de ces capteurs et nécessaires pour alerter les opérateurs, les chaînes de ce traitement de ces informations, les alertes élaborées et les chaînes de transformation de ces alertes pour informer les opérateurs en fonction de leurs activités, des dispositifs d'interface utilisateur mis en œuvre et en fonction des contextes d'usage. L'analyse doit être menée au regard des contraintes pesant sur ces éléments (dispositifs surveillés, capteurs, informations recueillies, alertes…), en particulier leur niveau de pertinence tant pour la sûreté que pour la sécurité, ainsi que leurs rythmes d'évolution différents.

De plus, mener ces activités d'architecture de concert permet de prendre en compte les interactions entre la sécurité et la sûreté. Il faut évaluer les conséquences en rapport à la sûreté d'actes malveillants ou d'intrusion. En contrepartie, il faut aussi évaluer les conséquences en rapport à la sécurité d'un accident ou d'un fonctionnement du système en dehors de sa zone de mise en œuvre de façon sûre. Les actions menées suite à un acte malveillant doit prendre en compte, outre les conséquences de cet acte malveillant, les conséquences probables pour les systèmes critiques en interaction avec le système attaqué. Cela se traduit par une politique de confinement qui doit être mesurée. Suffisante pour confiner une intrusion malveillante, sans excès, sinon les systèmes confinés n'interagissent plus entre eux et globalement, ils ne sont plus disponibles. De plus, confiner un système suppose que cela ne mette pas en péril son fonctionnement sûr et donc sa sûreté.

La sécurité et la sûreté doivent être intégrées pour que ces interactions puissent être efficaces et assurer tat la sécurité que la sûreté des systèmes concernés.

Le modèle d'architecture orientée service que nous présentons maintenant intègre la sécurité et la sûreté.

### 3.3 Vers un modèle d'architecture orientée service

Après avoir les besoins d'intégration de la sécurité et de la sûreté, nous proposons une architecture multi-niveaux couvrant la sécurité et la sûreté et permettant aux systèmes d'interagir de façon sûre et sécurisée.

Cette architecture comprend trois niveaux.

- le niveau des systèmes opérants (22) consistant à :
  - fournir des services aux systèmes qui en sont clients, de façon sécurisée et sûre ;

  - utiliser de façon sécurisée et sûre les services fournis par les systèmes ;
  - émettre des alertes de sécurité en cas de détection d'agent compromettant, d'attaque, précisant le système ciblé et l'identité de l'attaquant ;
  - émettre des alertes de sûreté en cas d'accident ou de défaillance détectée, ou en cas de proximité d'une zone de danger
- le niveau du système de sûreté et sécurité :
  - détecter les alertes de sécurité et de sûreté et mettre en place la protection appropriée des systèmes, en particulier des systèmes critiques ;
  - gérer les annuaires des services, des clients, des abonnements, avec les niveaux d'habilitation et les niveaux de priorité associés et des menaces identifiées ;
  - après contrôle de l'absence d'agent compromettant, fournir aux systèmes les patentes nettes dont ils ont besoin ;
  - lancer les défis auprès des systèmes opérants et enregistrer leurs résultats ;
  - confiner, mettre en quarantaine, blanchir et, après contrôle de l'absence d'agent compromettant, lever la quarantaine ;
  - enregistrer les attaques,
  - évaluer les nouvelles menaces non encore répertoriées ;
- le niveau du système de surveillance du système de sûreté et de sécurité :
  - lancer les défis auprès du système de sécurité et de sûreté et enregistrer leurs résultats ;
  - s'assurer de l'absence de compromission du système de sécurité et de sûreté.

Pour chaque système, et pour chaque élément de ces systèmes, il est nécessaire d'identifier le niveau de criticité, tant au niveau de la sûreté que de la sécurité. L'objectif est de protéger les systèmes ayant un niveau élevé de criticité et les ressources dont ils ont besoin. Cette démarche permet d'élaborer une politique de confinement visant, pour la sûreté, à limiter les impacts sur l'environnement du système, et pour la sécurité, à limiter les entrées non autorisées. Cela consiste à réduire le couplage des systèmes critiques et d'augmenter leur autonomie. Cela peut se traduire par la redondance en utilisant des sources différentes et variées pour les ressources des systèmes critiques. Cela se traduit par la priorité attribuée aux systèmes critiques pour accéder à des ressources communes sujettes à la concurrence des autres systèmes.

Il est possible d'élaborer des règles pour accéder à un service donné :

- le système requérant ce service doit en avoir le droit, ce qui suppose une authentification de ce système client et la publication de son niveau d'habilitation l'autorisant à requérir ce service ;
- le système requérant ce service doit exprimer s'il est critique et prioritaire et, si oui, quel est son niveau de priorité ;
- le système fournissant ce service doit exprimer le niveau d'habilitation exigé pour accéder à ce service et les règles de priorité pour accéder au service ;
- le système requérant et le système fournissant ce service doivent présenter une patente nette délivrée par le système de sécurité et de sûreté justifiant qu'ils sont exempts d'agent compromettant.

Cette architecture multi-niveaux permet de différencier plusieurs types de flux. Ce sont, d'une part les flux des services entre systèmes opérants, et d'autre part, les flux relatifs à la sécurité et à la sûreté, entre le système de sécurité et de sûreté et les systèmes opérants, ainsi qu'entre le système de surveillance du système de sécurité et de sûreté et ce dernier.

Le diagramme N² ou matrice de couplage (cf. Figure 2) permet de montrer les flux entre systèmes opérants et les caractéristiques de sécurité et de sûreté de ces flux. Dans ce cas de figure, nous avons trois systèmes opérants différents, réciproquement A, B et C. Nous supposons que le système C est un système critique.

Le système A fournit le service A' aux systèmes B et C. Le système B fournit le service B' au système A et le service B'' au système C. Enfin, le système C fournit le service C' au système A. Outre les propriétés opérationnelles de ces services, ces derniers présentent des caractéristiques de sécurité et de sûreté, à savoir le niveau d'habilitation requis pour accéder au service (sécurité), le niveau de priorité du service (sûreté), ainsi que la patente nette attestant que le service fourni est exempt d'agent compromettant. Cette patente nette est particulièrement importante pour les services A' et B' puisqu'ils sont consommés par le système critique C. Le système critique C est prioritaire par rapport au système B pour consommer le service A'.

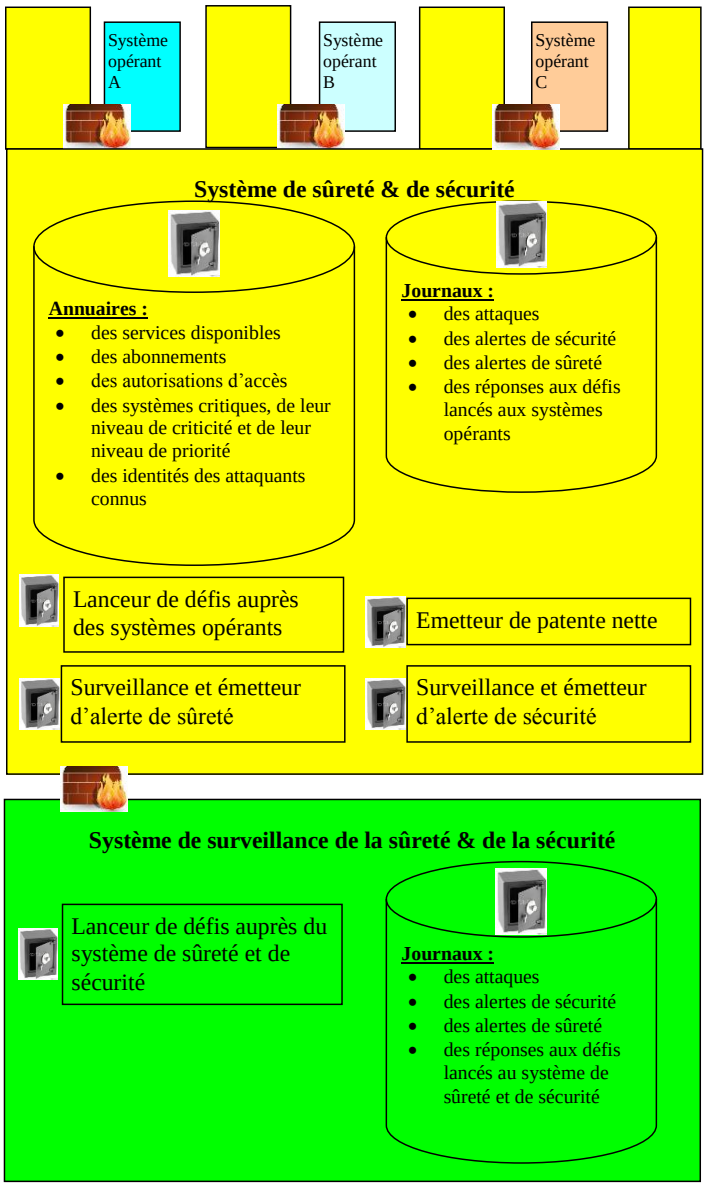
**Figure 2 : Matrice N² décrivant les flux de services entre systèmes opérants.**

| | | |
|---|---|---|
| Système opérant A | Service A' (niveau d'habilitation requis, niveau de priorité, patente nette) | Service A' (niveau d'habilitation requis, niveau de priorité, patente nette) |
| Service B'' (niveau d'habilitation requis, niveau de priorité, patente nette) | Système opérant B | Service B' (niveau d'habilitation requis, niveau de priorité, patente nette) |
| Service C' (niveau d'habilitation requis, niveau de priorité, patente nette) | | Système opérant C (système critique) |

L'architecture proposée en Figure 3 s'appuie sur une architecture orientée service et sépare les différents systèmes opérants (A, B et C) par le système de sécurité et sûreté. Cette architecture reprend et complète l'architecture sécurisée des cartes à microprocesseurs de type Java Card. Les systèmes opérants déclarent au système de sûreté et de sécurité les services qu'ils proposent, en précisant le niveau d'habilitation que ces services requièrent ainsi que leur niveau de priorité si ce sont des services critiques. Les systèmes opérants déclarent auprès du système de sûreté et de sécurité les services qu'ils fournissent, en précisant le niveau d'habilitation requis pour accé-

der à ces services et leur niveau de priorité des services requis si ce sont des services critiques.

**Figure 3.** Architecture proposée intégrant sûreté et sécurité



Le système de sécurité et de sûreté vérifie que le système opérant fournissant un service est exempt d'agent compromettant. Il inscrit le service opérant dans l'annuaire des services disponibles et délivre une patente nette d'une durée limitée au système opérant fournissant ces services.

Le système de sécurité et de sûreté maintient l'annuaire des services que proposent les différents systèmes opérants, ainsi que, pour chaque service, le niveau d'habilitation et le niveau de priorité nécessaires pour le requérir.

Les systèmes opérants s'abonnent auprès du système de sûreté et de sécurité aux services dont ils ont besoin, en précisant le niveau d'habilitation auquel ils ont accès ainsi que le niveau de priorité des services requis si ce sont des services critiques.

Le système de sécurité et de sûreté vérifie que le système opérant requérant un service est exempt d'agent compromettant. Il inscrit le service opérant dans l'annuaire des clients du service demandé et délivre une patente nette d'une durée limitée au système opérant requérant ce service. Le système de sécurité et de sûreté maintient l'annuaire les clients des services et de leur niveau d'habilitation propres leur permettant d'accéder aux services. Il maintient l'annuaire des systèmes critiques qui sont prioritaires en cas de situation dégradée ou de réduction des performances dues à des accidents ou à des actes de malveillance. Cet annuaire des systèmes critiques permet aussi de les alerter afin de remonter leur niveau de vigilance et de protection en cas d'attaque d'un système. Enfin, il maintient aussi l'annuaire des abonnements aux différents services en mentionnant le niveau de priorité attribué à chaque abonnement en fonction de la criticité du système client. Cet annuaire des abonnements aux différents services permet de cartographier les dépendances des systèmes, de façon dynamique, et d'effectuer dynamiquement les délestages en tenant compte de la criticité des systèmes clients en cas d'accident ou d'acte malveillant à l'encontre d'un système fournisseur. Cet annuaire permet aussi d'abonner les clients à d'autres systèmes fournisseurs, isofonctionnels, pour conserver les mêmes niveaux de performance, là encore, en cas d'accident ou d'acte malveillant à l'encontre d'un système fournisseur.

Outre les annuaires, le système de sécurité et de sûreté dispose aussi d'un émetteur de patente nette qui fournit ce type d'attestation après avoir vérifié que le système qui la demande est exempt d'agent compromettant. Enfin le système de sûreté et de sécurité dispose d'un lanceur de défis. Les défis sont lancés auprès des systèmes opérants de façon aléatoire et indépendamment de ces systèmes opérants. Le rôle des défis est de vérifier le bon fonctionnement des systèmes opérants du point de vue de la sécurité en particulier pour vérifier qu'ils ne sont pas contaminés ou n'ont pas subi d'actes malveillants, ainsi que pour vérifier que la vigilance ne s'émousse pas et que ne se développe pas des comportements laxistes. Les résultats du lanceur de défis contribuent à maintenir la patente nette des systèmes sains et de la retirer des systèmes qui s'avèreraient contaminés.

Une telle architecture implique que les annuaires, les journaux, l'émetteur de patente nette et le lanceur de défi soient protégés contre toute attaque, dans un coffre.

En cas d'attaque d'un système opérant critique ou d'un système fournissant des services à un système opérant critique opérant critique, ce dernier est confiné et mis en quarantaine. Ses services sont rendus indisponibles, le temps de la quarantaine, la patente nette est retirée du système opérant contaminé. De même, ses abonnements

aux services d'autres systèmes sont désactivés le temps de la quarantaine. Il est blanchi et sa remise en service est conditionnée, outre par le respect de la quarantaine, par la vérification de l'absence de résidus vestigiaux de l'agent compromettant. La remise en service s'accompagne de la délivrance d'une patente nette.

Enfin, la délivrance de patente brute permet de moduler le niveau de protection en fonction du niveau de contamination et du niveau de criticité et, in fine, limiter les impacts des délestages induits par la mise en quarantaine de systèmes contaminés.

Il s'agit là d'une proposition qui doit être précisée et validée expérimentalement.

## 4 Conclusion

La diffusion des technologies de l'information s'accompagne de la diffusion des risques liés à la sécurité des systèmes d'information. Cette dernière affecte la sûreté des systèmes, en particulier celles des systèmes critiques.

Après avoir analysé la complémentarité entre la sûreté et la sécurité, nous montrons en quoi ces deux dimensions affectent l'architecture d'un système. Pour concevoir une architecture cohérente et évolutive, il est nécessaire de trouver l'équilibre entre le découplage des aspects et leur tissage. Nous proposons un modèle d'architecture orientée service intégrant la sécurité et la sûreté. Ce modèle permet de délivrer des patentes nettes aux systèmes exempts d'agent contaminant et des patentes brutes à des systèmes non critiques faiblement contaminés. Cette architecture permet de moduler la protection en fonction de la sûreté et de la sécurité, en prenant en compte le niveau de criticité des systèmes et leur niveau de contamination.

Les axes de recherche consistent à préciser le contenu des patentes nettes et brutes, de définir le niveau acceptable de contamination pour délivrer des patentes brutes, ainsi que les conditions et modalités de délivrance des patentes nettes et brutes.

Cette proposition d'architecture doit être réalisée et validée expérimentalement.

## 5 Remerciements

Nous remercions Frédéric Pradeilles pour son aide et ses conseils.

## 6 Bibliographie

1. Amalberti R. (2009). Violations et migrations ordinaires dans les interactions avec les systèmes automatisés. Journal Européen des Systèmes Automatisés, vol 43, n° 6, pp. 647-660.
2. ANSSI (2004). Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) ; Section 5 ; outillage pour le traitement des risques SSI. Version du 5 février 2004.
3. ANSSI (2010). Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) ; Méthode de gestion des risques. Version du 25 janvier 2010.
4. ANSSI (2012). Cas pratique. La cybersécurité des systèmes industriels.
5. Brun, J-M., Platel L & Tea F. (2013). Cyber Sécurité des Systèmes de Contrôle Industriels : les spécificités des SCI, un challenge pour leur sécurité. Actes de Computer & Elec-

tronics Security Applications Rendez-vous, Sécurité des systèmes numériques industriels ; pp 6-16.
6. Brun, J-M., Platel L & Tea F. (2013). Sécurité informatique des systèmes de contrôle industriels ; détection et surveillance au niveau des équipements et du bus de terrain. Actes de Computer & Electronics Security Applications Rendez-vous, Sécurité des systèmes numériques industriels ; pp 62-75.
7. CLUSIF (2003). Plan de Continuité d'Activité – Stratégie et solution de secours du SI. Dossier technique.
8. CLUSIF (2014). Plan de Continuité d'Activité, Plan de Reprise d'Activité. Les Synthèses du CLUSIF.
9. EN 60300-3-15 (2010). Gestion de la sûreté de fonctionnement – Partie 3-15 : Guide d'application- Ingénierie de la sûreté de fonctionnement des systèmes.
10. Gad El Rab (2008). Evaluation des systèmes de détection d'intrusion. Université Paul Sabatier - Toulouse III, 2008.
11. ISO/IEC 27032 (2012). Information Technology – Security Techniques – Guidelines for security.
12. ISO/IEC 27001 (2013). Information Technology – Security Techniques – Information Security management systems – Requirements.
13. ISO/IEC/IEEE 15288 (2015). International Standard on Systems and software engineering – System life cycle processes.
14. Luzeaux D. (2011). Ingénierie des grands systèmes complexes. In Luzeaux D., Ruault J.-R. & Wippler J.-L. (Eds.), Maîtrise de l'ingénierie des systèmes complexes et des systèmes de systèmes : études de cas, Hermes-Lavoisier, Paris, pp. 21-106.
15. Ministère de l'écologie et du développement durable (2010). Guide pour l'estimation des dommages matériels potentiels aux biens des tiers en cas d'accidents majeurs.
16. MIL-STD-882E (2012). System Safety. Department of Defense Standard Practice.
17. Pariès J. (2006). Complexity, emergence, resilience. In Hollnagel E., Woods D. D. & Leveson N. (Eds.), Resilience Engineering. Concepts and precepts, Ashgate, Aldershot, pp. 43-53.
18. Pièrre-Cambacèdés L. (2010). Des relations entre sûreté et sécurité. Thèse de doctorat Informatique et Réseaux, soutenue le 3 novembre 2010 (Paris).
19. Rasmussen J. (1997). Risk management in a dynamic society: a modelling problem. Safety Science, vol. 27, n° 2/3, pp. 183-213.
20. Reddy G.S., Rao V.N. & Kumar N. (2012). An intrusion tolerance approach for Internet security. International Journal of Advanced Research in Computer and Communication Engineering, vol. 1, n°8, October 2012.
21. Ruault J.-R., Vanderhaegen F. & Kolski C. (2013). Sociotechnical systems resilience: a dissonance engineering point of view. 12th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systems, Las Vegas.
22. Ruault J-R. (2015). Proposition d'architecture et de processus pour la résilience des systèmes ; application aux systèmes critiques à longue durée de vie. Thèse de Doctorat d'Automatisme soutenue le 7 juillet 2015 (Valenciennes).
23. Secrétariat général de la défense nationale (1991). Fiche d'expression rationnelle des objectifs de sécurité.
24. Tarby J-C., Ezzedine H., & Kolski C. (2009). Trace-based Usability Evaluation Using Aspect-oriented Programming and Agent-based Software Architecture. In: Seffah A., Vanderdonckt J. and Desmarais M. (Eds.), Human-Centered Software Engineering: Architectures and Models-Driven Integration, Springer HCI Series, pp. 257-276.

# Coordonner sûreté et cybersécurité : quelques principes et perspectives issus de la conception de contrôle-commande des centrales nucléaires

Ludovic Pietre-Cambacedes (EDF, Septen), Victor Vuillard (EDF, C3D)

{ludovic.pietre-cambacedes, victor.vuillard}@edf.fr

**Abstract.** Sûreté et cybersécurité sont intimement liées. Concernant autrefois des systèmes distincts, elles doivent de plus en plus être traitées de façon coordonnée, dans un souci de gestion optimale des risques et de résilience des installations. Nous tentons de dégager dans cet article un certain nombre de principes et de retours d'expérience en la matière, tirés de la conception des systèmes de contrôle-commande des centrales nucléaires. Les propos sont génériques afin d'être utilisables par d'autres secteurs industriels. Au-delà de ces enseignements, nous donnons aussi un rapide panorama des initiatives en cours sur la thématique dans d'autres secteurs : industrie et recherche ont pris conscience des enjeux et travaillent activement sur le sujet.

## 1 Introduction

Les systèmes de contrôle-commande jouent un rôle fondamental dans le pilotage et la sûreté[1] des centrales nucléaires ; ils répondent de longue date à des contraintes strictes de conception et d'exploitation permettant d'assurer ces fonctions. La problématique de sécurité informatique[2] (ou cybersécurité) est également connue et prise en compte depuis plusieurs décennies, mais son traitement a évolué de façon plus significative ces dernières années : le domaine connait en effet des mutations rapides, stimulées par la généralisation des technologies numériques, mais aussi par l'intensification et la diversification des menaces. Dans cette communication, nous tentons de dégager un certain nombre de principes et de retours d'expérience tirés de la conception des systèmes de contrôle-commande des centrales nucléaires, en vue de mieux coordonner les dispositions de sûreté et celles prises pour la cybersécurité. La qualité de cette coordination conditionne directement la résilience de l'installation, i.e. son « aptitude intrinsèque à ajuster son fonctionnement avant, pendant ou après la survenue de changements ou de perturbations et ce afin qu'il puisse poursuivre son activité » [1].

Même si le contenu de cet article se base sur les pratiques dans le domaine nucléaire, notre communication n'est pas, ou peu, spécifique à ce secteur. Elle vise à partager des éléments que nous espérons utilisables par la communauté C&ESAR.

---

[1] Dans le sens de l'industrie nucléaire, c'est-à-dire visant *in fine* à protéger les travailleurs, le public et de l'environnement contre des risques radiologiques indus.

[2] Dans notre contexte, au sens protection contre des attaques par vecteur numérique.

## 2 Éléments de contexte et généralisation

### 2.1 Des architectures intrinsèquement conçues pour la sûreté

**Base de conception.** Les architectures de contrôle-commande des centrales nucléaires sont conçues pour assurer en priorité l'atteinte des trois objectifs principaux pour la sûreté de l'installation : le contrôle de la réaction de fission nucléaire, l'évacuation de la puissance résiduelle dégagée par cette réaction et le confinement des substances radioactives. Dans cette optique, la conception se base sur plusieurs principes, dont :

- un classement de sûreté pour chacun des systèmes de contrôle-commande, selon les fonctions de sûreté (elles-mêmes catégorisées) qu'il supporte. Ces classes et ces catégories impliquent le respect d'exigences croissant en nombre et en rigueur avec leur importance pour la sûreté ;
- le critère de défaillance unique (la fonction de sûreté doit être assurée malgré une défaillance quelconque d'un composant du système considéré) ;
- le concept de défense en profondeur, au sens nucléaire du terme (impliquant notamment des lignes de défense techniques et organisationnelles, hiérarchisées et indépendantes, permettent de prévenir, détecter et réagir vis-à-vis des situations de déviation du fonctionnement normal, des situations accidentelles, voire en cas d'accidents graves). Nous y reviendrons.

Ces principes, et bien d'autres aspects que nous ne pouvons pas développer ici faute de place (par ex. la qualification des matériels et des logiciels), sont normalisés dans des textes internationaux comme l'IEC 61513 [2], IEC 60880 [3], etc., et déclinés dans nos référentiels nationaux.

**Digressions autour du concept de défense en profondeur.** Le cheminement du concept de « défense en profondeur » (*defense-in-depth*), fondamental en sûreté et pour la conception et l'exploitation du contrôle-commande des centrales nucléaires, illustre bien les relations intimes entre sûreté et sécurité. Ce concept (lignes de défenses successives) existait déjà dans le domaine militaire au XV$^{ème}$ siècle (partie intégrante du tracé à l'italienne), pour la construction des forteresses. Il est aujourd'hui aussi considéré comme un principe clé en sécurité informatique. L'idée générale y est que tout dispositif de cybersécurité considéré individuellement est *a priori* insuffisant : une défense appropriée ne peut être atteinte qu'en combinant plusieurs barrières, complémentaires, idéalement indépendantes, combinant mesures techniques et organisationnelles, assurant prévention, détection et réaction. La référence [4], en plus de rappeler l'historique du concept, propose une analyse et une rationalisation de la démarche de défense en profondeur pour l'informatique. Cependant, il est intéressant de noter qu'elle a d'abord trouvé une place de choix dans la sûreté en conception des centrales nucléaires, où elle y est codifiée et déclinée depuis des décennies. Une déclinaison concrète tient dans les trois barrières successives indépendantes permettant de confiner la matière radioactive (gaine combustible, cir-

cuit primaire puis enceinte réacteur). La défense en profondeur se manifeste aussi par le cumul d'une conception spécifiquement tournée vers la sûreté, avec des procédures et un contrôle opérationnel également adaptés à cette fin. Elle conduit en outre à la diversification et la redondance des dispositifs de sûreté dans la centrale, y compris pour le contrôle-commande. À titre d'exemple, le système de protection réacteur, qui a pour principales fonctions la détection de situations anormales, l'arrêt automatique du réacteur et le déclenchement des systèmes de sauvegarde, est redondé sur plusieurs voies indépendantes. Chacune des voies suffit à remplir l'ensemble des fonctions de sûreté du système de protection. Il est de plus secondé par un moyen diversifié en cas de défaillance. On peut aussi remarquer que l'ensemble ne constitue qu'une « barrière » dans l'approche globale de défense en profondeur mise en œuvre du point de vue sûreté nucléaire.

### 2.2 Prise en compte de la cybersécurité dans le contexte nucléaire

**Généralités.** La cybersécurité est une problématique connue et prise en compte depuis plusieurs décennies sur les installations nucléaires, mais elle a évolué de façon significative ces dernières années, du fait de la pénétration accélérée des technologies numériques dans les équipements industriels, et de par l'intensification et la diversification des menaces. La médiatisation de Stuxnet a bien entendu constitué un tournant dans la prise de conscience de ces évolutions, et l'actualité est désormais régulièrement rythmée par des annonces concernant des campagnes d'attaques ciblant des systèmes numériques industriels. Le Chapitre IV de [5] donne un panorama historique des attaques ayant ciblés ces systèmes, et discute de l'évolution des menaces associées.

**Référentiels et approche graduée.** Les pratiques nationales diffèrent, cependant, à l'instar du chemin parcouru par la sûreté à quelques décennies de décalage, un certain nombre de consensus internationaux se dégagent. Le premier texte de référence a été publié par l'AIEA (Agence Internationale de l'Energie Atomique) en 2011, après de longues années de gestation [6]. Parmi les éléments qui y sont développés, on peut citer le concept d'approche graduée. L'approche graduée repose sur la définition et l'attribution aux systèmes de degrés de sécurité, auxquels sont associées des exigences graduées en fonction des conséquences potentielles (et en premier lieu la sûreté nucléaire) d'une attaque sur le système considéré.

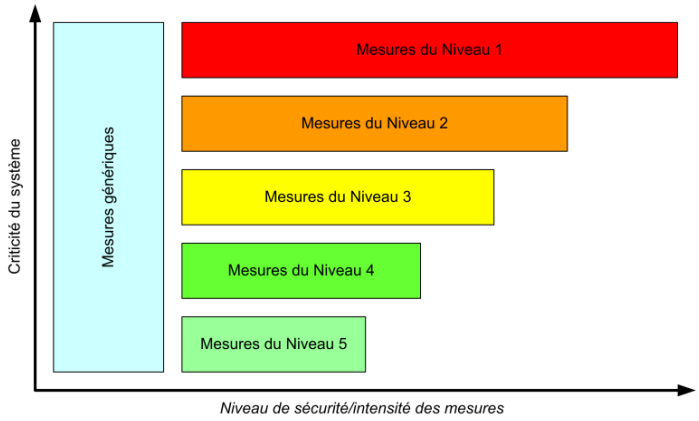



**Fig. 1.** – Approche graduée selon le guide NSS 17 de l'AIEA (extrait [6])

Ce guide contient un certain nombre de recommandations techniques, mais a également une portée managériale, développant des principes organisationnels importants, comme la vision intégrée de la sécurité, les processus de surveillance et d'amélioration, une filière indépendante de sécurité informatique à l'analogue de celle de la sûreté nucléaire, etc.

Les normes internationales ont suivi de peu, en cohérence avec les principes de l'AIEA, à commencer par l'IEC 62645 [7][3], publiée en 2014, et dédiée aux programmes de sécurité pour les systèmes de contrôle-commande des centrales nucléaires. On y retrouve entre autre l'approche graduée, avec la définition de trois degrés de sécurité (cohérents avec le sur-ensemble de 5 niveaux défini dans [6]), intimement liés avec le classement déjà existant du point de vue de la sûreté[4].

**Relations entre classement de sûreté et de cybersécurité.** Point clé : il n'y a pas de bijection directe entre classement de sûreté et classement de cybersécurité tels que définis dans les référentiels internationaux. Bien entendu, le premier constitue une donnée d'entrée fondamentale pour le second, et au plus un système est important pour la sûreté, au plus il sera classé d'un point de vue cybersécurité, mais il est tout à fait possible d'attribuer un degré élevé de sécurité à un système peu ou pas classé de sûreté, car d'autres aspects sont à prendre en compte (par ex. le rôle potentiel du système considéré dans une attaque). Cependant, dans le contexte d'une installation nucléaire, la cybersécurité sert en premier lieu la sûreté ; cet enjeu prioritaire prime en effet sur tous les autres. Le classement sécuritaire des systèmes le reflète bien. Plus largement, toute démarche de cybersécurité doit dans ce contexte prendre en compte les impératifs de sûreté. Cette situation explique que la communauté du contrôle-commande nucléaire se soit rapidement penchée sur la coordination de ces aspects.

[3] IEC, *International Electrotechnical Commission* (www.iec.ch)

[4] Les systèmes de sûreté sont de classe 1, 2, 3 ou NC (Non-Classé) et peuvent selon leurs classes supporter des fonctions de sûreté de catégorie A, B, C ou des fonctions NC [8].

### 2.3 Généralisation et illustration de la problématique

Les relations entre sûreté et cybersécurité ne se limitent bien sûr pas aux classements. Si on cherche à généraliser, indépendamment du contexte nucléaire, on peut en fait distinguer quatre grandes catégories d'interdépendance :

- renforcement mutuel : des mesures prises à des fins de sûreté contribuent également à la sécurité ou réciproquement ;
- antagonisme : les exigences ou mesures de sûreté et de sécurité mènent, considérées conjointement, à des situations conflictuelles ;
- dépendance conditionnelle de l'une vis-à-vis de l'autre ;
- indépendance : pas d'interaction (ce qui est en fait précieux à repérer, pour désamorcer certains raccourcis).

Certaines de ces relations peuvent s'illustrer directement sur une architecture de contrôle-commande. Prenons le cas fictif d'un système de propulsion nucléaire de croiseur intergalactique[5]. On suppose les systèmes de l'architecture de contrôle-commande associée utilisant notre référentiel de sûreté nucléaire, et son classement de sûreté. La Figure 2 donne la vue d'ensemble, avec pour chaque élément, indiquée entre crochets, la catégorie des fonctions de sûreté supportées. On y distingue :

- dans la partie supérieure, une salle de commande, comprenant les postes opérateurs informatisés et un synoptique (interfaces de 3 calculateurs redondants, regroupés en divisions indépendantes nommées div. 1, div. 2 et div. 3), et un panneau conventionnel utilisé en cas de défaillance des moyens de conduite principaux ;
- une salle de repli, s'appuyant sur la même infrastructure numérique, mais séparée géographiquement ;
- dans la partie inférieure, des systèmes importants pour la sûreté (IPS), regroupés dans des ensembles d'armoires d'automatisme distincts, selon les fonctions de sûreté supportées, et répartis dans des divisions géographiques séparées ;
- les systèmes en charge de la protection réacteur (évoqué en fin de Section 2.1), redondé sur 3 divisions indépendantes.

[5] Ce choix, qui peut surprendre, est lié à la sortie alors imminente de l'épisode 7 de la Guerre des Étoiles (16 décembre 2015) au moment de la conférence où sera présenté cet article. Les spécialistes sont partagés quant au niveau effectif de cousinage avec les architectures de contrôle-commande de centrales nucléaires.

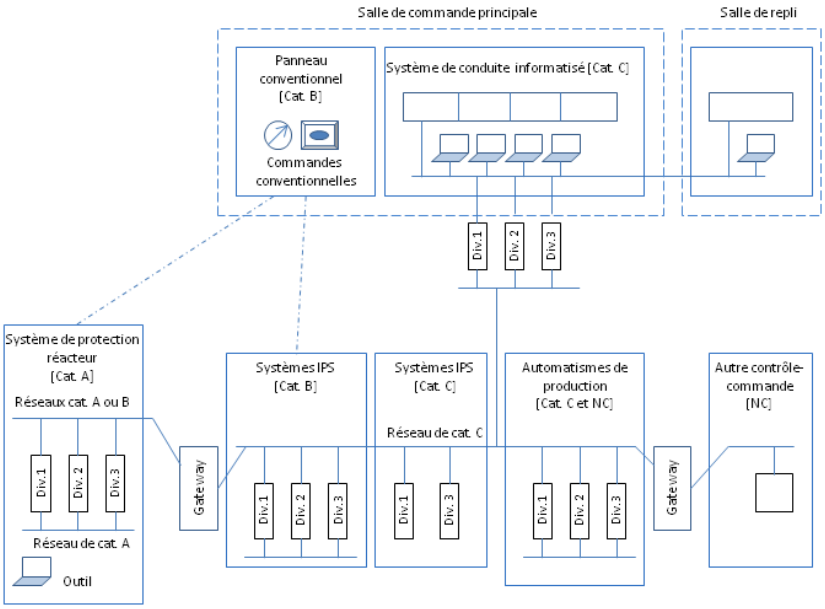



**Fig. 2.** Vue d'ensemble d'une architecture de contrôle-commande (exemple fictif)

Cet exemple permet en fait d'illustrer de nombreux exemples d'interactions entre dispositions de sûreté et de cybersécurité. Pour des raisons de place, nous n'en présenterons que quatre :

- le premier exemple concerne l'existence d'un panneau de supervision et des moyens de commande conventionnels. La diversité technologique permet de faire face à d'éventuelles défaillances liées aux systèmes de conduite principaux, prévues dans les études de sûreté. Cette disposition peut avoir toute son utilité d'un point de vue cybersécurité, en cas d'attaque informatique compromettant l'intégrité ou la disponibilité de ces systèmes, la surface d'attaque de ces moyens diversifiés étant très réduite. Soulignons que pour être pertinente, il faut être capable de détecter une telle compromission ; là encore, des dispositions prises au titre de la sûreté peuvent faciliter la détection ;

- le deuxième exemple concerne les passerelles (*gateways*). Elles ont un double rôle. Du point de vue de la sûreté, elles assurent que les systèmes les moins classés de sûreté ne peuvent pas « polluer » les systèmes plus classés (comme requis par les référentiels prescriptifs de sûreté). Elles ont aussi un rôle de segmentation et de filtrage, souvent extrêmement strict, d'un point de vue cybersécurité. Le point important ici est de souligner que si les dispositions prévues au titre de la sûreté sont souvent valorisables au titre de la cybersécurité, ce n'est d'une part pas toujours le cas, et d'autre part, souvent pas suffisant. En effet, un mécanisme permettant

d'arrêter un trafic issu d'une défaillance d'origine aléatoire peut s'avérer inefficace face à une attaque ciblée, impliquant un trafic spécifiquement créé pour la circonstance. Des mécanismes conçus pour la cybersécurité peuvent s'avérer nécessaires ;

- le troisième exemple concerne les systèmes de la salle de commande principale, et ceux de la station de repli. Séparés géographiquement pour de raison de sûreté (gestion de situations pour lesquelles la salle de commande n'est plus utilisable), les systèmes partagent un même réseau numérique ; sans dispositions particulières, il convient donc de les considérer comme faisant partie d'une même « zone » de cybersécurité, malgré leur séparation géographique. Les dispositions de sûreté et cybersécurité sont ici « indépendantes » ;

- le quatrième et dernier exemple porte sur les systèmes regroupés entre les deux passerelles dans la partie inférieure de la Figure 2. Ces groupes de systèmes portent des fonctions de sûreté avec des classements différents (B, C voire non classées) ; au même titre que pour le 3ème exemple, ces groupes de systèmes sont à considérer comme faisant partie d'une même zone de sécurité et doivent avoir le même degré de sécurité, défini en considérant les fonctions les plus classées de sûreté. Certaines distinctions faites au titre de la sûreté ne sont plus de mise d'un point de vue cybersécurité.

### 2.4 Vers une norme internationale sectorielle (IEC 62859)

La communauté nucléaire a rapidement pris conscience du besoin de coordonner sûreté et cybersécurité, dès la conception des architectures et des systèmes de contrôle-commande, puis durant toutes les étapes de leurs cycles de vie. Entre autres initiatives, une norme internationale, l'IEC 62859, est en cours de développement [9]. Ce développement respecte et précise les principes de haut niveau édictés par l'AIEA (comme toutes les normes IEC sur le contrôle-commande nucléaire), et s'appuie sur la participation de la plupart des pays opérant des centrales nucléaires. Initiée en 2012, la publication de l'IEC 62859 est prévue fin 2016. Son texte, sans être complètement stabilisé, est à un stade avancé ; il est structuré autour de trois clauses principales, elles même subdivisées :

- une clause traite de la coordination sûreté-cybersécurité à la maille de l'architecture globale. Elle comprend une série de principes fondamentaux et génériques, puis diverses sous-clauses traitant d'aspects plus spécifiques (notamment la délimitation des zones de cybersécurité vis-à-vis des découpages sûreté) ;
- une autre traite de la problématique à l'échelle des systèmes de contrôle-commande composant l'architecture. Là encore, la clause commence par fournir des principes fondamentaux, puis regroupe des exigences et recommandations cette fois par phase du cycle de vie du système considéré ; elle finit par des focus sur des thèmes techniques spécifiques tels que le contrôle d'accès en salle de commande, les modifications et mises à jour logicielles, l'utilisation de la cryptographie, etc. ;
- une dernière traite des aspects organisationnels et de gouvernance.

# 3 Quelques éléments généralisables

## 3.1 Préambule

Une partie des principes généraux présentés ci-après sont issus des travaux sur le développement de la norme IEC 62859 (où ils sont développés et « particularisés » pour le contexte nucléaire) évoquée en Section 2.4. D'autres contenus sont repris ou résumés du Chapitre IX de l'ouvrage [5]. L'exercice de généralisation effectué est périlleux : chaque secteur, chaque contexte industriel possède réellement son lot de spécificités gommées ici. Le résultat est donc forcément macroscopique. Aussi, nous invitons le lecteur à se saisir de ces quelques principes (de base en 3.2, sur les architectures en 3.3, sur le cycle de vie des systèmes en 3.4) comme point de départ, de les décliner, et sans doute les compléter, pour son environnement.

## 3.2 Quatre principes de base

Une intégration réussie des contraintes de sûreté et de cybersécurité pour les systèmes numériques industriels nécessite – sans pour autant que cela soit suffisant - le respect de quelques principes de base. Citons dans cet article les quatre suivants :

- Les dispositions de cybersécurité ne doivent pas empêcher les mécanismes de sûreté d'atteindre leurs performances requises (fiabilité, temps de réaction, etc.). Notons que la réciproque est aussi à considérer : les situations où les mécanismes de cybersécurité seraient entravés par des dispositions de sûreté sont à identifier, et des approches alternatives à privilégier.
  *Illustration :* un exemple bien connu tient dans l'utilisation de mécanismes cryptographiques. Les exigences de simplicité et de temps de réponse contraints des systèmes de sûreté peuvent être incompatibles avec les traitements supplémentaires, notamment cryptographiques, classiquement employés pour la cybersécurité de l'informatique tertiaire (notamment, signature numérique pour garantir l'intégrité des données) ; au-delà des problématiques de temps de traitement, que la loi de Moore tend à estomper même pour les systèmes industriels, l'emploi de ces mécanismes est vu comme des sources additionnelles d'erreurs et de défaillances potentielles. Dans le contexte de systèmes de sûreté, il est souvent préférable d'utiliser des mesures d'autre nature pour protéger l'intégrité des données et des traitements, et de limiter l'emploi de la cryptographie à des cas strictement bornés ou périphériques, supportés par des analyses de risque caractérisant le besoin et les conséquences potentielles de l'emploi de ces techniques.

- Les défaillances de mécanismes de cybersécurité peuvent avoir des conséquences sur des mécanismes de sûreté : ces conséquences devraient être évaluées. On veillera à ne pas créer de source supplémentaire de défaillances de cause commune qui pourraient dégrader la sûreté.
  *Illustration :* pour filer l'exemple précédent, l'utilisation de mécanismes cryptographiques pour authentifier les échanges réseaux entre systèmes de sûreté est sus-

ceptible d'entraver leur bon fonctionnement. Une défaillance de mode commun peut par exemple être introduite par une date d'expiration de certificat mal gérée.

- Les dispositions de séparation, d'indépendance, de diversification de systèmes et de prévention de défaillances de cause commune prises au titre de la sûreté peuvent souvent être valorisées au titre de la cybersécurité. Ces synergies devraient être identifiées et exploitées quand cela est possible : elles sont autant de sources d'économie et de simplicité à la conception et à l'exploitation.
  *Illustration :* les *gateways* de la Figure 2 sont de bons exemples ; après analyse, il peut s'avérer que leur mode de cloisonnement apportent de réelles garanties face à certaines attaques ; c'est souvent le cas vis-à-vis des attaques par dénis de service.

- D'une façon plus générale, une propriété ou caractéristique d'un système ou d'une architecture à l'origine prévue pour la sûreté (et plus particulièrement la fiabilisation d'une fonction nécessaire à la sûreté) peut être également considérée a posteriori comme une disposition de cybersécurité (durant une analyse de risque par exemple). Dans ce cas, elle doit cependant être réexaminée avec attention dans cette optique pour confirmer sa valeur ajoutée, en prenant explicitement en compte la nature malveillante et intelligente des événements à considérer (hors périmètre des analyses faites au titre de la sûreté), et ce par des experts en cybersécurité.
  *Illustration :* à titre d'exemple relativement trivial, les codes correcteurs d'erreurs type CRC (*Cyclic Redundant Check*) sont souvent employés dans les systèmes de sûreté. Ils sont vus sous cet angle comme des mécanismes de protection de l'intégrité de données. Ils ne peuvent plus être considérés de la sorte face à un attaquant intelligent ; il est en effet aisé de modifier les données et de recalculer le *checksum* en conséquence. Ainsi, d'un point de vue cybersécurité, la protection de l'intégrité des données nécessite la mise en œuvre de mécanismes spécifiques (par exemple, cryptographiques type MAC, *Message Authentication Code*… se référer alors aux deux premiers principes !).

### 3.3 Trois principes d'architecture

En plus (voire, à partir) des quelques principes de base mentionnés en 3.2, on peut également dégager des principes plus ciblés sur les aspects architecturaux. Nous nous limiterons dans cet article aux trois suivants :

- De nombreux référentiels préconisent de définir des « zones » de cybersécurité (cf. par exemple [6] et [7] pour le nucléaire, ou [10] pour les autres secteurs). Ce découpage doit prendre en compte les éventuelles séparations et indépendances, logiques et/ou géographiques, mises en œuvre au titre de la sûreté, ainsi que les cloisonnements physiques et les restrictions d'accès associés à la protection physique de l'installation. À noter que le simple critère de séparation géographique n'est bien souvent pas suffisant pour délimiter des zones de cybersécurité, des systèmes pouvant être séparés physiquement, mais intimement liés logiquement (communications régulières et/ou non filtrables) : ils ne peuvent pas être considérés comme

faisant partie de deux zones distinctes (et a fortiori, comme ayant un degré de sécurité différent).
*Illustration* : le troisième et le quatrième exemple de la Section 2.3 constituent des illustrations directes de ce principe.

- Si une même zone de cybersécurité réunit des systèmes ayant des rôles d'importances variées vis-à-vis de la sûreté, le niveau de cybersécurité de la zone doit se baser sur le système le plus critique vis-à-vis de la sûreté (entre autres considérations). Une telle situation se présente notamment quand ces systèmes partagent un même bus de données : il n'y a alors souvent pas d'autres choix que de les grouper dans une même et unique zone de cybersécurité, à protéger selon la sensibilité du système le plus critique.
  *Illustration* : le quatrième exemple de la Section 2.3 correspond à cette situation.

- D'une façon plus générale, une connaissance fine des architectures, des systèmes et de leurs communications (nature, sources/destinataires, aspects temporels, etc.) est indispensable aussi bien pour la sûreté de l'installation que pour sa cybersécurité.

### 3.4 Principes associés au cycle de vie

Les quelques principes ci-dessus, valables indépendamment des phases du cycle de vie considérées, peuvent être complétés de façon plus spécifique pour différentes phases du cycle de vie des systèmes ou des architectures.

Pour commencer, si les aspects sûreté sont bien souvent pris en compte dès la phase de spécification de systèmes industriels, ce n'est hélas pas toujours le cas des aspects concernant la cybersécurité, problématique plus récente et moins bien « ancrée » culturellement dans ce contexte. Il est pourtant primordial de les considérer au plus tôt dans le cycle de vie, idéalement dès les spécifications, pour optimiser la bonne intégration entre dispositions de sûreté et de sécurité, tirer profit des synergies éventuelles (voir discussion sur les principes d'architecture en Section 3.3 par exemple) et éviter d'éventuels antagonismes.

Si les renforcements devraient être recherchés et considérés dès les phases de conception, leur exploitation effective dépendra cependant de nombreux autres facteurs, dont l'éventuelle complexification des systèmes (il vaut mieux parfois traiter séparément les problèmes) ou les processus déjà en place (vérification & validation, qualification ou certification). Entre autres aspects considérés à la conception, les choix relatifs à la modularité des systèmes se baseront sur un compromis intégrant les aspects sûreté et cybersécurité : une grande modularité permettra des mises à jour de sécurité plus faciles et ciblées, mais une validation globale, y compris d'un point de vue sûreté et conformité aux référentiels associés, s'avérera plus délicate. Cependant, la multiplication des interfaces et l'augmentation de la complexité ne vont dans le sens ni de la sûreté, ni de la cybersécurité.

Les phases de validation ou de tests pourront éventuellement aussi être optimisées, certaines démarches orientées sûreté (par ex., vérification de code) pouvant apporter de la plus-value d'un point de vue cybersécurité et réciproquement. Un point d'attention récurrent concerne les éventuels tests d'intrusions qui sont à mener dans des conditions ne pouvant pas remettre en cause la sûreté des installations, sur banc de tests par exemple. Comme déjà signalé, les analyses de risque constituent aussi des étapes où une coordination étroite est nécessaire.

Lors de la vie des systèmes et des architectures, et de leur maintenance, les changements et évolutions faits au titre de la sûreté doivent être notifiés aux équipes chargées de la cybersécurité, et réciproquement. L'organisation de ces modifications et de la maintenance est d'ailleurs un processus à fort potentiel de synergie : les personnes en charge de l'ingénierie de ces systèmes ainsi que les personnes sur le terrain peuvent profiter d'interventions pour faire des gestes de sûreté et de cybersécurité (une telle optimisation implique une organisation *ad-hoc*, des compétences et des procédures idoines).

# 4 Zooms et discussions

## 4.1 Cycle de vie

Sûreté et cybersécurité nécessitent de couvrir l'ensemble du cycle de vie des systèmes pour assurer l'atteinte des objectifs de chaque domaine dans la durée. Cependant, chacune des disciplines a développé une approche différente, dans la logique d'activités indépendantes et cloisonnées qui a longtemps prévalu.

De nombreuses tentatives d'unification de ces cycles de vie ont été proposées de façon « conceptuelle » (voir par exemple [11]) mais en pratique une réelle unification n'est souvent pas possible, voire pas souhaitable, comme souligné dans [12]. Les compétences nécessaires et procédures impliquées (y compris au niveau de la gestion de la confidentialité des échanges) diffèrent selon le domaine, et ces différences doivent être considérées. De façon plus générale, on peut distinguer de façon très simpliste trois types de « rapprochement » des cycles de vie sûreté et sécurité, représentés schématiquement dans la Figure 3 page suivante :

- la première souvent discutée, jamais atteinte, correspond à l'unification des approches sûreté et cybersécurité. Elle correspond à une vision assez naïve et n'est en général ni praticable, ni souhaitable ;
- la seconde consiste à coordonner la progression et les étapes, via des échanges ponctuels, tout en gardant séparées les équipes et les processus. Les phases d'analyse de risque constituent un bon exemple : une analyse de risque cybersécurité doit absolument prendre en entrée les résultats des analyses de risques orientées sûreté ; la définition des contre-mesures est un autre moment clé où les principes mentionnés en Section 3 sont à considérer ;

– la troisième, qui correspond à une vision « intégrée », pousse le rapprochement en permettant à certaines étapes de fusionner tout en ménageant d'autres phases distinctes, respectant les spécificités des sujets sécurité et sûreté.

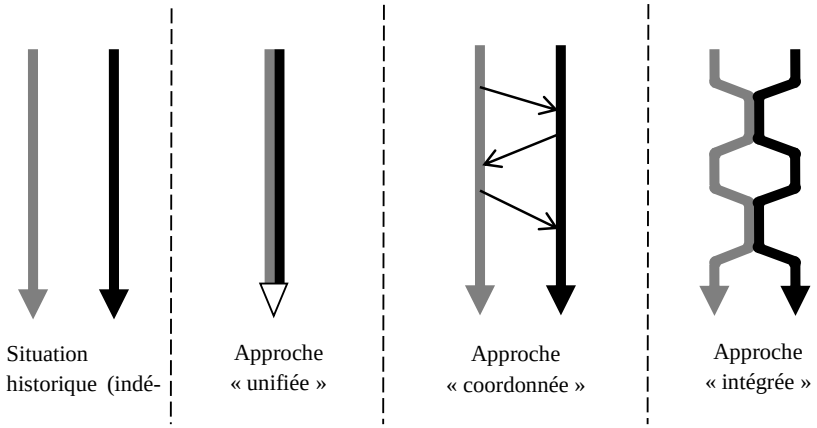



**Fig. 3.** – Différents rapprochements sûreté-cybersécurité envisageables

Le curseur « idéal » entre coordination et intégration est très dépendant des contextes industriels avec leurs risques, leurs contraintes techniques, organisationnelles et administratives ; il dépend aussi du niveau de maturité des organisations et de prise de conscience vis-à-vis des problématiques associées ; enfin, il dépend également de l'angle et la hauteur de vue adoptés (vue plus ou moins macroscopique, organisationnelle ou technique, phases considérées, etc.). Si l'on revient au domaine du contrôle-commande, les aspects évoqués en Section 2.1 rappellent à quel point la sûreté constitue le véritable ADN de l'architecture. Dans ce contexte, une approche « unifiée » n'a aucun sens, et le rapprochement des cycles de vie doit respecter la priorité catégorique à la sûreté nucléaire : l'IEC 62859 (cf. Section 2.4) implique ainsi une approche avec une base pilotée par la sûreté, puis itérée selon les exigences de cybersécurité. Si la cybersécurité y trouve donc toute sa place, la sûreté y a le premier et le dernier mot.

### 4.2 La diversification des systèmes

La diversification des systèmes est une approche utilisée en sûreté, mais aussi en cybersécurité. Les systèmes de sûreté redondants peuvent être « diversifiés » pour atteindre un haut niveau de fiabilité, les redondances s'appuyant sur des technologies différentes, limitant les risques de défaillance simultanée (hors défaillance de cause commune). En cybersécurité, il est également fréquent de « doubler » certains dispositifs, comme les coupe-feu, en mettant en œuvre des technologies ou solutions de vendeurs différents, pour bénéficier de fonctionnalités complémentaires et s'assurer que l'éventuelle vulnérabilité d'un dispositif peut être compensée par l'autre.

D'une façon générale, ces redondances diversifiées sont employées « en parallèle » pour la sûreté, en redondance à chaud par exemple, tandis qu'elles le sont plutôt « en série » pour la sécurité. Mises en série, l'attaquant doit alors franchir une barrière après l'autre pour atteindre son objectif ; l'utilisation « en parallèle », classique en sûreté, s'avère contre-productive en cybersécurité : la surface d'attaque est doublée et l'attaquant n'a plus qu'à compromettre un seul des deux systèmes.

Cependant, même d'un point de vue strictement sécuritaire, la diversité n'apporte pas que des avantages : elle introduit de la complexité menant à des risques d'erreurs de configuration, à des problèmes de gestion, de maintenance, etc.

### 4.3 Les mises à jour logicielles

Dans les industries à risques, les systèmes de sûreté font l'objet de qualification, homologation, accréditation ou certification (la terminologie et les modalités varient selon les secteurs). Il s'agit de processus longs et coûteux, historiquement développés et mis en œuvre sur des systèmes sans ou à faible composante logicielle qui n'avaient pas vocation à évoluer rapidement. L'utilisation toujours plus intensive de logiciel dans ces systèmes, et la découverte de vulnérabilités associées changent la donne. Une mise à jour logicielle permettant de corriger une vulnérabilité peut également être vue comme une évolution remettant en cause la qualification (homologation/accréditation /certification), ce qui constitue un frein considérable pour sa mise en œuvre.

En pratique, il n'existe pas à ce jour de solution générale permettant de résoudre cet antagonisme. La modularité discutée précédemment (Section 3.4) peut limiter ces problèmes. Le manuel de référence AIEA [6] dédie une courte section au sujet, et conclut par la recommandation suivante :

> *« Pour limiter ces effets, on devra faire une distinction entre maintenance normale, évitant de tels processus, et les modifications du système exigeant de re-tester le système, ou même une re-certification dans le cas de systèmes critiques. Dans tous les cas, toute modification aux systèmes de sûreté ou liés à la sûreté, et aux systèmes de sécurité doivent être effectuées selon des procédures approuvées. »*

La mise en œuvre d'une telle recommandation implique une discussion technique approfondie, et un accord avec l'entité responsable de la certification notamment vis-à-vis des critères et des procédures associées. Indépendamment de l'éventuel impact sûreté, de telles procédures doivent inclure un processus d'arbitrage et d'acceptation de risque prenant en compte l'installation dans une vue globale, y compris économique : même dans les cas où la sûreté ne serait pas en cause, la mise à jour d'un équipement industriel n'est jamais neutre en cas d'arrêt non programmé, qui est à peser vis-à-vis du risque cybersécurité.

Dans les cas où une installation de mise à jour logicielle orientée cybersécurité n'est pas possible, des stratégies alternatives de sécurisation sont possibles, voire nécessaires : défense en profondeur et limitation des vecteurs d'attaque, amélioration de la surveillance des événements, etc. D'une façon plus générale, on voit aussi tout

l'intérêt de procéder à un durcissement initial important : désactivation des services réseau, désactivation des fonctions dangereuses, désinstallation des logiciels inutiles, autant de mesures qui réduisent les besoins de mise à jour.

# 5 Initiatives et retours d'expérience à surveiller

## 5.1 Un autre domaine précurseur : l'aéronautique

Un autre domaine précurseur sur la problématique des interactions sûreté-cybersécurité est celui de l'aéronautique, notamment de défense, naturellement au carrefour des enjeux de sûreté et de sécurité. Dès la fin des années 90, la *Royal Air Force* britannique et l'Université de York constatent la convergence des contraintes de sécurité et de sûreté s'exerçant sur les systèmes de la défense aérienne, et proposent des pistes pour harmoniser les approches sûreté et sécurité en termes d'analyses de risques et de spécification [12]. Depuis, le domaine, élargi au civil, n'a cessé de travailler sur le sujet, que ce soit via des projets de recherche (voir par exemple au niveau national, le projet SEISES) et via ses instances de standardisation (EuroCAE en Europe, RTCA outre-Atlantique). Les textes de référence sont les guides émis conjointement par ces instances (voir par exemple l'ED-202/DO-326) : cependant, malgré le chemin accompli, les travaux continuent pour améliorer encore la coordination.

## 5.2 Les autres secteurs

Si le nucléaire et l'aéronautique sont déjà bien engagés dans un traitement structuré autour de textes internationaux, d'autres industries se penchent aussi, à des degrés divers, sur la problématique. On peut notamment citer l'industrie de la défense [13], le transport ferroviaire [14] ou l'automobile [15]. Ces travaux restent pour la plupart encore prospectifs, et n'ont pas encore convergé vers des documents normatifs. On peut toutefois signaler les initiatives du comité 99 de l'ISA[6] et du comité technique 65 de l'IEC, qui ont tous deux créé des groupes de travail dédiés à l'étude de la problématique, en vue de produire pour le premier, cadrer pour le second, des rapports techniques ou des normes, selon le niveau de consensus et la solidité industrielle des résultats obtenus. Un système de « liaison » est d'ailleurs institutionnalisé avec le comité normatif en charge de la rédaction de l'IEC 62859 pour le nucléaire, discutée en Section 2.4.

## 5.3 Une recherche très active sur la thématique

Le sujet de la coordination entre sûreté et cybersécurité est aujourd'hui bien identifié comme important pour une gestion de risque optimale et comme facteur de résilience pour les installations industrielles. Si les secteurs les plus concernés ont déve-

[6] *International Society of Automation*

loppé ou développent des référentiels regroupant principes faisant consensus et bonnes pratiques, de nombreux défis restent à relever. Aussi, un effort croissant de recherche industrielle et académique est logiquement investi sur cette problématique. Un état de l'art des différents travaux de recherche en cours est donné par la référence [16]. Un grand nombre des initiatives recensées s'appuient sur des analyses outillées et des modèles, souvent graphiques, semi-formels voire formels (i.e. avec une définition mathématique rigoureuse) : ce type d'approches fait partie des boîtes à outils actuelles du domaine de la sûreté, et de la cybersécurité ; elles doivent encore mûrir et trouver leur place pour une prise en compte opérationnelle des interactions entre sûreté et cybersécurité. Ceci-dit, elles constitueront à terme des outils précieux, compléments logiques aux documents de principes et référentiels prescriptifs en cours de développement. Le développement et l'exploitation de systèmes soumis à des contraintes de sûreté et de cybersécurité pourront bénéficier *in fine* de ces deux types d'approches.

## 6 Conclusion

À l'heure où exigences de sûreté et de cybersécurité convergent sur les systèmes numériques industriels, il est nécessaire de maîtriser leurs interdépendances. Sûreté et cybersécurité sont intimement liées, se renforçant ou se conditionnant l'une l'autre dans certains cas, s'opposant dans d'autres. Dans cette communication, nous avons présenté un certain nombre de principes et de retours d'expérience, tirés de la conception des systèmes contrôle-commande des centrales nucléaires, visant à maîtriser cette convergence. Le bénéfice des technologies numériques ne pourra être pleinement exploité que si les aspects de sécurité informatique qui y sont associés s'intègrent harmonieusement avec les exigences et les référentiels de sûreté des installations industrielles. Au-delà de ces quelques principes issus du domaine nucléaire, de nombreuses initiatives sont en cours, aussi bien chez les autres industriels que dans la communauté scientifique. Le développement de documents sectoriels pertinents (tels que ceux produits par l'AIEA et par l'IEC pour le nucléaire) constituent une étape clé dans le traitement de cette problématique. Demain, des outils d'évaluation et de modélisation adaptés, aujourd'hui du domaine de la R&D, viendront sans doute aussi épauler les industriels pour affiner les analyses et optimiser les arbitrages de conception et d'exploitation.

## 7 Références

[1] Hollnagel, E., Pariès, J., et Wreathall, J. (2011). Resilience Engineering in Practice: A Guidebook. Ashgate. ISBN 978-1409410355.

[2] IEC 61513 (2011) Ed. 2.0, "Nuclear power plants – instrumentation and control for systems important to safety – general requirements for systems"

[3] IEC 60880 (2006) Ed. 2.0, "Nuclear power plants – instrumentation systems important to safety – software aspects of computer based systems performing category A functions"

[4] “La défense en profondeur appliquée aux systèmes d’information,” Memento du SGDN/DCSSI, juillet 2004, version 1.1.

[5] « Cybersécurité des installations industrielles : Défendre ses systèmes numériques », Cépaduès Éditions, ouvrage collectif dirigé par Y. Fourastier et L. Pietre-Cambacedes, 2015.

[6] Agence Internationale de l’Énergie Atomique (AIEA), « La sécurité informatique dans les installations nucléaires », Collection Sécurité nucléaire de l’AIEA – N° 17, Manuel de référence, STI/PUB/1527, 2013 (édition en anglais publiée en 2011)

[7] IEC 62645 (2014) Ed. 1.0, “Nuclear power plants – Instrumentation and control systems – Requirements for security programmes for computer-based systems”

[8] IEC 61226 (2009) Ed. 3.0. “Nuclear power plants – Instrumentation and control systems important to safety – Classification of instrumentation and control functions”

[9] IEC 62859 (to be published in 2016) “Nuclear power plants – Instrumentation and control systems – Requirements for coordinating safety and cyber security”

[10] IEC 62443 (multi-part standard), “Industrial communication networks - Network and system security”; 2009-2015 (en cours d’évolution/revision)

[11] T. Novak, A. Treytl, “Functional safety and system security in automation systems - a life cycle model”, EFTA’08, Hambourg, Allemagne, p. 311–318,. 2008.

[12] D. P. Eames, J. Moffett, “The integration of safety and security requirements,” actes de SAFECOMP’99 (18th International Conference on Computer Safety, Reliability and Security), Springer LNCS1698, Toulouse, France, Sep. 1999, p. 468–480.

[13] A. Derock, P. Hebrard, et F. Vallée, “Convergence of the latest standards addressing safety and security for information technology,” Actes en ligne d’ERTS² 2010 (Embedded Real Time Software and Systems), Toulouse, France, mai 2010.

[14] J. Caire. Vers un cycle de vie de Sécurité globale pour les systèmes informatiques industriels. Actes du 19e congrès sur la sûreté de fonctionnement de l’IMdR, Dijon, Oct. 2014.

[15] B. Glas et al., Automotive Safety and security integration challenges, Proc. Automotive Safety & Security, 2015.

[16] S. Kriaa, L. Pietre-Cambacedes, M. Bouissou, Y. Halgand, "A survey of approaches combining safety and security for industrial control systems", Reliability Engineering & System Safety,Volume 139, p.156–178, juillet 2015

# SCADA Safety and Security joint modeling (S-cube): case study of a dam

Siwar Kriaa[1,2], Marc Bouissou[1], Youssef Laarouchi[1]

[1]Electricité De France, Clamart, France,{siwar.kriaa, marc.bouissou, youssef.laarouchi}@edf.fr
[2]CentraleSupélec, Châtenay-Malabry, France, {siwar.kriaa}@centralesupelec.fr

## 1 Introduction

Modern Industrial Control Systems (ICS) offer the necessary means to control and supervise critical infrastructures. In order to facilitate supervision and control of the industrial process and to reduce system exploitation cost, modern ICS are increasingly integrating new information and communication technologies (ICT) and migrating towards standardized and open protocols (e.g. Modbus TCP, OPC UA). This trend induces more complexity in ICS and exposes them to cyber-attacks that exploit vulnerabilities in ICT. Such attacks can reach some critical components within the system and alter the normal functioning of the industrial process causing adverse consequences on the safety of the system and its environment [1][2].

We adopt in this article the following definitions: safety is associated with accidental risks originating from the system that could result in unacceptable consequences (on the system itself, its users or its environment) while security is related to malicious risks and especially cyber-attacks.

For critical systems having safety issues and integrating new information technologies, safety and security risks converge and can have mutual interactions [3]. A joint risk analysis covering both for safety and security aspects is necessary to identify their interactions and conditions an optimal risk analysis.

The S-cube (SCADA (Supervisory Control and Data Acquisition) Safety and Security modeling) approach [4] is a model-based risk analysis framework for SCADA-based industrial architectures. The S-cube approach enables modeling the industrial system architecture with the associated safety and security aspects. It automatically generates the possible risk scenarios that can lead to safety-related issues (human loss, environment pollution, system damage).

We propose in this paper to apply the S-cube approach to a real case study: a pumped storage hydroelectric plant, in order to show how this approach enables to easily consider different hypotheses about the system architecture and generates automatically the qualitative and quantitative results. We particularly underline the ability of this approach to assess the system resilience to external malevolence and particularly cyber-attacks and to support decision making.

The paper is structured as follows: Section 2 presents the S-cube approach. Section 3 describes the use case and Section 4 gives the risk analysis resulting from the appli-

cation of S-cube to the use case. Section 5 concludes this work and gives perspectives.

## 2 The S-cube approach

S-cube is a model-based approach for SCADA safety and security joint modeling. It generates automatically safety-related risk scenarios (sequences of accidental events; e.g. components failures) and security-related risk scenarios (sequences of attack steps) likely to happen on a given architecture and that result into safety issues (an undesired event identified by safety analyses). S-Cube can be used either in the design phase of new safe and secure systems or in the operational phase for existing systems to optimize and master their safety and security. In the seminal paper on S-cube [4], we exposed our main conclusions on the state of the art we made in [5] and the rationale for developing this new approach. We have also summarized the contents of the S-cube knowledge base and sketched its use on a simple use case.

The S-cube approach, described in Figure 1, takes as input the system architecture, modeled by the user using a graphical interface (the KB3 HMI [6]), and gives as output the attack and failure scenarios likely to happen on it, that may lead to a given undesirable event related to safety issues.

The S-cube approach relies on a knowledge base (S-cube KB), that gathers expertise on ICS and particularly SCADA systems and their associated safety and security aspects. The S-cube KB can be seen as a Domain Specific Language that enables to describe the typical components of digital industrial infrastructures and the related security attributes (authentication, access control, redundancy). Each component is associated with attacks and failure modes that can happen on it. The generic models of the S-cube KB are then instantiated on the system architecture, given as an input. The textual model obtained is processed by calculation engines (Figseq or Yams) that generate automatically attack and failure scenarios, with an estimation of their probabilities (cf. Section 4).

