# Cyberdéfense des systèmes de contrôle-commande industriels : une approche par filtres basée sur la distance aux états critiques pour la sécurisation face aux cyberattaques

Franck Sicard, Éric Zamaï, Jean-Marie Flaus

# Cyberdéfense des systèmes de contrôle-commande industriels : une approche par filtres basée sur la distance aux états critiques pour la sécurisation face aux cyberattaques

Franck SICARD, Éric ZAMAI, Jean-Marie FLAUS

Univ.Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, F-38000 Grenoble, France

franck.sicard@grenoble-inp.fr

**Abstract.** Les systèmes de contrôle-commande industriels (Industrial Control Systems, *ICS* en anglais) sont de plus en plus présents dans notre vie quotidienne et intégrés à nos infrastructures critiques. A l'origine, les ICS ont été conçus pour assurer la productivité et garantir la sûreté des systèmes contrôlés. Toutefois, l'Histoire récente nous a démontré qu'ils étaient vulnérables à des attaques par vecteur numérique et que les architectures matérielles offraient des surfaces d'attaques importantes. A ces vulnérabilités, héritées principalement de l'introduction de technologies provenant de l'Information Technology (IT), deux types d'approches peuvent fournir des solutions intéressantes : celles basées sur la sûreté de fonctionnement (Operational Technology, OT) et celles basées sur la sécurité des systèmes d'information. L'approche proposée hybride deux types de solution : l'approche filtre et les IDS afin de détecter des intrusions dans les ICS pouvant amener le système dans des états critiques. La localisation des filtres est primordiale pour assurer la pertinence des mécanismes de détection. La notion de distance permettra de contrôler la distance aux états critiques et ainsi de prévenir des dérives vers ces états. Le concept de trajectoire permettra d'évaluer cette distance au cours du temps et de l'évolution du système. Dans une première section, nous présenterons les systèmes de contrôle-commande. Puis nous mettrons en avant les vulnérabilités et les surfaces d'attaque pouvant conduire à des cyberattaques. La troisième partie permettra d'expliquer l'approche proposée et de la positionner dans la littérature. Enfin, une partie expérimentale sera proposée où l'approche sera mise en œuvre en simulation sur un exemple inspiré d'un système industriel.

**Keywords:** Système de contrôle-commande industriel, Cybersécurité, Approche par Filtres, Model Based, Systèmes à événements discrets, Architecture de défense

adfa, p. 1, 2011.

# 1 Introduction

Les systèmes de contrôle-commande industriels, ou Industrial Control System (ICS) en anglais, mettent en œuvre un ensemble de couches numériques et physiques qui interagissent afin de réaliser un objectif dans un environnement industriel. La dénomination de systèmes cyber-physiques peut également être trouvée dans la littérature [1]. De nos jours, ces systèmes sont présents dans de nombreux domaines notamment dans des infrastructures critiques telles que : la production et la distribution d'énergie (électricité, eau, gaz, …), les systèmes de productions manufacturiers, les transports, la défense ou encore les services de santé [2]. Différentes typologies d'architectures matérielles peuvent être trouvées dans la littérature pour décrire un ICS ; toutefois, l'architecture CIM (computer-integrated manufacturing) sera utilisée dans ces travaux pour décrire les différents niveaux hiérarchiques [3]. Le découpage proposé dans CIM, et représenté sur la Fig. 1, permet de réduire la complexité d'un ICS.

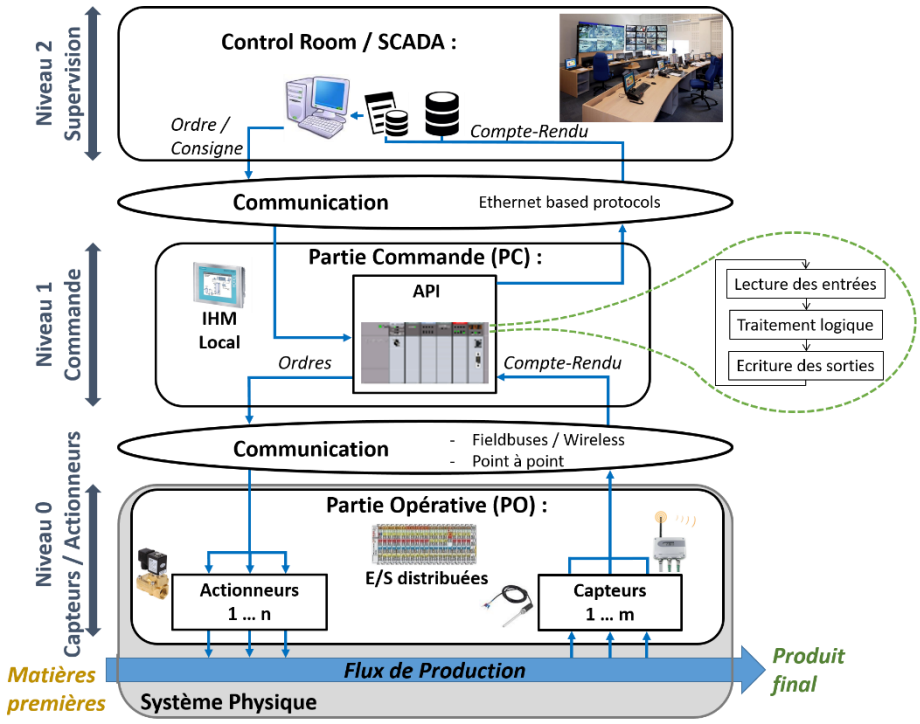


**Fig. 1.** Représentation d'un système de contrôle-commande industriel basé sur l'architecture fonctionnelle de la norme CIM (seules les couches temps-réel sont représentées)

Chaque niveau est composé d'éléments caractéristiques et échange des informations avec les autres niveaux. Nous présenterons dans cette étude les 3 premiers niveaux d'un ICS :

