Titre: Title:	Détection et investigation en temps réel de la menace interne
Auteur: Author:	Thibault Leblanc
Date:	2026
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Leblanc, T. (2026). Détection et investigation en temps réel de la menace interne [Master's thesis, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/76882/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/76882/
Directeurs de recherche: Advisors:	Frédéric Cuppens, & Nora Boulahia Cuppens
Programme: Program:	Génie logiciel

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Détection et investigation en temps réel de la menace interne

THIBAULT LEBLANC

Département de génie informatique et génie logiciel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

Génie Informatique

Mai 2026

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Ce mémoire intitulé :

Détection et investigation en temps réel de la menace interne

présenté par **Thibault LEBLANC**

en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

a été dûment accepté par le jury d'examen constitué de :

Alejandro QUINTERO, président
Frédéric CUPPENS, membre et directeur de recherche
Nora BOULAHIA CUPPENS, membre et codirectrice de recherche
Omar ABDUL WAHAB, membre

DÉDICACE

À Dieu, seul à qui je dois tout,

À mes parents, qui m'ont transmis, parmi beaucoup d'autres, la valeur du travail,

REMERCIEMENTS

La rédaction de ce mémoire a été une aventure enrichissante et stimulante, et je tiens à exprimer ma profonde gratitude à toutes les personnes qui m'ont soutenu et encouragé au cours de ce parcours.

Tout d'abord, je souhaite remercier mes directeurs de recherche, Nora et Frédéric CUPPENS, pour leur encadrement exceptionnel, leurs conseils avisés et leur soutien constant. Leur accompagnement a été inestimable et a grandement contribué à la qualité de ce travail.

Je tiens également à exprimer ma reconnaissance à mes collègues de laboratoire, Neda Moghadam, Marc-Antoine Faillon, Jean-Yves Ouattara, et à ceux que j'oublie qui ont participé, de près ou de loin, à la publication des articles qui constituent une partie intégrante de ce mémoire. Votre collaboration et votre expertise ont été précieuses pour enrichir ce travail de recherche.

Je remercie particulièrement ma famille, Papa, Maman et Marie, pour leur amour, leurs encouragements et leur soutien indéfectible. Votre éducation et votre confiance en moi m'ont donné la force de persévérer. Un remerciement spécial est à accorder à mon père pour m'avoir donné le goût et la culture de la sécurité informatique.

Un grand merci à tous mes amis de Montréal à ma Normandie, pour leur soutien moral, leurs encouragements et les moments de détente partagés. Votre amitié a été une source de motivation et de réconfort. J'adresse des remerciements spécifiques à l'ensemble des familles du groupe scout 1ère Saint Joseph de Montréal et de la Paroisse Ste Madeleine d'Outremont qui furent comme ma deuxième famille à Montréal.

Je souhaite aussi exprimer ma gratitude envers les institutions et organismes qui ont financé ou soutenu ce projet de recherche à travers l'entente Mitacs, notamment au Mouvement des Caisses Desjardins, à la Banque Nationale du Canada, à Mondata et particulièrement à Qohash pour l'expérience professionnelle qu'ils m'ont accordée.

Enfin, je remercie chaleureusement toutes les personnes qui ont contribué à la réalisation de ce mémoire. Votre aide et votre soutien ont été essentiels à la réussite de ce projet.

RÉSUMÉ

La menace interne, définie comme l'exploitation malveillante ou négligente d'accès légitimes aux systèmes d'information, constitue un défi majeur pour les organisations modernes, avec des coûts croissants et des délais de détection importants. Face à l'inefficacité des approches traditionnelles, centrées sur les menaces externes, ce mémoire propose une solution hybride et dynamique pour la détection et l'investigation en temps réel des scénarios de menace interne. Cette solution s'appuie sur une architecture innovante combinant des techniques avancées d'apprentissage automatique non supervisé et une approche multi-agents, conçue pour répondre aux exigences opérationnelles des centres de sécurité.

Plus spécifiquement, la méthodologie développée s'articule autour de deux modules complémentaires opérant en flux continu. Le premier module assure une détection d'anomalies en temps réel sur les flux d'événements, et filtre les flux, ne gardant que les évènements anormaux. Le second module, dédié à l'investigation, repose sur un système multi-agents propulsé par des LLM. Ces agents collaborent pour construire le scénario autour des alertes, exploiter le contexte des actions utilisateurs pour caractériser les anomalies et les positionner au sein d'une chaîne d'attaque adaptée à la menace interne.

La validation expérimentale de cette approche, menée sur le jeu de données de référence CERT r4.2, démontre la pertinence du couplage entre détection statistique et raisonnement multi-agentique. Les résultats indiquent que si la détection statistique seule peut générer des faux positifs, l'ajout de la couche d'investigation automatisée permet de comprendre les alertes pour filtrer efficacement non pertinentes tout en qualifiant avec précision les scénarios malveillants avérés, tels que l'exfiltration de données ou le sabotage.

En définitive, ce travail pose les bases d'une approche proactive qui dépasse la simple levée d'alerte. En fournissant une analyse contextuelle et intelligible quasi-instantanément, la solution proposée offre aux analystes des centres opérationnels de sécurité un outil pour réduire les délais d'investigation et anticiper les incidents critiques avant qu'ils ne causent des dommages irréversibles.

ABSTRACT

Internal threats, defined as the malicious or negligent exploitation of legitimate access to information systems, pose a major challenge for modern organisations, with rising costs and significant detection delays. Given the ineffectiveness of traditional approaches, which focus on external threats, this thesis proposes a hybrid and dynamic solution for the real-time detection and investigation of internal threat scenarios. This solution is based on an innovative architecture combining advanced unsupervised machine learning techniques and a multi-agent approach, designed to meet the operational requirements of security centres.

More specifically, the methodology developed is based on two complementary modules operating in a continuous flow. The first module detects anomalies in real time in event flows and filters the flows, retaining only abnormal events. The second module, dedicated to investigation, is based on a multi-agent system powered by LLM. These agents collaborate to build a scenario around the alerts, exploiting the context of user actions to characterise the anomalies and position them within an attack chain adapted to the internal threat.

Experimental validation of this approach, conducted on the CERT r4.2 reference dataset, demonstrates the relevance of combining statistical detection and multi-agent reasoning. The results indicate that while statistical detection alone can generate false positives, the addition of the automated investigation layer makes it possible to understand alerts in order to effectively filter out irrelevant ones while accurately qualifying proven malicious scenarios, such as data exfiltration or sabotage.

Ultimately, this work lays the foundation for a proactive approach that goes beyond simply raising alerts. By providing contextual and intelligible analysis almost instantly, the proposed solution offers security operations centre analysts a tool to reduce investigation times and anticipate critical incidents before they cause irreversible damage.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

AE	Autoencoder
AdaBoost	Adaptive Boosting
ADAMS	Anomaly Detection at Multiple Scales
ANN	Artificial Neural Network
BERT	Bidirectional Encoder Representations from Transformers
BGMM	Bayesian Gaussian Mixture Model
CERT	Computer Emergency Response Team
CNN	Convolutional Neural Network
CPT	Conditional Probability Table
CVE	Common Vulnerabilities and Exposures
DAG	Directed Acyclic Graph
DARPA	Defense Advanced Research Projects Agency
DL	Deep Learning
DLP	Data Loss Prevention
EDR	Endpoint Detection and Response
FBI	Federal Bureau of Investigation
GCN	Graph Convolutional Network
GMM	Gaussian Mixture Model
GNN	Graph Neural Network
GPT	Generative Pre-trained Transformer
IDS	Intrusion Detection System
IF	Isolation Forest

IIoT	Industrial Internet of Things
IPS	Intrusion Prevention System
KNN	K-Nearest Neighbors
LDAP	Lightweight Directory Access Protocol
LLM	Large Language Model
LOF	Local Outlier Factor
LSTM	Long Short-Term Memory
MAS	Multi-Agent System
ML	Machine Learning
MSE	Mean Squared Error
NIST	National Institute of Standards and Technology
NLP	Natural Language Processing
NSERC	Natural Sciences and Engineering Research Council of Canada
OCEAN	Openness, Conscientiousness, Extraversion, Agreeableness, Neuroticism
PCA	Principal Component Analysis
ReLU	Rectified Linear Unit
RF	Random Forest
RoBERTa	Robustly Optimized BERT Approach
RNN	Recurrent Neural Network
RUU	Are You You
SEM	Security Event Management
SIEM	Security Information and Event Management
SIM	Security Information Management

SOC	Security Operation Center
Spark	Apache Spark
SVM	Support Vector Machine
SUTD	Singapore University of Technology and Design
TTP	Tactics, Techniques, and Procedures
TWOS	The Wolf Of SUTD
UAM	User Activity Monitoring
UBA	User Behavior Analysis
UEBA	User and Entity Behavior Analytics
VAE	Variational Autoencoder
VAST	Visual Analytics Science and Technology
XGBoost	Extreme Gradient Boosting

CHAPITRE 1 INTRODUCTION

À l'aube du deuxième quart du XXIe siècle, le paysage organisationnel mondial connaît une transformation numérique sans précédent, accélérée par des événements disruptifs ayant redéfini les modèles opérationnels traditionnels. Cette évolution s'inscrit dans un contexte où les géants technologiques exercent une influence considérable, non seulement sur l'écosystème numérique, mais également sur les sphères sociales et politiques avec l'avènement des « BigTechs » et « BigState », termes proposés par Asma Mhalla [1]. Les entreprises dominant le marché de la technologie façonnent désormais les comportements individuels, les stratégies organisationnelles et jusqu'aux politiques nationales, comme en témoignent les approches divergentes entre les États-Unis, où les relations entre dirigeants politiques et magnats de la technologie influencent la gouvernance numérique, et la Chine, qui fait de sa politique de l'information un réel artefact aux perspectives interne et externes. Au cœur de cette transition accélérée, la crise sanitaire mondiale de 2020 a catalysé l'adoption massive du télétravail qui, initialement considérée comme une solution temporaire, s'est désormais ancrée comme une composante structurelle des organisations modernes, fragmentant les périmètres de sécurité autrefois bien délimités. Une dispersion géographique des ressources humaines et une décentralisation des accès aux systèmes d'information, sont alors nés, créant alors de nouveaux défis en matière de surveillance, d'authentification et de protection des données sensibles.

Les administrations publiques, les entreprises privées et les industries sont engagées dans une transformation numérique intense, déplaçant des ressources pouvant être critiques vers des infrastructures cloud complexes et interconnectées. Cette migration, motivée par des besoins d'agilité et d'optimisation des coûts, s'accompagne d'une fragmentation des responsabilités en matière de sécurité et d'une extension de la chaîne de confiance au-delà des limites organisationnelles traditionnelles. La visibilité sur les flux de données devient cruciale, tandis que l'adoption massive de l'intelligence artificielle automatise des processus décisionnels autrefois humains, marquant un tournant dans la gestion des systèmes d'information. L'avènement des industries 4.0 et 5.0, avec l'intégration des technologies cyber-physiques à l'infonuagique et la symbiose homme-machine, brouille les frontières entre technologies opérationnelles (OT) et informationnelles (IT), redéfinissant les approches de gestion des risques et de protection des actifs

informationnels. Les infrastructures d'importance vitale se trouvent contrainte d'intégrer la connectivité à leur stratégie de sécurité rendant plus complexe les politiques efficaces de sobriété.

L'hyper connectivité qui caractérise alors ce nouveau paysage numérique, tout en offrant des gains d'efficacité considérables, engendre également une surface d'attaque exponentiellement élargie et présente donc un dilemme sécuritaire. Constitués de milliers de composants logiciels, matériels et humains interdépendants, les systèmes d'information modernes présentent une complexité sans précédent qui défie les approches traditionnelles de sécurité périmétrique. Dans cet environnement, où la distinction entre l'intérieur et l'extérieur de l'organisation devient de plus en plus floue, les acteurs malveillants exploitent avec une efficacité croissante les vulnérabilités techniques et humaines, ayant possiblement une main mise interne pour compromettre des systèmes critiques. Cette situation est d'autant plus préoccupante que les outils d'attaque, autrefois réservés à des acteurs étatiques ou à des groupes hautement spécialisés, se démocratisent grâce à l'automatisation et à l'intelligence artificielle, permettant des attaques sophistiquées à moindre coût et avec une expertise technique réduite.

Dans ce paysage numérique en constante métamorphose, marqué par une expansion sans précédent des infrastructures informatiques et une dissolution progressive des frontières organisationnelles traditionnelles, émerge une préoccupation sécuritaire majeure : la menace interne. Longtemps sous-estimée malgré sa criticité, elle constitue aujourd'hui l'un des risques cybersécuritaires les plus complexes pour les organisations modernes [2]

Historiquement focalisées sur les menaces externes, les entreprises prennent progressivement conscience d'une réalité inquiétante : les incidents de sécurité les plus coûteux, tant financièrement que réputationnellement, impliquent fréquemment des acteurs disposant d'un accès légitime aux systèmes d'information. Cette vulnérabilité intrinsèque illustre un paradoxe fondamental : les mêmes accès qui garantissent l'efficacité opérationnelle constituent également des vecteurs d'attaque privilégiés. La détection tardive de ces incidents aggrave encore le problème, avec un délai moyen de 77 jours [3], période durant laquelle les dommages s'amplifient et les traces numériques s'effacent.

L'intensification du télétravail, l'essor des services cloud et la multiplication des prestataires externes ont élargi le périmètre des "initiés" potentiels tout en réduisant la visibilité des organisations sur les comportements utilisateurs. Cette évolution s'accompagne d'une

sophistication croissante des modes opératoires, qu'ils soient malveillants, négligents ou accidentels. De plus, le coût moyen par incident a augmenté de 111 % en quatre ans [4] illustrant non seulement l'ampleur des préjudices financiers, mais aussi la complexité croissante des processus d'investigation, de remédiation et de conformité réglementaire.

Au-delà des pertes économiques quantifiables, la menace interne fragilise la confiance des partenaires, détériore le climat organisationnel et peut causer des atteintes irréversibles à la réputation institutionnelle. Face à ce risque multidimensionnel, la mise en place de stratégies de détection et d'investigation en temps réel n'est plus une option, mais une nécessité stratégique.

Si les travaux récents ont permis des avancées significatives dans l'identification des anomalies associées aux menaces internes, les solutions existantes restent entravées par plusieurs limitations structurelles. Les modèles hybrides intégrant l'analyse de graphes et l'exploitation des logs système [5] offrent une capacité de corrélation avancée, mais leur implémentation repose sur des infrastructures lourdes, rendant leur déploiement complexe, notamment pour les environnements aux ressources limitées. Par ailleurs, les approches d'apprentissage non supervisé visent une détection en quasi-temps réel, mais souffrent d'un taux élevé de faux positifs, pouvant atteindre 40 % [6], [7], un écueil qui limite leur adoption en contexte opérationnel. De leur côté, les méthodologies de modélisation comportementale [8], [9] permettent d'identifier des écarts subtils dans l'activité des utilisateurs, mais leur approche essentiellement rétrospective ne répond que partiellement aux exigences de réactivité qu'imposent les environnements critiques. Cette dépendance à l'analyse post-hoc a d'ailleurs été identifiée comme une faiblesse dans des référentiels fondateurs tels que le NIST SP 800-94 sur la détection d'intrusions [10].

Ce constat souligne la nécessité d'une approche intégrée qui dépasse ces limitations en combinant la modélisation dynamique des scénarios de menaces internes, l'identification d'indicateurs de compromission spécifiques et une méthodologie d'investigation automatisée. Une telle approche permettrait de mieux anticiper les comportements anormaux, d'accélérer la détection et de limiter le volume d'alertes non pertinentes. En intégrant ces trois dimensions, il devient possible de proposer une solution capable d'opérer dans des environnements complexes, tout en réduisant les coûts opérationnels liés aux incidents internes.

L'objectif de cette recherche est ainsi de concevoir une approche en temps réel qui réponde aux défis posés par les menaces internes contemporaines. Il s'agit non seulement d'améliorer la

précision des systèmes de détection en s'appuyant sur des techniques avancées de corrélation et d'apprentissage automatique, mais aussi d'optimiser les mécanismes d'investigation afin de réduire les délais de réponse à une menace d'origine interne pour en espérer éviter la traduction en incident.

1.1 Objectifs de recherche

Les travaux de recherche consignés dans ce mémoire proposent une contribution à l'amélioration de la détection opérationnelle de la menace interne. Il est ainsi présenté une approche en temps réel intégrée, combinant la modélisation dynamique des scénarios de menaces internes et une méthodologie d'investigation automatisée.

Cet objectif se divise en les sous-objectifs suivants :

OS1 : Cartographier et comparer les solutions de représentation et de mesure de la MI.

OS2 : Concevoir une solution permettant la détection d'anomalies en continue.

OS3 : Intégrer la détection dans une architecture hybride en vue de l'investigation des scénarios de menace interne en temps réel.

OS4 : Valider la performance temporelle de la solution mise ne place.

1.2 Structure du mémoire

Ce présent chapitre introduit le sujet de recherche de ce mémoire et est suivi de la présentation de l'état de l'art, des concepts, et travaux antérieurs et des technologies en développement ou production visant à participer à la gestion de la menace interne. Cet état de l'art, introduit les trois chapitres suivant qui constituent la contribution de ce mémoire. Le chapitre 3 présente ainsi la méthodologie et la structure globale de l'adressage de la problématique et de la solution hybride de détection et d'investigation de la menace interne proposée. Les chapitres 4 et 5 exposent les deux blocs principaux de cette solution, reprenant, pour le premier, l'article présenté à la conférence « Network and System Security 2025 » à Wuhan. Enfin le chapitre 6 ouvre la discussion sur l'évaluation de la solution complète, et le chapitre 7 conclue ce mémoire, et présente les recommandations et ouvertures proposées.

CHAPITRE 2 ÉTAT DE L'ART

2.1 Définitions

2.1.1 Menace interne

La menace interne se définit comme l'ensemble des actions ou comportements, délibérés ou non, qui compromettent ou menacent de compromettre la triade CIA (Confidentialité, Intégrité, Disponibilité) des systèmes d'information et des actifs d'une organisation, lorsque ces actions émanent d'individus disposant ou ayant disposé d'un accès légitime à ces ressources [2].

La spécificité fondamentale de la menace interne réside dans le positionnement privilégié de ses acteurs au sein de l'infrastructure organisationnelle. Ces entités possèdent une connaissance approfondie des systèmes d'information, une compréhension détaillée de l'architecture technique, et bénéficient d'accès autorisés aux ressources organisationnelles, que ce soit de manière physique ou logique, temporaire ou permanente. Cette position privilégiée représente ainsi un défi particulier pour les mécanismes de sécurité traditionnels.

La menace interne présente trois caractéristiques distinctives qui la démarquent des autres formes d'attaques cybernétiques. Elle exploite fondamentalement le vecteur d'attaque humain en s'appuyant sur des comportements imprévisibles plutôt que sur des vulnérabilités techniques, ce qui la rend particulièrement redoutable. Par ailleurs, bénéficiant d'accès légitimes au réseau interne, elle contourne efficacement les défenses périmétriques traditionnelles, permettant une infiltration silencieuse qui échappe aux systèmes conventionnels de détection d'intrusion. Enfin, sa complexité de détection est considérable puisqu'elle utilise des identifiants légitimes, rendant son identification particulièrement ardue, comme en témoignent les statistiques révélant des délais de détection pouvant atteindre plusieurs mois, période durant laquelle les dommages s'accumulent significativement.

Les impacts de ces menaces se manifestent selon quatre vecteurs principaux dont la gravité s'amplifie avec la digitalisation croissante des actifs organisationnels [11]. Le vol de propriété intellectuelle constitue un premier vecteur critique, englobant l'exfiltration de données sensibles et l'appropriation d'actifs intellectuels stratégiques. La fraude représente un deuxième vecteur significatif, comprenant la manipulation des systèmes financiers et l'exploitation des vulnérabilités des processus automatisés. L'espionnage industriel forme un troisième domaine d'impact majeur,

caractérisé par la surveillance non autorisée et la collection systématique d'informations sensibles à des fins d'avantage concurrentiel. Enfin, le sabotage représente un quatrième vecteur particulièrement destructeur, incluant l'altération d'actifs numériques critiques et la perturbation des systèmes d'information, compromettant ainsi la continuité des activités organisationnelles.

Ces caractéristiques et impacts, illustrés par des incidents récents comme ceux de Desjardins (2017-2019) et GE Aviation (2019), démontrent la nature multidimensionnelle de la menace interne et soulignent l'importance cruciale d'une approche holistique pour sa détection, son investigation et sa mitigation dans les environnements organisationnels contemporains.

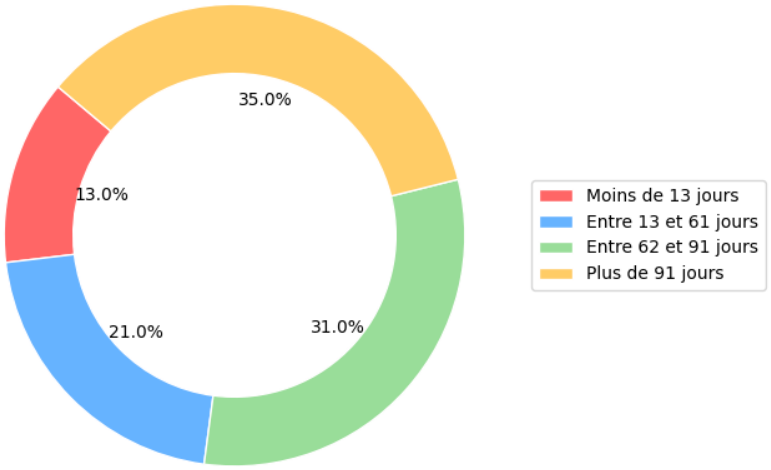

Figure 2.1 : Temps d'investigation d'une menace interne

La recrudescence de la menace interne constitue une préoccupation majeure pour les organisations contemporaines, comme en témoignent des statistiques particulièrement alarmantes. Seulement, comme explicité par la Figure 2.1, 13% des incidents liés aux menaces internes sont détectés et analysés dans un délai inférieur à un mois, laissant ainsi une fenêtre d'exploitation considérable aux acteurs malveillants. Ce constat est d'autant plus préoccupant que 71% des entreprises reconnaissent faire face à plus de 20 incidents de menace interne par an [4], soulignant l'omniprésence et la persistance de cette problématique dans le paysage cybersécuritaire actuel. Cette combinaison entre fréquence élevée des incidents et lenteur des mécanismes de détection illustre la nécessité impérieuse de développer des approches plus proactives et réactives pour l'identification et la mitigation des menaces d'origine interne.

