Convergence sûreté-sécurité des Systèmes de Contrôle Industriel

Mike DA SILVA
Pierre-Henri THEVENON
CEA-Leti, Université Grenoble Alpes,
F38000 Grenoble, France
Firstname.Name@cea.fr

Stéphane MOCANU
Laboratoire d'Informatique de Grenoble
Univ. Grenoble Alpes
CNRS, Inria, Grenoble-INP
Grenoble, France
stephane.mocanu@inria.fr

Maxime PUYS
Univ. Clermont Auvergne
CNRS, Mines de Saint-Etienne, LIMOS
Clermont-Ferrand, France
maxime.puys@uca.fr

***Abstract*—Les systèmes de contrôle industriel (ICS) sont conçus pour fournir un service, tel que la production d'électricité ou le traitement de l'eau, tout en protégeant les personnes, les biens et l'environnement. Aujourd'hui, les ICS évoluent en intégrant de plus en plus les technologies de l'information (IT), exposant ainsi leurs infrastructures aux cyberattaques. Cependant, contrairement aux technologies de l'information, les ICS présentent des risques et des contraintes en matière de sûreté et nécessitent des solutions de cybersécurité spécifiques qui empêchent les cyberattaques d'avoir un impact sur la sûreté du système. Dans cet article, nous présenterons les principaux mécanismes d'une méthode d'analyse de risque sûreté-sécurité que nous avons développé. Cette méthode permet d'identifier les vulnérabilités du système ainsi que leur impact sur la sûreté.**

***Index Terms*—Sûreté, cybersécurité, Analyse de risque, ICS, IT, OT**

I. Introduction

Au cours des vingt dernières années, les ICS sont devenus de plus en plus ouverts à internet grâce à l'interconnexion des technologies de l'information et des technologies opérationnelles (IT et OT). Cependant, la gestion des risques des systèmes OT est fondamentalement différente de celle des systèmes IT et est mise en évidence dans le guide NIST SP800-82r3 [4] (2023) à travers une comparaison entre ces systèmes. Cette analyse montre que l'IT et l'OT ont des objectifs différents (Gestion des données vs. Contrôle du monde physique) ce qui influe sur les propriétés à assurer. En effet, en informatique, la confidentialité et l'intégrité des données sont primordiales avec pour risque majeur retard dans l'activité de l'entreprise alors qu'en OT, la sûreté des biens et des personnes est primordiale et la disponibilité est critique pour le système. Ces différences soulignent la nécessité d'utiliser une évaluation des risques de cybersécurité spécifique aux ICS afin de concevoir des systèmes sûrs et sécurisés. Nous définissons la sûreté et la sécurité conformément à la norme CEI 62443-1-1 [2] : *Sûreté* est une propriété qui caractérise la capacité du système à se protéger contre des événements prévisibles, tandis que *Sécurité* est la capacité à protéger le système contre les cybermenaces. Dans le présent document, les termes "sécurité" et "cybersécurité" seront utilisés indifféremment. Dans cet article, nous présenterons un aperçu d'une méthode d'analyse de risque sûreté-sécurité que nous avons développé. Cette méthode permet d'identifier les vulnérabilités du système ainsi que leur impact sur la sûreté.

II. Méthode

Dans cette section, nous présentons notre méthode d'analyse de risque qui identifie et évalue les menaces de cybersécurité qui impactent la sûreté du système. Cette méthode est divisée en trois étapes: (1) modélisation de la menace, (2) identification des impacts sur la sûreté, et (3) évaluation des risques. Dans la suite de cette section, nous détaillons chacune des étapes de notre méthode.

A. Modélisation de la menace

L'objectif de cette première partie est d'identifier les vulnérabilités du système ainsi que leur attribuer une vraisemblance. Pour cela, nous avons utilisé un l'outil Microsoft Threat Modeling Tool (MTMT) [3]. Dans cet outil, le système analysé est modélisé sous la forme d'un diagramme de flux de données (DFD) reliant les composants entre eux par des flèches directionnelles. Chaque élément générique du DFD (processus, stockage de données, flux, etc.) est représenté par un *stencil* et peut être dérivé pour modéliser des éléments spécifiques. La modélisation d'un système nécessite un *template* (une bibliothèque de stencils et de stencils dérivés) et d'une base de connaissances de vulnérabilités classées par familles de menaces STRIDE[1]. De plus, MTMT propose une fonction d'analyse de modèle qui génère automatiquement une liste de vulnérabilités exploitables liées à une famille STRIDE à partir de la base de connaissances.

Cependant, MTMT manque de représentativité pour les composants des ICS afin de modéliser correctement les menaces de ce type de système. C'est pourquoi nous avons complété MTMT par un outil qui permet à la fois de modéliser des dispositifs ICS réels et de trouver leurs vulnérabilités. Cet outil est capable de créer, à partir d'une base de données extensible de composants de ICS, un modèle pour MTMT qui inclut les *Common Vulnerabilities and Exposures* (CVE) pertinentes de ces composants. De plus, nous avons étendu la liste des

[1]Spoofing: usurpation d'identité, Tampering: falsification, Repudiation: répudiation, Information disclosure: divulgation d'informations, Denial of service: dénis de service, Elevation of priviledge: élévation de privilège)

vulnérabilités exploitables extraite par l'outil avec une nouvelle propriété permettant de leur attribuer une vraisemblance. Nous déterminons la probabilité d'une vulnérabilité selon le score d'exploitabilité du *common vulnerability scoring system* (CVSS) V3.1 [1] qui a l'avantage d'harmoniser les scores et d'être largement utilisé.

Pour résumer, nous avons développé un outil qui permet d'effectuer une analyse de la menace des ICS avec Microsoft threat Modeling Tool afin d'identifier les vulnérabilités exploitables du système ainsi que leur probabilité d'exploitation.