- Niveau 0 (Capteurs – Actionneurs) : l'objectif d'un système automatisé est de transformer une matière première en produit fini via le flux de production. Ainsi, le système amène le procédé d'un état initial à un état final en agissant sur le flux de production en garantissant productivité et fiabilité. Pour ce faire, les capteurs et les actionneurs font le lien entre la partie numérique et la partie physique, l'ensemble formant le système physique. Ce niveau sera également appelé Partie Opérative (PO) dans la suite de cette étude,
- Niveau 1 (Contrôle) : Cette couche reçoit les données remontées du terrain par les capteurs (niveau 0), commande les actionneurs (niveau 0) et communique avec les opérateurs via les IHM locaux et la salle de supervision. L'élément principal de ce niveau est l'Automate Programmable Industriel (API) qui contrôle le système en temps-réel. Ainsi l'API répète les étapes du cycle suivant : (i) lecture des entrées, (ii) exécution logique de la loi de commande, (iii) écriture des sorties. Ce niveau sera également nommé Partie Commande (PC),
- Niveau 2 (Supervision) : le rôle de cette couche est de donner une image à un instant du procédé. Ainsi, les données acquises par les niveaux inférieurs sont remontées. Les opérateurs peuvent également piloter le système via des ordres de fabrication pour adapter la loi de commande et éviter des dommages. Le terme SCADA (Supervisory Control And Data Acquisition) peut être employé pour désigner cette partie, ce sera le cas dans ce papier, ou pour faire référence à un ICS.
- Réseaux de communication : les réseaux de communication permettent de connecter les différentes couches. A l'origine les ICS utilisaient des connections analogiques, numériques ou des protocoles de communication propriétaires pour échanger des informations, toutefois dans les architectures récentes, le protocole TCP/IP est largement utilisé. Il permet d'augmenter le volume et la vitesse des données transportées [4]. L'introduction de technologies héritées du monde de l'IT a fait apparaître des vulnérabilités dans les ICS qui seront développées dans la section 2. Afin d'être complet, les protocoles sans-fils [1], tel que le WirelessHart ou l'ISA100, occupent une place de plus en plus importante dans le cadre de l'industrie 4.0.

Notre étude se focalise uniquement sur les niveaux 0, 1 et 2 de l'architecture CIM. Les niveaux supérieurs, qui définissent l'ordonnancement de la production et le management général ne font pas l'objet de ces recherches.

## 2 Des systèmes soumis à des cyberattaques

Depuis le début du siècle, les systèmes de contrôle-commande industriels sont devenus la cible de hackers qui exploitent les vulnérabilités des équipements ou de l'architecture afin de réaliser des cyberattaques [5], [6]. L'intérêt des hackers pour ces

systèmes industriels vient du fait qu'ils sont facilement attaquables puisque les problématiques de sécurité n'y ont pas été prises en compte [7]. En effet, comme expliqué dans le paragraphe précédent, les ICS ont été mis œuvre pour résoudre des problématiques de production (productivité et fiabilité). Au-delà de ce manquement, l'introduction de technologies (Ethernet, réseau sans-fil, …) provenant du monde de l'IT a généré de nouvelles vulnérabilités : intrinsèques à ces technologies ou par effet d'empilement (accumulation de technologies hétérogènes). Enfin, cet intérêt provoqué par l'existence de surfaces d'attaques est accentué par les dommages que peuvent induire une cyberattaque sur le système physique (arrêt de production, temps de réparation, temps de redémarrage, …), sur son environnement (impact négatif sur l'humain, la santé ou encore l'environnement) et l'entreprise (pertes financières, image négative, …) [2], [5], [8].

Une liste exhaustive de cyberattaques menées contre des ICS peut être trouvée dans [6], [9] ou encore [10]. Toutefois, trois attaques majeures seront développées ci-dessous :

- Station d'épuration de Maroochy Shire (Australie, 2000) [11] : il s'agit là de la première attaque contre un ICS dans le sens où une intrusion a eu lieu et le procédé a été endommagé. Un ancien employé a utilisé ses accès pour s'introduire dans le SCADA et relâcher 800.000L d'eaux usées causant d'importants dégâts environnementaux,
- Stuxnet (Iran, 2010) [12] : Stuxnet est encore de nos jours la référence de ce que peut être une cyberattaque. Ce ver informatique a été introduit dans le réseau industriel via une clé USB. En ciblant les consoles de programmation ayant le logiciel Step7, Stuxnet a réussi à infecter les API Siemens S7. Une fois dans l'automate, le virus exécute son code malveillant en envoyant des commandes erronées aux centrifugeuses afin qu'elles tournent plus ou moins vite provoquant ainsi une usure prématurée. Dans le même temps, les données des capteurs faisant apparaître le dysfonctionnement étaient bloquées au niveau de l'automate. Les opérateurs de supervision ne pouvaient pas voir l'attaque puisque Stuxnet modifiait les données remontées (data spoofing),
- Cyberattaque contre une aciérie et sur un réseau électrique (Allemagne, 2014 et Ukraine, 2015) [13], [14] : ces attaques marquent un tournant puisque les attaquants ont pénétré d'abord le réseau bureautique avant de s'introduire sur le réseau industriel.

Une liste complète de vulnérabilités et des attaques subies est disponible sur l'ICS-CERT [15]. Comme les systèmes d'information classiques, les ICS peuvent être victimes d'attaques DDOS, Man-in-the-middle (MITM) ou replay, notamment au niveau 2 de l'architecture CIM [6]. Toutefois, des attaques plus spécifiques aux niveaux 1 et 0 peuvent être menées. Elles peuvent ne pas prendre en compte la logique du procédé (*attaques aléatoires*) ou justement s'appuyer dessus : *attaques par séquences* [16], [17] qui modifient la séquence d'actions envoyée par la PC, et *attaques par injection de données* [18], [19] où les données sont interceptées et modifiées. Les attaques peuvent cibler la remontée de données de la PO vers la PC afin d'induire une mauvaise commande ou alors affecter directement l'ordre envoyé à la PO. Afin d'être complet

sur les possibilités d'attaques, l'API possède également des vulnérabilités qui sont développées dans [6], [9] comme la modification de firmware ou l'altération de la configuration. Une bonne définition d'une cyberattaque sur un ICS peut être trouvée dans Rubio-Hernan et al [20].

Les solutions utilisées dans le domaine IT pour sécuriser les systèmes d'information sont peu satisfaisantes à cause des spécificités des ICS comme :

- Contraintes temps-réels très fortes,
- Priorités différentes (Disponibilité, Intégrité et Confidentialité),
- Empilements hétérogènes et non standardisés des protocoles et des technologies,
- Fonctionnement continu (24/24, 7/7) et peu de mises à jour,
- Ressources matérielles limitées (notamment dues à l'environnement industriel).