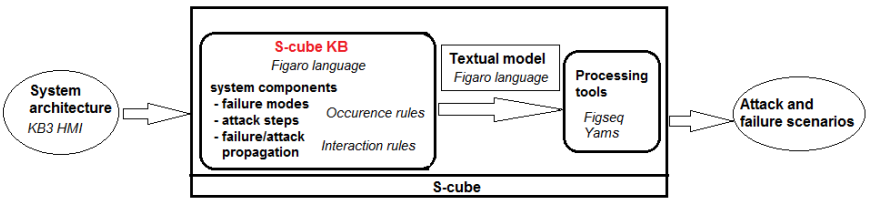



**Fig. 1.** The S-cube approach

The S-cube KB is written in Figaro language [7], an object oriented modelling language that is based on two kinds of rules: occurrence and interaction rules. The occurrence rules are used to model security-related (attacks) and safety-related (failures) events that may happen to each system component. These rules include the probability distribution of the time after which the event will happen. The interaction rules

model the propagation of instantaneous effects (e.g. compromised components, cascading failures) into the overall system architecture. The processing mechanism uses both occurrence and interaction rules in order to automatically generate the risk scenarios leading to the undesirable event.

The S-cube approach enables to model the different industrial architecture levels [8]:

- The field level: involving system components that enable sensing (sensors) and manipulating (actuators) the physical process;
- The process control level: involving system components that enable data acquisition and processing as well as controlling the physical process (e.g. Remote Telemetry Units (RTU), Programmable Logical Controllers (PLC));
- The supervision level: involving system components that enable the overall process monitoring and supervision (e.g. SCADA servers);
- The corporate level: involving system components that enable business-related activities.

We make the following assumptions in the S-cube KB for modelling SCADA systems.

- We differentiate between physical machines and software components running on them. This assumption enables to have a sufficient level of detail in which failures and attacks like physical access are associated with the physical machines and cyber attacks exploiting vulnerabilities are associated to the software component already containing some vulnerabilities;
- For IT level networks such as the corporate network, we are interested in the attack propagation (multistage multi-hop) between different IT level machines until reaching some component having a control action on the process. When reaching the control network, we are rather interested in data integrity/availability (wrong/not available) as the modification or unavailability is generally the main reason leading to undesirable events.

More details on the S-cube KB metamodel and content are given in [4].

The implementation of the S-cube approach, indicated in Figure 1 in Italic font, relies on the KB3 workbench [6], a set of Figaro based tools initially designed for dependability analysis. The KB3 Human Machine Interface enables the user to input the system architecture using the graphical items associated to the different components defined in the S-cube KB. The S-cube KB is next instantiated on the system description and a new Figaro textual model is generated. This model is next processed by quantification tools Figseq or Yams. Yams is a Monte Carlo simulator; Figseq is a tool for reliability and availability calculation of systems based on the exploration and quantification of sequences going from the initial state of the system to a failure state [9]. These tools enable to generate automatically attack and fault scenarios based on the occurrence rules described in the knowledge base. The scenarios are sorted by decreasing occurrence probability.

We propose in the next section to illustrate the S-cube approach on a real-world case study in order to show its potential to model complex systems and assess safety and security related risks

## 3 Case study

This case study is inspired from the Taum Sauk pumped storage plant. In 2005, a famous accident happened at this installation; it resulted in the destruction of a section of the upper reservoir and the sudden release of a large volume of water down the slopes. We give first a quick overview on the Taum Sauk accident; we describe next the system architecture (thanks to the accident and the subsequent analyses, it is relatively easy to find detailed information on the system design); we finally show the ability of the S-cube approach to yield a holistic analysis encompassing safety and security risks on such a system.

### Taum Sauk upper reservoir failure

The Taum Sauk hydroelectric Power Plant is a pumped storage hydroelectric power station; an example of such an installation is given in Figure 2. It consists of two water reservoirs: an upper reservoir and a lower reservoir separated by a penstock. In high electricity demand hours, water is released from the upper reservoir to the lower reservoir in order to generate electricity. In low electricity demand hours, water is pumped back from the lower reservoir to the upper reservoir for energy storage.

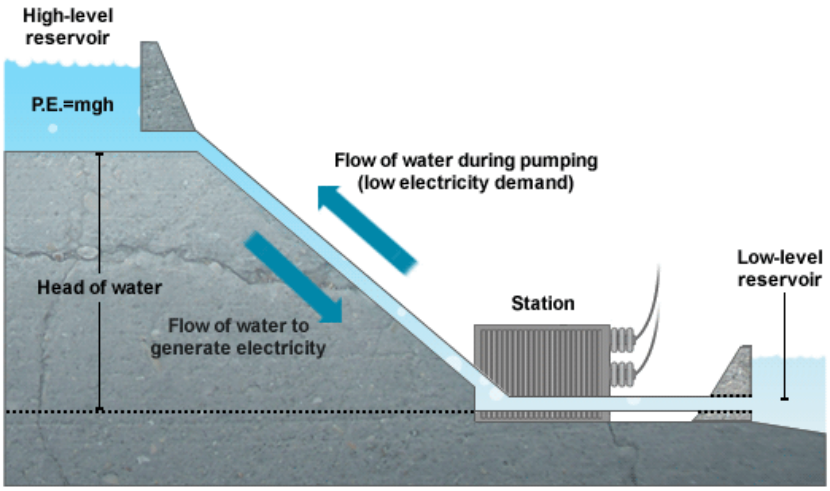



**Fig. 2.** A pumped storage hydroelectric power station

On December 2005, the Upper Reservoir of the Taum Sauk Pumped Storage Project was overtopped when water continued to be pumped from the lower reservoir to

the upper reservoir. This led to the failure of a section of the upper reservoir embankment and the sudden and complete evacuation of water. This breach of the upper reservoir had catastrophic consequences: flooding of the surrounding areas including Highways, campgrounds and adjacent properties.

The Taum Sauk upper reservoir breach was caused by a purely accidental failure of the instrumentation used for monitoring the reservoir level. In addition, the initial design of the system architecture was not respected at the implementation phase which decreased the system resilience and accelerated the upper reservoir failure.

In our study, we take the same control architecture as the one described for the Taum Sauk upper reservoir and we try to assess its resilience to cyber attacks. Indeed, we demonstrate in this paper that this undesirable event could as well be the result of a malicious cyber-attack.

**Description of the use case**

The system architecture, depicted in Figure 3, was inspired and built from information collected on the instrumentation and control system of the Taum Sauk Upper Reservoir [10][11][12].

The instrumentation system is composed of two sets of sensors:

- Three primary Pressure sensors, placed at a given elevation, which convert the pressure into water level. These measures are used to monitor the reservoir level and hence trigger the shutting down of the pumps/generators units;
- Four conductivity sensors placed in pairs above and below the highest and the lowest water level. These sensors are activated if the water reaches the level at which they are placed. The sensors *HI* and *HI-HI* determine whether the water level in the upper reservoir is too high. The sensors *LO* and *LO-LO* determine whether the water level is too low.

The control system relies on two main Programmable Logic Controllers (PLC): the Common PLC and the Upper Reservoir (UR) PLC. The Taum Sauk project was remotely controlled through a microwave system from the Osage Plant, under the direction of the load dispatching in St. Louis [11]. The dispatching control center provides generating Megawatt instructions (with pump start/stop mode) to the Operator control center at the Osage plant; from there, the operator remotely controls the Taum Sauk units (two reversible pumps) at the Lower Reservoir.

The primary pressure sensors send the water level measures to both the Common and the UR PLCs. These measures are transmitted to the Operator and dispatching control center. The HI and the HI-HI sensors are placed at two different elevations and used for emergency shutdown should extremely high water levels occur.

In its normal operation, the plant is controlled by pressure sensors. The average value of the three readings is considered by the controllers. In the pumping mode, the pumps are activated in order to stock the water in the upper reservoir. If for some reasons the pressure sensors did not operate correctly, the HI sensor would be reached which should activate the automatic shutdown of the pumps. If for some reasons, the pumping mode was not terminated, the HI-HI sensor would be encountered which activates a hard emergency stop of the pumps.

The HI and the HI-HI sensors are connected, with separate conduits, each one to a different PLC: the Hi sensor is connected to the Common PLC while the Hi-Hi sensor is connected to the UR PLC. This redundancy built in the design forms a safety mechanism.

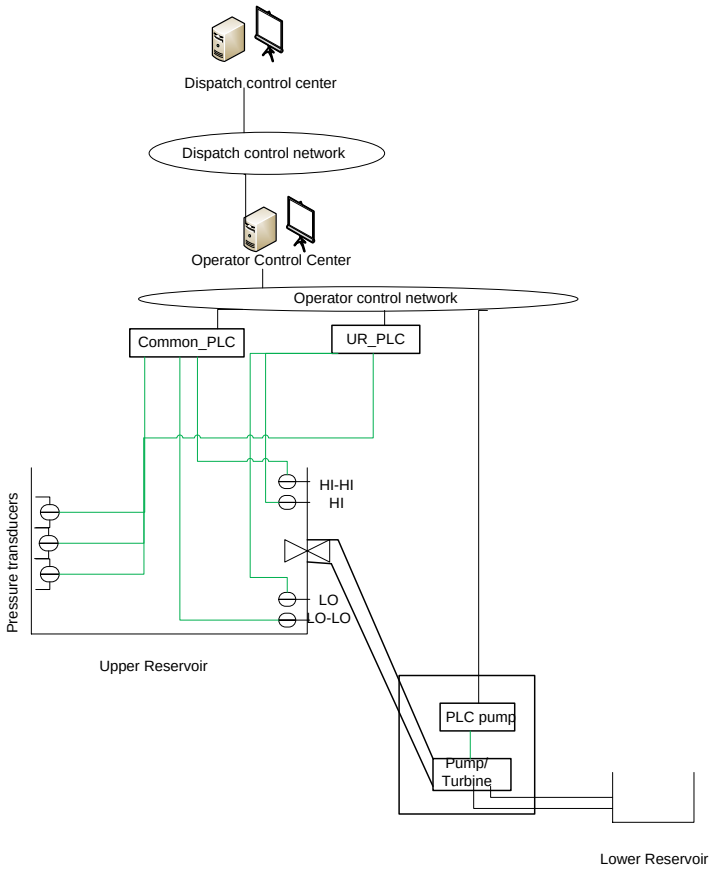



**Fig. 3.** The system architecture

### Modeling the system with the S-cube approach

The system architecture given in Figure 3 is modelled with the S-cube approach. The graphical model, as input in the KB3 HMI is given in Figure 4.

The primary pressure sensors are modeled with the physical components "*pressure1*", "*pressure2*" and "*pressure3*". The software components associated to each sensor, that enable to convert and send the measures, are named *P1*, *P2* and *P3*. Con-

cerning the conductivity sensors, we model only the high level sensors *S_HI_HI* and *S_HI* and the associated software, named *HI_HI* and *HI*.

The three water pressure sensors send their readings to the *k_n_gate_measure*, which is a 2/3 voter. If two readings out of the three are coherent, the measure will be sent to both software components *Common_water_level* resp. *UR_water_level* of the *Common_PLC* resp. the *UR_PLC*. If water level reaches the HI sensor, an alarm is sent to the *Common_HI* software of the *Common_PLC* which sends an instruction to *PLC_pump*; used to control the *pumping_unit*; to shut down the pump. If for some reason, the pump is not shut down, the water would reach the HI-HI sensor level and an alarm would be sent to the software *UR_HI_HI* of the *UR_PLC*. The *UR_HI_HI* would then send an instruction to *PLC_pump_soft* to shut down the pump. All feedback from the *UR_PLC* and the *Common_PLC* is reported to the *operator_control_center*. The *k_n_gate_feedback* is a 1/4 voter: if *operator_soft* receives one feedback (among the four input data flows) indicating the highest water level value, it would send an instruction to *PLC_pump* to shut down the *pumping_unit*. The *k_n_instruct* is a 1/3 voter; i.e. if the *PLC_pump_soft* receives one instruction (either from the operator or from the *UR_PLC* or from the *Common_PLC*), it would immediately shut off the pump.

The SCADA software (*operator_soft*) running on the operator control center sends feedback on process status to SCADA software (*dispatch_control_soft*) running on the dispatch control center, and receives back instructions on the Megawatts to generate.

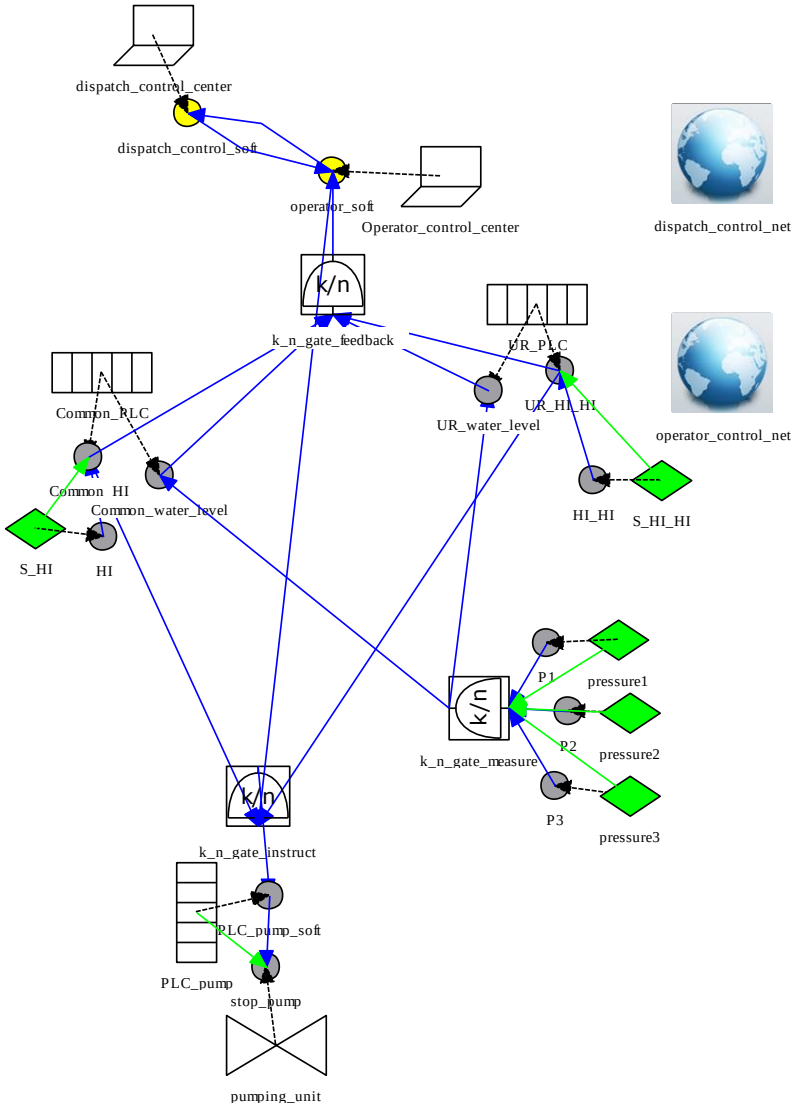



**Fig. 4.** The designed system architecture (as input in the KB3 HMI)

## 4 Risk analysis of this installation

After the Taum Sauk Upper Reservoir breach, incident reports have shown that the real system architecture, when the breach was produced, did not respect the initial design. The redundancy between the conductivity sensors was not respected in the

implementation phase and resulted in the safety system not activating upon the failure of one of the conductivity sensors.

Safety mechanisms including voters and redundant systems have been included in the system design in order to ensure its availability and reliability. These mechanisms are initially conceived to make the system dependable and operational in case of accidental failures of components. However, is the system able to resist and survive cyber-attacks which are less predictable?

Using the S-cube approach, we provide in this Section a risk analysis of both system architectures: the designed and the real architecture as implemented at the moment of the breach. We assess safety and security risks related to each architecture and evaluate the impact of the non-respect of the initial design on the probability of the breach and the system resilience.

**First use case: the designed system**

The designed system architecture is graphically modelled with the KB3 HMI as shown in Figure 4. We take the following assumptions for this model based on the description given in Section 3.2:

- The operator control network that connects the PLCs and the operator control center use a wireless communication technology;
- No authentication mechanism is used to access the operator control network.

These assumptions are included in the graphical model given in Figure 4. We also assume that the mean time to failure for the different physical components and the mean time to success for the different attack steps are exponentially distributed. The relevance of using the exponential distribution for security modelling is discussed in [13]. We give in Table 1 some of the quantitative parameters chosen for failures and attacks considered. Security parameters are mere estimations by security engineers of the time scale needed by an attacker to accomplish a given attack step (days/weeks/months). Therefore the results given below must be taken just as an illustration of what could be done with more refined data.

| **Attack/failure** | **MTTS/MTTF (hour)** |
|---|---|
| Physical component failure | MTTF= 1e5 |
| Jamming attack for a wireless network | MTTS = 1e6 |
| Accessing a network with no authentication | MTTS= 1e2 |
| Modify data sent from a compromised controller | MTTS= 1e2 |

**Table 1.** Estimation of some safety/security parameters

Using the S-cube knowledge base and the quantification tool Figseq [9], we process the graphical model and generate automatically the risk scenarios related to this architecture. We consider that the undesirable event leading to the Upper Reservoir breach happens if the action of stopping the pumping unit cannot be executed. This potentially leads to water overtopping the reservoir and causing erosion of the rockfill embankment and consequently its collapse under the water load. This hypothesis is

pessimistic as the malfunction of an actuator in real circumstances may not be sufficient to create a safety-related risk.

The risk scenarios generated are given in a table and sorted by their decreasing contribution to the overall undesirable event probability (c.f. Table 2). Each scenario is composed of one or many transitions, the rate of which is also indicated in the table. We can see that attack scenarios are the most likely to happen. This is due to the fact that failure rates are generally low compared to attack rates. Indeed, once determined in making his attack, the attacker spends hours/days to try an attack step. On the other hand components failures take several years before happening. We focus consequently on the most probable attack paths likely to happen on the system.

Based on the hypothesis taken for types of failures and attacks considered [4], and the corresponding failure rates and mean times to success, the quantitative analysis yields the following results: over 100 hours of operation, the probability of the breach is 1.9e-4. The first two attack scenarios likely to happen are given in Table 1.

| N° Scn. | Transitions | | Pr. | Ctrib. |
|---|---|---|---|---|
| | Name | Rate | | |
| 1 | Jamming attack(operator control net) | 1e-6 | 4.3e-5 | 2.1e-1 |
| 2 | access(operator control net) | 0.01 | 2.6e-6 | 1.3e-2 |
| | Compromise communication link(Operator control center) | 0.01 | | |
| | Send false/no instructions(operator soft) | 0.01 | | |
| | Compromise communication link(Common PLC) | 0.01 | | |
| | Send false/no instructions to actuator(Common HI) | 0.01 | | |
| | Compromise communication link(UR_PLC) | 0.01 | | |

**Table 2.** The most probable attack scenarios related to the designed system architecture

We can see that the first attack likely to happen is jamming the operator control network. This attack is specific kind of wireless networks and aims at interfering with the other legitimate communications (e.g. by continuously emitting a radio signal, sending regular packets, etc.) in order to deny their access to the medium. This results in the unavailability of all data supposed to be sent on the network and particularly instructions to stop the pumping unit; which would lead to the undesirable event. This attack remains however with a low probability of occurrence (4.3e-5) as it requires for the attacker to have special equipment and to be at the vicinity of the network access point.

The second attack scenario generated consists of seven attack steps. In the first step, the attacker accesses the operator control network. This attack step is likely to happen as the operator control network is wireless (e.g. GSM) and does not exert any authentication mechanism. Having access to the network the attacker can compromise any communication. The attack step *compromise_communication_link* models a man-in-the-middle attack in which the attacker impersonates the legitimate sender of data.

This enables the attacker to falsify or deny data flows sent on the compromised communication.

In order to compromise the instruction sent to *PLC_pump* to disable stopping the pumping unit the attacker must compromise instructions sent by the operator control center and both instructions sent by the *Common_HI* and the *UR_HI_HI* software components. This takes a lot of effort from the attacker and reduces his chances of achieving the attack. We infer that the redundancy between the HI and the HI-HI sensors, which consists in a safety mechanism initially designed to reinforce the system dependability, enables also to reinforce the system resilience against malicious attacks and cyber malevolence.

We propose in the next section to make the same analysis on the real system architecture in order to assess the impact of the implementation modifications on the system resilience to cyber-attacks.

**Second use case: the real system**

After the Taum Sauk Upper Reservoir breach, it was discovered that the conductivity sensors were tied in series rather than in parallel [12]. This implies that it was required that both HI and HI-HI sensors should be activated in order for the emergency pumps shutdown to be activated. This condition was never reached as the sensors were, additionally, detached from their mountings and were not indicating the true elevations of the reservoir.

We include these modifications in the system architecture in order to assess their impact on the overall system safety. The new system architecture is given in Figure 5. We place here two “AND” gates: before the two feedback signals sent for the operator and before the emergency shutdown instructions sent to the *PLC_pump_soft*. This latter should execute the pump shutdown if it receives an instruction form the *operator_soft* of from both PLCs.

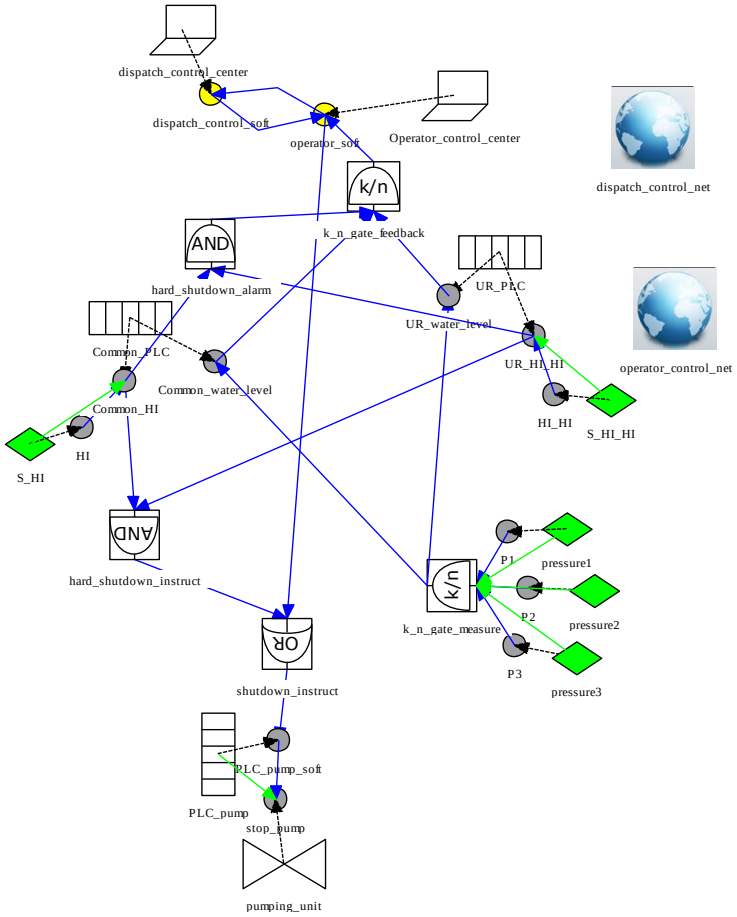



**Fig. 5.** The implemented system architecture

Thanks to the robustness of the S-cube approach, these changes in the system architecture have been easily added on the first model already built in Figure 4, without the need to revamp the whole model. This new architecture is processed again, as explained in § 0 and the results obtained are the following: over 100 hours of operation, the probability of the breach is 2.4e-2 (100 times higher than the breach probability for the designed architecture). The first two attack scenarios likely to happen on this new architecture are given in Table 3.

| N° Scn. | Transitions | | Pr. | Ctrib. |
|---|---|---|---|---|
| | Name | Rate | | |
| 1 | access(operator control net) | 0.01 | 6.7e-4 | 2.5e-2 |
| | Compromise communication link(Operator control center) | 0.01 | | |
| | Send false/no instructions(operator soft) | 0.01 | | |
| | Compromise communication link(Common PLC) | 0.01 | | |
| | Send false/no instructions to actuator(Common HI) | 0.01 | | |
| 2 | access(operator control net) | 0.01 | 6.7e-4 | 2.5e-2 |
| | Compromise communication link(Operator control center) | 0.01 | | |
| | Send false/no instructions(operator soft) | 0.01 | | |
| | Compromise communication link(UR PLC) | 0.01 | | |
| | Send false/no instructions to actuator(UR HI HI) | 0.01 | | |

**Table 3.** The first attack scenarios likely to happen on the system architecture given in **Fig. 5**

We can see that the jamming attack is no more the easiest scheme for the attacker but rather the attack starting by accessing the operator control network which is the more likely to happen. This attack consists of only five attack steps instead of seven attack steps in the previous case. With the real system architecture implemented, the attacker needs just to compromise either the *Common_HI* instruction or the *UR_HI_HI* instruction, in addition to compromising the instruction sent by the operator control center.

Instead of having two separate systems for emergency shutdown for true redundancy, the real system architecture included only one system. This reduces the system resilience to cyber-attacks as it simplifies the attacker's task and increases the probability of successful attacks.

Using the S-cube approach, we have been able to put into evidence how safety mechanisms reinforce the system's security. Comparing risk analyses of both system architectures, we infer that the system resilience is further undermined by the non-compliance with designed system architecture and lack of maintenance, which resulted in the catastrophic failure of the Taum Sauk Upper Reservoir.

## 5 Conclusion

This paper showed how the S-cube approach provides an easy and robust risk analysis framework for modeling industrial information and control architectures and assessing the related safety and security risks. By applying it on a real case study inspired from the Taum Sauk Upper Reservoir breach, we demonstrated how the S-cube approach enables to capture safety and security interdependencies and assess the

system resilience against accidental and malicious risks. This approach aims to identify the system weaknesses and vulnerabilities in order to support risk management and decision making.

The S-cube approach can be used by safety and security engineers in different phases of the system lifecycle:

- Design phase to help designing new safe and secure systems with the appropriate safety and security requirements. S-cube helps the designer to assess and compare different configurations and safety/security mechanisms before converging towards an architecture where safety and security requirements are coordinated;
- Operation phase to assess the risk on existent systems, to help defining new safety and security mechanisms and to support crisis management. Indeed, S-cube helps risk analysts identify vulnerable components in the system architecture, predict undesirable events and avoid paths leading to them by choosing the appropriate patch and defense strategy.

Future work will focus on enhancing the quantitative data associated to the different attacks and failure modes modelled in the S-cube knowledge base, in order to better model the reality.

## References


[1] Pauli, "Hackers pop German steel mill, wreck furnace," 22-Dec-2014. [Online]. Available: http://www.theregister.co.uk/2014/12/22/hackers_pop_german_steel_mill_wreck_furnace/.

[2] *Hijacking airplanes with an Android phone*. .

[3] L. Piètre-Cambacédès, "Des relations entre sûreté et sécurité," Télécom ParisTech, 2010.

[4] S. Kriaa, M. Bouissou, and Y. Laarouchi, "A Model Based Approach for SCADA Safety and Security joint Modeling: S-cube," in *IET System Safety and Cyber Security*, Bristol, 2015.

[5] S. Kriaa, L. Pietre-Cambacedes, M. Bouissou, and Y. Halgand, "A survey of approaches combining safety and security for industrial control systems," *Reliab. Eng. Syst. Saf.*, vol. 139, pp. 156–178, Jul. 2015.

[6] M. Bouissou, "Automated dependability analysis of complex systems with the KB3 workbench: the experience of EDF R&D," in *Proceedings of the International Conference on ENERGY and ENVIRONMENT, CIEM 2005*, 2005.

[7] M. Bouissou, H. Bouhadana, M. Bannelier, and N. Vilatte, "Knowledge modeling and reliability processing: presentation of the FIGARO language and of associated tools," in *proceedings of SAFECOMP 91*, Trondheim, Norway, 1991.

[8] International Electrotechnical Commission, *Enterprise-control system integration. Part 1, Part 1,*. 2013.

[9] M. Bouissou and Y. Lefebvre, "A path-based algorithm to evaluate asymptotic unavailability for large Markov models," in *Reliability and Maintainability Symposium, 2002. Proceedings. Annual*, 2002, pp. 32–39.

[10] J. David Rogers Conor M. Watkins, "Overview of the Taum Sauk Pumped Storage Power Plant Upper Reservoir Failure, Reynolds County, Mo.," presented at

the 6th International Conference on Case Histories in Geothechnical Engineering, Arlington, VA, 2008.
[11] Before the public service commission, state of Missouri, "Staff's initial incident report.," Case No. ES-2007-0474, Oct. 2007.
[12] FERC Taum Sauk Investigation Team, "Report of Findings on the Overtopping and Embankment Breach of the Upper Dam - Taum Sauk Pumped Storage Project," FERC No. 2277, Apr. 2006.
[13] L. Piètre-Cambacédès and M. Bouissou, "Beyond Attack Trees: Dynamic Security Modeling with Boolean Logic Driven Markov Processes (BDMP)," in *Dependable Computing Conference (EDCC), 2010 European*, 2010, pp. 199–208.

# Expression des besoins et identification des objectifs de resilience

Sylvain Conchon (CONIX), Jean Caire (RATP)
sylvain.conchon@conix.fr, jean.caire@ratp.fr

**Résumé** Aujourd'hui, les systèmes d'information industriels, notamment ceux des OIV, doivent non seulement garantir un haut niveau de fiabilité, mais aussi maintenir les capacités essentielles à la réalisation de leurs missions malgré des actes de malveillance. Cette communication présente une approche pour définir et mettre en œuvre la résilience sur des systèmes d'information. Elle est fondée sur la gestion des risques, en s'inspirant des principes reconnus de la méthode EBIOS. Cette démarche se décompose en quatre phases : définition des besoins de résilience, à partir de l'analyse des missions supportées par le système ; identification des scénarios de risque, pour tous modes d'agression ; détermination d'objectifs de résilience couvrant à la fois les exigences de sûreté de fonctionnement et de cybersécurité ; élaboration d'une stratégie globale permettant d'atteindre ces objectifs.

**Mots-clefs :**
Systémigramme de sûreté de fonctionnement et cybersécurité • Besoins de résilience • Scénarios de risque • Objectifs de résilience • Stratégie de résilience

## Préambule

En autres définitions, on peut aborder la résilience d'un système numérique comme la capacité d'adaptation de celui-ci à réaliser ses missions en situation d'agression. On généralise ici le terme « agression » qui est associé naturellement au champ de la malveillance mais peut être appliqué à la Sûreté de Fonctionnement pour désigner l'ensemble des actions et transformations générant in fine une atteinte à un élément sensible, dont on cherchera à évaluer la gravité. Cette terminologie est issue du concept RATP de Défense en Profondeur ([1]), développé initialement pour la sûreté de fonctionnement, puis étendu à certains problèmes de sécurité des biens et des personnes.
On remarque aussi que les définitions de la sûreté de fonctionnement ([2]) et de la cybersécurité ([3]), emploient la notion de risque. Mieux encore, les deux partagent une même idée d' « absence de risque inacceptable ».

## Systémigramme résilience / risque

La filiation entre le concept de risque (tous domaines confondus) et la notion de résilience est naturelle : le risque modélise et propose une représentation analytique des phénomènes susceptibles d'impacter le système étudier et de provoquer des événements redoutés Cette similitude nous permet de concevoir une approche globale de résilience enveloppant la sûreté de fonctionnement et la cybersécurité. Toutefois, si les principes généraux sont analogues, chaque domaine définit et met en œuvre une stratégie propre. (voir Tableau 1). La démarche de résilience devra donc hériter de ces approches.

| | SURETE DE FONCTIONNEMENT | CYBERSECURITE |
|---|---|---|
| **Définition** | Absence de risque inacceptable | Plusieurs définitions, dont l' absence de risque inacceptable |
| **Evénement redouté** | Défaillance majeure | Evénement qui permet à l'adversaire d'imposer sa volonté |
| **Equation du risque** | R = f ( faute, défaillance) | R = f ( menace , vulnérabilité , conséquence ) |
| **Scénarios de risque** | Les événements élémentaires sont combinés par des règles d'inférence logique exprimant les lois de la physique | Le scénario est pensé avant d'être mis en œuvre, les événements élémentaires sont combinés par :<br>• des règles logiques exprimant l'adéquation capacités (menace) / vulnérabilité (cible)<br>• un processus de décision exprimant l'adéquation motivations (menace) / conséquence (cible) |
| **Méthodes & outils pour construire les scénarios** | Modèle de référence:<br>Faute→Erreur…<br>Erreur→Défaillance<br>Outils : Analyse des Modes De Défaillances et de leurs Effets, Arbres de défaillance | Pas de modèle de référence.<br>Plusieurs outils notamment sur les arbres / graphes d'attaque |
| **But de la Stratégie** | Maîtriser les défaillances | Gagner la confrontation avec l'adversaire |
| **Objectifs opérationnels de la Stratégie** | Réduire le produit probabilité d'occurrence x gravité pour chaque évènement redouté<br>La stratégie doit prendre en compte la multiplicité des scénarios. | Diminuer le ratio coût / bénéfice de l'attaquant pour chaque attaquant La stratégie doit prendre en compte la diversité des adversaires et la multiplicité des scénarios. |
| **Tactiques** | Prévention des Fautes, Tolérance aux Fautes, Suppression des Fautes ; Prévision des Fautes | Pas de modèle de référence. |

**Tableau 1 : comparaison des domaines**

L'intégration de ce modèle dans le diagramme systémique (voir Figure 1) de la résilience montre les analogies entre la cybersécurité et la sûreté de fonctionnement, mais aussi les spécificités de la cybersécurité :

- Le fait que les adversaires soient intelligents, et qu'ils peuvent disposer d'une stratégie élaborée
- Le fait que l'évolution du système résulte de la confrontation avec un adversaire qui s'adapte

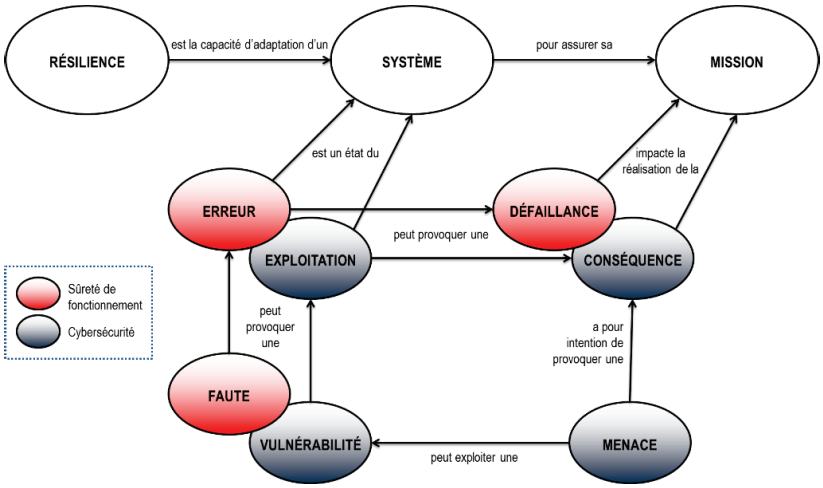



**Figure 1 : diagramme systémique de la résilience**

A ce stade des travaux, on dispose d'une décomposition analogue des risques de sûreté de fonctionnement et de cybersécurité au service de la résilience du système. La méthode EBIOS, issue du domaine de la cybersécurité, se positionne comme le candidat naturel pour rendre le modèle opérant : « La spécificité d'EBIOS réside dans sa souplesse d'utilisation. […] les activités […] seront adaptées à l'usage désiré. » ([4]). La présente proposition de communication vise à exposer les travaux sur l'extension des principes de la méthode EBIOS dans le cadre d'une méthode spécifique pour l'Expression de Besoin et l'Identification des Objectifs de Résilience.

## Description de la méthode

### Expression des besoins de résilience

Pour les parties prenantes du système, le besoin de résilience traduit la criticité de ses missions et s'exprime comme la nécessité d'organiser et de faire évoluer le système afin de maintenir coûte que coûte les opérations qui engagent la réussite des missions prioritaires.

A cet effet, il s'agit premièrement de représenter les missions en modélisant leurs fonctions essentielles et leurs interdépendances, d'analyser et hiérarchiser les exigences spécifiques à ces fonctions (dont le spectre s'étend de la fiabilité à la confidentialité), puis de déterminer les conséquences globales pour les missions du système des multiples modes d'agression, en raison notamment des interdépendances ([5]).

A la différence d'une approche classique de prévention – protection, la résilience intègre la possibilité de dysfonction de l'état du système, qu'il s'agisse de l'arrêt de certains services ou de la compromission de composants par un attaquant. Mais de telles dysfonctions doivent être « contenues » dans le but de maintenir les fonctions essentielles. Les seuils de dysfonction tolérables fixés par les parties prenantes circonscrivent une enveloppe de résilience (voir Figure 2). Ces seuils concernent une « valeur limite » de fonctionnement mais aussi la durée de la dysfonction : le temps est un paramètre fondamental dans l'expression du besoin de résilience.

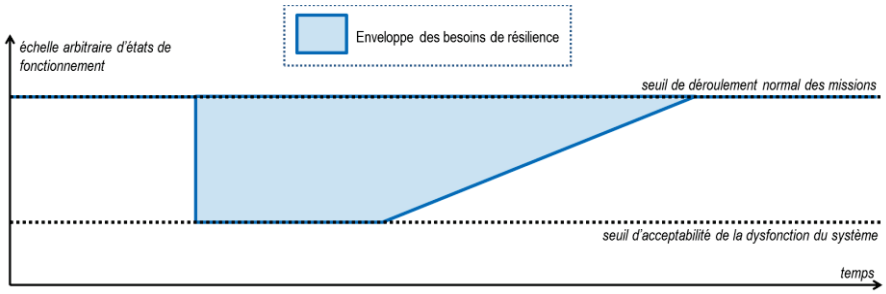



**Figure 2 : enveloppe des besoins de résilience**

On peut ajouter au sein de cette représentation l'évolution dans le temps des états du système, caractérisés par des objectifs relatifs aux capacités opérationnelles du système (voir Figure 3) :

- **Eviter** : capacité du système à prévenir l'apparition d'une menace ou à empêcher une agression de se produire
- **Survivre & Neutraliser** : capacité du système maintenir ses fonctions vitales malgré une agression, et à mener des actions actives pour neutraliser les sources de cette agression.
- **Rétablir** : capacité du système à reconstituer ses composants compromis, puis à rétablir l'ensemble de ses fonctions.
- **Evoluer** : capacité du système à tirer parti de l'expérience pour se transformer pour devenir plus efficace contre les agressions qu'il a déjà subies ou par anticipation d'agressions possibles.

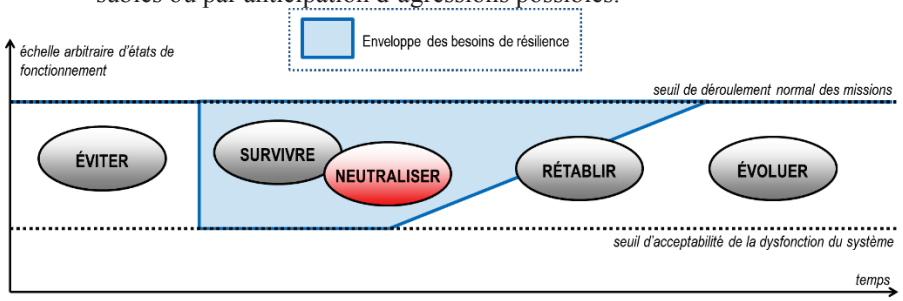



**Figure 3 : objectifs propres à chaque état du système**

## Modélisation du système au sein de l'espace

On modélise l'espace et le système au sein de l'espace en s'appuyant sur la théorie du système général ([6]). Cela permet de prendre en compte la multiplicité des causes et la diversité des scénarios, en représentant les composants physiques, les processus logiques et les acteurs humains. Par ailleurs, l'approche est conforme au modèle militaire US, dont l'usage se généralise ([7], [8]). On introduit donc un modèle de représentation stratifiée de l'espace en 3x2 strates (voir Figure 4) :

- [ANTHROPIQUE-1] **Strate cognitive** : représentations sociales des êtres humains
- [ANTHROPIQUE-2] **Strate humaine** : êtres humains, organisés en réseaux sociaux
- [CYBERNETIQUE-1] **Strate cyber persona** : identités numériques nécessaires pour les échanges
- [CYBERNETIQUE-2] **Strate logique** : processus et données informatiques

- [PHYSIQUE-1] **Strate composant** : composants physiques de l'espace (inclut les composants SI)
- [PHYSIQUE-2] **Strate géographique** : localisation géographique des composants et des personnes

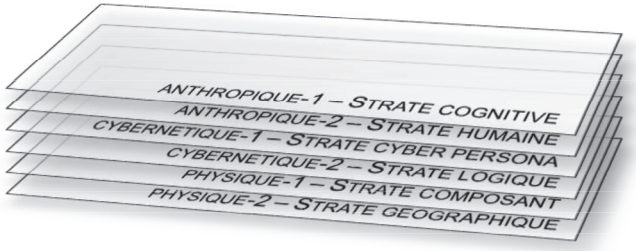



**Figure 4 : modélisation stratifiée de l'espace**

Chaque strate est composée de nœuds et (sauf exception, cf. ci-dessous) de liens intra-strates parcourus par des flux, de natures très différentes en fonction de la strate concernée (voir Tableau 2). Les échanges entre différentes strates sont matérialisés par des liens inter-strates parcourus par des flux.

La modélisation des systèmes au sein de ce modèle de l'espace est facilitée par l'apport de bases de connaissances existantes issues de domaines variés (i.e. non nécessairement des domaines de la Sûreté de Fonctionnement et de la cybersécurité) et alimentées par ailleurs, en fonction des besoins, par des travaux de recherche menés par les auteurs.

En outre, la modélisation des systèmes complexes est facilitée :

- Par l'usage de patterns de systèmes, qui représentent des systèmes fréquemment rencontrés et modélisés une fois pour toutes pour être intégrés dans des systèmes complexes
- Par l'application d'une méthodologie itérative, en modélisant des systèmes de haut niveau et en appliquant des raffinages successifs
- Par l'application de techniques de modélisation permettant de borner la complexité :
  - Etude macroscopique permettant de supprimer des « pans » de scénarios
  - Etude des symétries pour n'étudier qu'une partie de la symétrie
  - Outillage de modélisation

| STRATE | CARACTERISTIQUES | BASES DE CONNAISSANCE |
|---|---|---|
| [ANTHROPIQUE-1] **Strate cognitive** | Cette strate très caractéristique n'existe que par ses liens avec la strate [ANTHROPIQUE-2].<br>Elle inclut les représentations, perceptions etc. | Bases de connaissance issue du domaine des risques psycho-sociaux (notions associées : marque, connaissance, ego, influence, propagande, culture, psychologie) |
| [ANTHROPIQUE-2] **Strate humaine** | Modèle fractal : l'unité est la personne, les réseaux sociaux sont imbriqués<br>• Nœud : individus et groupes d'individus (cf. bases de connaissances)<br>• Lien : lien social (Autorité, Partenariat, Antagonisme…)<br>• Flux : flux associés aux liens sociaux (Subornation, Collaboration, Confrontation…) | Bases de connaissance de nœuds issues de divers domaines :<br>• [21] EBIOS V2 (actifs de type ORG, PER)<br>• [22] EBIOS 2010 (actifs de type PER, CAN)<br>• Domaine sécurité physique : [23] CNPP<br>• Domaine RSE : [24] ISO 26000 |
| [CYBERNETIQUE-1] **Strate cyber persona** | Cette strate très caractéristique n'existe que par ses liens avec d'autres strates (pas de lien intra-strate). Elle se matérialise par ses nœuds, qui correspondent à des identités numériques. Pour créer, modifier, supprimer une cyber-persona il faut agir sur la strate logique dont elle est une abstraction.<br>Des attributs peuvent être associés aux cyber-persona, comme des secrets de connexion | Des travaux ont été réalisés pour caractériser la nature des liens inter-strate :<br>• Liens d'identité : il s'agit de l'identité numérique d'un nœud (abstraction) :<br>  ○ [humaine] @mail, compte utilisateur, etc.<br>  ○ [logique] @IP, nom DNS, etc.<br>  ○ [composant] @MAC, etc.<br>• Lien de codage : l'identité numérique (en tant que donnée) est codée :<br>  ○ Au sein de la strate [logique]<br>  ○ Au sein de la strate [composant]<br>• Lien de connaissance : l'identité numérique (en tant qu'identité) est connue :<br>  ○ Au sein de la strate [logique]<br>  ○ Au sein de la strate [humaine] |
| | | |

| STRATE | CARACTERISTIQUES | BASES DE CONNAISSANCE |
|---|---|---|
| [CYBERNETIQUE-2] **Strate logique** | Modèle fractal : chaque système logique peut être constitué de sous-systèmes logiques et peut être le constituant de systèmes plus volumineux :<br>• Nœud : processus informatique<br>• Lien : protocole de communication<br>• Flux : flux d'information<br>L'ordonnancement des flux constitue un cycle de vie pour les informations ([9]) :<br>• Génération de l'information<br>• Traitement de l'information<br>• Stockage de l'information<br>• Transmission de l'information<br>• Consommation de l'information<br>• Destruction de l'information<br><br>La phase Consommation correspond à l'utilisation de cette information par une entité de la strate anthropique ou une entité de la strate physique | Bases de connaissance de nœuds issues de divers domaines :<br>• [21] EBIOS V2 (LOG, RES_REL, RES_INT)<br>• [22] EBIOS 2010 (LOG)<br>Des travaux ont été réalisés pour caractériser la nature des liens intra-strate :<br>• avec le système de fichiers<br>• avec la mémoire<br>• avec le système d'exploitation via les appels système<br>• avec d'autres composants du même système logiciel<br>• avec les bibliothèques chargées dynamiquement<br>• avec d'autres logiciels via des APIs<br>• avec d'autres logiciels via des IPCs<br>• avec les variables d'environnement ou le registre<br>• avec le réseau<br>Des travaux sont en cours pour caractériser la nature des liens inter-strate, dont :<br>• Avec la strate [humaine] : interactions avec l'utilisateur<br>  o Ecran d'information<br>  o GUI<br>• Avec la strate [composant] : interactions avec les équipements physiques :<br>  o Socle physique d'un composant logique<br>  o Périphériques<br>  o Lien avec un capteur<br>  o Lien avec un actionneur<br>• Avec la strate [cyber persona] : identité d'un nœud logique<br>  o Lien d'identité<br>  o Lien de connaissance |
| [PHYSIQUE-1] **Strate composant** | Modèle fractal : plan d'occupation des sols vs. plan d'un bâtiment<br>• Nœud : cf. KDB<br>• Lien : canalisation [matière], câble électrique [énergie], câble ethernet [information]<br>• Flux : flux de matière, flux d'énergie, flux d'information | Bases de connaissance de nœuds issues de divers domaines :<br>• [21] EBIOS V2 (PHY, MAT, RES_INF)<br>• [22] EBIOS 2010 (LOC, MAT, RSX, PAP)<br>• Domaine sécurité physique : [23] CNPP (biens immeubles & meubles, flux) |
| [PHYSIQUE-2] **Strate géographique** | La strate est soumise aux lois de la physique et est « continue » (pas nœuds, pas de liens, pas de flux).<br>Une entité [composant] dans un et un seul lieu géographique à un instant donné, une entité [humaine] dans un et un seul lieu géographique à un instant donné | Sans objet |

**Tableau 2 : base de connaissance des composants par strate**

Ce modèle constitue la matrice de tous les autres, et est de fait l'élément central de la méthodologie :

- Il permet de modéliser :
  - L'attaquant et le défenseur via les nœuds, les liens et les flux (voir Tableau 2)
  - Les ressources (voir Tableau 3) et les capacités de l'attaquant / du défenseur (voir Tableau 4)
- Il permet de matérialiser les scénarios d'attaque / de défense
  - Les vulnérabilités de la cible sont des caractéristiques des nœuds ou des liens
  - Les événements seront exprimés comme des transformations de la structure stratifiée

| RESSOURCES | RESSOURCES [PHYSIQUE] | RESSOURCES [CYBERNETIQUE] | RESSOURCES [ANTHROPIQUE] |
|---|---|---|---|
| | • Moyens de télécommunication<br>• Moyens de transport<br>• Armement<br>• Etc. | • Outils de développement<br>• Outils de cyberdéfense<br>• Cyberarmes : vecteurs & charges<br>• Etc. | • Organisation & personnels<br>• Connaissances et compétences<br>• Leadership<br>• Etc. |
| | RESSOURCES FINANCIERES | SOUTIEN EXTERIEUR | |

**Tableau 3 : caractérisation des ressources par strate**

| | [PHYSIQUE] | [CYBERNETIQUE] | [ANTHROPIQUE] |
|---|---|---|---|
| CAPACITES DE RENSEIGNEMENT | **Recherche par sources ouvertes**<br>• analyse de documents papier<br>• exploration de zones publiques<br>• observations zones non protégées<br>**Renseignement technique**<br>• imagerie<br>• recueil et analyse de signaux<br>• interception de communications<br>**Reconnaissance physique**<br>• opérations spéciales<br>**Vol de support physique** | **Recherche par sources ouvertes**<br>• cyber-réseaux sociaux<br>• bases de données spécifiques<br>**Recherche Darknet**<br>**Rétro-ingénierie & analyse de vulnérabilités logicielles**<br>• équipements sur étagère<br>• équipements particuliers<br>• cryptanalyse<br>**Cyber reconnaissance**<br>• cartographie logique de la cible<br>• identification des cyber-persona<br>**Cyber espionnage**<br>• analyse de trafic<br>• interception des données transmises<br>• exfiltration des données stockées | **Recherche par sources ouvertes**<br>• conférences, salons<br>• visites autorisées etc.<br>**Recueil d'information via des réseaux**<br>• associations diverses<br>• syndicats professionnels…<br>**Recueil ponctuel d'information auprès d'un personnel**<br>• élicitation (cf. influence)<br>**Renseignement par agent infiltré**<br>• cf. recrutement ou subversion |
| CAPACITES DEFENSIVES | **Sécurité physique des installations propres**<br>• extra-territorialisation, sanctuarisation<br>• sécurité périmétrique etc.<br>**Sécurité physique des opérations**<br>• protection des communications<br>• déception : dissimulation, camouflage, leurres etc. | **Cyberdéfense infrastructures propres**<br>**Protection de l'identité**<br>• emploi de ressources tierces<br>• dissimulation du point d'entrée dans le cyberespace (e. g. TOR)<br>• cyber-persona anonymes<br>• chiffrement des communications et obfuscation de code<br>**Furtivité des cyber-opérations**<br>• vulnérabilité ou exploits inédits<br>• malware non répertorié<br>• génération de bruit<br>• cycle OODA lent<br>**Persistance**<br>• attaque des moyens de défense<br>• enfouissement<br>• polymorphisme<br>• prolifération | **Sécurité de l'organisation**<br>• topologie de l'organisation (décentralisation, cloisonnement)<br>• culture du secret, codes de communication<br>• méthodes de recrutement (criblage, affiliation etc.)<br>• sécurité juridique<br>• blanchiment<br>**Sécurité du personnel**<br>• formation<br>• surveillance interne<br>**Sécurité des opérations**<br>• procédure OPSEC<br>• déception<br>• emploi de tiers |
| CAPACITES OFFENSIVES | **Entrée physique dans un site**<br>• pénétration ( force)<br>• infiltration ( ruse)<br>**Piégeage & sabotage des installations**<br>**Attaques physiques**<br>• armes légères<br>• moyens lourds ou sophistiqués (explosifs, armement spécial) | **Entrée cybernétique**<br>• accès par un vecteur humain, cyber, physique ou logistique<br>**Prise de contrôle de cyber-ressources**<br>• usurpation d'identité ou de connexion<br>• création de cyber-persona non autorisée<br>• insertion de ressources non autorisées<br>**Cyberattaque**<br>• manipulation des services et des données<br>• déni de service | **Influence**<br>• persuasion, séduction<br>• intimidation<br>**Subversion par cooptation**<br>• Intoxication, endoctrinement<br>• coercition<br>• stipendiement<br>**Subversion par infiltration**<br>• vol d'identité<br>• manipulation du recrutement<br>**Désinformation, Déstabilisation de l'organisation** |

**Tableau 4 : caractérisation des capacités par strate**

Ce modèle, initialement conçu pour la cybersécurité, permet de projeter sans difficultés particulières un périmètre de Sûreté de Fonctionnement, et notamment d'y modéliser les facteurs humains (paramètre à ce jour insuffisamment pris en compte dans plusieurs normes, dont [2]) :

- Les états du système (e.g. erreurs dans le cas de la sûreté de fonctionnement, usurpations ou altérations dans le cas de la cybersécurité) sont des caractéristiques des nœuds et des liens.
- Les événements (faute→ erreur→ défaillance ou attaque → conséquence) sont matérialisés par des flux qui transforment la structure stratifiée (nœuds et liens)

En termes de méthode, on aborde la modélisation d'un système de la même manière que dans les approches d'*architecture framework* sur la base du « substrat » que représente la modélisation stratifiée de l'espace. Les missions du système dictent sa projection sur ce substrat, et une méthode itérative par raffinements successifs permet d'obtenir une représentation complète et conforme à la réalité.

## Modélisation des scénarios de risque

On modélise les scénarios de risque comme des enchaînements d'événements élémentaires (voir Figure 5). Chaque événement élémentaire est applicable à un nœud au sein d'une strate de l'espace ou à un lien (intra-strate ou inter-strate), et il en résulte une transformation de l'espace. Les événements élémentaires de la Sûreté de Fonctionnement sont des dysfonctions d'une partie du système :

- [DYSFONCTION-1] Corruption des capacités de la cible
- [DYSFONCTION-2] Dégradation des capacités de la cible

Il est nécessaire, pour aborder le domaine de la cybersécurité, d'introduire des événements complémentaires permettant de mettre en évidence les caractéristiques spécifique de celui-ci : les activités de l'attaquant de transformation de l'espace pour générer ou acquérir de nouvelles capacités qu'il emploiera ensuite

Afin de couvrir tous les cas, on ajoute donc aux dysfonctions, des événements élémentaires, caractérisant des exploitations ou des usurpations du système ou d'un composant du système :

- [EXPLOITATION-1] Exécution de capacités propres
- [EXPLOITATION-2] Génération de capacités propres
- [USURPATION-1] Exécution de capacités de la cible
- [USURPATION-2] Insertion de nouvelles capacités dans la cible

Si l'on considère l'ensemble des scénarios d'attaque, on peut repérer des régularités ([10]) qui expriment la nécessité pour l'attaquant d'acquérir des connaissances puis de transformer l'espace pour se mettre en capacité d'obtenir l'effet majeur recherché. Les événements élémentaires s'enchaînent selon un processus de décision modélisable par une boucle OODA ([11]). L'utilisation de ce concept est dictée par les spécificités de la cybersécurité, mais il peut aussi être appliqué à la sûreté de fonctionnement, par exemple pour prendre en compte le facteur humain et appréhender les erreurs qui en résultent ([12]). Néanmoins, cet outil ne sera pas utilisé pour modéliser des défaillances uniquement techniques.

Comme les modèles de sûreté de fonctionnement disposent aujourd'hui d'un ensemble de méthodes et outils reconnus, ([13]), la suite se focalise sur les modèles de cybersécurité

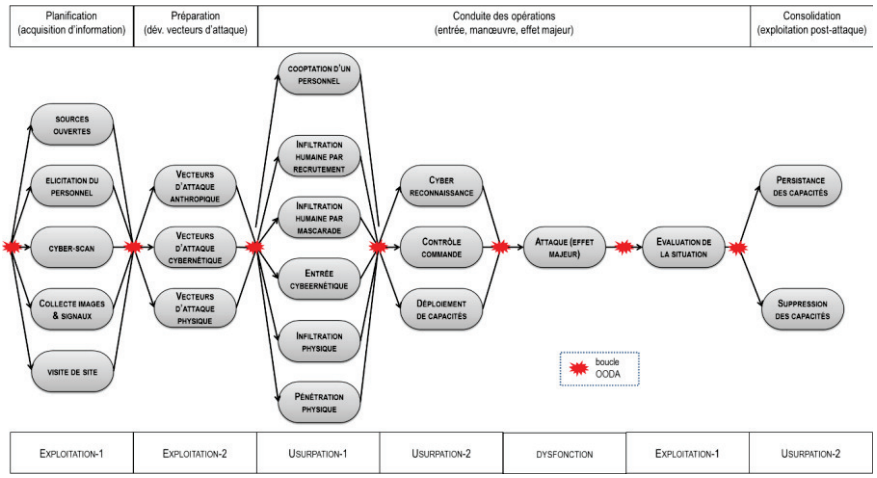



**Figure 5 : Kill Chain générique incluant les points de décision**

Il faut noter que même s'il s'agir d'une cyber-opération, les phases liminaires peuvent nécessiter des actions dans les strates anthropiques et/ou physiques, notamment pour acquérir les connaissances nécessaires ou entrer dans la cible.

La Cyber Kill Chain présentée est strictement fonctionnelle et elle permet de mener des analyses en phase de conception d'un système. Mais elle n'est pas adaptée à la modélisation des scénarios réels de risques. On doit donc la raffiner en introduisant progressivement les tactiques, techniques et procédés (TTPs) mis en œuvre par l'adversaire. Pour ce faire, on dispose d'une taxonomie de tactiques, construites à partir des capacités essentielles de cyber-opérations définies dans la doctrine US ([7]). On pourra alors déterminer l'ensemble des scénarios de risque possibles en appliquant ces TTPs sur les éléments (nœuds et liens) du système modélisé.

| Nomenclature | | Tactiques |
|---|---|---|
| C-ISR | C-ISR-REC RECONNAISSANCE | Captation de signaux (canaux cachés) |
| | | Analyse de trafic |
| | | Enumération et caractérisation des noeuds |
| | | Enumération et caractérisation des liens |
| | | Cartographie des cyber-persona |
| | C-ISR-ANL ANALYSE | Exploitation des traces |
| | | Rétro-ingénierie |
| | | Recherche et découverte de vulnérabilités |
| | C-ISR-ESP ESPIONNAGE | Interception de données transmises |
| | | Exfiltration de données stockées |
| | | Cryptanalyse |
| C-DEF | C-DEF-ATD ANTI-DETECTION | Génération de « bruit » |
| | | Effacement des traces produites |
| | | Enfouissement (zones particulières) |
| | | Mutation du code (en temps réel) |
| | C-DEF-PER PERSISTANCE | Clonage et prolifération des capacités |
| | | Activation de liens de communication multiples |
| | | Manipulation des services de démarrage |
| | C-DEF-NEU NEUTRALISATION | Manipulation des services de sécurité |
| | | Interdiction des services de sécurité |
| | C-DEF-AAN ANTI-ANALYSE | Chiffrement des communications |
| | | Chiffrement des données et instructions |
| | | Reconfiguration / polymorphisme |
| C-OPE | C-OPE-ACC ACCES | Connexion à un nœud distant |
| | | Elévation de privilège |
| | C-OPE-DEP DEPLOIEMENT | Manipulation de cyber-persona |
| | | Transfert ou insertion de code |
| | | Corruption de code |
| | | Modification non autorisée de configuration |
| | | Activation de capacités non autorisé |
| | C-OPE-C&C COMMANDE-CONTROLE | Contrôle-commande distant avec un serveur central |
| | | Contrôle-commande décentralisé entre capacités déployées |
| | C-OPE-MOD MODELAGE | Agrégation de données |
| | | Mise en forme de données (avant exfiltration) |
| C-ATK | C-ATK-MAN MANIPULATION | Manipulation de données (génération, suppression, falsification) |
| | | Exécution non autorisée de services |
| | | Mésexécution de service |
| | C-ATK-INT INTERDICTION | Disruption de service |
| | | Dégradation de service |
| | | Destruction de capacité |

Exemple de techniques et procédés tirés du modèle MAEC ([14])

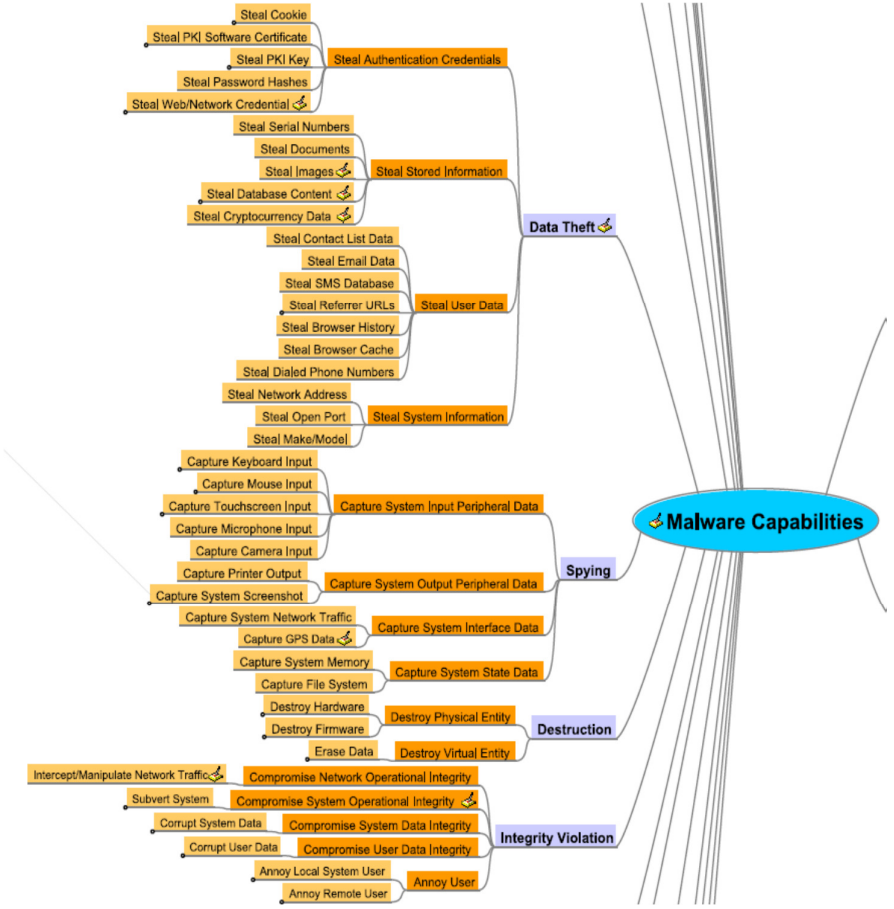



**Tableau 5 : Techniques, Tactiques et Procédés**

On procède ensuite par élagage pour identifier les scénarios plausibles :

- Les scénarios conformes aux capacités de l'attaquant, déterminées lors de son profilage (obtenu en appliquant le Tableau 5)
- Les états intermédiaires « obligatoires » du système, i.e. les « points décisifs » ([15]
- Les symétries du système et / ou des scénarios d'attaque
- Etc.

On peut ensuite synthétiser l'ensemble des scénarios d'attaque sous forme d'arbres structurés autour des kill chain ([16]), de la même façon qu'on peut représenter les scénarios de défaillance par des arbres. La comparaison des deux permet ensuite de dériver de nouveaux scénarios, combinant des défaillances non intentionnelles avec des attaques.

## Gestion des risques

Les sections précédentes ont permis de construire un cadre commun à la sûreté de fonctionnement et à la cybersécurité pour la modélisation des risques, dont l'intérêt est de permettre la prise en compte concomitante de l'ensemble de ces risques ; en particulier il facilite l'identification et l'anticipation les conflits possibles entre les deux domaines.

La gestion de l'ensemble des risques sur le système étudié suppose d'élaborer une stratégie globale capable de répondre à la variété des scénarios retenus. La définition et la mise en œuvre de cette stratégie se fait en quatre temps :

- D'abord, on fixe pour chaque scénario un objectif de résilience, représenté par une courbe dans l'espace des phases du système (voir Figure 6), qui exprime la réponse attendue du système en fonction de ses capacités (Eviter, Survivre, Neutraliser, Rétablir, Evoluer).

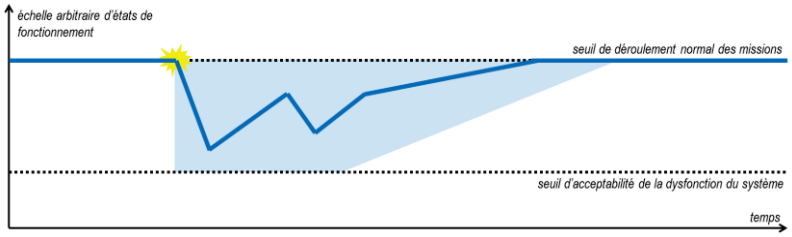



**Figure 6 : modélisation des objectifs de résilience**

- La mise en œuvre de ces capacités repose sur des fonctions fondamentales qui se combinent pour former une **Stratégie de résilience**. Une telle stratégie est conçue pour contrer un adversaire et doit donc être adaptée à ses opérations. Afin de proposer un modèle générique qui pourra être adapté à peu près toutes les situations, on reprend les grandes fonctions stratégiques définies dans la doctrine française de Sécurité Nationale, ([17])
- Ensuite, on détermine, en fonction de leur principe d'action sur le scénario, les combinaisons possibles de tactiques qui déclineront cette stratégie sur le système réel. Cette analyse est menée de manière systématique en s'appuyant sur une matrice de principes d'action entre un ensemble prédéfini de tactiques et les étapes du scénario. Cette matrice a été conçue à partir de plusieurs travaux dont le référentiel du MITRE ([18]) et la partie 7 de l'IEC 61508 ([13]). Ces tactiques pourront ensuite être raffinées en techniques spécifiques.

- Enfin, on effectue la synthèse des résultats précédents en choisissant un sous-ensemble jugé optimal selon les critères des parties prenantes. Cette étape doit notamment analyser les conflits éventuels entre les différentes techniques retenues ; c'est pourquoi il est important d'identifier l'ensemble des combinaisons adéquates afin de proposer plusieurs options.