2.1.2 Acteur interne ou « insider »

L'acteur de la menace interne, communément désigné par le terme anglophone "insider" puis "initié" en français, se définit comme une entité disposant ou ayant disposé d'accès légitimes aux ressources informationnelles, aux systèmes, aux données ou aux infrastructures physiques d'une organisation, et qui exploite ces privilèges, délibérément ou non, pour compromettre la sécurité de ces actifs. Cette définition englobante transcende la simple conception de l'individu isolé pour intégrer la complexité des écosystèmes numériques actuels.

Bien que la grande majorité des menaces internes émane d'individus spécifiques, l'évolution des infrastructures sociotechniques a élargi le spectre des acteurs potentiels. Ainsi, un initié peut se manifester sous différentes formes organisationnelles et techniques. Il peut s'agir d'un collaborateur individuel disposant d'accès directs aux systèmes d'information, mais également d'un processus automatisé compromis exploitant des privilèges légitimes pour exécuter des actions malveillantes. Dans certaines configurations complexes, une équipe entière peut constituer un vecteur de menace interne, notamment lorsque des protocoles de sécurité insuffisants régissent les accès partagés. Plus rarement, mais avec un impact potentiellement considérable, une organisation complète, telle qu'un prestataire externe ou un partenaire commercial, peut agir comme un insider à l'échelle systémique lorsque les mécanismes de gouvernance de la chaîne d'approvisionnement numérique présentent des vulnérabilités significatives.

Les motivations sous-jacentes à ces actions, dans le contexte actuel de transformation digitale, se structurent autour de trois axes principaux définis par Cole et Ring [12]. Le premier concerne les facteurs financiers. La possibilité de monétiser directement les accès privilégiés constitue un moteur essentiel, notamment par l'exploitation du marché noir des données sensibles, la vente d'informations aux concurrents sur le dark web ou encore l'utilisation de failles pour des activités de cryptomining. L'essor des crypto-monnaies favorise également ces pratiques en offrant des moyens discrets de percevoir des revenus illicites.

Le second facteur repose sur des considérations idéologiques. Certains individus opposés aux politiques de transformation numérique peuvent agir de manière délibérée afin de perturber l'évolution technologique de leur organisation. Le désaccord sur l'utilisation des données personnelles, l'activisme contre certaines pratiques, technologiques ou non, mais encore la résistance aux changements organisationnels sont autant de motifs pouvant inciter un acteur interne

à compromettre la sécurité d'un système d'information. Dans certains cas, des conflits éthiques liés aux nouvelles technologies, notamment à l'intelligence artificielle, peuvent exacerber ces tensions.

Enfin, les facteurs personnels jouent un rôle déterminant dans l'émergence des menaces internes. Le stress engendré par la transformation digitale, l'insatisfaction face aux évolutions technologiques, les tensions causées par la surveillance numérique ou encore le sentiment d'injustice lié aux politiques informatiques constituent autant d'éléments pouvant pousser un individu à adopter un comportement malveillant, qu'il soit intentionnel ou non.

La taxonomie contemporaine de la menace interne, issue des recherches académiques et industrielles, distingue trois catégories distinctes d'acteurs, selon Kim et al [13]. La première catégorie regroupe les traîtres, à savoir des acteurs internes disposant d'accès légitimes actifs et exploitant ces privilèges pour mener des actions malveillantes. Ces individus sont capables d'exploiter les vulnérabilités connues des systèmes et de contourner les mécanismes de sécurité grâce à leurs droits d'accès.

La seconde catégorie concerne les usurpateurs, qui cherchent à obtenir une élévation non autorisée de privilèges. Ces individus exploitent les failles des systèmes d'authentification, pratiquent des mouvements latéraux dans les réseaux et maîtrisent les techniques d'exploitation des vulnérabilités liées à la gestion des identités.

Enfin, la troisième catégorie englobe les initiés non-intentionnels, c'est-à-dire des utilisateurs qui, par méconnaissance ou négligence, causent des incidents de sécurité. Ces individus peuvent être victimes de campagnes de phishing ou d'ingénierie sociale, violer accidentellement des politiques de sécurité ou devenir involontairement la source d'une compromission des systèmes.

L'une des complexités majeures de la menace interne réside dans sa double nature : un insider peut non seulement être une source directe de menace, par exemple dans un contexte de vengeance ou de frustration professionnelle, mais aussi servir de vecteur d'attaque facilitant l'intrusion de menaces extérieures bien plus vastes. En effet, les cybercriminels, les États-nations ou d'autres groupes malveillants peuvent exploiter, par négligence ou mobile pécuniaire, la position privilégiée d'un insider pour déployer des attaques systémiques, contournant ainsi les défenses traditionnelles de l'organisation.

2.1.3 Scénario

Un scénario de menace interne se définit comme une séquence structurée d'actions, d'événements et d'interactions réalisés par un acteur, et dont l'exécution représente une compromission ou une menace en tant que potentiel vecteur de compromission. Cette formalisation séquentielle permet d'appréhender la dimension temporelle et contextuelle des comportements.

La représentation des scénarios de menace interne adopte une granularité variable selon les besoins analytiques. Elle peut s'articuler autour d'une session de connexion unique, révélant des anomalies comportementales à l'échelle microscopique, ou s'étendre sur une journée entière, capturant ainsi les variations des schémas d'activité sur un cycle opérationnel complet. Dans un cadre plus global, un scénario peut englober plusieurs jours, voire plusieurs semaines, permettant d'identifier les signaux faibles précurseurs et les comportements préparatoires qui précèdent souvent les incidents majeurs.

La structuration hiérarchique des scénarios constitue un aspect fondamental de leur modélisation. Un macro-scénario de compromission peut se décomposer en plusieurs sous-scénarios interconnectés, chacun ciblant un aspect spécifique de l'infrastructure informationnelle. Cette conception modulaire permet d'identifier les points de convergence entre différentes trajectoires d'attaque et d'élaborer des mécanismes de détection adaptés à chaque niveau d'abstraction. Ainsi, une exfiltration massive de données peut se décomposer en sous-scénarios d'élévation de privilèges, de reconnaissance interne, d'accès non autorisé aux bases de données et de transfert dissimulé vers des destinations externes.

Il est possible de corréler une représentation par scénarios avec les frameworks établis de modélisation des menaces externes, notamment la matrice MITRE ATT&CK qui recense exhaustivement les tactiques, techniques et procédures (TTP) des attaquants, ou encore le principe de chaine d’attaque présenté en 2.2.2.1. Une déclinaison spécifique des TTP aux menaces internes est en développement, et pourrait faire l’objet d’un travail de recherche auxiliaire. L'adaptation des schémas d'intrusion au contexte interne permettrait d'enrichir la compréhension des trajectoires d'attaque privilégiées mais cette transposition impose des ajustements significatifs pour intégrer la complexité des accès légitimes et des comportements attendus qui caractérisent fondamentalement les menaces d'origine interne.

2.1.4 Détection

Dans le contexte de la menace interne, la détection désigne l'ensemble des processus, techniques et outils mis en place pour identifier des comportements, activités ou signaux indiquant un risque. Elle repose sur l'analyse des accès aux systèmes, des modèles de comportement des utilisateurs et des tentatives de violation des politiques de sécurité.

Dans le cadre d'un SOC (Security Operations Center) cette détection plus ou moins manuelle est concrétisée par la surveillance des activités ou des événements de sécurité. La détection par un indicateur d'un incident de sécurité ouvre la voie à l'indentification de celui-ci, de son contexte et acteurs, qu'ils soient logiciels, matériels ou humains, mais cette identification de s'étend pas au-delà d'une caractérisation sommaire du sujet de la détection.

2.1.5 Investigation

L'investigation de menace interne constitue la phase analytique approfondie qui succède à la détection initiale et vise à caractériser précisément la nature, l'étendue et l'impact potentiel des comportements préalablement identifiés comme suspects. Cette démarche peut consister en la déconstruction d'un scénario détecté en ses composantes élémentaires pour isoler, au sein de la séquence d'actions observée, celles qui révèlent un potentiel significatif de compromission. L'investigation implique une évaluation qualitative et quantitative du risque associé à chaque action constitutive du scénario, permettant ainsi d'établir une hiérarchisation des menaces selon leur gravité potentielle, leur probabilité de succès et leur impact organisationnel anticipé. Cette approche analytique fournit le substrat informationnel indispensable, dans le cadre d'un SOC ou CERT, à la prise de décision concernant les actions de remédiation et les stratégies d'atténuation à déployer face à la menace identifiée. Cette importance opérationnelle absente des travaux de recherche existant motive l'intégration d'une telle composante à cette étude.

2.1.6 Temps réel

Un système temps réel est un système dont le comportement correct dépend non seulement de l'exactitude des calculs effectués, mais aussi du moment précis où ces calculs sont réalisés. Le concept de "temps réel" désigne la capacité d'un système informatique à traiter les données et à produire des réponses dans un délai suffisamment court pour influencer immédiatement l'environnement ou le processus en cours. Ces systèmes sont conçus pour interagir avec des

processus physiques en temps réel, nécessitant des réponses dans des délais strictement déterminés afin d'assurer le bon fonctionnement et la sécurité des opérations. La théorie distingue deux types de systèmes temps réel : les systèmes temps réel durs, où le non-respect des contraintes temporelles entraîne des conséquences catastrophiques, et les systèmes temps réel mous, où un dépassement de délai peut être toléré avec une dégradation des performances [14]. Dans les systèmes temps réel durs, la satisfaction des contraintes temporelles est une exigence absolue, comme dans les applications critiques de l'aéronautique ou du contrôle médical, tandis que dans les systèmes mous, tels que les applications multimédias, des variations minimes dans les délais sont acceptables [15]. La mise en œuvre de ces systèmes repose sur des mécanismes d'ordonnancement avancés et une analyse rigoureuse des échéances temporelles afin de garantir leur conformité aux exigences temporelles strictes [16].

Dans le cadre de la menace interne, les processus héritant des conséquences d'une menace peuvent être très variés et dépendants du type d'organisation concerné. Cependant, au regard des temps actuels de détection et de réponse aux cyber-incidents externe et internes, des contraintes temporelles journalières, si ce n'est plus large, pourraient se montrer suffisantes.

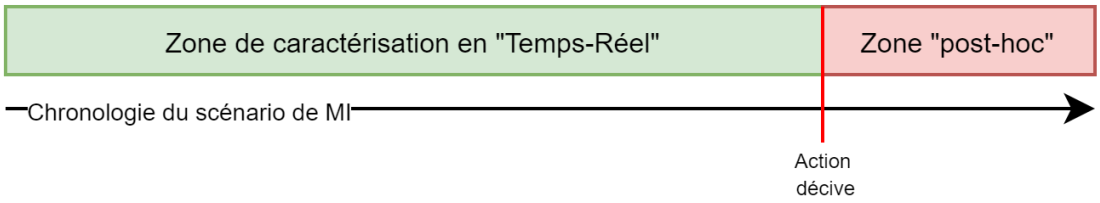

Figure 2.2 : Caractérisation en temps-réel

Il convient toutefois de souligner que l'objectif transcendant un tel projet de recherche est de prévenir plus que de détecter les actions de menace interne. Ainsi si une contrainte temporelle peut être définie, la caractérisation de la notion de temps réel dans le cadre de ce projet est d'avantage lié à une notion qualitative. En effet, une menace interne caractérisée par un scénario, orchestrant un certain nombre d'étapes allant du pré positionnement à l'action compromettante (vol, sabotage, …) ne peut être évité que si elle est identifiée et caractérisée en amont de l'occurrence d'une certaine étape décisive du scénario. Ainsi c'est la détection préalable à cette action particulière, tel qu'illustré en Figure 2.2 qui caractérisera la notion de temps-réel de ce mémoire. La caractérisation de l'action décisive devient alors clé dans le dimensionnement d'une solution dite en temps-réel et

peut être liée à la gravité ou l'irréversibilité du scénario engagé au moment de l'action. Ainsi user du changement d'étape dans une représentation comme une chaine d'attaque (2.2.2.1) pourrait caractériser cet aspect décisif.

Prenant un exemple simpliste de scénario, un employé sur le point d'être congédié fait part de son mécontentement aux ressources humaines au cours de divers échanges de courriels, avant de consulter des ressources web sur le fonctionnement d'un keylogger, puis de manipuler à travers un périphérique de stockage un exécutable de keylogging et de l'installer sur le poste d'un de ses collègues. Ainsi, si les échanges de courriels sont prémonitoires, ils ne sont pas décisifs, la manipulation d'un keylogger commence à être plus dangereuse et pourrait être considéré décisive.

2.2 Représentation de la menace interne

La représentation de la menace interne est un défi complexe et multidimensionnel. Les données impliquées sont non seulement volumineuses mais aussi extrêmement hétérogènes, englobant des traces réseau, des logs applicatifs, des considérations topologiques des organisations, des échanges de mails, des fichiers, et des données d'activité utilisateur, souvent intrusives. Cette diversité est compliquée par des contraintes éthiques et légales concernant les données personnelles, ainsi que par la nature sensible des informations internes aux organisations, qui ne peuvent souvent pas être partagées pour des raisons de confidentialité. Cette section explore les jeux de données existants pertinents pour la MI et discute des différentes méthodes de représentation de cette menace, telles que les représentations graphiques et vectorielles, afin de permettre la manipulation et l'exploitation des scénarios de MI.

2.2.1 Jeux de données

L'hétérogénéité des données à considérer pour l'étude des scénarios de menace interne, corrélée à la complexité de ces scénarios et à la jeunesse du sujet au sein de la communauté scientifique, se traduit par un nombre limité de jeux de données existants. En effet, face à l'immensité des données de toutes sortes concernant les intrusions, seuls quelques exemples proposent des données adaptées à l'étude de la menace interne.

2.2.1.1 Shonlau

Développé par Mathias Schonlau, professeur à l'Université de Waterloo, cet ensemble de données a été conçu pour étudier la détection d'usurpateurs à travers l'analyse des commandes exécutées

dans un environnement informatique [17]. Il repose sur l'observation du comportement de 50 utilisateurs aux profils variés, chacun étant associé à un fichier contenant 15 000 commandes.

L'originalité de ce jeu de données réside dans sa construction progressive : les 5 000 premières commandes de chaque utilisateur sont exclusivement issues de son activité et servent à modéliser son profil d'usage. En revanche, les 10 000 commandes suivantes intègrent un mélange d'instructions provenant tant de l'utilisateur lui-même que d'autres individus, l'objectif étant d'identifier ces dernières comme des anomalies révélatrices d'une éventuelle usurpation.

Cependant, cette approche comporte certaines limites. D'une part, elle ne prend en compte que les commandes saisies, sans offrir de visibilité sur l'ensemble des interactions qu'un utilisateur peut avoir avec son environnement numérique. D'autre part, elle repose sur l'hypothèse que les 5 000 premières commandes sont exemptes d'intentions malveillantes, alors qu'en réalité, un utilisateur légitime peut déjà adopter des comportements déviants. Enfin, l'évolution naturelle des missions d'un employé implique l'exécution de nouvelles commandes qui n'étaient pas présentes dans la phase d'apprentissage, ce qui peut entraîner un taux élevé de fausses alertes.

2.2.1.2 Are You You - RUU

Dans le cadre d'une étude conduite à l'Université de Columbia, un ensemble de données a été constitué à partir de l'activité numérique de 34 étudiants en informatique, avec leur consentement explicite [18]. Un agent logiciel, déployé sur leurs machines, a permis de collecter un volume substantiel de données, en moyenne 10 Go par étudiant, sur une période de sept jours. Afin de simuler les agissements d'un acteur interne malveillant, une expérimentation a été menée : 14 participants ont été invités, durant des sessions de 15 minutes à des moments distincts, à rechercher des informations leur conférant un avantage financier sur un système de fichiers mis à leur disposition, mais uniquement le temps de l'exercice.

Les traces ainsi recueillies offrent une vue détaillée des interactions avec le système, incluant les événements relatifs aux registres sous Windows, les processus créés ou suspendus, ainsi que leurs attributs (nom, chemin d'accès, etc.), les interactions avec l'interface et le chargement de bibliothèques dynamiques.

Toutefois, bien que ces données soient issues d'une activité réelle, leur périmètre se limite strictement à la phase d'accès aux fichiers ciblés lors de l'exercice. Elles n'intègrent ni

l'exploitation ultérieure des informations récupérées par les étudiants, ni la possibilité qu'un utilisateur manipule ces fichiers dans un but légitime. L'hypothèse sous-jacente considère ainsi que toute interaction enregistrée relève d'une démarche malveillante.

2.2.1.3 TWOS

Le jeu de données TWOS (The Wolf Of SUTD) trouve son origine dans une compétition organisée en mars 2017 par l'Université de Technologie et de Design de Singapour. Conçu pour capturer des manifestations réalistes de menaces internes malveillantes, cet événement a réuni six équipes de quatre étudiants, engagées dans une confrontation intense s'étalant sur cinq jours. Afin d'assurer une traçabilité fine de leurs actions, plusieurs agents de collecte ont été déployés pour enregistrer une multitude d'événements : mouvements de souris, frappes au clavier, exécutions de processus, modifications des fichiers système, flux réseau, échanges de courriels, ainsi que les connexions et déconnexions aux machines.

L'ensemble de ces enregistrements représente un total de 320 heures de données, dont 18 heures spécifiquement attribuées à des comportements d'usurpation, avec deux occurrences distinctes d'un acteur interne adoptant une posture malveillante [19]. En complément des traces techniques, le jeu de données intègre également des réponses à des questionnaires psychologiques, permettant de mieux appréhender les traits de personnalité des participants.

Toutefois, le cadre compétitif dans lequel ces données ont été générées s'éloigne considérablement des dynamiques observables en entreprise. Les interactions relevées, bien que riches en informations, traduisent une recherche de performance et d'affrontement stratégique propre au contexte du défi, plutôt que des comportements typiques d'un employé dans son environnement de travail. De plus, la majorité des actions capturées concerne des opérations strictement locales sur les postes utilisateurs, sans prise en compte des interactions avec des serveurs distants, bases de données ou autres systèmes critiques où résident généralement des informations sensibles. Par ailleurs, aucun élément contextuel ne précise ni la nature des actifs consultés ni les privilèges effectifs des utilisateurs.

2.2.1.4 VAST Challenge 2009

Le VAST Challenge 2009 s'inscrit dans une série d'événements annuels destinés à évaluer les capacités d'analyse de données massives en contexte de cybersécurité et de renseignement.

L'édition de 2009 s'est concentrée sur la détection de comportements suspects au sein d'une organisation fictive, en simulant des menaces internes à travers un jeu de données synthétique, mais conçu pour refléter des scénarios réalistes.

Les données collectées couvrent une large gamme d'interactions numériques : journaux de connexion et de déconnexion aux systèmes, échanges de courriels, requêtes vers des bases de données, transferts de fichiers et activités web des employés simulés. Ce corpus a été élaboré de manière à contenir des anomalies subtiles, suggérant la présence d'un acteur interne cherchant à exfiltrer des informations sensibles. L'objectif du challenge était d'amener les participants à identifier ces signaux faibles en s'appuyant sur des techniques de visualisation et d'analyse comportementale [20].

Toutefois, si ce jeu de données fournit une opportunité précieuse d'étudier la détection des menaces internes, il reste marqué par les contraintes inhérentes aux environnements simulés. Les comportements enregistrés, bien qu'inspirés de cas réels, sont artificiellement modélisés et ne reflètent pas nécessairement la complexité des actions d'un employé évoluant dans un cadre organisationnel réel. De plus, la structuration des logs et la présence explicite d'anomalies peuvent influencer l'approche des analystes en les guidant vers des schémas d'attaque préconçus.

2.2.1.5 PicoDomain

PicoDomain [21] est un jeu de données compact et réaliste proposé en 2020 par l'Université George Washington. Il est centré sur une campagne d'intrusion externe dans un environnement Windows simulé, avec des logs Zeek couvrant trois jours d'activité malveillante. Contrairement à des datasets comme CERT ou LANL, il offre un « ground truth » détaillé et des artefacts réseau de haute-fidélité (Kerberos, SSL, SMB), incluant des techniques avancées comme le *domain fronting*, les mouvements latéraux via DCOM/WMI, et l'escalade de privilèges (Hot Potato). Bien que conçu pour la détection d'intrusions, son réalisme opérationnel (ex. : corrélation entre pics d'authentification Kerberos et compromissions de comptes) et sa taille réduite (environ 500 000 logs) en font un outil utile pour le prototypage rapide d'algorithmes de détection, notamment en apprentissage non supervisé. Cependant, sa focalisation sur les attaques externes (compromission initiale via un dropper WinRAR, C2 via PowerShell Empire) le rend peu adapté à l'étude des menaces internes, où les comportements malveillants émergent souvent de comptes légitimes sans phase de compromission initiale.

De plus, PicoDomain souffre de limitations structurelles pour notre contexte : l'absence de logs hôtes (ex. : événements Windows, accès aux fichiers) et de comportements internes malveillants (ex. : exfiltration par un employé autorisé, abus de privilèges) réduit son utilité. Bien que les auteurs démontrent son efficacité pour détecter des anomalies via des méthodes comme LOF (Local Outlier Factor), le jeu de données ne capture pas les subtilités des menaces internes, comme les déviations progressives de comportement ou les activités légitimes détournées. Son environnement simulé (5 machines, trafic DHCP) et son manque de diversité scénaristique (un seul adversaire, pas de bruit organisationnel) en font un outil complémentaire plutôt qu'un choix principal malgré sa qualité technique.

2.2.1.6 CERT

Le jeu de données CERT (Computer Emergency Response Team) développé par le CERT Coordination Center de l'Université Carnegie Mellon (CMU) est une ressource incontournable pour la recherche sur les menaces internes. Il offre des scénarios synthétiques mais réalistes d'activités malveillantes au sein des organisations [22].