Ainsi, les techniques de cryptographie ne peuvent résoudre entièrement le problème à cause de contraintes temps-réel et de la gestion des clés de cryptage. Il en va de même pour les antivirus, avec en plus la gestion des bases utilisées pour la détection. De plus, un ICS est un environnement soumis à des aléas [21] qui ont les mêmes conséquences que des attaques, à savoir la perte de service. Les principales sources de défaillance sont : le facteur humain, les équipements, les recettes ou les produits. La Fig.2 regroupe les vulnérabilités et les sources de défaillances possibles sur un système de contrôle-commande. Enfin, la formation et la prévention sont encouragées par plusieurs règlementations comme l'ANSSI [7] ou la NIST [2] mais ces recommandations complémentaires nécessitent du temps pour les mettre en place.

Notre travail se concentre sur les niveaux 0 et 1 de l'architecture ICS, qui sont peu abordés dans la littérature en se concentrant sur l'aspect « connaissances métiers ». L'objectif principal de cette étude sera la détection de cyberattaques entre la PO et la PC. En effet, au vu de l'architecture ICS et des vulnérabilités, il s'agit du dernier rempart pour bloquer une attaque voulant exécuter une action malveillante sur le système. L'accent sera mis sur la protection des biens et des personnes et l'approche développée anticipera les déviations afin d'éviter autant que faire se peut les blocages. Toutefois, le problème scientifique soulevé par notre approche est la difficulté à distinguer une attaque d'une défaillance matérielle puisque seul le comportement « normal » du système sera pris en compte dans nos modèles. Un problème technologique est également traité dans cette étude concernant la localisation des mécanismes de détection et l'intégration dans les architectures matérielles de contrôle-commande.

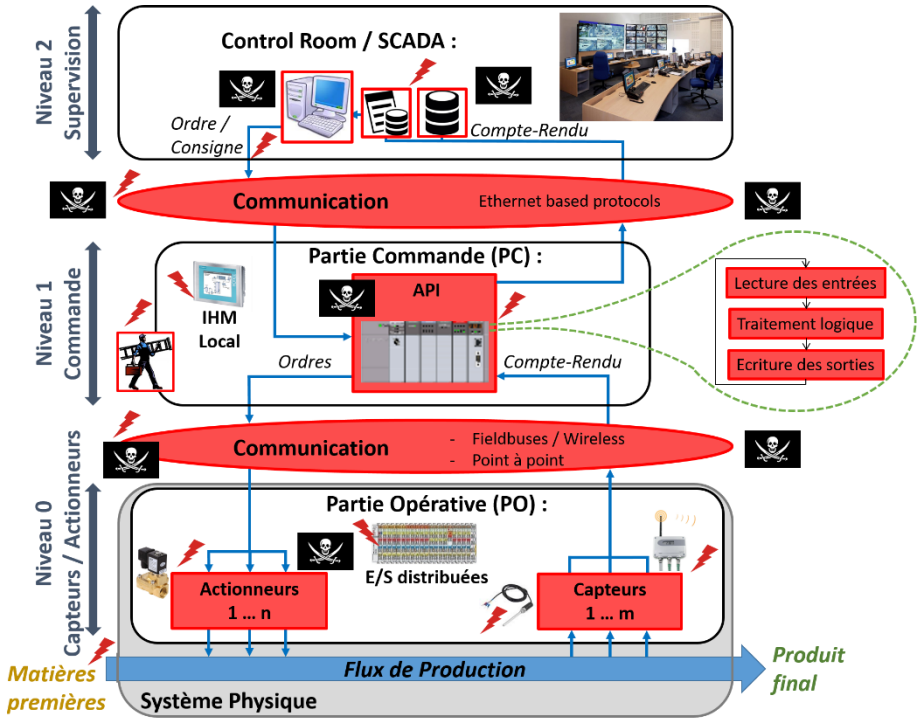


**Fig. 2.** Localisation des vulnérabilités (pirates) et des défaillances (éclairs rouges) dans un système de contrôle-commande industriel

## 3 Approche proposée : filtre basé sur la notion de distance pour la cyber sécurité des ICS

Dans cette partie, nous ferons un bref état de l'art de travaux en cyber sécurité des ICS. Nous détaillerons ensuite l'intégration dans les architectures matérielles de l'approche proposée et la méthodologie de conception. Enfin, nous aborderons les mécanismes de détection mis en œuvre notamment la notion de distance et de trajectoire.

### 3.1 Etat de l'art des solutions pour sécuriser les systèmes de contrôle-commande

La cyber sécurité des ICS est un domaine de recherche de plus en plus investigué depuis les années 2000. [6] détaille de nombreuses solutions développées qui peuvent être au niveau software, firmware, hardware, réseau et « ICS process ». Dans la suite de ce papier, nous nous intéressons à une technique de détection développée à partir de l'aspect réseau : les Intrusion Detection System (IDS). Les IDS sont des sondes déployées dans un réseau afin d'analyser les informations échangées par rapport à une

politique du réseau. Si une information ne respecte pas cette politique alors une alerte est envoyée. La réflexion autour de la localisation des sondes est essentielle pour assurer une bonne détection d'intrusion. [22] développe les approches de détection via les réseaux de communication notamment avec des IDS. D'autres techniques de détection basées sur des signatures ou des approches statistiques sont présentées dans [23]. L'approche par IDS est intéressante puisqu'elle permet de comparer les informations entre un modèle et les informations remontées par le système. La notion de distance développée par Carcano [24] permet de quantifier l'éloignement du système par rapport à des états définis comme critiques. Cette notion sera développée dans la suite de notre approche. Cependant, le nombre de sondes à déployer sur un réseau peut être important ce qui impacterait l'architecture matérielle des systèmes de contrôle-commande.

Par ailleurs, la cyber sécurité se rapproche de la problématique de détection dans les approches de surveillance des systèmes. Plusieurs de ces approches ont été considérées dans un contexte de cyberattaque dans [25]. L'approche filtre (Fig 3) développée par D.Cruette [26] présente des caractéristiques intéressantes. Elle permet d'analyser les ordres et les comptes rendus avant qu'ils ne soient distribués respectivement à la PO et à la PC. Si une discordance par rapport aux modèles est détectée alors l'information peut être bloquée ce qui préserve le système physique d'états critiques.