## Elaboration de la stratégie

Afin de proposer un modèle générique qui pourra être adapté à peu près toutes les situations, on reprend les grandes fonctions stratégiques définies dans la doctrine française de Sécurité Nationale, ([17]).
Ces grandes fonctions sont respectivement : CONNAISSANCE et ANTICIPATION, PREVENTION, DISSUASION, PROTECTION et INTERVENTION. On les complète par une fonction d'EVOLUTION, indispensable pour répondre au principe d'amélioration continue consubstantiel à la résilience
Ces fonctions stratégiques doivent être mises en œuvre par des modes d'action adaptés aux entités et aux opérations mises en jeu dans l'étape de l'attaque considérée.
Les entités associées sont l'Adversaire et les Eléments Sensibles du système cible ([1]) ; conformément au cycle OODA, les opérations entre l'Adversaire et les éléments sensibles du système se subdivisent en deux catégories générales : OBSERVATION et ACTION ; de la même façon , le Défenseur effectue des OBSERVATION et des ACTION.
On en déduit une liste complète de modes d'action déclinant les fonctions stratégiques (voir Tableau 5). Il faut noter que cette approche systématique vaut pour la sûreté de fonctionnement ([1]) et pour la cybersécurité ([19]).
Par contre, il ne s'agit pas d'appliquer aveuglément tous les modes d'action à toutes les phases, mais d'en considérer un sous-ensemble approprié vis-à-vis des objectifs de résilience. A priori il n'y a pas de solution unique et il peut être intéressant de concevoir des alternatives, pour :

- d'une part, construire une stratégie d'ensemble faisant la synthèse des stratégies particulières contrant les différents adversaires et/ou modes de défaillance ;
- d'autre part conserver une certaine liberté d'action pour s'adapter aux évolutions de l'environnement tout en demeurant efficace.

| Fonction stratégiques | Classes de Mode d'action | Modes d'action |
|---|---|---|
| **CONNAISSANCE et ANTICIPATION** | **Observation de l'Adversaire** | Identifier un Adversaire Externe |
| | | Surveiller un Adversaire Externe |
| | | Identifier un Adversaire Interne<br>• Insider sous contrôle de l'adversaire<br>• Insider « pur » |
| | | Surveiller un Adversaire Interne |
| | **Observation des Observation / Action de l'Adversaire** | Détecter et identifier une Observation / Action |
| | | Analyser après coup une attaque |
| | | Identifier un Elément Sensible compromis (i.e. usurpé par l'Adversaire)<br>• par malware<br>• par cyber persona non autorisée |
| | | Surveiller les activités d'un Elément Sensible usurpé |
| | **Observation des Eléments Sensibles** | Déterminer les Eléments Sensibles critiques |
| | | Identifier des vulnérabilités des Eléments Sensibles |

| Fonction stratégiques | Classes de Mode d'action | Modes d'action |
|---|---|---|
| | | Surveiller l'état des Eléments Sensibles |
| | | Identifier les Eléments Sensibles compromis ou altérés |
| **PREVENTION** | **Action contre l'Adversaire** | Attaque directe : neutraliser les ressources propres de l'Adversaire |
| | | Attaque indirecte : modifier l'environnement de l'Adversaire pour neutraliser ses soutiens dans un mode préventif |
| **DISSUASION** | **Action contre l'Adversaire** | Menacer l'Adversaire de représailles |
| | **Action contre l'Observation de l'Adversaire** | Tromper l'Adversaire pour aboutir à un calcul bénéfice/coût en sa défaveur |
| | **Action contre l'Action de l'Adversaire** | Dissuader l'Adversaire par déni en l'empêchant d'atteindre ses objectifs |
| | **Action sur les Eléments Sensibles** | Dissuader l'Adversaire par déni en l'empêchant d'atteindre ses objectifs |
| **PROTECTION** | **Action sur les Eléments Sensibles** | Mettre en place des leurres pour faire perdre du temps à l'Adversaire ou fausser sa prise de décision |
| | | Dissimuler les Eléments Sensibles pour empêcher l'Observation |
| | | Durcir les Eléments Sensibles pour les rendre insensibles aux Actions adverses |
| | | Modifier les Eléments Sensibles pour absorber les Actions adverses |
| | | Déplacer les Eléments Sensibles pour les mettre hors de portée des Actions adverses |
| **INTERVENTION** | **Action contre l'Observation de l'Adversaire** | Bloquer l'Observation pour entraver la prise de décision de l'Adversaire |
| | | Corrompre l'Observation pour intoxiquer la prise de décision de l'Adversaire et rendre son opération inefficace |
| | **Action contre l'Action de l'Adversaire** | Bloquer son Action pour entraver les opérations de l'Adversaire |
| | | Neutraliser son Action en modifiant son contenu (e.g. supprimer la charge d'un vecteur d'attaque) |
| | | Dévier son Action (e.g. modifier la trajectoire d'un vecteur d'attaque) |
| | **Action sur les Eléments Sensibles** | Régénérer les Eléments Sensibles touchés, rétablir les services dégradés ou désactivés |
| **EVOLUTION** | **Action sur les Eléments Sensibles** | Immuniser les Eléments Sensibles aux Observations / Actions précédentes en les transformant |

**Tableau 5 : mise en œuvre des fonctions stratégiques**

**Remarque** : ces attaques contre l'Adversaire sont a priori possibles dans toutes les phases, elles correspondent à la capacité *Defensive Cyberspace Operations – Response Action* de la doctrine US ([7]). Cette voie ne sera pas développée car elle est réservée aux services de l'Etat.

## Choix des tactiques

L'étape suivante consiste à déterminer des tactiques de défense qui mettront en œuvre les fonctions stratégiques en s'adaptant aux tactiques de l'adversaire. Le tableau présente le *draft* d'une matrice de correspondance entre le référentiel du MITRE et les tactiques offensives exposées précédemment.

| | C-ISR | | | C-OPE | | | C-DEF | | | | C-ATK | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **REC.** | **ANA.** | **ESP.** | **ACC.** | **DEP.** | **C&C** | **A-D.** | **NEU.** | **PER.** | **A-A.** | **MAN** | **INT** |
| **Réponse adaptative** | REDUIRE | | CONTENIR REDUIRE | PREVENIR REDUIRE | PREVENIR RECUPERER | DEGRADER RETARDER CONTENIR REDUIRE | REDUIRE | | CONTENIR REDUIRE | | CONTENIR REDUIRE | RECUPERER |
| **Surveillance analytique** | DETECTER ANALYSER | | DETECTER ANALYSER | DETECTER ANALYSER | DETECTER ANALYSER | DETECTER ANALYSER | REDUIRE | DETECTER | | REDUIRE | DETECTER ANALYSER | DETECTER ANALYSER |
| **Défense coordonnée** | | | DEGRADER RETARDER | | DEGRADER RETARDER | DETECTER DEGRADER RETARDER | DEGRADE R | REDUIRE | DETECTER DEGRADER RETARDER | DEGRADE R | DEGRADER RETARDER | DEGRADE RETARDER |
| **Déception** | DEGRADER RETARDER DETOURNER TROMPER DETECTER ANALYSER | | DISSUADER DETOURNER TROMPER DEGRADER DETECTER ANALYSER | DISSUADER DETOURNER TROMPER ANALYSER | DISSUADER DETOURNER TROMPER ANALYSER | DISSUADER DETOURNER TROMPER DETECTER ANALYSER | | DEGRADE R | DEGRADER | | DISSUADER DETOURNER TROMPER DEGRADER DETECTER ANALYSER | DISSUADER DETOURNER TROMPER DEGRADER DETECTER ANALYSER |
| **Diversité** | | | DEGRADER | | | DEGRADER CONTENIR | DEGRADE R | DEGRADE R | | | DEGRADER | DEGRADER |
| **Position dynamique** | DETECTER REDUIRE | | DEGRADER RETARDER REDUIRE EXPULSER | REDUIRE RETARDER | DEGRADER RETARDER REDUIRE EXPULSER RETABLIR | DEGRADER RETARDER REDUIRE EXPULSER RETABLIR | | REDUIRE | REDUIRE EXPULSER | | DEGRADER RETARDER REDUIRE EXPULSER RETABLIR | DEGRADER RETARDER REDUIRE EXPULSER RETABLIR |
| **Vision dynamique** | ANALYSER | | DETECTER | | | DETECTER ANALYSER | DEGRADE R | | DETECTER | | DETECTER RETABLIR | DETECTER RETABLIR |
| **Non-persistance** | DEGRADER RETARDER | DEGRADER RETARDER | DEGRADER | REDUIRE EXPULSER | REDUIRE EXPULSER | REDUIRE EXPULSER | | | DEGRADER EXPULSER | | REDUIRE | REDUIRE |
| **Restriction des privilèges** | | | EMPECHER RETARDER CONTENIR | EMPECHER RETARDER CONTENIR | EMPECHER RETARDER CONTENIR | EMPECHER RETARDER CONTENIR | | EMPECHE R RETARDE R CONTENIR | EMPECHER RETARDER CONTENIR | | EMPECHER RETARDER CONTENIR | EMPECHER RETARDER CONTENIR |
| **Réalignement** | DEGRADER RETARDER | | EMPECHER | DEGRADER | DEGRADER | EMPECHER DEGRADER | DEGRADE R | DEGRADE R | DEGRADER | | EMPECHER DEGRADER | EMPECHER DEGRADER |
| **Redondance** | | | | | | | | | | | DEGRADER REDUIRE RETABLIR | DEGRADER REDUIRE RETABLIR |
| **Segmentation Isolation** | CONTENIR | | DEGRADER RETARDER CONTENIR | EMPECHER CONTENIR | EMPECHER CONTENIR | DEGRADER RETARDER | | | DEGRADER RETARDER CONTENIR | | DEGRADER RETARDER CONTENIR RETABLIR | DEGRADER RETARDER CONTENIR RETABLIR |
| **Intégrité substantielle** | | | EMPECHER CONTENIR | | EMPECHER CONTENIR | DETECTER REDUIRE | | DETECTER REDUIRE | DETECTER REDUIRE | | REDUIRE RETABLIR | REDUIRE RETABLIR |
| **Imprédictibilité** | RETARDER | | | DETECTER RETARDER | | DETECTER RETARDER | | DETECTER | DETECTER | | DETECTER RETARDER | DETECTER RETARDER |

**Tableau 7 : Application des tactiques défensives du MITRE**

## De l'usage de la méthode

A ce stade de l'exposé, on dispose des éléments suivants :

- Une formalisation de la résilience, où convergent sûreté de fonctionnement et cybersécurité
- Une méthode d'identification des besoins et objectifs de résilience
- Une méthode d'identification et de traitement des scénarios de risque
- Des arguments permettant de lever les antagonismes entre les exigences de sûreté de fonctionnement et celles de cybersécurité

L'étape suivante doit permettre d'implémenter la stratégie de maîtrise des risques au sein du cycle de vie du système.

Un cycle de vie du système guidé par les objectifs de résilience et porté par une stratégie de résilience est nécessaire pour obtenir au final un système sécurisé, sûr, résistant, évolutif etc. garantissant l'accomplissement des missions essentielles. C'est également le meilleur, sinon le seul, moyen de maîtriser la complexité en hiérarchisant les exigences fonctionnelles et en les prenant en compte le plus tôt possible.

En outre, au regard de la complexité des systèmes et de l'effort d'analyse qui en découle, il apparaît indispensable de se doter d'outils puissants au service de la modélisation d'une part, et de la méthodologie d'autre part. L'outillage est un enjeu décisif pour l'applicabilité de cette démarche.

Pour conclure, on peut citer la théorie du *known and unknown* ([20]) qui matérialise le fait qu'on ne se protège pas contre ce que l'on ne connait pas. Une partie des cyberattaques ne pourront pas être modélisées, simplement parce qu'elles sont inconnues d'une part et impossibles à connaître d'autre part. C'est un dilemme bien connu des organisations de type SOC en charge de la détection des cyberattaques ; l'enjeu est alors de réduire au maximum l'étendue de l'inconnu. Cet enjeu doit être partagé par les objectifs de résilience.

## Références


[1] RATP, 2006, « Maîtrise des Risques d'Entreprise / Défense en Profondeur »

[2] NF EN 61508:2010, « Partie 4: définitions et abréviations »

[3] NIST, 2014, « Framework for Improving critical Infrastructure Cybersecurity »

[4] EBIOS 2010, « Guide méthodologique »

[5] Mitre, 2013, « Cyber Mission Assurance Engineering: A Risk-Based, Threat-Informed Approach to Address Advanced Adversaries »

[6] J.-L. Lemoigne, 1999, « La modélisation des systèmes complexes », Dunod

[7] US DoD, 2013, « Joint Publication 3-12 Cyberspace Opérations »

[8] O. Kempf, 2015, « Introduction à la cyberstratégie », 2e édition, Economica

[9] K. Jabbour et al., 2011,« The Science of Mission Assurance », Journal of Strategic Security, Vol.4, Issue 2

[10] M. Gell-Mann, 1998, « Le Quark et le Jaguar , voyage au cœur du simple et du complexe », Flammarion,

[11] F. Osinga, 2007, « Science, Strategy and War: The Strategic Theory of John Boyd », Routledge

[12] J. Reason, 2013, « L'erreur humaine », Presse de l'Ecole des Mines

[13] NF EN 61508:2010, « Partie 7 : présentation de techniques et mesures »

[14] http://maec.mitre.org

[15] Armée de Terre, 2014, « Tactique Générale », Economica

[16] S. Caltagirone et al.,2013, « The Diamond Model of Intrusion Analyses »

[17] Présidence de la République, 20343, « Livre Blanc – Défense et Sécurité Nationale », La Documentation Française
[18] Mitre, 2015, « Cyber Resiliency Engineering Aid -The Updated Cyber Resiliency Engineering Framework
and Gudance on Applying Cyber Resiliency Techniques »
[19] B. Williams, 2014, « The Joint Force Commander's Guide to Cyberspace Operations », Joint Forces Quarterly 73
[20] D. Rumsfeld, 2006, « Known and Unknown: A Memoir »
[21] EBIOS V2, « Section 4 : outillage pour l'appréciation des risques »
[22] EBIOS 2010, « Bases de connaissances »
[23] CNPP, « Référentiel CNPP »
[24] ISO 26000:2010 « Lignes directrices relatives à la responsabilité sociétale »

# Dependability of Programmable Networks: Concepts, Challenges and Future Research Directions

Kahina Lazri, Imen Grida Ben Yahia and Jean-Philippe Wary

Orange Labs. 38 rue du General Leclerc, 92130 Issy-Les-Moulineaux, France.

**Abstract.** After the surge of cloud computing, emerging paradigms such as SDN (Software-Defined Networks) and NFV (Network Function Virtualization) introduce fundamental changes in network design principles as well as in the execution and management of the network functions themselves (e.g., virtualization, programmability). These changes target the unification of IT resources with the network ones, bringing several new challenges for network dependability.
This paper aims at analyzing the current evolutions toward programmable networks from the perspective of dependability. To this end, we first discuss dependability and its main related concepts before addressing the uniqueness of programmable networks in terms of their dependability attributes and challenges. Besides the technological transformation introduced by programmable networks, they are also expected to provide new services. In this paper, we address the key dependability issues introduced by SDN/NFV based networks. Even though considering SDN/NFV based infrastructures, our analysis is more focused on the shortcomings of the current SDN designs. This is because we consider that cloud technologies constitute the main building blocks of NFVI infrastructures (e.g. virtualization, elasticity). Thus, virtualized network functions will be mostly exposed to cloud inherited anomalies and failures.
We argue that dependability requirements should be identified and integrated within the design phase of any SDN/NFV network. We propose future research directions to cover both dependability by-design and dependability by-operation aspects of SDN/NFV networks.

## 1 Introduction

The combination of SDN, NFV and Clouds are expected to shape the future of computing and telecommunication networks.

SDN proposes to make networks programmable by decoupling the control plane (decision-making) from the data plane (packet forwarding). In traditional networks, control logic of the entire network is integrated in the firmware of each network element. To change network behavior (routing protocols, packet management strategy, etc...), the firmware of all involved devices has to be updated. In SDN environments, the control logic is migrated from the distributed network elements to a logically centralized controller. This logical centralization of network control improves network management flexibility, provides independence to telecoms equipment manufacturers and lowers barriers to deploying new services.

NFV is a networking initiative led by Telcos [1], to replace Network Functions (NFs). These are currently implemented in sophisticated and expensive hardware causing costly integration and management within Telcos equipment. With NFV paradigm, this integration is realized by the embodiment of software within commodity hardware (high-volume standard servers, storage and switches). The objective behind NFV is to integrate network and IT domains in the same virtualized datacenters and to consolidate their currently separated management layers [2] into a single logically centralized entity. NFV promises operating expenditure (OPEX) savings, rapid deployment, scalability of the cloud and high flexibility in the management of virtualized network functions.

In cloud infrastructures, VMs can be seen as computing-defined environments and storage-defined environments [3], where the role of the controller is played by a cloud management layer, also referred to as cloud Operating System (Cloud OS). In this execution environment, a Virtual Network Function (VNF) is a network function (routing, load balancing, packet inspection, etc...) running within a simple hosted VM where virtual network behavior is driven by SDN controllers, possibly running in their turn within VMs. The underlying hardware infrastructure is abstracted into a form of virtual resources, becoming completely homogeneous to the running applications.

Thus, SDN and NFV technologies come with opportunities and also a set of new challenges. In this research paper, we discuss mainly dependability, which constitutes a major barrier for SDN/NFV adoption.

SDN controller whether it is centralized for a whole network or distributed for different network segments is considered as a potential single point of failure. Moreover, the privileged position of the controller makes it an attractive target for attackers. Taking the control of an SDN controller allows compromising the entire network. Regarding NFV, virtualization of network functions exposes them to a new type of issues which have so far been reserved to IT infrastructures. Moreover, consolidating network functions in shared commodity servers de facto exposes them to cloud inherited vulnerabilities and failures. Software and hardware failures, multi-tenancy or performance isolation, constitute new sources of issues for network functions.

The objective of this paper is to shed the light on the fundamental concepts from the field of dependable computing and to address dependability challenges introduced by SDN/NFV. We reconsider dependability requirements of cloud IT infrastructures to integrate those of network functions, which are more performance stringent. We also analyze the technical specificities of SDN/NFV and some recently demonstrated attacks against these infrastructures to understand their impact on dependability. In light of the observations delivered along the paper, we propose a research agenda to address SDN/NFV dependability challenges.

This paper is organized as follows. Section 2 provides fundamental concepts of dependable computing. Section 3 gives background on SDN/NFV technologies to consider their impacts on dependability attributes. Section 4 discusses dependability requirements of SDN/NFV infrastructures. Section 5 discusses the main potential sources of failures in SDN/NFV networks and presents the most recent demonstrated attacks on these infrastructures. Finally, section 6 closes the paper with future research directions in the field of SDN/NFV dependability.

## 2 Basic concepts and terminology

Before addressing dependability of SDN/NFV networks, we first give in this section, the fundamental concepts from the field of dependable computing.

### 2.1 Dependability, Resilience and Survivability

The scope of **dependability** has been subject of discussion for a long time. A first definition of dependability was proposed in [4], where dependability is defined as 'the probability that the system will be able to operate when needed'. This definition has two main shortcomings. First, it minimizes the perimeter of dependability to the availability of the system under operation. Second, it ignores the fact that even if a system is available for operation, it may deliver wrong results. In 1982, Carter [5] defines a dependable system as a system which is 'trustworthy enough that reliance can be placed on the service it delivers'. This later definition includes two main properties of dependability: **trustworthy and reliability**. It also stresses the need for reliance on the delivered service rather than on the availability of the system under operation.

**Reliability** is another word usually used interchangeably with dependability. In the primary sense of the word, reliability means the 'ability to rely upon', which connotes a notion of trust. However, as the initial concern of computing systems was to make systems operate without interruption, continuity of service has replaced the initial meaning of reliability.

In order to avoid any confusion between reliability as a reference to service continuity measurement, reliability as a reference to a qualitative characteristic of a system and dependability, members of the IFIP WG 10.4, led by J.-C. Laprie, attempt in 1992 to summarize and clarify the framework of dependability in a reference book, Dependability: Basic Concepts and Terminology [6]. They define dependability as an overarching concept including a set of attributes and threats, and propose the means to achieve it. This proposed definition is the one that is widely accepted today in the dependable computing community: the ability of a system to deliver a service that can justifiably be trusted. This latter definition introduces the need for justification of trust, which requires the definition of metrics for dependability quantification.

**Survivability** is a discipline of dependability, introduced in the field of military networks [7]. Survivability quantifies the degree to which a system is able to deliver essential services despite the presence of severe challenges [8], such as massive attacks or natural disasters causing failures in large parts of a communication network. Thus, to assess survivability, it is necessary to specify beforehand the minimum level of service performance that must be guaranteed and the maximum acceptable level of service disturbance (time or quality of service).

Another close term to dependability is **Resilience**. There exist multiple definitions of resilience, which differ according to the assumed system's application environment. For the dependable computing community, resilience is considered as the persistence of dependability when facing changes [9]. Changes include multiple forms of evolutions in the system execution environment such as increased load, attacks or disasters. Changes might also refer to system modules evolution or system parameters modification. More recently, resilience was reconsidered in the scope of the ResiliNets project [10], where

it was defined as the combination of two disciplines: trustworthiness (dependability, security and performability) and tolerance (survivability, disruption tolerance and traffic tolerance). Resilience assessment is based on dependability attributes. This makes us consider that despite the subtle differences between resilience and dependability, they target the same objective of making computer systems work adequately.

### 2.2 Attributes

Dependability attributes define the main requirements that should be guaranteed by a system to be considered as dependable. Depending on the system application domain, dependability attributes are emphasized differently. Still referring to the work led by Laprie [11], the attributes covered by the initial definition of dependability are:

- **Availability A(t)**: the probability that the system operates correctly at a given point of time;
- **Reliability R(t)**: the probability that the system operates without interruption during a given interval of time;
- **Safety**: the probability of occurrence of *unsafe-failures* [12] leading to catastrophic consequences on the environment;
- **Confidentiality**: non-occurrence of unauthorized disclosure of information;
- **Integrity**: non-occurrence of information alteration;
- **Maintainability**: ability to repair and recover from failures.

### 2.3 'Trinity of trouble' in dependability

Impairments to dependability are the causes that lead a system to cease performing adequately its function. Thus, they include all the events that might cause the system to deviate from its specified behaviour. These events might result from intentional or non-intentional actions. Aviziens *et al* [11] classified dependability impairments into three categories: **faults, errors and failures**, faults may also be considered as vulnerabilities.

Failures occur when a delivered service deviates from its intended behavior. Causes of failures are assigned to the presence of faults in the system. Faults may be defined as program flaws, which if activated, result in system erroneous state. This latest state is a latent error, which when activated, becomes an active one. In fact, a fault resides in the program code whereas an error appears during the program execution (triggering the faulty instruction by an appropriate input pattern). Finally, a failure occurs when erroneous data affect the delivered service (observed by end users).

A failure may also occur if the system specification (if any) is wrong. Thus, a system may fail because it does not comply with its specification or because its specification does not appropriately describe system expectations.

Since a system is generally composed by a set of interconnected subsystems, the chain *fault–> error –> failure* becomes recursive. If one subsystem fails, its delivered service (wrong) constitutes a fault for the subsystem asking this service.

Dependability aims at preventing introduction of faults in the system, preventing this causal chain and stopping fault propagation.

Following the taxonomy proposed in [11], existing techniques to achieve these objectives are described below.

### 2.4 Techniques for dependability

Techniques used in dependable computing find their origin in the field of fault-tolerance computing. They have been classified into four categories:

- **Fault avoidance:** involved in system construction phase. Its objective is to obtain dependability by preventing the introduction of faults.
- **Fault removal:** how to mitigate frequency and severity of faults. Fault removal is involved either during system construction phase or during its operational phase (maintenance techniques).
- **Fault tolerance:** assumes that even with the most well designed fault avoidance techniques, faults will be present in operational systems, potentially resulting in service failure. The field of fault-tolerance aims at designing techniques to guarantee that the system can continue to deliver correct service despite the presence of faults. Replication – running multiple instances of the same component – represents the most widely adopted technique to implement fault-tolerant systems.
- **Fault forecasting:** evaluates system behavior to catch symptoms of occurrence of failures.

Fault avoidance like fault removal, target the elimination of faults in the system, following dependability *by-design* strategy. Fault tolerance with fault forecasting accept that even with the most carefully designed systems, faults are inevitable. This makes them follow dependability *by-operation* strategy.

To summarize, dependability has four main dimensions: requirements, threats, means and cost. Regarding costs, resources and energy are the two main costs that any computing system seeks to minimize. According to the study made by Lawrence Berkley National Labs, power consumption in datacenters is predicted to grow by 1600% in the few next years and the energy is expected to become the major factor in the Total-Cost-of-Ownership for IT [13]. Thus, it is necessary to integrate energy and resource as constraints when designing dependability enhancement mechanisms. For example, even if replication has long been proved to be an effective means to ensure system dependability, it can be considered as costly when considering energy and resources.

## 3 Programmable networks (SDN/NFV)

SDN and NFV are widely accepted to constitute key building technologies of upcoming network architectures. For the sake of clarity, we first give in the underlying section the necessary background of both, before addressing their implications for dependability.

### 3.1 SDN background

SDN is a design approach, with several definitions: the separation of the data and the control plane, and the programmability of network elements in all networking domains through advanced network abstraction and independently from any southbound protocol

constitute the main defining characteristics of SDN. ONF (Open Networking Foundation) defines [14] SDN as a layered architecture composed of the data plane, the control plane, the application plane, as well as a transversal management plane whose implementation, inner blocks and functionalities are still fuzzy.

- **Data plane:** in pure SDN approach, the data plane is composed of physical switches whose role is limited to forward traffic and collect network statistics ;
- **Control plane:** it may be considered as a Network Operating System (NoS) [15]. It is responsible of modeling and programming the desired network behavior through dynamic installation of flow rules on the data plane elements. When packets fail to match existing flow rules, a notification is sent to the controller to ask for a new rule ;
- **Application plane:** it comprises network applications, which interact with the network controller by querying and modifying network state. The objective of SDN applications is to implement network options or to drive specific use cases [15]. Examples of network functions include QoS management, load balancing or security management. SDN use cases refer to the application of SDN in different domains, such as cloud computing, mobile networks or information content networking (ICN).

SDN controller exports a consistent view of the network to the network applications. OpenFlow is a standardized protocol [14] which enables translation between data plane switches and the controller. OpenFlow is not the ultimate solution for SDN, but it is increasingly deployed on production systems and is potentially becoming the de facto protocol for SDN networks.

To summarize, we consider that SDN controllers benefit from four main properties:

- **Logical centralization of network control:** network decision engine runs in a separated logically centralized entity.
- **Network-state encapsulation:** centralization of network control allows the controller to benefit from a complete and consistent view of the network at any point of time.
- **Mediation:** to avoid conflicts between applications, the controller should also have a role of a mediator between applications [16]. For example, a load balancing application may dictate traffic policy rules which are conflicting with the ones dictated by a security application. Thus, the controller should legitimately prioritize an application policy against another.
- **Interposition:** it is necessary for the controller to interpose between the data plane and the applications to acquire the capacity of preventing an application from enforcing a given network policy. This helps ensuring network consistency during state transition and can protect the network from potential misbehaving applications.

### 3.2 NFV background

According to the ETSI Industry Specification Group (ETSI-ISG) [1], NFV technology leverages hardware virtualization to run network equipments within virtual execution environments. With NFV, network functions are implemented as software modules, thus acquiring the ability to run in multiple execution environments: network nodes, cloud data-centers or even end user devices.

ETSI defines a high NFV architectural framework, composed of four main functional blocks, as depicted in figure 1.

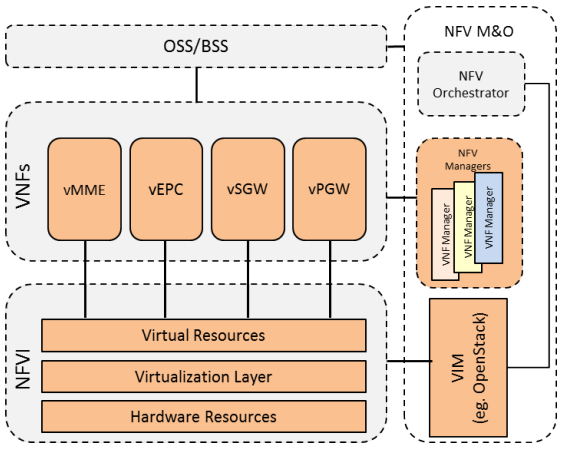



**Fig. 1.** Key functional blocks of an NFV framework [1]

- **Network Function Virtualization Infrastructure (NFVI):** provides the complete execution environment for the VNFs, including the hardware resources (memory, cpu, network and storage) and the virtualization layer which abstracts hardware resources into virtual ones (vMemory, vCPU, vNetwork and vStorage);
- **Virtual Network Functions (vNFs):** consist of any virtual execution environment (VMs or a containers) specifically configured to provide a given network service, like Serving Gateway and Packet Data Gateway S/P-GW or Mobility Management Entity (MME). Execution of network functions within VMs (or containers) empowers them with new properties, essentially scalability and flexibility in VNFs life cycle management. The expected service and performance of VNFs must be the same with the one guaranteed in traditional non virtualized network functions;
- **Virtualized Infrastructure Manager (VIM):** it is responsible of the management of the NFV infrastructure including measurement and forwarding of performance statistics, capacity planing, or troubleshooting of VNFs performance issues. This management layer may be integrated with a cloud management layer such as OpenStack;

– **NFV Managers:** this layer is responsible of the management of the virtual layer (VNFs) such as VNFs life cycle, network interaction among the instantiated VNFs or performance statistics measurement of VNFs. In order to deliver network services, multiple VNFs are instantiated and chained together dynamically, creating VNF Forwarding Graph (VNF-FG). Each VNF is mapped into one or several VMs. VNF-FG might be composed by VMs only or by virtual and physical functions;

NFV is predicted to be a multi-vendor environment. ETSI defines four NFV deployment models: private, exposed, hybrid, and community. While in case of private NFV, network functions are not exposed to subscribers, the other models seek to enable subscribers to access via VNFs standardized APIs to the network functions, the infrastructure or even the management layers.

SDN and NFV are complementary but completely independent technologies. We summarize in table 1, the main common and basic properties of NFV and SDN with different facets.

| **Properties** | **SDN** | **NFV** |
|---|---|---|
| **Programmability** | network behaviour | network function itself |
| **Abstraction** | virtual network topology, control plane, etc | hardware resources |
| **Split model** | data plane from control plane | hardware from software |

**Table 1.** Main properties of SDN and NFV

Characteristics of emerging technologies: resource sharing in the cloud, centralization of network control in SDN, and *softwarisation* of network functions with NFV makes it necessary for telecom operators to reconsider dependability requirements (attributes) in these contexts.

For example, in cloud infrastructures, scalability is an essential requirement. If a cloud management layer fails to be sufficiently scalable, this will result in an availability or reliability degradation of cloud services for end users. Hence, it is necessary for providers to elaborate new metrics and procedures for dependability assessment at their side. Detection of dependability degradation from provider side may enable to repair (increase allocated resources, migrate VMs, etc...) before customers observe the degradation of service performance. In the underlying section, we discuss dependability requirements of SDN/NFV based infrastructures.

## 4 Dependability attributes in SDN/NFV based networks

In this section, we discuss new requirements that should be integrated as dependability attributes in the context of emerging *cloudified* telecom infrastructures.

– **Elasticity:** is one of the most attractive attributes of virtualized infrastructures. Elasticity consists on automatically adjusting allocated resources to VMs, according to their resource consumption changes (memory, cpu, network, disk), by allocating and de-allocating resources such that at any point of time, VM allocated resources are as close as possible to current VM resource demand. Resizing allocated

resources can be achieved either through the readjustment of provided resources to a given VM  vertical elasticity – or by instantiating (or deleting) additional VMs horizontal elasticity–. Elasticity enables providers to optimize resource exploitation of their datacenters and users to pay for only the resources they use.

Elasticity constitutes one of the main drivers for NFV adoption. In current state of the art, most of the network functions identified as relevant candidates for virtualization are the ones with high resource fluctuation, like S/P-GW [17].

Elasticity assessment can be considered from two aspects [18] **i) efficiency:** quantifies the amount of allocated resources to a VM, for processing a given amount of work, **ii) performability:** may be evaluated by quantifying the average speed of scaling, accumulated time in under-provisioned state, and the average amount of under-provisioned resources during an under-provisioned period.

– **Scalability:** is defined as the ability of a system to instantiate additional VMs to deal with an increasing need of resources. Scalability does not consider how allocated resources are tailored to current resource demand. Scalability evaluation might be achieved through quantification of the average amount of time spent to scale-up or to scale-down resources.

– **Error isolation:** physical isolation between multiple execution environments constitutes a natural asset for error confinement. Information exchange (mainly through network) creates a kind of dependence between collaborating components which creates vectors for error propagation. With virtualization and resource sharing, multiple VMs – even if they run independently –, share the same underlying hardware resources, creating a new error propagation vector.

  In the context of NFV, an error in one VNF may spread to another co-resident VNF through multiple error propagation vectors. This makes it necessary to define models to evaluate whether faults in one VM can propagate their effect to a co-resident VM. Moreover, techniques to evaluate the set of VMs to which an error may spread are necessary. Quantification of the level of error isolation offered by a virtualization platform may constitute a differentiating attribute for adopting a given virtualization platform.

– **Performance isolation:** in the context of virtualized infrastructures (hardware virtualization, NFV), besides security, performance degradation constitutes an important issue for multi-tenant shared platforms. Performance isolation means that one VM will not suffer from intensive resource consumption made by another co-resident VM. Lack of performance isolation is due to resource consumption interference, which occurs when multiple VMs compete for same limited resources. Quantifying the level of isolation offered by a virtualization platform is necessary for dependability evaluation. For measuring the level of performance isolation offered by a virtualization platform, a QoS oriented approach is proposed in [19], where isolation is quantified in term of the impact caused by a VM with high resource consumption on the QoS experienced by co-resident VMs.

## 5 Impairments to dependability in SDN and NFV

In this section we discuss the main causes of faults in an SDN/NFV infrastructure. We classify these faults according to the logical layer from the SDN architectural framework (Figure 2) where they can occur.

Based on the observation that cloud computing technologies constitute the main building blocks of NFVI infrastructures, we focus our analysis on design the new issues introduced by current SDN designs which are less documented in the state of the art. Dependability issues of the virtualized network functions are well addressed by the ETSI NFV ISG (NFV Resiliency Requirements [20])

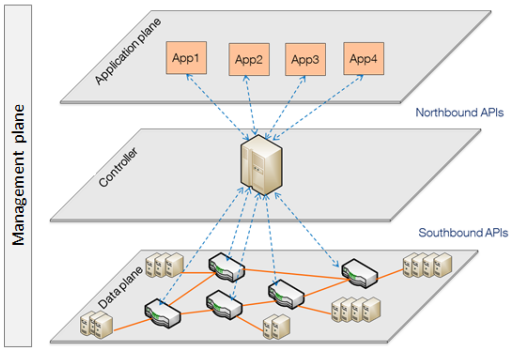



**Fig. 2.** A simplified view of SDN Architecture

### 5.1 Controller

The controller is the cornerstone of SDN architectures. Hence, we consider that the major dependability issue of SDN resides in the centralization of network control. Errors and vulnerabilities in the controller may result in failures impacting the execution of the entire network.

**Single point of failure:** the logical centralization of network control concentrates all the network intelligence within the same entity. However, the controller itself remains a piece of software which is exposed to faults, errors and failures. To avoid situations where the entire network is broken down because of controller failures, the controller must be physically distributed and replicated. Implementing distributed execution of controllers involves dealing with several questions, among them: **i)** how to guarantee that at any point of time, all the controllers have the same consistent view of the network (synchronization), **ii)** how much controllers are needed to guarantee that the SDN network is dependable, **iii)** where, in the network topology, these controllers should be placed to avoid performance issues, **vi)** which is the maximum tolerated distance (in

term of number of nodes) between one controller and its attached network elements such that communication between the controller and network elements will not suffer from performance degradation.

**Scalability:** controllers can rapidly become the bottleneck in the network. When the size of network increases, more requests need to be handled by the controller. For example, with the OpenFlow protocol, the controller handles flows reactively. This means that when an incoming packet does not match any flow entry, a notification is sent to the controller which has to set a new forwarding rule within the network elements. This can possibly introduce significant communication delays depending on controller's performance and its available resources.

**Conflicts in traffic engineering policies:** occur when an application sets a flow policy which contravenes either with existing enforced network policy, or with another candidate policy issued by another application. The controller should implement mechanisms to mediate all exchanged information between the application layer and the data layer. A mediation mechanism is proposed in [16], where a conflict analyser in charge of evaluating whether a candidate rule is conflicting with an existing rule, is added to the FloodLight controller.

**Inconsistency in network topology view** : network topology may change over time, driven by: control applications updates, failures, traffic load evolution, and so on. A wrong or incomplete view of the network can lead controller applications to take ineffective decisions. These decisions may cause the network to deviate from its expected behaviour. Hence, controller's view of the network should be updated as soon as possible after a change is made by any network element [21].

### 5.2 Applications

In most current versions of SDN controllers, SDN applications run as kernel modules. Hence, the dependability of these applications is strongly bounded to the one of the controller itself. However, the basic principal of SDN is to make it possible to run multiple applications within the same controller. Multi-tenant controllers have to deal with two main issues [16]:

- **Accountability:** co-existence of multiple applications, possibly under the control of different tenants, within the same controller raises the need for implementing application accountability mechanisms. This can help the controller to know which application has issued a given flow policy. Accountability can also empower SDN controllers with troubleshooting capabilities.
- **Privilege separation:** current controllers implement control applications as kernel modules running in the same process space with the controller. This is contrary to elementary dependability principles, privileging strong separation of roles.

### 5.3 Data plane

**Virtualization of data plane:** when SDN is empowered by NFV, network elements composing the data plane are virtualized and possibly moved to cloud datacenters. Virtualization of network functions involves considering implications of virtualization on **i) performance of execution of the VNFs:** impacts of virtualization overhead on performances experienced by the network functions and the delivered services, **ii) hardware faults:** standard servers are more error prone than specialized network equipments, **iii) software faults:** virtualization layer, VMs Operating System or the VNFs themselves.

**Open-flow derived attacks and flaws:** OpenFlow protocol itself contains multiple design and implementation flaws. A security study of OpenFlow is presented in [22], where authors demonstrate multiple IP-spirit attacks on SDN networks, exploiting OpenFlow flaws. Host Location Hijacking with Link Fabrications attacks are two examples.

- **Host Location Hijacking attack**: this attack exploits an OpenFlow service, namely the Host Tracking Service, supported by OpenFlow controllers for dynamic update of hosts location. If a Packet-In message sent to the controller contains new host location, a host profile table is updated with this new location. Due to the lack of authentication, this service is vulnerable to host location hijacking attacks. Such an attack is demonstrated in [22], where an attacker succeeds to hijack traffic addressed to a target host by impersonating its IP address. The attacker misleads network devices and makes them send to the controller, *Packet-In* messages with the target host IP address associated to a new host location. When receiving this message, the controller updates the host profile table with new location, which is actually the attacker's location. Thus, all new traffic addressed to the target host will be hijacked by the attacker.
- **Link Fabrication attack:** this attack exploits the Link Discovery service, supported by OpenFlow controllers to dynamically discover links. OpenFlow uses LLDP packets (Link Layer Discovery Protocol) for Link discovery. Two flaws in link discovery service, at the origin of the link fabrication attack, have been identified [22]. First, origin and integrity of LLDP packets is not ensured. This allows attackers to forge fake LLDP packets. Second, any switch, host or VM is allowed to relay an LLDP packet, contributing to exacerbate effects of this attack. In [22], authors demonstrate an LLDP injection attack where an attacker creates and relays fake internal links between two switches.

Host hijacking and link fabrication attacks can be leveraged to execute more sophisticated attacks such as Man-In-The-Middle, black-hole, or denial of service attacks.

**Network visibility poisoning:** as mentioned above, topology consistency and correctness is a major requirement for the controller. All SDN applications rely on the network topology presented by the controller. Leveraging the already described Link fabrication and host location spoofing attacks, an attacker is able to poison the network topology view of the controller, thus, the one of all network management applications.

**Denial of Service:** as in traditional networks, OpenFlow switches are also vulnerable to denial of service attacks. An OpenFlow switch, which receives intensive incoming traffic, will delete a part or the whole traffic. A denial of service attack exploiting Spanning Tree Protocol (SPT) has been demonstrated in [22]. Existing versions of OpenFlow controllers support SPT. With SPT, when a topology update occurs, the SPT service is triggered to block redundant ports. When this option is exploited with Link fabrication attack, SPT can block ports in use, resulting in denial of service against OpenFlow switches.

The success of these attacks demonstrates that although data plane does not contain any decision capacity, attacks on network elements, can impact the controller decisions which represents a real threat for the entire network.

## 6 Future research

In this paper, we discussed the dependability challenges introduced by SDN and NFV based networks. According to the observations made along the paper, we argue that there is a crucial need for reconsidering dependability objectives and techniques, to make them more specific to the emerging infrastructures. As future research work, we address dependability challenges from two main perspectives: dependability **by-design** and dependability **by-operation**.

### 6.1 Dependability by-design

Dependability by-design is involved during the system specification and development phases. Because of the critical position of SDN controllers, dependability avoidance techniques should be applied to avoid the introduction of faults during the specification and the development phases of the SDN controller. As a first step, it is necessary to define formal properties that must be verified by SDN controllers. A second step should target applying formal methods on existing SDN controllers to evaluate their operational behaviour.

### 6.2 Dependability by-operation

Dependability by-operation observes the system behavior in run-time in order to detect any deviation from its desired/expected behavior. In the context of SDN/NFV, monitoring frameworks and metrics, as cornerstones of supervision, need to be revised to include the under-definition requirements of softwarisation. In this regards, we are interested by three directions:

– **Benchmarking through fault injection:** for dependability assessment, we first need to understand fault scenarios which can impact performance of an SDN/NFV infrastructure. In cloud infrastructures, most of fault injection techniques are focused on hardware faults such as I/O interruption faults, or faults affecting CPU, memory, network and disk. In SDN/NFV environments, it is necessary to integrate new faults for the evaluation of the SDN controllers, the management layers and

third party tools. Benchmarking through fault injection can allow forecasting the occurrence of critical failures and also helps for the definition of more appropriate metrics to detect and prevent the occurrence of failures.

- **Modeling of inter-dependencies between VNFs in a VNF-FG service:** we seek to identify, characterize and model, new error propagation patterns within a VNF-FG service. The objective is to evaluate the impact of the virtualization layer on the exposure of network services to error propagation.
- **Autonomic diagnosis and recovery**
  - Proposition of adaptable and learning based preventive mechanisms to avoid attacks and failures;
  - Elaboration of algorithms allowing cross-layer (SDN, NFV Layers) correlation of alarms to detect the root cause;
  - Management protocols and interfaces especially for security, fault and performance need to be revised and defined in the context of SDN and NFV while adding features ensuring dependability.

## References


1. http://www.etsi.org/.
2. Justine S, Shaddi H, Colin S, Arvind K, Sylvia R, and Vyas S. Making middleboxes someone else's problem: Network processing as a cloud service. In *In proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '12, NY, USA, 2012. ACM.
3. C. Dixon, D. Olshefski, V. Jain, C. DeCusatis, W. Felter, J. Carter, M. Banikazemi, V. Mann, J.M. Tracey, and R. Recio. Software defined networking to support the software defined environment. *In IBM Journal of Research and Development*, 58, March 2014.
4. John E. Hosford. Mesures of dependability. *Operations Research*, 1960.
5. W. C. Carter. A time for reflection. In *In proceeding of the Symposium on Fault-Tolerant Computing*, 1982.
6. J.C. Laprie, A. Avizienis, and H. Kopetz, editors. *Dependability: Basic Concepts and Terminology*. In Springer-Verlag New York, Inc., NJ, USA, 1992.
7. M. Al-Kuwaiti, N. Kyriakopoulos, and S. Hussein. A comparative analysis of network dependability, fault-tolerance, reliability, security, and survivability. *In Communications Surveys Tutorials, IEEE*, 11, Second 2009.
8. John C. Knight and Kevin J. Sullivan. On the definition of survivability. technical report. Technical report, University of Virginia. Department of Computer Science, 2000.
9. Jean claude Laprie. From dependability to resilience. In *In Proceeding of the 38th IEEE/IFIP International conference on Dependable Systems and Networks*, 2008.
10. James P.G. Sterbenz, David Hutchison, Egemen K. Ãetinkaya, Abdul Jabbar, Justin P. Rohrer, Marcus SchÃller, and Paul Smith. Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines. *Computer Networks*, 54(8):1245 – 1265, 2010. Resilient and Survivable networks.
11. A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *In proceeding of the IEEE Transactions on Dependable and Secure Computing,*, Jan 2004.
12. John Knight. *Fundamentals of Dependable Computing for Software Engineers*. Chapman & Hall, 1st edition, 2012.

13. S. Kounev, P Reinecke, F Brosig, JT. Bradley, Kh Joshi, V Babka, S T. Gilmore, and A Stefanek. Providing Dependability and Resilience in the Cloud: Challenges and Opportunities. pages 65–81. Springer Verlag, June 2012.
14. https://www.opennetworking.org/.
15. Y. Jarraya, T. Madi, and M. Debbabi. A survey and a layered taxonomy of software-defined networking. *In proceeding of the IEEE Communications Surveys Tutorials*, 2014.
16. P Porras, S. Cheung, Fong M, Skinner . K, and Yegneswaran .V. Securing the software-defined network control layer. In *Proceedings of the 2015 Network and Distributed System Security Symposium (NDSS)*, February 2015.
17. D. Moldovan, G. Copil, Hong-Linh Truong, and S. Dustdar. Mela: Monitoring and analyzing elasticity of cloud services. In *In proceeding of the 5th IEEE International Conference on cloud Computing Technology and Science (CloudCom 13)*, 2013.
18. Nikolas Roman Herbst, Samuel Kounev, and Ralf Reussner. Elasticity in cloud computing: What it is, and what it is not. In *In proceedings of the 10th International Conference on Autonomic Computing (ICAC 13)*, San Jose, CA, 2013. USENIX.
19. Rouven K, Christof M, and Samuel K. Metrics and techniques for quantifying performance isolation in cloud environments. *Science of Computer Programming*, 2014. Special Issue on Component-Based Software Engineering and Software Architecture.
20. ETSI. *Network Function Virtualisation (NFV); Resiliency RequirementsNetwork Function Virtualisation (NFV); Resiliency Requirements*, 2015.
21. Wenxuan Z, Dong J, Jason C, Matthew C, and P. Brighten G. Enforcing customizable consistency properties in software-defined networks. In *In proceeding of the 12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*, Oakland, CA, 2015. USENIX Association.
22. Hong S, Xu L, Wang H, and Gu G. Poisoning network visibility in software-defined networks: New attacks and countermeasures. In *In Proceedings of the 2015 Annual Network and Distributed System Security Symposium (NDSS'15)*, February 2015.

# De l'hameçonnage ciblé à la compromission totale du domaine : démonstration, état des lieux et comment le concept de cyber résilience peut aider à limiter les impacts

Johanne Ulloa

johanne_ulloa@trendmicro.fr

Trend Micro

## 1 Introduction

La problématique des attaques ciblées est difficile à appréhender par des personnes non expertes du domaine. Cette présentation a donc pour but de livrer des leviers de compréhension à un public non spécialisé en essayant de vulgariser au maximum.

Pour ce faire, nous allons tout d'abord expliquer quelles sont les différentes phases d'une attaque ciblée. Notre démonstration permettra ensuite de mettre en exergue un scénario d'attaque-type dans lequel seront illustrées ces différentes phases. Nous constaterons alors que la plupart des attaques de ce type n'est pas très complexe et serons amenés à nous poser la question suivante : qu'est-ce qui a changé au cours de ces dernières années ?

C'est ici que nous introduirons le concept de cyber résilience, à travers deux images : la citadelle et le supermarché. Enfin, nous justifierons la nécessité de cette résilience et proposerons des méthodes et techniques permettant d'implanter des mécanismes fournissant de la résilience.

## 2 Description des principales phases de l'attaque ciblée

### 2.1 Prise d'informations sur la cible

Qui est la cible ? Quels sont ses centres d'intérêt ? Avec qui la cible est-elle en relation ? Bien souvent, ces informations sont disponibles sur les réseaux sociaux professionnels, qui s'avèrent être une très bonne source d'information pour les attaquants. Dans une approche d'hameçonnage ciblé, des informations pertinentes serviront à l'attaquant pour convaincre sa cible de cliquer sur la pièce jointe du mail qui lui sera envoyé.

### 2.2 Première compromission

En cliquant sur la pièce jointe, plusieurs scénarios sont possibles :

- Cette pièce jointe est tout simplement une archive qui contient un fichier exécutable et le malware s'installe sur la machine de cible (ce scénario arrive plus souvent qu'on ne le pense)

- Le fichier exploite une vulnérabilité de l'environnement de l'utilisateur et permet de télécharger un malware

- Le fichier contient une macro qui sera exécutée avec l'accord de l'utilisateur. L'objectif étant d'installer sur la machine de la cible un « RAT » (Remote Access Tool) qui n'est, ni plus ni moins, qu'un logiciel d'administration à distance avec en sus, des fonctionnalités d'espionnage

### 2.3 Communication avec le serveur de contrôle de l'attaquant

Une fois le RAT en place, une communication de la machine cible avec la machine de l'attaquant va être établie. Cette communication est initialisée par la machine de la cible. Elle est donc très souvent autorisée par les mécanismes de filtrage réseau.

### 2.4 Mouvement latéral

Une fois le RAT en place, l'attaquant peut cibler d'autres machines sur le réseau interne. Toutes les machines accessibles depuis la machine compromise sont potentiellement à la portée de l'attaquant. Durant cette phase, on observe des exploitations de vulnérabilités qui permettent d'obtenir des élévations de privilèges et des login/mots de passe. On observe également l'utilisation d'outils légitimes tels que psexec. A ce stade, l'objectif pour l'attaquant est d'obtenir des données auxquelles il n'avait pas accès ou bien d'étendre sa compromission de manière à pouvoir pérenniser son accès au système d'information.

### 2.5 Exfiltration de données

Reste ensuite à l'attaquant à exfiltrer les données. Cette phase peut être réalisée grâce au RAT de l'attaquant. D'autres méthodes classiques sont également possibles : http/HTTPS, FTP/SFTP, tunnels DNS,…

## 3 Démonstration

Scénario de la démonstration :

- La cible de l'attaquant est Obiwan Kanobi

- La prise d'information de l'attaquant permet de déterminer qu'Obiwan Kanobi s'intéresse aux sabres laser. L'attaquant se servira de cette information pour rédiger son mail.

- Création d'un fichier piégé au format RTF

- Transmission de la charge par mail

- Prise de contrôle de la machine de la personne ciblée

- Démonstration des possibilités de contrôle de meterpreter

o Récupération de fichiers

o Upload de fichiers

o Capture d'écrans

o Keyloger

- Mouvement latéral

o Escalade de privilège sur la machine d'Obiwan Kanobi

o Récupération des éléments nécessaires pour exploiter la vulnérabilité ms14-068: Nom du domaine, Adresse IP du contrôleur de domaine, SID utilisateur, mot de passe de l'utilisateur

o Exploitation de la vulnérabilité et récupération du ticket Kerberos donnant les droits d'accès de l'administrateur du domaine

o Compromission du contrôleur de domaine avec un meterpreter

## 4 Etat des lieux

- Le rôle du défenseur est de plus en plus complexe suite à un certain nombre de changements :

o Les systèmes d'information sont beaucoup plus complexes, d'autant que les utilisateurs ont désormais tendance à connecter leur propre matériel, ne serait-ce que leurs smartphones, au réseau de l'entreprise (BYOD).

Les métiers ont besoin de plus de fonctionnalités, de sécurité et de redondance. Pour satisfaire ces besoins, de nombreux équipements et processus sont mis en place. Cela tend à complexifier la gestion du système d'information et, plus largement, à étendre la surface de vulnérabilité.

En plus des environnements IT traditionnels, les entreprises doivent gérer des objets et des équipements industriels connectés

o Données décentralisées :

Par le passé, les données étaient très centralisées et disponibles uniquement depuis le réseau interne. Les phénomènes de Cloud et de mobilité font que les données sont de plus en plus dispersées.

o Nombreux échanges avec l'extérieur :

Par le passé, les échanges avec l'extérieur étaient faibles, mais sont désormais très nombreux : applications Web, interaction machines-machines via des API, Web Services…

o Professionnalisation des cyber-délinquants (outils d'attaque beaucoup plus simples à utiliser) :

Par le passé, il fallait un niveau compétence technique extrêmement fort pour mener une attaque comme celle qui qui vient d'être présentée. Aujourd'hui, de nombreux outils sont disponibles et des tutoriels existent sous de nombreux formats. On voit également l'émergence de CaaS (Crime As A Service), par exemple une plateforme de ransomware as a service récemment mise en œuvre. Et il est possible pour des personnes n'ayant aucune compétence de distribuer des ransomware dont ils vont partager les bénéfices avec les offreurs de la plate-forme.

o Les attaquants contournent très facilement les lignes de défense traditionnelles. Comment ?

- Les antimalware traditionnels reposent sur des mécanismes de signatures. A chaque fichier correspond une signature. S'il y a une correspondance entre la signature du fichier et l'une des signatures contenue dans la base de l'antimalware, le fichier est reconnu comme étant malveillant. Les attaquants contournent ces mécanismes en personnalisant leurs charges à l'aide de « Crypter ».

- Utilisation de Crypters/Packers :

Dans les grandes lignes, un « crypter » est un logiciel permettant de modifier la signature du fichier tout en conservant ses propriétés initiales. Il existe des logiciels prêts à l'emploi disponible sur Internet avec des offres de support optionnelles.

o Les attaquants exploitent des vulnérabilités souvent connues, mais que les entreprises ont du mal à patcher.

Dès lors qu'une vulnérabilité est découverte sur un système ou un applicatif, s'ouvre une fenêtre d'exposition qui correspond à la période de temps pendant laquelle le système ou l'applicatif en question est vulnérable. Cette fenêtre sera fermée dès lors que le système sera patché.

Parfois, aucun patch n'est disponible alors même que la vulnérabilité est connue et exploitable. Durant cette période, les choix se limitent à assumer le risque ou retirer le système de la production. Dans un très grand nombre de cas, le fait qu'un patch soit rendu disponible ne permet pas de fermer la fenêtre d'exposition, car le patch doit être évalué avant d'être mis en production. Il arrive aussi que le patch ne puisse pas être appliqué, par exemple parce que le système perdrait sa conformité (très fréquent dans le domaine du bio médical). Des problèmes de compatibilité de versions empêchent également de pouvoir patcher dans des délais raisonnables. Les fenêtres d'exposition tendent donc à s'allonger, laissant ainsi le champ libre aux attaquants.

## 5 La résilience induit un changement de posture

Le concept de résilience peut aider à mieux se défendre. Avant tout, il convient de définir brièvement ce qu'est la résilience : revenir rapidement à un état initial après un choc ou une altération.

La notion de résilience prend donc d'emblée en considération que le choc va survenir. Il induit ainsi un changement de posture puisque nous étions jusqu'ici concentrés sur le fait d'éviter le risque. Il faut maintenant accepter que des compromissions vont inévitablement survenir et il faut donc s'y préparer.

On peut représenter ce changement de paradigme en faisant une comparaison entre ce qui pourrait représenter le modèle basé sur la résistance (on prendra l'image de la citadelle) et le modèle basé sur la résilience (image du supermarché).

- Modèle de la citadelle (résistance) :

Ce modèle est très fortement orienté sur la protection périmétrique. L'idée est d'empiler des lignes de défense pour empêcher l'attaquant de compromettre le système d'information. Ce modèle a perdu de son efficacité, car les données sont très décentralisées, les échanges avec l'extérieur sont nombreux et les attaquants arrivent relativement facilement à contourner les moyens de défense traditionnels qui sont essentiellement basés sur de la défense périmétrique. Il ne s'agit pas de dire que les firewalls ou les anti-malware classiques ne sont plus utiles (ils sont toujours des éléments indispensables à la sécurité des systèmes). On peut toutefois dire qu'ils ne sont plus suffisants à eux seuls pour assurer un niveau de sécurité satisfaisant.

- Modèle du supermarché (résilience) :

Le supermarché adopte d'emblée le principe de la résilience en admettant que les vols sont inévitables. Par conséquent, un certain nombre de mesures est systématiquement mis en œuvre :

o Classification des données : Si l'on compare un supermarché à un système d'information, on peut comparer des lames de rasoir à de la donnée sensible. En effet, elles sont petites (donc facile à voler) et leur coût est relativement élevé. Elles sont donc une cible de choix pour les voleurs. Ce type de marchandise est mieux protégée que les gâteaux secs (donnée non sensible). On observe ici que c'est la sensibilité de la donnée qui permet de déterminer le niveau de protection. Il est bien entendu hors de question de mettre en œuvre des mesures de sécurité similaires sur les gâteaux secs, tout simplement parce que cela coûterait trop cher. Encore faut-il être en mesure de déterminer ce qui est sensible de ce qui ne l'est pas. Cela demande un très gros effort à réaliser pour les entreprises, d'autant plus que cette classification doit être réalisée au fil du temps. De plus, par opposition au monde physique, une donnée peut être sensible pendant une période et devenir publique par la suite. Dans le monde physique, on ne voit jamais de lames de rasoir se transformer en gâteaux sec…

La classification demeure un exercice complexe. Elle n'en demeure pas moins nécessaire pour appliquer des politiques de sécurité de manière granulaire, ce qui permet de diminuer les coûts de gestion.

o Gagner en visibilité : dans le modèle de la citadelle, on ne cherche pas à savoir ce qui s'y passe, car on se dit que personne ne va rentrer. Parfois, quelques caméras sont présentes, mais leur champ optique n'est pas optimum (IDS basée sur des signatures) et il est rare que des personnes soient en charge de vérifier ce qu'il s'y passe à temps plein.

Pour le supermarché, les caméras de surveillance sont un élément clé. Les caméras sont donc nombreuses et perfectionnées (champs de vision importants, possibilité de zoomer, elles sont mobiles, etc….). Des agents de surveillance (SOC) scrutent à longueur de journée ces caméras. Ils ont donc l'habitude de détecter les comportements suspects.

o Réponse à Incident : Lorsqu'un vol est constaté, des agents sont envoyés sur place pour intercepter le voleur. Dans le modèle de la citadelle, étant donné que les moyens de détection sont manquants, les incidents ne sont pas ou sont mal détectés. La détection est la première étape de la réponse à incident.

- Résilience ≠ résistance :

Il existe une nuance entre résistance et résilience. La résistance tend à minimiser l'occurrence du risque, alors que la résilience tend à minimiser l'impact.

Il ne s'agit pas de remettre totalement en cause les infrastructures existantes en disant que les firewalls, les antimalware classiques ne sont plus utiles (ils sont toujours des éléments indispensables à la sécurité des systèmes). On peut toutefois dire qu'ils ne sont plus suffisants à eux seuls pour assurer un niveau de sécurité satisfaisant.

Idéalement, Le modèle de la citadelle doit donc être complété par le modèle du supermarché.

## 6 Pourquoi entrer en « cyber résilience »

Empêcher absolument l'occurrence du risque est devenu totalement illusoire. Il faut donc être convaincu que le choc va survenir et s'y préparer, afin de retrouver une activité normale le plus rapidement possible. Adopter les principes de cyber résilience permet de diminuer les risques et les impacts :

o Des fraudes (ex : arnaque au président) :

Fraude qui utilise l'ingénierie sociale pour forcer un employé à réaliser un virement dans l'urgence sur un compte à l'étranger. Ce qui est clé dans cette fraude, c'est la connaissance de la structure par l'attaquant. Cette connaissance lui permettra d'actionner les bons leviers pour faire croire à l'employé que l'ordre vient bien de la Direction Générale et qu'il faut l'exécuter dans l'urgence. Cette prise de connaissance est souvent réalisée avec la méthodologie de l'attaque ciblée décrite plus haut.

o Du cyber espionnage (préserver son capital informationnel) :

Rappelons que, sur le moment, le cyber-espionnage est totalement indolore pour la victime. En effet, l'attaquant restera le plus discret possible (tout dysfonctionnement pouvant mener à des investigations qui révéleraient sa présence). Il arrive que les attaquants corrigent eux-mêmes les systèmes en appliquant des patchs de façon à garantir l'exclusivité de la compromission. Rappelons aussi que, contrairement au monde physique où l'on peut se rendre compte facilement d'un vol, il n'en va pas de même dans le monde numérique où les données sont duplicables.

o De la divulgation massive de données et du chantage à la divulgation :

De nouveaux risques émergent. Nous avons vu récemment des attaques visant à divulguer des données de façon massive dans le but de nuire à l'image de l'entreprise (Sony, Hacking team, Ashley Madison). On voit également émerger des chantages à la divulgation : les attaquants compromettent par exemple la base client de l'entreprise et demandent une rançon pour ne pas divulguer ces données sur Internet.

o Sur l'accessibilité des données (ransomware) :

Les ransomware sont devenus un business très lucratif pour certains cybercriminels. Les attaquants n'hésitent donc pas à passer du temps à la customisation des malware de façon à ce que ceux-ci ne soient pas détectés. Cette phase de personnalisation est coûteuse pour les attaquants, mais ils savent qu'ils auront un retour sur investissement significatif.

## 7 Entrer en Cyber résilience (se préparer au choc !)

- Aspects techniques :

o Détecter :

La détection des malware avancés est devenue un point clé. Dans les grandes lignes, plusieurs types d'analyses permettent de déterminer qu'un fichier est un malware. Pour illustrer ces méthodes, nous allons prendre l'image d'un colis dans lequel on doit détecter si une bombe est présente ou non.

- Analyse statique :

Consiste à comparer le fichier à une base de signatures connue. (on vérifie dans la base strictement identique si un colis déjà répertorié est présent)

- Analyse heuristique :

Analyse basée sur le comportement. Le colis va être examiné via différentes méthodes : Analyse au rayon X, détection de traces d'explosifs. Est-ce qu'un bruit d'horloge émane du colis ? Ce type d'analyse permet d'affiner la détection en repérant des fichiers qui ne sont pas encore connus.

- Analyse par signature « multi critères» :

Cette analyse se sert de signatures fines définies de façon à détecter, non pas des souches uniques de malware, mais également des versions proches et légèrement modifiées. Le format de règle YARA illustre parfaitement ce genre de signatures. Le colis présente ainsi des similitudes avec un autre colis, qui permet d'affirmer qu'il s'agit d'une variante du colis déjà connu.

- Analyse dynamique :

Le colis est ouvert et l'on observe ce qu'il se passe. Le fichier est envoyé dans une machine virtuelle (appelée sandbox). On observe la séquence d'exécution pour déterminer s'il s'agit d'un malware ou non.

- Détecter la première compromission :

L'analyse heuristique et dynamique des pièces jointes dans les flux mail est une priorité, car le premier vecteur de compromission est le mail.

- Détecter les communications avec les C&C :

Lors de l'analyse dynamique, on pourra observer que le malware établit une communication avec une adresse IP. Une sonde placée entre Internet et les utilisateurs permettra de déterminer si des machines du réseau communiquent avec ce C&C ou d'autres C&C connus.

- Détecter les mouvements latéraux :

Il est important d'être en mesure de déterminer si des tentatives d'exploitation de vulnérabilités ou des élévations de privilèges sont réalisées sur le réseau interne. Il est également important de pouvoir détecter des attaques "brute force" ou encore des machines communiquant avec le réseau TOR, envoyant des mails alors qu'elles ne sont

pas un serveur mail, envoyant des réponses DNS alors que ce ne sont pas des serveurs DNS. En outre, il faut se doter de capacités permettant de surveiller les utilisations d'outils légitimes tels que PsExec sur les postes de travail.

- Détecter l'exfiltration de données massive :

Etre notifié des fichiers uploadés via FTP, HTTP ou HTTPS. L'objectif n'étant pas de détecter toutes les exfiltrations (tâche qui semble impossible), mais d'empêcher une exfiltration massive.

o Exploitation de vulnérabilités :

Pour limiter le risque d'exploitation de vulnérabilités, on pourra utiliser des solutions de type « virtual patching ». Le principe est le suivant : pour chaque vulnérabilité, une sorte de signature comportementale qui caractérise l'exploitation de la vulnérabilité sur le réseau est réalisée. Si l'on observe une ou des trames correspondant à cette « signature », les paquets sont bloqués.

L'objectif est de réduire la fenêtre d'exposition et de pouvoir planifier la mise en œuvre des patchs sans avoir à subir l'urgence. Ces mécanismes de virtual patching sont présents sur les IPS. Toutefois, les IPS agissent au niveau périmétrique. Il faut donc que l'attaque passe à travers l'IPS pour que celui-ci puisse agir et, lorsque cela est possible, préférer les HIPS (installés sur chacune des machines) qui permettent de limiter le risque des attaques susceptibles d'être perpétrées depuis le réseau interne.

o S'assurer de l'intégrité :

Lorsque la compromission surviendra, il sera très utile de pouvoir déterminer avec certitude si une machine a été impactée ou pas. Les solutions de contrôle d'intégrité de fichiers permettent de répondre à cette problématique. La classification de données aidera à déterminer sur quelles machines mettre en œuvre ces mécanismes.

- Aspects organisationnels :

La détection mise en œuvre va faire remonter un certain nombre d'incidents qui devront être traités selon un processus de réponse à incident défini. Par exemple :

o Qualification :

Pour chaque incident il convient de réaliser une levée de doute pour confirmer ou infirmer qu'il s'agit d'un incident. Si c'est le cas, il faudra s'efforcer de préserver les traces.

o Limitation de l'impact :

Selon le contexte, on pourra par exemple isoler les machines infectées, limiter ou bloquer l'accès aux données sensibles.

o Un certain nombre d'éléments de la réponse pourront être traités de manière automatique pour limiter l'impact :

- Diffusion des signatures custom :

Les mécanismes de détection, par exemple le sandboxing, doivent être en mesure de diffuser les signatures custom sur l'ensemble des postes.