Le projet CERT a débuté en 2009 pour combler un vide crucial dans la recherche sur les menaces internes : l'absence de données empiriques exploitables. Les organisations victimes de menaces internes étaient réticentes à partager leurs données réelles, rendant difficile le développement de solutions efficaces. Le CERT Insider Threat Center a donc créé un ensemble de données synthétiques reflétant fidèlement les comportements normaux et anormaux des utilisateurs.

Le projet ADAMS (Anomaly Detection At Multiple Scales) de la DARPA (Defense Advanced Research Projects Agency) et le FBI (Federal Bureau of Investigation) ont joué un rôle crucial dans l'origine et le développement du jeu de données CERT. Lancé en 2011, ADAMS visait à développer des méthodes pour détecter les comportements anormaux pouvant indiquer des menaces potentielles. La DARPA a financé une partie du développement des versions ultérieures du jeu de données CERT, alignant ainsi les objectifs des deux initiatives pour améliorer la détection des menaces internes basées sur l'analyse comportementale.

Le jeu de données est maintenu et a évolué de manière significative selon les versions présenter dans le Tableau 2.1 et son contenu est détaillé dans le Tableau 2.2.

Tableau 2.1 : Évolution du jeu de données - CERT

Version	Année	Caractéristiques
r1	2010	Première version avec des logs d'activités simulées d'une petite organisation fictive.
r2	2011	Extension avec davantage de scénarios de menaces internes et amélioration du réalisme des données.
r3	2013	Intégration de données plus diversifiées, incluant des logs de systèmes de contrôle d'accès physique.
r4	2014	Ajout de scénarios plus complexes et amélioration de la granularité des données.
r5	2015	Incorporation de données volumineuses sur plusieurs mois d'activité organisationnelle.
r6	2016	Enrichissement avec des scénarios multi-étapes et des comportements plus sophistiqués.
r6.2	2018	Dernière version majeure publique, incluant des améliorations basées sur les retours d'expérience.

Tableau 2.2 : Types de données – CERT r4.2

Nomenclature	Description
logon.csv	Connexions réussies et échouées aux systèmes.
devices.csv	Connexion et utilisation de périphériques USB, CD, etc.
http.csv	Sites visités, durée des sessions, téléchargements.
email.csv	Métadonnées des courriels envoyés et reçus (expéditeur, destinataire, objet, taille des pièces jointes).
file.csv	Accès, création, modification et suppression de fichiers.
psychometric.csv	États émotionnels simulés des employés, stress, satisfaction au travail (dans certaines versions).
Ldap (dossier)	Structure hiérarchique, départements, rôles.

Cinq scénarios réalistes de MI sont présents dans les différentes versions de ce jeu de donnée [22]:

- Un employé qui n'utilisait pas auparavant de périphériques amovibles ni ne travaillait en dehors des heures de bureau commence à se connecter après les heures de travail, utilise un périphérique amovible et télécharge des données sur le site wikileaks.org. Peu de temps après, il quitte l'organisation.

- Un employé commence à consulter des sites d'emploi et à solliciter des postes chez un concurrent. Avant de quitter l'entreprise, il utilise une clé USB (à un rythme nettement plus élevé que son activité précédente) pour voler des données.

- Un administrateur système devient mécontent. Il télécharge un enregistreur de frappe et utilise une clé USB pour le transférer sur l'ordinateur de son superviseur. Le lendemain, il utilise

les journaux de frappe collectés pour se connecter en tant que son superviseur et envoyer un courriel de masse alarmant, semant la panique dans l'organisation. Il quitte immédiatement l'organisation.

- Un utilisateur se connecte à l'ordinateur d'un autre utilisateur et recherche des fichiers intéressants, qu'il envoie ensuite à son adresse courriel personnelle. Ce comportement devient de plus en plus fréquent au cours d'une période de trois mois.

- Un membre d'un groupe décimé par des licenciements télécharge des documents sur Dropbox, prévoyant de les utiliser à des fins personnelles.

Ce jeu est beaucoup cité et permet une comparaison objective des différentes approches de détection et représente des comportements réalistes avec une vérité terrain connue, facilitant ainsi l'évaluation des algorithmes. Son volume significatif le prédispose à des approches par apprentissage automatique, tout en couvrant un large éventail de types de menaces internes. Un des atouts majeur qu'il est important de noter et sa structure organisationnelle complète et réaliste, illustrée en Figure 2.3, essentielle pour mettre en place et évaluer une solution voulue opérationnelle.

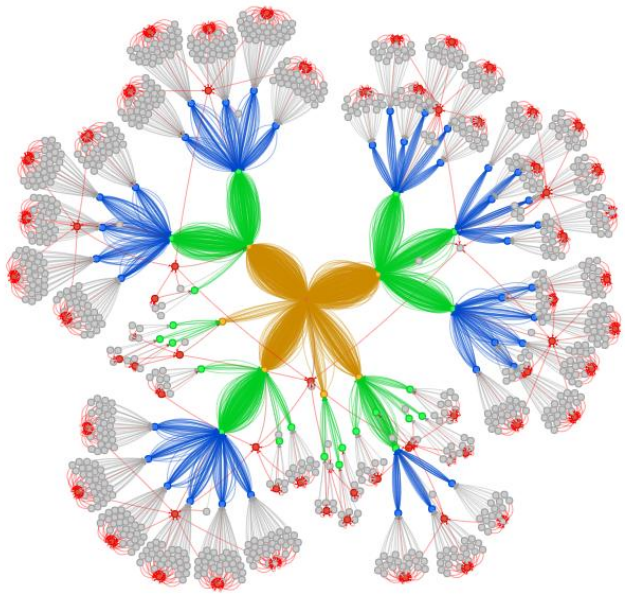

Figure 2.3 : Graphe organisationnel - CERT v4.2

En effet, l'activité de plus de 1000 employés est transcrite, et ces employés sont organisés avec des unités d'affaires (en orange dans la Figure 2.3), des unités fonctionnelles (nœuds verts), des équipes (nœuds bleus), des responsables (nœuds en rouge), et des rôles. Chaque équipe est aussi au sein

d'une unité fonctionnelle, elle-même au sein d'une unité d'affaires. Par exemple, dans la Figure 2.4, *Alan Benjamin Holder* est *Assembly Supervisor* au sein du quatrième département d'assemblage (son équipe) et fais partie de l'unité fonctionnelle *Manufacturing*. Son supérieur directe est hors de la figure mais est représenté par la flèche rouge.

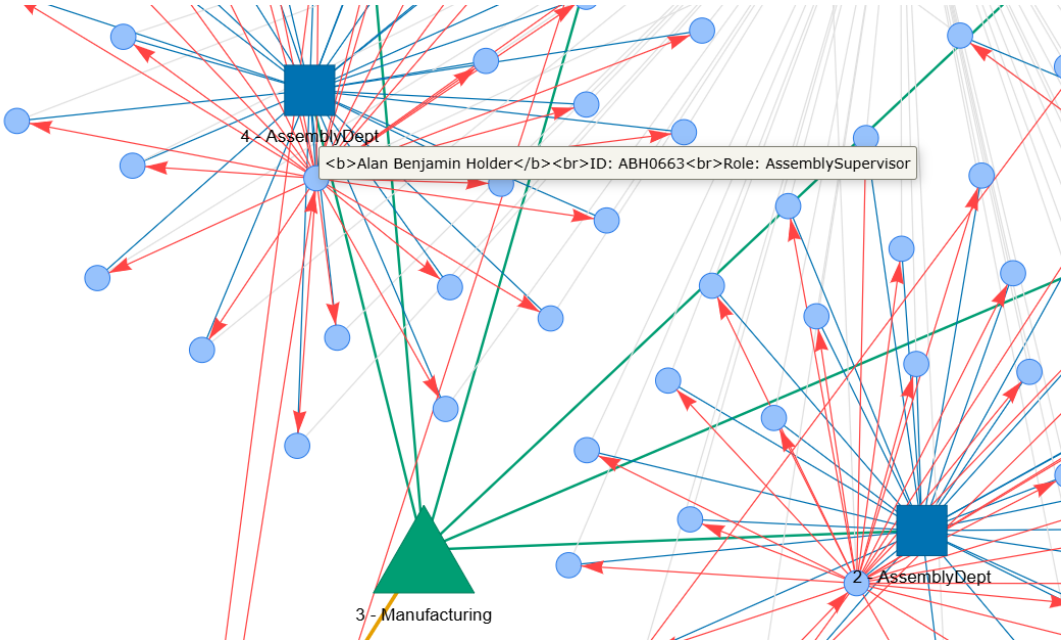

Figure 2.4 : Détail des relations hiérarchiques

Comparé aux jeux de données présentés antérieurement, le jeu de données CERT se distingue par sa diversité élevée des données et son excellent contexte organisationnel, tel que le pointe le Tableau 2.3 comparatif.

Tableau 2.3 : Comparaison des jeux de données

Jeu de Données	**Réalisme**	**Volume**	**Diversité des Données**	**Contexte Organisationnel**	**Scénarios de MI**	**Actualité**
CERT	Bon	Élevé	Très élevée	Excellent	Nombreux et variés	Bon (2018)
Shonlau	Moyen	Moyen	Moyenne	Bon	Variés	Mauvais (2000)
RUU	Moyen	Faible	Moyenne	Bon	Peu nombreux	Moyen (2011)
TWOS	Moyen	Moyen	Moyenne	Bon	Variés	Bon (2017)
VAST	Moyen	Faible	Moyenne	Bon	Très peu nombreux	Faible (2009)

De plus, ces données ont été spécifiquement conçues pour l'étude des menaces internes. Il est accompagné d'une documentation détaillée sur les scénarios, facilitant l'évaluation des algorithmes, et a bénéficié d'améliorations continues sur plusieurs années. Cependant, il reste moins authentique que des données réelles et offre moins de détails au niveau système d'exploitation.

Parmi les ensembles de données étudiés, le jeu de données CERT (version 4.2) s'impose comme le plus adapté à notre recherche en raison de sa structure temporelle séquentielle et de la diversité de ses sources (journaux de connexion, activités réseau, métadonnées organisationnelles), essentielles pour tester des algorithmes de détection en temps réel dans un cadre opérationnel. Contrairement à des corpus comme Shonlau ou RUU, limités à des interactions ponctuelles, il intègre des scénarios évolutifs au sein d'un contexte organisationnel réaliste (hiérarchie, rôles, comportements), permettant une modélisation crédible des menaces internes. Bien que sa vérité terrain soit documentée, les scénarios manquent de détails analytiques, et le réalisme se trouve impacté par certains manquements. Malgré ces contraintes, son équilibre entre réalisme, volume et compatibilité avec les méthodes non supervisées (GNN, LSTM), couplé à son adoption généralisée dans la littérature, en fait le choix optimal pour nos expérimentations.

2.2.2 Modèles de représentation

L'étude de la menace interne nécessite non seulement des jeux de données adaptés, mais aussi des modèles de représentation efficaces pour analyser et interpréter ces données complexes. La diversité et la nature hétérogène des informations impliquées dans les scénarios de menace interne rendent indispensable le développement de modèles capables de capturer les relations subtiles, pour certaines abstraites dont résultent les comportements des individus au sein des organisations. Cette sous-section explore le travail de la communauté scientifique à ce jour concernant approches de modélisation, allant des représentations graphiques aux vectorielles, en passant par des modèles pensés pour l'apprentissage automatique. Chaque méthode présente ses avantages et ses défis, et leur compréhension est cruciale pour définir un modèle permettant l'investigation de la menace interne en temps-réel.

2.2.2.1 Notion de chaine d'attaque

La notion de chaine d'attaque ou « kill-chain » consiste en un modèle conceptuel qui décrit les étapes successives d'une attaque informatique, depuis la reconnaissance initiale jusqu'à l'atteinte

des objectifs finaux par les attaquants. Initialement développé par Lockheed Martin et schématisée dans la Figure 2.5, cette modélisation a plusieurs utilités, de la structuration des mécanismes d'attaque pour leur compréhension à l'élaboration de stratégies de défense adaptées. Ce modèle permet l'étude d'une attaque des phases préparatoires à l'atteinte de l'objectif des attaquants.

Les versions classiques d'une chaine d'attaque présentent sept catégories suivantes, dans lesquelles égrainer les différentes actions, outils ou contextes permettant aux attaquants d'arriver à leurs fins.

Reconnaissance : étape de préparation de l'attaque, les attaquants y collectent des informations sur la cible pour identifier les vulnérabilités exploitables. Cette étape peut inclure la surveillance des réseaux, l'analyse des configurations système et la collecte de données publiques, ou privées s'ils en disposent.

Armements : les attaquants développent ou adaptent des outils malveillants spécifiques aux vulnérabilités identifiées. Cette phase peut impliquer la création de malwares personnalisés ou l'adaptation de logiciels malveillants existants à la stratégie d'attaque définie.

Livraison : l'attaque est déployée, cela consiste en l'utilisation de vecteurs tels que du phishing, des sites web compromis ou des supports amovibles infectés. L'objectif est de transmettre le logiciel malveillant à la cible, ou d'y insérer des outils d'attaques.

Exploitation : les vulnérabilités identifiées sont exploitées pour exécuter les outils malveillants sur le système cible. Cette phase peut inclure par exemple l'exploitation de failles logicielles ou l'utilisation de techniques d'ingénierie sociale.

Installation : établissement d'un accès persistant, afin de pouvoir mener à bien la suite de l'attaque, dans le temps, sans voir sa porte d'entrée révoquée. Sont dans cette phase des étapes telles que la création de comptes utilisateurs ou l'installation de portes dérobées.

Commande et contrôle (C2) : les attaquants établissent une communication avec le système compromis pour envoyer des commandes et recevoir des données. Cette phase est cruciale la gestion des opérations malveillantes.

Actions sur les objectifs : Les attaquants atteignent leurs objectifs finaux, qui peuvent inclure le vol de données sensibles, la perturbation des opérations ou l'extorsion financière.

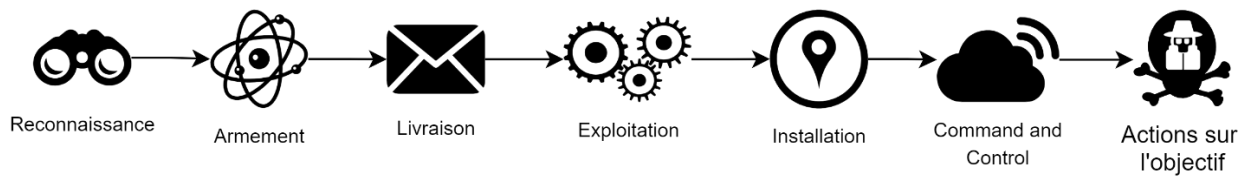

Figure 2.5 : Chaine d'attaque - Lockheed Martin

Au fur et à mesure des progrès en sécurité informatique, cette modélisation a bénéficié de contributions pour l'enrichir et la généraliser à l'ensemble des stratégies d'attaques rencontrées ou connues, comme en témoigne une seconde version « unifiée » illustrée en Figure 2.6 et liée à la cartographie des Techniques, Tactiques et Procédures par Mitre (cf. 2.2.2.6).

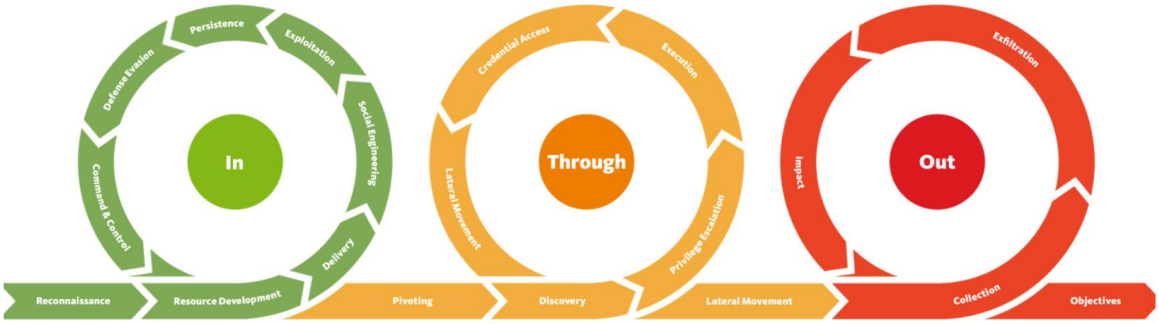

Figure 2.6 : Chaine d'attaque unifiée - [23]

Ainsi la chaine d'attaque dans ses différentes version est un outil crucial pour comprendre les méthodes et stratégies d'attaques et mettre en place des stratégies de défense, préventives, détectives, dissuasive ou réactives. Cependant, d'après plusieurs acteurs de la cybersécurité, cette chaine bien qu'applicables aux attaques « classiques » venant de l'extérieur ne semble pas être adaptée à la modélisation de la menace interne. Ainsi, lors d'une conférence BlackHat en 2013, le FBI propose une version de la chaîne d'attaque spécifique à la menace interne comme présentée par la Figure 2.7.

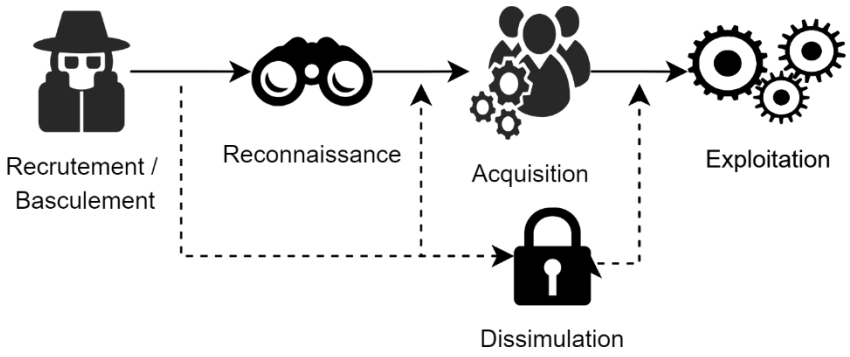

Figure 2.7 : Chaine d'attaque - Menace Interne

Recrutement ou Points de Basculement : les attaquants peuvent être malveillants dès le début ou après un événement majeur. Cette étape, souvent négligée, implique la vie privée des individus, et peut être ouverte dès les entrevues de l'individu avec les équipes des ressources humaines de l'entreprise ou de ses prestataires. Par exemple, identifier des événements comme un licenciement, un divorce ou des problèmes de dépendance peut aider à repérer les employés à haut risque. Ces événements renforcent la crédibilité d'autres alertes potentielles. Par exemple, un employé en difficulté financière pourrait manipuler les actifs de l'entreprise.

Reconnaissance : cette phase fournit des avertissements précoces et renforce la confiance lorsque les alertes sont corrélées. L'attaquant cherche à localiser les données ciblées et tente d'y accéder. Cette étape est cruciale pour anticiper les actions malveillantes.

Acquisition ou Accumulation : l'insider utilise des techniques manuelles et automatiques pour récupérer et centraliser les données identifiées. Cela peut inclure des logiciels spécialisés ou des méthodes discrètes pour éviter la détection.

Exécution : l'insider met son plan à exécution, par exemple en transférant les données vers des emplacements hors du contrôle de l'entreprise. Cette phase concrétise les intentions malveillantes, nécessitant une vigilance accrue pour détecter et contrer ces actions.

Dissimulation : l'insider, sans ordre vis-à-vis des autres étapes de la chaine d'attaque emploi des méthodes comportementales ou techniques pour camoufler sa déviance en tant que menace interne

Cette réduction et adaptation de la chaîne d'attaque cyber usuelle est semblablement adoptée par la coopérative bancaire Québécoise Desjardins, qui opte pour les cinq étapes suivantes : reconnaissance, privilège, accumulation, exécution et masquage. Cette segmentation, à l'intersection entre les deux modèles précédents apporte une nouvelle catégorie autour de la dissimulation des activités malveillantes, qui peut être non-négligeable dans un contexte interne ou l'employé ayant un accès privilégié peut agir dans l'ombre pour le conserver.

2.2.2.2 Simple flux d'évènements

Le fonctionnement actuel de la plupart des entreprises disposant d'un département de sécurité informatique repose sur des équipes d'analyste plus ou moins épaulés par des solutions informatiques. Ces équipes d'analystes forment un SOC, qui reçoit les évènements des systèmes informatiques supervisés. Chaque analyste classe alors les évènements qu'ils reçoit, et décide de la procédure à suivre lors d'une suspicion d'incident. La Figure 2.8 présente un schéma très simplifié de la collection de ces évènements, pour une usine réelle. Le flux d'évènement peut rapidement être insurmontable, et l'hétérogénéité de ceux-ci rend difficile un jugement humain, d'autant plus lorsqu'il s'agit de corréler des évènement qui s'entrecroise pour reconstruire et analyser des scénarios. Ainsi si des outils existent pour aider les analystes dans cette tâche, il n'en demeure pas moins l'importance de définir un système de représentation de l'activité des utilisateurs en fonction d'un flux d'événements, d'autant plus que la menace interne leur échappe par sa complexité.

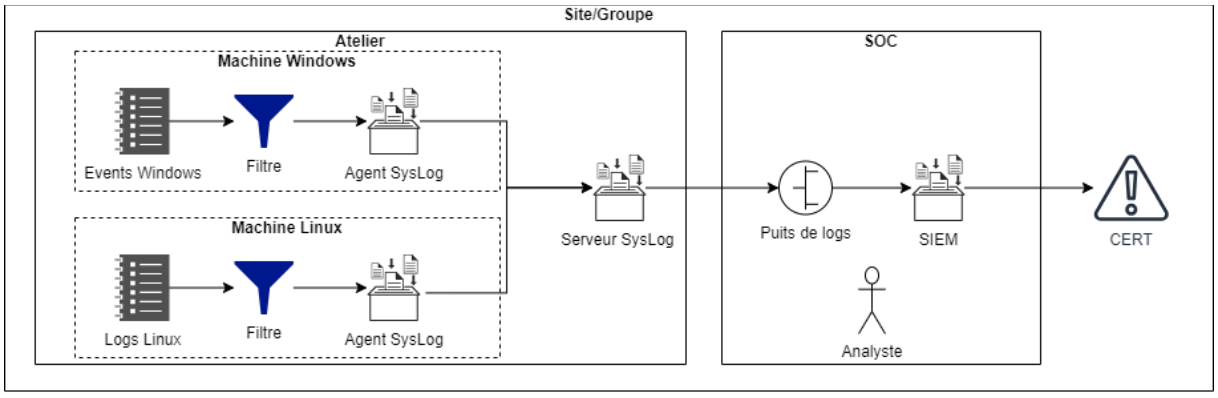

Figure 2.8 : Exemple du chemin des évènements pour une usine

2.2.2.3 Représentation vectorielle

Représenter le comportement d'un utilisateur comme la suite de ses activités informatique, c'est-à-dire son activité réseau et applicative, concentrée par les centres de gestion des informations et

des événements de sécurité (SIEM) est une première approche sobre et facile à comprendre. L'activité d'un utilisateur est donc résumée à un ou plusieurs vecteurs, dépendamment de la granularité appliquée, représentant l'ensemble chronologique de tous ses mouvements et actions [24]. Ainsi, l'application de méthodes communes à la détection d'intrusion comme la veille d'action interdite ou identifiée est possible. Cependant, cette approche simpliste ne prend pas en compte la spécificité des acteurs internes, pouvant constituer une menace à partir d'une suite d'action considérées légitimes.