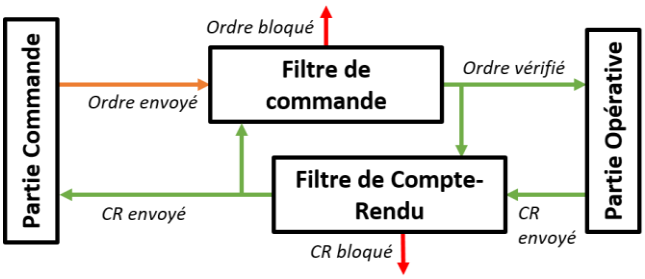


**Fig. 3.** Illustration de l'approche filtre

L'approche proposée hybride les deux techniques présentées précédemment selon l'idée du renforcement mutuel développée dans [5].

### 3.2 Approche proposée : localisation dans l'architecture

Comme présenté dans le paragraphe précédent, les filtres développés dans notre approche sont basés sur le concept d'IDS et sur l'approche Filtre. Dans la première approche, des sondes sont déployées sur le réseau pour vérifier si les données échangées respectent la politique de sécurité du réseau. Dans ces approches, aucune limitation du nombre de sondes existe dans les systèmes d'information. Dans les ICS, les filtres doivent être faciles à déployer dans l'architecture matérielle, ainsi le nombre de sondes doit être maîtrisé. L'approche filtre satisfait à cette contrainte avec 2 blocs de

vérification, le filtre de commande et de compte-rendu (CR). Contrairement à l'approche initiale de Cruette, ces filtres ne sont plus à l'intérieur de l'API, puisque ce dernier est vulnérable. De plus, ces filtres seront situés au plus près des organes de commande afin de réduire la surface d'attaque. En effet, une connexion réseau est attaquable contrairement à une connexion physique. Ainsi, le filtre de commande peut être comparé à un interrupteur électrique qui peut s'ouvrir si les modèles du système ne sont plus respectés protégeant ainsi le système physique. Le filtre de CR ne bloquera pas la remontée d'information mais communiquera avec le filtre de commande sur d'éventuelles discordances. La communication entre les filtres se fait sur un réseau de communication différent de celui utilisé pour le réseau industriel. L'implémentation de cette approche est détaillée en Fig 4.

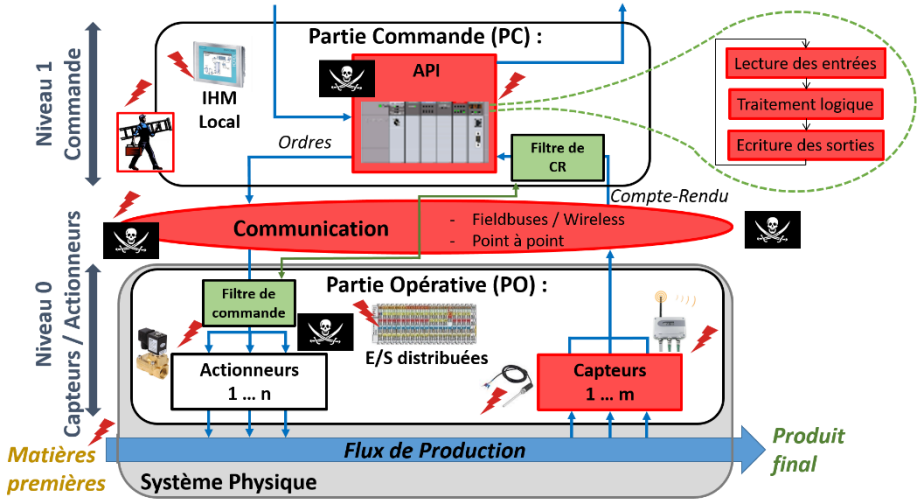


**Fig. 4.** Implémentation de l'approche proposée dans l'architecture ICS

### 3.3 Approche proposée : méthodologie de conception des filtres

L'approche proposée aboutit à la construction d'algorithmes de détection implantés dans des filtres. Ils ont pour but d'assurer la sécurité des biens et des personnes, en se basant sur des modèles du procédé (ce que le système peut faire) et de commande (ce que l'on veut qu'il fasse). Cette méthodologie de conception, illustrée en Fig 5, se décompose en 3 étapes détaillées dans [25] : l'analyse de risques, l'exploration des états du système, la synthèse des filtres.

La première étape consiste en une analyse de risques qui permet d'identifier les états critiques (états pouvant dégrader le système), optimaux (états respectant la loi de commande) et les états dangereux (états qui ne respectent pas la loi de commande mais ne sont pas critiques) du système considéré. Cette étape s'appuie sur les connaissances maîtrisées en milieu industriel en sûreté de fonctionnement. Plusieurs méthodologies ont été développées afin de réaliser une analyse de risques d'une infrastruc-

ture industrielle, en prenant ou non en compte le risque de cyberattaques. Dans la suite de notre étude, nous nous appuierons sur la méthode EBIOS [27], notamment recommandée par l'ANSSI. Un autre enjeu de cette étape est de déterminer les paramètres nécessaires et suffisants pour modéliser correctement le système à protéger. Notre approche n'a pas vocation à protéger tout un système industriel mais seulement la ou les parties les plus critiques. Ainsi, nous limitons également les problématiques d'explosions combinatoires de la partie modélisation.