- Blocage automatique des communications avec des C&C :

Les mécanismes de détection doivent être en mesure de diffuser les adresses IP des C&C sur l'ensemble des postes et/ou sur les éléments filtrants (Firewall et/ou proxy).

o Investiguer :

Déterminer quel est le vecteur de compromission ? Quel est l'impact ? Que s'est-il passé ? Qu'a réalisé le malware sur la machine cible ? Est-ce que d'autres machines ont été en contact avec ce malware ? Les sondes réseau et le sandboxing permettent de révéler des IOC (Indicator Of Compromise) : Hashs, adresse IP, URL, nom de domaine, adresses mails, clés de registre, présence de certains fichiers etc. qui sont en relation avec un ou plusieurs malware. Il est important de pouvoir mener des investigations sur son parc de manière industrielle pour déterminer quelles machines ont présenté ces IOC.

o Former et sensibiliser le personnel :

- Sensibilisation à la sécurité :

La sensibilisation est un point crucial. Les utilisateurs doivent connaître les mécanismes de compromission utilisés par les attaquants de manière à élever leur vigilance. Toutefois, des préconisations extrêmes sont parfois contre-productives : il est souvent indiqué aux utilisateurs de ne pas ouvrir les pièces jointes provenant d'expéditeurs qu'ils ne connaissent pas. Toutefois, un attaquant peut facilement usurper une identité et, surtout, il parait difficile de demander à un responsable des ressources humaines de ne pas ouvrir les CV qu'il reçoit ou à un comptable de ne pas ouvrir les factures. Il est donc logique que l'utilisateur ne suive pas des mesures qui l'empêchent de travailler.

- Mise en situation technique :

Tout comme les pompiers qui s'entrainent à éteindre des feux, les équipes techniques ont besoin d'être entrainées à détecter et à répondre aux incidents de sécurité. Parmi les possibilités, on pourra faire réaliser des pentest en mode blue team/Red Team : des consultants accompagnent les équipes internes pendant le pentest. Ils les aident à comprendre ce qu'il se passe afin d'avoir les bons réflexes durant une attaque réelle. Des exercices sur plateformes de simulation peuvent également être réalisés. De nouveaux types de simulations, les simulations d'APT, apparaissent depuis quelques années. Elles sont souvent plus parlantes que des sessions de pentest.

o Mettre en place des processus de gestion de crise :

En cas de crise il faudra agir vite. Il convient donc de ne pas perdre de temps sur les aspects logistiques. Des questions doivent être posées avant la crise : Qui sont les experts, Comment les joindre ? Qui communique ? Comment communiquer ?

o Mettre à l'épreuve ces processus par de la mise en situation :

Vous apprenez que votre base client est disponible sur les réseaux peer to peer. Comment réagissez-vous ? Des sociétés de service peuvent aider à réaliser ce type d'entrainement pour avoir les bons réflexes lorsque le choc surviendra.

o Anticiper les aspects judiciaires :

Disposer d'un conseil juridique qui maîtrise les aspects de cyber-sécurité.

Avoir mis en place des processus de collecte de preuves.

## 8 Conclusion

Dans un contexte où la surface d'exposition a été démultipliée, où les données sont complètement dispersées et où les attaquants se sont largement professionnalisés, il est totalement illusoire de penser que les systèmes sont à l'abri des compromissions. Il semble donc fondamental d'adapter sa stratégie de défense en conséquence en adoptant le principe de la résilience qui consiste avant tout à considérer que l'impact va inévitablement survenir. Il faut donc s'y préparer de façon à retrouver une activité normale le plus rapidement possible. La mise en œuvre de processus et d'outils qui prennent en compte la notion de résilience est indispensable. Il convient de progresser sur l'axe de la détection (être en mesure de capter des signes de compression le plus rapidement possible) de façon à minimiser les impacts. La détection devra être complétée par des mécanismes de remédiation automatisés. Pour le moment, rares sont les solutions qui permettent de réagir automatiquement suite à une détection. Souvent parce que les briques proviennent d'éditeurs différents, mais également par manque d'API ouvertes et documentées. C'est pourtant cette remédiation automatisée qui pourra réduire le délai entre la détection et la réaction en permettant ainsi de réduire les impacts.

## Bibliographie

Ouvrages :

PERNET, Cédric. Sécurité et espionnage informatique. Editions Eyrolles. 2014

RASCAGNERES, Paul. Malwares identification, analyse et éradication. Editions ENI. 2013

Outils et services :

HYNESIM. Plateforme de simulation pour l'entrainement en cyber sécurité. http://www.hynesim.com/

SEKOIA. Pentest en mode blue team et read team. http://www.sek-oia.fr/

PROVADYS. Préparation à la crise par la mise en situation. http://www.provadys.com/

TREND MICRO. Sonde de détection (Deep Discovery) et Solution de virtual patching (Deep Security) http://www.trendmicro.fr

Etude :

PROVADYS. Cyber attaques : où en sont les entreprises françaises. http://www.observatoire-fic.com/wp-content/uploads/2013/12/LB-Provadys.pdf

# Architecture des systèmes d'automatisation des postes résiliente aux attaques des trames GOOSE

M. Kabir-Querrec[1,2], S. Mocanu[1], P. Bellemain[1,] J.-M. Thiriet[1], E. Savary[2]
*1) Univ. Grenoble Alpes, GIPSA-lab, F-38000 Grenoble, France*
*CNRS, GIPSA-lab, F-38000 Grenoble, France*
*2) Euro-System, F-38760 Varces, France*

**Résumé :** Notre travail concerne la spécification et la mise en œuvre d'un système d'automatisation des postes électriques conformes à la norme IEC 61850 capable de fonctionner en présence d'attaques sur les systèmes de communication temps-réel (communication GOOSE). Notre architecture repose sur trois concepts : la réalisation des sondes capables de détecter les attaques sur les trames GOOSE, la remontée des alertes au SCADA et la réalisation d'une commande des équipements de terrain intégrant l'information de cybersécurité.

**Mots clés :** IEC 61850 ; Attaques GOOSE ; Protection électrique ; Architecture résiliente ; SCADA ; SAS.

## I. Introduction

### 1. L'automatisme des postes IEC 61850

Les réseaux de distribution électrique modernes (smart-grids) intègrent des technologies intelligentes dont l'objectif est de régler en temps-réel la production de l'électricité ainsi que sa distribution afin d'optimiser le comportement global du système (satisfaction de la demande *versus* minimisation des pertes). Cette fonctionnalité de contrôle en continu du comportement du réseau est réalisée via l'automatisme du réseau de distribution qui implémente les fonctions classiques d'acquisition de données, supervision et commande (*Supervisory Control and Data Acquisition – SCADA*). Etant données la taille et la complexité du réseau électrique un contrôle centralisé n'est pas envisageable. L'intelligence du réseau repose sur des fonctions de contrôle distribuées. La brique de base de ce système de contrôle distribué est le *système d'automatisation de poste (Substation Automation System – SAS)*. Le standard IEC 61850 définit le poste comme étant "un ensemble d'équipements électriques interconnectés avec des fonctionnalités communes". Le système d'automatisation de poste est constitué de l'ensemble des équipements informatiques réalisant des fonctions de mesure, contrôle, protection électrique. Ces équipements sont appelés *Equipements Electroniques Intelligents (Intelligent Electronic Device – IED)*. La réalisation de certaines fonctions complexes dans les SAS, notamment des fonctions de protection électrique, nécessitent l'utilisation des informations fournies par des IED distincts (par exemple des mesures de courant en plusieurs points de réseau). Il s'ensuit que la réalisation des SAS nécessite un système de communication adéquat.

### 2. SAS et sûreté de fonctionnement

La définition d'un tel système de communication est l'un des objectifs de la norme IEC 61850. Les deux grands atouts de ce standard (première édition 2003) sont l'interopérabilité des systèmes de contrôle, qui met fin à la dépendance d'une infrastructure envers un unique fournisseur, et le passage de l'information par le réseau Ethernet qui permet d'éliminer quantité de câbles jusqu'alors nécessaires au transfert de l'information point à point. Afin de pouvoir déployer les technologies 61850 à l'ensemble du réseau électrique, il s'est révélé nécessaire d'apporter plus de fiabilité dans ce système d'automatisme basé sur les communications. C'est chose faite dans la seconde édition (2013) qui supporte l'Ethernet gigabit et qui est enrichie de méthodes de

redondance au niveau couche liaison : boucles PRP (Parallel Redundancy Protocol) et HSR (High-availability Seamless Redundancy). Ces mesures vont dans le sens de la fiabilité, de la disponibilité et plus globalement de la sûreté de fonctionnement. Ainsi, il est désormais concevable d'utiliser les technologies IEC 61850 dans des systèmes critiques tels qu'une centrale de production thermique ou nucléaire dans laquelle une perte ou une altération des communications impacterait la continuité de service mais pourrait aussi avoir des conséquences plus dramatiques.

Notre travail s'inscrit dans la continuité de ces mesures visant à assurer la sûreté de fonctionnement des systèmes IEC 61850 : il concerne l'étude d'une architecture résiliente à des attaques sur les communications en temps-réel dur, basée sur la détection d'intrusion. La détection d'intrusion est l'une des deux seules mesures de sécurité énoncées comme faisables pour le protocole GOOSE dans le standard IEC 62351 [IEC 2009], l'autre étant l'authentification par signature numérique, hors du scope de ce travail. Ce standard porte sur la sécurité des données et des communications dans les infrastructures électriques dont la sûreté de fonctionnement, la sécurité et la fiabilité reposent de plus en plus sur l'intégrité du système d'information et de communication.

## 3. Communication dans les postes : flux horizontaux et verticaux

La norme IEC 61850 spécifie plusieurs méthodes de communication selon les besoins temps-réel des différentes fonctions ainsi que les protocoles associés. Dans le contexte de cette étude deux méthodes seront utilisées : les communications locales, en temps-réel dur (4ms de délai de propagation application) et la communication avec le SCADA.

La figure 1 présente la disposition typique des flux de communication dans le SAS. La réalisation des fonctions nécessite une communication en temps-réel entre les IED. Cette communication est implémentée via des transmissions multicast de trames Ethernet (type 0x88b8 dites *GOOSE – Generic Object Oriented System Event*). La communication avec le SCADA est réalisée par TCP/IP avec un protocole application MMS (*Manufacturing Message Specification* - ISO 9506). Notons que, souvent, les flux "verticaux" et "horizontaux" circulent sur des réseaux physiquement différents (réseaux supervision et temps-réel séparés).

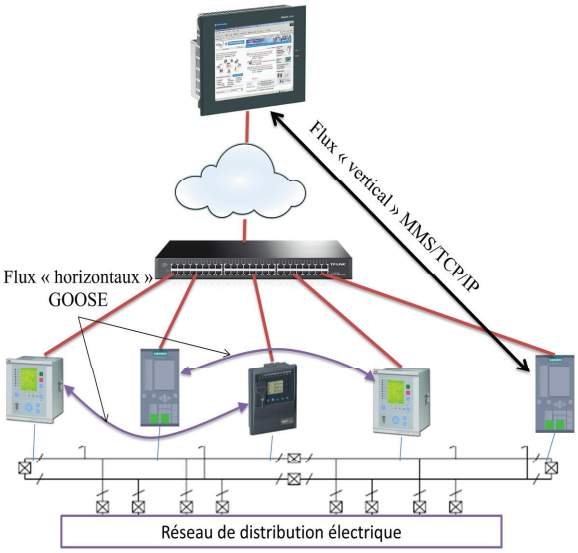



Figure 1. Flux de données 61850 : commuté "horizontal" (GOOSE) vs. "vertical" SCADA

## II. Fonctionnement du protocole GOOSE

### 1. Structure d'une trame GOOSE

Le protocole GOOSE est mappé sur la couche liaison Ethernet. Les messages sont envoyés en broadcast selon un mécanisme d'éditeur – abonné (*publisher – subscriber*) : les appareils branchés sur le réseau voient l'ensemble des trames GOOSE mais n'interceptent que celles qui les intéressent, les messages GOOSE auxquels ils sont abonnés.

La structure de la trame GOOSE est standardisée (ISO/IEC 8802-3), elle est rappelée dans le standard IEC 61850 et a fait l'objet de plusieurs publications dont [Kriger 2013]. Elle est détaillée dans la figure 2. L'APDU (*Application Protocol Data Unit*) correspond aux données transmises par l'application émettrice. Il est spécifié en ASN.1 comme une séquence de 12 éléments (à droite sur la figure 2).

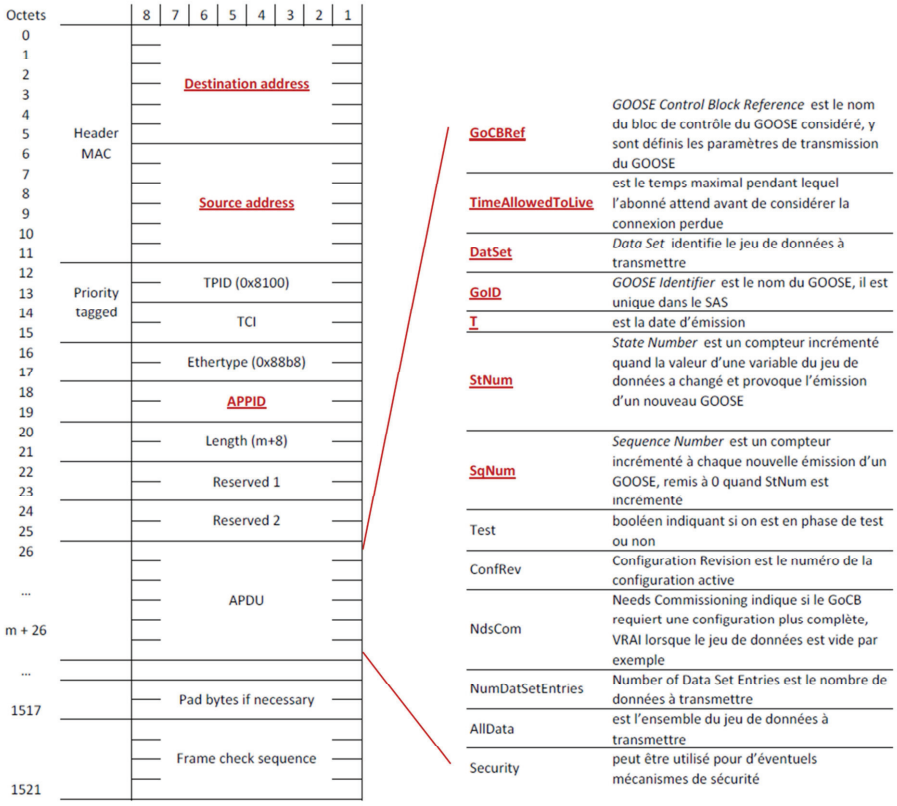



Figure 2. Structure d'une trame GOOSE (d'après IEC 61850-8.1)

Notons que le champ "Security" est prévu dans l'APDU de la trame GOOSE présenté dans la norme IEC 61850 mais son utilisation n'est pas explicitée. Les mécanismes de sécurité sont en fait recommandés dans le standard mais leur mise en œuvre est laissée à la discrétion des concepteurs d'IED.

Les champs dont le nom est souligné seront exploités dans ce travail pour vérifier la conformité du message avec la configuration du SAS.

### 2. Mécanisme de transmission du protocole GOOSE

Le mécanisme de transmission éditeur – abonné ne permet pas d'acquitter la réception du message. Pour assurer un certain niveau de fiabilité, le protocole GOOSE utilise un schéma particulier de retransmission comme illustré dans la figure 3. Lorsqu'un événement se produit, impliquant un changement de valeur de l'une ou plusieurs des données transmises par GOOSE, il déclenche l'envoi d'un message GOOSE en incrémentant StNum et en remettant à 0 SqNum. Cette trame GOOSE est retransmise à grande fréquence d'abord (T1) puis le rythme ralentit (T2, T3) jusqu'à atteindre la durée correspondant à des conditions stables. SqNum est incrémenté à chaque émission.

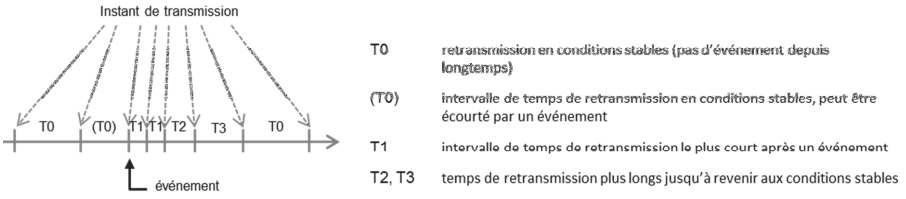



Figure 3. Schéma de transmission des messages GOOSE

## III. Détection d'intrusion sur le réseau GOOSE

### 1. Description des attaques

Nous considérons deux types d'attaques sur les trames Ethernet, attaques déjà répertoriées dans la littérature. Le premier type d'incident considéré est la tempête Ethernet. Ce type d'incident a été, par exemple, à l'origine d'un arrêt d'urgence d'une centrale nucléaire aux Etats-Unis [NRC 2007]. Dans le cas d'un poste IEC 61850, la conséquence d'une tempête Ethernet sur le réseau dédié aux communications GOOSE serait un incident de type DoS (*Denial of Service*). Les trames GOOSE ne pourraient alors pas être réceptionnées par leurs abonnés dans le temps imparti de 4ms mettant en péril le bon déroulement des mécanismes de protection. Malgré la procédure de retransmission détaillée ci-dessus, il est probable qu'un IED ne reçoive pas plusieurs occurrences d'un message GOOSE d'affilée. Or chez certains fabricants, l'hypothèse est faite qu'au maximum une occurrence d'un message GOOSE est perdue et les IED sont paramétrés pour ignorer les trames dont les numéros de séquence et d'état ne respecteraient pas cette hypothèse [Siemens 2014]. La connexion entre l'émetteur et l'abonné est alors rapidement considérée comme perdue.

Le second type d'attaque est l'injection sur le réseau de messages GOOSE usurpés, faussement interprétés par les appareils abonnés comme valides. Comme cela a été montré dans [Hoyos 2012], une telle attaque peut être implémentée dès qu'un accès au réseau temps-réel GOOSE est assuré. Il est alors possible de déclencher des actions non souhaitées telles que basculement d'un switch, ouverture d'un disjoncteur, etc… ou d'envoyer des messages frauduleux de coordination à d'autres sous-stations.

### 2. Etat de l'art

La sécurisation des systèmes embarqués est une préoccupation grandissante dans la mesure où elle est garante de la sécurité des personnes et des infrastructures. Divers domaines sont concernés tels que l'automobile [Koopman 2013], l'aéronautique [Dessiatnikoff 2012], les usines chimiques et autres infrastructures critiques, etc... Comme souligné dans [Koopman 2013], une attaque de type spoofing (usurpation d'adresse source) permet à un attaquant de compromettre la sécurité d'un système embarqué et de ses usagers de multiples manières, quasi illimitées. Il suffit d'un accès au réseau Ethernet pour injecter des messages de contrôle aux actionneurs qui tiennent un rôle critique du point de vue sûreté de fonctionnement.

La sécurisation des réseaux de systèmes de contrôle et en particulier des SCADA passe en général par la détection de communications suspectes :

- soit au périmètre du système afin d'empêcher qu'elles ne pénètrent la zone à sécuriser (firewall) [Fovino 2012]. Le défaut principal de cette approche est de ne pas apporter de protection pour les attaques lancées depuis l'intérieur du réseau, ce qui est le cas par exemple des deux types d'attaques considérées dans cette étude.
- soit au sein même du système afin d'avertir les opérateurs et de déclencher des mesures défensives (*IDS* et *IPS – Intrusion Detection / Protection System*) [Cheung 2007], [Premaratne 2010], [Hong 2014]. Ces deux derniers articles traitent spécifiquement de la détection d'intrusion réseau pour les protocoles multicast IEC 61850 tels que GOOSE.

Le travail proposé ici ne consiste pas uniquement à identifier des communications suspectes. Il s'agit d'apporter une réponse avec un mode de fonctionnement dégradé tolérant à une situation d'insécurité cyber. C'est l'idée sur laquelle s'appuient également les travaux de [Kirsch 2014]. L'architecture de supervision est constituée de plusieurs réplications du SCADA, implémentées différemment les unes des autres, qui doivent collaborer pour traiter les informations reçues du poste et générer une commande acceptée par toutes les entités (redondance hétérogène). Ainsi, l'application peut continuer à fonctionner de façon fiable dans la mesure où le nombre d'instances du SCADA corrompues ne dépasse pas un certain seuil. Cependant cette approche s'applique pour des communications verticales SCADA – sous-station quand nous nous intéressons aux échanges horizontaux temps-réel internes à la sous-station pour lesquels une telle stratégie n'est pas envisageable (puissance et temps de calcul).

Une piste proposée dans [Koopman 2013] pour sécuriser les communications multicast temps-réel est de développer des fonctions d'intégrité et d'authentification, afin d'empêcher des attaques de type spoofing qui seraient détectées directement par les appareils destinataires des messages. C'est une idée intéressante. Toutefois, nous avons fait le choix dans ce travail de proposer une solution extérieure aux IED et qui, de plus, va un pas plus loin que la détection avec un mode de fonctionnement alternatif.

## IV. Notre approche

Notre proposition d'architecture résiliente repose sur les principes suivants :

- des flux "verticaux" (échanges de données avec le SCADA) sûrs et sur un réseau distinct du réseau temps-réel [Kirsch 2014],
- l'existence des sondes capables de détecter les tempêtes Ethernet ainsi que les GOOSE usurpés ("IDS niveau Ethernet"),
- la remontée des alertes des IDS Ethernet vers le SCADA,
- la réécriture des programmes de commande des IED en prenant en compte les alertes des IDS envoyés via le SCADA.

La suite de l'article décrit notre solution aux trois derniers points ci-dessus.

### 1. Les "IDS Ethernet"

Nous avons modifié deux logiciels de monitoring du réseau afin d'obtenir les détecteurs des tempêtes Ethernet et des messages GOOSE usurpés.

A partir d'un moniteur de bande passante disponible en Linux (nous avons choisi ifstat, mais bmon, bmonNG, slurm, ntop on vnstat peuvent également être utilisés), nous avons créé une sonde de mesure de l'utilisation de bande passante instantanée et moyenne sur des durées configurables par l'utilisateur. Les données sont remontées au SCADA via un serveur de données utilisant un protocole industriel (dans la première version nous avons utilisé un serveur Modbus/TCP de par sa simplicité).

La détection des messages GOOSE usurpés est bien plus délicate. Tel qu'il a été présenté dans [Hoyos 2012] une attaque GOOSE envoie une séquence rapide de trames GOOSE (un faux changement d'état au niveau de l'émetteur) qui, dans le cas d'une attaque "parfaite", respecte les numéros de séquence et les horloges des trames légales. La figure 4 présente le chronogramme d'une telle attaque.

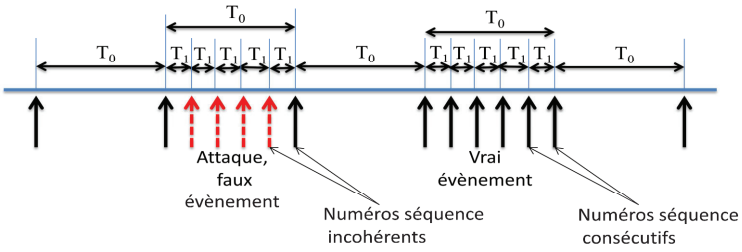



Figure 4. Vrais (traits pleins) et faux (pointillés) évènements dans les séquences de GOOSE

Une attaque GOOSE "imparfaite" peut être détectée en comparant les numéros des trames et horodatages de deux GOOSE successifs. Mais même dans l'hypothèse d'une attaque GOOSE parfaite, celle-ci sera détectée au plus tard après un temps $T_0$, le premier message GOOSE légal après ou pendant l'attaque mettant en évidence l'incohérence des numéros de séquence.

Notre détecteur de trames GOOSE usurpées est créé à partir de l'analyseur de trafic tcpdump. Nous vérifions pour chaque trame GOOSE :

- que le message GOOSE reçu correspond à une connexion GOOSE définie dans le système en contrôlant les champs suivant : Source address, APPID, GoCBRef et GoID,
- la cohérence des compteurs StNum et SqNum par rapport au message GOOSE précédent ayant le même GoID (GOOSE Identifier),
- la cohérence de la date d'émission avec le message GOOSE précédent ayant le même GoID ainsi qu'avec les compteurs StNum et SqNum.

Afin de réduire le délai de propagation des alarmes vers le SCADA en cas de détection de faux messages GOOSE (messages fabriqués par un intrus), nous avons couplé notre sonde à un client Modbus/TCP, le superviseur étant configuré en serveur.

## 2. Intégration au SCADA

Les données des sondes remontées soit périodiquement (scrutation du serveur Modbus de la sonde mesurant la bande passante), soit spontanément (messages événementiels envoyés par le détecteur des messages GOOSE usurpés) sont exploitées au niveau du superviseur SCADA pour la prise de décision du basculement des équipements du système (IED) en mode dégradé/sécurité. L'architecture générale est présentée en figure 5.

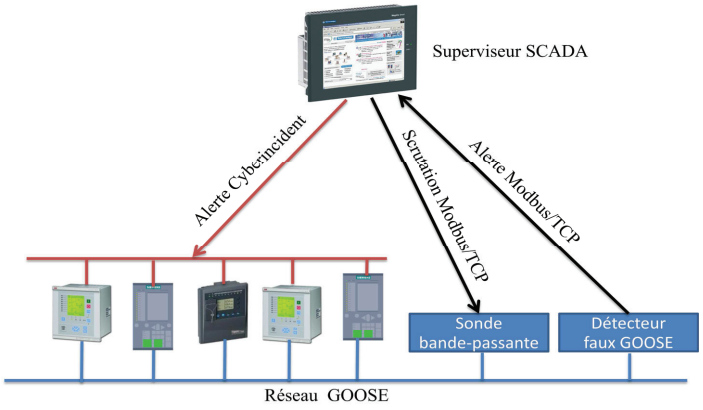



Figure 5. L'architecture de communication proposée

## 3. Programmation résiliente des IED

La dernière étape dans la réalisation de notre système résilient aux attaques est la prise en compte de l'alerte cyber-incident au niveau du programme de l'IED. Basculer d'un mode de fonctionnement à l'autre est simple au niveau de l'écriture des diagrammes de commande (voir figure 6). La difficulté réside dans la synthèse de la fonction "sûre".

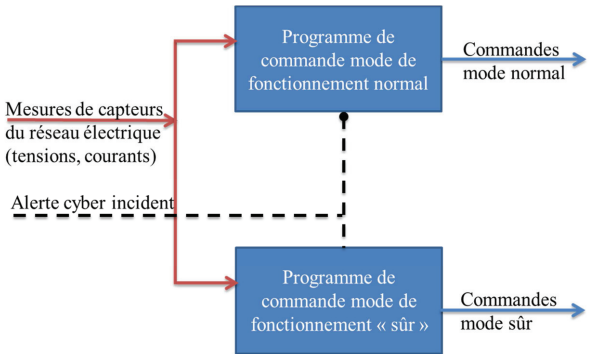



Figure 6. Basculement du mode de fonctionnement sûr / cyber-incident

### 3.1 Synthèse du mode de fonctionnement sûr : cas général

Nous proposons une méthode dérivée de l'AMDE (Analyse des Modes de Défaillance et leurs Effets), inspirée de la synthèse des modes de fonctionnement dégradés en cas de panne. Nous considérons la perte de la fonction "communication" comme une défaillance du système pour analyser les conséquences et chercher les comportements sûrs basés sur l'expertise du domaine. Evidemment, cette approche hérite des inconvénients de l'AMDE : longue, lourde, complexe et nécessitant une connaissance approfondie du système analysé.

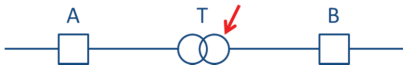



Figure 7. Portion de ligne électrique, exemple

Prenons un exemple fictif pour illustrer la démarche. Sur la portion de ligne de la figure 7, dans le cas d'un défaut de surintensité au niveau du transformateur T, supposons que le scenario de protection prévoit que le disjoncteur A s'ouvre puis le disjoncteur B. Un faux message GOOSE indiquant à B que A est bien ouvert alors que ce

n'est pas le cas correspond à une défaillance dont l'effet serait l'ouverture de B et l'apparition d'un arc électrique, ce qui doit être éviter. L'AMDE doit nous permettre d'identifier l'ensemble des défaillances aux conséquences indésirables pour implémenter les programmes alternatifs "mode de fonctionnement « sûr »".

Pour le cas particulier des SAS, nous nous inspirons d'une technique de temporisation dérivée des schémas de protection classiques.

### 3.2. La sélectivité électrique

Dans le domaine de la protection électrique, la sélectivité consiste à localiser et déconnecter la partie du réseau en défaut, et seulement celle-ci, en maintenant sous tension la plus grande partie possible de l'installation [Nereau 2001]. Il existe différentes méthodes pour assurer la sélectivité : la sélectivité ampèremétrique par les courants, chronométrique par le temps, logique par échanges d'information, par protection directionnelle, par protection différentielle [MG 2003]. Il est possible de combiner deux types de sélectivité pour exploiter leurs avantages de façon complémentaire, apportant par exemple redondance ou secours. Par exemple, il est possible d'associer une sélectivité chronométrique à une sélectivité logique pour palier un défaut d'attente logique.

Dans le cas d'une sélectivité chronométrique, le défaut représenté sur la figure 8 est vu par les protections A, B, C et D. C'est le disjoncteur en D qui déclenche le premier (0,2s après la détection du courant de défaut) puis si celui-ci est défaillant et ne s'est pas ouvert, la protection en C se déclenche après un temps plus long, etc… Les relais en amont ne déclenchent que si les précédents ont échoué à éliminer le courant de défaut. La sélectivité logique permet de réduire considérablement le temps d'élimination du défaut car il n'est plus nécessaire d'avoir une temporisation croissante quand on se rapproche de la source : c'est le relais le plus proche du défaut qui envoie un ordre d'attente empêchant les relais amont de déclencher. Un secours est en général ajouté à la sélectivité logique avec une sélectivité chronométrique en cas de défaut d'attente logique tel qu'un ordre d'attente permanent.

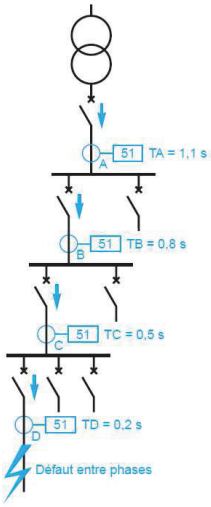



Figure 8. Principe de la sélectivité chronométrique [MG 2003]

Nous nous sommes inspirés de cette technique de protection électrique par sélectivité logique doublée d'un secours chronométrique pour proposer un mode de fonctionnement du SAS résilient aux "défauts" des communications GOOSE. Dans le cas normal (niveau de confiance des messages GOOSE élevé), un IED reçoit un message GOOSE et attend une éventuelle alerte de

cyber-incident du SCADA. Si après un temps t configuré, il n'y a pas eu d'alerte alors l'IED exécute le programme déclenché par le contenu du message. Par contre si l'IED reçoit une alerte, il va lancer un programme alternatif indépendant de la communication GOOSE et reste dans ce mode tant que l'alerte n'est pas levée. L'exemple suivant détaille cette technique qui permet d'assurer le temps de réaction de la sonde de détection des faux messages GOOSE.

## 4. Architecture de test

Le système électrique sur lequel nous menons notre étude est un couplage de deux jeux de barres. De part et d'autre du couplage, chacune des deux sections alimente plusieurs lignes de transformation par son propre générateur. En fonctionnement normal le couplage est ouvert. Dans le cas d'un défaut au niveau du générateur de la première section, celle-ci n'est plus alimentée. Il s'agit alors de faire automatiquement la commutation permettant au générateur de la deuxième section d'alimenter la section 1, en ouvrant le disjoncteur du générateur 1 afin d'isoler le défaut puis en fermant le couplage. Chacune des alimentations ainsi que le couplage sont gérés par leurs propres IED. L'architecture électrique représentée sur la figure 6 est simplifiée : les lignes de transformation n'apparaissent pas.

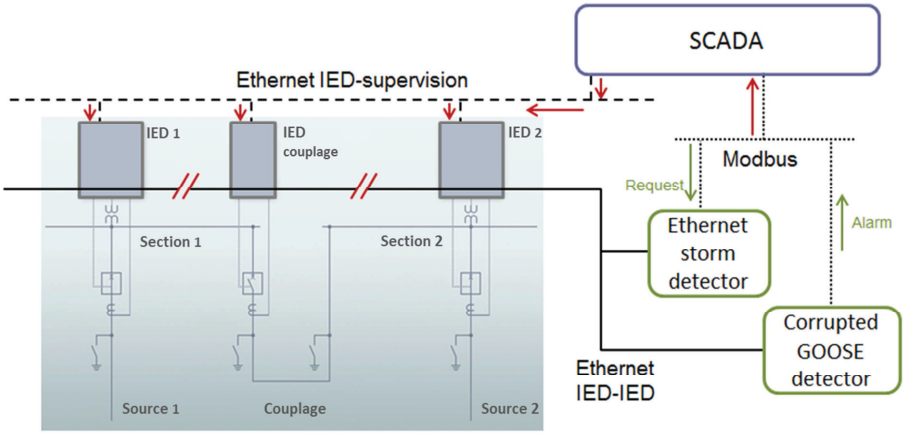



Figure 6. Architecture électrique et de communication

En fonctionnement normal, une temporisation force les IED à attendre jusqu'à l'expiration du temps maximal de détection de fausses trames GOOSE. Cela est envisageable car les grandeurs temporelles des fonctions de protection sont de l'ordre de 100ms voire 1s [MG 2003] alors que le temps de transfert total d'un message GOOSE doit être au maximum de 4ms.

Le mode sûr (obtenu par AMDE) est particulièrement simple dans ce cas. Il suffit de positionner le couplage en position ouverte et de ne pas générer le signal de fermeture des disjoncteurs. Le système revient en mode normal dès que le superviseur désactive l'alarme.

Les tests sur les performances du système en termes de temps de réaction et dégradation de la performance du réseau de distribution sont en cours.

## V. Conclusions

Nous avons proposé une architecture intégrant les capteurs de cyber-incident dans le superviseur SCADA afin de permettre la détection des attaques sur les communications temps-réel dans les SAS 61850 et le fonctionnement du système dans un mode dégradé malgré les attaques au niveau temps-réel. Nous avons réalisé deux sondes simples dont une qui permet de répondre à la clause 6.10.2 (détection des faux messages GOOSE) du standard IEC 62351-1. Après l'évaluation des performances des sondes sur notre maquette de test (temps de détection des tempêtes et faux

messages GOOSE) ainsi que du temps de réaction du système d'alerte (passage en mode dégradé) nous envisageons d'améliorer le fonctionnement en mode dégradé en introduisant, par exemple, une propagation des informations d'état par le réseau de supervision en cas de cyber-incident, autrement dit d'utiliser la communication "verticale" pour assurer le transfert de l'information normalement réalisé par la communication "horizontale".

## Références


[Cheung 2007] Cheung S., Dutertre B., Fong M., Lindqvist U., Skinner K., Valdes A. (2007), Using model-based intrusion detection for SCADA networks. SCADA Security, Proc. intern. scientific symp., Miami Beach, Florida, USA, 24-25 January 2007

[Dessiatnikoff 2012] Dessiatnikoff A., Deswarte Y., Alata E., Nicomette V. (2012),Potential Attacks on Onboard Aerospace Systems, Security & Privacy, IEEE 10(4): 71-74

[Fovino 2012] Nai Fovino I., Coletta A., Carcano A., Masera M. (2012), Critical state-based filtering system for securing SCADA network protocols, IEEE Transactions on Industrial Electronics 59(10): 3943-3950

[Hong 2014] Hong J., Liu C.-C., Govindarasu M. (2014). Integrated Anomaly Detection for Cyber Security of the Substations, IEEE Transactions on Smart Grid 5(4): 1643-1653

[Hoyos 2012] Hoyos J., Dehus M., Brown T. (2012), Exploiting the GOOSE protocol: A practical attack on cyber-infrastructure, IEEE Globecom Workshops

[IEC 2009] IEC 62351 Parts 1-8 - Power systems management and associated information exchange – Data and communications security. 2009

[IEC 2012] IEC 61850 Communication Networks and Systems for Power Utility Automation, 2012

[Kirsch 2014] Kirsch J., Goose S., Amir Y., Wei D., Skare P. (2014), Survivable SCADA via Intrusion-Tolerant Replication, IEEE Transactions on Smart Grid 5(1): 60-70

[Koopman 2013] Koopman P., Szilagyi C. (2013), Integrity in Embedded Control Networks, Security & Privacy, IEEE 11(3): 61-63

[Kriger 2013] Kriger C., Behardien S., Retonda-Modiya J. (2013), A Detailed Analysis of the GOOSE Message Structure in an IEC 61850 Standard-Based Substation Automation System, INT J COMPUT COMMUN 8(5):708-721

[MG 2003] Merlin Gerin, Protection des réseaux électriques – Guide de la protection, 2003.

[Nereau 2001] Nereau J.-P., Schneider Electric, Cahier technique n°201, Sélectivité avec les disjoncteurs de puissance basse tension, mars 2001

[NRC 2007] NRC, Effects of Ethernet-based, Non-safety Related Controls on the Safe and Continued Operation of Nuclear Power Stations, NRC Information notice 2007-15

[Premaratne 2010] Premaratne U., Samarabandu J., Sidhu T., Beresh R., Tan J.-C. (2010), An intrusion detection system for IEC 61850 automated substations, IEEE Transactions on Power Delivery 25: 2376–2383.

[Siemens 2014] SIPROTEC 5 PIXIT, PICS, TICS IEC 61850 Manual v06.00, C53000-G5040-C013-4.00, 2014

# Etude comparative des formats d'alertes

Guillaume Hiet[1], Hervé Debar[2], Selim Menouar[3], and Vérène Houdebine[3]

[1] CentraleSupélec, Cesson-Sévigné, France,
guillaume.hiet@centralesupelec.fr
[2] Télécom SudParis, Evry, France,
herve.debar@telecom-sudparis.eu
[3] CS, Le Plessis Robinson, France,
prenom.nom@c-s.fr

**Résumé** Une des approches contribuant à la résilience des systèmes informatiques consiste à surveiller en continu leur fonctionnement afin de détecter les comportements indésirables (attaques, intrusions). L'objectif final est de ramener le système dans un état sain, éventuellement dans un mode dégradé, en exécutant des contre-mesures adéquates. Il est pour cela nécessaire de déployer différents composants qui doivent échanger de l'information sous forme d'alertes : sondes de détection, manager, base de données, interface de visualisation, etc. Dans ce contexte, la définition d'un format standard et ouvert pour les échanges d'alertes apparaît crucial. Ce format doit offrir une structuration et une richesse sémantique qui facilitent le classement et la contextualisation des alertes. Ces aspects sont essentiels pour optimiser le traitement automatiques des alertes (par exemple, la corrélation). L'utilisation d'un format standard facilite non seulement l'inter-opérabilité mais il permet également aux exploitants de capitaliser leurs efforts. Nous présentons dans cet article les résultats d'une étude comparative des formats d'alerte existants et nous proposons des pistes d'amélioration issues de ce retour d'expérience. Cette étude a été réalisée dans le cadre du projet RAPID SECEF qui s'intéresse notamment à proposer des améliorations au format IDMEF [1] pour l'adapter au contexte actuel et faciliter son adoption.

**Keywords:** supervision de sécurité, format d'alertes, IDMEF

## 1 Introduction

La supervision de sécurité est une approche de cyber-sécurité réactive qui participe à la Lutte Informatique Défensive. Elle consiste en premier lieu à détecter les attaques ou intrusions et à émettre le cas échéant des alertes (ou événements de sécurité). Une deuxième étape consiste à gérer automatiquement ces différentes alertes (processus de collecte, de stockage et de corrélation). Ces informations sont ensuite présentées aux personnes en charge de la sécurité du SI sous différentes formes (évolutions au fil de l'eau, rapports d'incident, tableaux de bord, résultats de recherches, etc.). L'objectif final est d'apporter une information la plus pertinente possible pour que les opérateurs de sécurité puissent mettre en œuvre des contre-mesures. Cette approche contribue à la résilience des

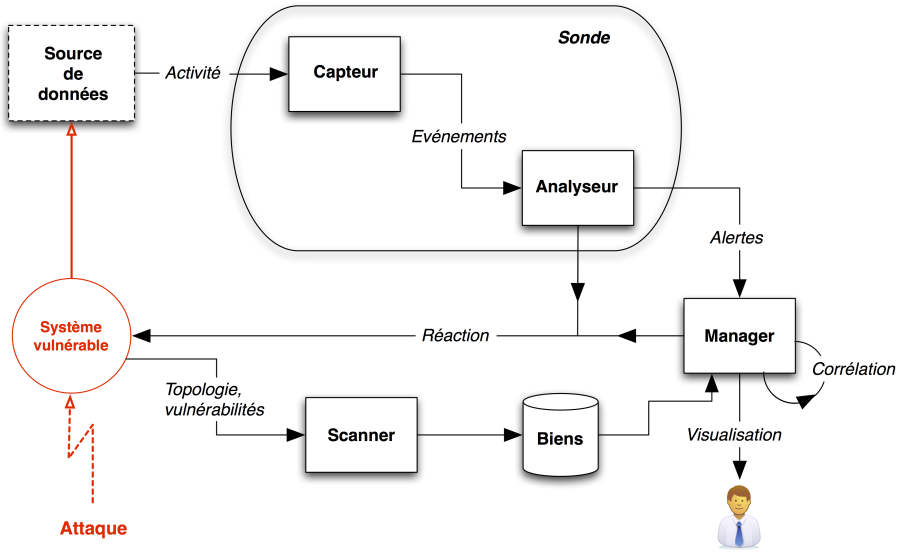



**Figure 1.** Architecture de supervision de sécurité

systèmes informatiques, comme le souligne la grille d'analyse de la résilience [2] proposée par le Professeur Erik Hollnagel. Celle-ci s'appuie sur quatre caractéristiques d'un système pour évaluer sa capacité de résilience. L'une d'entre-elles réside justement dans la capacité du système à surveiller les évolutions de son comportement et de son environnement.

Comme l'illustre la figure 1, la mise en œuvre d'une architecture de supervision de sécurité nécessite de déployer différents composants hétérogènes qui doivent communiquer entre eux au sein d'un *Security Operationnal Center*. Typiquement, les SOC sont constitués de différentes sondes qui doivent remonter l'information concernant les événements qu'elles ont détectés à des *managers* (*Security Information and Event Manager*). En pratique, les composants du SOC proviennent de différents éditeurs.

Dans ce contexte, la définition d'un format standard d'échange des alertes apparaît crucial. Plusieurs formats, plus ou moins spécifiques au domaine, ont été proposés par le passé mais force est de constater qu'aujourd'hui, aucun d'eux n'a été adopté massivement par les différents éditeurs. Généralement, chaque éditeur de sonde utilise un format propriétaire pour décrire les alertes émises par ses sondes. Il est donc nécessaire que le manager connaisse ces différents formats qu'il doit analyser et traduire dans un format interne, ce dernier étant lui aussi généralement propriétaire. Cette étape de normalisation est primordiale. En effet, il est nécessaire que les alertes soient exprimées dans un format commun afin de pouvoir leur appliquer facilement par la suite des traitements automatisés (corrélation, requête, etc.).

Cette situation (normalisation par le manager), n'est pas optimale. En effet, l'intégration repose alors en grande partie sur les capacités de l'éditeur du manager à normaliser de manière pertinente un grand nombre de formats d'alertes [4]. Pour les formats non reconnus nativement, l'utilisateur doit, si le produit retenu le lui permet, développer lui-même le traducteur, ce qui peut s'avérer complexe et difficile à maintenir dans le temps. Les éditeurs de managers et les utilisateurs finaux sont alors fortement dépendants des éditeurs de sondes, notamment en cas de modification du format utilisé par ces derniers.

A l'inverse, l'utilisation d'un format standard commun aux différents acteurs du domaine permettrait de s'affranchir de cette dépendance et faciliterait l'inter-opérabilité. En outre, l'utilisation par les managers d'un format standard, au lieu des multiples formats propriétaires, permettrait aux utilisateurs finaux (opérateur de sécurité, administrateur, etc.) de capitaliser leurs efforts lors du développement des traitements automatisés, tels que la corrélation. Actuellement, le développement de ces traitements nécessite d'acquérir une compétence spécifique à un produit, notamment en termes de format. Enfin, l'utilisation d'un format standard permet de développer des traitements automatiques qui soient génériques et ne dépendent pas du format propre à chaque sonde.

L'objectif du projet SECEF [5] est de promouvoir IDMEF, l'un des rares formats standards dédiés au domaine. Pour cela, le projet s'attelle à différentes tâches dont :

- l'étude comparative des différents formats plus ou moins spécifiques au domaine qui ont été proposés jusqu'à maintenant afin notamment de dégager les bonnes pratiques ;
- la proposition d'évolutions du format IDMEF, prenant notamment en compte les résultats issus de l'étape précédente, afin d'adapter le standard existant au contexte actuel ;
- la rédaction de documentation et de tutoriaux permettant de faciliter le travail des développeurs désireux d'adopter IDMEF ;
- le développement et la mise à disposition de bibliothèques permettant d'échanger et de manipuler des alertes au format IDMEF.

Nous proposons dans cet article de présenter les travaux relatifs aux deux premières tâches de ce projet : l'étude comparative des formats d'alertes et les pistes d'amélioration envisagées.

Dans un premier temps, nous rappelons en section 2 le périmètre de l'étude et la démarche retenue. Puis nous présentons la comparaison des formats en section 3 ainsi que les pistes d'amélioration en section 4. La section 5 conclut cet article.

---

4. Il s'agit d'ailleurs d'un argument commercial avancé par bon nombre d'éditeurs
5. SECEF/COSCOM est un projet financé par la DGA via le dispositif RAPID : `http://www.secef.net`

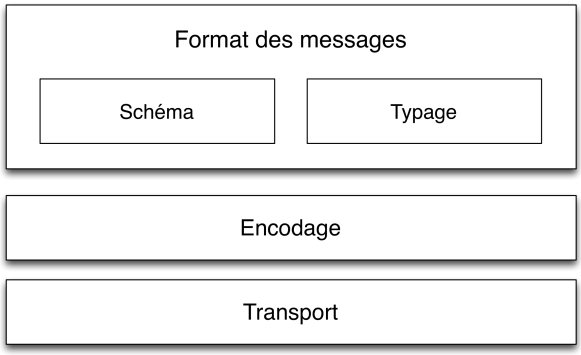



**Figure 2.** Format de messages

## 2 Périmètre de l'étude et démarche

Nous présentons par la suite les concepts relatifs à la définition d'un format de message. Nous présentons également la démarche que nous avons suivie.

### 2.1 Format d'alerte

Les informations remontées par les sondes vers les managers le sont sous forme de messages. Il convient donc de définir le format de ces messages. L'étude des différents formats existants fait apparaître qu'il existe en fait différents niveaux de définition pour un format donné, ce qui est illustré par la figure 2.

Classiquement, les message sont décrits par un ensemble d'attributs (ou de champs) auxquels sont associées des valeurs. Le format des messages à proprement parler comprend :

— le **schéma**, c'est-à-dire la structure des messages et la définition des différents attributs standards, ainsi que la sémantique de ces attributs ;
— le **typage**, c'est-à-dire le format des valeurs que peuvent prendre ces différents attributs.

Selon les formats, le typage peut-être plus ou moins précis (par exemple « chaîne de caractères » vs. « date au format ISO 8601 »). Le typage décrit l'ensemble des valeurs possibles pour un attribut donné. Pour certains attributs, ce typage peut prendre la forme d'une énumération ou d'un dictionnaire.

L'usage de dictionnaire apparaît comme une nécessité afin de pouvoir comparer et traiter automatiquement (notamment durant la phase de corrélation) les valeurs de certains champs issus d'alertes produites par des sondes développées par différents éditeurs. Toutefois, la définition de ces dictionnaires, c'est-à-dire d'une énumération de valeurs non ambiguës qui couvre les besoins de chacun tout en permettant de distinguer les différents cas, s'avère en pratique souvent

difficile. Il s'agit donc d'un point important dans la définition et la comparaison des formats.

Le format des messages est une spécification abstraite. L'encodage (ou format de sérialisation) détermine la manière dont les messages, c'est-à-dire les champs et les valeurs associées, vont être codés. Il est possible d'utiliser un format d'encodage ad-hoc mais il paraît préférable d'utiliser des formats génériques pré-existants (par exemple JSON ou XML). Cela permet notamment, en termes d'implémentation, de faciliter les développements (ré-utilisation de bibliothèques existantes) ainsi que de favoriser la robustesse (ré-utilisation de parseurs robustes) et l'interopérabilité.

Classiquement, on peut distinguer les encodages textuels reposant sur ASCII ou UTF (par exemple, JSON et XML) des encodages binaires (par exemple BER, CER, BSON, binary XML, etc.). Les premiers ont l'avantage d'être directement interprétables par un humain (ce qui peut être utile notamment lorsque les messages sont stockés directement dans des fichiers). Ils sont dans notre cas particulièrement adaptés car l'information à transporter est principalement de nature textuelle. Le transport d'information binaire (par exemple, capture réseau au format PCAP,) nécessite un encodage particulier, par exemple Base64 ou Uuencoding. Les encodages binaires permettent un encodage plus compact et donc d'optimiser les performances.

Le protocole de transport permet d'échanger les messages en utilisant une pile protocolaire standard (TCP/IP, étant donné le cas d'usage). Comme pour l'encodage, l'intérêt est d'utiliser des protocoles génériques existants (HTTP, SYSLOG, AMQP, etc.). Le protocole de transport et l'encodage sont en général plus ou moins couplés. Par exemple, l'utilisation de HTTP comme protocole de transport tend à favoriser l'utilisation d'encodage textuel (JSON, XML). Certains protocoles, par exemple AMQP, imposent un encodage particulier. A l'inverse, il est parfois envisageable d'utiliser différents encodages pour un même protocole (JSON ou XML sur HTTP) ou un même encodage sur différents protocoles (par exemple JSON sur HTTP ou SYSLOG).

Classiquement, le transport et l'encodage assurent un certain nombre de fonctionnalités essentielles pour le bon acheminement des messages. Ils doivent notamment fournir des fonctions de sécurité (authentification des émetteurs et des récepteurs, signature et chiffrement des messages), de haute-disponibilité (ré-émission des messages, mécanismes de redondance) et d'optimisation de la bande-passante (compression, gestion de la congestion, etc.). Ces fonctionnalités sont particulièrement importantes dans le contexte de l'échange d'alertes de sécurité. Toutefois, elles sont a priori indépendantes du format des messages à proprement parler car elles sont liées au type d'encodage et de transport utilisés.

La présente étude se focalise essentiellement sur le format de messages (c'est-à-dire le schéma et le typage). En effet, les besoins finaux (standardisation en vue de faciliter le traitement automatique des données) imposent prioritairement que le schéma et le typage soient standards. Ces derniers devraient être en grande partie agnostiques de l'encodage et du protocole de transport utilisés, le choix de ceux-ci relevant de l'implémentation.

### 2.2 Démarche

L'analyse de l'existant a consisté d'une part à étudier et analyser les formats que l'on souhaite promouvoir et d'autre part à les comparer à des formats similaires. L'identification de ces formats similaires a été réalisée en s'appuyant sur des études existantes, notamment celle réalisée par l'ENISA [3], l'expertise préalable des membres du consortium ainsi que les études de marché.

Nous avons comparé le format IDMEF avec 5 formats « concurrents ». Parmi ces formats, deux sont des formats propriétaires proposés par des éditeurs de SIEM :

— LEEF (IBM QRadar)
— CEF (HP ArcSight)

Les trois derniers formats sont des formats ouverts proposés par des organismes de standardisation :

— CEE (MITRE)
— XDAS/CADF (The Open Group, DMTF)
— CIM (DMTF)

Dans un premier temps, nous avons effectué une comparaison synthétique des différents formats en s'appuyant sur des critères présentés par la suite. Dans un deuxième temps, nous avons réalisé une comparaison détaillée de ces formats en établissant une table de correspondance entre les champs des différents formats. Cette deuxième étape nous a permis d'évaluer précisément la richesse sémantique de chaque format. Elle fournit également une référence pour le développement d'outils de normalisation ou de passerelles de traduction vers/depuis les formats étudiés. Enfin, elle permet de mettre en évidence les champs spécifiques à certains formats qui nous paraissent pertinents et qui constituent des pistes d'amélioration ou, à l'inverse, des champs dont la sémantique est ambiguë et/ou l'intérêt limité. Le tableau de correspondance obtenu à l'issue de cette étude est disponible sur le site du projet SECEF.

Pour l'analyse et la description synthétique des différents formats, nous avons retenus les critères suivants :

— le critère **références** permet de lister les documents de références (spécification) et donne un aperçu sur la nature et la qualité de ces documents (précision, complétude, historique, etc.) ;
— le critère **transport et encodage** permet d'indiquer quels sont les encodages et protocoles de transport utilisés (imposés par le format ou utilisés en pratique par les implémentations) ;
— le critère **pouvoir d'expressivité** permet d'identifier la nature et le type d'information que le format permet d'exprimer (sur les systèmes ciblés, la source de l'attaque, la sonde, etc.) ;
— le critère **structuration** permet de décrire brièvement la structure des messages et l'imbrication des différents attributs (format « à plat » vs. orienté objet, utilisation de l'héritage, de classes agrégées, etc.) ;
— le critère **extensibilité** permet de préciser les moyens prévus dans les standards pour étendre le format (champ additionnels, héritage, etc.).

## 3 Comparaison des différents formats

Dans un premier temps, nous décrivons les format étudiés en nous appuyant sur les critères énoncés ci-dessus. Dans un deuxième temps, nous présentons la synthèse des résultats de cette étude comparative.

### 3.1 Présentation des différents formats étudiés

**CEF** CEF est le format propriétaire du SIEM HP ArcSight. Il s'agit clairement d'un format orienté « événements de sécurité » en générale, c'est-à-dire non spécifique à la détection d'intrusion (il permet facilement d'exprimer l'information remontée par différents équipements de sécurité).

*Références* HP a publié une documentation complète mais succincte (30 pages environ) du format CEF [6]. Globalement, tous les champs sont documentés de manière claire et la documentation permet de générer une alerte CEF de manière non ambiguë. Toutefois, la sémantique exacte de certains champs est décrite de manière trop superficielle (par exemple « Application Protocol », « cat » ou « reason »).

*Transport et encodage* Les produits ArcSight utilisent le format Syslog comme transport/encodage. Plus précisément, un message CEF est encodé dans un champ Syslog sous la forme d'une liste d'entrées décrites par un ensemble de couples clé/valeur. Toutefois, l'utilisation d'autres encodages et transports ne semble pas présenter de difficulté technique (par exemple JSON/HTML).

*Pouvoir d'expressivité* Le format est assez expressif. Il permet d'exprimer les différents types d'information prévus par IDMEF (source, cible, sonde, etc.). Pour chaque type d'information, on retrouve les attributs qui paraissent essentiels. IDMEF est plus complet même si les champs non présents dans CEF semblent d'une importance plus limitée. CEF propose en outre une vingtaine d'attributs qui, à première vue, sont difficilement traduisibles en IDMEF. Certains de ces champs (par exemple, la gestion des adresses avec le NAT) constituent des pistes intéressantes pour étendre IDMEF.

*Structuration* Le schéma est peu structuré. Il s'agit d'une organisation « à plat » comportant 117 attributs qui n'ont pas de relations les uns par rapport aux autres. Cette absence de structure se traduit parfois par la présence de champs distincts contenant des informations similaires mais de différents types. Par exemple, les adresses IPV4, IPV6, après le NAT, etc. sont contenues dans des attributs distincts (IDMEF permet à l'inverse de spécifier un nombre illimité d'adresses et utilise un champ catégorie pour préciser le type d'adresse). Le format n'utilise pas d'attribut avec une multiplicité supérieure à un (il n'est pas possible de spécifier plusieurs sources ou plusieurs cibles dans une même alerte).

6. `https://protect724.hp.com/docs/DOC-1072`

Le format n'utilise pas de dictionnaire. Peu de champs le nécessitent mais cela serait nécessaire pour certains d'entre-eux (typiquement « outcome », « reason » ou « cat »).

*Extensibilité* Le format propose nativement un mécanisme d'extension assez limité : quelques champs additionnels typés mais en nombre restreint (entier, chaîne de caractère). Il est également possible d'étendre facilement le format de manière non standard en ajoutant des attributs.

**LEEF** LEEF est le format propriétaire du SIEM IBM QRadar. Ce « format amélioré pour les événements de journaux » (Log Event Enhanced Format) est clairement un format orienté « événements de sécurité », en particulier ceux en lien avec des problématiques de sécurité « réseau ». Le format est assez similaire à CEF dans sa structuration et dans le choix du transport et de l'encodage. En revanche, il existe des différences notables concernant le nombre et la sémantique des attributs.

*Références* IBM a publié une documentation complète mais très succincte (23 pages environ) du format LEEF [7]. Globalement tous les champs sont documentés mais souvent de manière trop superficielle. La sémantique de certains champs n'est pas claire, en particulier la série des champs « identXXX » (par exemple « identSrc » ou « identSecondIp »).

*Transport et encodage* Les produits QRadar utilisent le format Syslog comme transport/encodage de manière similaire à CEF. Toutefois, l'utilisation d'autres encodages et transports ne semble pas présenter de difficulté technique (par exemple JSON/HTML).

*Pouvoir d'expressivité* L'expressivité du format est limitée. La sonde est uniquement décrite en termes de noms de produits et de vendeurs (pas d'information sur les adresses, par exemple). Le format ne permet pas d'exprimer des informations sur les processus ou les fichiers (il est donc plutôt restreint aux événements « réseau »). Il ne permet donc pas d'exprimer tous les types d'informations que permet d'exprimer IDMEF. Pour les types d'informations communs (par exemple la source ou la cible de l'attaque), LEEF propose un nombre limité d'attributs permettant essentiellement de renseigner les informations relatives au réseau (par exemple, l'adresse IP).

*Structuration* La structuration des messages LEEF est similaire à celle de CEF. Il s'agit d'un format « à plat » comportant 50 attributs qui n'ont pas de relation les uns par rapport aux autres. Tout comme CEF, la multiplicité des attributs

7. `https://www.ibm.com/developerworks/community/wikis/form/anonymous/api/wiki/9989d3d7-02c1-444e-92be-576b33d2f2be/page/3dc63f46-4a33-4e0b-98bf-4e55b74e556b/attachment/a19b9122-5940-4c89-ba3e-4b4fc25e2328/media/QRadar_LEEF_Format_Guide.pdf`

est au plus de un et le format utilise des champs distincts pour des informations similaires de différents types (adresse IP avant/après NAT). Le format ne propose pas de dictionnaire mais peu d'attributs le nécessitent (« ressource » ou « cat »).

*Extensibilité* La documentation ne mentionne aucun mécanisme natif permettant d'étendre le format. Il est cependant possible d'étendre facilement le format de manière non standard en ajoutant des attributs.

**CEE** Common Event Expression (CEE) est un format initié par MITRE, un organisme de recherche états-unien à but non lucratif, financé principalement par des fonds publics. L'objectif (ambitieux) est de fournir un format standard pour les journaux générés par les systèmes informatiques en général. Il s'agit donc d'un domaine d'application plus large que la gestion des alertes ou événements de sécurité.

Les travaux ont été réalisés par un groupe de travail comportant des représentants de différents éditeurs et d'organismes gouvernementaux états-uniens (NIST et DoD). MITRE a joué le rôle de modérateur. Le gouvernement états-unien a depuis décidé de stopper le financement de ce projet, ce qui a conduit à l'arrêt (visiblement définitif) des travaux, d'après le site internet du projet [8].

Le groupe de travail a pris le parti de faire, dès le départ, une séparation claire entre le format des messages (CEE Event Model ou CEE Profil), l'encodage (CEE Log Syntax) et le transport (CEE Log Transport). Un profil CEE est lui-même composé d'un schéma, qui définit la structure des messages, de la spécification des différents attributs (field dictionary) et d'un ensemble de dictionnaires (event taxonomy).

*Références* A ce jour, les travaux réalisés n'ont conduit qu'à une version très préliminaire des spécifications. Celles-ci sont disponibles sur le site du projet CEE [9] à des fin d'archivage. La documentation est très succincte. Elle décrit essentiellement le schéma des messages sous la forme d'une page HTML, de schémas XML (XSD) et de fichiers CSV. Souvent, la sémantique des champs est ambiguë. Certains champs semblent redondants (par exemple, les champs « appname » et « app.name », « username » et « usr.name »). Il est souvent difficile de déterminer si les informations (adresse IP, nom d'utilisateur, etc.) sont relatives à la source de l'attaque, la cible ou la sonde.

*Transport et encodage* L'objectif de CEE était de permettre d'utiliser différents encodages et transports. A ce jour, seuls les encodages en JSON et en XML sont proposés [10]. Concernant le transport, le site dédié au format fournit un ensemble d'exigences et évoque seulement l'utilisation de Syslog avec un encodage JSON [11].

8. `https://cee.mitre.org/`
9. `https://cee.mitre.org/language/1.0-beta1/`
10. `https://cee.mitre.org/language/1.0-beta1/cls.html`
11. `https://cee.mitre.org/language/1.0-beta1/clt.html`

*Pouvoir d'expressivité* Le format, tel qu'il est décrit dans la version publiée sur le site de MITRE, est assez expressif. Toutefois, la sémantique exacte de beaucoup de champs est difficile à déterminer. Visiblement, le format ne permet pas de décrire la sonde de manière précise (adresse, processus, etc.) mais il est souvent difficile de savoir si les attributs sont relatifs à la source, la cible ou la sonde, chaque attributs n'étant décrits que par quelques mots. Par exemple, le champ « app » et ses attributs « app.name », « app.vend », etc. s'appliquent-t-il à la sonde ou à la cible ? Le problème se pose également pour le champ « proc » censé décrire un processus. La notion de « source » est ambiguë dans la documentation. Celle-ci évoque par exemple la notion de « event source », ce qui laisserait à penser qu'il s'agit de la sonde et non pas de la source de l'attaque. A priori, le format reprend les différents types d'informations proposés par IDMEF. Toutefois, IDMEF est plus complet et, surtout, la sémantique des champs IDMEF est moins ambiguë (par exemple, IDMEF permet clairement de préciser des informations relatives aux processus pour la sonde, la source et la cible). Il est donc difficile de traduire un événement CEE en IDMEF.

*Structuration* CEE est un format faiblement structuré offrant cependant une structuration intermédiaire entre les formats fortement structurés comme IDMEF et ceux reposant sur une structure totalement « à plat », comme CEF et LEEF. Il comporte 56 champs et sous-champs. Certains champs sont regroupés (utilisation de sous-champs). Par exemple, le champ « app » regroupe les sous-champs « app.name », « app.vend », etc. Le format propose quelques dictionnaires pour certains champs qui le nécessitent : « action », « domain », « object », « service » et « status ». Toutefois, certains dictionnaires mériteraient d'être complétés (par exemple « service »). En outre, certains champs associés à des dictionnaires semblent regrouper des informations très hétérogènes (par exemple « object » ou « action »).

*Extensibilité* En théorie, le format est extensible via l'ajout de profils additionnels (qui permettent d'ajouter des attributs et des dictionnaires). Toutefois, la documentation ne propose aucun exemple de mise en œuvre de profil additionnel.

### IDMEF

*Références* IDMEF est décrit dans la RFC 4765. Il s'agit d'un format ouvert mais le statut de la RFC est expérimental car le groupe de travail a été dissout avant que le résultat de son travail n'ait été approuvé. L'objectif de la RFC est donc seulement de documenter le travail réalisé. Le format IDMEF est censé répondre aux exigences formalisées dans la RFC 4766. Le protocole de transport recommandé, IDXP, est décrit dans la RFC 4767 comme un « profil » BEEP.

Dans l'ensemble, le format IDMEF est bien documenté par la RFC 4765 qui spécifie la structure des messages ainsi que chacun des champs en fournissant des exemples. La sémantique de chaque champ est précisée. Le document est relativement volumineux (157 pages) ce qui traduit la complétude et la précision de la

documentation. Toutefois, sa lecture peut rebuter l'utilisateur qui souhaite créer une alerte IDMEF comportant un nombre réduit de champs (ce qui correspond à un cas d'usage assez fréquent). En outre, le format imposé par l'IETF pour les RFC (document texte ASCII) ne permet pas de faire apparaître simplement et de manière immédiate la structuration du format IDMEF (diagramme de classe complet). Enfin, si la sémantique de chaque champ est assez claire, la stratégie de peuplement d'une alerte IDMEF peut être ambiguë (ce qui est dû en partie à la grande richesse du format). Par exemple, lorsqu'une application génère des journaux qui sont par la suite analysés par une sonde HIDS, l'application en question doit-elle être considérée comme un analyseur ou la cible de l'alerte ? De telles ambiguïtés peuvent conduire à un peuplement hétérogène des alertes par des éditeurs différents, ce qui n'est pas souhaitable.

*Transport et encodage* La RFC 4765 n'impose pas un encodage ni un protocole de transport et, a priori, différents encodages et transports peuvent donc être utilisés. Toutefois, la RFC 4765 illustre le format en utilisant XML et fournit un schéma sous la forme d'une DTD. Par ailleurs, la RFC 4767 décrit le protocole IDXP dont l'utilisation était préconisée par le groupe de travail ayant œuvré au développement d'IDMEF. En pratique, ce protocole n'a, à notre connaissance, jamais été utilisé dans les implémentations existantes.