L'analyse de groupement d'actions et donc des scénarios dans leur ensemble est nécessaire, et peut aussi être basée sur ces représentations vectorielles. En effet, la caractérisation des actions d'un scénario, comme décrite en 2.3.3.7 permet l'utilisation d'outils d'intelligence artificielle pour analyser des scénarios dans leur ensemble. En particulier, l'utilisation de modèles de type Word2Vec, permet de construire un contexte propice à l'utilisation de Natural Language Processing (NLP) permettant une détection contextuelle des anomalies [25]. Ainsi si une « simple » représentation de l'activité utilisateur ne semble pas suffisante pour saisir et analyser des scénarios comportementaux complexes, elle peut tout de même être intégrée au sein d'une modélisation plus riche, en tant que formalisme de prédilection pour certains outils.

2.2.2.4 Représentation par graphe

L'objectif d'un modèle de représentation des scénarios efficace est la représentation des interactions et des comportements au sein d'un système. Les graphes, en tant que structures mathématiques, offrent une méthode puissante pour modéliser ces interactions de manière visuelle et analytique. En particulier, la menace interne, présente des défis uniques en raison de la complexité des relations et des privilèges d'accès.

Les graphes permettent de capturer ces relations complexes en représentant les entités (comme les utilisateurs, les dispositifs, ou les ressources) sous forme de nœuds et les interactions entre elles sous forme d'arêtes. Cette représentation permet non seulement de visualiser les chemins potentiels qu'un attaquant pourrait emprunter, mais aussi d'analyser les comportements anormaux qui pourraient indiquer une activité malveillante [26].

Graphes topologiques

Les graphes topologiques constituent une représentation fondamentale des réseaux informatiques, mettant en évidence les connexions physiques et logiques entre les différents nœuds du réseau. Dans le contexte de la menace interne, ces graphes jouent un rôle crucial en fournissant une vue d'ensemble de la structure du réseau, ce qui permet d'identifier les points de vulnérabilité potentiels et les chemins d'attaque possibles.

Les nœuds représentent les entités du réseau, telles que les ordinateurs, les serveurs, les routeurs, et autres dispositifs connectés. Chaque nœud peut être identifié par une adresse IP, un nom d'hôte, ou un autre identifiant unique. [27]

Les arêtes quant à elles représentent les connexions entre les nœuds. Elles peuvent être directionnelles ou non, indiquant la nature de la relation entre les entités. Par exemple, une arête peut représenter une connexion physique (comme un câble Ethernet) ou une connexion logique (comme une session de communication). Cette topologie n'est pas nécessairement physique mais peut être relationnel, représentant les employés d'une entreprise, les différentes unités, départements et équipes, reliés par leurs relations d'appartenance ou de subordination, comme illustré en Figure 2.3 représentant la structure organisationnelle du jeu de données du CERT.

Dans le cadre de la menace interne une représentation purement topologique et basée sur le réseau ne semble pas suffisante en tant qu'elle ne permet pas forcément de saisir l'ensemble des éléments d'un scénario de menace interne comme des mails échangés, ou le contenus de fichiers consultés. Cependant le principe de représentation de chaque élément d'un système par des nœuds (utilisateur, machine, mais aussi fichier, périphérique, …) offre à une représentation topologique généralisée un potentiel intéressant pour modéliser un scénario de menace interne [28].

Une autre évolution des graphes topologique est la considération de plusieurs niveaux d'abstraction, et donc de plusieurs couches. Par exemple, une couche pourrait représenter les interactions physiques entre les dispositifs (graphe topologique « classique »), tandis qu'une autre couche pourrait modéliser les flux de données logiques, la politique d'accès ou les échanges de courriels. Cette approche est ainsi particulièrement pertinente, car elle permet de capturer la complexité des interactions.

Graphes attribués

Les graphes attribués constituent une avancée par rapport aux graphes topologiques traditionnels, en intégrant des informations supplémentaires sous forme d'attributs associés aux nœuds et aux arêtes.

Ils permettent de modéliser des réseaux complexes où chaque nœud et chaque arête est décrit par un ensemble d'attributs. Ces attributs peuvent inclure des informations telles que les rôles des utilisateurs, les types de dispositifs, les logs d'activité, et d'autres métadonnées pertinentes.

Gamachchi & Boztas mettent en avant certains avantages concrets à l'utilisation d'attributs dans la construction de graphes pour la représentation de scénarios de menace interne [29] :

> Les attributs ajoutent une dimension contextuelle qui permet une analyse plus fine des comportements des utilisateurs. Par exemple, un utilisateur qui accède à des ressources sensibles en dehors de ses heures de travail habituelles peut être facilement identifié grâce aux attributs temporels associés à ses actions.
>
> En comparant les attributs des nœuds avec des modèles de comportement normal, il est possible de détecter des anomalies qui pourraient indiquer une menace interne. Cette approche est particulièrement efficace pour identifier des comportements qui dévient des normes établies dans un groupe ou une communauté d'utilisateurs.
>
> Les graphes attribués permettent d'intégrer des indicateurs de compromission (IoC) directement dans le modèle de graphe. Cela facilite l'identification rapide des nœuds et des arêtes associés à des activités suspectes, permettant une réponse plus rapide aux incidents de sécurité.

Graphes d'attaque

S'il est donc clair qu'une attaque issue d'une menace interne est un scénario complexe, proche d'un scénarios d'utilisation normal des actifs informatiques. La difficulté principale étant ainsi de différencier les scénarios d'attaques parmi les scénarios normaux, représenter lesdits scénarios sous-formes de graphes, ou assurer un suivi de ceux-ci à l'aide de graphe pourrait-il permettre d'adresser cette difficulté ?

En cybersécurité, les graphes d'attaques modélisent les différentes étapes qu'un attaquant pourrait suivre pour atteindre un objectif malveillant. Ils peuvent être structurés de différentes manières, représentant le réseau d'une organisation avec tous ces actifs physiques ou logiciels afin de suivre ou prédire la progression d'une compromission identifiée, liant des vulnérabilités connues pour visualiser d'éventuels chemins d'attaques, ou encore représentant les différentes étapes d'une attaque avérée ou éventuelle, en se basant par exemple sur les TTP de Mitre (cf. 2.2.2.6).

Adapter ces graphes à la menace interne implique de prendre en compte les spécificités des acteurs internes, tels que leurs privilèges d'accès et leur connaissance du système. Bien que peu de travaux académiques se concentrent actuellement sur cette adaptation, l'intérêt pour cette approche est croissant, notamment en raison de son potentiel pour la détection d'intrusion et la remédiation. [30]

Une telle représentation peut être le support d'algorithmes de prédiction de chemins d'attaques, ou encore de propositions de remédiations sur la base de la connaissance générale (comme les rapports CVE) et la connaissance spécifique (inventaire des actifs d'une organisation) comme le proposent Kéren et al. [31]

2.2.2.5 Utilisation d'ontologies

D'origine philosophique, l'ontologie est la théorie de la nature de l'être, introduite pour décrire le monde réel et résoudre les problèmes sémantiques entre entités hétérogènes. Utilisée en sciences appliquées, en particulier en informatique et en biologie, sa définition évolue pour inclure de nouvelles perspectives. Une ontologie est une "spécification explicite, formelle et partagée d'une conceptualisation d'un domaine d'intérêt" [32]. La conceptualisation consiste à modéliser un phénomène en identifiant ses concepts clés. La spécification doit être explicite en tant que les types de concepts et leurs contraintes sont définis de manière claire. Sa formalité indique que ces concepts peuvent être interprétés par des machines. Enfin, la notion de partage implique que l'ontologie est acceptée et utilisée par un groupe d'individus. Ainsi l'intérêt principal d'une ontologie est de proposer une représentation pertinent, basée sur un vocabulaire commun permettant une conceptualisation partagée de l'information entre machines et humains.

Une ontologie se compose de quatre éléments principaux : Concept, Instance, Relation et Axiome. Le Concept, ou Classe, est un élément fondamental du domaine, comparable à une classe d'objets en programmation orientée objet (comme un animal ou un véhicule). L'Instance est un individu spécifique d'un concept, par exemple, un chat est une instance du concept "animal". La Relation

démontre la liaison entre deux concepts, le premier étant le domaine et le second la portée. Enfin, l'Axiome impose des contraintes sur les concepts et relations pour assurer la cohérence de l'ontologie.

En cybersécurité, les ontologies jouent un rôle crucial en fournissant un cadre structuré pour représenter et analyser les menaces et les vulnérabilités. Elles permettent de standardiser les terminologies et les concepts utilisés par les différents outils et équipes de sécurité, facilitant ainsi la communication et la collaboration [33]. Par exemple, une ontologie peut définir des concepts tels que "attaque", "vulnérabilité", et "contre-mesure", ainsi que les relations entre ces concepts. Ainsi, un des outils de détections d'intrusion peuvent bénéficier d'une connaissance mise à jour automatiquement sur la base d'ontologies partagées, comme cela est proposé pour l'enrichissement de graphes d'attaques [34].

Dans le cadre plus précis de la menace interne, une ontologie peut ainsi être le support d'outils automatisés comme le propose Saint-Hilaire [34] pour les graphes d'attaques. Les travaux de la division CERT de l'Université de Carnegie Mellon, dont la Figure 2.9 présente un extrait de l'ontologie proposée, ou ceux d'Amine Badaoui [8], [35] proposent ainsi une approche ontologique de la menace interne et des indicateurs associés. Ils reprennent la base d'ontologie plus génériques de sécurité informatique, mais tronquée et étendue pour transcrire l'importance comportementale du domaine. Alors une ontologie, adaptée et partagée par définition semble être un support de prédilection pour modéliser la menace interne, quel qu'en soit le format numérique.

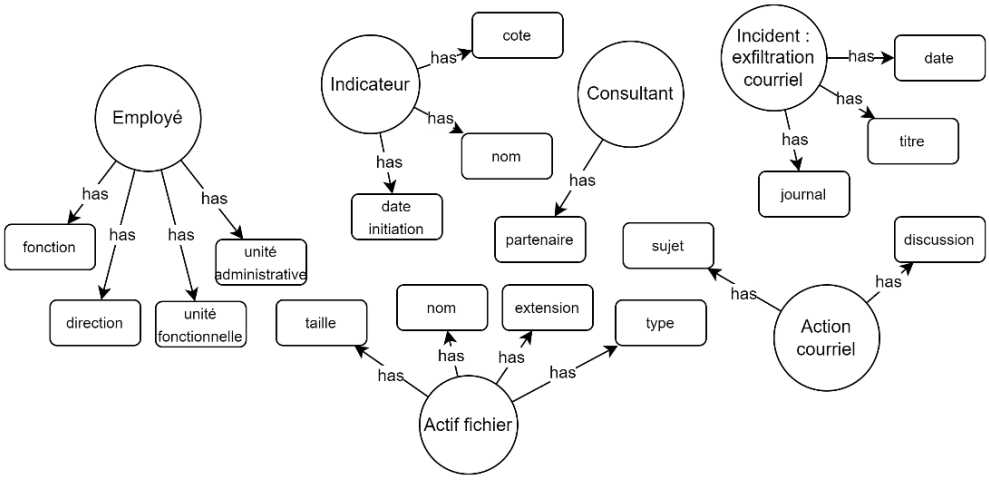

Figure 2.9 : Exemple de classes et leurs attributs de l'ontologie du CERT

2.2.2.6 Représentation par TTP

La représentation par Techniques, Tactiques et Procédures (TTP) constitue un cadre méthodologique essentiel pour analyser et comprendre les comportements des attaquants en cybersécurité. Cette approche, largement développée par le MITRE ATT&CK, permet de décomposer les actions malveillantes en trois niveaux distincts : les techniques spécifiques employées, les tactiques ou objectifs stratégiques poursuivis, et les procédures opérationnelles mises en œuvre. En structurant ainsi les informations, les équipes de sécurité peuvent mieux anticiper, détecter et répondre aux menaces, en s'appuyant sur une base de connaissances partagée et constamment mise à jour. Le travail de MITRE dans ce domaine est particulièrement notable, offrant une taxonomie détaillée des comportements adverses qui s'enrichit quotidiennement de nouvelles données et expériences, comme l'intégration des notions de résilience [33]. Cette taxonomie permet non seulement de cataloguer les actions des attaquants, mais aussi de les contextualiser dans des scénarios d'attaque complets, facilitant ainsi la mise en place de contre-mesures adaptées.

Cependant, l'applicabilité de cette représentation pour la menace interne reste nuancée. Actuellement, les TTP sont principalement orientés vers les intrusions et les attaques externes, ce qui limite leur pertinence pour les menaces internes. En effet, la menace interne présente des caractéristiques spécifiques, telles que l'absence de phase d'accès initial et l'importance des facteurs psychologiques et comportementaux. De plus, les motivations et les comportements des insiders peuvent varier considérablement, allant de la simple négligence à des actions délibérément malveillantes, ce qui complique encore la modélisation des menaces selon des TTP précises définie hors du spectre de la menace interne. La majorité des actions d'un scénario de menace interne étant des actions habituellement normales, les considérer comme des TTP classique ne permettrait une représentation précise. Néanmoins, des efforts sont en cours pour adapter les TTP à ce contexte particulier. Par exemple, un projet de version "insider" de MITRE ATT&CK est en développement [34], et un extrait des TTP recensées est présenté en Figure 2.10. Il vise à revoir la taxonomie, au niveau des procédures, en se basant sur les techniques standards. Cette initiative pourrait permettre de mieux capturer les nuances des menaces internes en intégrant des indicateurs comportementaux et des scénarios spécifiques, mais la normalité des étapes d'un scénario peut remettre en question le modèle « Technique – Tactique – Procédure ». Une telle adaptation permettrait de tout de même la classification des différentes actions d'in scénario et la conceptualisation du chemin parcouru

pour atteindre l'objectif visé, offrant ainsi un indicateur de distance à l'objectif, comme présenté, dans un contexte plus global par une étude de l'université de Zurich.[36]. En somme, bien que les TTP ne soient pas encore pleinement adaptés à la menace interne, leur évolution et leur enrichissement continu promettent de fournir un cadre et des ressources précieuses pour modéliser en vue de contrer ces menaces spécifiques.

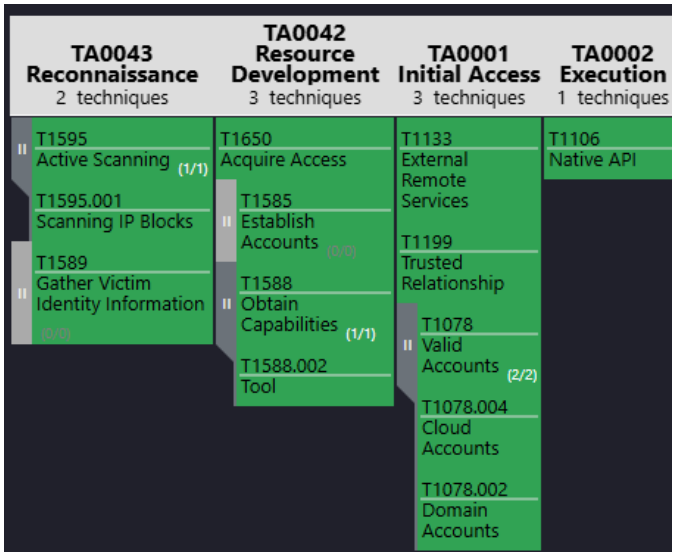

Figure 2.10 : Extrait des TTP impliquées dans des scénarios de menance interne

2.2.2.7 Un défi autour de la granularité d'étude

A quelle échelle faire notre représentation, en effet, manipulant de la donnée en nombre, nous avons une double exigence paradoxal, le plus nous étudions une donnée dans son contexte, la plus précise sera notre jugement à son propos, mais dans le même temps, nous cherchons une solution en temps réel, et donc il nous faut une détection avant qu'un scénario soit complété. Si nous étudions semaine par semaine, un scénario peut subvenir au sein d'une semaine et alors s'il est détecté il sera trop tard. Considérer une granularité journalière nous permet de détecter les scénarios sur plusieurs jours et une activité journalière est une échelle plus naturelle. Cependant pour un scénario court, on a encore le même problème. Cela est une des difficultés majeures à mettre en avant par rapport à l'immense majorité des papiers scientifiques existants, faisant une analyse différée de détection, ayant ainsi beaucoup de données de contexte et de comparaison, mais le tout, après la crise. Il est donc nécessaire d'étudier l'impact de la granularité sur une solution se disant temps réel.

Un groupe de chercheurs de l'université de Dalhousie a mené une telle enquête sur les données du CERT [37] et souligne que l'impact de la granularité peut dépendre de la solution de détection utilisée, mais une granularité fine, comme entre une connexion et une déconnexion, permettent une détection rapide et relativement efficace, tandis que les granularités grossières, comme l'analyse de semaines d'activité, offrent une vue d'ensemble des comportements sur une période plus longue, permettant une détection légèrement meilleure. La même équipe de chercheurs montre ainsi dans une deuxième étude par exemple qu'une granularité plus grosse semble être la mieux adaptée pour la détection d'anomalies. [38] En revanche, les données au niveau des sessions apparaissent comme le meilleur candidat pour la détection de menaces internes à l'aide de l'apprentissage supervisé, car elles permettent un système avec un taux élevé de détection des initiés malveillants et des temps de réponse rapides. Dans le contexte d'un souhait de détection en temps réel, une granularité fine donne une bonne réactivité, mais une granularité plus grosse permet une meilleur performance, allier plusieurs niveaux d'analyse pourrait se montrer intéressant.

2.2.2.8 Représentation plus « haut niveau »

Une analyse efficace des menaces internes nécessite une approche contextuelle, dépassant les événements isolés pour considérer des schémas comportementaux globaux. Les travaux de l'université de Dalhousie [37], [38] montrent qu'une granularité élargie peut permettre une détection améliorée. Cela révèle l'importance de la considération d'indicateurs abstraits : écarts par rapport aux habitudes d'un utilisateur, accès inhabituels liés à son rôle, ou signaux psychologiques (via l'analyse sémantique des échanges ou des profils OCEAN fournis par les RH, comme dans le jeu CERT). Cette représentation intègre alors des données techniques, organisationnelles et comportementales pour modéliser les scénarios par une vision transverse et donc comme une trajectoire cohérente plutôt qu'une suite d'actions. Elle peut ainsi permettre de détecter des scénarios plus complexes, comme une exfiltration progressive corrélée à un profil à risque. L'enjeu est d'exploiter ce contexte riche sans compromettre la réactivité nécessaire au temps réel.

2.3 Méthodes de détection et d'investigation de la menace interne

2.3.1 Outils commerciaux

Comme évoqué précédemment, un bon nombre de solutions existent sur le marché concernant la détection d'intrusion, allant d'outils de gestion d'accès à des solution d'EDR (Endpoint Detection and Response) ou d'IPS/IDS (Intrusion Prevention/Detection System). Ces outils, fruits de la recherche académique et industrielle du domaine font leurs preuves. Certaines de ces solutions peuvent se montrer utiles quant à des cas de menace interne, mais il n'existe que peu ou prou de solutions spécifiques, et donc vraiment efficaces pour la détection de la menace interne. Parmi les rares entreprises orientant leurs travaux sur le sujet se trouvent Exabeam ou Dtex. Les pôles d'activité de ces solutions peuvent être l'activité des individus, ou bien la vie des fichiers. L'analyse du comportement des utilisateurs, et le plus grand défi de la recherche aujourd'hui, et est concrétisé par quelques solution d'UBA (User Behaviour Analysis) ou UEBA (User Extended Behaviour Analysis). Ces solutions de marché représentent la concrétisation, organisationnellement intégrable de la recherche passé, comme en témoigne l'évolution de la solution d'Exabeam, passant d'un outil basé sur des réseau Bayésiens en 2021 [8], et présente aujourd'hui plusieurs niveaux dont l'utilisation de réseau bayésiens mais aussi d'apprentissage machine [39].

Les solutions de type UAM (User Activity Monitoring) sont aussi partie prenante du front contre la menace interne. Un UAM permet de surveiller et d'analyser les activités des utilisateurs sur les systèmes informatiques d'une organisation. Il peut détecter des comportements anormaux ou suspects, comme des accès non autorisés, des modifications inhabituelles de fichiers, ou des tentatives de contournement des politiques de sécurité. Certaines solutions propriétaires indiquent évoluer vers une fusion, ou collaboration entre les UAM et les UEBA qui d'après les producteurs permettrait une analyse plus complète et aboutie des activités utilisateurs, intégrant entre autres des algorithmes de machine learning. Par exemple, des solutions comme ObserveIT (maintenant intégré à Proofpoint), Forcepoint UEBA et DTex sont mises à jour régulièrement en affinant les algorithmes utilisés pour le suivi des comportement utilisateurs.

Enfin, la majeure partie des scénarios de menace interne ayant pour finalité une influence sur la donnée, visant son intégrité ou sa confidentialité, les solutions de Data Loss Prevention (DLP) dimensionnées à l'origine pour lutter contre les menaces externes peuvent se transposer à une protection contre la menace interne, supposant une analyse des résultats de telles solutions. Les

différents types de DLP sont portés par des solutions spécifiques, ou des solutions globales comme Qostodian de Qohash, il s'agit de :

DLP réseau : Ces solutions surveillent le trafic réseau pour détecter et bloquer les tentatives de transfert de données sensibles en dehors de l'organisation. Des outils comme Symantec DLP (maintenant Broadcom) ou McAfee Total Protection for DLP peuvent être déployés sur les passerelles réseau pour analyser le trafic sortant et appliquer des politiques de sécurité.