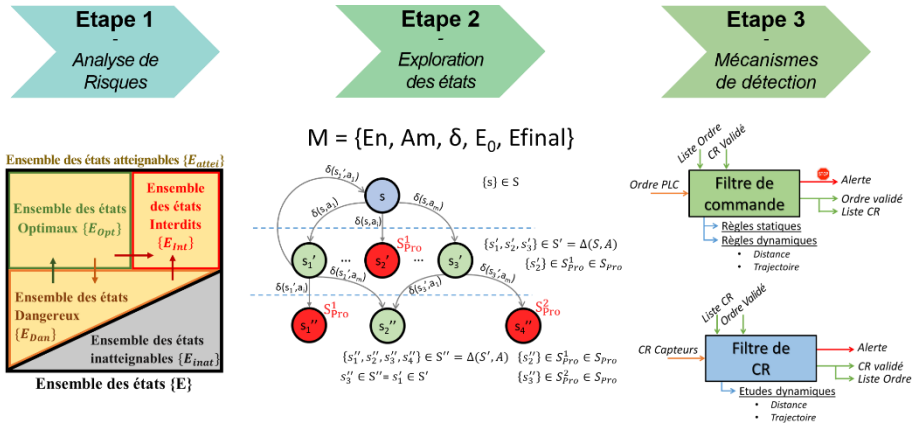


**Fig. 5.** Schéma simplifié illustrant la méthodologie de conception des filtres

La seconde étape permet de modéliser le système de contrôle-commande. Pour cela, les réseaux de Pétri sont utilisés pour modéliser la commande du système [25]. Un automate fini déterministe $M = \{E_n, A_m, \delta, E_0, E_{final}\}$ est utilisé pour décrire le comportement de la PO. L'ensemble $E_n$ désigne l'ensemble des états possibles pour le système, $A_m$ l'ensemble des ordres exécutables, $E_0$ l'état initial et $E_{final}$ les états finaux. L'ensemble des états $E_n$ se compose des états atteignables $E_{reach}$ lui-même composé des états optimaux $E_{Opt}$, dangereux $E_{Dan}$ et interdits $E_{Pro}$. La fonction de transition δ permet d'exprimer les effets des actions sur le système physique comme expliqué dans l'équation (1) et généralisé à des ensembles d'états dans (2).

$$\forall s \in E_i \subset E_n,\ \forall\ a \in A_m \text{ tel que } s'=\delta(s, a) \in E_{i+1} \subset E_n \quad (1)$$

$$E_{i+1} = \Delta(E_i, A_m) = \{s'=\delta(s, a) \mid s \in E_i, a \in A_m\} \subset E_n \quad (2)$$

Il devient alors possible de construire un algorithme permettant d'identifier des contextes menant à ces états interdits. Un contexte est défini comme un couple {état ; action} où l'exécution de l'ordre conduit le système dans un état interdit. Cet algorithme calcule l'image d'un ensemble d'états au travers de δ en appliquant toutes les combinaisons d'ordres possibles. Ainsi, lors de la première itération seul l'état initial est évalué. Afin de limiter l'explosion combinatoire, des conditions d'arrêt sont ajou-

tées dans $E_{final}$. Ainsi, l'algorithme arrête l'exploration d'une branche si un état s' de l'ensemble des états suivants $E_{i+1}$ appartient à l'un des ensembles suivants:

- Etat initial $E_0$ : $E_{i+1} \cap E_0 \neq \{\emptyset\} \leftrightarrow \exists\, s' \in E_{i+1}$ such that $s' \in E_0$,
- Etats interdits $s' \in E_{Pro} \subset E_{reach}$. Les contextes menant à ces états sont intéressant puisque leurs exécutions impactent sévèrement le système,
- Boucles ω. Si l'algorithme calcule un état déjà rencontré précédemment alors l'exploration de la branche se termine.

L'algorithme explore donc toutes les combinaisons d'états possibles atteignables. L'explosion combinatoire est limitée par des conditions d'arrêts qui garantissent qu'une branche unique est explorée à chaque itération. A la fin de cette étape, nous avons à notre disposition l'ensemble des états optimaux, dangereux, et interdits ainsi que les contextes menant à ces états.

### 3.4 Mécanismes de détection d'anomalie

Dans cette partie, nous nous intéressons aux mécanismes de détection d'anomalie ainsi qu'à celui de discrimination et de mise en repli du système protégé.

**Détection du contexte : blocage immédiat**

Ce mécanisme se base sur les contextes obtenus lors de l'étape précédente d'exploration des états. Lorsque le filtre de commande repère un couple état-action conduisant à un état interdit alors l'action est bloquée. On peut définir ce mécanisme comme une règle R décrite dans l'équation (3)

$$R=\text{vrai} \leftrightarrow \exists\, s' \in E_{i+1},\ \exists\, s \in E_i,\ \exists\, a \in A_m \text{ tel que } s'=\delta(s,a) \in E_{Pro} \quad (3)$$

Ce mécanisme permet de mettre en sécurité le système mais le blocage d'un ordre est pénalisant pour une installation industrielle.

**Distance entre états et trajectoire du système : anticipation des déviations**

Afin d'éviter les blocages, un mécanisme permettant aux filtres de détecter des dérives est implémenté. Il se base sur la notion de distance développée par Carcano [24] qui calcule l'éloignement du système par rapport à des zones définies comme critiques. Elle est parfaitement adaptée pour des systèmes avec des variables continues mais moins pour des systèmes à évènements discrets. Pour ces derniers, nous introduisons une nouvelle définition de la distance en (4) : la distance est le nombre minimum d'actions à appliquer à un état s pour arriver dans un état interdit.

$$\forall\, s \in E_i,\ \forall\, a \in A_m,\ D(s|E_{Pro}) = \min_n \Delta^n(E_i, A_m) \in E_{Pro} \quad (4)$$

Ainsi, à chaque instant, les filtres peuvent calculer le plus court chemin vers un état critique. Le concept de trajectoire permet d'étudier l'évolution de la distance sur plusieurs séquences. Il devient alors possible de détecter des dérives lorsque la trajectoire tend vers un état critique (distance nulle). La détection se base sur l'évolution de la

distance de l'état courant par rapport aux états interdits ainsi qu'aux états optimaux mais également sur la distance entre l'ordre analysé et l'ordre attendu par le modèle de commande. La discrimination entre une attaque et une défaillance se base sur le même principe. En étudiant l'évolution de ces indicateurs et en prenant l'hypothèse qu'un attaquant cherchera toujours à amener le système vers un état critique, il devient possible discriminer l'anomalie rencontrée. Toutefois, si l'attaquant imite une défaillance pour dégrader la qualité du procédé ou augmenter la durée du flux de production alors les mécanismes ne détecteront pas une cyberattaque.