*Pouvoir d'expressivité* IDMEF est un format très riche qui permet d'exprimer différents types d'informations relatives à une alerte. Typiquement, il est possible de préciser une ou plusieurs sources d'attaques, une ou plusieurs cibles, l'attaque ou le comportement suspicieux, la sonde de détection ainsi que des informations temporelles. Il est également possible de préciser les informations relatives aux adresses réseau, aux services, processus, fichiers et utilisateurs impliqués dans la source, la cible ou la sonde.

*Structuration* IDMEF est un format très structuré, orienté objet, comportant 166 attributs différents répartis dans 33 classes. Beaucoup de classes sont agrégées (certains attributs sont eux-mêmes des classes). Le format a parfois recours à l'héritage mais cela est utilisé seulement pour préciser le type d'alerte (classes `CorrelationAlert`, `ToolAlert` et `OverflowAlert`) ou le type de service (classes `SNMPService` et `WebService`).

Beaucoup d'attributs sont des listes (multiplicité supérieure à un) : `Source`, `Target`, `Address`, `Reference`, etc. Cela permet de modéliser le fait qu'une attaque puisse avoir plusieurs sources ou cibles, qu'un équipement puisse avoir plusieurs interfaces (donc plusieurs adresses), etc. Ces champs sont également parfois couplés avec des champs de type « catégorie » et permettent dans ce cas de préciser différents types d'adresses ou d'identités correspondant à une même interface ou un même utilisateur. Les valeurs possibles du champ catégorie sont alors précisées dans un dictionnaire. Il s'agit d'une spécificité d'IDMEF dans le domaine des formats d'alertes. Cela souligne la richesse du format : avec seulement deux attributs, il est possible d'exprimer différents types d'adresses ou

d'identité, là où les autres formats ont plutôt tendance à dédier un attribut distinct à chaque type d'adresse ou d'identité. Cela permet également de facilement étendre le format sans augmenter le nombre d'attributs : il suffit de modifier le dictionnaire associé au champ catégorie.

De manière générale, IDMEF a souvent recours à des dictionnaires. Cela permet d'avoir des résultats facilement comparables d'une implémentation à une autre. Toutefois, ces dictionnaires ne sont pas toujours complets et nécessiteraient une mise à jour. En outre, l'usage de dictionnaires pourrait être étendu à d'autres champs (typiquement, les catégories d'attaques).

*Extensibilité* IDMEF propose nativement un mécanisme d'extension sous la forme d'un champ dédié (`AdditionnalData`). Il est également possible d'étendre le format de manière non standard par héritage ou en étendant les dictionnaires.

**CIM** Common Information Model est un standard du Distributed Management Task Force [12], un organisme états-unien de standardisation. Le DMTF est une organisation ouverte aux entreprises, organisations et personnes physiques qui développent des standards pour l'administration des systèmes informatiques distribués. Les travaux autour de CIM font l'objet de plusieurs groupes de travail au sein du DMTF. Il s'agit en effet du format central qui est utilisé par d'autres standards du DMTF (par exemple, WBEM).

CIM permet de représenter l'ensemble des composants d'un SI ainsi que les mécanismes nécessaires pour administrer et superviser ces composants. L'utilisation du format correspond donc à un périmètre beaucoup plus large que la simple supervision de sécurité. La description des alertes de sécurité représente en réalité un sous-ensemble du format, ce sous-ensemble étant, à la date de rédaction de cet article, considéré comme « expérimental ». Par la suite, la description que nous faisons se concentre sur ce sous-ensemble relatif aux alertes.

Il est à noter que CIM, dans son ensemble, est un format qui semble mature et qui a acquis une certaine notoriété pour l'administration des systèmes distribués. Il est notamment utilisé dans la solution WMI de Microsoft. Il est possible d'utiliser CIM pour décrire les composants (par exemples les sources, les cibles ou les sondes) dans d'autres formats d'alertes (XDAS semble envisager cette solution [13]).

*Références* CIM fait l'objet d'une spécification du DMTF (DSP). L'ensemble des documents de spécification est accessible publiquement sur le site internet du DMTF [14]. Le site référence les différentes versions du format et les documents correspondants.

12. `http://www.dmtf.org/`
13. `http://scap.nist.gov/events/2011/emapdd/presentations/EMAP%20-%20The%20Open%20Group%20Distributed%20Audit%20Services%20%28XDAS%29%20v2.pdf`
14. `http://www.dmtf.org/standards/cim`

La spécification comprend la description du méta-modèle, du langage IDL associé (MOF) ainsi que le schéma standard. Le méta-modèle est utilisé pour les évolutions du schéma ou la création d'extensions. Les différentes versions du schéma sont décrites sous forme de diagrammes UML. Ceux-ci sont fournis dans différents formats (MOF, XML, XSD, Visio et PDF). Une documentation d'API est également fournie sous forme HTML [15]. Cette documentation complète les schémas UML en décrivant les différents attributs de chaque classe.

La documentation est très complète et le format utilisé (schéma UML graphique associé à une description des différents champs) permet d'appréhender facilement et rapidement le format. Toutefois, CIM utilise souvent le concept d'héritage ce qui nécessite d'analyser les classes parentes pour identifier les champs hérités.

Le sous-ensemble de CIM relatif aux alertes de sécurité est relativement restreint et correspond au sous-schéma `CIM_Security`. Il s'agit principalement de la classe `SecurityIndication` qui étend la classe `AlertIndication`, issue du « cœur » de CIM. La classe `SecurityIndication` est héritée par la classe `IPNetworkSecurityIndication`. Cette dernière est elle-même héritée de la classe `IPPacketFilterIndication`. Ces classes sont considérées comme expérimentales.

*Transport et encodage* L'objectif du DMTF est de fournir un modèle abstrait standard qui puisse être implémenté dans différents langages. CIM peut donc en théorie utiliser n'importe quel format d'encodage et de transport. Le DMTF fournit un schéma XML. En outre, CIM est utilisé dans WBEM, un standard permettant la gestion à distance d'un SI distribué. Ce standard repose sur un transport HTML et un encodage XML de CIM (CIM-XML).

*Pouvoir d'expressivité* L'expressivité du format, en ce qui concerne les alertes de sécurité, est assez limitée. Le format ne permet pas de décrire précisément la sonde (son adresse, etc.). Aucun champ n'est prévu pour décrire les processus, les fichiers ou les utilisateurs. Cela est paradoxal car le format CIM, dans son ensemble, possède un fort pouvoir d'expressivité, permettant notamment de décrire les différents nœuds réseau et les services qu'ils hébergent. Toutefois, le schéma `CIM_Security` ne réutilise pas les classes fournies par le cœur de CIM. Seule la classe `AlertIndication` est utilisée via l'héritage.

*Structuration* CIM dans son ensemble est un format très structuré, suivant une approche orienté objet. Toutefois, les classes implémentant les alertes de sécurité se contentent d'hériter de la classe `AlertIndication` et n'utilisent que des champs de types « primitifs » (chaînes de caractères, entiers, etc.). Ces classes n'utilisent pas d'agrégation (aucun de leur champs ne correspond à une autre classe de CIM). On peut donc considérer que ce sous-ensemble du format est relativement peu structuré. Les classes relatives aux alertes comprennent 58 champs. Le format tend à utiliser des champs différents pour des informations

15. `http://schemas.dmtf.org/wbem/cim-html/`

de même nature (différents champs adresse). Le format propose quelques dictionnaires. Toutefois, il est souvent possible de spécifier des valeurs « libres » dans un champ lié à un champ restreint par un dictionnaire. Par exemple, le champ `SecurityIndication.MoreSpecificResources` complète le champ `SecurityIndication.Resources`, dont les valeurs sont restreintes par un dictionnaire.

*Extensibilité* CIM permet d'étendre le schéma standard par héritage.

**XDAS/CADF** XDAS est un standard défini dans les années 80 par des acteurs du monde UNIX réunis dans un groupe de travail de l'Open Group [16], un consortium de normalisation neutre et indépendant. Peu d'implémentations de ce standard ont vu le jour mais en 2007 le groupe de travail a été reformé, notamment à l'initiative de Novell [17]. Une implémentation open-source a alors été proposée : openXDAS [18]. Le groupe envisage visiblement la publication d'une nouvelle version mais, à la date de rédaction de ce présent document, cette nouvelle version n'est pas disponible sur le site de l'Open Group. Il semble que l'activité de Novell concernée par ce travail ait été cédée à NetIQ [19]. Récemment, le groupe de travail a initié une collaboration avec le DMTF pour intégrer XDAS V2 aux standards CADF et CIM du DMTF [20]. Les personnes concernées ont visiblement également participé au groupe de travail CEE. Actuellement, le DMTF a publié une version stable du format CADF qui reprend les principes communs aux formats de la « famille » XDAS. Il s'agit à l'heure actuelle du format le plus abouti et le plus actif de cette famille. C'est donc cette version du format que nous avons considérée dans l'étude détaillée des champs des différents formats.

Au départ, le format est dédié à la génération et au filtrage d'événements en général au niveau du système d'exploitation (UNIX). En pratique, il est plutôt adapté pour des événements provenant d'un système d'exploitation, en particulier ceux liés à l'authentification. La première version définit un schéma de manière succincte, une taxonomie d'événements ainsi qu'une API. La version fournie par NetIQ se concentre sur la définition du schéma et de la taxonomie correspondante. Le format CADF du DMTF est dédié à l'audit en général mais dans le contexte particulier du Cloud.

*Références* Les spécifications de la version originelle (XDAS V1) du format XDAS sont disponibles publiquement sur le site de l'Open Group [21]. NetIQ do-

16. `http://www.opengroup.org/`
17. `https://collaboration.opengroup.org/projects/security/xdas/`
18. `http://openxdas.sourceforge.net/`
19. `https://www.netiq.com/documentation/idm401/idm_sentinel/data/bqxvslh.html`
20. `http://www.opengroup.org/node/3037`
21. `https://www2.opengroup.org/ogsys/catalog/P441`

cumente la version de XDAS utilisée dans ses produits [22] (qui semble préfigurer le format XDAS V2). Le DMTF a publié une version stable de CADF sous la forme d'un document de spécification (DSP0262) sous format pdf. Cette spécification est complétée par d'autres documents qui illustrent l'implémentation de CADF au sein d'OpenStack. L'ensemble des documents est disponible sur le site Internet du DMTF [23].

Les différentes versions de XDAS reprennent des concepts communs, notamment le triplet « originator/observer », « initiator » et « target » qui correspondent peu ou prou aux concepts de « sonde », de « source » et de « cible » d'IDMEF. Toutefois, des différences notables apparaissent dans la définition des champs de chacune de ces classes suivant les formats.

La documentation du format XDAS V1 est très succincte (les champs ne sont pas toujours identifiés ni nommés clairement). La documentation fournie par NetIQ est succincte mais tous les champs y sont clairement identifiés. Toutefois, la sémantique de chaque champ est décrite très brièvement. En outre, la composition des champs agrégés n'apparaît clairement que sur la description de l'encodage en JSON. Le document de spécification de CADF est très complet (183 pages). Il précise clairement les objectifs et les concepts de ce format. La signification de chaque champ est explicitée ainsi que la philosophie générale du format, notamment à travers différents exemples. La spécification décrit en outre les types primitifs (entier, date, chaîne de caractères, etc.) et structurés qui sont ensuite utilisés pour spécifier les types des différents champs.

*Transport et encodage* A priori XDAS devrait être indépendant du transport et de l'encodage. Toutefois, NetIQ fournit un schéma JSON. Les exemples fournis laissent à penser que Syslog est utilisé comme transport. La spécification de CADF n'impose ni le transport ni l'encodage. Elle fournit des règles pour l'encodage en XML et en JSON, ainsi que des exemples utilisant ces deux encodages pour chaque champ du format.

*Pouvoir d'expressivité* Il est difficile d'évaluer précisément le pouvoir d'expressivité de XDAS en raison des différences entre les versions. Les concepts commun permettent d'exprimer des informations relatives à la sonde, la source et la cible. Toutefois, la version XDAS V1 et la version documentée par NetIQ proposent un nombre limité de champs pour chacune de ces classes. CADF offre un plus grand nombre de champs. Les formats XDAS et CADF ne proposent pas de champ permettant d'exprimer des informations relatives aux processus ou aux fichiers.

*Structuration* XDAS et CADF sont des formats relativement structurés. Ils s'appuient essentiellement sur trois classes qui elles-même comprennent des champs agrégés génériques (Entity, Account...). Toutefois, le nombre de champs est limité : CADF propose 48 champs. La version initiale du format propose une

22. `https://www.netiq.com/documentation/edir88/edirxdas_admin/data/bqppfzw.html`
23. `https://www.dmtf.org/standards/cadf`

taxonomie des événements (un dictionnaire de « catégories » d'événements) reprise par la version de NetIQ. Toutefois, ces catégories sont plutôt orientées « authentification ». CADF définit trois types de taxonomies : *Resource*, *Action* et *Outcome*. La première permet de décrire une ressource (associée à la source, la sonde ou la cible) sous la forme d'un arbre spécifié par une URI. Action permet d'exprimer le type d'activité décrite dans l'alerte sous une forme hiérarchique similaire à la précédente. Outcome permet de spécifier le résultat de cette action. Il s'agit d'un simple dictionnaire beaucoup plus limité que les deux précédents.

*Extensibilité* La documentation de NetIQ précise que n'importe quel champ additionnel peut-être ajouté dans le format pour compléter les champs standards. La spécification de CADF précise que le format peut-être étendu en publiant des profils qui respectent les règles principales de la spécification.

### 3.2 Synthèse des résultats

Le tableau 1 présente les résultats de l'analyse détaillée de chaque champ des différents formats.

**Table 1.** Richesse relative des formats

| **Format** | IDMEF | CEF | LEEF | CIM | CEE | CADF |
|---|---|---|---|---|---|---|
| **Nombre de champs** | 166 | 117 | 50 | 58 | 56 | 48 |
| **Nombre de champs normalisés** | 259 | 84 | 49 | 48 | 49 | 76 |
| **Nombre de champs traduisibles** | 259 | 65 | 20 | 29 | 39 | 72 |
| **Nombre de champs non traduisibles mais pertinents** | 0 | 15 | 11 | 11 | 5 | 3 |
| **Nombre de champs non traduisibles et peu pertinents** | 0 | 4 | 18 | 8 | 5 | 1 |
| **Nombre de champs pertinents** | 248 | 80 | 31 | 40 | 44 | 75 |
| **Couverture du format IDMEF** | 100 % | 25 % | 8 % | 11 % | 15 % | 28 % |
| **Richesse relative par rapport à IDMEF** | 100 % | 32 % | 13 % | 16 % | 18 % | 30 % |

La deuxième ligne de ce tableau comptabilise le **nombre de champs** proposés par la spécification de chaque format. Toutefois, lorsque nous souhaitons comparer ces champs, il apparaît assez rapidement que la granularité n'est pas toujours la même selon les formats. Ainsi, certains formats comme IDMEF ou CADF proposent des constructions permettant de stocker différentes informations à l'aide d'un nombre plus restreint de champs que les autres formats. A l'inverse, certains formats utilisent des champs distincts pour stocker des informations qui sont regroupées dans un seul champ par d'autres formats. Pour réaliser une étude comparative quantitative, nous avons calculé un **nombre de champs normalisés** (ligne 3) pour chaque format. Ce calcul s'appuie sur une représentation canonique arbitraire. Nous avons ensuite augmenté le nombre de champs pour les formats qui, par rapport à cette représentation canonique, stockent l'information dans un nombre plus restreint de champs. A l'inverse, nous avons diminué le nombre de champs des formats qui disséminent la même information dans des champs distincts. Cette métrique permet de mesurer quantitativement la richesse des différents formats de manière homogène. Cette première étape fait apparaître clairement que le format IDMEF est celui qui propose le plus grand nombre de champs.

Pour chaque format, nous avons analysé l'ensemble des champs et nous avons tenté de les traduire dans le format IDMEF, qui paraît le plus complet. Nous avons ainsi comptabilisé le nombre de champs, après normalisation, qu'il était possible de traduire de la sorte (**nombre de champs traduisibles**, ligne 4). Cela nous permet de calculer la **couverture du format IDMEF** (ligne 8). Nous définissons cette métrique comme le ratio du nombre de champs qu'il est possible de traduire par rapport au nombre total de champs d'IDMEF, après normalisation.

Pour les champs impossibles à traduire, deux cas de figure se présentent. Le format peut proposer des champs pertinents non prévus par IDMEF (**nombre de champs non traduisibles mais pertinents**, ligne 5). Ces champs constituent des pistes intéressantes pour étendre ou mettre à jour le format IDMEF. Nous avons également identifié des champs dont la sémantique n'était pas claire, qui sont très dépendants d'une solution ou dont l'intérêt pour le domaine des formats d'alertes de sécurité n'est pas évident (**nombre de champs non traduisibles et peu pertinents**, ligne 6). Nous avons également identifié le **nombre de champs pertinents d'un format** (qui peuvent être traduits en IDMEF ou non). Pour IDMEF, nous avons soustrait le nombre les champs que nous proposons de supprimer ou de rendre obsolètes (cf. section 4). Ces résultats nous permettent de calculer la **richesse relative par rapport à IDMEF**. Nous définissons cette métrique comme le ratio entre le nombre de champs pertinents du format par rapport à celui d'IDMEF.

Ces résultats confirment qu'IDMEF présente, de loin, l'expressivité la plus importante même s'il existe des champs proposés par d'autres formats qui ne peuvent être exprimés en IDMEF.

Le tableau 2 synthétise les résultats de l'analyse comparative des différents formats selon les critères définis en section 2.2. Comme décrit précédemment,

**Table 2.** Synthèse

| **Format** | CEF | LEEF | IDMEF | CIM (sec) | XDAS (CADF) | CEE |
|---|---|---|---|---|---|---|
| **Origine** | HP | IBM | IETF (RFC 4765) | DMTF | The Open Group & DMTF | MITRE |
| **Expressivité** | ++ | − | +++ | + | ++ | +? |
| **Structuration** | −− | −− | +++ | + | ++ | − |
| **Transport** | Syslog | Syslog | IDXP | HTML | Syslog | Syslog |
| **Encodage** | Syslog + clé / valeur | Syslog + clé / valeur | XML | HTML | JSON XML | JSON XML |

IDMEF se démarque clairement des autres formats en ce qui concerne l'expressivité. CEF et, dans une moindre mesure, CADF, sont également des formats très expressifs. CEF couvre la plupart des catégories proposées par IDMEF et il propose un certain nombre de champs pertinents qui ne sont pas présents dans IDMEF. Les autres formats sont en retrait. Toutefois, tous les formats étudiés permettent d'exprimer les champs les plus courants et les plus utiles dans le domaine (adresses de la source, de la cible, message, description de la sonde, etc.)

Concernant la structuration, les formats se rangent en deux catégories bien distinctes. La première catégorie correspond aux format peu structurés qui, tel CEF, on adopté une structure « à plat » sous la forme d'un ensemble de couples clé/valeur. A l'inverse, les formats très structurés comme IDMEF ou XDAS utilisent des classes agrégées. Cette solution permet de regrouper les champs selon le type d'information qu'ils sont censés décrire. Les défenseurs d'une solution « à plat » plaident pour une plus grande simplicité. Toutefois, cela reste vrai uniquement si le nombre de champs reste faible. Ceci est en contradiction avec le critère précédent : l'expressivité suppose un nombre de champs important. Dès lors, la forme structurée facilite la compréhension. On peut également noter qu'un format structuré peut facilement être transformé sous une forme clé/valeur : il suffit d'encoder la clé en concaténant les différentes classes agrégées jusqu'au champ souhaité.

Concernant le transport et l'encodage, les formats s'appuient principalement sur des encodages textuels classiques (XML, JSON) et des protocoles standards. En outre, tous les formats sont en principes indépendants de l'encodage et du transport et pourraient a priori utiliser d'autres encodage/transport que ceux préconisés. C'est particulièrement vrai pour les formats issus de standard pour

lesquels le transport et l'encodage ne sont généralement proposés qu'à titre d'exemple (la spécification de ces formats n'impose aucun encodage ou transport particulier).

## 4 Pistes d'améliorations de l'existant

IDMEF apparaît comme le format le plus adéquat pour l'échange d'alertes. Ce résultat n'est guère surprenant puisqu'il s'agit d'un format dédié à ce besoin. Toutefois, l'étude fait apparaître quelques lacunes que nous envisageons de corriger en proposant de faire évoluer le format. Nous pouvons distinguer trois types de lacunes :

- la complexité du format et de sa documentation ;
- l'absence de mise-à-jour ;
- l'absence de certaines informations qui sont proposées par d'autres formats ;
- l'absence d'implémentation supportant différents types de transports et d'encodages.

En pratique, IDMEF n'est pas si compliqué qu'il n'y parait à premier abord. Il est ainsi relativement aisé de construire une alerte avec les champs les plus courants, IDMEF n'imposant que très peu de champs obligatoires. Toutefois, la grande richesse du format peut dérouter le néophyte. La forme de la RFC peut également constituer un frein. Il convient donc selon nous de compléter la documentation du format par des tutoriaux permettant d'illustrer en pratique l'utilisation d'IDMEF sur différents cas de figure représentatifs des différentes catégories de sondes couramment utilisées de nos jours (NIDS, firewall, Anti-Virus, WAF, etc.). En outre, nous avons réalisé des diagrammes UML permettant de mieux visualiser la structure du format, notamment via l'utilisation de couleurs pour différencier les différentes catégories de classes. Ces diagrammes permettent à la fois d'avoir une vue d'ensemble et de zoomer sur une sous-partie afin d'identifier, par exemple, l'ensemble des champs possibles pour une classe donnée.

IDMEF souffre également de son âge et de l'absence de mise-à-jour. Ainsi, le format reflète les problématiques qui étaient d'actualité lors de sa création. Certains aspects peuvent aujourd'hui apparaître comme moins cruciaux tandis que d'autre sont sous-représentés (par exemple, les services Web). A l'inverse, l'analyse des formats concurrents nous a permis d'identifier un certain nombre de champs qui ne sont pas présents dans IDMEF mais qui semblent pertinents.

Enfin, il nous paraît important de proposer une implémentation de référence, sous la forme d'une bibliothèque utilisable dans différents langages, supportant différents types de transport et d'encodage. En effet, à l'heure actuelle, peu de produits implémentent IDMEF. Il n'existe pas d'implémentation générique (i.e. qui ne soit pas spécifique à un produit, notamment en termes de transport et d'encodage) et utilisable par tous. Cette limitation n'est pas propre à IDMEF mais il nous parait essentiel de fournir une telle implémentation pour facili-

ter l'adhésion. Le développement de cette bibliothèque, en cours de réalisation, constitue l'un des objectifs du projet SECEF.

Nous présentons par la suite certaines pistes d'amélioration que nous envisageons. Ces pistes seront analysées dans la suite du projet SECEF afin de proposer des évolutions du format IDMEF. Les travaux sur ce sujet sont en cours de réalisation au sein du projet.

Nous avons identifiés quatre types d'évolution :

— l'ajout de champs dans les classes existantes ;
— la suppression (ou la déclaration obsolète) de certaines classes ou mécanismes proposés dans IDMEF V1 ;
— l'ajout de classes ;
— la mise à jour des dictionnaires ;

Nous détaillons par la suite ces différents points d'amélioration.

### 4.1 Ajouts de champs dans les classes existantes

L'étude des différents formats nous a permis d'identifier certains champs manquant dans IDMEF.

**Identification de la fin d'une attaque** IDMEF propose plusieurs champs d'horodatage pour spécifier l'observation d'un événement ou l'émission de l'alerte. Ces champs sont typiquement associés au début d'un événement. Il serait intéressant de pouvoir spécifier la fin d'un événement, notamment pour les attaques telles qu'un scan de port. Ce type de champs est proposé par CEF et certaines versions d'XDAS.

**Identification du protocole de transport** IDMEF permet d'identifier le protocole et le port associé à un service. Toutefois, ces champs permettent a priori de spécifier le protocole de plus haut niveau qu'il soit possible d'identifier. Dans beaucoup de cas, il s'agit d'un protocole applicatif comme HTTP ou DNS. Il parait intéressant de pouvoir également préciser le protocole de transport utilisé, notamment pour des protocoles applicatifs qui peuvent utiliser différents protocoles de transport (par exemple DNS). Les formats CEF, LEEF et CIM proposent un champ distinct pour spécifier ce type d'information.

**Identification des interfaces d'entrées et de sortie** IDMEF permet de spécifier une interface réseau pour la source et la cible d'une alerte. Toutefois, le format ne permet pas de spécifier d'interface pour l'analyseur. En outre, il n'est pas possible de distinguer l'interface d'entrée de l'interface de sortie empruntées par un paquet réseau. Ce type d'information paraît pertinent pour les alertes remontées par des sondes réseau qui possèdent de multiples interfaces tels les routeurs, pare-feux ou NIPS/NIDS. CEF propose des champs permettant d'identifier l'interface d'entrée et de sortie de la sonde à l'origine de l'alerte.

**Gestion explicites des données de géolocalisation** IDMEF permet de spécifier la géolocalisation d'un noeud, qu'il s'agisse de la source, de la cible ou de l'analyseur. Toutefois, cette information est contenue dans un unique champ dont la sémantique est vague (il s'agit d'une chaîne de caractère). Suivant les implémentations et les déploiements, la structuration des données stockées dans ce champ peut varier, ce qui constitue une limitation pour les traitements automatiques. Il parait nécessaire de préciser le format de ce champ et/ou de proposer des champs additionnels distincts permettant d'identifier les données de géolocalisation (longitude/latitude/altitude, coordonnées GPS, etc.). Le format CADF propose de tels champs.

**Attachement du log originel** IDMEF permet d'attacher des données issues des journaux à l'aide du champs `AdditionnalData`. Toutefois, ce champ n'est pas dédié à cet usage et permet d'étendre le format de manière général. Il parait intéressant de proposer un champ dédié permettant d'embarquer le log originel ayant permis de créer une alerte, comme le proposent CEE et XDAS.

**Identification du thread** IDMEF permet d'identifier le numéro de processus (PID) mais pas le numéro de thread. Une modification mineur consiste à ajouter un champ dédié pour cette information comme le propose le format CEE.

**Identification de la catégorie d'un noeud** IDMEF permet de spécifier la catégorie d'un analyseur (champ `Analyzer.Class`). Cela permet par exemple de regrouper des équipements différents (nom ou vendeur différents) offrant la même fonctionnalité (par exemple NIDS, parefeux, HIDS, etc.). Toutefois, IDMEF ne permet pas d'associer une catégorie aux autres noeuds (la source et la destination d'une alerte). Il parait intéressant d'ajouter des champs similaires aux classes `Alert.Source` et `Alert.Target` ou de déplacer ce champ dans la classe `Node` afin de pouvoir associer une catégorie à chaque noeud (par exemple serveur, poste client, routeur, etc.).

**Compteur d'occurence** IDMEF permet de spécifier des alertes de corrélation qui peuvent agréger différentes alertes primaires. Pour cela, IDMEF propose un champ qui contient l'identifiant de toutes les alertes agrégées. Cela suppose que ces alertes aient été transmises. Parfois, il peut être nécessaire d'agréger un nombre important d'alertes très similaires afin de limiter le nombre d'alertes transmises. Dans ce cas de figure, il parait intéressant d'indiquer le nombre d'alertes agrégées dans un champ dédié de la classe `CorrelationALert`. Un tel champ est proposé par les formats CEF, CIM et XDAS.

**Domaine d'authentification** IDMEF permet de spécifier les identifiants associés à un utilisateur. Toutefois, ces identifiants n'ont de sens qu'au sein d'un « domaine d'identification ». Ce dernier peut correspondre à un noeud réseau

dans le cas de comptes locaux. Il peut également s'agir d'un serveur LDAP ou d'un domaine ActiveDirectory. Il parait intéressant d'ajouter un champ dédié associé à chaque identifiant afin de pouvoir préciser son domaine. LEEF, CEE et XDAS proposent un tel champ.

**Catégorie d'utilisateur** IDMEF permet de spécifier les privilèges associés à un utilisateur via le champ `User.UserID`. Toutefois, il s'agit d'une vision très « bas niveau ». Il paraît intéressant de pouvoir spécifier une catégorie d'utilisateur (par exemple administrateur, utilisateur authentifié, utilisateur anonyme, etc.). CEF, LEEF et XDAS proposent des champs dédiés à cet usage.

### 4.2 Obsolescence de certains mécanismes d'IDMEF

L'analyse détaillée du format IDMEF et le retour d'expérience lié à son utilisation permettent d'identifier certains points qui sont peu utilisés ou paraissent peu pertinents aujourd'hui. Nous proposons de supprimer ces point dans la future version du format ou du moins de les déclarer obsolètes pour garantir une compatibilité descendante.

Ainsi IDMEF propose un mécanisme d'identification unique des classes via le champ `ident`. Ce mécanisme permet d'associer un identifiant unique à chaque instance de classe. Une instance de classe est définie par un ensemble de valeur identique pour les différents champs qui la composent. L'objectif de ce mécanisme était de compresser les messages en évitant de répéter des informations déjà spécifiées au préalable mais la documentation de la RFC est peu claire à ce sujet. En pratique, il n'est pas utilisé dans les implémentations. En outre, il apparaît plus pertinent de gérer la compression au niveau de l'encodage et du transport, en utilisant des mécanismes standards. Par exemple, il est possible d'utiliser différentes techniques de compression des messages XML (gzip, XGrind, XMill, etc.). Nous proposons de déclarer ces champs obsolètes ou de préciser leur utilisation dans un cadre plus actuel. Ainsi, cet identifiant pourrait servir de « clé externe » pour référencer des objets d'un autre format, par exemple les formats de description de la topologie.

IDMEF propose la classe `OverflowAlert` qui hérite de la classe `Alert`. L'idée était de spécialiser les alertes suivant le type d'attaque. Toutefois, il s'agit de la seule classe fille proposée par le format dans ce but. Cela pose un problème d'homogénéité et de représentativité des attaques modernes. En outre, les informations supplémentaires qu'elle permet de spécifier sont très précises et spécifiques (taille et nom du buffer). En pratique, peu de sondes sont capables de renseigner ces champs et l'intérêt de ce type d'information pour un traitement automatisé des alertes n'est pas évident. Nous proposons de supprimer cette classe ou de la déclarer obsolète.

### 4.3 Ajout de classes

IDMEF propose deux classes héritants de la classe `Service` : `WebService` et `SNMPService`. Il s'agit, comme dans le cas de la classe `OverflowAlert` de

spécialiser la classe mère en fonction du type de service. En l'état, le nombre de classes filles proposées n'est pas suffisant pour couvrir les services les plus courants. Cela traduit encore une fois un problème d'homogénéité. Toutefois, il parait nécessaire de compléter le schéma car les différents services nécessitent de spécifier des informations différentes. Nous proposons donc d'ajouter des classes pour couvrir les services les plus courants (par exemple LDAP et SIP). En outre, la classe `WebService` mérite d'être étoffée. En particulier, il parait intéressant d'ajouter des champs dédiés permettant de spécifier des paramètres tels que host, referer ou cookie.

### 4.4 Mise à jour des dictionnaires

IDMEF utilise des dictionnaires pour contraindre le type de certains champs. Cette pratique est louable car elle permet d'homogénéiser les valeurs. Toutefois, elle pose le problème de la mise-à-jour de ces dictionnaires afin de tenir compte des nouveaux usages et de l'évolution de l'éco-système en général. Actuellement, ces dictionnaires sont définis dans la spécification du format. Pour faciliter leur mise à jour, nous envisageons de les extraire et de les gérer indépendamment, par exemple en s'appuyant sur l'IANA pour maintenir les énumérations.

En pratique, les mises-à-jour devraient notamment porter sur les dictionnaires associés aux champs suivants :

— `Impact.Type` ;
— `Action.Category` ;
— `Reference.Origin` ;
— `Node.Category` ;
— `File.fsType` ;
— `Checksum.Algorithm` ;
— `Address.Category`.

En particulier, nous envisageons d'étendre ce dernier dictionnaire afin de pouvoir spécifier des adresses translatées (utilisation du NAT).

## 5 Conclusion

Nous avons présenté dans cet article une étude comparative de différents formats d'alertes réalisée dans le cadre du projet SECEF. Les résultats de cette étude montrent que tous les formats permettent de spécifier les champs les plus couramment utilisés lors de la génération d'alerte. En outre, ils sont tous relativement indépendants de l'encodage et du transport, même si la spécification de certains formats s'appuie sur un encodage ou un transport particulier, essentiellement à des fins d'illustration. Toutefois, des différences significatives existent concernant l'expressivité et la structuration de ces formats. Les résultats font clairement apparaître la supériorité d'IDMEF dans ce domaine, ce qui n'est guère surprenant car il a été développé spécifiquement pour l'échange d'alertes de sécurité, ce qui n'est pas le cas des autres formats. Il s'agit du format le

plus structuré et de celui qui offre la plus grande richesse en termes d'expressivité. Toutefois, cette étude a également permis d'identifier quelques lacunes d'IDMEF, notamment au regard des possibilités offertes par les autres formats. Ces résultats permettent d'identifier des pistes d'amélioration visant à créer de nouvelles classes, de nouveau champs mais également de gérer l'obsolescence de certains mécanismes d'IDMEF. Enfin, il est selon nous important de pouvoir mettre à jour les dictionnaires, ce qui nécessite peut-être de les dissocier de la spécification du format afin de pouvoir les faire évoluer plus fréquemment.

Les résultats de cette étude vont ainsi permettre au consortium de proposer une mise à jour du format IDMEF prenant en compte les pistes d'évolution identifiées. Le travail en cours consiste à valider les points qui feront l'objet d'une mise à jour et la forme adoptée (ajout/suppression de champs, de classe, etc.). En outre, l'étude comparative détaillée des différents champs nous a permis de constituer une table de conversion entre les différents formats et IDMEF. Cette table pourra servir de référence pour l'implémentation de passerelles entre les formats.

La présente étude s'est concentrée sur le format des messages (schéma et typage). Les problématiques d'encodage et de transport, bien qu'orthogonales aux problèmes évoqués dans cette étude, sont également d'une importance cruciale pour l'implémentation d'une solution efficace. Nous pensons que l'adoption d'un standard passe en grande partie par la mise à disposition d'outils de référence permettant d'utiliser ce standard. Un des objectifs futur du projet SECEF consiste donc à mettre à disposition une bibliothèque permettant d'échanger des messages IDMEF dans différents langages. Nous envisageons, pour cette bibliothèque, de supporter différentes forme d'encodage et de transport.

## Références


1. Debar, H., Curry, D., Feinstein, B. : The Intrusion Detection Message Exchange Format (IDMEF). RFC 4765 (Experimental) (March 2007), http://www.ietf.org/rfc/rfc4765.txt
2. Hollnagel, E., Paries, J., Woods David, D., Wreathall, J. : Resilience engineering in practice : A Guidebook. Ashgate Studies in Resilience Engineering, Ashgate Publishing (Dec 2010), https://hal-mines-paristech.archives-ouvertes.fr/hal-00613345
3. Pawliński, P., Jaroszewski, P., Urbanowicz, J., Jacewicz, P., Zielony, P., Kijewski, P., Gorzelak, K. : Standards and tools for exchange and processing of actionable information. Tech. Rep. TP-04-14-999-EN-N, ENISA (November 2014), https://www.enisa.europa.eu/activities/cert/support/actionable-information/standards-and-tools-for-exchange-and-processing-of-actionable-information

# Convergence sûreté de fonctionnement et supervision de sécurité : une rationalisation nécessaire

## Catégorie : Didactique

Gilles Lehmann, CS

Gilles.Lehmann@c-s.fr

**Résumé** La résilience des systèmes d'information est aujourd'hui au cœur des préoccupations. D'une façon générale la résilience d'un système est sa capacité à continuer à fonctionner en cas de panne.

A l'origine, les pannes des systèmes d'information étaient essentiellement liées à des dysfonctionnements « techniques ». Aujourd'hui et avec la recrudescence des actes de cybercriminalité la panne peut aussi venir d'une agression externe. Pire encore elle peut venir d'une agression externe et être confondue avec une panne matérielle.

La supervision est ainsi un chaînon indispensable pour assurer la sûreté et la continuité de fonctionnement des systèmes d'information et de communication en anticipant pannes et attaques.

Historiquement, ce sont les performances qui sont supervisées et ces techniques de supervision sont désormais matures. Aujourd'hui un nouvel enjeu apparaît donc pour assurer la résilience des systèmes, la gestion de la sécurité. Pour assurer son fonctionnement, il ne suffit plus de surveiller le système mais aussi de le protéger contre des attaques éventuelles

Les sujets de performances et de sécurité sont cependant souvent traités par des intervenants différents au sein des équipes d'exploitation, et le premier réflexe est de construire deux systèmes distincts pour accomplir ces deux tâches de surveillance. A travers cet article, nous souhaitons démontrer que la distinction n'est pas aussi évidente et qu'il faut dès à présent réfléchir aux interactions et convergences entre les deux « mondes ». Cette convergence est souhaitable à plusieurs titres. Une meilleure efficacité, une meilleure utilisation des ressources matérielles et humaines mais aussi une rationalisation des coûts qui permet de meilleurs investissements et donc à nouveau plus d'efficacité. C'est donc d'un cercle vertueux qu'il s'agit.

Nous introduirons d'abord l'administration et la supervision de performances avant de présenter l'administration et la supervision de sécurité puis de mettre en évidence les rapprochements possibles entre ces deux métiers. Nous conclurons par une présentation des limites, des risques et des freins à cette convergence.

Keywords : sécurité, performances, sureté de fonctionnement, SIEM, NMS, SOC, NOC, SNMP, IDMEF, Log Management, Monitoring, Corrélation, Métrologie, Forensic, Hypervision, convergence sûreté de fonctionnement et sécurité, convergence NOC/SOC.

# 1 Supervision de performances

## 1.1 Introduction

La supervision de performances permet de connaître à tout instant l'état du système d'information en termes de performances et de savoir précisément si chaque composant du système fonctionne correctement ou non. Pour cela, des indicateurs de performance sont relevés périodiquement et comparés à des seuils fixés au préalable. Si un seuil est dépassé, le système déclenche une alarme.

## 1.2 Les fonctions de la supervision de performances

Les fonctions de la supervision de performances sont :

**La supervision des états :** elle constitue le fondement de la supervision. Il s'agit grâce à des collecteurs (et parfois des sondes déployées sur les équipements) de remonter en un point central la totalité des alarmes sur le réseau. On peut représenter ces alarmes de plusieurs façons mais les plus classiques sont le « bac à alarmes » (un tableau dans lequel s'affichent toutes les alarmes avec une évaluation de leur gravité) ou une cartographie (une représentation « visuelle » du parc géographique, fonctionnel ou logique dans lequel l'exploitant peut naviguer). C'est l'outil principal des exploitants de premier niveau.

**La métrologie :** elle consiste à étudier dans le temps l'évolution des valeurs des indicateurs remontées par la supervision. C'est un complément d'information, en particulier pour la recherche de la cause fondamentale d'un événement mais aussi pour l'analyse de tendances et l'anticipation des problèmes. La métrologie est un des outils des exploitants de second niveau.

**Le reporting :** il génère des synthèses périodiques sous forme de graphiques et de tableaux et permet de fournir à la direction ainsi qu'aux utilisateurs finaux et aux clients une vision de la qualité de service rendu par le système d'information et son infrastructure. C'est l'outil de mesure des engagements. Il est utilisé par les exploitants mais aussi par leur direction.

# 2 Administration et supervision de sécurité

## 2.1 Introduction

En résumé, ce métier plus récent permet de connaître à tout moment l'état du système d'information en termes de sécurité.

Cette définition, certes un peu simpliste, met en évidence les similitudes entre la supervision de performances et la supervision de sécurité. Les objectifs et les effets sont identiques mais les causes différentes. Par exemple, un serveur peut tomber en panne pour des raisons distinctes, liées soit à une défaillance matérielle (identifiable grâce à la supervision de performances) soit à une attaque réseau (identifiable grâce à la supervision de sécurité).

### 2.2 Les fonctions de la supervision de sécurité

**Analyse temps réel :** Cette fonction identifie en temps réel des comportements suspicieux via l'analyse des journaux où grâce à des sondes IDS dédiées. L'ensemble de ces alertes est remonté à un manager central qui les stocke dans une base de données avant affichage dans un "bac à alertes".

**Archivage et indexation des traces :** Cette fonction consiste à centraliser, stocker et indexer l'ensemble des traces du système dans l'objectif de les analyser en cas d'incident.

**Reporting :** La brique reporting est celle qui se rapproche le plus naturellement de celle qu'on utilise en supervision de performances puisqu'elle a les mêmes fonctions. Dans le cas de la sécurité elle permet de produire périodiquement des états sur la sécurité globale du système : nombre de tentatives d'intrusion, pays d'origine des attaques, etc.

## 3 Problématiques communes

Analysons maintenant les problématiques communes à ces deux mondes :

### 3.1 Les fonctions

Au regard des descriptions ci-dessus il apparait clairement des similitudes entre les fonctions majeures des deux métiers.

TAB. 1: Fonctions

| **Performances** | **Sécurité** | **Commentaires** |
|---|---|---|
| Monitoring | Temps réel | Dans les deux cas on compare des valeurs à un référentiel et s'il y a divergence et en fonction de la divergence on déclenche une alarme/alerte associée à un équipement avec un niveau de gravité/criticité. |
| Métrologie | Archivage | On archive toutes les données qui ont permis d'activer la fonction temps réel puis on peut rechercher dans ces données la cause d'un incident ou bien analyser une tendance. |
| Reporting | Reporting | La fonction est ici identique et consiste à présenter l'activité sous une forme « didactique ». |

### 3.2 Les sondes

La supervision de performances peut s'effectuer à distance via SNMP sans « agents locaux ». Néanmoins, il est parfois nécessaire d'installer une sonde sur

les serveurs pour analyser des fichiers de journaux par exemple ou lancer une commande de test. C'est aussi le cas des sondes de sécurité qui analysent les logs à la recherche d'expressions type. On peut ainsi, dans certains cas, installer deux sondes sur le même serveur dont une partie de l'activité est commune. Au-delà des problèmes d'administration (déploiement, mise à jour, etc.) et de consommation de ressources, il arrive que ces deux sondes ne puissent fonctionner simultanément, leurs ressources étant identiques. Il serait donc pertinent, dans une architecture commune, de partager éventuellement ces sondes qui remonteront aux deux systèmes les informations nécessaires.

### 3.3 Analyse des journaux

L'analyse des journaux est au cœur de la supervision de sécurité. C'est en effet dans ces derniers qu'on peut identifier des comportements suspects comme les tentatives de connexion, les tentatives d'authentification, etc. Mais il ne faut pas oublier que les journaux ont toujours servi aussi à identifier des problèmes de fonctionnement (entre autres au démarrage du système). Il serait donc logique que l'analyse de ces journaux soit orientée sécurité et fonctionnement.

### 3.4 Archivage et indexation des journaux

Cette fonctionnalité est au cœur des outils de sécurité de "log management". Ils permettent de conserver des traces de l'ensemble de l'activité du système pour de l'analyse post-mortem suite à une tentative d'intrusion par exemple. Mais ce module peut tout aussi bien servir à rechercher les causes d'une panne de fonctionnement.

### 3.5 La corrélation

En termes de corrélation on peut donner comme exemple la corrélation topologique qui consiste à corréler les événements avec la topologie du réseau. Par exemple, si un serveur est derrière un routeur « en panne », il est inutile de traiter les événements signalant que ce serveur est non joignable tant que le routeur n'est pas rétabli. Cette remarque est valable que l'on se place du point de vue des performances comme de la sécurité. Il est donc intéressant de partager ces informations de topologie et de ne les saisir qu'en un seul point. La corrélation des événements est aussi nécessaire. Un serveur « tombé » à cause d'un disque défaillant (performances) peut provoquer des alarmes de sécurité car il n'est plus joignable. De la même façon si le serveur est attaqué par un DOS (Denial Of Service, problématique de sécurité) cela peut générer une alerte de performance sur sa consommation CPU. Dans les deux cas il est important que les événements de chaque système soient corrélés afin d'éviter des analyses inutiles de chaque côté à la recherche d'une cause qui est déjà connue.

### 3.6 Gestion de parc et inventaire

La première donnée nécessaire à la supervision est la connaissance aussi fine que possible du parc : quels sont les équipements, les serveurs, les logiciels, leurs versions, etc.

Ces informations ainsi que leurs évolutions sont nécessaires pour toute supervision. La fonction inventaire va permettre de suivre cette évolution. Lors d'une analyse sécurité il est important de connaître l'état de mise à jour d'un serveur, quelles sont les versions de logiciels installées, quelles sont les dernières opérations menées sur ce serveur (ont-elles provoqué une nouvelle faille ?), etc.

Il est important que la base d'inventaire soit partagée entre les différents mondes et qu'elle ne soit pas gérée en deux points distincts.

### 3.7 Les processus

Un effort important est conduit actuellement pour améliorer les processus d'administration et de supervision « standard ». Des méthodes comme la méthode ITIL en particulier visent à modéliser l'ensemble des processus et à apporter une meilleure cohérence. On y trouve la notion de gestion d'événements, de gestion d'incidents, la notion de CMDB (Configuration Management Data Base), etc. La gestion de la sécurité est en plusieurs points similaire à la gestion « standard », notamment les notions d'événement (alerte), d'incident (cause connue), de problème (cause inconnue). La gestion des interventions et de leurs documentations est similaire. La notion même de CMDB est utile tant pour les informations d'inventaire que pour celles de services aux utilisateurs. Là encore la convergence est nécessaire.

### 3.8 La télé-administration et le télé-déploiement

L'administration « standard » d'un grand parc nécessite de disposer d'un outil d'automatisation de nombreuses tâches : déploiement d'un logiciel, relance d'un service, changement d'une configuration, etc. Ce type d'outil est indispensable à l'administration de la sécurité (et en particulier au Maintien en Condition de Sécurité). En préventif, assurer la sécurité d'un système c'est, entre autres, déployer régulièrement les patches de sécurité. En réactif, la sécurité peut nécessiter le changement rapide et global de certaines configurations afin de se protéger : par exemple en isolant une partie du système. Il est donc intéressant de partager des outils de télé-administration.

### 3.9 L'analyse comportementale

En complément de l'analyse temps réel des comportements suspects, l'analyse comportementale peut permettre d'identifier des "signaux faibles" au sein de volumes importants de données. Le principe de base est de modéliser un comportement "normal" et de mesurer les divergences par rapport à cet étalon. L'analyse comportementale se fait classiquement aujourd'hui sur les journaux ainsi que sur les flux réseau. On peut imaginer transposer ces mécanismes sur les données de métrologie, voire même sur les données d'états.

### 3.10 Les équipes

Dans l'idéal une supervision de performances comme une supervision de sécurité doit fonctionner en 24/7. Historiquement, les équipes étant parfois séparées, ce poste représente une charge économique importante qui conduit souvent à réduire d'un côté ou de l'autre la couverture horaire. Si le premier réflexe consiste à justifier ce cloisonnement par une différence de compétence et de responsabilité, une analyse plus fine met en évidence que la problématique est un peu plus complexe. Il est important que la responsabilité finale de sécurité reste aux mains du Responsable Sécurité. C'est un problème de compétence mais aussi de responsabilité. Les intérêts du RSSI ne sont pas toujours compatibles avec ceux des exploitants classiques et il faut que les deux visions puissent être confrontées lors des décisions importantes.

Par contre, et c'est le poste le plus important économiquement, qu'en est-il des veilleurs, des superviseurs, des exploitants de niveau 1 qui doivent assurer une présence 24/7 devant les "bacs à alarmes" et les "bas à alertes" ? Ceux dont la principale activité consiste à surveiller ces bacs, traiter les "incidents connus" à l'aide de fiches réflexes et escalader les autres ? A l'évidence ces personnels peuvent être communs afin de fluidifier les actions mais aussi de réduire les coûts. Reste alors à organiser les niveaux d'escalade et de responsabilité supérieurs pour conserver un minimum de cloisonnement mais aussi de coopération entre les fonctions performances et sécurité.

## 4 Pistes de convergence

Forts de ces différents constats nous travaillons depuis plusieurs années sur cette convergence au travers de nos outils Vigilo (www.vigilo-nms.com : supervision de performances) et Prelude (www.prelude-siem.com : supervision de sécurité).

La route est longue tant ces deux mondes sont encore cloisonnés mais les intérêts sont indéniables. Parmi les pistes concrètes de convergence on peut mentionner les éléments suivants.

### 4.1 Un portail commun

Aujourd'hui nos deux applications peuvent être disponibles dans un "portail commun". La gestion des droits et des autorisations est en partie mutualisée au travers d'un annuaire ainsi que du SSO. Ce portail permet d'autre part de naviguer simplement d'une application à l'autre en fonction des besoins. Il est simple par exemple d'accéder à l'historique sécurité d'un serveur à partir du bac à alarmes de performances.

### 4.2 Des modules communs

A partir de ce portail on va retrouver des modules communs entre les deux outils.

**Les mêmes modules "externes" :**

*Gestion de l'inventaire :* l'inventaire contient classiquement l'ensemble des équipements et des logiciels déployés sur le système. La version des logiciels déployés est typiquement une information qui intéresse les deux parties. Le NOC pour les éventuelles mises à jour concernant les bugs et le SOC plus directement pour les patches de sécurité. Il est cependant parfois difficile de définir une priorité de mise à jour sur des parcs de grande taille, doit-on corriger les bugs en priorité ou se préserver des failles.

*Gestion des tickets :* la gestion des tickets est un des éléments fondamentaux de la convergence. C'est lui qui va permettre de définir des processus communs impliquant les deux mondes. C'est au travers des workflows que se définissent les rôles et prérogatives de chaque équipe.

*Outils de télé-déploiement :* ces outils déploient des correctifs quels que soit leur nature, ils sont intégrés dans le NOC comme dans le SOC.

*Base de connaissance :* il existe une culture déjà ancienne des fiches réflexes et autres fiches de procédures dans l'exploitation classique. C'est un domaine qu'on doit maintenant adapter à la sécurité. Pour cela on peut entre autres s'appuyer sur de bonnes pratiques éprouvées telles que ITIL. L'objectif étant en sécurité d'arriver à gérer les cas simples avec des profils techniques non experts qui sont capables d'assurer des surveillances H24/7, les cas plus complexes sont transférés aux analystes (comme un « problème » ITIL).

*etc.*

**Les modules "internes" :**

*Cartographie :* A partir de notre module cartographique de supervision de performances nous avons réalisé un module commun qui va afficher les informations des deux supervisions pour une vision unifiée. C'est une fonction qu'on retrouve à différents niveaux dans d'autres solutions. Cette surcouche de présentation permet de présenter sur des écrans communs un état de situation global.

*Reporting :* Le module de reporting est aussi commun aux deux applications, ce qui permet de générer des rapports complets de supervision et de tendances (sécurité et fonctionnement). Il permet ainsi de générer un rapport unique pour l'ensemble de l'activité du SI : pannes, attaques, contention, etc. Cette vision globale peut permettre d'anticiper de futurs problèmes ou d'identifier des problèmes en cours de façon plus efficace.

*Journaux :* Le module d'analyse des journaux peut être partagé entre les deux applications. Il est aujourd'hui essentiellement proposé dans les outils SIEM et les outils de gestion de journaux mais il peut être complémentaire à une supervision de performances classique pour identifier les causes d'une panne ou même anticiper un dysfonctionnement. Reste que les types de recherche et les outils associés pour la représentation ne sont pas forcément identiques.

*Bac à alertes/alarmes :* Nous travaillons enfin sur un bac "commun" qui permettrait une hypervision unifiée tout en conservant les bacs de chaque outil. Les informations des deux mondes présentent quelques similitudes. On retrouve une adresse IP cible que ce soit d'une panne ou d'une attaque, on retrouve une date et une éventuelle durée, un nombre d'occurrences. Une piste pour alléger la charge de surveillance globale est de proposer un bac alarmes/alertes commun qui pourrait ensuite rediriger vers des bacs plus spécialisés. Les formats et les informations des deux mondes sont en effet un peu différents et spécialisés pour faciliter l'enquête.

*Bus de communication :* Nos deux outils fonctionnent chacun autour d'un bus de communication applicatif offrant des possibilités classiques d'abonnement, de sécurité et résilience, etc. Historiquement les deux technologies ne sont pas identiques, nous travaillons donc à les connecter afin qu'alarmes et alertes puissent être reçues dans chaque module de chaque produit.

**Des interactions internes :** Le moteur de corrélation de sécurité peut s'appuyer sur des données de supervision pour confirmer ou au contraire infirmer une alerte, permettant ainsi de réduire le nombre de faux positifs.

On peut aussi imaginer intégrer à l'analyse comportementale (qui consiste souvent à mesurer un écart de comportement du système par rapport à un fonctionnement « normal ») des données telles que les données de monitoring (états), de métrologie (performances) pour identifier des comportements suspects.

## 5 Risques, limites et freins

Après avoir présenté tous les avantages de convergence NOC / SOC, arrêtons-nous un instant sur l'envers du décor, c'est-à-dire les risques, les limites et les freins à cette convergence.

### 5.1 Les éditeurs

Développer des outils communs ou convergents nécessite de disposer d'équipes multi-compétences. C'est une première difficulté pour les éditeurs, dont les produits sont souvent issus de rachats et de sociétés différentes. Se rajoute à ce point le poids d'un existant fort qui n'a pas prévu au départ de convergence et qu'il n'est pas simple aujourd'hui d'adapter si ce n'est aux limites (couche de présentation par exemple). Enfin les éditeurs souhaitent souvent conserver

un marché très segmenté quitte à contraindre les clients à acheter deux fois la même chose pour deux besoins pourtant similaires.

### 5.2 La technique

Pour ce qui est de la supervision de performances et de sécurité, même si elles partagent beaucoup de concepts elles conservent des spécificités :

**Les formats :** chaque domaine a ses formats et ses standards, et les données qui sont traitées ne sont pas de même nature.

**La corrélation :** si le mot est le même, le type de corrélation n'est pas identique. On peut imaginer « chainer » des corrélateurs mais il semble compliqué d'utiliser un seul et même corrélateur pour les deux mondes.

### 5.3 Les compétences en exploitation

Si on peut imaginer mutualiser des équipes de niveau 1, les niveaux 2 de chaque domaine requièrent des compétences particulières qui sont rarement communes. Les équipes exploitation et sécurité sont souvent disjointes, parfois délocalisées et n'ont pas toujours l'habitude de travailler de concert.

### 5.4 Les responsabilités

C'est probablement l'un des points les plus complexes. La sécurité par définition est souvent en opposition à la simplicité de fonctionnement du SI. Des flux en clair seront toujours plus simples à administrer que des flux chiffrés. Sauf dans le cas des toutes petites structures il est donc indispensable de conserver au final deux responsabilités bien distinctes entre l'exploitation et la sécurité. En cas de panne grave et pour assurer la résilience d'un système on peut par exemple avoir à décider faute de mieux de le « déprotéger », lui permettant ainsi de fonctionner mais offrant des surfaces d'attaque considérables. Ce type de décision doit donc être prise avec la plus grande précaution et sans contraindre un responsable à être juge et partie. Il en va de même avec les techniques de cloisonnement ou de mise en quarantaine en cas d'infection, qui par définition encore vont affaiblir la résilience du système en le privant de ressources voire altérer la mission qui lui est confiée.

## 6 Conclusion

La convergence des fonctions "NOC" et "SOC" est aujourd'hui indispensable. Non seulement elle permettra de réduire globalement les coûts d'exploitation de ces deux fonctions mais elle permettra aussi d'en améliorer l'efficacité globale.

C'est d'autant plus vrai aujourd'hui où la composante sécurité devient primordiale mais où les utilisateurs ont encore beaucoup de mal à s'équiper tant par manque de budget que par manque de retour d'expérience. En s'adossant à un domaine mature et largement maitrisé on pourra accélérer et améliorer la protection des systèmes d'information.

Le dernier frein est psychologique, mais c'est peut-être le plus résistant.

# Retours d'expériences de cyber-attaques et orientations pour une meilleure résilience

Frédéric CHOLLET (Solucom), Anthony DI PRIMA (Solucom)

frederic.chollet@solucom.fr, anthony.diprima@solucom.fr

**Abstract.** La présentation vise à délivrer des récents retours d'expériences d'authentiques cyber-attaques ayant affecté des SI tertiaires, notamment de nos clients : vecteurs d'attaques, modalités d'intrusion et de propagation latérales, déploiement de camps de base et principes d'exfiltration de données. Nous présenterons les grandes mesures réactives que nous déployons lorsque nous accompagnons nos clients victimes de cyber-attaques. Nous verrons dans quelle mesure ces techniques peuvent également s'appliquer aux systèmes industriels. Il s'agira de délivrer des orientations pragmatiques techniques et organisationnelles en matière de cyberrésilience et de lutte préventive contre les cyber-attaques des SI tertiaires et industriels (ces derniers étant assujettis aux évolutions lentes et par paliers en matière de sécurité de leurs constructeurs respectifs). L'objectif est de sensibiliser sur les limites des plans de continuité des SI tertiaires (censés accompagner les organisations dans leur résilience) tout comme des mécanismes de redondance et de sûreté des SI industriels.

**Keywords:** Cyber-menaces, cyber-attaques, vecteur d'attaques, exfiltration, mesures préventives, mesures réactives, palliatifs, résilience, SI tertiaires, SI industriels, SCADA

## 1 Introduction

La continuité d'activité est souvent présentée comme un des éléments de la stratégie de résilience des organisations. Ainsi, face à des sinistres d'ampleur entrainant l'indisponibilité de ressources informatiques, d'infrastructures de communication, d'immeubles voire de collaborateurs, les organisations se sont dotées de plans de continuité d'activité (PCA) de manière à assurer leur survie.

Or les cyber-attaques, dans leur forme moderne, n'ont pas été prises en compte lors de l'élaboration des PCA. Ces derniers focalisés sur un enjeu de disponibilité, n'appréhendent pas la problématique de perte de confiance dans le SI induite par les cyber-attaques. Aussi les dispositifs de continuité du SI, le plus souvent liés aux ressources qu'ils protègent, sont affectés par ces attaques.

Pourquoi ces dispositifs sont-ils en tout ou partie inadaptés ? Quelles sont les actions de renforcement de sa résilience à mener face aux cyber-menaces ? Comment développer sa cyber-résilience, notamment sous l'angle de la continuité d'activité.

## 2 Comprendre les cyber-attaques

Nous avons cartographié sept grandes familles d'impact rencontrées lors de cyber-attaques d'envergure et identifiés trois finalités :

- la recherche de gains (financiers, technologiques, espionnage) à travers l'exfiltration de données confidentielles, la conduite d'opérations frauduleuses appuyées par des manipulations de paramétrages d'applications métiers (plafonds de cartes bancaires) ou la demande de rançons. Les exemples sont légions :
  - secteur de la grande distribution américaine en 2014 avec Target en principale victime [1]
  - secteur bancaire avec Carbanak [2] et les opérations sur les banques Muscat [3], Rakbank [4], JP Morgan Chase [5], Citigroup [6], Fidelity Investments [7], Royal Bank Of Scotland, etc
  - secteur des télécommunications avec Belgacom [8], Orange [9] et AT&T [10]
  - secteur de la santé et des assurances avec Anthem [11], Premera [12], Carefirst [13], etc
  - de manière plus diffuse, campagnes cryptolockers du premier trimestre 2015
- la destruction des données, des postes de travail ou des serveurs de production informatique. Les principaux cas sont attribués le plus souvent à des états avec motivation d'ordre géopolitique ou à des groupuscules hacktivistes : Saudi Aramco [14] et RasGas [15][16][17], médias et établissements bancaires en Corée du Sud, Sony [18], #OpFrance, #OpIsrael, etc
- l'atteinte délibérée à l'image de marque, notamment à travers la publication des données confidentielles de l'organisation ou la diffusion de messages sur les sites web ou les comptes de réseaux sociaux. Les cas de Target, TV5 monde [19] ou encore Sony, en sont une parfaite illustration. L'utilisation des canaux de communication officiels ou la diffusion au grand public de données clients ou confidentielles marquent fortement l'opinion

La cinématique des cyber-attaques est globalement articulée autour de 5 phases : (I) reconnaissance préalable, (II) intrusion, (III) propagation & élévation de privilèges, (IV) persistance, (V) réalisation de l'objectif final.

Dans tous les secteurs touchés, les cyber-attaquants n'ont négligé aucune des portes d'entrée légitimes du SI de l'organisation ciblée (collaborateurs, partenaires, tiers-mainteneurs, sites web, postes de travail en agence) en procédant à une reconnaissance scrupuleuse des informations accessibles librement sur internet (réseaux sociaux, identification de collaborateurs pertinents, revue automatisée de vulnérabilités des services exposés) [phase I].

Sur l'observation de plus d'une vingtaine de cyber-attaques (menée début 2015), on estime que :

- 70% d'entre elles ont exploité une technique d'intrusion de type de phishing accompagnée du déploiement d'un malware en ciblant directement des collaborateurs de l'organisation

- 20% ont exploitées des vulnérabilités de services accessibles depuis internet
- Le reste concerne des intrusions via les clients finaux, des SI de partenaires ou de prestataires/fournisseurs (ex. Rakbank, Bank of Muscat, Target) et dans une moindre mesure via des moyens physiques déployés dans sur les sites de l'organisation ciblée (ex. boitier KVM à connexion 3G pour Barclays Bank, média physique USB pour Saudi Aramco)

Les cyber-attaques révélées sur le premier trimestre 2015 [20] font infléchir ces chiffres au profit de la catégorie du social engineering et notamment du spear-phishing.

L'intrusion initiale [phase II] ne réclame souvent que quelques heures à quelques jours : campagne de spear-phishing avec différentes techniques de contournement des fonctions de sécurité (detection evasion) : pièce jointe comportant un malware inconnu, URL d'un site malveillant portant des codes malveillants, URL d'un site « mimant » un intranet de l'organisation (saisie d'identifiants), URL d'un site légitime compromis renvoyant vers un site malveillant (technique du drive-by download), etc.

Une fois l'intrusion perpétrée, les attaquants cherchent à étendre leur emprise sur le SI. Ils ont, d'un point de vue tactique deux cibles privilégiées : les comptes d'administration techniques (root, dbadmin, administrateurs AD, etc.) ou fonctionnels (gestionnaire des applications métiers) et les services d'infrastructure (exploitation de vulnérabilités et comptes de service).

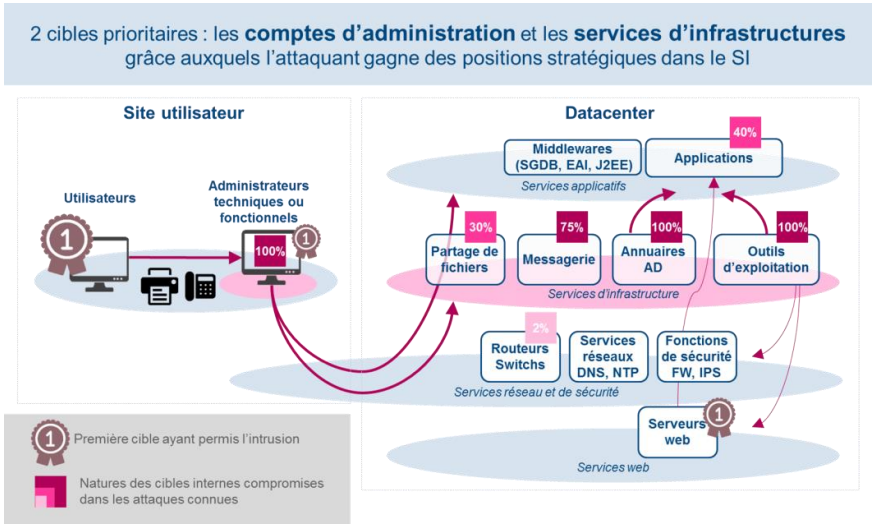



**Fig. 1.** Cibles tactiques visées par les attaquants dès l'intrusion

Du ou des premiers postes compromis, les attaquants consultent toutes les informations pertinentes accessibles dans l'organisation (intranets, annuaires, numéros de support par exemple pour réinitialiser les mots de passe) et contaminent de nouveaux postes (spear-phishing interne, utilisation d'exploit sur le réseau local moins filtré,

etc). Durant cette phase de propagation et d'élévation de privilèges [phase III], qui peut s'étaler de plusieurs jours à plusieurs mois les attaquants procèdent le plus souvent à l'installation d'un « camp de base » où leurs outils d'exploration, de collecte et d'exfiltration seront concentrés.

La phase IV de persistance vise pour les attaquants à assoir leur emprise sur le SI en déployant des portes dérobées complémentaires dans l'hypothèse de la perte ou de la détection de l'une d'entre elles. Cette phase est souvent concomitante à l'élévation de privilèges mais n'est pas systématique.

En effet, pour une grande part des attaques observées, la phase V de réalisation de l'objectif est rapidement menée, à titre d'exemples :

- Saudi Aramco : effacement de plus de 50.000 PC de la compagnie pétrolière suite au déploiement d'un « wiper »
- Rakbank & Bank of Muscat : retrait coordonné de 45 millions de dollars américains en 10h sur les distributeurs automatiques de 21 pays avec de fausses cartes prépayées sans limite de retrait grâce à une modification des plafonds dans le backoffice bancaire sous-traité en Inde
- Target & Home Depot: vol de 40 millions de numéros de CB pour le premier et de 56 millions pour le second. Target a également reconnu être victime du vol de 70 millions de comptes clients (noms, coordonnées, emails)
- L'attaque connue par Sony Pictures est un cas d'école car elle conjugue l'ensemble des finalités habituellement recherchées : exfiltration de plus 100 To (5 films inédits, mails, contrats, données personnelles, salaires, etc.) entre septembre et octobre 2014, demande de rançon le 21/11/14, destruction des postes de travail et des serveurs le 24/11/14, enfin chantage médiatique début décembre

En 2014 le temps moyens avant détection atteignait 205 jours, contre 229 en 2013 et 243 en 2012. La notification de l'attaque provient dans 69% des cas par une source externe et donc 31% seulement par les victimes elles-mêmes [21].