DLP endpoint : Installées sur les postes de travail et les serveurs, ces solutions surveillent les activités locales pour empêcher la fuite de données via des périphériques USB, des courriels, ou des applications cloud. Digital Guardian et Forcepoint DLP en sont des exemples.

DLP de stockage : Ces outils analysent les données stockées dans les bases de données, les serveurs de fichiers et les solutions de stockage cloud pour identifier et protéger les informations sensibles. Varonis est une solution bien connue qui utilise des techniques d'apprentissage machine pour détecter les comportements anormaux et sécuriser les données stockées.

DLP cloud : ces solutions, comme Netskope, surveillent les activités et les données dans les applications SaaS, PaaS et IaaS.

L'intérêt au niveau organisationnel de solutions telles que celles proposées par DTex sont l'intégration de plusieurs technologies de détection et d'investigation de la menace interne au sein de rapports et d'évaluations de risques en connaissance de l'environnement de la compagnie, rendant l'évaluation des risques liés aux comportement suspicieux compréhensible et intégrable dans une stratégie organisationnelle. Beaucoup de solutions émergent et il peut être difficile de différencier celles-ci, entre les revendications commerciales et leurs réelles performance, Gartner propose un rapport comparant les différentes solutions pour la gestion de la menace interne, résumé en tableau x. Il n'en demeure pas moins que de telles solutions ne comportent pas d'implémentation des dernières avancées de la recherche, rendant de trop nombreuses fausses alertes et souvent après l'action de compromission.

Tableau 2.4 : Solutions commercialisées, d'après Gartner [40]

Catégorie	**Top 3 Solutions / Fournisseurs**
UAM	DTEX InTERCEPT
	Teramind UAM
	Veriato Cerebral
UEBA	Exabeam UserXDR
	Securonix UEBA
	Gurucul Risk Analytics
DLP	Symantec DLP
	Netskope (infonuagique)
	Forcepoint DLP
Solutions Hybrides (IRM + DLP + UEBA/UAM)	Microsoft Purview
	DTEX InTERCEPT
	Cyberhaven DDR

2.3.2 Vers des méthodes d'intelligence artificielle

Les systèmes traditionnels de détection de menaces reposent très souvent sur des règles prédéfinies et des signatures, mode de fonctionnement typique des IDS contemporains, pour identifier les évènements suspects. Ces systèmes présentent ainsi un certain nombre de limites dont l'incapacité à relever des suites d'évènements suspectes, ou leur incapacité à s'adapter aux nouvelles menaces sans lourde mise à jour par un vecteur humain ou encore la génération d'un nombre élevé de faux positifs, nécessitant une intervention humaine fréquente pour l'analyse [41].

Ainsi, et avec l'arrivée en « mode » de l'intelligence artificielle, de plus en plus de solution commerciales se vantent de l'utilisation de l'apprentissage automatique dans leurs produits. Au-delà de la vague de communication commerciale à ce sujet, la communauté scientifique quant à elle démontre les capacités réelle de ces algorithmes, leurs possibles utilisations sans en négliger les limites d'intégrations opérationnelles. Des premières utilisation d'apprentissage machine se basent sur des mots-clés, et des scores d'anomalies, comme l'outils XABA [42], mais leur dépendance à des seuils manuels limite leur performance et illustre le manque d'opérabilité. Cependant, ces méthodes d'apprentissage machine, ou « Machine Learning (ML) » offrent une alors alternative plus flexible et adaptative. Elles permettent de détecter des comportements

anormaux de manière précoce, ne se limitant pas à l'analyse d'évènements uniques, sont extensibles et peuvent fournir une analyse prédictive [43].Pour autant elles nécessitent des ressources computationnelles importantes et peuvent être difficiles à appliquer en temps réel et dans un contexte opérationnel.

2.3.3 Méthodes de Machine Learning pour la détection de la menace interne

Le traitement de la donnée ainsi que l'intelligence artificielle sont depuis la fin du vingtième siècle un sujet important pour la recherche en tant que le monde s'articule autour de la donnée. Ainsi de très nombreuses méthodes de traitement de la donnée en quantité existent. La présente section présente ainsi les différentes méthodes, et modèles proposés par la communauté scientifique. Cette revue synthétique permet la compréhension des modèles applicables ou déjà appliqués à la modélisation et détection de la menace interne.

2.3.3.1 Modèles de Classification et de Régression

Régression Logistique

La régression logistique est une technique statistique qui permet de prédire une variable binaire. Elle repose sur la fonction logistique, ou sigmoïde, qui transforme une combinaison linéaire des caractéristiques en une probabilité. Cette probabilité indique la vraisemblance qu'une observation appartienne à une classe spécifique, par exemple, qu'un comportement soit malveillant. Le modèle est entraîné pour minimiser l'erreur entre les prédictions et les valeurs réelles, souvent en utilisant la méthode de maximisation de la vraisemblance. La régularisation L2, ou "ridge", est souvent appliquée pour pénaliser les grands coefficients, ce qui aide à prévenir le surapprentissage et améliore la généralisation du modèle. Concrètement, sans cette régularisation, une variable comme le "nombre de fichiers copiés" pourrait avoir un coefficient si élevé qu'un simple archivage de routine déclencherait immédiatement une fausse alerte.

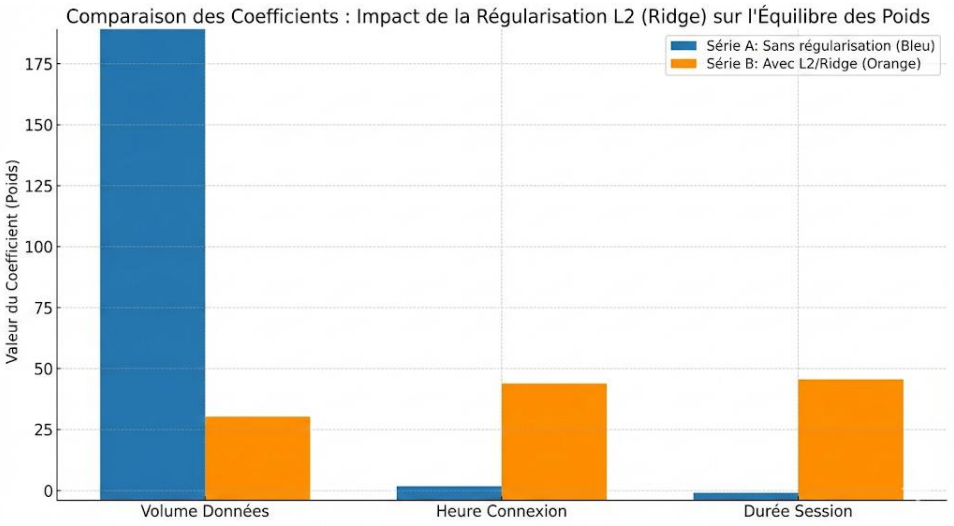

Figure 2.11 : Exemple d'impact d'une régularisation L2

La méthode L2 réduit mathématiquement ce poids (par exemple en passant le coefficient de 10 à 0,5) comme illustré dans l'histogramme en Figure 2.11, obligeant ainsi le modèle à valider la menace en croisant cette donnée avec d'autres indices comme l'horaire ou la destination des fichiers.

La régression logistique relativement interprétable, car les coefficients du modèle indiquent l'importance relative de chaque caractéristique dans la prédiction de la classe et donne de bons résultats de détection, au même titre que d'autres classificateur comme la classification naïve bayésienne, pour certaines modélisations de la menace interne, mais reste souvent moins précis que d'autres algorithmes plus complexes. [44]

Machine à vecteur de support

La machine à vecteur de support, « Support Vector Machine » (SVM) est une méthode d'apprentissage supervisé qui cherche à maximiser la marge entre les classes pour une meilleure séparation des données. Elles sont souvent utilisées pour détecter des intrusions, car elles sont performantes [45]. Leur but est de réduire les erreurs de classification et de créer une séparation claire entre deux groupes de données, même si ces données ne sont pas facilement séparables. Un des défis de la SVM est alors le déséquilibre entre les groupes de données, défi particulièrement présent dans les jeux de données, et les cas réels de menace interne ou l'écrasante majorité de l'activité est normale. Une variante appelée SVM à une classe est alors utilisée, de plus dans le cas particulier de flux de données [6]. Elle est conçue pour repérer les données anormales, c'est-à-dire

celles qui sont très différentes des données normales sur lesquelles le modèle a été entraîné. Cette technique essaie de maximiser la distance entre un point et la majorité des données, et tout ce qui est en dehors de cette distance est rapporté anormal. Contrairement au SVM classique, le SVM à une classe n'est entraîné qu'avec des données normales.

L'efficacité de cette approche est avérée dans des environnements statiques et dont la normalité est facilement certifiable. Néanmoins, la nature dynamique et non linéaire des menaces internes rend difficile l'application d'un seuil fixe de détection. Par exemple, la classification comme suspect peut varier d'un individu à l'autre (deux évènements sur une dizaine de minutes versus plusieurs sur une période étendue). En outre, le paradigme d'apprentissage à une seule classe assume l'absence d'anomalies dans l'ensemble des données d'entraînement, ce qui est peu imaginable avec les données présentées en 2.2.1 et dans une perspective de déploiement au sein d'une compagnie.

K-plus proches Voisins

La définition même de la problématique de menace interne, c'est-à-dire identifier un comportement anormal d'un individu, pour lequel tout semble normal et où d'autres individus agissent de la même manière est propice à l'utilisation du concept des k plus proches voisins (KNN). C'est un algorithme simple qui classe un point de données en fonction de la majorité des classes parmi ses k voisins les plus proches. Il est utilisé pour la classification des comportements anormaux dans différents travaux sur le jeux de données TWOS [46] après un traitement des données à l'aide d'autres méthodes, et sur les données du CERT de nombreuses fois, résultant de bonnes performances mais surpassées par les forêts aléatoires ou AdaBoost [47] .

2.3.3.2 Méthodes à base d'arbres

.La forêt aléatoire, « Random Forest » (RF) est une technique qui utilise plusieurs arbres de décision, « Decision Tree » (DT), chacun formé à partir d'un échantillon aléatoire des données d'entraînement et des caractéristiques, grâce à des méthodes comme le bagging ou le boosting et les sous-espaces aléatoires [48]. La prédiction finale est déterminée par la majorité des votes des arbres. Chaque arbre crée un modèle de classification sous forme d'arborescence, où chaque feuille représente une règle de décision pour une classe spécifique. La classification d'un point de données se fait en descendant l'arbre de la racine jusqu'à une feuille, en évaluant une caractéristique à chaque nœud pour décider de la direction à prendre. L'objectif est de maximiser le gain d'information à chaque division, améliorant ainsi la précision du modèle.

C’est un algorithme souvent cité pour les performances élevées qu’il fournit pour la détection d’anomalies, ce dépendamment des modèles sur lequel il agit, mais sa précision est balancée par un manque d’adaptabilité ou le temps de calcul non approprié pour du temps réel [49], [50]

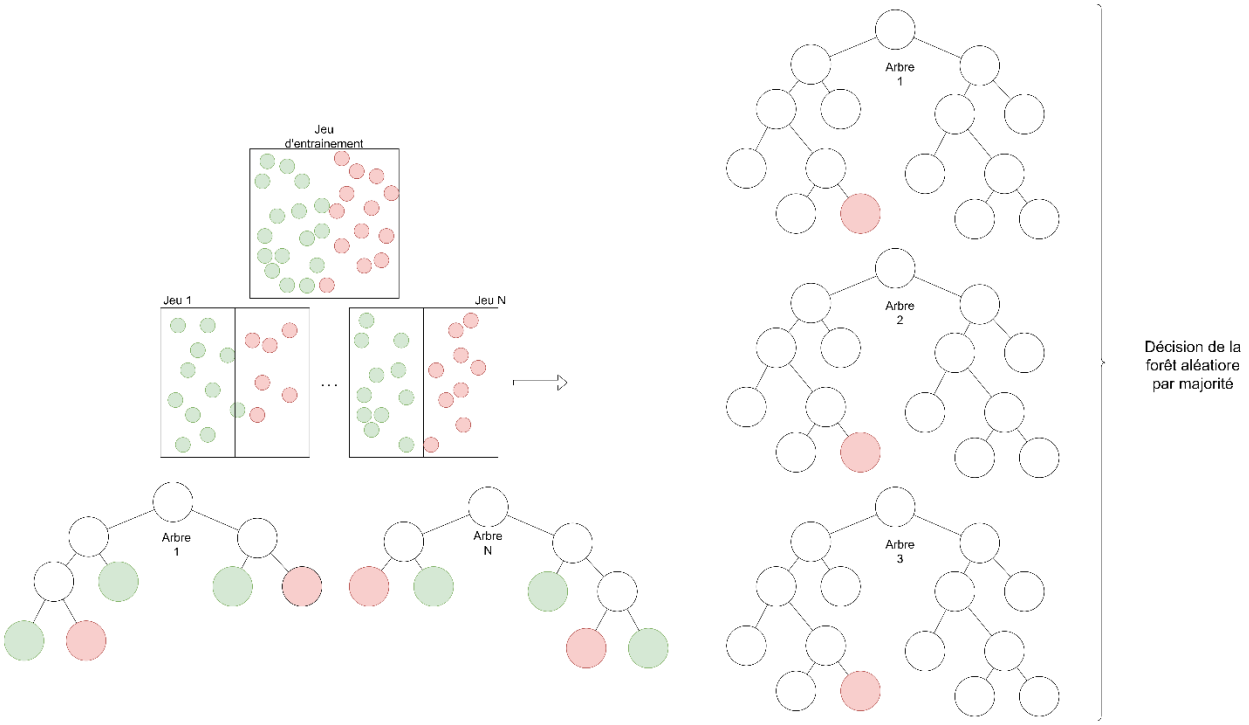

Figure 2.12 : Fonctionnement d'une forêt aléatoire

Les « Isolation Forests » (IF) sont une autre technique basée sur les arbres, mais conçue spécifiquement pour la détection d'anomalies. Contrairement aux forêts aléatoires, les forêts d'isolement fonctionnent en isolant les observations en les séparant des autres points de données. L'idée est que les anomalies sont plus susceptibles d'être isolées plus rapidement, nécessitant moins de divisions pour être séparées. Ainsi, les points qui nécessitent moins de divisions pour être isolés sont considérés comme des anomalies. Cette approche est particulièrement efficace pour identifier les points aberrants dans les ensembles de données de grande taille, ce qui est le cas des anomalies de menace interne au milieux d’évènements normaux. Cette méthode de forêt a montré de bon résultats de détection concernant les données du CERT, prévoyant les utilisateurs sur le point de quitter la compagnie à partir des emails [51] partant du principe qu’ils sont plus enclins à compromettre le SI, ou bien en analysant l’activité web [52] avec la limite des requêtes http du jeu du CERT.

2.3.3.3 Méthodes d'ensemble

Le Gradient Boosting construit un modèle prédictif en combinant plusieurs modèles plus simples, généralement des arbres de décision. Cette méthode commence par un modèle initial, souvent une prédiction constante, et ajoute de manière itérative de nouveaux arbres de décision pour corriger les erreurs des modèles précédents. Chaque nouvel arbre est ajusté pour prédire le gradient de la fonction de perte par rapport aux prédictions actuelles, permettant ainsi de minimiser les erreurs de manière progressive. Le modèle est mis à jour à chaque itération en ajoutant le nouvel arbre, pondéré par un facteur d'apprentissage, aux modèles précédents. Ce processus est répété jusqu'à ce qu'un nombre prédéfini d'arbres soit atteint ou qu'une condition de convergence soit satisfaite.

Une implémentation du Gradient Boosting est la méthode XGBoost, et montre de très bons résultats pour l'identification ad-hoc des anomalies dans les évènements du CERT [47]. Ces résultats se montrent améliorés par l'intégration de l'analyse des traits psychologiques comme l'a fait N'Famoussa Kounon [53]. Des résultats presque parfaits sont obtenus par un algorithme de pseudo temps-réel [54], perfection à considérer avec attention au vue de la préparation des données sur mesure pour l'étude en question.

L'AdaBoost, ou Adaptive Boosting, est une méthode d'ensemble comparable au Gradient Boosting basé sur des méthodes plus simples. Cette technique commence par attribuer des poids égaux à toutes les instances de l'ensemble de données. À chaque itération, un nouveau modèle faible est entraîné sur les données pondérées, avec une attention particulière portée aux instances mal classées par les modèles précédents, grâce à une augmentation de leur poids. Les poids des instances sont ensuite mis à jour en fonction des erreurs du modèle actuel, réduisant ainsi le poids des instances correctement classées et augmentant celui des instances mal classées. Les modèles faibles sont alors combinés en un modèle fort, chaque modèle étant pondéré par sa précision. Ce processus est répété jusqu'à ce qu'un nombre prédéfini de modèles soit atteint ou qu'une condition de convergence soit satisfaite. Cette méthode montre de très bons résultats, légèrement supérieurs à XGBoost d'après [47].

2.3.3.4 Modèles probabilistes

Chaînes de Markov cachées

Les chaînes de Markov modélisent les systèmes qui évoluent entre différents états avec des probabilités de transition définies, permettant de prédire les états futurs basés sur les états précédents. Les chaînes de Markov cachées, quant à elles, supposent que les états ne sont pas directement observables, mais peuvent être inférés à partir de données observables. Ainsi, une étude utilise les chaînes de Markov cachées pour modéliser les comportements des utilisateurs du jeu de données du CERT [55]. Pour ce faire, les données sont structurées en séquences d'actions réalisées par les utilisateurs sur une semaine, formant ainsi des séries temporelles. Les activités ont été classées selon divers critères, notamment la connexion pendant les heures de travail en semaine, de 8h à 17h, ainsi que la connexion en dehors de ces heures et durant les week-ends. Les chercheurs ont également pris en compte les autres caractéristiques du jeu de donnée.

Les chaînes de Markov cachées ont permis d'analyser ces séquences d'activités sur une période d'apprentissage de quatre semaines, puis de prédire la probabilité des séquences pour les semaines suivantes. Le modèle a démontré un bon taux de détection tout en maintenant un faible taux de faux positifs et permet de retracer facilement les transitions d'état lors de la détection d'une intrusion interne, ce qui participe non seulement à la détection mais aussi à l'investigation. Cependant, cette méthode suppose comme pour la SVM à une classe que l'activité observée pendant la période d'apprentissage est normale, ce qui n'est pas toujours vérifiable.

Réseaux Bayésiens et BGMM

Les réseaux bayésiens, ou réseaux de croyance, sont des modèles graphiques probabilistes utilisés pour représenter et raisonner sur des situations incertaines en combinant la théorie des probabilités et la théorie des graphes. Ils se composent d'un graphe acyclique dirigé (DAG) où les nœuds représentent des variables aléatoires et les arêtes dirigées indiquent les dépendances conditionnelles entre ces variables, ainsi que des tables de probabilités conditionnelles (CPT) qui quantifient ces dépendances. Ces réseaux permettent de mettre à jour les croyances sur les variables à la lumière de nouvelles preuves grâce au théorème de Bayes. Ils permettent alors d'émettre une décision sur le caractère anormal d'évènements.

Cette méthode est employée comme bloc final de décision dans une solution de détection corrélant des évènement [56], mais aussi, analysant les donnée psychologique du modèle OCEAN, comme support pour la prédiction des comportements [57].

Une autre approche bayésienne, les Bayesian Gaussian Mixture Models (BGMM), s'avère particulièrement utile pour la modélisation de données continues et la détection de clusters. Contrairement aux réseaux bayésiens, les BGMM ne reposent pas sur une structure graphique mais supposent que les données proviennent d'un mélange de plusieurs distributions gaussiennes. Cette méthode permet d'identifier des groupes au sein des données sans nécessiter de connaissances préalables sur les relations entre variables. Les BGMM utilisent l'inférence bayésienne pour estimer les paramètres des distributions sous-jacentes, offrant ainsi une compréhension probabiliste des données et une capacité à détecter des anomalies ou des comportements atypiques. C'est une approche qui peut offrir de bons résultats dans la détection de comportement anormaux, et génère peu de faux positifs (~6% sur CERT r4.2) et surpasse les Gaussian Mixture Models [58].

Cependant ces modèles probabilistes sont négligés par certains scientifiques, les probabilités d'action ne tiennent pas nécessairement compte du contexte, faussant la probabilité finale et donc la décision [59].

2.3.3.5 Réseaux de neurones

Réseaux de neurones artificiels

Les réseaux de neurones artificiels (ANN) sont constitués de neurones organisés en couches : une couche d'entrée, des couches cachées et une couche de sortie. Chaque neurone applique une pondération à ses entrées, passe le résultat dans une fonction d'activation, puis transmet la sortie aux neurones suivants. L'apprentissage se fait par ajustement des poids, via l'algorithme de rétropropagation, pour minimiser une fonction de perte entre les sorties du réseau et les valeurs cibles.

Grâce à leur capacité à modéliser des relations non linéaires complexes se montrent très performant dans la détection d'anomalies sur le jeu de donnée du CERT [38]. Cependant, leur manque d'interprétabilité rend leur utilisation plus que limité dans le cadre de l'investigation.

Réseaux de neurones récurrents

Les réseaux de neurones récurrents (RNN) sont une classe spécialisée de réseaux de neurones artificiels conçus pour traiter des séquences de données. Contrairement aux réseaux de neurones traditionnels, les RNN possèdent des boucles qui leur permettent de maintenir une mémoire des informations passées, ce qui les rend particulièrement adaptés aux tâches où l'ordre des données est crucial, ce qui est le cas des séquences d'évènements analysées dans le cadre de la MI.