**Mise en repli après une attaque**

Les mécanismes mis en place précédemment permettent de protéger le système contre des attaques directes ou de séquences pouvant l'amener dans un état critique. Toutefois, après le blocage d'un ordre, le système doit être mis en repli en attendant un redémarrage. L'API pouvant être potentiellement compromis, les filtres utilisent les modèles pour conduire le système vers un état stable. Les filtres peuvent ainsi conduire le système vers l'état initial ou un autre état en utilisant une séquence d'ordre préétablie ou en utilisant la notion de distance pour s'éloigner des états critiques.

# 4 Mise en application

Dans cette partie, nous allons illustrer l'approche décrite précédemment sur un exemple connu de la littérature et similaire à celui développé dans [25]. Le système, représenté en Fig. 6 est composé de 5 cuves. Les cuves $C_1$ et $C_2$ contiennent respectivement les produits A et B. Chaque cuve déverse à tour de rôle son produit dans la cuve de mélange $C_4$ via les vannes $V_1$ et $V_2$ afin d'obtenir un produit C. La vanne $V_3$ permet de vidanger la cuve $C_4$ dans la cuve $C_5$. La cuve $C_3$ contient un produit D qui se déverse dans la cuve $C_5$. La vanne $V_5$ permet de vidanger la cuve $C_5$. Chaque cuve de mélange est équipée de capteurs de niveau. Lorsque la cuve $C_4$ est vide ($H_0^{C4}$ inactif), la loi de commande impose un remplissage par $V_1$ jusqu'au capteur $H_1^{C4}$, puis par $V_2$ jusqu'au capteur $H_2^{C4}$. L'ouverture de la vanne $V_3$ permet de vidanger la cuve jusqu'à ce que le capteur $H_0^{C4}$ ne soit plus actif. Pour la cuve $C_5$, la loi de commande impose un remplissage par $C_4$ jusqu'au capteur $H_1^{C5}$ puis un remplissage par $V_3$ et $V_4$ jusqu'au capteur $H_3^{C5}$. La vidange de la cuve $C_5$ est ensuite ordonnée.

En appliquant la méthodologie présentée dans la partie précédente, une analyse de risques est menée sur le système. Deux états interdits ont été identifiés et ils sont atteints lorsqu'un ordre de remplissage est envoyé au système alors que :

- Le capteur $H_2^{C4}$ est activé dans le cas d'une ouverture de $V_1$ ou $V_2$ ou le capteur $H_1^{C4}$ est activé dans le cas d'ouverture des deux vannes $V_1$ et $V_2$ pour la cuve $C_4$.
- Le capteur $H_3^{C5}$ est activé dans le cas d'une ouverture de $V_3$ ou $V_4$ ou le capteur $H_2^{C5}$ est activé dans le cas d'ouverture des deux vannes $V_3$ et $V_4$ pour la cuve $C_5$.

Cette étape sert également à identifier les paramètres nécessaires à la modélisation du système physique. Aucun paramètre critique ne doit être oublié tout en limitant le nombre de capteurs et d'actionneurs à prendre en compte. Le système sera défini par les vecteurs et ensembles suivants :

- Le vecteur d'état s qui modélise le fonctionnement normal du système, les défauts n'étant pas pris en compte. Le vecteur d'état est composé par la valeur des capteurs et des actionneurs tel que s = [Capteurs; Actionneurs] = $[H_i^{C4}\ H_j^{C5}\ V_1\ V_2\ V_3\ V_4\ V_5]$ avec $H_i^{C4} \in \{0..2\}$, $H_j^{C5} \in \{0..3\}$. L'ensemble des états possibles E est composé de 386 états (384 états possibles et 2 états interdits),
- Le vecteur d'ordre regroupe les actions qui peuvent être exécutées sur le système (32 combinaisons). On a : $a \in A_{32}$ avec $a=(a_1, a_2, a_3, a_4, a_5)$ où $a_i \in$ {0 ferme vanne $V_i$, 1 ouvre vanne $V_i$}, $i \in \{1..5\}$,
- L'ensemble des états atteignables $E_{Reach} \in E$ est composé de 209 états dont 11 états optimaux, 2 états interdits (2580 contextes) et 196 états dangereux $E_{Dan}=E_{Reach} \setminus (E_{Opt} \cap E_{Pro})$.
- L'ensemble des états inatteignables $E_{Unreach}=E \setminus E_{Reach}$ est lui composé de 175 états.

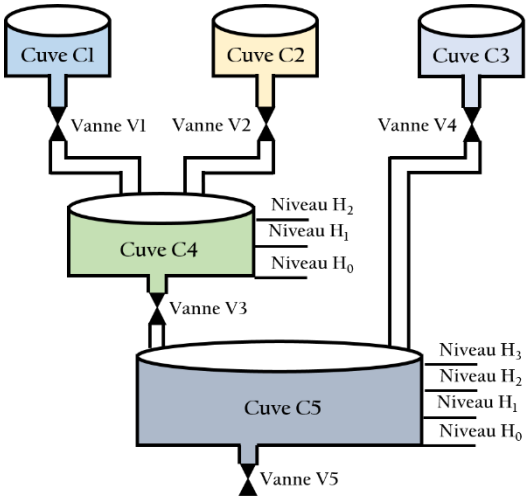


**Fig. 6.** Illustration de l'exemple d'application

Dans cet exemple, on prendra l'hypothèse que l'exécution d'un ordre entraine immédiatement le remplissage/vidange de la cuve ainsi que l'activation du capteur de niveau supérieur/inférieur. Les états dangereux et interdits sont différents puisque l'on cherchera à éviter les deuxièmes qui endommagent le système (par exemple δ([2 0 0 0 0 0 0], [1 0 0 0 0]). Les états dangereux, eux, sont une transition entre les états opti-

maux et critiques, lorsque le système ne respecte pas la loi de commande, par exemple l'état de remplissage [2 0 1 1 0 0 0] qui correspond au remplissage de $C_4$ qui est vide par $V_1$ et $V_2$.