Ces délais s'expliquent à la fois par le manque de maturité des organisations en matière de prévention et de détection du risque cyber. La plupart ne dispose pas de la combinaison de CERT (computer emergency response team) et de SOC (security operation center).

Ces dispositifs, quand ils existent, ne disposent pas encore d'une vision 360° des SI de à protéger. Ils sont en outre insuffisamment ancrés dans les productions informatiques des DSI. Par ailleurs, les missions des CERT/SOC et des productions sont parfois contradictoires. Par exemple, lorsqu'un épiphénomène d'indisponibilité survient, les productions informatiques visent en priorité à le lever, quitte à réinstaller l'équipement défaillant. Or cette défaillance peut-être la conséquence d'une action d'un cyber-attaquant ; l'équipement aurait ainsi mérité d'être préservé et analysé.

Enfin, les organisations disposent rarement de compétences forensics et d'outillage pour analyser techniquement les compromissions ni de méthodologie de gestion de crise cyber. Cette situation freine indubitablement l'interprétation de l'incident et la compréhension des objectifs de l'attaquant.

Ces délais sont aussi le fruit de l'ingéniosité des attaquants qui développent et/ou acquièrent sur le marché noir d'internet (darknet) des kits d'intrusion (exploit kits)

furtifs, indétectables des systèmes antiviraux du marché, en mesure d'affecter tous les navigateurs web du marché. Plusieurs études d'éditeurs antivirus mettent en lumière les capacités industrielles « d'obfuscation » des malwares : l1 malware peut donner lieu de manière automatisée à 10.000 variantes. Enfin dans les cyber-attaques les plus complexes, les attaquants utilisent des malwares exploitant des failles applicatives inconnues des éditeurs (zero-days). En 2014 et selon Symantec, 24 zero-days ont ainsi été exploités dans le cadre d'attaques, chacun disposant d'une durée de vie moyenne de 60 jours avant la parution d'un correctif [22].

## 3 De la défaillance des dispositifs de continuité

Comme évoqué en introduction, les dispositifs de continuité constituent souvent un des éléments de la stratégie de résilience d'une organisation.

Or, depuis plus d'une décennie, à la fois pour répondre aux exigences de reprise rapide des métiers (recovery time objective) et au besoin d'une meilleure exploitabilité, les dispositifs de continuité (repli utilisateurs ou secours informatiques) ont adopté les principes suivants :

- Infrastructures (réseaux et de services techniques) partagées avec celles de la production qu'elles secourent (réseau virtuels étendus, annuaires d'authentification et d'habilitation distribués, services DNS, NTP partagés, etc.)
- Recours à des mécanismes de secours « à chaud » pour les applications (redondance active d'équipements, partage de charge de serveurs, réplication de baies à baies)
- Exploitation commune de la production et de son secours (mêmes équipes, mêmes comptes d'administration, mêmes outils d'exploitation et de supervision)

De fait, cette « proximité » entre le SI nominal et son secours rend vulnérables les dispositifs de continuité aux cyber-attaques.

À titre d'exemple, en cas d'attaque destructive, les postes de secours dédiés et connectés des sites de repli sont exposés à la même contamination (et destruction) que les postes nominaux.

Les pratiques de plan de reprise/secours à froid concernent désormais de moins en moins d'applications, et souvent il s'agit d'applications secondaires.

Enfin, les sauvegardes dites de « recours », établies sur une base souvent quotidienne, constituent pour la plupart des organisations le dispositif de dernier recours pour reconstruire le SI. Malheureusement, du fait de l'antériorité de l'intrusion (plus de 200 jours avant sa détection), ces sauvegardes embarquent de fait les éléments de compromission : malwares, camps de base, mais aussi les modifications déjà opérées par les attaquants. Ainsi leur utilisation réclamerait un assainissement préalable défini à travers une compréhension complète de la démarche des attaquants.

## 4 Cas des SI Industriels

Les cas d'attaques avérées et ayant réussi sur des systèmes industriels se multiplient mais restent à ce jour encore d'un nombre assez limité. Ils ont néanmoins mis en évidence la faiblesse des systèmes industriels aux attaques modernes. Du bien connu Stuxnet au plus récent ayant affecté une mystérieuse aciérie Allemande [23], la destruction de l'outil industriel n'a pu être évitée. Pourtant, le monde industriel s'est attelé depuis de nombreuses années à concevoir des systèmes fiables et dont la continuité est une des composantes majeures, fondamentale et aujourd'hui bien ancrée dans leurs ADN. Mais ces efforts demeurent inadaptés face aux cyber-attaques modernes.

Les architectures ont su intégrer les mécanismes de redondance pour diminuer au mieux les interruptions des procédés industriels mis en œuvre. Au sein d'un système industriel cohabitent généralement deux types de réseau : le nominal et le secours. Très souvent déployés selon des architectures en anneaux ou boucles, ces réseaux assurent une disponibilité et une parade efficace contre les défaillances. Une bascule de l'un vers l'autre s'opère dès lors qu'un problème de liaison apparait. Pour renforcer encore plus l'objectif de disponibilité visé, certains réseaux disposent de caractéristique d'« auto-cicatrisation » qui leur confère une capacité à reconfigurer leur topologie de manière à prendre en considération des portions « éteintes » du réseau sans impacter le processus industriel [24].

On observe donc là une première similitude avec les mécanismes de continuité mis en œuvre sur des SI tertiaire : proximité du nominal et du secours, redondance active et secours à chaud. Quand bien même ces réseaux savent faire face à des défaillances et autres pannes, ils ne sont en aucun cas résilient contre des cyber-attaques.

Particularité des systèmes industriels, les mécanismes dits de « sûreté » peuvent eux aussi s'avérer inefficace en cas d'attaques. Parmi ces mécanismes, on peut citer les Systèmes Instrumentés de Sécurité (SIS). Ils sont censés protéger les installations et limiter les impacts en cas de défaillance importante. Il s'agit d'une barrière primordiale, souvent citée dans les études de dangers comme un moyen important de réduction du risque. Ils ont vocation à activer des mécanismes de repli permettant de repositionner une installation dans un état stable (dérive de pression, température, vitesse…) et sous contrôle. Dans certains cas, il s'agit tout simplement de l'unique barrière. Dès lors on comprend très vite que la confiance que l'on accorde à la sûreté d'une installation repose essentiellement sur ces systèmes. D'ailleurs, les autorisations d'exploitation de certaines installations sont en partie délivrées au regard de la présence ou non de ce type de protection. C'est le cas très souvent pour des sites SEVESO.

Or, cette confiance accordée est mise à mal par les évolutions récentes dont ont fait l'objet ces systèmes. On observe une forte tendance à la mutualisation des ressources dédiées à la conduite pure d'un procédé industriel avec les ressources visant à garantir sa sûreté. Cette mutualisation plus ou moins étendue selon les cas peut concerner :

- Les capteurs : vérification à la fois des dérives acceptables dans la conduite d'un procédé et mesure des dérives aux limites de fonctionnement sensées enclencher la mise en repli des installations
- Les solveurs logiques de sûreté embarqués dans les SIS : cohabitation sur un même fond de panier d'un SIS et d'un contrôleur de procédé voire dans certain cas, fusion pure et simple du contrôleur et du SIS
- Le réseau : « flux de conduite » et « flux SIS » véhiculés sur une même liaison physique et logique
- Les applications de contrôle et de gestion des procédés et de la sûreté : les fonctions de gestion sont embarqués dans un même applicatif très souvent accessible sur une même et unique station

Et pour les systèmes qui reposent sur des technologies très souvent issues du monde de l'IT tertiaire (serveur de domaine AD par exemple), parfois mutualisées entre fonctions de production et de sûreté, une compromission de ces derniers entraine de facto la compromission du système de sécurité.

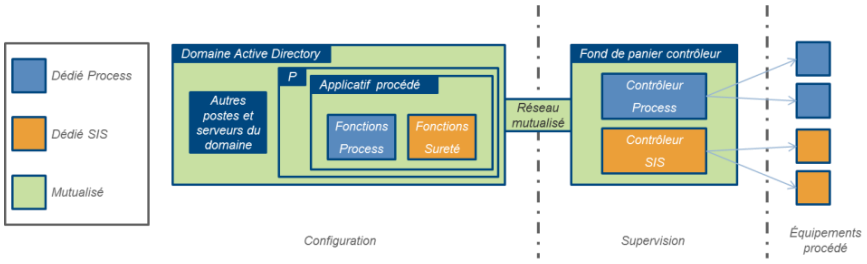



**Fig. 2.** Principe d'architecture d'un contrôleur et de son SIS

La situation actuelle prend entre autre son origine dans le fait que pour pouvoir augmenter d'avantage les rythmes de production, obtenir de meilleur rendement et optimiser ses installations, les fournisseurs et constructeurs de solutions industrielles ont massivement adoptés des technologies issues de l'IT conventionnel. Or cela s'est opéré sans prendre en considération également les problèmes que cela engendre. Et avec, ce sont des principes fondamentaux de la cybersécurité qui ont été laissés de côté: sécurité des architectures réseaux, principe de segmentation et de cloisonnement entre autres.

Pour des raisons de verrouillage de marché et de captation client, ils ont développé une multitude de solutions propriétaires pas toujours interopérables. La cybersécurité n'a pas toujours été non plus l'une de leur première préoccupation. L'une des conséquences de ces arbitrages commence d'ailleurs à être de plus en plus perçue : le manque de compétences combinées en matière IT (Information Technology), d'OT (Operationnal Technology) et de cybersécurité touche le secteur industriel.

Corriger ces erreurs prendra un certain temps au rythme des modernisations et nouvelles constructions qui dans un monde industriel peuvent s'avérer extrêmement lentes. Temps durant lequel, la menace toujours grandissante consolidera d'avantage

son niveau d'ingéniosité et lui permettra de s'en prendre d'avantage aux installations industrielles.

## 5 Renforcement de la cyberrésilience

Afin de renforcer la cyberrésilience des organisations nous avons identifié plus de 70 mesures de sécurités réparties en 8 thématiques :

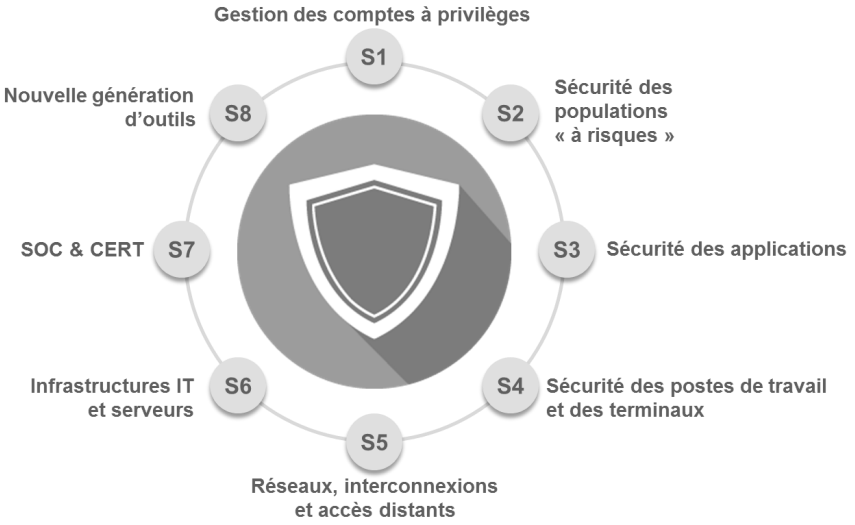



**Fig. 3.** Les huit thématiques de mesures de cyberrésilience

Ces mesures s'appliquent également sur toutes les composantes du SI Industriels (réseaux de terrain, systèmes d'historisation, SCADA, automates…) en prenant en compte leur particularité.

Nous proposons ci-dessous quelques gros plans sur quelques une d'entre elles.

### 5.1 Identifier ses « zones rouges »

Le principe d'une « zone rouge » est de définir un espace dans lequel un processus métier jugé vital doit impérativement être préservé, que ce soit dans sa continuité de fonctionnement, sa confidentialité ou son intégrité. Cette zone devra pouvoir être « isolée » ou même « arrêtée » soit pour en assurer le fonctionnement soit pour en préserver l'intégrité ou la confidentialité.

Cette zone comprend des éléments techniques, des parties prenantes internes et/ou externes. Sa constitution réclame la compréhension du processus métier, de ses modes de fonctionnement dégradés et de ses palliatifs.

Le schéma qui suit montre qu'en cas de situation de compromission, seuls les éléments maintenus dans la zone rouge communiquent ; tous les équipements de sécurité (pare-feu) ou réseau sont mis à contribution pour opérer cette isolation (stratégie d'isolation à définir). Pour les éventuels autres éléments requis au fonctionnement du processus, des modes palliatifs sont déployés.

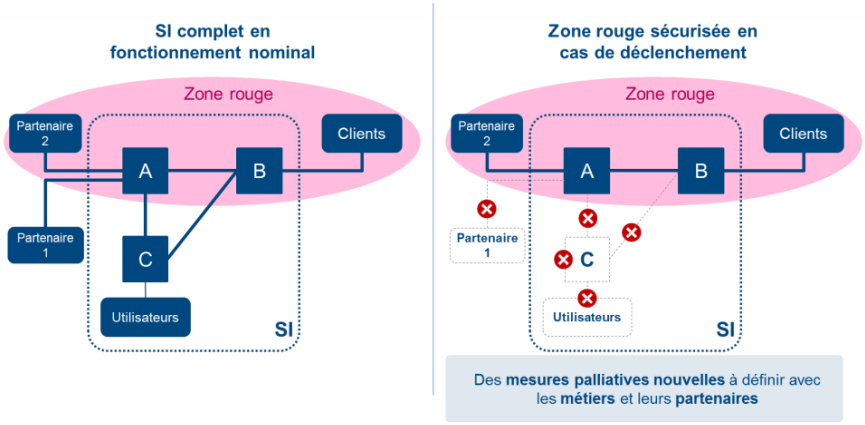



**Fig. 4.** Principe des zones rouges

Le principe de zone rouge trouve ces sources dans les modèles de sécurité de type « floodgate ».

Ce principe peut s'appliquer dans une certaine mesure aux systèmes industriels. En première approche on pourra considérer le système industriel comme étant une « zone rouge » à lui tout seul vis-à-vis du SI bureautique de l'entreprise. Il faudra donc anticiper des processus et workflow adaptés à un isolement de l'outil de production avec les outils de type ERP ou de pilotage centralisé de la production de manière à conserver un fonctionnement acceptable.

Définir des zones rouges au sein même du système industriel peut s'avérer en revanche plus complexe. Cela dépendra bien évidement du type de procédé mis en œuvre. Pour un procédé dit « continu » il est généralement difficile de poursuivre uniquement une partie du processus. Pour les procédés plus « séquencés », on pourra organiser la zone rouge selon que le système représente un risque critique ou non pour l'entreprise : préservation d'un outil innovant (secret industriel) ou dont le coût en cas de destruction pourrait être de nature à lourdement impacter l'entreprise.

### 5.2 Bâtir des « chaînes de contrôle d'intégrité fonctionnelles »

Renforcer sa capacité de détection, c'est également être en mesure de compléter l'arsenal des outils techniques de sécurité (centrés sur les malwares) pour identifier des comportements suspects. Ces comportements atypiques relèvent de pratiques non conformes aux processus des organisations. C'est donc au cœur des processus les plus critiques que la détection doit être renforcée.

La construction d'une chaîne de contrôle d'intégrité fonctionnelle a pour objectif d'opérer une réconciliation de l'enchaînement de plusieurs actes menés dans un processus pour témoigner de son intégrité. Dès lors qu'un des contrôles fonctionnels attendus n'est pas positif, le traitement opéré est suspecté être illégitime.

Dans deux cas rencontrés chez nos clients, c'est ce type de disposition qui a permis de détecter la création de comptes à privilèges ou de modification de paramètre métier d'application critique.

Ce principe dérive des modèles dits de « multi-level controls ». À titre d'exemple, les métiers de la banque d'investissement conduisent ce type de contrôles en séparant leurs fonctions de front, middle et back office. Il s'agit d'automatiser ces mécanismes de contrôles et de les effectuer sur des systèmes disjoints pour en accroître la fréquence et la profondeur.

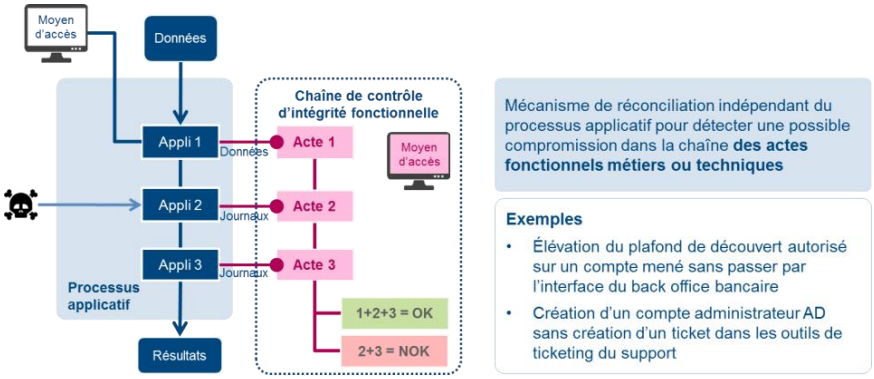



**Fig. 5.** Principe de chaîne d'intégrité fonctionnelle

Cette mesure qui peut paraitre nouvelle en environnement IT Tertiaire est finalement déjà très présente en environnement industriel. Il est aisé de considérer cette chaîne d'intégrité fonctionnelle comme un mécanisme de « sûreté » en tant que tel. Le fonctionnement reste relativement semblable aux tests qu'effectuerait un automate de sureté selon une matrice de sûreté et des conditions bien définies. Dans un cas, le résultat est de nature à remonter une alerte, dans l'autre, il a vocation à déclencher les mécanismes de mise en sécurité d'une installation.

L'efficacité d'une telle mesure réside aussi dans l'indépendance dont doit faire preuve la chaine de contrôle d'intégrité. Et nous l'avons vu précédemment, cette indépendance dans le domaine industriel est quelque peu mise à mal.

### 5.3 Construire des « chaînes applicatives alternatives »

Dans les contextes à très forts enjeux de disponibilité, le fonctionnement d'un processus ne peut souffrir une interruption, à titre d'exemple les infrastructures financières à caractère systémique (SIPS). L'ensemble de la « chaîne applicative », c'est-à-

dire l'écosystème applicatif qui supporte ce processus doit impérativement rendre le service.

Le principe d'une « chaine applicative alternative » prend ses sources dans les modèles dits de « non similar facilities ». L'ensemble de l'écosystème applicatif nominal est remplacé par une chaîne alternative délivrant les mêmes fonctionnalités mais reposant sur des technologies (OS différents), des codes (redéveloppement complet par des équipes différentes), une exploitation (outillage et équipe différentes) et des modes d'accès différents.

Dans certains cas, la chaîne alternative peut fonctionner concomitamment à la chaine nominale pour détecter des altérations dans les traitements. Ce type de solution est extrêmement coûteux et ne convient qu'à des ensembles applicatifs peu complexes.

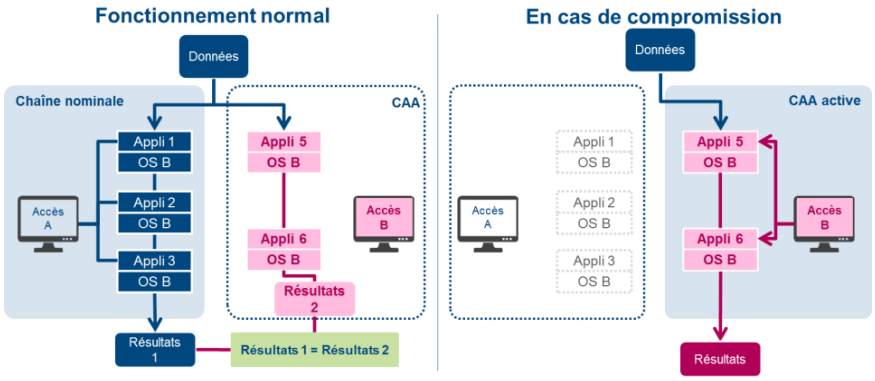



**Fig. 6.** Principe de chaîne applicative alternative

La mise en œuvre de ce concept en environnement industriel peut paraitre compliquée. L'une des pistes naturelles à envisager serait de l'appliquer aux applications de type SCADA. Toutefois l'un des freins à sa mise en œuvre est le caractère assez fermé de certaines solutions propriétaires. Recourir à des applicatifs alternatifs nécessiterait dans certains cas un coût important en développement. Mais cela pourrait être aussi une opportunité intéressante pour viser d'avantage d'interopérabilité et de compatibilité entre les solutions du marché.

Étendre ce principe aux couches inférieures, celles qui intègrent notamment les automates industriels peut sembler être de l'ordre de l'impossible au regard des contraintes de cout mais aussi de dimensionnement des installations (vu le nombre d'équipement qui peut être déployé sur certains systèmes). C'est pourquoi il pourrait être envisageable que sur des systèmes très critiques à dimension raisonnable.

Il convient toutefois de relativiser l'efficacité d'une telle approche. Si recourir à une deuxième chaine d'automatisme alternative peut donner le sentiment de renforcer la sécurité de l'installation en cas d'attaque, il ne faut pas oublier que sur ce domaine, les protocoles couramment utilisés sont nativement vulnérables, ModBus notamment. Et les protocoles industriels propriétaires ne sont pas non plus connus pour mettre en

œuvre les principes de bases en matière de sécurité (authentifications, signature, chiffrement…).

### 5.4 Faire évoluer son organisation de surveillance, détection et réaction

Nous l'avons vu, la détection d'une attaque avancée intervient très tardivement et dans les 2/3 des cas par des acteurs externes à l'organisation touchée.

Il devient urgent de réviser l'organisation et l'outillage de surveillance et de détection. Cela passe par le déploiement de dispositifs CERT, SOC, la constitution, la centralisation et le traitement de journaux sur tous les systèmes sensibles (métiers et IT) et le renforcement de tous les fondamentaux en matière de sécurité.

L'analyse technique seule ne suffit pas. La contribution des métiers ou, tout du moins, d'un savoir-faire métier devient cruciale dans l'interprétation des actions suspectées illégitimes. Nous préconisons la mise en œuvre d'une cellule à forte composante métier : CBAT (Cybersecurity Business Analyst Teams).

Elle a vocation à être en appui des cellules SOC, ayant une connaissance avancée des infrastructures et CERT, disposant des connaissances avancées sur la menace. Ses principaux objectifs sont de recenser les ressources et les données critiques des métiers, de définir les règles de surveillance et de contrôle intégrité fonctionnelles, d'identifier les « zones rouges » à préserver à travers un BIA évolué et de fournir l'appui nécessaire à l'interprétation des alertes et des erreurs de réconciliation du point de vue métier.

Sur les systèmes industriels, la surveillance et la détection deviennent un enjeu majeur, surtout lorsque l'on sait qu'une attaque peut engendrer des dommages irréversibles. Il convient donc de détecter le plus rapidement possible les opérations malveillantes en cours. La cellule CBAT prend dès lors tout son sens dans un contexte où monde bureautique et industriel ont encore du mal à interagir. La compréhension fine du procédé industriel est donc un pré-requis indispensable pour établir les scénarios de supervision qui permettront d'identifier de potentiels actes malveillant.

Toutefois, il peut s'avérer illusoire de considérer la supervision comme barrière ultime pour un système industriel. La finalité étant de pouvoir réagir aux attaques en cours et rétablir au plus vite le service lorsqu'il est rendu indisponible. Comment faire lorsque l'attaque a eu pour conséquence la destruction de l'installation comme ce fut le cas pour cette aciérie Allemande ?

Dans ce cas, seul une protection sans faille constituera cette barrière ce qui au premier abord semble relever de l'utopie. Certaines approches peuvent être envisagées :

- Prise en compte de la malveillance dans la sûreté de fonctionnement : les études très poussées en sûreté de fonctionnement notamment de certains codes doivent trouver leur pendant au regard de la problématique de cybersécurité. Réservée à des applications militaires, l'analyse formelle des codes, notamment effectuées dans certaines évaluations Critères Communs, est une piste d'étude.
- Recours à des solutions non informatisées pour certains systèmes très critiques : la logique câblée pour les systèmes de sûreté reste à ce jour un des meilleurs moyens

pour se prémunir d'actes malveillant ayant pour origine une action « informatique ».

- Généralisation des approches dites « déterministes » : pour beaucoup d'entre eux, les systèmes industriels sont des systèmes à états finis. Toute dérive doit dès lors pouvoir être rejetée. Des solutions du monde de l'IT embrassent depuis quelque temps ce concept (mécanisme de « whitelisting » par exemple).

### 5.5 Ajuster son dispositif de gestion de crise

Parce qu'il est primordiale de s'exercer en amont afin d'être prêt le jour où il faut faire face à la crise et anticiper certaines réponses, il convient d'ajuster les dispositifs existant en intégrant les particularités relatives aux scénarios d'attaques cyber. Il s'agira entre autre d'identifier les « crises au long cours » et fortement mobilisatrice des métiers.

Afin d'être certains que l'ensemble des personnes et des équipes utiles lors d'une crise cyber soit rapidement mobilisable, des astreintes doivent être définies. Il convient de mobiliser les équipes des ressources humaines et juridiques pour les contractualiser. Des schémas de rotation des équipes doivent également être pensés.

Lors de crise cyber, les opérationnels SI sont généralement les seuls à disposer de la connaissance pour réaliser les actions techniques sur le SI. Pour autant, durant la crise, ils reçoivent des ordres qui peuvent générer un important stress, contradictoires ou redondants, non priorisés et par différentes lignes managériales. Toutes les demandes doivent être centralisés et coordonnées par un unique organe. L'objectif est d'éviter le phénomène de pyramide inversée et mettre en place une organisation miroir à celle des attaquants.

Acteur centrale de la crise cyber, la DSI ne doit pas être sur-mobilisée sur l'investigation et la défense au détriment de la production et du secours. Cet aspect constitue un point d'anticipation important à ne pas négliger. Cela peut également éveiller les soupçons au sein de l'entreprise, alors qu'en de telles circonstances il peut être plus judicieux de préserver le secret sur la crise en cours. Les attaques devenant de plus en plus complexes, il est essentiel de se doter ou d'avoir accès à des compétences en matière d'investigation numérique (forensics). Il s'agira aussi de bien organiser les interventions en formant les équipes à des méthodologies de pilotage des investigations et de construction de plan de défense. A ce titre le « Diamond Model : Graphe activity-attack » et le « Kill Chain Model » peuvent respectivement répondre à ces besoins.

Enfin, les exemples récents l'ont montrés, une crise ne peut plus se vivre en autarcie. La mise en place de leviers externes pour en faciliter sa gestion devient un véritable facteur clé de succès :

- Les clients et les fournisseurs : prévoir des clauses permettant de pouvoir assurer une coupure de service sans pénalité durant une attaque avec pour objectif de ne pas être un vecteur de propagation vis-à-vis d'eux

- Les autorités : cartographier ou construire le réseau des acteurs à mobiliser en face de chaque autorité pour rendre plus efficace leur mobilisation et faciliter les réponses à leur apporter.
- Les avocats : anticiper le traitement judiciaire d'une cyber-attaque en disposant d'un ou plusieurs cabinets d'avocats spécialisés à même de traiter le sujet

D'autres acteurs tels que les « pairs », syndicats ou fédérations de son secteur peuvent être sollicités : partages d'expériences, alertes, faire adhérer aux nouvelles orientations les collaborateurs directement impliqués (cas des astreintes notamment).

## 6 Conclusion

Les évolutions proposées dans cet article, souvent majeures, doivent s'inscrire dans une revue des stratégies de secours existantes afin d'évaluer leur vulnérabilité et l'intérêt de déployer des nouvelles solutions de cyberrésilience, en particulier sur les systèmes les plus critiques. L'évolution des Business Impact Analysis (BIA) pour inclure cette dimension est certainement une première étape clé.

Implémenter ces nouvelles mesures de cyberrésilience nécessite des efforts importants. Des efforts qui seront vains notamment si les solutions de secours et les systèmes nominaux ne sont pas déjà sécurisés correctement et surveillés avec attention.

Le RSSI reste un acteur clé pour faire aboutir ces démarches parfois entamées mais rarement finalisées. L'aide du Responsable du Plan de Continuité d'Activité (RPCA) sera alors un plus indéniable! Il est aujourd'hui impossible de sécuriser des systèmes à 100%, il faut donc accepter la probabilité d'occurrence d'une attaque et c'est à ce moment-là que le RPCA prendra tout son rôle.

Pour les responsables sécurité et sûreté des installations industrielles, la tâche s'annonce plus complexe. Les évolutions devront se faire au grès des modifications apportées par les constructeurs et fournisseurs. À noter que des efforts sont constatés notamment au regard d'une règlementation qui se veut de plus en plus contraignante. Mener des premiers exercices de crise à portée cyber sur des systèmes industriels pourra constituer une première action mobilisatrice.

La cyberresilience des systèmes d'information, qu'ils soient « Tertiaire » ou « Industriel », ne pourra être atteinte que si les sphères sécurité, continuité et sûreté travaillent main dans la main.

## 7 Références


1. *A « Kill Chain » Analysis of the 2013 Target Data Breach*. Majority staff report for chairman Rockefeller. 26, Mars 2014, Committee on Commerce, Science, and Transportation [En ligne] http://www.commerce.senate.gov/public/_cache/files/24d3c229-4f2f-405d-b8db-a3a67f183883/23E30AA955B5C00FE57CFD709621592C.2014-0325-target-kill-chain-analysis.pdf
2. *The Great Bank Robbery: the Carbanak APT*. Février 2015, KasperskyLab. [En ligne] https://securelist.com/files/2015/02/Carbanak_APT_eng.pdf

3. *Bank Muscat hit by $39m ATM cash-out heist*. Mars, 2013. [En ligne] http://www.theregister.co.uk/2013/03/01/bank_muscat_atm_mega_fraud/
4. *Banks lose millions to hackers in ATM card breach*. 10, Mai 2013, Gulf News Banking. [En ligne] http://gulfnews.com/business/sectors/banking/banks-lose-millions-to-hackers-in-atm-card-breach-1.1181774
5. *2014 JPMorgan Chase data breach*. Wikipedia. [En ligne] https://en.wikipedia.org/wiki/2014_JPMorgan_Chase_data_breach
6. *Citi Credit Card Hack Bigger Than Originally Disclosed*. Juin 2011. Wired. [En ligne] http://www.wired.com/2011/06/citibank-hacked/
7. *Fidelity hack points to JPMorgan link*. Octobre 2014, CNBC. [En ligne] http://www.cnbc.com/2014/10/09/fidelity-hack-points-to-jpmorgan-link.html
8. *The Inside Story of How British Spies Hacked Belgium's Largest Telco*. Décembre 2014, The Intercept. [En ligne] https://theintercept.com/2014/12/13/belgacom-hack-gchq-inside-story/
9. *Nouveau vol massif de données personnelles chez Orange*. 6 Mai 2014, Le Monde. [En ligne] http://www.lemonde.fr/technologies/article/2014/05/06/vol-de-donnees-chez-orange-1-3-million-de-clients-et-de-prospects-touches_4412570_651865.html
10. AT&T Data breach notification. 19 Mai 2014, Office of the Attorney General. [En ligne] http://oag.ca.gov/system/files/CA%20Customer%20Notice.doc__1.PDF?
11. *The Anthem Hack: All Roads Lead to China*. Threatconnect Intelligence Research Team (TCIRT). 27 Février 2015, ThreatConnect. [En ligne] http://www.threatconnect.com/the-anthem-hack-all-roads-lead-to-china/
12. *What you need to know about the Premera data breach*. Christopher Budd. 17 Mars 2015, TrendMicro. [En ligne] http://blog.trendmicro.com/premera-databreach/
13. *CareFirst BlueCross BlueShield has been the target of a cyberattack*. Mai 2015, CareFirst. [En ligne] http://www.carefirstanswers.com/
14. *Hack on Saudi Aramco hit 30,000 workstations, oil firm admits*. John Leyden. 29 Aout 2012, The Register. [En ligne] http://www.theregister.co.uk/2012/08/29/saudi_aramco_malware_attack_analysis/
15. *The cyber-attacks on Saudi Aramco, RasGas, and U.S. banks in the context of international law*. Dimitar Kostadinov. 26 Décembre 2012, InfosecInstitute. [En ligne] http://resources.infosecinstitute.com/cyber-attacks-in-the-context-of-international-law/
16. *The Shamoon Attacks*. Symantec Official Blog. 16 Aout 2012. [En ligne] http://www.symantec.com/connect/blogs/shamoon-attacks
17. *Shamoon the Wiper – Copycats at Work*. Kaspersky Lab. 16 Aout 2012. [En ligne] https://securelist.com/blog/incidents/57854/shamoon-the-wiper-copycats-at-work/
18. *Piratage de Sony Pictures Entertainment*. Wikipedia. [En ligne] https://fr.wikipedia.org/wiki/Piratage_de_Sony_Pictures_Entertainment
19. *Piratage de TV5Monde: «L'attaque montre la détermination des hackers à faire un maximum de bruit»*. 9 Avril 2015, 20 Minutes. [En ligne] http://www.20minutes.fr/societe/1582647-20150409-piratage-tv5monde-attaque-montre-determination-hackers-faire-maximum-bruit
20. *IT threat evolution in Q1 2015*. SecureList. 6 Mai 2015, Kaspersky Lab. [En ligne] https://securelist.com/analysis/quarterly-malware-reports/69872/it-threat-evolution-in-q1-2015/
21. *M-Trends® 2015: A View From The Front Lines*. Mandiant & FireEye. Avril 2015. [En ligne] http://www2.fireeye.com/rs/fireye/images/rpt-m-trends-2015.pdf

22. *Internet Security Threat Report April 2015 Volume 20 - Appendices*. Symantec. Avril 2015. [En ligne] https://www4.symantec.com/mktginfo/whitepaper/ISTR/21347931_GA-internet-security-threat-report-volume-20-2015-appendices.pdf
23. *Bericht zur Lage der IT-Sicherheit in Deutschland 2014*. Bundesamt für Sicherheit in der Informationstechnik. 15 Décembre 2014. [En ligne] https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Lageberichte/Lagebericht2014.pdf
24. *Des "chaînes" remplacent les anneaux redondants*. Frédéric Parisot. Février 2011, Mesures n°832. http://www.mesures.com/pdf/old/832-reseaux-ethernet-redondant.pdf
25. *Inspection des Installations Classées – Contenu d'une étude de dangers*. [En ligne] http://www.installationsclassees.developpement-durable.gouv.fr/Contenu-d-un-etude-de-dangers.html

# Détection des chevaux de Troie matériels pour l'amélioration de la résilience des systèmes numériques

Julien Francq[1], Florentin Demetrescu, Franck Courbon, Philippe Loubet-Moundi, Xuan Thuy Ngo, Jean-Luc Danger, Sylvain Guilley, Éliane Jaulmes, Karim Khalfallah, Victor Lomné, Ingrid Exurville, Bruno Robisson, Jean-Baptiste Rigaud, Papa-Sidy Ba, Manikandan Palanichamy, Sophie Dupuis, Giorgio Di Natale, Marie-Lise Flottes, Bruno Rouzeyre

Consortium du projet Fonds Unique Interministériel (FUI) HOMERE+ coordonné par :
[1]Airbus Defence & Space − CyberSecurity,
1 Bd Jean Moulin, CS 40001, MetaPole, 78996 Elancourt Cedex, France.
julien.francq@airbus.com

**Résumé** Les chevaux de Troie matériels (*Hardware Trojans*, HTs) sont des menaces émergentes pour la résilience des systèmes numériques. Ce sont des modifications frauduleuses d'un circuit intégré qui peuvent être réalisées à n'importe quelle étape de sa fabrication. L'attaquant peut implanter un HT induisant sous certaines conditions de déclenchement un effet malicieux pouvant conduire à des conséquences économiques et sociétales catastrophiques (déni de service, fuite d'informations confidentielles, etc.) suivant le système infecté. Trouver des méthodes de détection de HTs efficaces devient donc une nécessité pour les systèmes à haut niveau de sécurité requis. C'est l'objectif du projet FUI HOMERE+ (2012-2015). Le but de cet article est de présenter le contexte des HTs, et les principaux résultats de ce projet.

**Catégorie.** Spécialisée.
**Mots-clés.** Chevaux de Troie matériels, détection en phase de test et de fonctionnement, méthodes préventives.

## 1 Introduction

De nos jours, de nombreuses étapes du cycle de production d'un Circuit Intégré (CI) sont confiées à des entreprises différentes de celle qui est à la source du développement d'un nouveau produit. Pour des raisons économiques, les étapes de fabrication et de tests de production notamment sont confiées à des fonderies très compétitives souvent situées dans des pays étrangers, sur lesquelles il est difficile d'opérer un contrôle total de leurs activités. De plus, le flot de conception d'un CI est si complexe qu'il est difficile d'assurer une confiance totale dans toutes les étapes de sa fabrication [CNB09].

Dans ce contexte de délocalisation, les fournisseurs d'applications à haut niveau de sécurité sont de plus en plus anxieux lorsqu'ils confient les spécifications de leurs CIs pour fabrication. Il s'avère en effet très difficile d'immuniser les circuits produits contre toutes altérations ou inclusions malicieuses produites lors de la fabrication, aussi appelées chevaux de Troie matériels (*Hardware Trojans*, HTs). En effet, la menace amenée par les HTs, qui était considérée pendant longtemps comme théorique, commence à se matérialiser. Ces HTs, pouvant être insérés par des attaquants aux moyens quasiment illimités tels des organisations criminelles ou encore des agences gouvernementales, peuvent être un vecteur efficace de déstabilisation d'états et d'entreprises [1]. En 2005, la commission scientifique de la Défense américaine a tiré la sonnette d'alarme à propos des conséquences que la délocalisation de la fabrication de CIs pourrait engendrer sur la sécurité de certaines applications à haut niveau de sécurité requis. Cela a conduit le DARPA, l'aile R&D du Pentagone, à lancer le programme "Trust in IC" en 2007. Le but de ce programme était de mettre au point des procédures efficaces de détection et de neutralisation de HTs. Ce programme a connu une suite : le projet IRIS (*Integrity and Reliability in Integrated Circuits*), financé une nouvelle fois par le DARPA entre 2011 et 2014. Depuis peu, les HTs intéressent également de plus en plus la communauté des chercheurs en sécurité des systèmes embarqués : on peut par exemple citer la conférence de référence CHES (*Cryptographic Hardware and Embedded Systems*) 2009, dont le "sujet chaud" (*Hot Topic*) était consacré aux HTs. Tout ceci montre que les HTs sont des menaces émergentes et réelles auxquelles les fournisseurs d'applications à haut niveau de sécurité requis doivent absolument se prémunir.

Voici une liste non exhaustive d'effets connus amenés par l'insertion de HTs dans les CIs : ils peuvent éteindre un composant grâce à un interrupteur accessible à distance (*kill switch*) [Ade08], faire dysfonctionner un CI en altérant ses noeuds internes, faire fuir volontairement un secret (clé de chiffrement) à travers un canal auxiliaire du CI tel le rayonnement électromagnétique, assister une attaque logicielle (*malware*) en fournissant une (*backdoor*) matérielle qui peut engendrer des opérations frauduleuses sur un PC (escalade de privilèges, connexions automatiques à un système, vol de mots de passe), ou encore empêcher le CI de rentrer en mode "économie d'énergie".

En résumé, tous les fondeurs et fournisseurs d'applications à haut niveau de sécurité requis doivent se protéger contre les menaces que constituent les HTs. En ce sens, des méthodes de neutralisation des HTs dans les CIs ont été proposées dans l'état de l'art. Malheureusement, elles ne sont pas pleinement satisfaisantes aujourd'hui. Le projet HOMERE+ (Hardware trOjans : Menaces et robustEsse des ciRcuits intEgrés) est un projet de recherche industrielle ayant pour but de développer de nouvelles méthodes de détection et de neutralisation de HTs.

Cet article se propose de présenter un état de l'art des méthodes de détection de HTs, puis, un résumé des principaux résultats de recherche du projet HOMERE+.

1. C'est pour cela que, même si le risque amené par les HTs est bien avéré, les éléments le prouvant sont classifiés.

## 2 Méthodes de détection de HTs de l'état de l'art

Cette section présente l'état de l'art des HTs. Dans une première partie, nous les classifierons, puis dans une seconde partie, nous décrirons les techniques de détection/neutralisation des HTs actuellement publiées et les verrous technologiques auxquels nous sommes confrontés.

### 2.1 Taxonomie des HTs

Un HT est un circuit ajouté dans un CI qui peut être activé (ou déclenché) par un événement rare. Ainsi, il causera un dysfonctionnement (*trojan payload*) dans le CI. Les HTs peuvent être ainsi classés selon leur(s) condition(s) de déclenchement (*trigger(s)*) et leur *payload*. Un mécanisme de *trigger* peut être numérique ou analogique.

**Triggers** Les chevaux de Troie déclenchés **numériquement** (*Digitally Triggered Trojans*, DTTs) peuvent également être divisés en deux catégories : combinatoires et séquentiels. Pour minimiser la détection des HTs, un DTT combinatoire (respectivement séquentiel) doit seulement induire des dysfonctionnements quand une valeur rare (resp. séquence rare d'événements) est détectée. Un DTT séquentiel (aussi appelé bombe à retardement, ou "*time-bomb*") est typiquement un compteur (a)synchrone sur $k$ bits qui s'active quand le compteur est égal à $2^k - 1$. Compteurs synchrones et asynchrones peuvent aussi être associés pour implémenter un déclenchement hybride. Enfin, des machines à états finis plus complexes peuvent être imaginées pour déclencher un HT séquentiel uniquement lorsque de rares séquences d'événements apparaissent dans le CI. Les chevaux de Troie déclenchés **analogiquement** peuvent être implantés à l'aide de capteurs sur puce basés sur des résistances et des capacités. Ils peuvent aussi utiliser l'activité du CI : par exemple, la chaleur générée par un oscillateur en anneaux utilisant des inverseurs peut à la fois générer de la chaleur et mesurer la température d'une zone particulière d'un CI. Si cette dernière atteint une limite, le HT peut être déclenché.

**Trojan Payloads** Les HTs peuvent également être classés suivant la *payload* qu'ils engendrent. Les *payloads* numériques peuvent affecter des noeuds internes du CI infecté et le contenu des mémoires. Les *payloads* analogiques affectent quant à eux des paramètres du CI infecté (performance, consommation, marge de bruit, *etc.*) par l'activation de résistances et de capacités parasites. La durée de vie d'un CI peut aussi être raccourcie en augmentant anormalement l'activité du CI infecté grâce, par exemple, à l'insertion d'oscillateurs en anneaux utilisant des inverseurs. D'autres *trojan payloads* peuvent également faire fuir volontairement une information secrète (par exemple, une clé de chiffrement) grâce à des canaux auxiliaires (par exemple le rayonnement électromagnétique émis par le CI infecté), ou encore provoquer une attaque par déni de service.

**Bilan** Conformément à ce qui a pu être observé dans cette section, le bestiaire des HTs est très riche, ce qui complique énormément la tâche des fournisseurs d'applications à haut niveau de sécurité requis. En effet, et comme nous le verrons dans la section suivante, une seule stratégie de détection ne sera pas capable de détecter tous les HTs de la littérature : il faudra donc combiner plusieurs techniques afin d'obtenir un taux de détection maximal.

## 2.2 Méthodes de détection de l'état de l'art

Plusieurs techniques de détection et de neutralisation des HTs ont été proposées dans la littérature mais aucune ne paraît pleinement satisfaisante aujourd'hui. Cette section décrit leurs avantages et leurs inconvénients. Deux types de méthodes peuvent être utilisés : des méthodes destructives et non-destructives.

**Méthodes destructives** Les méthodes de détection de HTs destructives consistent essentiellement à effectuer de la rétro-ingénierie inverse (*reverse-engineering*) des CIs testés, c'est-à-dire "remonter" à la structure du CI et ses fonctionnalités à partir d'une inspection nanoscopique, puis à comparer la structure avec un modèle de référence [Kum00]. Malheureusement, ces techniques ont plusieurs défauts. Tout d'abord, quand un CI subit un *reverse-engineering*, il ne peut pas être réutilisé et est donc jeté à la poubelle. Ensuite, c'est un processus très coûteux en termes de temps (la validation d'un seul CI prend des mois) et d'argent (plusieurs centaines de milliers d'euros) nécessaires à l'inspection d'un CI. De plus, cette procédure sera de plus en plus en complexe à réaliser dans les années qui viennent de par loi de Moore, qui indique que la densité d'intégration des CIs sera de plus en plus grande dans le futur. Enfin, puisqu'un attaquant peut insérer des HTs uniquement sur un petit échantillon de CIs que contient un *wafer*, et dans le cas où on ne teste qu'un nombre réduit de CIs par *wafer*, des CIs infectés peuvent passer à travers le crible.

**Méthodes non-destructives** Les méthodes non-destructives peuvent être invasives et non-invasives. Dans le premier cas, du matériel dédié à la détection des HTs doit être inséré dans le CI. Dans le second cas, le CI reste inchangé.

*Méthodes invasives.* Des méthodes invasives non-destructives peuvent être préventives : elles peuvent être utilisées pour rendre l'insertion de HTs plus difficile. Par exemple, l'obfuscation d'un CI [CB09] peut rendre plus difficile son *reverse-engineering*, et donc un attaquant aura plus de difficultés à retrouver l'architecture complète du CI : du coup, l'insertion de HTs aura toutes les chances d'être soit bénigne, soit facilement détectable.

Des méthodes invasives non-destructives peuvent également être utiles en ce sens qu'elles peuvent aider à améliorer l'efficacité d'autres méthodes de détection, telles celles basées sur l'analyse de testabilité et sur l'analyse de canaux auxiliaires (*Side-Channel Analysis*, SCA) qui seront décrites plus loin : ce sont des méthodes dites "aidantes".

Les méthodes invasives présentées précédemment comportent au moins deux limitations. Tout d'abord, même si ces méthodes amènent un surcoût souvent acceptable, certaines méthodes disposent d'un taux de détection assez inférieur à 100% (par exemple, 80% dans [CB09]), ce qui est plutôt faible comparé aux autres méthodes que nous détaillerons plus loin. Ensuite, la mise à l'échelle de certaines des méthodes décrites (notamment les méthodes d'obfuscation), c'est-à-dire la faisabilité de leur application à des circuits plus complexes, n'a jamais été prouvée, et semble par nature très difficile à obtenir. Pour prendre l'exemple de l'obfuscation d'un circuit intégré complexe, il semble impossible d'obtenir un circuit obfusqué en un temps raisonnable pour le concepteur du circuit de par le très grand nombre d'états internes à obfusquer. En tout cas, à l'heure actuelle, aucun résultat tangible sur des circuits complexes n'est venu confirmer la mise à l'échelle de l'obfuscation de codes RTL/VHDL à des fins de protection contre les HTs.

*Méthodes non-invasives.* Dans le cas de méthodes non-invasives, les comportements des CIs testés sont comparés avec ceux de CIs dits "de référence". Il existe 2 types de méthodes non-invasives : celles opérant en phase de **fonctionnement** et celles opérant en phase de **test**.

Plusieurs méthodes opérant en phase de **fonctionnement** ont été proposées dans la littérature. Par exemple, l'approche séduisante décrite dans [AB09] propose d'ajouter de la logique reconfigurable nommée DEFENSE (Design-For-ENabling-SEcurity) dans l'architecture du CI qui implémente des moniteurs de sécurité en temps réel. L'utilisateur peut insérer cette logique au niveau RTL (*Register Transfer Level*). Les signaux importants à contrôler sont reliés à des machines à états finis. Un processeur est également ajouté : son but est de reconfigurer les zones où les signaux doivent être contrôlés et les machines à états finis correspondantes. Cette reconfiguration n'interrompt pas le flot d'exécution d'opérations du CI testé. Toutes ces reconfigurations sont stockées chiffrées dans une mémoire Flash. DEFENSE est peu documenté : notamment, aucune estimation de son surcoût n'est fournie. Et il en est de même pour la plupart des dispositifs de détection de HTs en phase de fonctionnement.

Des méthodes de détection de HTs non-invasives en phase de **test** (avant déploiement) ont également été proposées dans la littérature. Deux méthodes principales ont été mises en oeuvre : l'analyse de testabilité et la SCA. Ces méthodes ont l'avantage de ne pas engendrer de surcoût sur les performances du CI. Malheureusement, la plupart des méthodes nécessitent un modèle de référence (*golden circuit*) du CI, autrement dit un composant dont on est sûr qu'il ne contient pas de HT, ce qui est difficile à garantir.

Les méthodes de détection basées sur l'analyse de testabilité sont intrinsèquement difficiles à utiliser car les HTs peuvent avoir différentes formes et engendrer un grand nombre de *payloads* différents. Ainsi, générer un ensemble exhaustif de vecteurs de test permettant de détecter tous les HTs possibles pouvant être insérés dans le CI cible est une tâche difficile. C'est pourquoi il est souvent proposé d'utiliser des techniques statistiques plutôt qu'exhaustives pour générer des vecteurs de test permettant de détecter des HTs. Par exemple, [CWP+09] détecte

tout d'abord les conditions à faible probabilité d'occurrence des valeurs logiques sur les noeuds internes du CI testé. Ensuite, il sélectionne tous les HTs possibles qui peuvent être déclenchés par ces conditions rares. Enfin, il trouve un ensemble compact de vecteurs de test qui peuvent engendrer ces rares conditions sur chaque noeud du circuit pris individuellement. Cette méthode semble être efficace : les résultats de simulation montrent un taux de détection des HTs comparable ou meilleur que l'état de l'art avec une réduction du temps de test de l'ordre de 85% comparé à une approche privilégiant une injection de vecteurs de test aléatoires.

Les méthodes de détection basées sur les SCAs ont quant à elles pour but de détecter des HTs en analysant les paramètres physiques du CI testé tels que la consommation électrique, les émanations électromagnétiques, ou encore les délais des chemins combinatoires du CI. Cette méthode non-destructive possède un avantage principal : même si un HT n'est pas déclenché (par exemple par une méthode basée sur une analyse de testabilité [CWP+09]), celui-ci émet de toute façon des canaux auxiliaires, et ces derniers peuvent être utilisés pour le détecter. Il est souvent proposé dans la littérature de comparer la consommation électrique d'un CI suspect à celle d'un *golden circuit*. Pour s'assurer que le CI *golden circuit* est bien dépourvu de HTs, il doit être investigué à l'aide de méthodes destructives (*reverse-engineering*). Malgré le fait que cette étape soit coûteuse, on peut estimer que vu que seuls quelques CIs d'une famille entière subissent un *reverse-engineering*, le coût global de l'opération reste abordable. Une fois que les *golden circuits* de consommation ont été recueillis, les CIs restants peuvent être testés sans les détruire.

En règle générale, il est difficile de développer des méthodes de détection de HTs basées sur les SCAs à cause des variations de *process* de fabrication de CIs (ces dernières provenant du fait que deux CIs d'une même série ne sont pas totalement identiques). En effet, ces variations peuvent cacher la contribution d'un HT dans la consommation électrique globale du CI testé et ce d'autant plus que la part de la consommation d'un HT est très faible en regard de la consommation totale du CI. Il est proposé dans [DNCB10] une approche basée sur les SCAs, appelée "*self-referencing*" associée à un algorithme de génération de vecteurs de tests permettant d'améliorer le taux de détection de HTs en présence de larges variations de *process*. Le *self-referencing* consiste à comparer les courbes de consommation électrique provenant de la même région de plusieurs CIs, ce qui permet d'annuler les effets du bruit amenés par les variations de *process* en utilisant des corrélations spatiales entre régions. Pour amplifier l'effet du HT inséré sur la consommation électrique, [DNCB10] propose une méthode de génération de vecteurs de tests qui permet d'amplifier l'activité électrique dans certaines régions des CIs testés, tout en minimisant l'activité dans d'autres régions.

**Bilan** Conformément à ce qui a pu être constaté dans cette section, aucune méthode de détection de HTs n'est capable de détecter seule 100% des HTs dans n'importe quelles conditions. De plus, chaque méthode doit affronter ses propres

limitations (variations de *process*, surcoût de la méthode, *etc.*). Le projet de recherche français HOMERE+ vise justement à faire progresser ces méthodes.

## 3 Quelques résultats du projet HOMERE+

Le projet HOMERE+ a apporté à la communauté des améliorations significatives pour la détection de HTs. Il a été développé notamment des méthodes de détection de HTs :

- par inspection visuelle à bas coût via rétro-conception partielle [CMFT15],
- par analyse de délais internes des circuits [NEB+15],
- par analyse des canaux auxiliaires du circuit cible [NNG+14],
- par activation de sa *payload* par injection de vecteurs de tests adaptés ([DBF+15] décrit une approche identifiant les signaux pouvant déclencher un *trigger* de HT, cf. section 3.4 pour plus de détails),
- par vérification en phase de fonctionnement du système des propriétés temporelles des signaux internes du circuit en utilisant le principe de la PSL (*Property Specification Language*).

Par ailleurs, des méthodes préventives ont été également développées permettant de rendre les infections bénignes ou plus difficiles pour l'attaquant :

- Une méthode de congestion de circuit [BDG+13], rendant très difficile l'insertion de HTs ;
- Des méthodes d'obfuscation [DBN+14] du circuit cible et de codage de ses états internes [NGB+14] rendant difficile la compréhension du circuit par l'attaquant et masquant les valeurs internes du circuit propices à l'infection.

### 3.1 Détection visuelle de HTs bas coût via rétro-conception partielle

Vis-à-vis de l'état de l'art et les contraintes industrielles de temps et de coût, nous proposons une méthode de détection de HTs basée sur une rétro-conception matérielle partielle [CMFT15]. Nous proposons ainsi une méthodologie permettant d'accéder, d'enregistrer et d'analyser une seule couche physique d'un circuit intégré (zones actives). Elle est représentative des fonctions de base implémentées dans un circuit.

Notre méthode se compose de trois étapes :

- préparation d'échantillon bas coût pour accéder aux zones actives,
- acquisition et recalage automatisés d'images multiples,
- corrélation entre référence et circuit à tester par traitement d'image.

Nous avons validé expérimentalement la méthodologie proposée sur un couple de circuits dont l'un est réellement infecté [MGK+13]. La figure 1 met en évidence les disparités entre circuits sur une surface réduite. Sur ce dernier (à gauche), on remarque que des cellules de remplissage ont notamment été remplacées par d'autres cellules [CLMFT15]. Le HT est composé de quelques dizaines de portes (0,5% de portes supplémentaires par rapport au circuit authentique).

Nous indiquons que nous couvrons l'intégralité de la surface de cette couche d'intérêt et que notre méthode est indépendante du positionnement spatial du HT. Le flot de détection complet requiert seulement une heure et la plupart des tâches ont été automatisées et rendues accessibles à bas coût. De plus, outre la détection du HT, notre méthode permet également de connaître le nombre de portes rajoutées/modifiées ainsi que leur localisation.

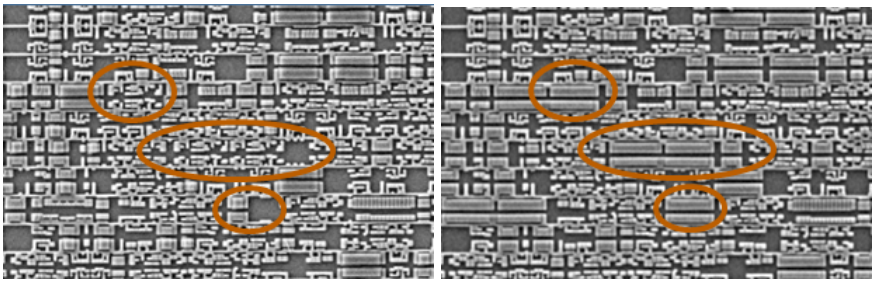


**Figure 1.** a) Circuit infecté (à gauche), b) Circuit non infecté (à droite)

La méthode est robuste vis-à-vis du noeud technologique visé, des variations de *process* ou encore des variations des équipements de mesure. La détection de modifications malicieuses est ainsi possible et efficace vis-à-vis de la grande variété de HTs possibles. En effet, notre méthodologie est indépendante du moment d'insertion, de l'activité, de l'effet, de la localisation ou encore des caractéristiques physiques du HT.

### 3.2 Détection de HTs par analyse des délais internes du circuit

**Principe de la méthode** L'approche de cette méthode de détection repose sur le principe de fonctionnement des circuits synchrones. Tout calcul est cadencé par le signal global au circuit : l'horloge. À chaque front montant de l'horloge, la donnée quitte le registre (noté DFF en figure 2) à travers la logique combinatoire jusqu'au prochain registre et sera échantillonnée au front montant de la prochaine période. La figure 2 illustre ce principe de fonctionnement.

La période d'horloge doit cependant respecter les contraintes de temps particulières définies par l'équation

$$T_{clk} > D_{clk2q} + D_{pMax} + T_{setup} - T_{skew} + T_{jitter} \quad (1)$$

Ici, on exploite le mécanisme de *glitch* d'horloge proposé par [ADN+10]. En effet, la réduction d'une période ciblée dans le calcul du circuit par pas constant d'une durée de $\delta$ ps (35 ps dans notre cas) conduit à la violation du temps de *set-up* et donc à la métastabilité de certaines bascules. De cette façon, cette diminution locale de la période d'horloge est un bon vecteur pour induire des violations de délais à l'intérieur d'un circuit intégré.

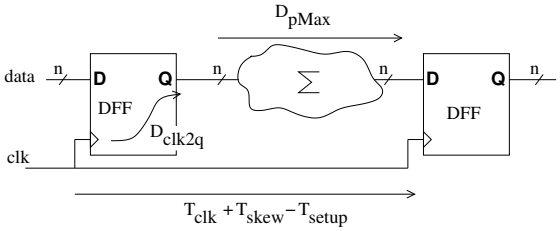



**Figure 2.** Modèle de fonctionnement d'un circuit synchrone

L'hypothèse pour la détection est la suivante : l'ajout d'un HT dans le circuit aura un impact sur ses chemins internes. La figure 3 illustre le procédé expérimental de détection. Les délais des chemins internes $a$, $b$, $c$ sont mesurés, pour une période choisie, par des diminutions successives de la période d'horloge avec des pas de 35 ps. On mesure, dans un premier temps, la valeur de ces délais avec un circuit de référence (non infecté). Ensuite, on recommence la même opération avec les circuits sous test (potentiellement infectés). L'augmentation du délai de certains chemins induit par la présence du HT pourra ainsi être distingué par une mesure différentielle entre le circuit dit sain et le circuit infecté.

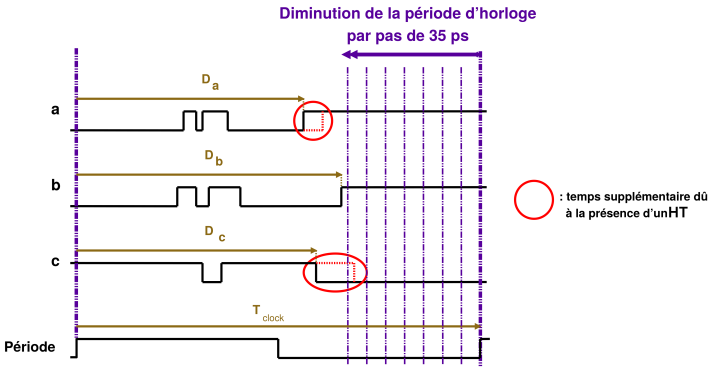



**Figure 3.** Mécanisme de mesure interne de délais

Dans l'exemple de la figure 3, le chemin critique est $b$. Bien que le HT ait pu ne pas impacter la structure logique de ce chemin, celui-ci pourrait être influencé (par le biais du réseau d'alimentation) par le *trigger* du HT.

**Modèle et résultats** La validation de la méthode de détection s'est faite par des campagnes sur un algorithme de chiffrement symétrique AES 128 bits embarqué sur des cartes de développement à base de FPGA Xilinx Spartan 3. Le contrôle de l'horloge est fait par une seconde carte à base de Virtex 5. La dixième

ronde de l'AES a été ciblée, ainsi le chiffré fauté est directement obtenu. Les mesures de délais internes sont faites sur une version saine du circuit puis sur deux versions infectées (une par un HT combinatoire, l'autre par un HT séquentiel).

Chaque campagne est composée de 10000 chiffrements avec texte et clé aléatoires. Ainsi, les chemins impactés varient en fonction des entrées. Pour chaque calcul, la période est diminuée de 51 pas de 35 ps. Enfin, une moyenne de ces mesures est effectuée sur dix itérations pour limiter l'impact de la métastabilité entre deux incréments du pas de réduction de l'horloge [NEB+15].

Le calcul des délais repose sur le modèle présenté dans [EZRR15] qui prend notamment en compte les variations de *process* inter et intra circuit. Le calcul des différences de délais, illustré figure 4, montre l'impact de l'insertion d'un HT sur différents bits du chemin de données de l'AES. On remarque, d'une part que l'infection est significative sur plusieurs bits même si ce ne sont pas les bits du chemin critique. D'autre part, le type d'infection combinatoire ou séquentielle peut être différencié par cette méthode.

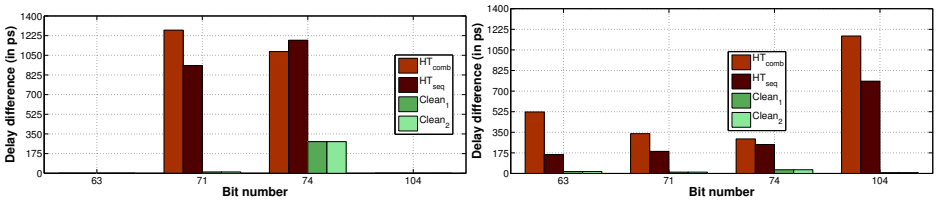



**Figure 4.** Résultats de campagne pour 2 chiffrés

La détection par analyse de délais interne présentée dans cette partie, repose sur la modification locale de la période d'horloge lors des calculs effectués par un circuit. Elle nécessite l'accès à l'horloge interne du circuit. Mais, elle permet, à partir d'un circuit de référence, de déterminer des différences entre circuits sains et circuits sous test à partir d'un modèle de délais qui prend en compte les variations de *process* qui pourraient venir parasiter les campagnes de mesures.

### 3.3 Détection des HTs par analyse des canaux auxiliaires du circuit cible

Cette technique de détection est basée sur l'observation suivante : les HTs sont des modifications malveillantes du circuit original, c'est-à-dire qu'ils sont censés modifier les délais internes, la consommation électrique ou le rayonnement électromagnétique du circuit testé. La méthode de détection des HTs par analyse des canaux auxiliaires consiste à mesurer et comparer ces comportements physiques du circuit de test avec ceux du circuit de référence (circuit non infecté) afin de détecter les changements liés aux HTs. Mais les impacts de l'insertion des HTs sur le circuit original dépendent de la taille du HT et aussi des variations de *process*.

Dans le papier [NNG+14], nous présentons une méthode qui évalue la probabilité de détection des HTs en fonction de sa taille malgré la présence des variations de *process*. Pour cette expérience, trois HTs combinatoires, de taille 0,5%, 1% et 1,7%, sont insérés afin de calculer la probabilité de détection par mesures du rayonnement électromagnétique du circuit. 3 approches ont été utilisées pour traiter les traces électromagnétiques et pour calculer le probabilité de détection :

- $1^{ère}$ approche : la somme des valeurs absolues de différences entre les traces EM.
- $2^{ème}$ approche : les maximums locaux des valeurs absolues de différences entre les traces EM.
- $3^{ème}$ approche : utiliser un seuil sur les valeurs absolues de différences entre les traces EM.

| | **HT 1 (0,5%)** | **HT 2 (1%)** | **HT 3 (1,7%)** |
|---|---|---|---|
| $1^{ère}$ **Approche** | 43% | 34% | 9% |
| $2^{ème}$ **Approche** | 26% | 17% | 5% |
| $3^{ème}$ **Approche** | 24% | 0,017% | 0,011% |

**Table 1.** La probabilité de faux négatifs

Les résultats dans la table 1 montre que l'on peut facilement détecter des HTs de taille supérieure à 1% avec un taux d'erreur inférieur à 1%. On a évalué également l'impact du placement des HTs sur les mesures électromagnétiques. Pour le faire, on a placé le HT de taille de 1,7% à 3 endroits différents : à **l'intérieur** du circuit original, à **l'extérieur** du circuit original, et **dispersé** dans 6 endroits différents du circuit original.

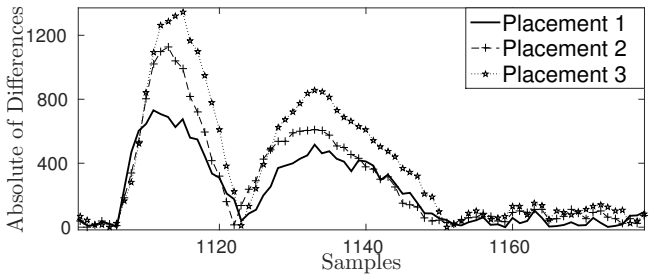



**Figure 5.** L'impact du placement des HTs sur les mesures électromagnétiques

La figure 3.3 montre que plus le HT est éloigné du circuit original, plus il est facile à détecter. Cette méthode permet de détecter des HTs même s'ils sont

inactifs. Elle peut également détecter des petits HTs. Mais l'inconvénient principal de cette méthode est la nécessité d'un (ou un ensemble) circuit de référence. L'application de la méthode visuelle (cf. section 3.1) peut être envisagée afin d'obtenir ces dits circuits.