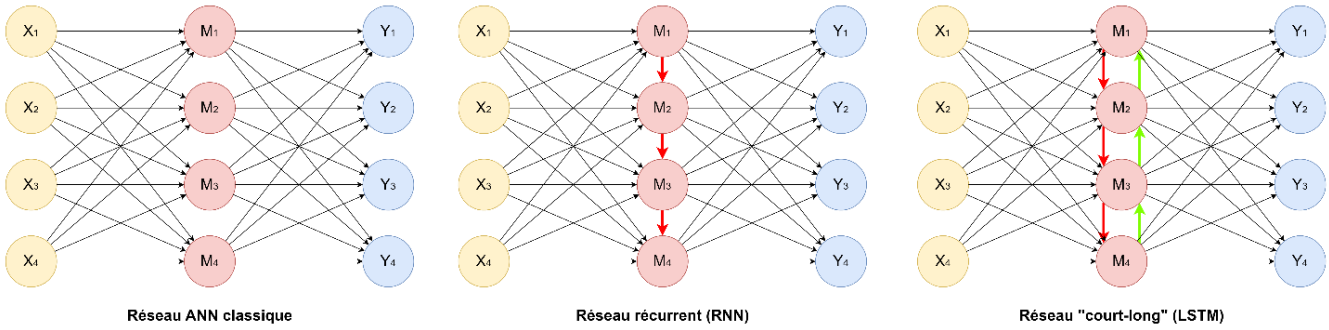

Figure 2.13 : Comparaison des réseaux de neurones principaux

Leur architecture leur permet de capturer des dépendances temporelles et de conserver un contexte à travers les séquences. Cela peut cependant poser des défis liés à la rétention d'informations sur de longues distances et donc présenter une forte limitation liées aux ressources computationnelles lorsque l'on évoque de la détection en temps réel. C'est une problématique en partie adressée par des variantes comme les réseaux LSTM (Long Short-Term Memory) et GRU (Gated Recurrent Units). Les LSTM, dont le fonctionnement est comparé aux ANN et RNN en Figure 2.13, sont amplement sujets des travaux sur la menace interne et sont une des méthodes présentant les meilleures métriques [60], [25] et leur intégration pour le traitement de flux de données est étudié et détaillé en 2.4.1.

Réseaux de neurones convolutifs

Les réseaux de neurones convolutifs (CNN) se distinguent des autres architectures de réseaux de neurones par leur capacité à traiter efficacement les données structurées en grille, comme les images. Contrairement aux réseaux de neurones artificiels (ANN) ou récurrents (RNN), les CNN utilisent des filtres convolutifs, des matrices de petite taille qui se déplacent sur les entrées pour détecter des motifs locaux pertinents, facilitant ainsi la reconnaissance de formes ou de comportements.

Lorsqu'on transpose ces principes au domaine des graphes, où les données ne suivent pas une structure régulière, on fait appel aux réseaux de neurones convolutifs sur graphes, ou Graph Convolutional Networks (GCN). Les GCN représentent une extension naturelle des CNN aux données non-euclidiennes, telles que les réseaux sociaux, les graphes de connaissances ou encore un modèle de scénarios de MI. Contrairement aux CNN classiques, qui ne peuvent capturer que des informations individuelles sur les entités, les GCN intègrent à la fois les propriétés des nœuds (features) et les relations entre eux (structure du graphe), en exploitant la topologie du réseau sous-jacent.

Jiang et al. ont proposé un modèle innovant de détection d'anomalies basé sur les GCN, capable de détecter non seulement des comportements déviants individuels, mais aussi des groupes d'utilisateurs malveillants potentiellement organisés. Leur approche repose sur la construction d'un graphe où les nœuds représentent les utilisateurs et les arêtes encodent la similarité comportementale et les interactions. En combinant les caractéristiques comportementales extraites des données du CERT, avec une matrice d'adjacence pondérée reflétant les similitudes entre utilisateurs, leur modèle surpasse significativement des méthodes plus traditionnelles (SVM, forêts aléatoires, CNN) en précision et en rappel [61]. Cependant le caractère supervisé de leur étude ne permet pas immédiatement sa considération en temps réel.

2.3.3.6 Traitement du Langage Naturel (NLP)

Le Traitement du Langage Naturel (NLP) utilise des techniques d'apprentissage automatique et de linguistique pour permettre aux machines de comprendre et de générer du langage humain. Les modèles NLP capturent les nuances syntaxiques et sémantiques du texte. Les applications commerciales incluent la traduction automatique, l'analyse de sentiments et la reconnaissance d'entités nommées. Les modèles pré-entraînés comme BERT, RoBERTa, GPT ou Mistral améliorent les performances grâce à des représentations contextuelles riches.

Les méthodes de NLP peuvent elles aussi représenter un outil clé dans la détection ou l'investigation de la menace interne. Contrairement aux méthodes d'apprentissage machine précédentes, les NLP ne sont pas forcément employé directement pour l'évaluation d'évènements au regard des autres et d'un contexte plus général. Ils peuvent cependant être utiles pour la tokenisation, la complétion prédictive de scénarios, ou la génération de score d'anomalies avant l'intervention d'un modèle de classification comme un LSTM [25]. En particulier, ils sont

performant pour le traitement de certains types d'évènements comme l'analyse des contenus de courriels envoyés ou de fichier, afin de les classifier comme ayant un potentiel anormal ou non. Leur emploi est donc intéressant en complément de l'extraction de caractéristiques, et peut se présenter comme un élément clé dans la recherche sur la détection de la menace interne [62].

2.3.3.7 Techniques d'ingénierie des caractéristiques

Traditionnellement, des caractéristiques telles que la connexion hors des heures normales de travail, la consultation d'URL non autorisées, ou d'autres indicateurs spécifiques sont extraite des évènements et utilisées comme indicateurs pour identifier des comportements suspects. Bien que ces caractéristiques soient cruciales, elles ne peuvent pas être exhaustives ni adaptatives par définition. Par conséquent, pour l'analyse des comportements en vue de la détection de menaces, cette approche pourrait ne pas être optimale. C'est pourquoi plusieurs études proposent des méthodes d'extraction de caractéristiques adaptables et automatiques. Cependant, une fois un événement identifié, il peut être intéressant pour son investigation de conserver une part de caractérisation fixe.

Dans ce contexte, plusieurs techniques d'ingénierie des caractéristiques se distinguent par leur capacité à automatiser et à adapter l'extraction de caractéristiques. Parmi elles, les autoencodeurs, l'Analyse en Composantes Principales (PCA) et la projection aléatoire (Random Projection) sont particulièrement prometteuses[38].

Autoencodeurs

Les autoencodeurs jouent un rôle crucial dans de nombreux travaux de la communauté scientifique. Ils sont souvent employés comme module de feature engineering, extrayant des logs la matière nécessaire à la détection par des méthodes d'apprentissage machine. En transformant les données brutes en représentations compactes et informatives, ils permettent de capturer les caractéristiques essentielles des logs tout en réduisant la dimensionnalité. L'encodeur extrait les caractéristiques pertinentes, tandis que le décodeur tente de reconstruire les données originales, minimisant ainsi les pertes d'information. Ils permettent de se libérer de l'encodage manuel des caractéristiques et donc de sa non-exhaustivité inhérente. Cependant, le manque d'explicite dans cette solution peut rendre plus complexe la compréhension du processus et, par conséquent, rendre la solution moins adaptée et efficace.

Analyse en composantes principales

L'Analyse en Composantes Principales est une procédure statistique qui utilise une transformation orthogonale pour décomposer un ensemble de données multivariées en un ensemble de composantes principales, c'est-à-dire des vecteurs propres. Alors que les caractéristiques des données d'entrée peuvent être corrélées, les composantes PCA sont linéairement non corrélées. Ces composantes présentent la quantité maximale de variance dans les données originales, où chaque composante principale a la variance la plus élevée possible, étant donné qu'elle est orthogonale aux composantes précédentes. La première composante rend compte de la plus grande quantité de variabilité dans les données. Le processus d'application de la PCA pour la réduction de dimensionnalité est le suivant : d'abord, calculer la matrice de covariance des données, puis déterminer les valeurs propres et les vecteurs propres correspondants de cette matrice. En fonction de la dimensionnalité réduite choisie, les données sont projetées en utilisant les vecteurs propres avec les plus grandes valeurs propres. Les caractéristiques dans les données transformées sont des combinaisons linéaires des caractéristiques d'entrée.

Projection aléatoires

La projection aléatoire est une technique qui permet de projeter des données d'origine de dimension d dans un espace de dimension k (où k est beaucoup plus petit que d) en utilisant une matrice aléatoire R. Chaque colonne de cette matrice a une longueur unitaire. La projection des données d'entrée X dans l'espace de dimension k est obtenue en multipliant X par R. La projection aléatoire repose sur le lemme de Johnson-Lindenstrauss, qui affirme que si les points dans un espace vectoriel sont de dimension suffisamment élevée, ils peuvent être projetés dans un espace de dimension inférieure tout en préservant approximativement les distances entre les points. Cette méthode est simple et computationnellement efficace pour réduire la dimensionnalité des données. De plus, elle préserve bien les similarités entre les vecteurs de données, même à faible dimensionnalité [63].

2.3.3.8 Approches Hybrides et Multi-Agents

Les différentes méthodes présentées jusqu'à présent, bien qu'efficaces dans des contextes spécifiques, montrent des performances variables selon les missions et les configurations. L'hétérogénéité de la problématique de la menace interne nécessite en effet une diversité

d'analyses, de représentations des scénarios et d'événements. Cette complexité a poussé la communauté scientifique à explorer des approches hybrides, combinant les forces de chaque méthode pour une détection plus robuste. Parmi ces approches, les systèmes multi-agents (MAS) se distinguent comme une piste prometteuse pour répondre à ces défis. Les MAS reposent sur des entités autonomes et spécialisées, capables d'interagir de manière collaborative pour analyser des données hétérogènes et par exemple reconstituer des trajectoires d'utilisateurs ou des scénarios d'attaque. Leur architecture distribuée permet de réaliser des analyses dynamiques et contextuelles, de données complexes et hétérogènes.

Parallèlement, le domaine informatique voit émerger des outils d'intelligence artificielle, tels que les agents conversationnels ou les systèmes spécialisés comme présenté en 2.3.3.6, capables d'interagir avec d'autres outils ou entre eux. Ces agents IA, déployés dans des environnements cadrés et dotés de connaissances spécifiques, permettent d'automatiser des tâches complexes autrefois réservées à l'interaction homme-machine. Leur collaboration, ou interaction machine-machine, ouvre la voie à l'automatisation de processus d'analyse ou de production entiers. Les architectures multi-agents, intégrant des algorithmes variés, allant de l'apprentissage par renforcement profond aux modèles génératifs en passant par les grands modèles de langage, font désormais l'objet d'études approfondies, notamment dans des domaines comme la modélisation des menaces ou l'IIoT (Industrial IoT). Par exemple, des travaux récents ont démontré comment des MAS optimisés par des modèles génératifs (comme les CTGAN [64]) peuvent réduire les biais liés aux jeux de données déséquilibrés, améliorant ainsi la précision de la détection d'anomalies. D'autres recherches ont exploré la collaboration entre agents spécialisés pour l'analyse des logs, où des agents « auditeurs » et « coordinateurs » utilisent des LLM pour interpréter des événements complexes et générer des alertes en temps réel [65], [66].

Cependant, malgré ces avancées, aucune recherche n'a encore exploré l'application d'une solution hybride multi-agents pour le suivi de la menace interne, l'analyse d'événements et la reconstitution de scénarios en temps réel. Les approches existantes se limitent souvent à des aspects ponctuels, comme la détection d'intrusions [64] ou la simulation d'attaques, sans offrir une vision intégrée combinant analyse comportementale, raisonnement sémantique et opérabilité organisationnelle. C'est précisément ce besoin d'une approche hybride, capable de fusionner ces dimensions pour une détection proactive et une réponse adaptative, qui peuvent permettre une investigation à l'image de ce que produirait une équipe d'analystes spécialisés.

2.4 Temps réel

2.4.1 Méthodes de traitement de données

Parmi principales technologies de manipulation et traitement de données par flux, ou en temps-réel, les travaux de la Fondation Apache [67] semble particulièrement se distinguer. Technologies phares dans ce domaine, Apache Spark se distingue par sa capacité à traiter de vastes volumes de données en mémoire avec son module Spark Streaming, tandis qu'Apache Kafka excelle dans la gestion des flux de messages à haute vélocité, ce qui en fait une excellente solution de manipulation d'événements informatiques [68]. Les bases de données SQL traditionnelles ont également évolué vers des solutions temps réel avec des extensions comme PostgreSQL avec sa fonctionnalité de réplication logique ou MySQL avec ses capacités de Change Data Capture (CDC). D'autres frameworks comme Apache Flink [69] proposent une gestion native du temps d'événement, particulièrement adaptée aux analyses complexes sur fenêtres temporelles. Ces technologies s'appuient sur des architectures distribuées permettant le passage à l'échelle et la tolérance aux pannes, deux des huit caractéristiques essentielles pour garantir le traitement en temps réel [70].

Dans un contexte opérationnel, ces infrastructures permettraient de traiter dynamiquement des flux d'événements (comme les logs SOC) en ajustant la granularité d'analyse aux caractéristiques des données, plutôt qu'à des intervalles prédéfinis. Cependant, pour un travail de recherche, leur déploiement complet peut s'avérer superflu en tant qu'il est documenté et validé. L'essentiel réside dans la validation des méthodes d'apprentissage machine et de leur adaptation à un traitement en flux continu, qui pourrait être supporté par de telles infrastructures. L'enjeu n'est alors pas l'architecture distribuée, mais la capacité à modéliser et analyser un flux de données continu par le biais de telles méthodes. Une méthode proposée dans la littérature et l'analyse à travers des fenêtres temporelles glissantes.

La manipulation de fenêtres glissantes dans les architectures d'apprentissage machine pour le traitement en temps réel de flux séquentiels se montre une méthodologie intéressante. Ces approches transcendent les limites des méthodes batch traditionnelles en permettant un découpage dynamique des séries temporelles, où la taille et le chevauchement des fenêtres peuvent s'adapter aux caractéristiques intrinsèques des données ou à des heuristiques fixes. Zhan et Kim [71] illustrent comment une segmentation adaptative, couplée à des mécanismes de feedback instantané, permet de capturer les changements de régime dans les données financières, tout en évitant les biais

introduits par les agrégations temporelles statiques. De leur côté, Yhdego et al. [72] poussent cette logique en proposant des stratégies de fenêtrage hiérarchique, alignant les fenêtres sur les motifs critiques des signaux. Ces travaux soulignent que le fenêtrage glissant ne doit plus être considéré comme un simple prétraitement, mais comme un composant actif et adaptatif du modèle, capable de s'ajuster dynamiquement pour concilier réactivité et robustesse dans des environnements critiques.

2.4.2 Travaux sur la détection de la MI en temps réel

Bien que très peu nombreux, quelques travaux de recherche portent en leur centre une notion en détection en temps réel, et non plus post-hoc. Ils présentent des méthodes différentes, tant dans les solutions d'apprentissage machine employés que dans l'architecture globale de leurs approches, mais révèlent une certaine ambiguïté relative à la notion de temps réel.

En 2017 déjà, Böse et al proposent un pipeline complet de détection de la menace interne basée sur de l'apprentissage machine en temps-réel. Leur approche innovante dénommée RADISH est de séparer les parties détection et apprentissage, se débarrassant des contraintes computationnelle impliquant un temp long [73]. Leur approche, bien que novatrice dans sa séparation des processus, se heurte à deux écueils majeurs. D'une part, elle adopte une définition procédurale du temps réel, réduite à une contrainte de latence entre la mise à jour des modèles et la génération d'alertes, sans intégrer la dimension caractérielle d'une interception instantanée des actions malveillantes. D'autre part, les performances opérationnelles restent médiocres, avec un rappel de seulement 0,5 et une précision de 0,08 sur le jeu de données du CERT, révélant des performance bien inférieurs aux travaux de la littérature évalués post-hoc. Leur système, basé sur une agrégation temporelle des événements en sessions utilisateurs, introduit aussi un délai intrinsèque incompatible avec une détection véritablement temps réel, où chaque action doit être évaluée indépendamment de son contexte sessionnel pour intercepter des trajectoires d'attaques in statu nascendi.

Les travaux plus récents, tels que LAN [74], tentent de pallier ces limites en combinant une structure d'analyse par graphes et une prédiction séquentielle via des GNN, atteignant des taux de détection prometteurs (0,88). Cependant, leur évaluation du "temps réel" se limite à une borne temporelle computationnelle, mesurant le temps de traitement par activité sans garantir que la détection intervienne avant l'action critique, telle qu'une exfiltration de données ou une élévation de privilèges. De même, Qawasmeh et al. [54], bien que les résultats semblent spectaculaires en

précision avec des techniques de streaming appliquées à des logs synthétiques, voient leur évaluation biaisée par un jeu de données sur-mesure inspiré de celui du CERT, où les poids d'anomalies et les scénarios sont prédéfinis, masquant ainsi des problèmes potentiels de généralisation. Leur définition du temps réel se résume à une analyse en continu de flux, sans mécanisme de feedback instantané ni adaptation dynamique aux changements de comportement, ce qui constitue une lacune critique face à des attaquants adaptatifs capables de modifier progressivement leurs habitudes pour échapper à la détection.

2.5 Défis et perspectives

L'exploration de la littérature scientifique sur la détection de la menace interne met en lumière plusieurs défis persistants qui freinent l'adoption de solutions réellement efficaces et opérationnelles. Ces limites sont d'autant plus notables que, malgré des progrès significatifs dans certaines dimensions techniques, les solutions restent largement inadaptées aux exigences concrètes du terrain. Trois défis majeurs structurent cette problématique et orientent les contributions de ce mémoire.

Défi de performance : concilier précision et pertinence des alertes

Un premier défi fondamental concerne la performance des systèmes de détection. Si de nombreuses approches basées sur l'apprentissage automatique, en particulier non supervisé, parviennent à détecter des anomalies comportementales avec des taux de détection encourageants, elles restent souvent pénalisées par un taux de faux positifs élevé – pouvant dépasser 30 à 40 % dans certains cas. Ce déséquilibre, bien documenté, limite l'utilisabilité de ces systèmes en contexte réel, où une surcharge d'alertes entraîne rapidement une perte de confiance des analystes et une érosion de la réactivité opérationnelle. L'enjeu n'est donc pas uniquement d'améliorer la précision brute des algorithmes, mais de concevoir des mécanismes capables de hiérarchiser, contextualiser et corréler les alertes de manière à en maximiser la pertinence et à en réduire la redondance. Cette problématique de performance est au cœur des orientations de ce mémoire, qui vise une architecture combinant détection fine et mécanismes de réduction des alertes non critiques.

Défi de temporalité : dépasser la stricte notion de « temps réel »

Le second défi est d'ordre conceptuel : il réside dans une interprétation trop restrictive de la notion de « temps réel ». La majorité des travaux considèrent uniquement les contraintes temporelles

physiques – à savoir la vitesse de traitement ou de réponse – sans s'interroger sur la temporalité intrinsèque aux scénarios de menace interne. Or, la détection efficace d'une menace interne ne dépend pas tant de la latence d'un système que de sa capacité à identifier une action suspecte avant qu'elle ne devienne critique ou irréversible. Cela suppose une compréhension dynamique de la progression d'un scénario – depuis ses prémices jusqu'à une phase d'exécution décisive – et la capacité de détecter des indicateurs faibles dans les étapes amont. Cette approche qualitative du temps réel, centrée sur la notion d'action décisive, constitue un axe méthodologique différenciateur dans les travaux de ce mémoire, qui cherche à détecter non pas une attaque en cours, mais une menace en train d'émerger.

Défi d'intégration opérationnelle : absence d'approches orientées investigation

Enfin, un troisième défi – et non des moindres – concerne l'absence quasi-généralisée d'un lien fonctionnel entre détection et investigation. De nombreux travaux identifient des événements potentiellement compromettants, mais ceux-ci sont rarement replacés dans une logique de scénario global ou intégrés dans une démarche d'analyse décisionnelle. Les solutions proposées se concentrent sur des instants disjoints, là où les analystes de SOC ont besoin de reconstruire des chaînes d'événements, de comprendre la cohérence ou l'intention d'un comportement, et d'établir des relations de causalité entre actions suspectes. Ce manque de support à l'investigation rend difficile l'exploitation effective des alertes générées, surtout dans des contextes à forte contrainte de temps et de ressources. La capacité à structurer les événements en scénarios interprétables, à travers des représentations explicites (graphes, chaînes, ontologies) et à guider les analystes dans leur prise de décision, est donc un enjeu majeur. C'est précisément cette lacune que ce mémoire s'attache à combler en proposant une architecture orientée « détection pour l'investigation ».

CHAPITRE 3 UNE SOLUTION HYBRIDE ET DYNAMIQUE

La menace interne bénéficie alors d'une attention croissante de la part de la communauté scientifique, motivée par la part croissante de son implication dans les cas de menaces ou d'attaques au sein des compagnies. Pour autant, les performances encourageantes, si non positives d'un grand nombre d'étude ne se transfèrent pas dans le monde commerciale et les organisations ne disposent pas d'outils intégrant ces avancées scientifique, si tant est qu'elles s'arment face à la menace interne. Il est ainsi sans outrecuidance possible de supposer que des éléments clés de ce décalage entre la recherche et son application industrielle sont le manque d'adaptabilité et d'opérabilité des solutions proposées dans les différents travaux de recherches.

En effet, bien que certains travaux proposent des solutions fournissant des très hauts taux de détections sur les données utilisées, les algorithmes et architectures sous-jacents sont très souvent dimensionnés pour ces données particulières et actent dans un environnement propice au succès. De plus, beaucoup de ces travaux sont statiques, et ne s'adaptent donc pas à une organisation vivante, ou bien tentent d'intégrer un certain dynamisme mais au péril de la performance ou d'une détection réelle de la menace au moment opportun.

3.1 Diviser pour mieux régner

Ce constat, contrasté entre des résultats positifs et encourageants d'une part, et un manque de réalisme et de lien avec une application concrète d'autre part invite à corréler les différentes avénacées de la communauté sous la forme d'une solution hybride, adaptable et opérable.

L'objectif d'un département de cybersécurité au sein d'une entreprise est double au regard de la menace interne. La finalité étant de prévenir toute menace, il est nécessaire pour lui de détecter le plus tôt possible un cas de menace interne, ou l'implication dans un tel scénario. Ce premier point supposé résolu, il est important pour les analyste de comprendre ces implications et de suivre le contexte global des éventuelles alertes afin de prendre part aux contres mesures de protection nécessaire. Ainsi émergent deux aspects liés mais distinct, aux contraintes propres que sont la détection des évènements potentiellement impliqués dans un cas de menace interne, et l'investigation, ou le suivi de chaque individu en lien avec le SI par rapport à leur propension à être impliqué dans un tel cas.