Une fois l'algorithme d'exploration des états terminé, les mécanismes de détection présentés dans la partie précédente sont implantés dans les filtres de commande et de procédé. Une attaque « man in the middle » est simulée sur notre système de cuves. Elle affecte la communication entre l'API et les actionneurs en interceptant un ordre a $\in A_{32}$ pour le remplacer par un ordre $a_{attaque} \in A_{32}$. Cette attaque fermera la vanne de vidange $V_3$ et ouvrira les vannes $V_1$ et $V_2$ au lieu de $V_4$ ce qui devrait entrainer un débordement de la cuve $C_4$. La Fig.7 représente la surveillance du système par le filtre de commande durant le scenario d'attaque. On remarque que le remplissage de la cuve $C_4$ se déroule normalement car même si le système se rapproche d'un état critique de débordement (entre 0 et 6, la distance du système à l'état interdit le plus proche dStateInt passe de 2 à 1 puis se stabilise) la séquence d'ordre suit la trajectoire optimale (entre 0 et 6, les trajectoires des ordres dOrdreOpt et des états optimaux dStateOpt sont nulles). Lorsque l'ordre d'ouverture de la vanne $V_3$ et $V_4$ est envoyé, il est intercepté et remplacé par $a_{attaque}$. La cuve $C_4$ risque de déborder puisque le capteur $H_1^{C4}$ est actif, les vannes $V_1$ et $V_2$ ouvertes et la vanne $V_3$ fermée. Ainsi, l'ordre est bloqué par le filtre de commande (en 7, la distance à l'état interdit le plus proche dStateInt devient nulle) et le système est mis en repli par l'ouverture des vannes $V_3$ et $V_5$. Après le blocage de l'ordre, la discrimination privilégie une attaque puisque l'ordre analysé conduit directement le système vers un état interdit en ne respectant pas la loi de commande (la distance dOrdreOpt passe à 1).

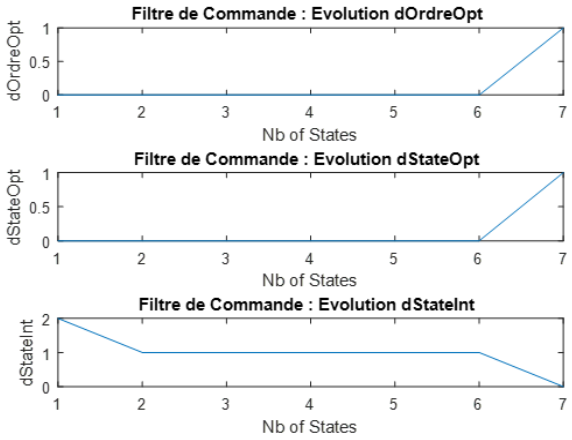


**Fig. 7.** Evolution de la trajectoire du système vue par le filtre de commande (*haut* : trajectoire de l'ordre analysé en fonction de l'ordre attendu, *milieu* : trajectoire de l'état courant par rapport aux états optimaux, *bas* : trajectoire du système par rapport à l'état interdit le plus proche)

La détection d'une attaque peut ne pas conduire au blocage d'un ordre par le filtre de commande. Par exemple, si l'ordre $a_{attaque}$ est remplacé par l'ouverture de la vanne $V_4$ à la place de $V_1$ alors le filtre de commande détectera que l'ordre envoyé ne correspond pas à la loi de commande. L'ordre ne sera pas bloqué car il ne conduit pas directement à un état interdit mais une alerte sera émise par le filtre. Là encore, la discrimination indiquera une attaque plutôt qu'une défaillance.

En revanche, si l'attaquant corrompt la remontée de données alors la discrimination sera plus difficile à réaliser. Par exemple, lorsque le capteur $H_1^{C4}$ est atteint, le pirate envoie le $CR_{attaque}$ correspondant à $H_0^{C4}$. Ainsi, l'attaque correspond également à une défaillance du capteur $H_1^{C4}$ et provoque un décalage entre l'état réel du système et l'image que s'en fait l'API. Ce problème est dû au fait que le filtre de CR est placé avant l'API ainsi ce filtre remarque une différence entre le CR obtenu et le CR attendu mais il ne peut pas faire la distinction entre une défaillance et une attaque. S'il était placé juste après les capteurs, alors le filtre aurait accès aux données non corrompues. Cette hypothèse sera explorée dans le cadre de travaux ultérieurs. En revanche, le filtre de commande bloquera toujours un ordre entrainant le système dans un état critique. Cependant, la qualité du procédé peut être dégradée et la discrimination ne donne pas un résultat probant.

# 5 Conclusion

Les systèmes de contrôle-commande industriels sont utilisés dans de nombreuse infrastructures critiques et domaines d'application. Ces systèmes assurent productivité et sûreté des installations. Cependant, l'utilisation de technologies héritées de l'IT ont introduit des vulnérabilités. Ces surfaces d'attaque peuvent être utilisées par des hackers pour impacter gravement un système industriel et son environnement.

Dans ce papier, nous avons présenté une approche innovante s'appuyant sur des filtres de détection d'attaques dans les couches basses de l'architecture ICS. Elle s'appuie sur les avantages de solutions utilisées dans la sécurité et la sûreté de fonctionnement. La méthodologie de conception des filtres se compose de trois étapes. La partie analyse de risques permet d'identifier les états critiques et les paramètres nécessaires à la modélisation. La phase d'exploration des états permet d'identifier les contextes menant à des états interdits et d'identifier les états atteignables du système. L'algorithme développé permet de limiter l'explosion combinatoire. Enfin, la phase de synthèse des filtres permet d'implémenter les mécanismes de détection. Les attaques directes sur le système peuvent être détectées par les contextes trouvés lors de la phase d'exploration des états. Les notions de distance et de trajectoire permettent de protéger le système contre les attaques par séquence. Enfin, nos algorithmes ont été testés sur un exemple de simulation inspiré d'un système réel.

La détection d'attaques impactant la remontée de données devra être améliorée en modifiant notamment la position du filtre de compte-rendu. Le fait de rapprocher ce filtre des capteurs permettra de détecter des discordances entre l'état réel du système et l'image donnée à l'API. La discrimination des anomalies entre attaque et défaillance devra être également investigué. Ce papier a développé des algorithmes de dé-

tection basé sur des contraintes combinatoires. Une attaque peut également consister à respecter la séquence d'ordre-état sans tenir compte de l'aspect temporel. Dans nos prochaines études, nous investiguerons des algorithmes de détection basés sur des contraintes combinatoires et temporelles. Enfin, des expérimentations sur des plateformes industrielles seront également réalisées pour compléter et approfondir les résultats de cet article.