### 3.4 Analyse logique

La méthode proposée dans [DBF+15] se base sur le même principe que dans [CWP+09] : identifier les signaux susceptibles d'être les signaux d'entrée du *trigger* d'un HT potentiel puis générer des vecteurs aptes à déclencher ces HTs. Cependant, nous améliorons ici l'identification des *triggers* potentiels en nous basant sur 3 critères. L'hypothèse de base est que les HTs sont "furtifs" par nature, non seulement d'un point de vue test (ils ne doivent se déclencher que dans de très rares conditions), mais également d'un point de vue performances (ils ne doivent pas modifier les délais internes d'un circuit) et visibilité (ils doivent modifier le *layout* le moins possible). De plus, nous améliorons la définition d'une "condition rare". Dans toutes les approches de la littérature, il est supposé que la valeur de déclenchement d'un *trigger* potentiel est une combinaison de valeurs rares sur des signaux faiblement contrôlables. Toutefois, cette hypothèse est trop restrictive : il est également possible qu'un ensemble de signaux individuellement parfaitement contrôlables génère une valeur rare.

En ce sens, notre approche identifie les signaux pouvant déclencher un *trigger* de HT comme suit : (1) tri des signaux ayant une marge temporelle non nulle (i.e. pour lesquels l'insertion d'une porte n'a pas d'impact sur la fréquence de fonctionnement du circuit cible), (2) identification des signaux précédemment triés proches dans le *layout* pour former des groupes, (3) calcul de la probabilité d'occurrence de chaque valeur de chaque groupe de signaux de façon à trouver lesquels génèrent une valeur rare. À partir des groupes et valeurs rares identifiés, une dernière étape consiste à trouver des vecteurs capables de justifier les dites valeurs. Pour ce faire, un outil d'ATPG (pour *Automatic Test Pattern Generation*) est utilisé. Un ensemble réduit de vecteurs est ainsi généré, dédié à l'activation accélérée de *triggers* potentiels, activation étant espérée pendant la phase de test “classique” du circuit.

### 3.5 Détection des HTs en *run-time*

La détection en *run-time* représente la dernière ligne de défense contre les HTs. Cette méthode permet de détecter des HTs qui ne sont pas détectés par les méthodes de test tels que les tests logiques ou l'analyse des canaux auxiliaires. Le principe de cette méthode est basé sur l'observation suivante : quand le HT est activé, il peut changer le comportement du circuit, faire un déni de service ou voler des informations sensibles. Dans la majorité des cas, la *payload* du HT va modifier une ou plusieurs propriétés du circuit original. Dans cet esprit, l'idée de notre méthode de détection en *run-time* est de définir une liste de propriétés importantes/sensibles du circuit (extraites via d'intenses simulations ou grâce aux spécifications du circuit) et créer un module qui les vérifie en temps réel.

Une fois que l'une de ces propriétés est violée par un HT, un signal d'alarme va être généré et une contre-mesure pourra ensuite être déclenchée pour contre-carrer la *payload* du HT. Pour ce faire, nous avons utilisé le langage d'assertion PSL (*Property Specification Language*) pour décrire et simuler les propriétés qui seront vérifiées. Ensuite, un logiciel nous permet de transformer le code PSL en un module synthétisable et intégrable (appelé *Hardware Property Checker*) dans le circuit original. Ce module va réaliser la détection des HTs en *run-time*.

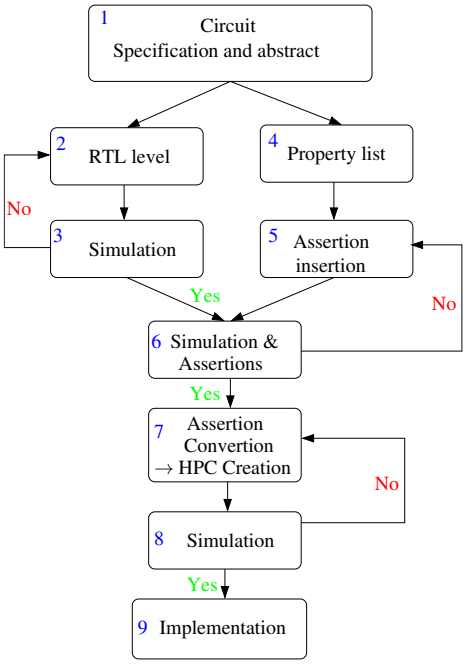



**Figure 6.** Flot de conception en intégrant le *Hardware Property Checker* (HPC)

La figure 3.5 présente le flot de conception en intégrant le HPC dans le circuit original. Cette méthode permet de réduire le temps de développement du HPC. Les résultats actuels sur le processeur Leon 2 montrent que la taille du HPC est relativement petit par rapport au circuit original. Nous avons implanté dans ce processeur plus de 200 capteurs vérifiant des propriétés temporelles de signaux internes, et le surcout matériel n'est que de 8% seulement.

## Conclusion et bilan

Cet article a présenté une cause de défaillance des systèmes méconnue car très spécialisée : les chevaux de Troie matériels. L'attaquant considéré est ici

très puissant et doté de moyens importants, mais ce risque, vu les conséquences catastrophiques qu'il peut engendrer sur des systèmes critiques, ne peut être négligé.

Il a été détaillé dans cet article les méthodes développées dans le projet HOMERE+ permettant d'augmenter la résilience des systèmes face aux HTs. L'approche multidisciplinaire d'HOMERE+ peut (doit) être appliquée comme suit.

1. Tout d'abord, nous conseillons aux concepteurs de circuits d'intégrer des méthodes préventives lui permettant de rendre très difficile l'insertion de HTs ou de rendre cette insertion bénigne pour le système. Par exemple, les méthodes préventives permettant de "congestionner" un circuit [BDG+13] [BPD+15] nous autorisent à obtenir des circuits occupés à 95%, ce qui laisse objectivement très peu de place pour insérer un HT ayant un effet utile pour un attaquant [2]. De même, si on utilise la technique de codage des états internes du circuit cible [NGB+14], cela forcera un attaquant à insérer un HT avec un nombre important d'entrées pour son *trigger*, ce qui facilitera sa détection visuelle [CLMFT15] (un gros HT se verra plus facilement) et par analyse de canaux auxiliaires (un HT plus gros consomme plus d'énergie, rayonne plus ou augmente certains délais internes du circuit).
2. Une fois ces méthodes préventives implantées et le circuit cible sorti de fonderie, une série de tests additionnels doit être effectuée avant son déploiement sur le terrain opérationnel. Concernant la détection de HTs par analyse de canaux auxiliaires (consommation électrique, rayonnement électromagnétique et analyse de délais internes), un pas significatif a été franchi dans HOMERE+, puisque nous avons pu valider nos méthodes sur des circuits réels (et non des simulations), et ce, en prenant en compte les variations de *process* [NNG+14] [EZRR15]. Les temps de test des circuits cibles (par analyse logique, visuelle ou par canaux auxiliaires) tendent à être compatibles avec les contraintes industrielles du secteur de la microélectronique. Pour le cas particulier du test logique, nous devons faire des hypothèses sur les HTs les plus probables pour le circuit cible [3] pour ne pas devoir appliquer un trop gros nombre de vecteurs d'entrée au circuit : il faut donc identifier les sites propices à l'insertion de HTs afin qu'ils puissent rester furtifs et échapper aux tests additionnels (place disponible dans le *layout*, peu de modifications des délais internes, signaux à valeurs rares pour le *trigger*).
3. Une fois les méthodes préventives implantées dans le circuit cible, et les tests additionnels sécuritaires effectués, la probabilité d'insérer un HT pouvant passer à travers les mailles de tous les filets est objectivement très faible. Cependant, pour détecter ces très hypothétiques HTs, une troisième et dernière ligne de défense peut (doit) être implantée : la détection en *run-time*.

2. Il est même montré dans [BPD+15] que les outils de conception de circuits refusent cette insertion dans ces conditions car le routage du HT, c'est-à-dire sa connexion au circuit cible, est techniquement impossible.

3. Ce qui est au fond une méthodologie courante en cybersécurité : pour savoir protéger, il faut savoir comment attaquer.

Un de nos principaux résultats se rapproche de l'architecture idéalisée de DEFENSE, en étant plus concret, et en ne mettant pas en jeu de reprogrammation dynamique du circuit en cas de détection de HTs, phénomène dont il est difficile d'avoir une totale confiance. Nous avons fait le choix de placer des capteurs statiquement, mais obfusqués (pour rendre difficile l'infection de ces capteurs) et dont les configurations sont programmées après fonderie (ainsi, la fonderie voit des capteurs, mais ne peut pas comprendre les propriétés scrutées). *A minima*, nous proposons une méthodologie de détection *post mortem* via l'utilisation d'assertions PSL. D'autres méthodes plus classiques ont été également développées, basées sur de la redondance d'instructions (une instruction vérolée peut être définitivement black-listée, et remplacée par une série d'instructions équivalentes), ou de la triplication (*Triple Modular Redundancy*, TMR) de circuits ayant un codage interne différent afin d'éviter l'infection simple et simultanée de plusieurs unités par un seul HT.

En résumé, malgré le fait que le consortium d'HOMERE+ ait été confronté pendant 3 ans à des attaquants aux moyens illimités (pouvant même tenter de deviner nos contre-mesures insérées), nous avons réussi à fournir à la communauté des avancées multiples et significatives dans le domaine de la neutralisation de HTs.

## Références


[AB09] M. Abramovici and P. Bradley. Integrated Circuit Security - New Threats and Solutions. In *Proc. Workshop en Cyber Security and Information Intelligence Research – CSIIRW*, 2009.

[Ade08] S. Adee. The Hunt for the Kill Switch. In *Proc. IEEE Spectrum*, volume 45, pages 34–39, 2008.

[ADN+10] M. Agoyan, J.-M. Dutertre, D. Naccache, B. Robisson, and A. Tria. When Clocks Fail : On Critical Paths and Clock Faults. In *Proc. Smart Card Research and Advanced Application – CARDIS*, pages 182–193, 2010.

[BDG+13] S. Bhasin, J.-L. Danger, S. Guilley, X. T. Ngo, and L. Sauvage. Hardware Trojan Horses in Cryptographic IP Cores. In *Proc. IEEE Fault Diagnosis and Tolerance in Cryptography – FDTC*, pages 15–29, 2013.

[BPD+15] P.-S. Ba, M. Palanichamy, S. Dupuis, M.-L. Flottes, G. Di Natale, and B. Rouzeyre. Hardware Trojan Prevention using Layout-Level Design Approach. In *Proc. IEEE European Conference on Circuit Theory and Design – ECCTD*, 2015.

[CB09] R. S. Chakraborty and S. Bhunia. Security against Hardware Trojan through a Novel Application of Design Obfuscation. In *Proc. IEEE International Conference on Computer-Aided Design – ICCAD*, pages 113–116, 2009.

[CLMFT15] F. Courbon, P. Loubet-Moundi, J. J. A. Fournier, and A. Tria. SEMBA : a SEM Based Acquisition Technique for Fast Invasive Hardware Trojan

Detection. In *Proc. IEEE European Conference on Circuit Theory and Design − ECCTD*, 2015.

[CMFT15] F. Courbon, P.-L. Moundi, J. J. A. Fournier, and A. Tria. A High Efficiency Hardware Trojan Detection Technique Based on Fast SEM Imaging. In *Proc. Design, Automation and Test in Europe − DATE*, pages 788–793, 2015.

[CNB09] R. S. Chakraborty, S. Narasimhan, and S. Bhunia. Hardware Trojan : Threats and Emerging Solutions. In *Proc. IEEE Workshop on High Level Design Validation and Test − HLDVT*, pages 166–171, 2009.

[CWP+09] R. S. Chakraborty, F. Wolff, S. Paul, C. Papachristou, and S. Bhunia. MERO : A Statistical Approach for Hardware Trojan Detection. In *Proc. Cryptographic Hardware and Embedded Systems − CHES*, volume 5747, pages 396–410, 2009.

[DBF+15] S. Dupuis, P.-S. Ba, M.-L. Flottes, G. Di Natale, and B. Rouzeyre. New Testing Procedure for Finding Insertion Sites of Stealthy Hardware Trojans. In *Proc. Design, Automation and Test in Europe − DATE*, pages 776–781, 2015.

[DBN+14] S. Dupuis, P.-S. Ba, G. Di Natale, M.-L. Flottes, and B. Rouzeyre. A Novel Hardware Logic Encryption Technique for thwarting Illegal Overproduction and Hardware Trojans. In *Proc. IEEE International On-Line Testing Symposium − IOLTS*, pages 49–54, 2014.

[DNCB10] D. Du, S. Narasimhan, R. S. Chakraborty, and S. Bhunia. Self-Referencing : A Scalable Side-Channel Approach for Hardware Trojan Detection. In *Proc. Cryptographic Hardware and Embedded Systems − CHES*, volume 6225, pages 173–187, 2010.

[EZRR15] I. Exurville, L. Zussa, J.-B. Rigaud, and B. Robisson. Resilient Hardware Trojans Detection based on Path Delay Measurements. In *Proc. IEEE International Symposium on Hardware Oriented Security and Trust − HOST*, pages 151–156, 2015.

[Kum00] J. Kumagai. Chip Detectives. In *IEEE Spectrum*, volume 37, pages 43–48, 2000.

[MGK+13] M. Muehlberghuber, F. K. Gürkaynak, T. Korak, P. Dunst, and M. Hutter. Red Team vs. Blue Team Hardware Trojan Analysis : Detection of a Hardware Trojan on an Actual ASIC. In *Proc. Hardware and Architectural Support for Security and Privacy − HASP*, 2013.

[NEB+15] X. T. Ngo, I. Exurville, S. Bhasin, J.-L. Danger, S. Guilley, Z. Najm, J.-B. Rigaud, and B. Robisson. Hardware Trojan Detection by Delay and Electromagnetic Measurements. In *Proc. Design, Automation and Test in Europe − DATE*, pages 782–787, 2015.

[NGB+14] X. T. Ngo, S. Guilley, S. Bhasin, J.-L. Danger, and Z. Najm. Encoding the State of Integrated Circuits : A Proactive and Reactive Protection against Hardware Trojans Horses. In *Proc. ACM Workshop on Embedded Systems Security − WESS*, 2014.

[NNG+14] X. T. Ngo, Z. Najm, S. Guilley, S. Bhasin, and J.-L. Danger. Method Taking into Account Process Dispersion to Detect Hardware Trojan Horse by Side-Channel. In *Proc. Security Proofs for Embedded Systems − PROOFS*, 2014.

# Chiffrement des données dans le cloud : Comment allier sûreté de fonctionnement et sécurité ?

Aurélien MAGNIEZ, Stéphanie MBAPPE, Georges MILLET-LACOMBE

Orange Cyberdefense

## 1 Introduction

La résilience peut être définie de plusieurs manières, mais le point commun à toutes ces définitions est la capacité à retrouver ses propriétés initiales après une agression. Dans le domaine des systèmes d'information, la résilience est souvent assimilée à :

- la tolérance aux pannes (continuité d'activité)
- la reprise d'activité
- la redondance des systèmes et des équipements
- la garantie de la disponibilité du service du point de vue de l'utilisateur

L'informatique dans les nuages ou le « cloud computing » (cloud) est souvent considéré comme intrinsèquement résilient et apparait comme l'un des moyens d'assurer cette capacité à un meilleur coût. Au sein de ce type d'infrastructure, le logiciel en tant que service ou Software as a Service (SaaS) est le « fer de lance de la révolution cloud » et est utilisé par 55% des entreprises en France.

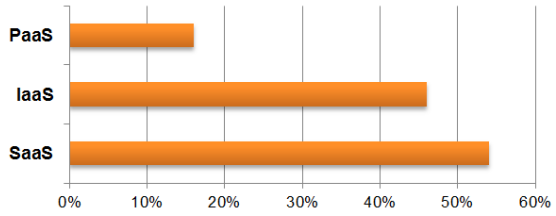



**Fig. 1.** : Types de cloud utilisés (Source PAC 2014)

Dans les entreprises, les métiers poussent de plus en plus pour l'utilisation d'applications SaaS, ceci souvent avec ou sans l'approbation de la DSI. L'expression « shadow IT » est communément utilisée pour nommer ce phénomène d'usage de SaaS sans approbation de l'IT, aujourd'hui très largement sous-estimé dans les entreprises.

Les promesses du modèle SaaS sont nombreuses, notamment :

- La promesse du partout, à tout instant, sur tout support (ATAWAD[1])

[1] ATAWAD : Any time, Any Where With Any Device

- Le paiement à l'usage d'un ensemble de services englobant les coûts de licence, de maintenance et d'infrastructure (OPEX[2])
- L'absence ou la faible immobilisation de ressource (CAPEX[3])
- La prise en charge de la gestion des configurations et des sauvegardes
- La mise à jour continue des outils
- Le faible coût de développement et d'intégration par une standardisation des outils et des usages

Ces promesses (quoi que discutables) sont attrayantes quand la résilience d'applications critiques doit être garantie.

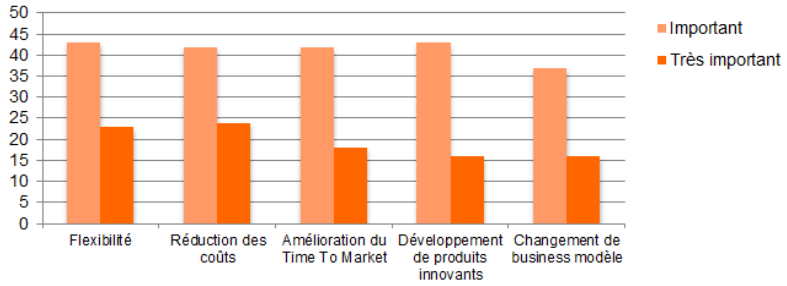



**Fig. 2.** : Facteurs d'adoption des solutions cloud - Décembre 2014 (Source PAC)

Dans un contexte de forte croissance autour de l'utilisation des services cloud par les entreprises, les DSI ont aussi la volonté d'adopter une posture de « cloud-enabler » et d'accompagner les directions métiers dans leur transformation digitale.

L'ouverture du système d'information demande aujourd'hui aux DSI et RSSI de requalifier et d'évaluer en continu les risques auxquels s'expose leur organisation.

La limitation majeure qui doit être prise en compte par les entreprises désireuses de recourir à un service cloud de type SaaS concerne la confidentialité et l'intégrité des données exportées hors du périmètre du système d'information contrôlé par celles-ci.

Les mécanismes de protection du patrimoine informationnel dans le cloud (contrôle d'accès, protection des données, ...) ne sont pas sans impact sur la résilience d'une application SaaS. L'usage de mécanismes tiers pour assurer la protection des données peut en effet altérer ses fonctionnalités, ses performances et le plus dommage sa disponibilité.

[2] OPEX : Operating Expenses
[3] CAPEX : Capital Expenditure

L'entreprise Orange Business Services est confrontée aux problématiques liées à la sécurisation des usages SaaS à la fois pour ses besoins internes et pour ceux de ses clients. Plusieurs études ont été menées ces derniers mois :

- protection des données dans le cadre de l'externalisation d'un outil CRM[4]
- étude de marché des solutions de protection des données SaaS
- analyse de risques liées à l'externalisation d'applications CRM, ITSM [5] ou encore bureautique (messagerie, productivité, transfert de fichiers…)

Sur la base de notre retour d'expérience, serons détaillées dans la suite :

- les problématiques de sécurisation des données envoyées dans le cloud
- les solutions permettant de garantir la confidentialité des données les plus sensibles manipulées par les applications SaaS via des techniques de chiffrement
- les limites de ces solutions, en particulier leur impact sur la disponibilité
- et les alternatives possibles pour bénéficier de la résilience du cloud avec un minimum de confidentialité

## 2 Besoins de sécurité des données

Dès lors que l'entreprise a recours à une application fournie par un hébergeur à l'extérieur de son périmètre, il devient nécessaire d'envisager que des données sensibles, sur le plan de leur confidentialité, soient manipulées par cette application.

Une analyse de risque est souvent recommandée / demandée à ce stade, afin de :

- prendre en compte le contexte de l'entreprise
- identifier les risques inhérents à ce choix
- et classer les risques identifiés selon leurs degrés.

Dans un contexte général, les menaces suivantes sont généralement retenues :

- Les menaces de rang étatique, s'intéressant aux données pouvant concerner les intérêts des états, et / ou des grandes entreprises se donnant les moyens de faire appel à des services d'intelligence économique, plus ou moins légitimes selon les pays. La majorité des fournisseurs d'applications SaaS est américaine.
- Les menaces d'origines malveillantes ayant le plus souvent un but lucratif et/ou personnel, se matérialisant sous la forme de logiciels malveillants introduit au sein du SI par tous les moyens possibles, et dont le spectre d'activité est assez peu ciblé.

[4] CRM : Customer Relationship Management
[5] ITSM : Information Technology Service Management

- Les menaces d'origine internes à l'entité, qui seront toujours présentes, sauf qu'il faut y ajouter celles de l'hébergeur dans le cas d'une externalisation d'un service applicatif.

Trois catégories de risques peuvent se distinguer pour ce type de scénario :

- L'hébergeur compromet des données sensibles appartenant à l'entreprise (qui est son client), en exploitant ses droits d'accès liés à son rôle d'administration de l'application.
- Un État (au sein duquel évolue l'hébergeur ou une partie significative de son système d'information) demande et obtient un accès aux données sensibles de ce client en exploitant ses prérogatives régaliennes et en les imposant à l'hébergeur, de manière officielle et légale ou non (Patriot Act par exemple).
- Une autre entreprise utilisant les mêmes services d'application SaaS chez le même hébergeur, compromet ses données sensibles en exploitant une ou plusieurs vulnérabilités inhérentes au système lui-même et ou à l'application utilisée, notamment en ce qui concerne l'isolation "multi-tenant" censée assurer la confidentialité par confinement des activités de chaque entreprise vis-à-vis des autres.

Que les données aient un niveau de sensibilité élevé ou non, le constat est que l'hébergeur a de facto accès à ces données en permanence en raison de son positionnement privilégié d'administrateur et d'opérateur de l'application.

Les principales réponses pour mitiger ces risques sont:

- **qualifier la confiance de l'entité envers le fournisseur / hébergeur de l'application**, selon des critères qui sont à définir dans son contexte propre, car ils devront tenir compte des besoins de sécurité des données à protéger, des sources de menaces considérées, des contraintes opérationnelles métiers de l'entreprise.
- **protéger spécifiquement la confidentialité de ces données particulières**, sans pour autant renoncer au bénéfice de l'usage de l'application visée, par exemple en ayant recours à des mécanismes de chiffrement ou de tokenisation (Cf. §3.13.1).

Le retour d'expérience montre qu'aucune de ces réponses n'est vraiment satisfaisante, comme présenté dans la partie suivante.

Le point essentiel sur lequel il est important de s'arrêter ici, tient au fait que les capacités de résilience promises par l'application SaaS visée ne pourront être réelles que si le périmètre fonctionnel de l'application est conservé malgré le recours à des moyens de chiffrement ou de tokenisation. Cela implique de mener une réflexion approfondie sur les besoins métiers la sensibilité de l'information et les impacts opérationnels de ces choix.

Cela implique également de disposer d'un outil capable de mettre en œuvre une capacité de protection de la confidentialité à un niveau de granularité très fine, typiquement au niveau d'un champ unitaire pour un formulaire de données, ces applications complexes manipulant en général des données très structurées.

## 3 Mise en place d'un CASB

Les DSI sont de plus en plus nombreuses à intégrer dans leur stratégie cloud la mise en place d'une passerelle de sécurité communément appelé CASB (cloud Access Security Broker) pour encadrer, faciliter et accompagner le développement de leurs activités via l'usage de services cloud.

Plus précisément, la mise en place d'un CASB répond aux enjeux suivants :

- Centraliser la gestion des identités et des accès aux services cloud (SSO ex)
- Mesurer et contrôler l'utilisation d'applications SaaS avec une approche orientée « risque » et une vision « cockpit centralisé »
- Accompagner les utilisateurs à préférer des services cloud de confiance
- Conserver le contrôle et assurer la sécurité des données les plus sensibles via des techniques de chiffrement

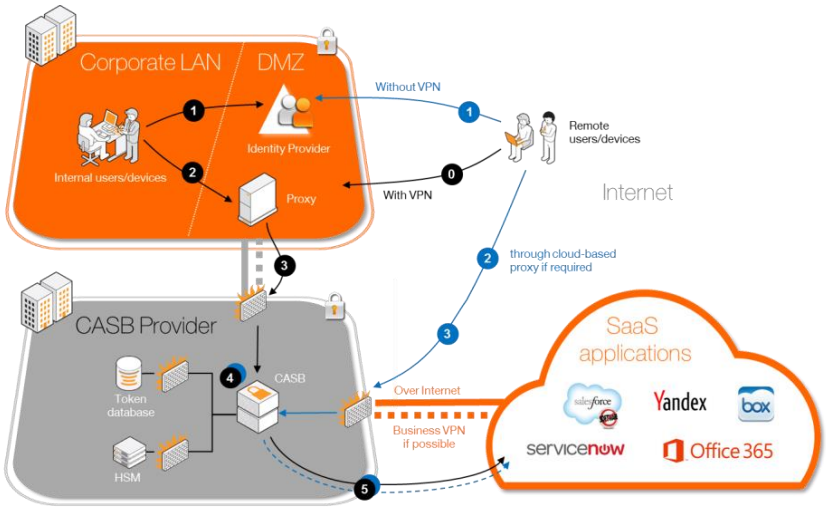



**Fig. 3.** : Sécurisation des usages SaaS avec un CASB

Le marché des fournisseurs de CASB est assez récent (4 à 5 ans) et en plein développement. Les acteurs sont relativement nombreux et ne répondent pas tous aux mêmes enjeux. Selon les fonctionnalités offertes, les applications SaaS couvertes varient de plusieurs milliers pour la découverte et le contrôle des usages SaaS à moins d'une dizaine pour la protection en confidentialité des données les plus sensibles. La conso-

lidation du marché s'est accélérée ces derniers mois avec le positionnement, la prise de participation ou l'acquisition de plusieurs start-ups par des géants tels que BlueCoat (Perspecsys), IBM, Microsoft (Alladom) ou encore Salesforce.com (Sky-High Networks).
Dans la suite de ce chapitre, seront approfondies les solutions permettant d'assurer la confidentialité des données sensibles manipulées par les applications SaaS ; fonction d'un CASB généralement dénommée « Protect » ou « Secure ».

La fonction « Protect » a pour objet de proposer à l'utilisateur une interface sécurisée d'une application SaaS. Il s'agit le plus souvent d'un(e) logiciel/appliance spécialisé(e) qui a deux fonctions principales :

- chiffrement / déchiffrement des données à destination / en provenance de l'application
- portage de certaines fonctionnalités de l'application SaaS afin de les rendre disponibles à l'utilisateur suite au chiffrement des données dans le cloud

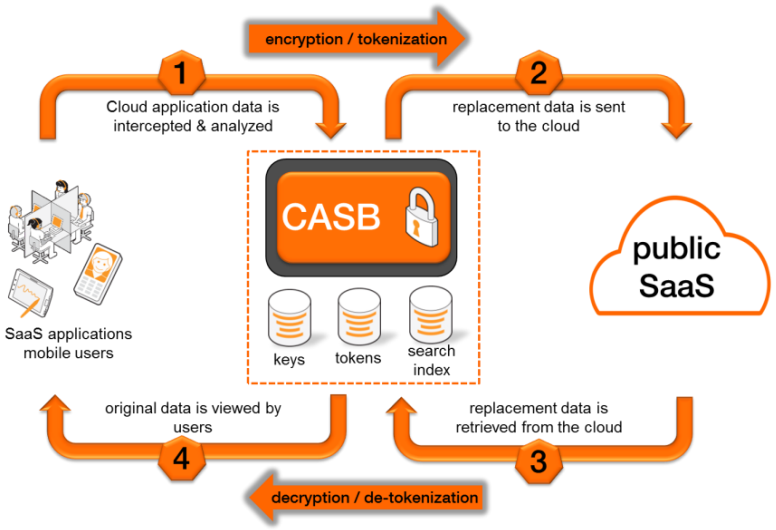



**Fig. 4.** : Aperçu de la fonction « Protect » du CASB

### 3.1 Modes de fonctionnement

Deux méthodes sont utilisées par les solutions du marché pour assurer la confidentialité des données stockées dans le cloud : la tokenisation et le chiffrement.

- La **tokenisation** consiste à faire une association entre un label – dont la nature peut être numérique ou alphanumérique – et un texte clair. Seuls les labels (tokens) sont envoyés vers l'application SaaS, les données (texte clair) restant confinées au sein du périmètre de confiance de l'entreprise.

L'ensemble des associations « tokens <=> données en clair » sont stockées dans une base de données qui peut être hébergée par l'entreprise ou chez un tiers de confiance.

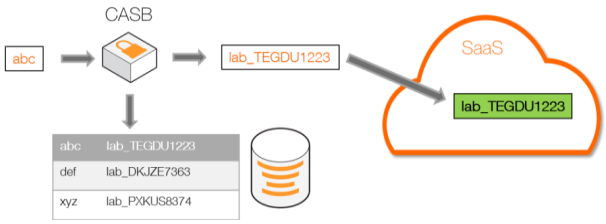



**Fig. 5.** : Principe de la tokenisation

- Le **chiffrement** propose de chiffrer / déchiffrer les données vers / depuis l'application et garantir ainsi la confidentialité des données. En général, la gestion des clés cryptographiques est réalisée par d'autres moyens sous contrôle de l'entreprise, tels que des boitiers HSM par exemple.


**Fig. 6.** : Principe du chiffrement

Les avantages / inconvénients de chaque mode de fonctionnement sont les suivants :

| | **Tokenisation** | **Chiffrement** |
|---|---|---|
| **Avantages** | ▪ Absence de tout lien algorithmique entre la donnée et son token relatif<br>▪ Maintien de la donnée dans le périmètre de confiance de l'utilisateur<br>▪ Conservation dans le temps ; pas de cryptanalyse possible<br>▪ Amélioration des performances et des coûts en ce qui concerne la gestion des fichiers | ▪ Souplesse d'utilisation<br>▪ Sécurité reposant la maitrise des clés cryptographiques<br>▪ Mécanisme stateless (sans état)<br>▪ Pas d'impact en cas d'évolution de l'architecture de déploiement |

| | | |
|---|---|---|
| **Inconvénients** | ▪ Base de données indispensable pour assurer la correspondance entre les tokens et leurs données associées<br>▪ Pas de réduction de l'empreinte du stockage sur le SI de l'utilisateur<br>▪ Mécanisme stateful (à états)<br>▪ Mécanismes de réplication bases de données nécessaire en cas d'architecture distribuée | ▪ Compromission théoriquement possible du mécanisme de chiffrement |

### 3.2 Scénarios de déploiement

Le CASB doit être mis en coupure des flux entre l'utilisateur et l'application SaaS. Deux architectures sont possibles :

- **« Forward-Proxy »** : L'utilisateur accède à l'application SaaS à travers le proxy web de l'entreprise qui se charge alors de relayer au CASB tous les flux à destination de l'application. L'URL d'accès à l'application reste inchangée.
- **« Reverse-Proxy »** : L'utilisateur accède à l'application SaaS à travers un reverse-proxy et donc une URL spécifique. Le proxy de l'entreprise traite les flux à destination de l'application SaaS comme toute autre application hébergée au sein de l'entreprise ou chez un tiers de confiance.

Dans les deux cas, le CASB doit déchiffrer le trafic HTTPS entre l'utilisateur et l'application SaaS pour protéger les données sensibles de l'entreprise. Ce processus complexifie l'architecture d'accès à l'application SaaS et déporte la sécurisation du canal de communication avec le fournisseur SaaS sur le CASB. Les recommandations de sécurité de l'ANSSI concernant l'analyse des flux HTTPS doivent être prises en compte dans le cadre du déploiement du CASB afin de garantir le niveau de sécurité du tunnel TLS et l'expérience utilisateur.

Les avantages / inconvénients de chaque architecture sont résumés ci-dessous :

| | **Forward-Proxy** | **Reverse-Proxy** |
|---|---|---|
| **Avantages** | ▪ L'expérience utilisateur est inchangée<br>▪ La configuration du terminal de l'utilisateur est inchangée | ▪ L'expérience utilisateur est la même quelle que soit sa localisation et son terminal (pro/perso)<br>▪ La configuration du terminal |

| | Forward-Proxy | Reverse-Proxy |
|---|---|---|
| | | de l'utilisateur est inchangée<br>▪ Une connexion VPN n'est pas nécessaire pour accéder à l'application SaaS depuis Internet<br>▪ La configuration du proxy de l'entreprise est simple |
| **Inconvénients** | ▪ La configuration du proxy peut être complexe<br>▪ L'utilisateur final doit passer systématiquement par le proxy de l'entreprise pour accéder à l'application SaaS<br>▪ Certaines applications mobiles ne sont pas supportées | ▪ L'URL d'accès à l'application est spécifique à l'entreprise<br>▪ La configuration de certains composants du SI peut être complexe (reverse proxy, authentification/SSO)<br>▪ L'utilisateur peut se connecter directement à l'application SaaS via son URL d'origine |

Un CASB peut être déployé « on-premise », autrement dit au sein du SI de l'entreprise, chez un tiers de confiance tel qu'un MSSP ou encore chez un fournisseur de services de sécurité dans le cloud (SECaaS).

- **On-premise :** La solution est déployée au sein du périmètre de confiance de l'utilisateur. Il s'agit d'un positionnement en coupure entre l'application SaaS et les utilisateurs de cette application.
- **Trusted Partner :** La solution est gérée / hébergée par un tiers de confiance, proposant le service à l'utilisateur. Les requêtes à destination de l'application SaaS sont ainsi redirigées vers ce dernier via Internet ou un réseau privé opérateur qui permet de garantir la confidentialité des échanges mais aussi les performances et la disponibilité de l'application SaaS. Le tiers de confiance est en charge de protéger les données en confidentialité entre lui et l'application SaaS.
- **Public cloud :** La solution est directement gérée / hébergée soit par le fournisseur de CASB, soit soit par le fournisseur de l'application SaaS qui cumule alors dans ce modèle les fonctions d'hébergeur de l'application et de tiers de confiance.

### 3.3 Contribution à la résilience d'une application SaaS

D'un point de vue utilisateur, les entreprises attendent d'une telle solution qu'elle reste transparente et sans impact sur les fonctionnalités (indexation/recherche, reporting, mobilité, collaboration externe…), les performances et la disponibilité de l'application SaaS. La réversibilité des données protégées doit aussi être garantie.

Contrairement aux fonctions de découverte (Discover) ou de contrôle des usages SaaS (Analyze), la fonction « Protect » d'un CASB propose un principe de sécurité véritablement actif, à même de modifier les données de l'utilisateur à la volée afin d'en garantir la confidentialité et donc de jouer sur la résilience d'une application SaaS.
L'utilisation d'un CASB et plus précisément de la fonction « Protect » sur une application SaaS permet de renforcer la sécurité et lutter efficacement contre les attaques visant la confidentialité et/ou l'intégrité des données sensibles dont pourrait être victimes les cloud service providers (CSP). En effet, les données sensibles stockées dans le cloud sont chiffrées par une tierce partie. Elles sont illisibles sans passer par le CASB. Certaines solutions du marché disposent en complément d'un mécanisme de contrôle d'intégrité des données stockées dans le cloud.

# 4 Limites des solutions de type CASB

Le CASB semble être la solution la plus appropriée pour traiter des problèmes d'intégrité et de confidentialité des données de l'entreprise utilisant les applications SaaS. Toutefois, cette solution peut engendrer d'autres difficultés.

## 4.1 Portage des fonctionnalités de l'application SaaS

L'intérêt des CASB est la mise à disposition de moyens de protection adaptés à l'application à sécuriser, par le portage des fonctionnalités natives de l'application et la définition d'une politique de sécurité à un niveau granulaire.

Par exemple, un CASB permettant de protéger une application SaaS CRM offrira des fonctions de protection des fonctionnalités au travers des interfaces existantes SaaS telles que la création/modification/liste des clients ou encore les messages instantanés échangés entre les commerciaux…

Mais certaines fonctionnalités sont généralement plus difficiles à « porter » au niveau du CASB. C'est le cas des fonctions de reporting ou encore de prévisualisation pour lesquelles l'application SaaS a besoin de l'information stockée « en clair ». À partir du moment où les informations (chiffre d'affaire, fichier, …) sont chiffrés, la génération de rapport ou de miniatures ne fonctionne plus.

## 4.2 Gestion de la mobilité

Malgré les promesses des fournisseurs de solutions SaaS et CASB, l'usage en mobilité peut poser des difficultés, parmi lesquelles les deux cas de figures suivants :

1. Accès à l'application SaaS en dehors du périmètre de l'entreprise via le CASB
2. Authentification SSO des utilisateurs mobiles depuis l'extérieur du SI

Lorsque le CASB n'est pas directement joignable depuis Internet, son accès depuis l'extérieur du peut s'effectuer de plusieurs manières :

- par un accès VPN : l'utilisateur mobile monte un tunnel VPN depuis son mobile vers le SI de l'entreprise afin de pouvoir être authentifié et connecté au CASB, puis redirigé vers l'application SaaS. Dans ce cas de figure, les autres flux générés par l'utilisateur seront également envoyés vers l'entreprise notamment son usage Internet.
- par un APN[6] dédié : l'entreprise peut négocier un APN dédié avec son opérateur mobile ; ce qui permettra aux utilisateurs externes d'accéder au CASB depuis leur mobile en passant par cet APN dédié.

L'implémentation du SSO très plébiscité par les utilisateurs, peut être difficile à mettre en place. Il est important de vérifier la compatibilité entre le fournisseur d'identités interne à l'entreprise et l'application mobile du fournisseur de service SaaS. En l'absence d'interopérabilité, les utilisateurs externes ne pourront pas s'authentifier via l'annuaire de l'entreprise, et seront obligés d'utiliser des comptes locaux à l'application SaaS. Dans ce cas, l'entreprise doit alors faire le choix:

- d'interdire les usages mobiles depuis l'extérieur du SI
- autoriser un accès via le CASB à l'application SaaS qui effectuera elle-même l'authentification des utilisateurs mobiles. Les utilisateurs ne bénéficient plus du SSO dans ce cas.

### 4.3 Impact sur la résilience

Pour assurer la confidentialité et l'intégrité des données les plus sensibles, le CASB devient le point de passage obligé pour accéder à l'application SaaS car seul le CASB peut transformer les données en clair. La mise à disposition de ces données et plus généralement la disponibilité de l'application, ne dépend donc plus uniquement du fournisseur SaaS.

La résilience d'une application SaaS se trouve ainsi dépendante de la résilience du CASB et donc de sa capacité à continuer de fonctionner en cas d'agression, de panne ou de sollicitation extrême. La résilience du CASB dépend en premier lieu du choix du mode de fonctionnement (chiffrement, tokenisation) et de l'architecture de déploiement de la passerelle (forward/reverse proxy, on-premise, trusted partner, cloud).

### 4.4 Impact sur les performances

La fonction de transformation des données que ce soit en données lisibles ou en données illisibles, est portée par le CASB. L'exécution de cette fonction nécessite du temps, même si ce temps est estimé en microsecondes. Plus la quantité de données à protéger sera importante et les requêtes de lecture/écriture seront nombreuses, plus la

[6] APN : Access Point Name

latence pourra être perceptible par les utilisateurs finaux. La localisation géographique du CASB par rapport aux utilisateurs est également importante.

### 4.5 Nombre restreint d'applications compatibles

Si un CASB est considéré comme une passerelle de protection des données vers une application SaaS, il est important que le CASB sache « porter » les fonctionnalités de l'application SaaS ; ceci afin de protéger les données manipulées (nom, prénom, adresse, téléphone…) par ces fonctionnalités (création d'une fiche client par exemple).

Aucun éditeur de CASB n'est en mesure de supporter TOUTES les applications SaaS. Il peut être nécessaire de choisir différents CASB pour protéger différentes applications.

### 4.6 Architecture multi-locataires

L'architecture multi-locataires (multi-tenant) permettant à plusieurs entreprises, ou à différentes filiales d'une entreprise d'utiliser le même CASB, n'est pas souvent proposée par les éditeurs. Pour une entreprise souhaitant offrir à chacune de ses filiales ou directions une autonomie en matière de sécurité des applications SaaS via l'usage d'un CASB, il est nécessaire :

- de pouvoir isoler les bases de données de tokens entre elles : les tokens de la direction financière ne doivent pas être mélangés avec les tokens de la direction marketing par exemple
- de pouvoir séparer les politiques de sécurité des données : pour la direction informatique, les clients sont internes, et il n'est pas besoin de chiffrer les informations les concernant ; ce qui n'est pas le cas pour la direction de vente qui aimerait protéger toutes les informations relatives à leurs clients par exemple
- Les clés de chiffrement utilisées doivent pouvoir être différentes selon les filiales ou les directions.

De même, pour un éditeur de CASB proposant la fonction « Protect », il n'est pas aisé d'utiliser un seul CASB pour gérer toutes les applications supportées. Aujourd'hui peu de solutions permettent de faire du « multi-applications », « multi-clients » sur une seule instance de CASB. Les entreprises sont donc obligées aujourd'hui de démultiplier les instances de CASB.

### 4.7 Gestion des secrets

Quelle que soit l'option choisie pour protéger les données, la gestion des secrets reste un point important pour assurer la confidentialité et l'intégrité des données. Dans le

cas du chiffrement des données, le problème de gestion des clés se pose : cycle de vie (création, répudiation, …), algorithme de chiffrement utilisé, stockage des clés…
Dans le cas de la tokenisation, la sécurité des données de l'application repose intégralement sur le processus de génération des tokens (séquentiel, aléatoire), la portée des tokens et la protection de la base de données de tokens.

### 4.8 Niveau de maturité des solutions

Force est de constater que les solutions de CASB proposées sur le marché ne sont pas toujours suffisamment matures aujourd'hui. Ce fait est également confirmé par des retours utilisateurs métiers. Ainsi, il est courant de rencontrer des problèmes :

- de dimensionnement : capacité à monter en charge
- de dysfonctionnement (bug)
- de non séparation des rôles

Les CASB répondent à un besoin certain, mais sont appelées à s'améliorer et à gagner en maturité.

### 4.9 Risques induits par ces solutions

Les résultats de plusieurs analyses de risque portant sur ce thème révèlent que même si le niveau de risques diminue avec l'introduction du CASB sur le périmètre d'une application SaaS, de nouveaux risques apparaissaient en même temps.

- **Point de défaillance unique** : La CASB est un point de passage obligé pour celui qui veut lire les données protégées ou écrire des données de manière protégée. Par conséquent, en cas de défaillance du CASB, les données protégées ne peuvent plus être lues et l'application devient indisponible.
- **Divulgation des secrets :** La divulgation des secrets notamment des clés de déchiffrement permettrait à un utilisateur non autorisé ayant accès aux données chiffrées de pouvoir tenter de les déchiffrer.

**Divulgation des données :** L'architecture d'accès à l'application SaaS est modifiée et la sécurisation du canal de communication avec le fournisseur est déportée sur le CASB. Des données normalement chiffrées sont présentes en clair au niveau du CASB. Si ce dernier est compromis, des informations et pas uniquement les plus sensibles peuvent être exposées.

- **Acceptation des utilisateurs :** La réussite d'un projet d'intégration de CASB est soumise à son adoption par les utilisateurs, mais aussi à la mise en place d'une politique de sécurité contraignant l'accès par le CASB. Lorsque le choix est laissé aux utilisateurs de passer ou non par le CASB pour utiliser l'application SaaS, l'accès direct (sans passer par le CASB) est souvent privilégié.
- **Gestion des rôles :** L'absence de séparation de rôles d'administration sur une solution de CASB peut entrainer des dérives qui permettraient à un ad-

ministrateur de consulter voire de modifier des données de l'application SaaS. Le rôle « Administrateur » du CASB ne devrait pas permettre un accès aux données en clair. De même, il serait nécessaire d'avoir un rôle « Responsable de la politique de sécurité des données » de manière à ce qu'un « Administrateur » mal intentionné ne puisse pas modifier la politique pour que toutes les données soient en clair.

- **Suivi des releases/partenariat avec les fournisseurs SaaS** : Les évolutions chez les fournisseurs d'applications SaaS sont nombreuses et fréquentes. Elles doivent rapidement être intégrées par l'éditeur de CASB sans quoi, l'utilisateur final perd le bénéfice du choix d'une telle application et rejette de facto les mécanismes de protection des données sensibles. Il convient donc que l'éditeur de CASB dispose d'un partenariat fort avec le fournisseur SaaS, afin d'anticiper ou d'être rapidement informé de la publication de nouvelles fonctionnalités ou correctifs pour faire évoluer son produit.

# 5 Alternatives au CASB

Plusieurs alternatives au CASB peuvent être envisagées dans le but de maintenir à la fois le périmètre fonctionnel d'une application SaaS et la résilience que ce modèle amène, ainsi que la confidentialité des données sensibles appelées à être manipulées dans ce cadre. Trois d'entre elles sont explorées dans ce chapitre, sans prétendre à l'exhaustivité.

## 5.1 Assurance / Protection juridique

Faire appel à une protection juridique, telle qu'une assurance et / ou des clauses contractuelles engageant le fournisseur quant à ses obligations en terme de confidentialité des données semble une option intéressante. Selon PwC, le marché des assurances contre les menaces informatiques est appelé à tripler d'ici la fin de décennie, pour atteindre 7,5 Md$ en 2020.

Cela sous-entend donc que le préjudice potentiel de la compromission de l'information soit effectivement gérable par ce biais, en particulier via des compensations financières. Selon le dernier rapport de l'Institut Ponemon, le coût d'une donnée perdue est estimé en France à environ 165 €, un montant non négligeable qui peut recouvrir des réalités très variées : atteinte à la réputation, à la continuité de l'activité, perte de liquidités, etc.
Là encore, le problème ramène à l'analyse de risque, dont le résultat doit permettre d'apporter une réponse à cette question. Force est de constater que dans bien des cas, certaines de ces données sensibles relèvent de la stratégie de l'entreprise utilisatrice du service, de telle sorte que la compromission de ces dernières pourrait avoir un impact que ne pourrait pas compenser ce type d'approche, soit parce qu'elle met en cause la survie à court / moyen terme de l'entreprise, soit que les montants d'une telle compensation seraient trop complexes à évaluer suffisamment précisément.

Ceci ne doit pas empêcher de souscrire un certain nombre de garanties et d'engagements auprès du fournisseur, mais ces derniers ne sauraient être suffisants pour prétendre à devenir une approche alternative sérieuse au CASB dans un contexte impliquant des données stratégiques ou encore classifiées de défense.

### 5.2 Sécurisation des données déléguée à l'hébergeur

L'option qui consiste à confier à l'hébergeur / fournisseur la sécurité de ses données est probablement l'option qui revient sur la table des RSSI le plus fréquemment. L'idée est la suivante : le fournisseur étant l'entité qui maitrise complètement les éléments techniques de son application, il est en mesure de proposer des solutions de sécurité adaptées à cette dernière en tenant compte d'une politique de sécurité définie par l'entreprise, y compris du chiffrement avec une gestion de clés restant sous le contrôle de l'entreprise.

Il est important de noter ici que le fournisseur maitrisant tous les éléments de la chaine, il est également en mesure de mettre à disposition de son application des données de l'entreprise en clair lorsque certaines de ses fonctionnalités l'exigent, telles que la création de graphes, de tri, et autres rapports, garantissant ainsi un maintien complet du périmètre fonctionnel et ce malgré la mise en œuvre de ces fonctions de sécurité.

Sur les trois catégories de risques identifiées en début d'article, cette approche permet de proposer des mesures couvrant le risque lié aux autres utilisateurs / entreprises de l'application et ou hébergés au sein de la même infrastructure cloud. En aucun cas les menaces que font peser les Etats ou l'hébergeur lui-même ne sont couvertes par ce scénario. Cette possibilité pourra donc être retenue de manière efficace si l'analyse de risque montre que ces deux entités majeures ne constituent pas une source de menace crédible dans le contexte de l'entreprise.

Il est important de noter que le choix de la méthode de protection engage vraisemblablement l'entreprise pour une période significative, et en tous les cas ses données resteront accessibles à l'hébergeur durant une très longue période. La question de la source de menace est donc à évaluer en tenant compte de ce fait.

### 5.3 Délégation à un tiers de confiance

La troisième option envisagée consiste pour l'entreprise à faire une distinction claire entre le fournisseur de l'application et la surcouche de sécurité destinée à protéger la confidentialité des données sensibles.

Pour ce faire, il s'agit de faire appel à un tiers de confiance dont le rôle est de gérer les instances de CASB pour le compte de ses différents clients. Cette approche, sans être parfaite, résout plusieurs problèmes parmi ceux détaillés auparavant dans cet article.

- Ce tiers de confiance peut mutualiser son architecture entre différents clients. Cela lui permet d'atteindre une « masse critique » en termes de taille d'infrastructure CASB, ce qui autorise une meilleure répartition des coûts et un apport en termes de résilience, avec un effet de dilution du point de défaillance unique (SPOF) détaillé plus haut.
- Il devient ainsi possible de proposer plusieurs types de CASB permettant de mettre en œuvre cette approche pour plusieurs types d'applications. Les liens forts entre les éditeurs de CASB et les applications impliquent de recourir à plusieurs éditeurs pour un besoin plus large ; ce qui devient complexe à gérer pour l'entreprise.
- Pour peu que cet opérateur ait la taille critique et la couverture géographique suffisante, il serait en mesure de déployer des instances de CASB en adéquation avec les besoins de ses clients, notamment en termes de disponibilité et de performances
- L'indépendance de ce tiers de confiance vis-à-vis des fournisseurs / hébergeurs d'applications doit permettre de réaliser une certification portant sur la sécurité de son périmètre CASB, créant les conditions de confiance acceptables pour une entreprise souhaitant protéger ses données sensibles. Ainsi des critères tels que l'emplacement géographique, les procédures de sécurité mises en œuvre ou la politique de gestion des secrets pourraient être évalués et audités en suivant par exemple le référentiel cloud établi par l'ANSSI.

Ce modèle ne résout pas le problème du maintien du périmètre fonctionnel, lors de la manipulation de données chiffrées ou tokenisées. Il est clair que de ce point de vue, la proximité des éditeurs de CASB vis-à-vis des fournisseurs d'application, leur maturité et leur réactivité restent des points majeurs pour garantir le portage des fonctionnalités attendues par les entreprises.

# 6 Conclusion

L'informatique dans les nuages ou le « cloud computing » (cloud) est souvent considéré comme intrinsèquement résilient et apparait comme l'un des moyens d'assurer cette capacité à un meilleur coût.

Dans un contexte de forte croissance autour de l'utilisation des services cloud par les entreprises, l'ouverture du système d'information demande aujourd'hui aux DSI et RSSI de requalifier et d'évaluer en continu les risques auxquels s'expose leur organisation.

Le risque majeur concerne la confidentialité des données envoyées dans le cloud public. Les deux principales réponses pour mitiger ce risque sont:

- **qualifier la confiance de l'entreprise envers le fournisseur / hébergeur** de l'application, selon des critères qui sont à définir dans son contexte propre. Ces critères doivent en effet tenir compte des besoins de sécurité des données à protéger, des sources de menaces considérées et des contraintes opérationnelles métiers de l'organisation.
- **protéger spécifiquement la confidentialité des données les plus sensibles**, sans pour autant renoncer au bénéfice de l'usage de l'application visée, par exemple en ayant recours à des mécanismes de chiffrement ou de tokenisation.

Sur ce dernier point, de nouveaux éditeurs mais aussi les acteurs historiques de la sécurité proposent désormais des passerelles de sécurité sous forme de solutions ou de services communément appelés CASB (Cloud Access Security Broker) ayant pour ambition de sécuriser les usages associés au SaaS et les données qui y sont envoyées.

La fonction « Protect » d'un CASB propose un principe de sécurité véritablement actif, placée en coupure entre l'utilisateur et l'application SaaS et à même de chiffrer/déchiffrer les données les plus sensibles à la volée afin d'en garantir la confidentialité. Elle permet de renforcer la sécurité et lutter efficacement contre les attaques visant la confidentialité et/ou l'intégrité des données sensibles dont pourrait être victimes les fournisseurs SaaS.

A l'inverse, l'introduction d'une telle fonction de chiffrement se traduit mécaniquement par une baisse de la disponibilité et de la résilience de l'application SaaS. Le recours à des moyens de chiffrement ou de tokenisation peut avoir un impact sur les fonctionnalités de l'application. L'architecture de déploiement du CASB influence aussi les capacités de résilience. La délégation à un tiers de confiance semble être le meilleur compromis.

Des réponses d'ordre juridique ou encore assurantiel (souscription d'une couverture cyber) peuvent aussi être envisagées dans le but de maintenir à la fois le périmètre fonctionnel d'une application SaaS et la résilience que ce modèle amène. Elles ne sauraient être suffisantes pour prétendre à devenir des approches alternatives sérieuses au CASB dans un contexte impliquant des données stratégiques ou encore classifiées de défense.

Ce retour d'expérience montre qu'un choix guidé par un objectif de résilience ne peut faire l'économie d'une réflexion approfondie quant aux conséquences en matière de confidentialité et d'intégrité de l'information. La sécurité dans le cloud n'y fait pas exception. Les solutions possibles devront donc nécessairement faire l'objet de compromis spécifiques au contexte de l'utilisateur, tenant compte de ses besoins opérationnels, métiers et de sécurité propres.

# Le routage, talon d'Achille des réseaux

Valentin Allaire (Orange France),
Sarah Nataf, (Orange France),
Pascal Nourry, (Orange France)
valentin.allaire@orange.com,
sarah.nataf@orange.com,
pascal.nourry@orange.com

**Résumé** L'Internet est constitué d'une multitude de réseaux qui échangent des « routes » pour joindre des adresses IP. Le protocole à la base de ces échanges (BGP – Border Gateway Protocol) ne comporte pourtant aucun mécanisme de sécurité. La version 4 de BGP a été définie en 1995 ; or depuis 1997, des erreurs de configuration ou des erreurs d'implémentation ont ainsi mis à mal la résilience de l'Internet. Des attaques informatiques exploitant ces vulnérabilités ont aussi été identifiées dès 2008. La présente communication propose dans un premier temps de rappeler les incidents significatifs rencontrés sur Internet. Elle présente ensuite les mesures usuellement prises « localement » sur un réseau d'opérateur afin de se prémunir de ces menaces et indirectement d'assurer la résilience locale du réseau. Elle aborde enfin les solutions techniques actuellement proposées à l'image de RPKI.

**Mots clés :** routeurs, IP, BGP, Hijacking, RPKI

## L'Internet, une multitude réseaux autonomes

L'Internet est composé d'une multitude de réseaux autonomes (AS ou *Autonomous System*) qui échangent entre eux les adresses connues et les routes associées pour joindre ces n. Le protocole utilisé par les réseaux pour échanger ces routes est BGP ([RFC4271]), qui est un protocole à vecteur de chemin.
L'exemple ci-dessous explique comment l'opérateur Canadien Eastlink identifie les routes possibles pour joindre les clients d'Orange France.

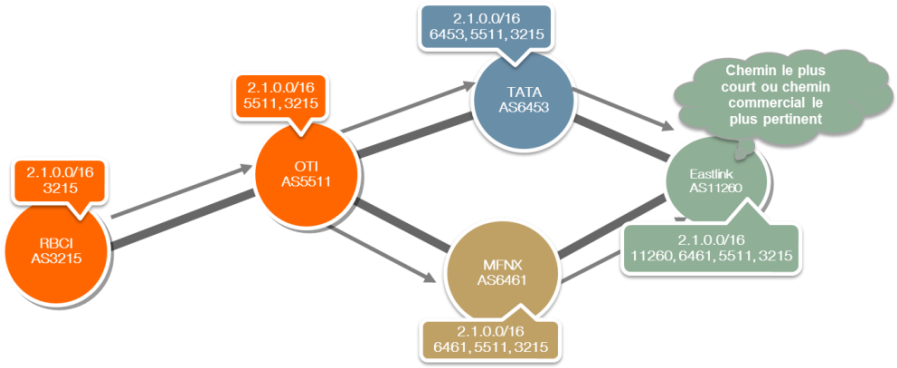



**Figure 1 : Annonces BGP entre réseaux**

Le réseau Orange France – RBCI (identifié par le numéro d'AS 3215) annonce à son transitaire international OTI (identifié par le numéro d'AS 5511) le bloc d'adresses IPv4 2.1.0.0/16 en précisant être à l'origine de l'annonce. OTI annonce ensuite à ses voisins TATA (identifié par l'AS 6453) et MFNX (identifié par l'AS 6461) le bloc d'adresses 2.1.0.0/16 en précisant que le réseau à l'origine de l'annonce est l'AS 3215 et qu'il est possible de passer par lui pour joindre ces adresses. TATA annonce à ses voisins, notamment Eastlink (identifié par l'AS 11260), le bloc d'adresses 2.1.0.0/16 en précisant que le réseau à l'origine de l'annonce est l'AS 3215 et qu'il est possible de passer par OTI (AS 5511) et par TATA (AS 6453) pour joindre ces adresses. MFNX fait de même vis-à-vis de Eastlink.

Ainsi Eastlink sait que pour joindre l'adresse 2.1.0.1/32, il doit passer par l'AS6461 ou l'AS6453, puis par l'AS5511 puis par l'AS3215 qui est à l'origine de l'annonce. Ce chemin s'appelle l'AS Path.

Le choix d'Eastlink entre les routes proposées par TATA et par MFNX se fait selon la politique de routage définie par l'opérateur :

- sur les bases commerciales négociées (coût au Gbit/s, capacité déployée, qualité du réseau, etc.), Eastlink peut par exemple systématiquement privilégier TATA et n'utiliser MFNX qu'en secours.
- sur des bases techniques :
    - Les annonces correspondant à des préfixes les plus spécifiques (e.g. plus petit bloc d'adresses annoncé) sont prioritaires sur les annonces correspondant à un préfixe moins spécifique (e.g. plus gros bloc d'adresses annoncé englobant le pool précédant). Ainsi, pour joindre l'adresse 2.1.3.3/32, la route sera celle choisie pour le préfixe 2.1.3.0/24 et non celle du préfixe 2.1.0.0/16, moins spécifique. L'annonce la plus spécifique usuellement utilisée sur l'Internet IPv4 est le /24 (correspondant à 256 adresses).
    - Les annonces ayant un chemin plus court (e.g. transitant par le moins de réseau) sont prioritaires sur les annonces ayant un chemin plus long.

Il est déjà intéressant de noter à ce stade qu'en cas de défaillance de l'intégralité du réseau MFNX, Eastlink peut basculer sur TATA pour acheminer les paquets IP vers le préfixe 2.1.0.0/16, et inversement, en cas de défaillance de l'intégralité du réseau de TATA, Eastlink peut basculer sur MFNX pour acheminer les paquets IP vers ce même préfixe. Il s'agit d'une caractéristique de l'Internet qui contribue fortement à sa résilience.

En zoomant sur l'interconnexion entre les AS 3215 et 5511, des mécanismes de redondance sont présents. Plusieurs routeurs du RBCI annoncent les mêmes routes vers plusieurs routeurs d'OTI et réciproquement, permettant de couvrir les cas communs de pannes simples.

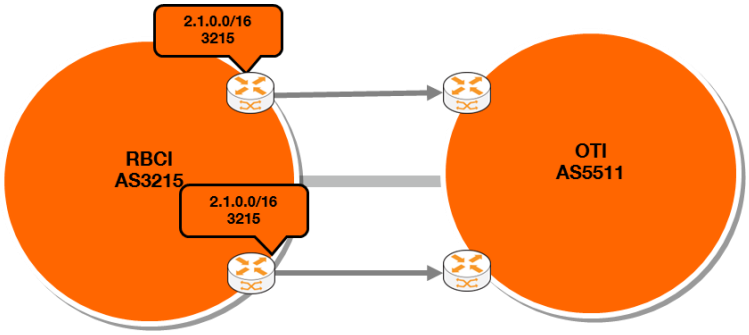



**Figure 2 : redondance entre deux réseaux**

## Le talon d'Achille de l'Internet

La section précédente a introduit la notion de chemin pour joindre une adresse IP. Le protocole BGP ne prévoit aucun mécanisme de contrôle de ce chemin, ce qui ouvre la porte à toute une série de vulnérabilités qui nuit à la résilience de l'Internet.
Les occurrences connues sont généralement liées à des erreurs de configuration à l'image du célèbre incident créé par Pakistan Telecom le 24 février 2008 (voir [RIPE] et [DYN1]). Avant l'incident, Youtube (AS 36561) annonçait le bloc 208.65.152.0/22. Sur ordre du gouvernement pakistanais, le fournisseur Pakistan Telecom (AS 17557) a annoncé à son transitaire, PCCW (AS 3491), un préfixe plus spécifique (208.65.153.0/24) que ceux annoncés par YouTube à 18h47 (UTC). Le transitaire n'ayant pas mis en place de filtre sur les annonces de son client, la totalité du trafic à destination de ce préfixe de YouTube a ainsi été redirigé chez Pakistan Telecom, rendant indisponible Youtube sur l'ensemble de l'Internet.

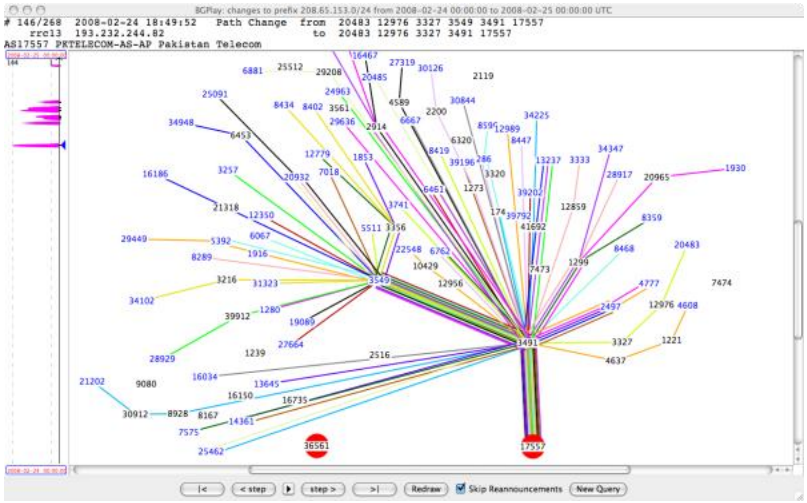



**Figure 3 : Erreur Pakistan Telecom en 2008 (source [RIPE])**

Il faudra attendre 20h07 (UTC) pour avoir une réaction de Youtube. Youtube a envoyé une annonce concurrente en /24 (le chemin le plus court était alors choisi) puis deux annonces /25 afin de devenir plus attrayantes que l'annonce erronée. Le transitaire PCCW a commencé à réagir vers 20h50 en allongeant artificiellement les routes vers Pakistan Telecom puis en supprimant toutes les routes annoncées par Pakistan Telecom. L'incident sera clos vers 21h00 (UTC).

Un autre exemple d'erreur concerne l'annonce par un réseau vers son transitaire d'une partie de sa table de routage en se plaçant comme AS de transit, à l'image d'un incident qui s'est déroulé le 12 juin 2015 (voir [BGPMon1]). Telekom Malaysia (AS 4788) a annoncé à son transitaire Level3 (AS 3549) 179 000 blocs d'adresses IPv4. L'incident qui a duré près de deux heures a impacté Level3 et ses clients (y compris en France).

Toujours dans le registre d'erreurs dans le chemin d'AS, deux incidents sont venus rappeler la fragilité des implémentations du protocole BGP dans les routeurs :

- Le 16 février 2009, un opérateur Tchèque Supronet (AS47868) a annoncé ses routes avec un attribut AS Path particulièrement long pendant une heure (voir [ARBOR] ou [DYN2]). L'opérateur a utilisé la technique dite de « Prepend » afin que cette route ne soit pas prioritaire en cas d'annonce concurrente. La propagation de cette annonce a eu des effets de bord jusqu'alors jamais constaté sur certains routeurs d'un constructeur très répandu sur l'Internet, créant une instabilité chez un très grand nombre d'opérateurs, et ce jusqu'à ce que Supronet cesse d'émettre l'annonce.
- En décembre 2011, plusieurs constructeurs de routeurs se sont rendu compte que la présence de l'AS0 dans l'ASPath (annonce erronée) pouvait conduire à un dysfonctionnement du routeur et clore la session BGP [IETF_AS0].

Au-delà de ces erreurs, des évènements suspects attirent l'attention des opérateurs réseaux et des entreprises connectées sur l'Internet.

Il s'agit tout d'abord de s'approprier temporairement des blocs d'adresses IP.
Ainsi début 2014, des chercheurs de Dell SecureWorks Counter Threat Unit ont identifié une série d'annonces erronées qui se focalisaient sur des blocs d'adresses utilisés pour le paiement en Bitcoin (voir [DELL]). Une personne manifestement malveillante a redirigé le trafic à destination de ces serveurs vers ses propres équipements probablement dans l'optique de voler des bitcoins.
Dans un autre registre, les spammers ont pris par exemple l'habitude de « squatter » des blocs d'adresses tournant afin de contourner les mécanismes de réputation. Il s'agit ici d'une usurpation d'identité afin de commettre des faits répréhensibles [BGPMon4].

Dans un tout autre registre, Alex Pilosov et Tony Kapela ont montré en 2008 [PK] qu'il était possible de lancer des attaques de type Man in the Middle en détournant du trafic via des annonces BGP spécifiquement générées.
Le détournement de BGP est entré depuis dans la boîte à outils des Etats, à l'image des pratiques révélées sur la société « Hacking Team » et les autorités italiennes (voir

[BGPMon3]). Des pratiques similaires existent probablement en Chine dans le contexte du « grand pare-feu » (voir [NetrSec]). Il s'agit ici clairement de techniques de type Man in the Middle. Il ne s'agit pas de cas isolés (voir [BGPMon2]).