3.2 Des contraintes temporelles et d'acuité propres

La nécessité de séparer l'étude de l'objectif de détection de celle de l'objectif d'investigation est hormis la différence entre les solutions qui leurs sont applicables motivée par la différence entre les contraintes que l'on peut émettre vis-à-vis de ces deux pans d'une solution de profilage de la menace interne.

Concernant l'aspect temps réel, véritable fil conducteur de cette recherche, la nuance apportée en 2.1.6 nous rappelle que cette exigence, bien que primordiale, ne doit pas occulter les autres impératifs techniques. Pour rappel, la notion de temps réel évoque ici la connaissance avant une étape définie comme « décisive » d'un scénario de menace interne. De manière plus concrète, cela implique alors de bénéficier d'une analyse continue de l'activité des individus, mais aussi une compréhension réfléchie de ce que sont ces activités. C'est pourquoi une détection, même hypothétique doit être réalisée couramment, alertant les analystes sur la probable implication d'un évènement x ou y dans une action de menace interne, quitte à avoir une quantité de faux positifs non négligeables. Ainsi si l'évènement reporté porte un risque évident et explicite, une action immédiate de remédiation peut être entreprise. Considérant une réelle réactivité à ce niveau, une analyse plus approfondie de connaissance des actions et tendances de chaque individu peut se détacher de cette immédiateté, exploitant tout de même, tel un analyste d'un renforcement continue de sa connaissance de chaque individu.

Alors, si la contrainte temporelle est moindre mais existante concernant l'investigation, la performance d'un tel bloc doit être maximale pour fournir un suivi précis du potentiel de compromission de chaque individu en fonctions de ses habitudes, son environnement et des alertes qui peuvent être détectées par son homologue.

3.3 Une architecture adaptable et pragmatique

Cette segmentation architecturale offre l'avantage de conjuguer l'instantanéité requise pour la détection avec la finesse d'analyse nécessaire à l'investigation, tirant ainsi parti du meilleur de ces deux paradigmes. Le module de détection, confronté au flux brut en « temps réel », agit tel un filtre intelligent : il a pour vocation de réduire le débit de données, sans altération de l'information pertinente, avant leur entrée dans le module d'investigation, optimisant de fait la performance temporelle de ce dernier.

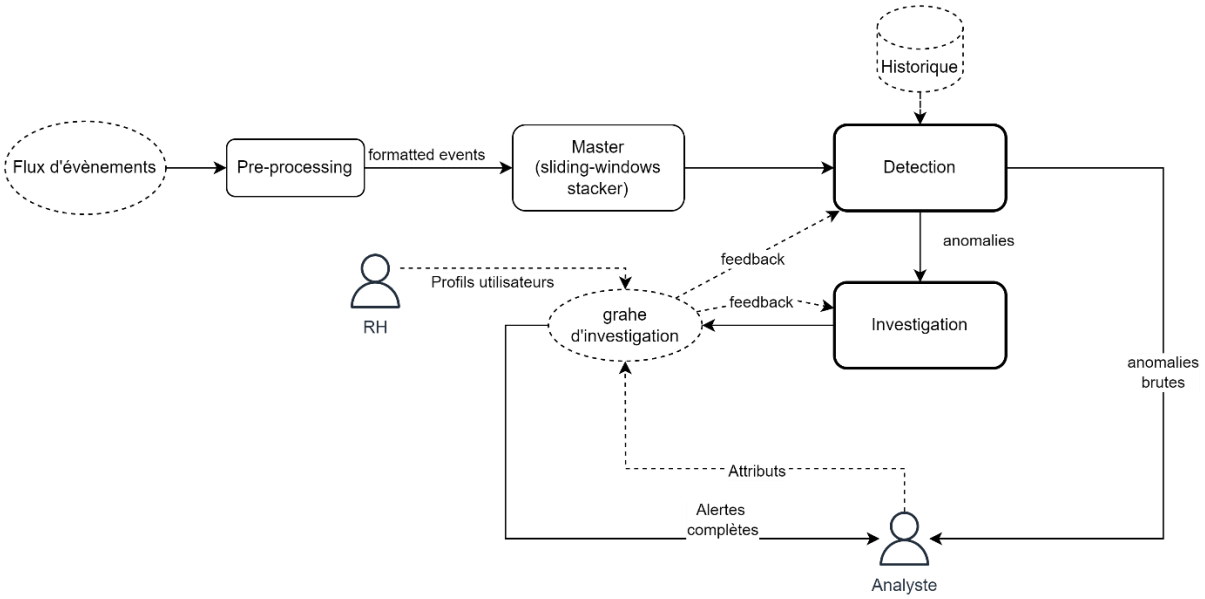

Figure 3.1 : Architecture globale

Comme l'illustre la Figure 3.1, l'architecture globale s'articule donc autour d'une chaîne de traitement séquentielle : les données brutes sont d'abord normalisées par un prétraitement, puis filtrées par le module de détection (CHAPITRE 4) qui ne transmet que les anomalies significatives au module d'investigation (CHAPITRE 5) pour une analyse contextuelle approfondie.

Ce mémoire a en ligne de mire l'intégration opérationnelle, dans une organisation. C'est pourquoi la solution dispose d'interactions utilisateur, intégrables en une interface homme-machine, afin qu'un analyste puisse monitorer les anomalies remontées par le module de détection, d'explorer l'investigation produite par le module éponyme. Il est aussi possible de mettre à jour les caractéristiques, issus d'ontologies, formant la base de connaissance du module d'investigation, ou encore de modifier les attributs de la représentation graphique. Des données formant les profils utilisateurs peuvent aussi être mises à jour par l'interaction d'un utilisateur, comme un membre des ressources humaines. Enfin, les modules de détection et d'investigation peuvent recevoir des feedbacks permettant d'adapter leur fonctionnement à des profils identifiés comme évoluant (en mettant par exemple l'emphase sur l'analyse des évènements attribués à un certain utilisateur plutôt qu'à d'autres.

3.4 Architecture applicative

Afin d'évaluer la proposition dans sa globalité, c'est-à-dire dans un contexte proche d'une intégration à une organisation réelle, la solution pensée comme l'ensemble des deux modules (détection et investigation) présentées aux chapitres 4 et 5 nécessite de raccorder ces deux modules pour traiter un flux de logs normalisé tel qu'il pourrait être analysé en SOC. Ainsi l'architecture théorique présentée en Figure 3.1 est déclinée par l'architecture applicative en Figure 3.2.

Le flux formatté est orienté par un script « *master.py »* qui interface aussi la dorsale et l'IHM qui est une page *http*. Les données sont rassemblées en fenêtres par le script « *aggregateur.py* », puis ces fenêtres son envoyées dans le module de détection, « *detection.py* ». Ensuite les anomalies vont dans le module d'investigation composé de son orchestrateur « *master_invest.py* », et la déclaration des agents est contenue dans le script « *analyse.py* ». Enfin, les scripts « *graphe.py* » et « *graphe_classes.py* » ne servent qu'à la déclaration des classes du graphe et des opérations sur celui-ci. Pour finir, le script « *visualiser_graphe.py* » est utile au rendu visuel et le fichier « *config.yaml* » porte les différents paramètres, utiles à l'ensemble des scripts.

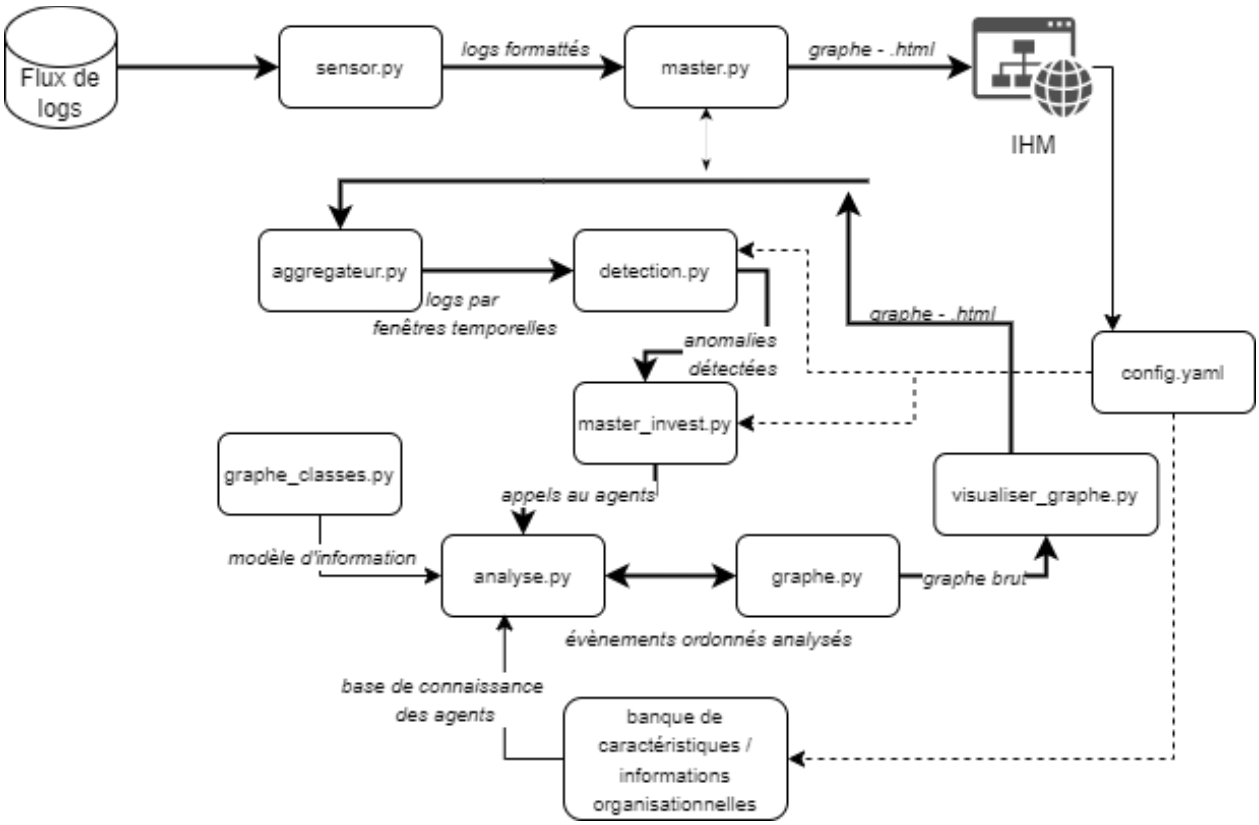

Figure 3.2 : Architecture applicative

Si le contexte de recherche pousse à l'exploitation des données du CERT de Carnegie et Mellon pour réaliser son évaluation, le flux d'évènement est formatté à son entrée pour envisager l'analyse d'évènements génériques et non spécifiquement ceux du CERT.

Dans une organisation réelle, les flux de données pourraient être exposés sur le réseau et orchestrés avec une solution du type Apache Spark, présentée dans la partie 2.4.1, et le capteur des données d'entrées, dans la solution présentée représenté par le script *« sensor.py »* pourrait être étendu pour prendre en charge plusieurs sources de logs, de manière spécifique ou générale avec les logs Linux, Syslog, ou évènements Windows.

CHAPITRE 4 DÉTECTION EN TEMPS-RÉEL DE LA MENACE INTERNE PAR MACHINE LEARNING

Le premier module à la base de notre solution, que j'ai participé à élaborer, se concentre sur la détection en temps réel des anomalies indicatrices d'une menace interne. Ce chapitre présente les travaux de recherche et les résultats qui ont fait l'objet d'une publication lors de la conférence Network and System Security 2025 à Wuhan, et dont l'article intégral, intitulé « Real Time Anomaly Detection For Event-Based Insider Threat Hunting », est disponible en [75].

La problématique centrale abordée ici est la difficulté de détecter des menaces internes au fil de l'eau, dans des flux continus d'événements, contrairement aux approches traditionnelles qui analysent souvent les données a posteriori.

Pour répondre à cet impératif de réactivité, nous avons développé une méthodologie basée sur un mécanisme de fenêtres glissantes (sliding windows). Cette approche permet de segmenter le flux d'activités (connexions, accès fichiers, courriels, etc.) en séquences temporelles chevauchantes, offrant ainsi un contexte dynamique pour l'analyse comportementale sans attendre la fin d'une session ou d'une journée de travail. Cela permet aussi l'utilisation de contextes suffisamment grands pour employer des algorithmes d'apprentissage machine tout en ayant une analyse continue.

La méthodologie proposée a été évaluée sur le jeu de données de référence CERT r4.2, en comparant trois architectures d'apprentissage automatique distinctes pour le calcul des scores d'anomalie :

Les Réseaux de Neurones sur Graphes (GNN) : Cette approche modélise les interactions entre utilisateurs et ressources sous forme de graphe, permettant de capturer les relations structurelles et les changements de topologie suspects.

Le Traitement du Langage Naturel (NLP) : En utilisant des modèles de langage pré-entraînés (comme BERT ou RoBERTa), cette méthode analyse la sémantique des actions et des contenus (par exemple, le corps des courriels ou les chemins d'accès aux fichiers) pour identifier des déviations contextuelles.

Les Réseaux de Neurones Récurrents (LSTM Autoencoders) : Cette technique classique est utilisée pour apprendre la séquence "normale" des événements d'un utilisateur et signaler toute suite d'actions s'écartant du schéma habituel (erreur de reconstruction).

Les résultats expérimentaux démontrent que l'approche par fenêtres glissantes est viable pour une détection en temps réel. Bien que chaque algorithme présente des forces spécifiques (les GNN excellant dans la détection structurelle et les LSTM dans la modélisation séquentielle) l'architecture globale permet de lever des alertes pertinentes avant que les scénarios de menace n'atteignent leur phase critique. Notamment, l'évaluation face aux scénarios de menace avérés du jeu de données CERT a montré que 100 % des événements décisifs (ceux marquant un point de non-retour dans l'attaque, comme l'exfiltration de données) ont été détectés, souvent précédés par l'identification de plusieurs étapes préparatoires.

Ce module de détection constitue ainsi le premier filtre de notre solution hybride. En identifiant les séquences suspectes en temps réel avec une latence maîtrisée (inférieure à la durée de la fenêtre glissante), il alimente le module d'investigation présenté au chapitre suivant, permettant de passer d'une simple alerte technique à une compréhension contextuelle de la menace.

CHAPITRE 5 INVESTIGATION CONTINUE PAR GRAPHE DE COMPROMISSION

Dans le cadre réel d'une organisation, la détection d'éléments anormaux ne suffit pas, car il est nécessaire d'établir la liste des actions à effectuer à compter d'une telle détection, qu'il s'agisse de contremesure, ou bien d'une surveillance renforcée. De plus, il est fortement possible qu'un évènement détecté soit, au regard du contexte complexe de la menace interne, non compromettant, ou part d'un scénario de compromission.

L'état de l'art montre que la menace interne, de part sa complexité, nécessite plusieurs niveaux d'analyse pour être comprise. Ces analyses, logiques ou comportementales par exemple, n'exploitent pas les mêmes données et ont besoin de l'expertise humaine pour être correctement effectuées. Ce constat, croisé avec le perfectionnement de systèmes multi-agentiques présentés en 2.3.3.8 souligne l'intérêt que peut présenter l'utilisation d'agent pour comprendre la menace interne.

Un agent est un LLM spécialisé par un entrainement sur des données spécifiques ou évoluant avec des ressources choisies. Ainsi plusieurs agents, sur-mesure, chacun ayant une tâche d'analyse spécifique avec un contexte et une connaissance spécifique, peuvent travailler ensemble pour permettre une investigation renforcée sur la base d'un modèle de chaine d'attaque adaptée à la menace interne a été mise en place.

Il s'agit d'une solution se présentant comme un outil de conseil en vue d'épauler un analyste en SOC et proposant ainsi une interface graphique de suivi des comportements des utilisateurs, et de l'évolution de potentiels scénarios de menace interne.

5.1 La collaboration multi-agents pour l'investigation de la menace interne

Les systèmes multi-agents (Multi-Agent Systems, MAS) sont apparus comme un paradigme prometteur pour répondre aux complexités des tâches où une méthode classique d'IA serait insuffisante. Les architectures MAS répartissent des tâches spécialisées entre des agents autonomes, permettant une détection et une réponse collaboratives. Dans le domaine de la sécurité, des recherches ont exploré les systèmes multi-agents pour la détection d'intrusions, comme les travaux de Mouyart et al. [64], qui combinent l'apprentissage par renforcement et des modèles génératifs pour améliorer la détection des menaces externes en équilibrant les jeux de données et

en optimisant les performances du système. Cependant, ces recherches n'ont pas abordé la menace interne, et encore moins son investigation. Pourtant, le principe même d'une solution multi-agents multiplie la puissance d'analyse et semble intéressant pour une tâche de si haut niveau.

L'appel à des agents est majoritairement en rapport avec l'IA mais peut aussi concerner des nœuds décisionnels plus sommaires, comme ayant un raisonnement logique. Cependant des travaux récents montrent leur puissance placent l'IA et les LLM comme des outils intéressants pour l'investigation [66], [76], [77]. Les LLM ont démontré des capacités remarquables en compréhension du langage naturel, en analyse de logs et en classification comportementale, ce qui en fait des outils précieux pour les applications en cybersécurité. Des études récentes ont exploré l'utilisation des LLM pour des tâches telles que la synthèse d'événements de sécurité, la génération d'hypothèses sur des activités suspectes et la classification de comportements malveillants sur la base d'analyses de contenu [25]. En ce qui concerne le développement d'agents, l'utilisation des LLM repose principalement sur l'ingénierie de prompts, avec une forte recommandation de la communauté pour exploiter les LLM au mieux de leurs capacités [78].

Malgré ces avancées, l'intégration des LLM dans l'investigation des menaces internes en est encore à ses débuts. Une étude de Song et al. offre un premier aperçu d'une solution basée sur des agents pour les menaces internes [65], mais elle se concentre principalement sur la détection *post-hoc*, et la répartition des agents n'exploite pas la spécialisation des connaissances de chaque agent. Leur structure repose sur un agent qui organise la recherche de certains chemins d'attaque connus dans les logs, en orchestrant deux agents qui définissent des points de vérification et les exécutent. Ferraro et al. [66] adoptent une approche similaire. Cependant les agents spécialisés sur des périodes temporelles, et orchestrés par un superviseur. Ce dernier agent décide, sur la base des rapports des autres agents, ce qui est malveillant. Comme la solution de Song et al., ils se concentrent sur la détection et n'agissent pas pour une investigation plus approfondie, alors que des agents spécialisés et connaissant le contexte le permettraient.

Qu'il s'agisse des travaux de Song et al.[65] ou de Feraro et al.[66], la plupart des recherches se concentrent sur l'analyse statique de logs historiques plutôt que sur des pipelines d'investigation en temps réel et pilotés par des agents. Des défis tels que la latence élevée des inférences, le risque d'hallucinations dans des contextes critiques pour la sécurité et le besoin d'intégration avec des connaissances structurées (par exemple, des ontologies, des profils utilisateurs) ont freiné le

déploiement des LLM dans des environnements opérationnels, soulignant la nécessité d'approches hybrides pour utiliser les LLM en temps réel, mais avec l'aide de méthodes plus rapides.

5.2 Méthodologie

L'objectif de cette investigation est dans un contexte de temps réel, avec un flux données d'entreprise en entrée l'analyse d'évènements au regard de leur contexte et de leur type. Dans le cadre plus restreint de ce mémoire, les évènements analysés sont issus du jeu de données du CERT. Une préparation des données, uniformément formattées, et un rejeu du flux d'évènements est donc nécessaire.

C'est l'objectif des trois premiers blocs : « dataset », « pre-processing » et « flow simulator » présentés dans la Figure 5.1. Lorsque le module d'investigation fonctionne à la suite du module de détection, tel qu'est son objectif décrit en 3.3, la préparation des données n'est réalisée qu'en amont de la détection et les évènements pointés comme anormaux prennent place en entrée du bloc « path construction ».

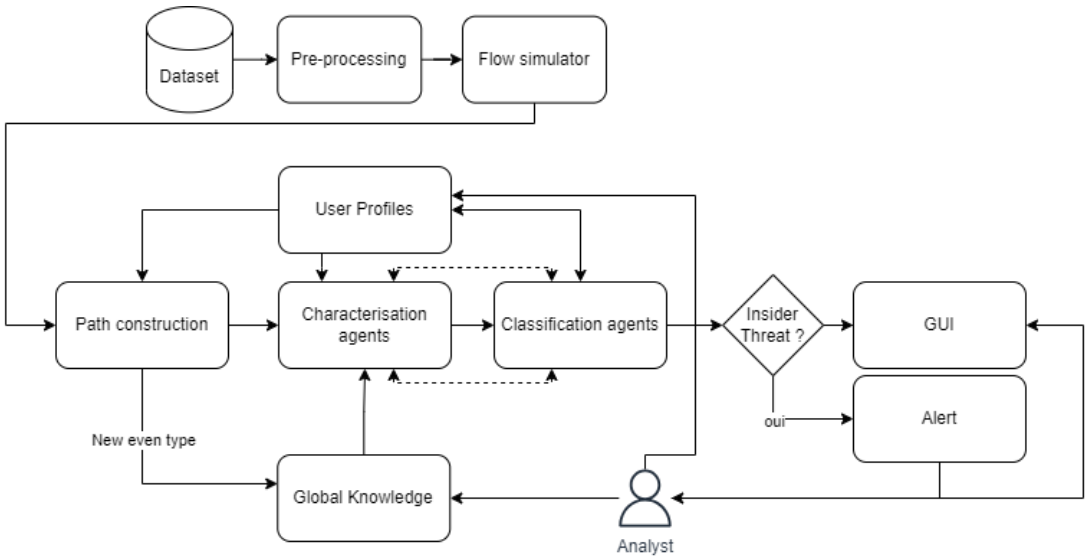

Figure 5.1 : Architecture globale de la solution d'investigation

Ce bloc « path construction » attribut chaque évènement son type (http, logon, …) puis le place au sein d'une session. Cette étape est détaillée dans la partie 5.3.

En fonction du type de chaque évènement, un agent de caractérisation va alors être appelé, spécifique à celui-ci, rendant un mot caractérisant l'évènement au regard des nœuds parents, de

leur caractérisation et du profil de l'utilisateur. Ce mot provient de la base de connaissance propre à l'agent en fonctionnement sont issue des ontologies du domaine et de la connaissance autour des types d'évènements. Cette base de données, propre à chaque agent peut être augmentée par un analyste.