**Remerciement :** Ces recherches sont financées par la Direction Générale de l'Armement (DGA) – Maîtrise de l'information situé à Bruz, France.

## 6 Références

[1] Y. Ashibani et Q. H. Mahmoud, « Cyber physical systems security: Analysis, challenges and solutions », *Comput. Secur.*, vol. 68, p. 81-97, juill. 2017.

[2] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, et A. Hahn, « Guide to Industrial Control Systems (ICS) Security », National Institute of Standards and Technology, NIST SP 800-82r2, juin 2015.

[3] J. Clarhaut, N. Dupoty, F. Ebel, J. Hennecart, et F. Vicogne, *Cyberdéfense: La sécurité de l'informatique industrielle (domotique, industrie, transports)*. France: Editions ENI, 2015, 2015.

[4] E. D. Knapp, *Industrial network security: securing critical infrastructure networks for smart grid, scada, and other industrial control systems*, 2nd edition. Waltham, MA: Elsevier, 2014.

[5] Y. Fourastier et L. Pietre-Cambacedes, *Cybersécurité des installations industrielles : défendre ses systèmes numériques*. Cépaduès Editions, 2015.

[6] S. McLaughlin *et al.*, « The Cybersecurity Landscape in Industrial Control Systems », *Proc. IEEE*, vol. 104, nº 5, p. 1039-1057, mai 2016.

[7] ANSSI, « Maîtriser la SSI pour les systèmes industriels », ANSSI (Agence nationale de la sécurité des systèmes d'information), Paris, V1.0, juin 2012.

[8] Y. Yang, T. Littler, S. Sezer, K. McLaughlin, et H. F. Wang, « Impact of cybersecurity issues on Smart Grid », in *2011 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe)*, 2011, p. 1-7.

[9] F. Khorrami, P. Krishnamurthy, et R. Karri, « Cybersecurity for Control Systems: A Process-Aware Perspective », *IEEE Des. Test*, vol. 33, nº 5, p. 75-83, oct. 2016.

[10] « RISI - The Repository of Industrial Security Incidents », 09-sept-2016. [En ligne]. Disponible sur: http://www.risidata.com/Database/event_date/asc. [Consulté le: 09-sept-2016].

[11] M. Abrams et J. Weiss, « Malicious Control System Cyber Security Attack Case Study - Maroochy Water Services, Australia », *Secur. Water Wastewater Syst.*, juill. 2008.

[12] N. Falliere, L. O. Murchu, et E. Chien, « W32. stuxnet dossier », Symantec Security Response, Version 1.4, févr. 2011.

[13] R. M. Lee, M. J. Assante, et T. Conway, « German steel mill cyber attack ». SANS ICS 2014, déc-2014.
[14] R. M. Lee, M. J. Assante, et T. Conway, « Analysis of the Cyber Attack on the Ukrainian Power Grid ». SANS ICS 2016, mars-2016.
[15] ICS-CERT, « ICS-CERT / The Industrial Control Systems Cyber Emergency Response Team », 15-sept-2016. [En ligne]. Disponible sur: https://ics-cert.us-cert.gov/. [Consulté le: 15-sept-2016].
[16] M. Caselli, E. Zambon, et F. Kargl, « Sequence-aware Intrusion Detection in Industrial Control Systems », in *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security*, New York, NY, USA, 2015, p. 13–24.
[17] W. Li, L. Xie, Z. Deng, et Z. Wang, « False sequential logic attack on SCADA system and its physical impact analysis », *Comput. Secur.*, vol. 58, p. 149-159, mai 2016.
[18] R. Deng, G. Xiao, R. Lu, H. Liang, et A. V. Vasilakos, « False Data Injection on State Estimation in Power Systems #8212;Attacks, Impacts, and Defense: A Survey », *IEEE Trans. Ind. Inform.*, vol. 13, nº 2, p. 411-423, avr. 2017.
[19] Y. Wang, Z. Xu, J. Zhang, L. Xu, H. Wang, et G. Gu, « SRID: State Relation Based Intrusion Detection for False Data Injection Attacks in SCADA », in *Computer Security - ESORICS 2014*, M. Kutyłowski et J. Vaidya, Éd. Springer International Publishing, 2014, p. 401-418.
[20] J. Rubio-Hernan, L. De Cicco, et J. Garcia-Alfaro, « Event-Triggered Watermarking Control to Handle Cyber-Physical Integrity Attacks », in *Secure IT Systems: 21st Nordic Conference, NordSec 2016, Oulu, Finland, November 2-4, 2016. Proceedings*, B. B. Brumley et J. Röning, Éd. Cham: Springer International Publishing, 2016, p. 3-19.
[21] D.-T. Nguyen, « Diagnostic en ligne des systèmes à événements discrets complexes: approche mixte logique/probabiliste », Université Grenoble Alpes, 2015. Français. 〈NNT : 2015GREAT067〉. 〈tel-01227260〉.
[22] R. Mitchell et I.-R. Chen, « A survey of intrusion detection techniques for cyber-physical systems », *ACM Comput. Surv.*, vol. 46, nº 4, p. 1-29, mars 2014.
[23] F. Sicard, J. M. Flaus, et É. Zamaï, « Approche filtre basée sur la notion de distance pour la détection des cyberattaques », présenté à 15ème Colloque National AIP-Primeca, La Plagne, France, 2017.
[24] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. Nai Fovino, et A. Trombetta, « A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems », *IEEE Trans. Ind. Inform.*, vol. 7, nº 2, p. 179-186, mai 2011.
[25] F. Sicard, É. Zamaï, et J. M. Flaus, « Distance Concept Based Filter Approach for Detection of Cyberattacks on Industrial Control Systems », présenté à IFAC World Congress, GdR MACS Young PhD Researchers - Open Invited Track of Extended Abstract, Toulouse, France, 2017.
[26] D. Cruette, J. P. Bourey, et J. C. Gentina, « Hierarchical specification and validation of operating sequences in the context of FMSs », *Comput. Integr. Manuf. Syst.*, vol. 4, nº 3, p. 140–156, 1991.
[27] ANSSI, « EBIOS MÉTHODE DE GESTION DES RISQUES », janv. 2010.