B. Identification des impacts sur la sûreté

Cette seconde partie s'intéresse à identifier les impacts sur la sûreté des vulnérabilités extraites par l'outil d'analyse de la menace. Pour cela, nous avons développé une méthode qui génère des scénarios de cyberattaque qui compromettent la sûreté. Ces scénarios impliquent la manipulation des mesures des capteurs et des commandes de contrôle afin de compromettre les fonctions de sûreté assurées par le système. Notre approche suit 3 étapes séquentielles: (1) modélisation du procédé physique piloté par l'ICS, (2) identification des fonctions de sûreté dans le modèle, et (3) génération de cyberattaque qui compromettent la sûreté.

1) Modélisation du processus: Un automate programmable agit selon un programme généralement écrit graphiquement (par exemple, SFC) qui définit l'état du système, les transitions entre ces états, ainsi que les commandes. De plus, ce programme garantit à la fois la fourniture du service (par exemple, la production d'électricité) et la sûreté du système, qui protège les personnes, les biens et l'environnement contre les dangers. La première étape de cette méthode consiste à modéliser le système en convertissant chaque programme des automates programmables en automate à localisation stable (SLA) [5]. Le SLA permet d'expliciter le comportement de chaque programme, c'est-à-dire, tous les états stables du programme ainsi que toutes les transitions entre ces états.

2) Identification des fonctions de sûreté dans le modèle: Afin de trouver des attaques contre les fonctions de sûreté, définis par un expert du processus, nous devons d'abord les identifier dans le modèle. Un automate programmable fonctionne en trois étapes : il lit ses entrées (capteurs), puis il exécute son programme et enfin il calcule les sorties (actionneurs). Cette procédure fonctionne comme une implication logique entre les entrées et les sorties. Par conséquent, nous définissons les fonctions de sûreté du système par une implication logique entre les entrées et les sorties ($entrees \implies sorties$). Par exemple, dans le cas de la gestion d'un réservoir, nous pouvons définir une fonction de sûreté $Plein \implies \neg Vanne$ (réservoir plein implique la fermeture de la vanne) afin d'éviter un débordement du réservoir. Dans le SLA, les fonctions de sûreté sont représentées par des transitions entre deux états. Nous qualifions ces transitions du SLA de critiques car, si elles ne sont pas satisfaites, le système est en danger.

3) Génération de scénarios de cyberattaque qui compromettent la sûreté: L'objectif des adversaires est de compromettre de manière malveillante les fonctions de sûreté afin de mettre le système en danger. Afin d'identifier le risque de cybersécurité, nous modélisons les attaques dans le SLA pour définir comment les adversaires peuvent compromettre la sûreté.

Deux attaques possibles sont définies dans le modèle. La première bloque un événement, c'est-à-dire qu'elle bloque un changement d'état (entrées) ou un changement de commande (sorties) du programme. Par exemple, dans le cas de la gestion d'un réservoir, nous pouvons définir une transition critique entre l'état *"Remplir le réservoir"* et *"Réservoir plein"* où un capteur *"Haut"*, qui détecte que le réservoir est plein, déclenche la transition et force la vanne à se fermer. Les adversaires peuvent falsifier la valeur du capteur *"Haut"* pour ne pas satisfaire la transition, ce qui entraîne un débordement du réservoir ou, après que la transition critique a eu lieu, bloquer le changement de la valeur de sortie de la vanne pour la maintenir ouverte, ce qui entraîne également un débordement.

La deuxième attaque possible vise à forcer un événement, c'est-à-dire à forcer un changement d'état ou un changement de commande. Toujours dans l'exemple de la gestion du réservoir, après la transition critique, les adversaires peuvent injecter un ensemble d'entrée pour rouvrir la vanne ou simplement envoyer une commande d'ouverture de la vanne à l'actionneur afin de faire déborder le réservoir.

Cette modélisation définit à la fois les données (entrées ou sorties) que les adversaires doivent manipuler et l'état du système à partir duquel les fonctions de sûreté peuvent être compromises.

C. Évaluation des risques

La dernière étape de notre méthode consiste à remplir une matrice des risques en fonction de la probabilité et de l'impact des risques. La section II-A attribue une probabilité d'exploitabilité aux vulnérabilités du système et la section II-B relie ces vulnérabilités à des scénarios d'attaque qui compromettent la sûreté du système. Ainsi, nous déterminons la probabilité d'un risque (scénario d'attaque) en fonction de la probabilité d'exploitabilité des vulnérabilités qui composent le scénario d'attaque et l'impact en fonction de la valeur quantitative de l'impact du danger associé à la fonction de sûreté compromise par le scénario d'attaque.

III. Remerciements

Ce travail a bénéficié d'une aide de l'Etat Français au titre du programme d'Investissements d'Avenir, IRT Nanoelec, portant la référence ANR-10-AIRT-05.

References

[1] First, "Common vulnerability scoring system version 3.1 : Specification document revision 1," 2019. [Online]. Available: https://www.first.org/cvss/v3-1/cvss-v31-specification_r1.pdf

[2] IEC 62443-1-1, "Terminology, concepts and models," Geneva, CH, International Standard, Jul. 2009.

[3] Microsoft, "Microsoft Threat Modeling Tool," 2022. [Online]. Available: https://learn.microsoft.com/en-us/azure/security/develop/threat-modeling-tool

[4] NIST, "Guide to Operational Technology (OT) Security," Tech. Rep. NIST Special Publication (SP) 800-82, Rev.3, 2023.

[5] J. Provost, J.-M. Roussel, and J.-M. Faure, "Translating Grafcet specifications into Mealy machines for conformance test purposes," *Control Engineering Practice*, Sep. 2011.