## Contre-mesures élémentaires

L'ANSSI (Agence Nationale de la Sécurité des Systèmes d'Information) a publié un guide qui rappelle les bonnes pratiques en matière de configuration de BGP (voir [ANSSI]), ayant pour objectif d'améliorer la résilience des connexions BGP et donc des interconnexions sur l'Internet :

- Authentification des messages BGP afin de sécuriser les sessions.
- Filtrage des annonces BGP afin de limiter l'impact d'erreurs
    - Filtrage sur les préfixes réservés
    - Filtrage sur les préfixes attribués à un pair
    - Filtrage sur les préfixes trop spécifiques
    - Filtrage des routes par défaut
    - Suppression des numéros d'AS privés
    - Filtrage sur le nombre maximum de préfixes
    - Filtrage sur l'AS_PATH des routes annoncées par les pairs

Il s'agit d'un premier niveau de réponse intéressant.

Dans la même lignée, l'IETF a modifié le comportement attendu des routeurs en cas de réception d'une annonce BGP erronée (voir [RFC7606]) et expliciter l'usage de l'AS0 (voir [RFC7607]), toujours dans l'optique d'améliorer la résilience de l'Internet.

Un autre aspect concerne la supervision des annonces relatives aux blocs d'un opérateur donné afin de corriger en mode réactif une éventuelle « erreur ». Cette supervision peut être réalisée localement par l'opérateur lui-même à partir des annonces reçues par son réseau ou sous-traité à des sociétés qui proposent cette supervision avec une couverture mondiale. Il s'agit ici de surveiller les nouvelles annonces à partir de sondes BGP déployées sur plusieurs réseaux, à l'image du service proposé par BGPMon (voir [BGPMon5]). Lorsqu'un tiers annonce par erreur un bloc équivalent ou plus spécifique à un bloc annoncé légitimement par un réseau alors l'opérateur réseau reçoit une alerte. Il peut alors d'une part annoncer temporairement des annonces plus spécifiques ou équivalentes que l'annonce erronée afin de reprendre la main sur le trafic et/ou contacter l'exploitant du réseau ou ses transitaires pour demander de corriger l'erreur. Une illustration est la réaction de Youtube face à Pakistan Telecom en 2008 détaillée dans la section précédente.

## Vers une meilleure sécurité des échanges de route

L'une des réponses apportées en termes de standardisation à l'usurpation d'annonce BGP est RPKI (Resource Public Key Infrastructure). L'idée est de suivre les principes

d'affectation d'adresses et d'y associer une PKI afin de certifier l'association entre bloc d'adresses et AS d'origine (voir [RFC3779] et [RFC6480]).

L'IANA est chargé de la gestion de l'intégralité de l'espace adressable en IPv4 et en IPv6. L'IANA délègue à des RIR régionaux (le RIPE pour l'Europe) la gestion de sous-ensemble de cet espace adressable (généralement des /8). Le RIR lui-même délègue à des tiers, opérateurs, entreprises ou administrations, la gestion de sous bloc, ce tiers l'affectant directement ou indirectement à des clients.
RPKI propose donc de définir une architecture de gestion de clés fondée sur des entités de confiance (les Trust Anchor) qui vont venir par exemple générer les certificats des RIR qui eux-mêmes vont venir générer les certificats des opérateurs / entreprises / administrations, etc. (voir les standards [RFC6480] – [RFC6494], [RFC6810], [RFC6811], [RFC7128]).

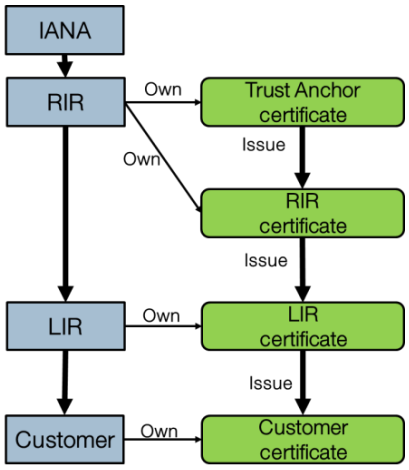



**Figure 4 : Certification RPKI**

Le standard RPKI vient ensuite définir des objets pour assurer l'intégrité des données, notamment l'association entre blocs d'adresses et AS origine :

- Les objets « manifest » donnent la liste exhaustive des objets générés par une entité ainsi que la localisation.
- Les objets « ROA » (Route Origin Authorization) permettent explicitement d'autoriser un AS à annoncer une liste de blocs d'adresses IP en étant origine de la route.
- Un certificat « End Entity » permet de vérifier la signature d'un seul objet (Manifest ou ROA) auquel il est lié dans un conteneur de type CMS (Cryptographic Message Syntax)

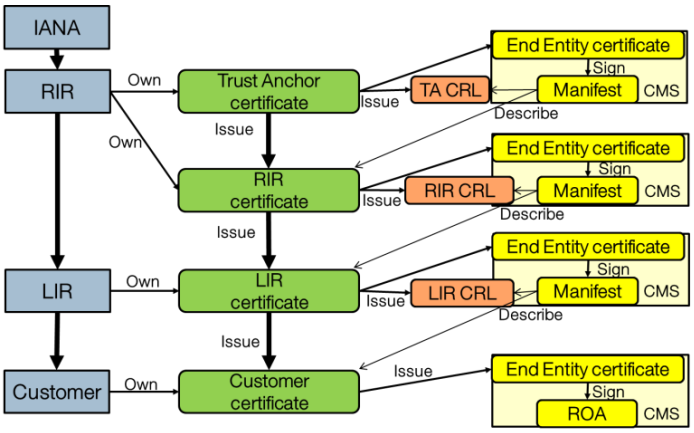



**Figure 5 : Vue globale RPKI**

Dans l'architecture RPKI, les routeurs n'ont pas à effectuer d'opérations cryptographiques. Une nouvelle interface vers un « serveur de validation RPKI » doit être créée. Ce serveur est chargé de contacter les « Trust Anchor » afin de récupérer l'ensemble des objets RPKI à jour et de vérifier l'intégrité des données (y compris en terme cryptographique) et de répondre aux demandes de validation de routes émises par les routeurs.

Le standard RPKI offre théoriquement plus d'avantages :

- RPKI permet de lutter contre les attaques et erreurs humaines reposant sur une origine d'annonce de route erronée.
- Les conflits entre AS annonçant le même préfixe spécifique (exemple /24 identiques) ne sont plus possibles.
- Le mécanisme est standardisé, testé et implémenté par de nombreux constructeurs (voir par exemple [ALU], [Cisco1] ou [Juniper]).
- La cryptographie apporte la preuve que l'annonce a été faite par son titulaire.
- L'utilisation d'une plateforme comme le RIPE pour la certification est sécurisée et automatisée, l'interface graphique simplifie la déclaration des objets ROAs.
- Même si toutes les routes annoncées sur l'Internet n'ont pas l'objet ROA, il est possible de moduler le comportement des routeurs en fonction de l'état de validation (route valide, route non valide et route inconnue) et d'ajouter des listes blanches.

Néanmoins, la situation est loin d'être idyllique.
Tout d'abord, la solution est complexe. Le standard « RPKI » est composé d'une vingtaine de RFC. Il introduit des contrats d'interface jusque-là inexistants ou embryonnaires à l'image de la relation atypique entre les routeurs et les serveurs de validation RPKI. Néanmoins, il est important de noter que ce type de contrat d'interface

IT-routeurs a vocation à se développer à l'image des travaux autour de la virtualisation de certains composants réseaux (voir [ONF]).
Ensuite, si de nombreux constructeurs ont implémenté RPKI, peu d'opérateurs ou d'entreprises ont entrepris de créer les objets RPKI sur leur domaine de responsabilité. Ainsi, actuellement seulement environ 5% (voir [NIST]) des blocs d'adresses annoncés sur Internet disposent d'un ROA valide. La croissance est faible et marquée par de fortes disparités géographiques.

% of Declared IPv4 Space Covered By ROAs

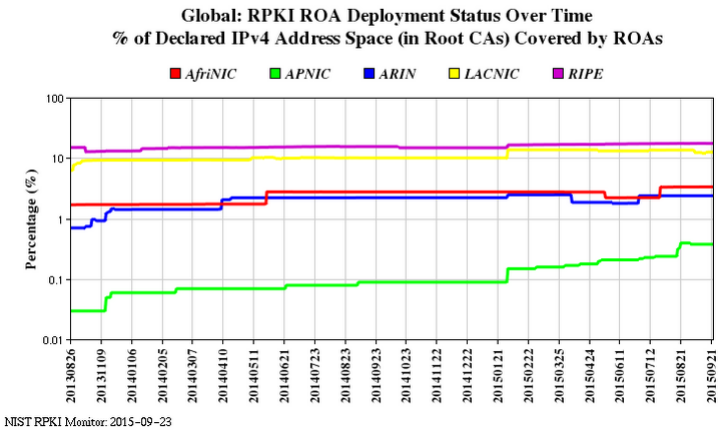



**Figure 6 : Adoption de RPKI au niveau mondial par RIR (source NIST- voir [NIST])**

Au vu du nombre de routes valides, il n'est clairement pas possible de n'autoriser que ces routes valides pour assurer l'acheminement des paquets sur l'ensemble des destinations de l'Internet. Il faut au minimum autoriser les routes inconnues voire les routes invalides ou déclarer une liste blanche de larges plages d'adresses. Il est important de noter que les principales sources d'invalidité sont (source [NIST]) :

- Une route est annoncée par le mauvais AS – environ 50% des routes invalides.
- Une route est annoncée par un AS légitime mais avec un mauvais préfixe (mauvais masque) – environ 50% des routes invalides.

Si un AS X référence correctement ses objets ROAs et active la validation des routes sur son réseau, cela ne garantira en rien le fait que des tiers usurpent ses préfixes avec des routes plus spécifiques sur des réseaux qui n'auraient pas activé RPKI. Pire, si le transitaire de l'AS X a activé la validation de RPKI en bloquant les routes invalides, l'AS X risque de ne pas pouvoir émettre de routes plus spécifiques pour lutter contre les routes non légitimes. L'AS X sera obligé dans un premier temps de modifier la taille du maxlength pour le positionner à 24 avant d'émettre les routes /24 en concurrence de la route illégale.

Un dernier point négatif majeur est que les ROAs ne valident que l'attribut AS origine de la route mais pas le chemin (AS-PATH). Un pirate peut très bien lancer des at-

taques de type Man In The Middle en modifiant l'AS_PATH sans en modifier l'AS origine. RPKI ne peut rien faire contre ce type d'attaque qui est décrit depuis 2008 (voir [PK]).

### Sécuriser un AS-Path

Plusieurs initiatives ont vu le jour pour tenter de sécuriser un AS-Path.
Le projet sBGP (secure-BGP, voir [SBGP]) a publié un draft à l'IETF en 1999 (v0) puis en 2003 (v1, voir [IETF_SBGP]) avant de tomber aux oubliettes.
SoBGP (Secure Origine BGP) est une initiative de Cisco lancée en 2003 puis publiée sous la forme d'un draft à l'IETF. La dernière mise à jour du draft date de 2006 (v2, voir [IETF_SoBGP]). Là-encore la solution proposée n'a pas trouvé son public.

### Une position pragmatique

Outre les bonnes pratiques proposées par l'ANSSI ([ANSSI]), une démarche pragmatique et relativement simple/rapide à mettre en œuvre pour pallier les lacunes du RPKI consisterait pour se défendre à automatiser la mise en œuvre de contre-annonces dès qu'une usurpation est détectée.
Au niveau de l'AS légitime, il s'agirait d'automatiser l'émission d'annonces plus spécifiques ou équivalentes dès la détection d'une annonce non légitime.
Au niveau des transitaires (notamment des transitaires internationaux de type Tier1), un service dédié pourrait être contractualisé avec les AS clients (au même titre que la détection et les contre-mesures DoS). Il s'agirait pour les blocs d'adresses de l'AS client légitime, d'automatiser l'émission d'annonces plus spécifiques ou équivalentes dès la détection d'une annonce non légitime provenant d'un peer. Afin de lutter plus efficacement contre l'annonce non légitime, l'ASPath pourrait être réduit au strict minimum avec uniquement le numéro d'AS du transitaire suivi de l'AS client qui peut légitimement annoncer le bloc en question.
Ce type de fonctionnalités pourrait être développé localement ou intégré dans les produits sur étagère déjà déployés par les transitaires :

- Les outils de détection d'attaques de dénis de service : ce type de produits a déjà un module d'analyse BGP pouvant permettre la détection d'évènements suspects et des capacités à émettre des annonces de façon automatisée afin de contrer une attaque DoS (BGP flowspec, Blackhole, ou routage du trafic vers un centre de nettoyage).
- Les outils dédiés à la supervision du plan de contrôle : ce type de produits a un module d'analyse BGP plus complet.

## Conclusion

Des solutions comme RPKI ou sBGP mettront dans le meilleur des cas du temps à s'imposer au niveau mondial. Il est nécessaire d'entrer dans une démarche volontariste ou chaque AS doit générer des ROA sur les blocs dont il a la responsabilité avant d'imaginer d'activer la validation de route avec destruction des routes invalides et/ou inconnues. Des solutions comme sBGP pourront venir en complément.

La mise en œuvre des règles d'hygiène BGP au niveau des points de peering peut déjà permettre de réduire la propagation d'annonces erronées dans certains cas de figure.

En complément, une démarche pragmatique qu'il conviendrait d'approfondir, consisterait à développer des contre-mesures automatisées afin de réduire l'impact d'annonces non légitimes. Les transitaires peuvent avoir un rôle particulier à jouer du fait des règles de priorisation des annonces qui privilégient les chemins les plus courts. Les éditeurs de logiciels peuvent également y voir une opportunité de développer les services proposés.

## Références


[ALU] Alcatel-Lucent ServiceRouter Release 12.0 R4, 7750 SR OS Routing Protocols Guide
https://infoproducts.alcatel-lucent.com/cgi-bin/dbaccessfilename.cgi/9300741102_V1_7750%20SR%20OS%20Routing%20Protocols%20Guide%2012.0.R4.pdf
[ANSSI] Bonnes pratiques de configuration BGP, septembre 2013
http://www.ssi.gouv.fr/uploads/IMG/pdf/guide_configuration_BGP.pdf
[ARBOR] Ahh, The Ease of Introducing Global Routing Instability, par Danny McPherson, 16/02/2009
https://asert.arbornetworks.com/ahh-the-ease-of-introducing-global-routing-instability/
[BGPMon1] Massive route leak causes Internet slowdown, par Andree Toonk, 12/06/2015
http://www.bgpmon.net/massive-route-leak-cause-internet-slowdown/
[BGPMon2] BGP routing incidents in 2014, malicious or not?, par Andree Toonk, 17/02/2015
http://www.bgpmon.net/bgp-routing-incidents-in-2014-malicious-or-not/
[BGPMon3] How Hacking Team Helped Italian Special Operations Group with BGP Routing Hijack, par Andree Toonk, 12/07/2015
https://www.bgpmon.net/how-hacking-team-helped-italian-special-operations-group-with-bgp-routing-hijack/
[BGPMon4] Using BGP data to find spammers, Andree Toonk 03/09/2014
http://www.bgpmon.net/using-bgp-data-to-find-spammers/
[BGPMon5] Portail commercial de BGPMon
http://www.bgpmon.net/
[Cisco1] IP Routing: BGP Configuration Guide, Cisco IOS XE Release 3S - BGP Origin AS Validation
http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_bgp/configuration/xe-3s/irg-xe-3s-book/irg-origin-as.pdf
[Cisco2] Securing BGP Through Secure Origin BGP, Russ White, Cisco Systems, septembre 2003
http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_6-3/securing_bgp_sobgp.html
[DELL] BGP Hijacking for Cryptocurrency Profit, par Pat Litke and Joe Stewart, 07/08/2014
http://www.secureworks.com/cyber-threat-intelligence/threats/bgp-hijacking-for-cryptocurrency-profit/
[DYN1] Pakistan hijacks YouTube, par Martin Brown, 24/02/2008
http://research.dyn.com/2008/02/pakistan-hijacks-youtube-1/
[DYN2] Reckless Driving on the Internet, par Earl Zmijewski, 16/02/2009
http://research.dyn.com/2009/02/the-flap-heard-around-the-world/
[IETF_AS0] Discussion on draft-ietf-idr-as0-00
https://www.ietf.org/mail-archive/web/idr/current/msg05816.html

[IETF_sBGP] Secure BGP (S-BGP), par Charles Lynn, Joanne Mikkelson et Karen Seo (BBN Technologies), juin 2003
https://tools.ietf.org/html/draft-clynn-s-bgp-protocol-01
[IETF_SoBGP] Architecture and Deployment Considerations for Secure Origin BGP (soBGP), par R. White (Cisco Systems), 15/06/2006
https://tools.ietf.org/html/draft-white-sobgp-architecture-02
[Juniper] Junos OS 12.3 - Example: Configuring Origin Validation for BGP
http://www.juniper.net/techpubs/en_US/junos12.3/topics/topic-map/bgp-origin-as-validation.html
[NetreSec] Analysis of Chinese MITM on Google, par Erik Hjelmvik, 04/09/2014
http://www.netresec.com/?page=Blog&month=2014-09&post=Analysis-of-Chinese-MITM-on-Google
[NIST] NIST – Advanced Network Technologies Division
http://rpki-monitor.antd.nist.gov/
[ONF] Open Networking Fundation
https://www.opennetworking.org/
[PK] Stealing The Internet : An Internet-Scale Man In The Middle Attack, par Alex Pilosov et Tony Kapela, Defcon2008
https://www.defcon.org/images/defcon-16/dc16-presentations/defcon-16-pilosov-kapela.pdf
[RIPE] YouTube Hijacking: A RIPE NCC RIS case study, 17/03/2008
https://www.ripe.net/publications/news/industry-developments/youtube-hijacking-a-ripe-ncc-ris-case-study
[RFC3779] X.509 Extensions for IP Addresses and AS Identifiers par M. Charles Lynn, M. Stephen Kent et M. Karen Seo, juin 2004
http://www.ietf.org/rfc/rfc3779.txt
[RFC4271] A Border Gateway Protocol 4 (BGP-4), par Y. Rekhter, T. Li, S. Hares, janvier 2006. La RFC4271 remplace les RFC initiales 1654 et 1771. Des mises à jour sont apportées dans les RFC6286, 6608 et 6793
https://www.ietf.org/rfc/rfc4271.txt
[RFC6480] An Infrastructure to Support Secure Internet Routing, par M. Lepinski et S. Kent (BBN Technologies), février 2012
http://www.rfc-editor.org/rfc/rfc6480.txt
[RFC6481] A Profile for Resource Certificate Repository Structure par M. Geoff Huston, M. Robert Loomans et M. George Michaelson, février 2012
http://www.rfc-editor.org/rfc/rfc6481.txt
[RFC6482] A Profile for Route Origin Authorizations (ROAs) par M. Matt Lepinski, M. Stephen Kent et M. Derrick Kong, février 2012
http://www.rfc-editor.org/rfc/rfc6482.txt
[RFC6483] Validation of Route Origination Using the Resource Certificate Public Key Infrastructure (PKI) and Route Origin Authorizations (ROAs) par M. Geoff Huston, et M. George Michaelson, février 2012
http://www.rfc-editor.org/rfc/rfc6483.txt
[RFC6484] Certificate Policy (CP) for the Resource Public Key Infrastructure (RPKI) par M. Stephen Kent, M. Derrick Kong, M. Karen Seo et M. Ronald Watro, février 2012
http://www.rfc-editor.org/rfc/rfc6484.txt
[RFC6485] The Profile for Algorithms and Key Sizes for Use in the Resource Public Key Infrastructure (RPKI) par M. Geoff Huston, février 2012
http://www.rfc-editor.org/rfc/rfc6485.txt
[RFC6486] Manifests for the Resource Public Key Infrastructure (RPKI) par M. Rob Austein, M. Geoff Huston, M. Stephen Kent et M. Matt Lepinski, février 2012
http://www.rfc-editor.org/rfc/rfc6486.txt

[RFC6487] A Profile for X.509 PKIX Resource Certificates par M. Geoff Huston, M.George Michaelson et M. Robert Loomans, février 2012
http://www.rfc-editor.org/rfc/rfc6487.txt
[RFC6488} Signed Object Template for the Resource Public Key Infrastructure (RPKI) par M. Matt Lepinski, M. Andrew Chi et M. Stephen Kent , février 2012
http://www.rfc-editor.org/rfc/rfc6488.txt
[RFC6489] Certification Authority (CA) Key Rollover in the Resource Public Key Infrastructure (RPKI) par M. Geoff Huston, M. George Michaelson et M. Stephen Kent, février 2012
http://www.rfc-editor.org/rfc/rfc6489.txt
[RFC6490] Resource Public Key Infrastructure (RPKI) Trust Anchor Locator par M. Geoff Huston, M. Samuel Weiler, M. George Michaelson et M. Stephen Kent, février 2012
http://www.rfc-editor.org/rfc/rfc6490.txt
[RFC6491] Resource Public Key Infrastructure (RPKI) Objects Issued by IANA par M. Terry Manderson, M. Leo Vegoda et M. Steve Kent, février 2012
http://www.rfc-editor.org/rfc/rfc6491.txt
[RFC6492] A Protocol for Provisioning Resource Certificates par M. Geoff Huston, M. Robert Loomans, M. Byron Ellacott et M. Rob Austein, février 2012
http://www.rfc-editor.org/rfc/rfc6492.txt
[RFC6493] The Resource Public Key Infrastructure (RPKI) Ghostbusters Record par M. Randy Bush, février 2012
http://www.rfc-editor.org/rfc/rfc6493.txt
[RFC6494] Certificate Profile and Certificate Management for SEcure Neighbor Discovery (SEND) par M. Roque Gagliano, M. Suresh Krishnan et M. Ana Kukec, février 2012
http://www.rfc-editor.org/rfc/rfc6494.txt
[RFC6810] The Resource Public Key Infrastructure (RPKI) to Router Protocol par M. Randy Bush et M. Rob Austein, janvier 2013
http://www.rfc-editor.org/rfc/rfc6810.txt
[RFC6811] BGP Prefix Origin Validation par M. Pradosh Mohapatra, M. Pradosh Mohapatra, M. David Ward, M. Randy Bush et M. Rob Austein, janvier 2013
http://www.rfc-editor.org/rfc/rfc6811.txt
[RFC7128] Resource Public Key Infrastructure (RPKI) Router Implementation Report par M. Randy Bush, M. Rob Austein, M. Keyur Patel, M. Hannes Gredler et M. Matthias Waehlisch, février 2014
http://www.rfc-editor.org/rfc/rfc7128.txt
[RFC7606] Revised Error Handling for BGP UPDATE Messages, E. Chen et K. Patel (Cisco Systems), J. Scudder (Juniper Networks), P. Mohapatra (Sproute Networks), août 2015
http://www.rfc-editor.org/rfc/rfc7606.txt
[RFC7607] Codification of AS 0 Processing, W. Kumari et H. Schiller (Google), R. Bush (Internet Initiative Japan), K. Patel (Cisco Systems), août 2015
http://www.rfc-editor.org/rfc/rfc7607.txt
[SBGP] Secure BGP Project (S-BGP)
http://www.ir.bbn.com/sbgp/

# Le cas des opérations d'armement
# L'exemple de SCORPION

Lieutenant-colonel Régis DEMAIE,

Ecoles de Saint-Cyr Coëtquidan

regis.demaie@intradef.gouv.fr

Résumé : La résilience numérique s'inscrit dans le cadre plus large de la sécurité et de ses deux grands piliers de protection et de défense. C'est un sujet particulièrement sensible dans le domaine des systèmes d'armes qui sont très largement numérisés aujourd'hui et qui devront faire face à tous types d'agressions en conservant l'essentiel de leurs performances. Le cas de SCORPION, programme majeur de l'armée de Terre, est emblématique : tirant la leçon des dernières années, il intègre en effet les exigences de cybersécurité dès sa conception.

Mots-clés : cyber – sécurité – systèmes – armes – SCORPION

Références

RDIA-2011/005_RESILIENCE n° 202/DEF/CICDE/NP du 13 décembre 2011
DIA-3.40_CYBER n° 82/DEF/CICDE/DR du 28 mars 2014
IG n° 125/DEF/EMA/PLANS/COCA – n° 1516/DEF/DGA du 26 mars 2010
Directive n° 02/DEF/EMAT/PS/BPIL/NP du 8 janvier 2013

En vertu des textes qui régissent les opérations d'armement et en particulier l'instruction générale 125-1516, tout programme est conduit conjointement par les armées et par la direction générale de l'armement. Dans l'armée de Terre, l'expertise des opérations d'armement est confiée à un organisme qui agit au nom de l'Etat-major : la section technique de l'armée de Terre (STAT). Si le chef d'état-major conserve les prérogatives de l'adoption et de la mise en service opérationnel, toutes

les tâches de conception, puis d'évaluation et d'expérimentation sont confiées à ce pôle d'expertise.

Cela permet au responsable de la cybersécurité des opérations d'armement terrestres d'avoir une vision particulièrement large ; celle-ci passe par la plupart des matériels majeurs, allant du char LECLERC à l'hélicoptère TIGRE, couvrant l'ensemble des systèmes d'information opérationnelle et de renseignement, qui permettent de traiter des informations SECRET OTAN ou souveraines, et s'intéressant tout particulièrement aux équipements de demain comme les drones ou les robots.

Parmi les 400 opérations d'armement suivies par la STAT, plus de 70 sont numérisées, c'est-à-dire composées d'un système d'information électronique, qui aide généralement les autres composants à communiquer entre eux et les utilisateurs à disposer de données, *a minima* sur le fonctionnement du système et, le plus souvent, sur l'ennemi, le terrain ou les amis. Cette numérisation constitue un des avantages décisifs d'une armée moderne. En effet, ces traitements de données valorisent les capacités offertes, apportant la précision, la vitesse de calcul, la prise en compte de paramètres plus nombreux, et au final la mémoire et la synthèse des actions entreprises, qui sont autant de facteurs d'initiative et de suprématie pour les forces terrestres. Mais elles sont également porteuses de nouvelles vulnérabilités, contre lesquelles il est indispensable de se prémunir si l'on souhaite conserver les avantages qu'offrent ces armements modernes. C'est alors que se pose la question de la résilience de ces systèmes numériques, de leur capacité à encaisser accidents ou agressions, à s'en remettre et à fournir de nouveau le service attendu.

Aujourd'hui, la résilience s'inscrit dans le cadre plus large de la sécurité, pour laquelle une démarche complète doit être menée si l'on veut pouvoir résister aux menaces numériques qui pèsent sur les armements modernes.

Après avoir replacé la résilience numérique dans un contexte élargi où protection et défense se combinent pour atteindre un état de sécurité satisfaisant, seront détaillées les étapes majeures de la démarche de cybersécurité, qui constitue le niveau d'engagement nécessaire pour pouvoir accepter le risque face aux menaces numériques. Enfin, le programme SCORPION illustrera la prise en compte croissante de ce

sujet : il est en effet le premier programme majeur pour lequel l'effort est marqué dès la conception.

## 1/ La place de la résilience dans une sécurité partagée entre défense et protection

Il importe avant tout de revenir sur ces notions de résilience et de cybersécurité, qui sont des termes relativement nouveaux et qui pourraient paraître encore nébuleux. A cette fin, cet article s'appuiera sur les parutions doctrinales de ces dernières années, données en référence *supra*.

### 1.a. La résilience, notion nouvelle et complexe

Apparue en France en 2008 dans le domaine de la sécurité et de la défense (*Livre blanc sur la défense et la sécurité nationales*, 2008), la résilience se définit comme la volonté et la capacité d'un pays à résister aux conséquences d'une agression ou d'une catastrophe majeures, puis à rétablir rapidement sa capacité de fonctionner normalement, à tout le moins dans un mode socialement acceptable. Si on l'applique aux armées, la résilience apparaît non comme une qualité définitivement acquise mais comme le résultat d'un processus évolutif, complexe et difficile à évaluer, qu'il y a lieu d'envisager par phases et de traiter aux plans humain, organisationnel et technique.

Par phases, car être résilient, c'est successivement se préparer avant la crise, résister pendant celle-ci, se rétablir par la suite et se consolider dans le long terme. Par domaines également, qui vont de l'humain au technique en passant par l'organisation qui les fédère : si le premier est sensible aux traumatismes physiques ou psychologiques, le deuxième est menacé de dysfonctionnements ou de pannes, voire de neutralisations ou de destructions, ce qui rend la troisième susceptible de s'affaiblir à des degrés divers jusqu'à devenir partiellement ou totalement inopérante.

Enfin, la notion de résilience couvre les champs matériel et immatériel, associant capacité et volonté : la première repose sur des ressources, des compétences et une certaine liberté d'action, la seconde s'entend comme détermination à agir et à se mobiliser, tant sur le plan individuel que sur le plan collectif.

### 1.b. La nouvelle résilience inscrite dans le champ de l'ancienne sécurité

Si l'on s'intéresse maintenant à l'espace numérique, on va trouver toute une série de termes nantis du préfixe cyber, lui-même étant une abréviation de la cybernétique, cette science du contrôle des machines dont l'étymologie renvoie au gouvernail grec (Norbert WIENER, *Cybernétics or control and communication in the animal and the machine*, 1948 - *La cybernétique, information et régulation dans le vivant et la machine*, Seuil 2014). En dehors d'un usage parfois abusif en recherche d'un effet de futurisme, ce préfixe s'applique aujourd'hui au domaine des systèmes d'information électroniques. C'est dans ce cadre que la doctrine militaire a défini récemment la cyber-résilience, c'est-à-dire la résilience des systèmes numériques. Cette définition s'inscrit dans le champ plus large de la cybersécurité, qui va être abordée sous l'angle de trois grands volets.

En préalable, il est important de rappeler que le cyberespace, ou espace numérique, est aujourd'hui considéré comme un milieu à part entière, c'est-à-dire un lieu de confrontation comme la terre, la mer, l'air ou bien l'espace extra-atmosphérique. Les systèmes d'information étant désormais une donnée constitutive de nos sociétés, le cyberespace omniprésent devient un enjeu stratégique civil et militaire et le théâtre de diverses opérations qui s'y déroulent déjà depuis un certain temps.

La cyber-résilience est définie comme la capacité d'un système d'information à résister à un incident ou une agression et à revenir à son niveau de service initial par la suite ; il s'agit donc de la capacité à encaisser une panne ou une cyberattaque et à s'en relever. C'est une qualité d'un système ou d'une organisation que l'on assure par une protection et une défense adéquates.

La cyberdéfense pour sa part regroupe l'ensemble des actions menées dans le cyberespace en vue de détecter les cyberattaques et d'y réagir. Dans son volet militaire, la cyberdéfense ne s'interdit plus de mener des actions d'accompagnement des opérations militaires, c'est-à-dire de faire usage d'actions offensives si nécessaire, mais ce volet ne sera pas abordé dans le présent article.

La cyberprotection, enfin, recouvre les mesures techniques ou d'organisation qui permettent d'assurer le niveau attendu de confidentialité, d'intégrité et de disponibilité d'un système et des informations qu'il traite, stocke ou transmet. Cette dernière définition reprend donc celle qui était auparavant dévolue à la sécurité des systèmes d'information, ou encore à la sécurité des traitements automatisés de

données si l'on veut revenir à une plus vieille appellation en usage dans le ministère de la Défense.

La cybersécurité, qui regroupe la cyberprotection, la cyberdéfense et la cyber-résilience, se définit également comme l'état recherché pour un système d'information, lui permettant de résister aux événements issus du cyberespace et susceptibles de compromettre la confidentialité, l'intégrité ou la disponibilité des données traitées et des services assurés par ce système d'information.

La simple définition des trois domaines subordonnés permet de voir que le champ de la cybersécurité s'est largement étendu, et que le souci de « l'état de l'art » dont on parlait il y a une dizaine d'années a été remplacé par une formalisation plus précise d'un besoin de maintien en condition de sécurité pour tous nos systèmes. C'est un des rôles de la cyberdéfense d'aujourd'hui que d'y veiller.

Il sera encore intéressant de noter que la cyberprotection et la cyberdéfense disposent de moyens dédiés, en ressources tant humaines que techniques, alors que la cyber-résilience et la cybersécurité sont plutôt des qualités attendues d'un système et qu'elles ne seront véritablement mesurables qu'*in fine*, à l'expérience de l'épreuve. Jusqu'à ce que cette dernière advienne, les efforts porteront donc essentiellement sur la préparation à ce danger potentiel.

Il est parfois d'usage aujourd'hui de comparer le dispositif général de la cybersécurité à une force qui doit soutenir un siège. Cette force dispose de murailles, qui ont été construites par la cyberprotection. Or, on sait avec le célèbre adage qu'on attribue généralement à Thucydide que l'épaisseur d'une muraille compte moins que la volonté de la franchir. C'est pourquoi la force dispose également d'une capacité d'action mobile, représentée par la cyberdéfense. Ces deux actions sont complémentaires : une muraille est vite trouée de multiples brèches lorsque les engins de siège commencent à sévir. La force de réaction permet de combler ces brèches et d'empêcher des intrusions dans l'enceinte. Si l'on regarde maintenant le point de vue de cette force, elle préférera se battre derrière des murailles qu'en terrain découvert : car sans protection, elle ne pourrait tenir aussi longtemps que lorsqu'elle bénéficie de points d'appuis construits dans une architecture prévue à cet effet. Ainsi, protection et défense viennent se compléter mutuellement et l'on sent que la sécurité ne serait pas assurée sans la présence de ces deux composantes complémentaires. La cyber-résilience enfin, sera l'effet

obtenu lorsque l'on dispose d'enceintes successives et de forces de réserve, disponibles soit pour une défense en profondeur et un maintien à l'abri du cœur du dispositif, ou bien pour une contre-attaque et un rétablissement de la ligne de défense initiale.

Plus récemment, Howard Shrobe, membre de la *Defense Advanced Research Projects Agency*, sur la base d'une autre analogie, empruntée à la biologie, constatait les limites du système immunitaire inné et sa complémentarité avec le système immunitaire adaptatif. Cette réflexion avait pour objet d'orienter la construction d'une sécurité adaptative interne aux systèmes et aux applications.

Ces deux images métaphoriques illustrent bien le caractère toujours temporaire d'un bon état de sécurité, et le besoin d'un suivi constant pour maintenir cet état. C'est dans cet esprit que la réglementation en vigueur a été rédigée, puis déclinée, dans l'esprit si ce n'est à la lettre, pour les opérations d'armement terrestres.

## 2/ La cybersécurité des opérations d'armement terrestres

### 2.a. Des responsabilités partagées

Il s'agit maintenant de voir comment se déclinent ces principes dans les opérations d'armement. Ce domaine est régi par une instruction ministérielle : l'instruction générale 125-1516, cosignée par le chef d'état-major des armées (125/EMA) et par le délégué général de l'armement (1516/DGA). Elle découpe schématiquement le déroulement d'une opération d'armement en six stades : l'initialisation et l'orientation sont confiées aux armées, l'élaboration et la réalisation ressortissent à la DGA, à ces quatre étapes initiales de conception succèdent l'utilisation puis, bien plus tard, le retrait du service.

Si l'on adopte la vision des utilisateurs, celle des forces qui mettront en œuvre les équipements, on voit que l'armée exprime, au travers d'une fiche de caractéristiques militaires, le besoin d'un équipement, dans le cadre plus large d'une capacité, puis reçoit cet équipement et procède à une évaluation et à une expérimentation pour pouvoir successivement l'adopter puis le mettre en service opérationnel. Un des points majeurs de la nouvelle instruction générale par rapport à l'ancienne tient au volet du maintien en condition opérationnelle, qui est devenu incontournable dans l'étude d'un programme.

La DGA pour sa part est chargée de rédiger la spécification technique du besoin, qui consiste à transformer les besoins militaires en performances quantifiables et mesurables que l'on pourra consigner dans les cahiers des clauses particulières, préalables à la notification à la main d'œuvre industrielle d'un marché. C'est dans le cadre de ce marché que seront achetés les équipements, éventuellement précédés d'un prototype lorsque des essais s'imposent, en particulier lorsque l'on envisage la mise en œuvre d'une nouvelle technologie. La DGA tient donc un rôle intermédiaire majeur entre l'expression du besoin en amont et la réception de l'équipement.

La cybersécurité de son côté a longtemps souffert de son cantonnement à quelques experts et d'un vocabulaire propre qui ne la rendait pas aisément lisible de l'extérieur. Pourtant, son organisation est des plus classiques et on peut aisément la mettre en parallèle avec les textes qui régissent les conduites de projets. Elle exprime un besoin par le biais d'une fiche d'expression rationnelle d'objectifs de sécurité (FEROS), elle procède également à des étapes de qualification, d'évaluation et d'audit, elle prononce des autorisations d'emploi qui prennent le nom d'homologation de sécurité ; et elle s'est organisée pour que l'état de sécurité puisse durer, en formalisant le besoin d'un maintien en condition de sécurité qui consiste à mettre très régulièrement à jour tous les composants logiciels d'un programme.

Dans ces éléments, il y a deux points clés à retenir.

- Le besoin de cybersécurité doit être formalisé très peu de temps après le besoin fonctionnel, de manière à entrer lui aussi dans la spécification technique que la DGA exprimera vers la main d'œuvre industrielle. L'écriture d'une FEROS a posteriori demeure souvent sans effet concret, en particulier dans les programmes dont les budgets sont arrêtés.
- Le maintien en condition de sécurité est l'élément majeur qui permet de s'assurer que l'on dispose d'un système dont le niveau de sécurité est pérenne. C'est donc la condition *sine qua non* d'une homologation, il mérite à ce titre de recevoir une attention particulière. Il fait appel à la maîtrise technique car le correctif doit être combiné avec l'évolutif sous peine de négliger une partie importante des travaux à mener.

### 2.b. Une sensibilité croissante des armées à la cybersécurité

Au-delà du rappel de ces principes, il est intéressant d'observer la montée en puissance du sujet de la cybersécurité dans les armées depuis quelques années à peine. Plutôt relégué à un niveau d'importance secondaire et ignoré par la majorité des décideurs il y a encore quinze ans, il a connu un regain de faveur par étapes successives. Le cadre global est bien entendu lié à l'importance croissante des systèmes numériques déjà évoquée plus haut.

La première prise de conscience notable est venue avec le retour des armées françaises dans l'organisation du traité de l'Atlantique Nord (OTAN) au début des années 2000. Les anglo-saxons, qui participent grandement à l'édiction des normes de cette alliance, sont réputés plus sensibles au sujet de la sécurité de l'information que nos compatriotes. Cela se traduisait en particulier par une profusion des textes réglementaires sur le sujet et par leur application relativement plus rigoureuse dans les échelons subordonnés que ce n'était le cas en France. La cybersécurité était l'un des critères majeurs de contrôle pour qu'un état-major français soit qualifié selon la norme des états-majors de l'OTAN. La Marine et l'armée de l'Air en 2005, puis l'armée de Terre en 2007, se sont vu accorder la capacité opérationnelle de force de haute disponibilité (*high readiness force, full opérational capability*) à laquelle a succédé le statut de force de riposte de l'OTAN (*NATO response force*).

Le deuxième facteur majeur de prise de conscience est lié aux incidents de cybersécurité sur les réseaux. Ils représentaient désormais une gêne dans le travail quotidien de tout le ministère et n'étaient plus circonscrits à un niveau local. Les vers *Blaster* en 2003, et *Conficker* en 2008 ont autant touché la Défense que les particuliers et les entreprises, alors même que les réseaux de travail n'étaient pas connectés sur l'Internet. Les besoins d'entrée et de sortie de données en provenance d'autres sources et l'accès accordé à la majorité des agents, combiné à une faible sensibilisation à l'hygiène cybernétique, en bref la simple vie de ces réseaux, concourrait à les rendre tout aussi vulnérables que ceux de la société civile.

Paradoxalement, la troisième et plus importante des étapes de la prise de conscience est liée à un événement extérieur à la Défense et même à la France. Il s'agit de la découverte vers 2010 du virus *Stuxnet*, ou encore la fabrication par des organisations gouvernementales d'une attaque ciblée

contre les intérêts d'une autre nation. Bien que décrite depuis longtemps en théorie, cette hypothèse venait de sortir de la science-fiction dans laquelle on la cantonnait auparavant pour devenir une réalité particulièrement inquiétante. En effet, l'attaque non seulement combinait l'exploitation de plusieurs vulnérabilités inconnues auparavant, mais elle touchait aussi des systèmes industriels pour lesquels la question de la cybersécurité n'avait quasiment jamais effleuré les esprits. L'effet en a été sensible, tant sur la transformation et la montée en puissance des organisations, qui est toujours en cours, que sur l'étude des systèmes et tout particulièrement des programmes d'armement.

Les armées se sont dotées en 2012 d'une politique globale de cybersécurité, et cette dernière a été mieux intégrée dans les autres processus auxquels elle devait participer. Cela a été en particulier le cas dans l'armée de Terre dont la charte de fonctionnement de 2013 faisait enfin apparaître le sujet dans le processus de l'équipement, mené par le bureau programmes et systèmes d'armes. Désormais perçue comme une priorité, la cybersécurité a pu enfin progresser significativement et le taux de production documentaire à ce sujet a fait un bond remarquable, notamment au niveau des expressions de besoin. Ce besoin désormais exprimé débouchera ultérieurement sur un meilleur niveau de cybersécurité.

### 3/ Le cas de SCORPION

Le programme SCORPION constitue un cas d'école car il est l'un des premiers parmi les programmes majeurs à avoir suivi la démarche complète dans l'ordre où elle doit se dérouler.

#### 3.a. Un programme composite

Dans les années 1970, l'armée de Terre avait effectué des choix structurants décisifs sur le couple d'engins qui devaient doter le cœur des groupements tactiques interarmes médians : l'AMX10RC et l'ERC90, engins de combat à roues, légers, avec une puissance de feu significative, et le VAB, engin blindé à roues transporteur de troupes, de cellules de commandement ou de systèmes d'armes. Cette décision a pris tout son sens opérationnel en 1984, lorsqu'en pleine guerre froide, la France a créé la force d'action rapide, se dotant *de facto* d'un outil de combat fortement réactif, facilement projetable et employable. Depuis les années 1980, ces engins blindés ont pris part à tous les conflits et règlements de

crises, au Liban, en Irak, dans les Balkans, plus récemment en Côte d'Ivoire et aujourd'hui au Mali et dans le reste de la bande sahélo-saharienne. Conçus au début des années 1970, ces blindés ont connu, dans la limite de leurs capacités d'évolution, plusieurs cycles de rénovation et de modernisation, en près de 40 ans de vie opérationnelle. Mais l'ERC90 a simplement été remotorisé, l'AMX10RC a connu une rénovation limitée avec l'installation d'un système d'information, des modifications de suspension, et l'acquisition d'un nombre très réduit de kits de surprotection. Le VAB quant à lui a fortement évolué, grâce à une surprotection, à une mitrailleuse de calibre 12,7 télé-opérée et enfin à la « félinisation » ou adaptation au moderne fantassin à équipement et liaisons intégrés. Mais seuls 344 VAB sur les 1200 de l'infanterie, eux-mêmes extraits des 3000 VAB de l'armée de Terre, ont pu en bénéficier. Indispensables mais insuffisantes, toutes ces améliorations se sont faites au détriment de la mobilité et de l'autonomie. Les engins sont arrivés en limite mécanique, leurs coûts de maintien en condition devient préoccupant, leur disponibilité est même sous le seuil critique d'entraînement des unités. Toute nouvelle amélioration pour atténuer leur vulnérabilité est devenue impossible. Ainsi, à l'horizon 2020, le segment des blindés médians est à remplacer dans son ensemble.

C'est la vocation de SCORPION, dans une approche capacitaire globale et cohérente des performances de mobilité, de protection, d'autonomie et d'agression sous blindage. L'enjeu de SCORPION réside dans les choix d'homogénéité et de rationalisation des performances pour réduire de façon significative les coûts d'acquisition et de soutien, tout en conservant un emploi efficace et simple de ces équipements de combat. Outre la gestion des obsolescences du char LECLERC, SCORPION regroupe quelques opérations structurantes.

- En premier lieu, l'unique système d'information et de combat SCORPION (SICS) remplacera les six dispositifs actuels, différents, compliqués et restreints aujourd'hui essentiellement à l'usage d'une messagerie. Il s'agira d'un système d'information convivial, simple, qui permettra de libérer les hommes de tâches répétitives et fastidieuses, pour leur permettre de se concentrer uniquement sur la réflexion et l'action tactique ;

- ensuite, le JAGUAR sera l'unique engin blindé de reconnaissance et de combat, pour remplacer les ERC90, les AMX10RC et les VAB missiles HOT ;

- le GRIFFON sera un véhicule blindé multi-rôles, limité à six variantes au maximum pour tout le reste des groupements tactiques interarmes, soutien et appuis compris, contre plus de trente variantes de VAB aujourd'hui ;
- enfin, un autre véhicule blindé multi-rôles, plus léger, viendra équiper l'échelon national d'urgence. Il n'a pas encore reçu de baptême.

Le binôme JAGUAR-GRIFFON constitue bien le cœur de l'efficacité opérationnelle des groupements tactiques futurs et il est le pivot principal des économies du programme SCORPION.

**3.b. Les enjeux de l'infovalorisation**

La valorisation des capacités de combat par l'information mutuelle des divers acteurs en temps quasi-réel est un des objectifs majeurs de ce nouveau programme. C'est une amplification des effets qui étaient attendus de la numérisation de l'espace de bataille entamée dans la décennie précédente. Le système d'information unique SICS y contribuera évidemment. La cybersécurité est un élément important de sa définition, comme c'est le cas pour tout système d'information opérationnelle et de communication. Il devra bénéficier de la classification diffusion restreinte OTAN, qui rendra les informations traitées aisément échangeables avec nos alliés, tout en les protégeant au juste niveau de suffisance en raison de leur caractère tactique, circonscrit dans le temps et dans l'espace. Il devra garantir avant tout un haut niveau de disponibilité et d'intégrité, ce qui conférera à l'information traitée la fiabilité nécessaire pour que l'on atteigne la supériorité informationnelle sur l'adversaire et l'accélération des boucles de décision, qui favorisent la prise et la conservation de l'initiative dans le combat. Comme il y a maintenant une dizaine d'années que les armées savent homologuer de tels systèmes, ce dossier ne devrait pas constituer un défi particulier.

La réelle nouveauté se situe en fait dans les véhicules évoqués plus haut. En effet, dans ces plateformes de combat, non seulement toutes les fonctions sont numérisées, mais elles sont par ailleurs connectées entre elles au moyen d'une architecture vétronique SCORPION. Il en va ainsi de la mobilité (moteurs, trains, essieux, organes de direction etc.) au soutien (états du réservoir, de la batterie etc.) mais également des fonctions d'observation, de protection ou d'agression (images numérisées puis traitées en temps réel, ajout d'informations sur les vues réelles par le biais de vitrages actifs – réalité augmentée –, déclenchement

automatiques de leurres et de masques selon des dispositifs de détection de départ de coups adverses, ralliement automatique de tourelles ou d'armement dans la direction d'une agression en cours etc.). Grâce à une communication automatique et permanente entre véhicules voisins, cela ouvrira la voie à un combat collaboratif semi-automatisé : lorsqu'un véhicule sera sous le feu ennemi, ses voisins partageront automatiquement sa perception de la provenance des coups et les équipages pourront alors riposter dans des délais très faibles.

La vétronique, cette électronique de véhicule qui connecte chacune des fonctions les unes aux autres, ouvre donc la voie à une accélération des processus de décision, de défense ou d'attaque, mais elle comporte également une amplification du risque numérique : de fait, toute vulnérabilité d'un composant peut avoir des conséquences sur l'ensemble de la plateforme, voire dans toute l'unité de combat, via les réseaux radios et le système d'information.

D'autre part, comme tout milieu numérisé, le véhicule SCORPION ne peut être cloisonné. L'infovalorisation, tout au contraire, repose sur la connexion et le partage d'information. Par ailleurs, la maintenance et la simulation, fonctions toutes deux largement informatisées, auront leurs entrées dans le véhicule. Enfin, au sein de ce que l'on nomme encore la numérisation de l'espace de bataille, il aura vocation à être en interface avec plus d'une trentaine d'autres systèmes militaires numérisés, allant de l'antenne satellite au radar de renseignement, du terminal d'artillerie aux systèmes de pilotage de drones ou de robots, d'une connexion aux systèmes de commandement des échelons supérieurs comme aux capteurs techniques les plus divers, qu'il soient visuels ou acoustiques.

### 3.c. Une démarche de cybersécurité exemplaire

La vertu de SCORPION consiste à avoir procédé à l'analyse de risques et avoir exprimé son besoin de cybersécurité dès sa définition. Le projet de FEROS a quitté la STAT plus d'un an avant la notification du marché, et il était déjà partagé auparavant dans les groupes de travail menés en commun avec la DGA. C'est ce qui permet de faire entrer au contrat diverses conditions qui vont permettre d'assurer une cybersécurité du bon niveau.

Les études préliminaires ont en particulier porté sur la notion de survivabilité, qui est une autre forme d'expression du besoin de résilience. La survivabilité se définit en effet comme la capacité à

continuer d'assurer sa mission dans un contexte marqué de nombreuses agressions intentionnelles qui cherchent à s'opposer à l'accomplissement de cette mission. Elle se caractérise pour un système complexe par le souci de la survie de ses divers composants. Dans l'exemple du groupement tactique SCORPION, la survivabilité consiste en l'évaluation de la gravité en terme d'impact opérationnel sur l'aptitude de ce groupement à réaliser sa mission et en une analyse de vulnérabilités prioritairement axée sur les constituants identifiés comme critiques, et non pas en la survivabilité individuelle des sous-systèmes le constituant.

On applique lors de cette évaluation et de cette analyse les critères habituels que l'on connaît dans les études de risques : la démarche nécessite l'identification des fonctions majeures (localiser, faire mouvement, faire feu etc.) des systèmes et des personnes critiques (personnes dont la perte entraîne une perte de capacité, équipements sensibles, points faibles de l'organisation) et des menaces. L'étape suivante consiste à élaborer des parades de tous types, portant sur le système (ajout de blindage, partage de données etc.), sur l'organisation technique (redondance de certains équipements etc.) ou celle des combattants, ou encore des solutions mixtes, comme le passage du système dans un mode dégradé dans certaines circonstances. Par ailleurs et de manière à assurer une résilience qui n'est pas seulement du domaine cyber, des modes manuels mécaniques de secours sont prévus sur tous les systèmes. Il est hors de question qu'une panne informatique empêche le véhicule d'assurer des fonctions essentielles à sa survie comme la mobilité ou l'agression.

Prise en compte très tôt dans la conception des systèmes, la survivabilité est une approche innovatrice, pluridisciplinaire, qui dépasse l'approche « système » classique et qui permet d'envisager une approche « système de systèmes », nécessaire pour pouvoir traiter toute la complexité du groupement tactique interarmes SCORPION de demain.

# Méthodologie de résilience des systèmes d'information et des systèmes industriels par technique de simulation au sein du centre de gestion de crise de l'UBS

Jean-Philippe Périn (IT-OPSLINK), Charles Préaux

**Mots clefs** : méthodologie de résilience, besoins en résilience, propagation interzones, plans de résilience, virtualisation, simulation hybride

## Introduction

Le pôle de cybersécurité de l'Université de Bretagne Sud (UBS) s'est doté d'un Centre Commun de gestion de Crise Cybernétique(C4) utilisant une plateforme technologique permettant de virtualiser tout ou partie des éléments d'un système d'information d'entreprise ou d'un système industriel de contrôle-commande et de tester leur résilience face à des scénarii d'attaques cybernétiques.

La méthodologie de gestion de crise applicable à la résilience d'un système numérique est le résultat d'un programme de recherche mené par l'UBS qui intègre :

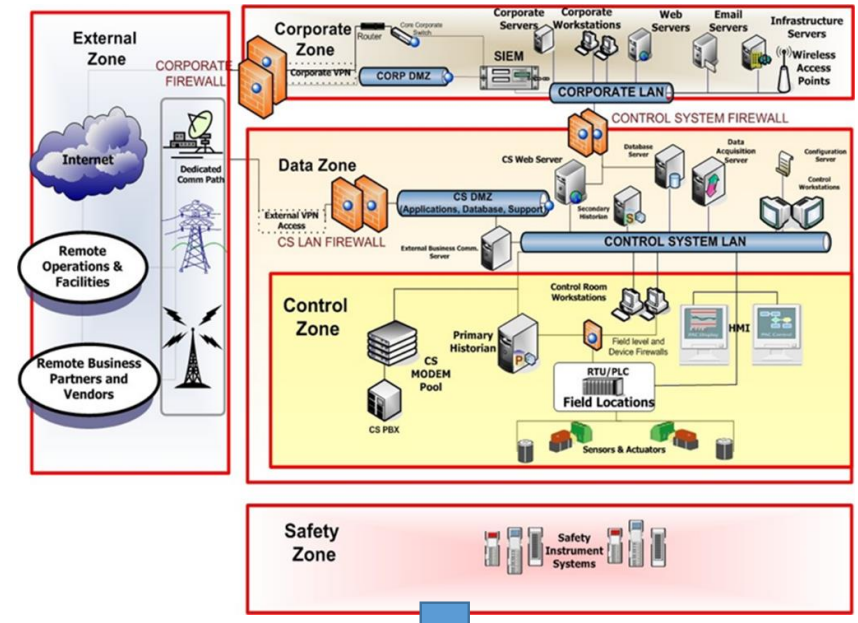
- les notions de résilience interzones, les vulnérabilités des flux échangés et la propagation possible des perturbations,
- les modalités d'attaque et de résistance des systèmes,
- Les plans et mesures de continuité d'activité à mettre en place pour la restauration d'un système en état de dysfonctionnement,
- Les modalités de tests et d'analyse à chaud des résultats pour tester des nouvelles mesures d'amélioration de résilience.

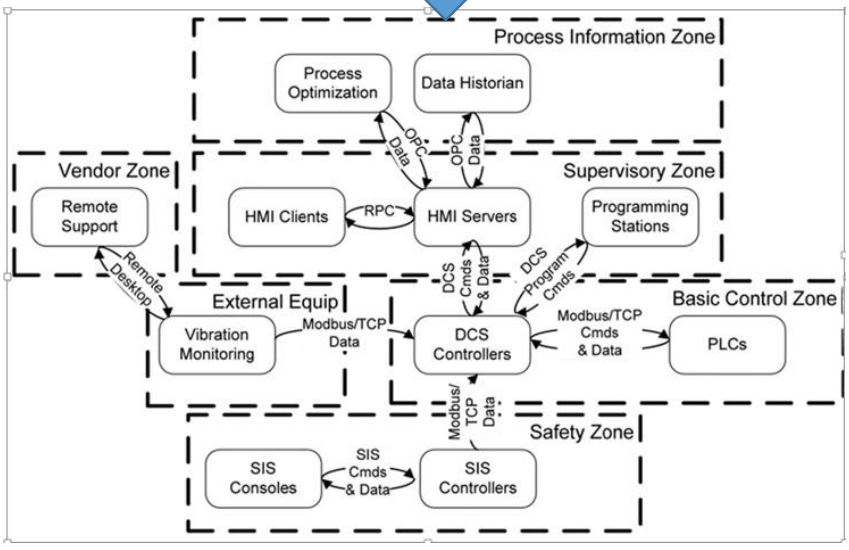
L'avantage pour les entreprises, utilisant cette méthodologie et les moyens du C4, est de tester via une simulation la résilience de leurs systèmes numériques en production sans aucun risque de les endommager. Il est même possible par rapport au système virtualisé de connecter physiquement de nouveaux composants réels pour mesurer l'impact de leurs niveaux de résilience par rapport à la résilience globale du système. On parle alors de test de résilience hybride.

## Contexte

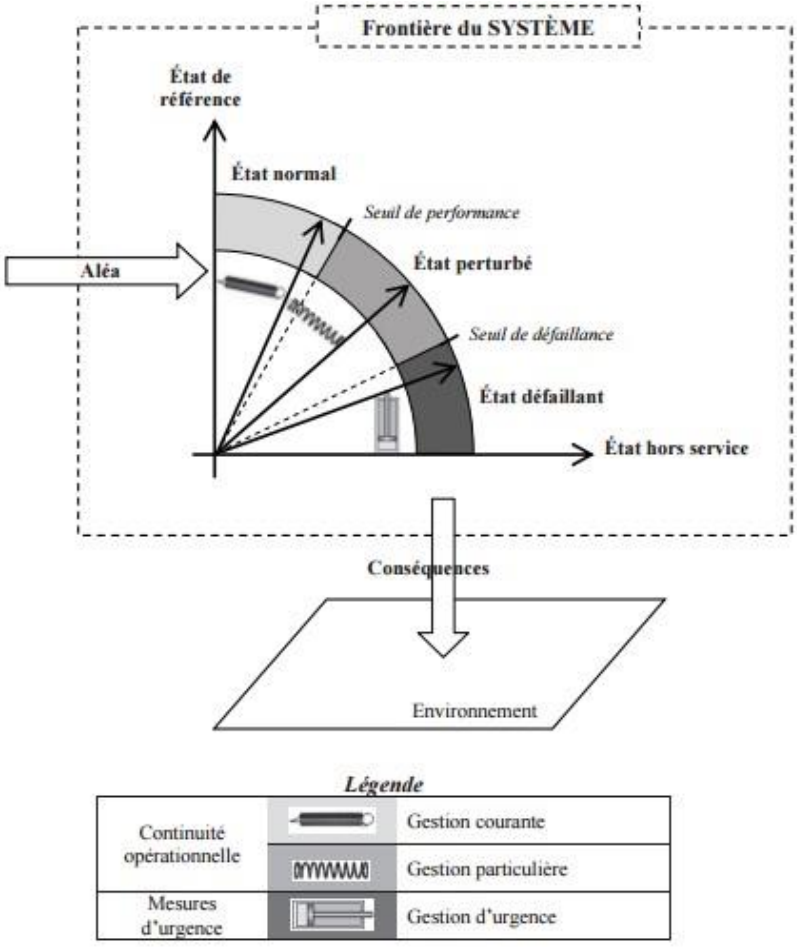
La résilience est la capacité d'un système à maintenir ou à rétablir un niveau de fonctionnement acceptable malgré des perturbations ou des défaillances. (Pinel, 2009, p. 71) Les trois concepts clés de cette définition sont les suivants :

- **« système »** : le système est représenté sous forme de zones interdépendantes caractérisées par l'échange de flux numériques et les possibilités de propagation possibles des perturbations entre zones adjacentes,



- **« malgré des perturbations ou des défaillances »** : chaque zone est représentée par un ensemble de données incluant nom, définition, flux fonctionnels, zones adjacentes, héritages des risques, évaluation des risques de la zone, conséquences d'une brèche de sécurité, criticité, objectifs de résilience, possibilité de propagation. Ces données permettent de comprendre et d'intervenir sur les dysfonctionnements,
- **« capacité […] à maintenir ou à rétablir »** : La résilience d'un système numérique se caractérise par sa capacité à fonctionner en mode dégradé et aux délais nécessaires pour restaurer son fonctionnement en mode nominal.



## Méthodologie de résilience par technique de simulation

Cette méthodologie, développée dans le cadre d'un projet de recherche de l'UBS est encore dans sa phase expérimentale. Elle devrait être testée de façon opérationnelle au centre commun de gestion de crise cybernétique en début 2016.

Cette méthodologie comprend deux phases :

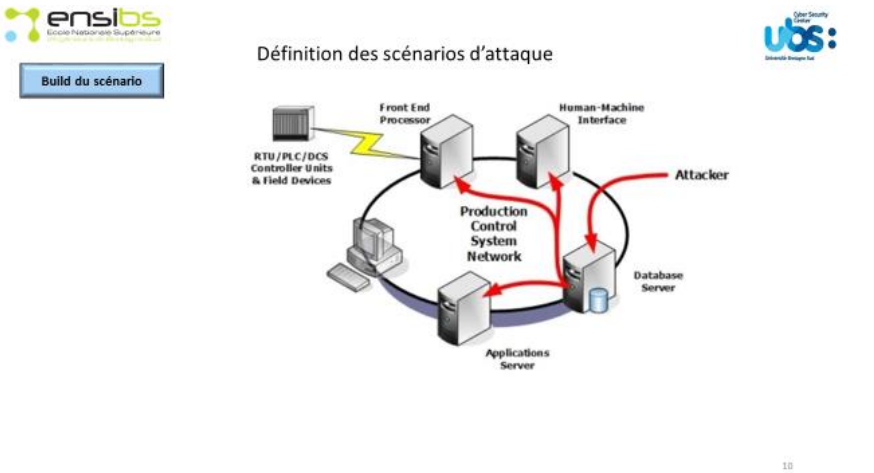
- Une phase BUILD correspondant à la virtualisation du système ou du composant industriel à tester, à l'étude de ses vulnérabilités cybernétiques connues ainsi que des mesures de cyber sécurité mises en place, enfin à la définition des scénarios techniques d'attaque pour tester sa résilience,
- Une phase RUN durant laquelle les scénarios techniques et opérationnels seront joués afin de mesurer la résilience du système.

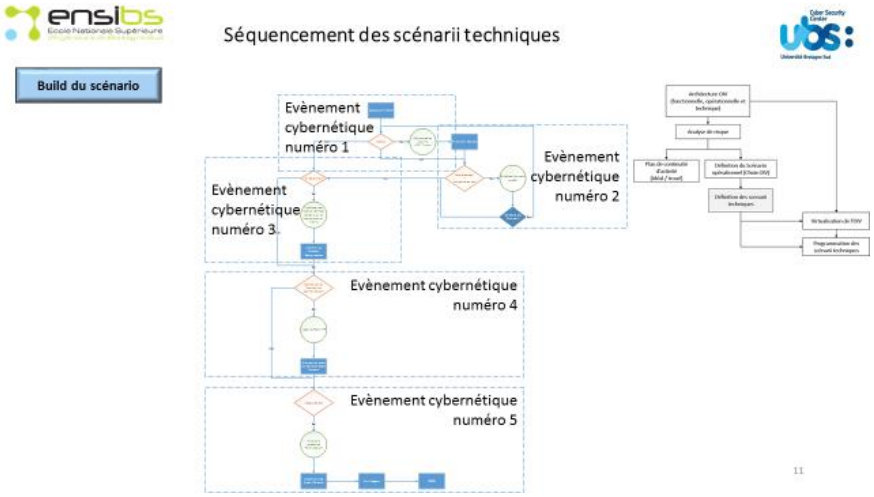
Phase BUILD

Pour l'étude de la résilience d'un système numérique, nous retrouvons donc les étapes suivantes :

- **Virtualisation du système** par construction des zones (groupement des composants ou sous-systèmes) dans le simulateur incluant cartographie et caractéristiques des flux et interdépendances des zones adjacentes ainsi que la représentation dans l'architecture système des composants systèmes utilisés. La plateforme utilisée est la plateforme hynesim.

- **Construction des scénarii techniques** de perturbations ou de défaillances (cyber attaques),




- Analyse des besoins de résilience du système,

Build du scénario

Equipements industriels attaqués
Source : Risidata.com

| Equipements industriels attaqués | |
|---|---|
| PLC | 12 |
| Ordinateurs réseau contrôle | 10 |
| Serveurs | 6 |
| Contrôleur industriel | 5 |
| HMI | 5 |
| DCS | 5 |
| Routeur | 5 |

| Points d'intrusion |
|---|
| Réseau de gestion |
| Accès à distance via LAN du réseau de gestion de l'entreprise |
| Local - HMI |
| Dial-up modem |
| Accès internet direct |
| Accès par les fournisseurs - prestataires |

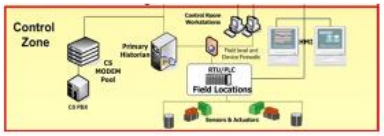
- **Elaboration des plans de résilience des zones** (plans de continuité d'activité, mesures d'urgence et de restauration etc.),

Build du scénario

Fiche de défense de zone

Nom de la zone
Définition de la zone
Responsable opérationnel
Flux fonctionnels
Zones adjacentes
Héritage des risques
Evaluation des risques de la zone
- Protections des composants actuelles
- Vulnérabilités des composants
- Possibilités de propagation d'une cyber attaque

Conséquences d'une brèche de sécurité
Criticité
Objectifs de défense
- Mesures préventives
- Connections interzones
- Processus d'isolation en cas d'attaque
- Processus de restauration




- Sélection des normes et protocoles de mesures puis construction des capteurs virtuels,  Définitions des niveaux de fonctionnement : normal, dégradé, défaillant.

Phase RUN

Le test de résilience commence par le choix des scénarii techniques à effectuer pour mesurer la résilience du système. Les données de mesures envoyées par les capteurs

permettent de visualiser la propagation des dysfonctionnements et de mesurer les seuils de résilience du système.

Nous pouvons donc exécuter deux types de test pour mesurer la résilience du système selon les critères fournis au préalable par l'entreprise ou l'industrie.

- Simulation virtuelle totale : test de résilience sur un réseau (ou partie) de réseau numérique totalement virtualisé
- Simulation virtuelle hybride : test de résilience sur un réseau (ou partie) de réseau numérique partiellement virtualisé et incluant des composants (Hardware) connectés physiquement au réseau virtuel.

Les scénarii d'attaque cybernétique étant reproductibles, il est intéressant pour les responsables techniques du réseau de tester en cours de simulation des solutions d'amélioration de résilience du système puis d'en visualiser rapidement les résultats.

## Conclusion

L'utilisation de la méthodologie de résilience d'un système numérique développée par l'UBS permet donc :

- pendant la phase BUILD de caractériser les zones en terme de résilience et d'identifier les possibilités de propagation de dysfonctionnent, d'analyser les besoins en résilience de ces zones face à un éventail d'attaques ou de perturbations cybernétiques prédéfinies puis de définir et mettre en place des contre-mesures visant à accroître la résilience du système,
- pendant la phase RUN de mesurer et d'améliorer en continu la résilience du système selon les critères de mesure préalablement choisis par l'entreprise.

L'utilisation de la plateforme de virtualisation du C4 permet à une entreprise de tester de manière sécurisée différentes solutions pour définir ou renforcer la résilience de ses systèmes.

Pour les entreprises, le service commercial relatif à l'élaboration ou à la mesure de la résilience d'un réseau numérique est accessible via le Club des Partenaires qui est une organisation hébergée par la Fondation de l'Université de Bretagne Sud.