Le nœud de l'évènement est alors enrichi de cette caractéristique, puis analysé par un second agent, dont l'objectif est de classer l'évènement selon une chaine d'attaque spécifique à la menace interne présentée en 2.2.2.1 d'après son mot caractéristique, ses nœuds parents et le profil de l'utilisateur. Ainsi le nœud de l'évènement bénéficie de cette place dans la chaine d'attaque que l'analyste peut visualiser ou pour lequel il peut être alerté.

Chaque utilisateur est alors racine d'un arbre dont les branches sont les différentes sessions, dont un exemple est illustré en Figure 5.2, et les nœuds les évènements temporellement ordonnés de chaque session ainsi que leur caractéristiques et leur place dans la chaine d'attaque.

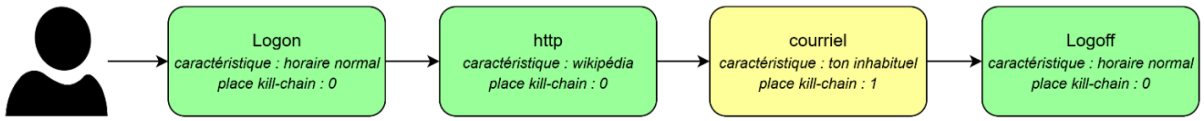

Figure 5.2 : Exemple de session analysée

Enfin, la sortie de la solution est une interface homme-machine présentant le graphe d'activité par utilisateur contenant le résultat de l'investigation. La Figure 5.3 montre un exemple de différents scénarios dans lesquels plusieurs utilisateurs sont impliqués.

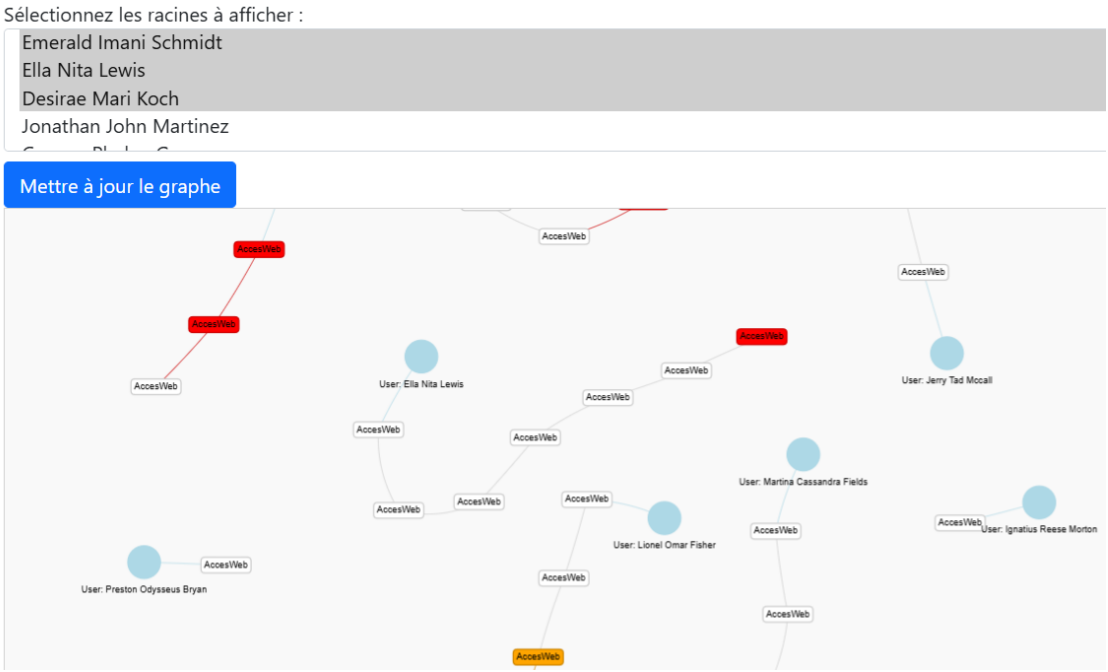

Figure 5.3 : IHM avec un exemple de scénarios analysés

5.3 Architecture spécifique

L'architecture présentée se traduit par les étapes de traitement schématisée dans la Figure 5.4. Comme présenté en CERT 2.2.1.6, le jeu de donnée du CERT un évènement est une connexion (heure, machine), l'utilisation d'un périphérique USB, une consultation web (url, contenu) ou bien l'envoi d'un courriel (contenu, destinataire, copies, pièce jointe, …).

Lorsqu'un évènement est en entrée, la première étape est donc la reconnaissance de son type afin de permettre la sélection du bon agent en charge de son analyse. Ensuite l'évènement est placé dans le graphe, c'est-à-dire que son nœud est créé, et relié à ses parents et à son utilisateur. C'est ce nœud qui portera les métadonnées et les résultats de l'investigation. Par exemple si l'utilisateur A ouvre sa session à 10h, l'évènement va être identifié comme une connexion puis une session va être ouverte, c'est-à-dire un nœud « logon » va être créé, fils du nœud « utilisateur A » et l'évènement est fourni aux agents. Ensuite d'autres évènements d'autre utilisateurs sont traités, et à 10h02, l'utilisateur A effectue sa seconde action, une connexion http. Alors le type « consultation http » est reconnu et un nœud http est créé fils du nœud « logon » de 10h, et l'évènement est fourni aux agents.

Une fois le scénario reconstruit, l'évènement est donné en analyse à un des 5 agents de caractérisation, celui spécialiste de son type : « logon », « device », « web », « file » ou « e-mail ». Cet agent peut être logique (5.3.1) ou sémantique(5.3.2, basé sur un LLM). Son rôle est de rendre un mot caractérisant l'évènement, en prenant en compte les évènements qui le précédent dans le scénario en cours de construction, mais aussi le profil de l'utilisateur.

Un dernier agent (5.3.3, agent « Kill-Chain » ou de « classification ») prend en charge l'évènement, et son mot caractéristique et le place dans la chaine d'attaque, de la même manière en analysant les évènements préalables, le profil utilisateur et le mot fourni par l'agent précédent.

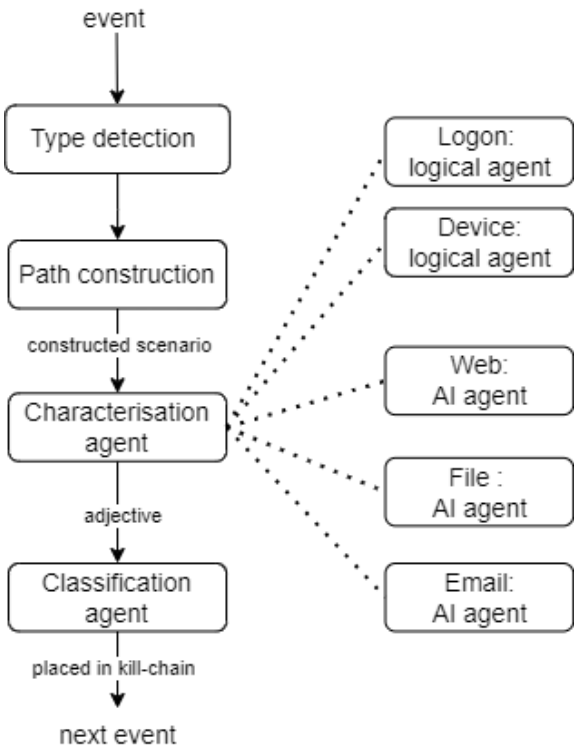

Figure 5.4 : Chemin suivi par un évènement

Les événements qui ne contiennent pas de contenu sémantique nécessitant une interprétation de haut niveau ne nécessitent pas nécessairement l'utilisation d'un modèle linguistique pour être caractérisés. En effet, lorsqu'il s'agit des périodes de connexion ou de la fréquence d'utilisation d'un périphérique USB, un raisonnement numérique logique peut suffire à caractériser l'événement en comparant, par exemple, une plage horaire issue de ses moyennes et ses données historiques dans les informations utilisateur.

La structure organisationnelle permet également d'aller au-delà de la simple comparaison de l'heure de connexion avec l'historique de l'utilisateur (moyenne, écart type, ...) en le comparant également avec les historiques des groupes de travail ou des rôles. L'ensemble de données CERT contient des informations sur les rôles des utilisateurs, ainsi que sur leur unité de production et leurs unités fonctionnelles.

Comparer des éléments avec l'historique d'activité de l'utilisateur peut néanmoins s'évérer dangereux. Un utilisateur malveillant peut préparer son acte malicieux plusieurs mois en avance et adapter petit à petit son comportement afin que le jour de la compromission paraisse normal. La comparaison de l'utilisateur avec ses différents groupes peut remédier à cela dans une certaine mesure.

Une politique de mise à jour de la norme de comparaison adaptée à l'indicateur est également nécessaire, permettant de mettre à jour l'historique uniquement manuellement par les employés des ressources humaines à la suite d'un entretien pour les profils psychologiques, par exemple.

5.3.1 Agents logiques

Cet agent spécifique est chargé de classer un événement de connexion ou d'utilisation d'un périphérique (ainsi qu'un événement de déconnexion) comme normal ou non. Ces deux types d'évènements ne contiennent pas de contenu sémantique. Il est donc plus aisé de définir une logique d'analyse, alors que les évènement qui en contiennent peuvent bénéficier de la capacité d'analyse des LLM.

Pour ce faire, la Figure 5.5 explique que l'agent analyse l'heure de la connexion et la compare au comportement connu de l'utilisateur, mais aussi à celui de ses groupes (rôle, BU, FU et toute équipe hors de l'organisation du jeu de données CERT). Si ce créneau n'est pas couramment utilisé par l'utilisateur ou ses équipes, il est signalé. L'agent utilise cette comparaison pour analyser la connexion. Il peut être utile d'ajouter une étude de la fréquence de connexion ou de la machine concernée. Un utilisateur qui passe son temps à se connecter et à se déconnecter est tout aussi suspect qu'un utilisateur qui se connecte au poste de travail d'un collègue. Ces indicateurs sont couramment surveillés dans les solutions de sécurité disponibles dans le commerce aujourd'hui, mais ne suffisent pas.

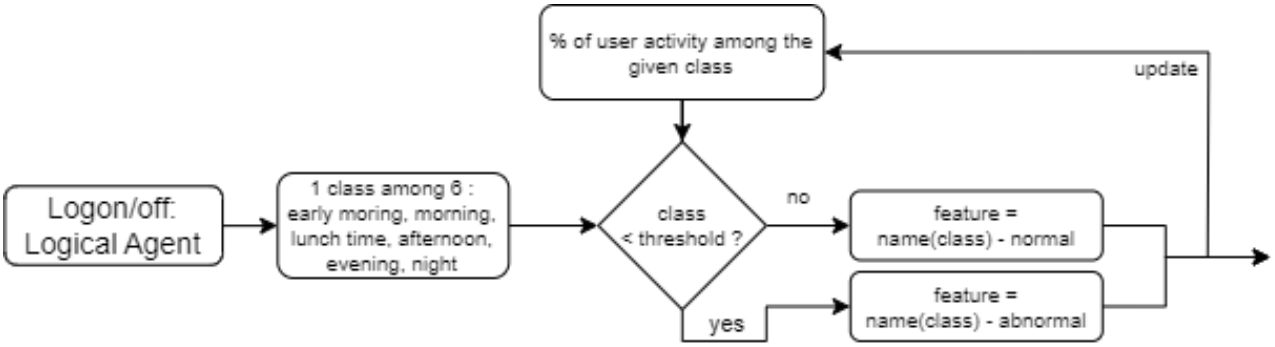

Figure 5.5 : Agent en charge des connexions

L'agent chargé des événements liés aux appareils fonctionne à peu près de la même manière que celui chargé des connexions/déconnexions. Il suit la fréquence de connexion des appareils et la compare aux habitudes de l'utilisateur et de ses équipes, avec un coefficient de tolérance comme dans la Figure 5.6.

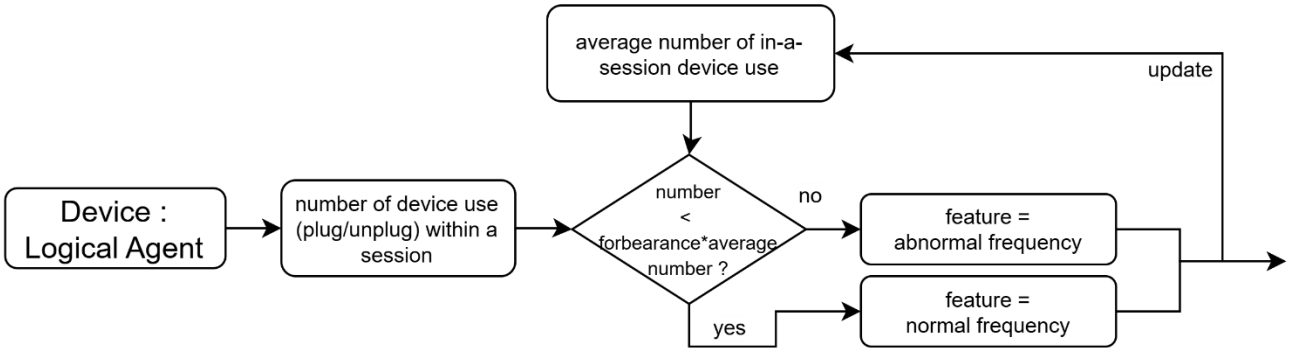

Figure 5.6 : Agent en charge des périphériques

5.3.2 Agents sémantiques

Les agents sémantiques sont tous basés sur le même modèle, schématisé dans la Figure 5.7 pour l'agent web, mais adapté à chaque type d'événement. L'objectif étant de caractériser l'événement en un seul mot, en tenant compte du contexte, l'entrée de l'agent est constituée de toutes les informations relatives à l'événement, et la sortie est la caractéristique choisie. Chaque agent dispose des connaissances des événements précédents d'une session donnée pour évaluer l'événement dans son contexte. Ensuite, chaque agent doit fournir une caractéristique (un mot) provenant d'une banque de mots, issue des connaissances générales en matière de cybersécurité et d'ontologies. Par

exemple la caractérisation d'un évènement web se fait selon les adjectifs suivants, modifiable dans les paramètres de la solution pour une mise à jour manuelle de la base de connaissance de l'agent : *« "normal", "non-working", "attack-tool", "malware", "data-exfiltration", "credential-harvest", "job-search", "career-transition", "cloud-storage-use", "file-sharing-service", "anonymous-browsing", "vpn-use", "dark-web-access", "leak-checking", "documentation-crawl", "burnout-indicator", "disgruntlement-signals", "personal-browsing", "financial-distress", "contract-review", "peer-comparison"»*

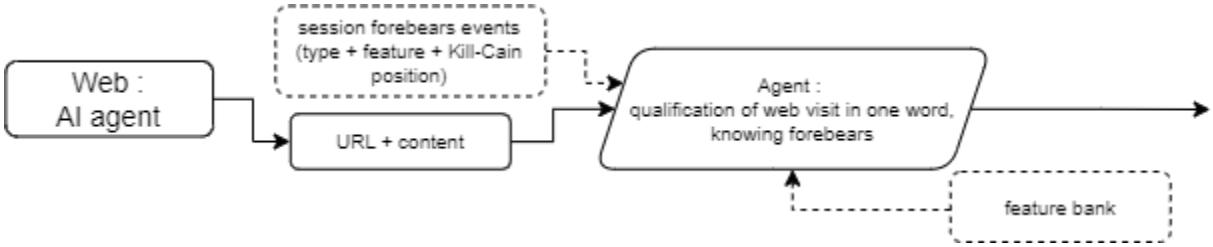

Figure 5.7 : Agent en charge de l'activité web

Chaque agent est conçu spécifiquement en fonction du type d'événement et des informations disponibles.

Par exemple, une solution d'ingénierie rapide pour l'agent web est la suivante, où $\{carac\}$ est la banque de caractéristiques :

Instruction:

You are a cybersecurity analyst in a SOC (Security Operations Center).

You must analyze internal web access and classify it.

Examples:

- URL: https://wikileaks.org/data is data-exfiltration

- URL: https://fr.wikipedia.org/wiki/France ;

Content: "encyclopedia article" is normal

Now analyze this access:

- URL: {url}

- Content: {content}

Respond with exactly one word from the following list: {carac}.

Response:

L'agent analysant les accès à des fichiers utilise la même technique d'invite que l'agent Web et l'agent de messagerie électronique. Les caractéristiques d'entrée spécifiques sont le nom du fichier traité et son contenu brut. La banque de caractéristiques de sortie est adaptée au type de fichier, à des fins normales ou compromettantes.

Enfin, pour analyser les courriels échangés, un journal peut contenir des informations, mais peut également être enrichi par des métadonnées. Le jeu de données CERT part de cette hypothèse et contient les caractéristiques suivantes : *{expéditeur, destinataire, cc, cci, corps, longueur, fichier joint}*. Cet agent utilise ensuite ces caractéristiques comme entrées et suit la même architecture que les autres pour lui attribuer une caractéristique relative au contexte.

5.3.3 Agent de classification

L'agent Kill-Chain est légèrement différent des autres. En fait, il agit comme un deuxième analyste, avec une vision et une expertise plus transversales. Comme le montre la Figure 5.8, il prend en compte l'ensemble du scénario ainsi que les spécifications de l'utilisateur et ses réponses précédentes pour déterminer à quelle catégorie Kill-Chain appartient l'événement donné.

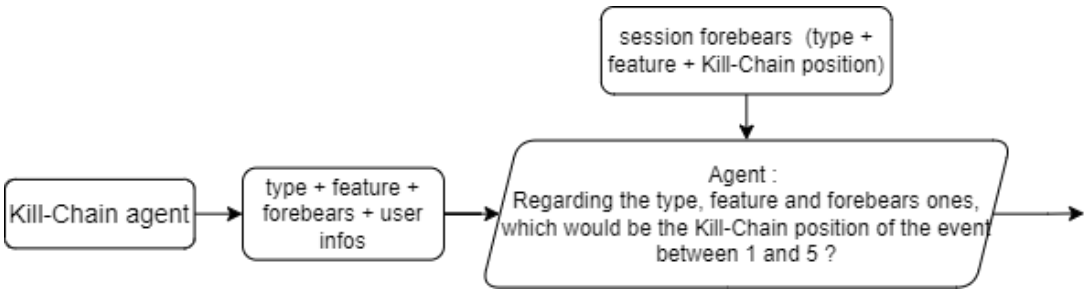

Figure 5.8 : Agent de placement dans la chaine d'attaque

5.4 Implémentation et évaluation

Deux hypothèses sont nécessaires. Premièrement, l'ensemble de données utilisé classe chaque événement comme faisant partie ou non d'un scénario de menace interne. La solution proposée vise à reconstruire le comportement des utilisateurs, à l'analyser et à le classer selon un modèle spécifique de chaîne d'attaque. Nous proposons donc un niveau d'analyse plus élevé que celui effectué par les personnes qui ont constitué l'ensemble de données. De plus, cette analyse

comportementale est destinée à aider les analystes SOC, qu'ils soient humains ou non. Il n'y a donc pas de classification binaire en tant que menace interne ou non. Il s'ensuit qu'une analyse quantitative des résultats ne peut être effectuée en termes de taux de détection, de vrais positifs et de faux positifs. Nous proposons donc une approche qualitative de l'analyse en termes de précision de la classification. Cette analyse qualitative est complétée par une analyse quantitative des performances dans le temps et en termes de ressources afin d'évaluer la capacité de déploiement de la solution.

Deuxièmement, pour évaluer une solution en temps réel, l'idéal serait d'analyser les données sur une période prolongée, comme sur la totalité de la période disponible dans l'ensemble de données CERT. Cependant, afin de rendre cette évaluation moins consommatrice en ressources, nous avons sélectionné des extraits représentatifs de l'ensemble de données. Plusieurs périodes de durées différentes ont donc été testées comme entrée pour notre solution.

En raison de contraintes pratiques liées au temps d'exécution, l'ensemble complet des données n'a pas été traité à l'aide de ce pipeline, mais seulement plusieurs extraits d'une semaine d'activité. Cependant, l'évaluation d'une mise en œuvre à grande échelle reste une piste prometteuse pour de futurs travaux.

Enfin, le choix des modèles des agents est crucial. C’est en effet la performance du modèle et sa capacité à saisir l’évènement informatique ou comportemental dans son contexte qui dictera la performance de notre investigation. Cependant, les modèles avec une connaissance de base évoluent rapidement, et chacun d’eux bénéficient de spécialisation plus ou moins importantes apportées par la communauté. Ce mémoire propose la comparaison entre des modèles pré-entrainés dans des contextes informatiques et des plus généraux afin de comprendre leur incidence. Le choix des modèles a été pensé comme un paramètre du module pour permettre à celui-ci une évolution permanente et de bénéficier des modèles les plus adéquats. Il est donc de mise que les modèles dont ce mémoire fait usage pour la démonstration des performances de l’investigation et de la solution complète ont été fixés et pourraient être surclassés dans le futur.

5.4.1 Effet des différents LLM

Dans le cadre de notre solution multi-agents pour enquêter sur les menaces internes, le choix du modèle linguistique utilisé comme base pour certains agents s'est avéré déterminant pour la performance globale du système. Chaque LLM présente des forces et des faiblesses distinctes qui

influencent directement des aspects clés tels que la précision de la détection des anomalies, la capacité à analyser des séquences d'actions sur de longues périodes et la réactivité face à des scénarios en temps réel. Par exemple, certains modèles, tels que ceux dotés d'une grande fenêtre contextuelle comme *BaronLLM* [79], sont plus efficaces pour identifier les comportements suspects dans les données historiques. D'autres plus légers, avec pour exemple Gemma (1B) [80] qui a beaucoup moins de paramètres, offrent une latence réduite, idéale pour l'analyse en temps réel. Afin de présenter une évaluation de la solution proposée différents modèles ont été sélectionnés et présentés dans le Tableau 5.1.

Tableau 5.1 : Caractéristique des modèles étudiés

Modèle noyau des agents	**Caractéristique**
Llama (8B) [81]	Modèle généraliste polyvalent avec 8 milliards de paramètres, offrant un équilibre entre performance et ressources pour des tâches variées
BaronLLM (8B) [79]	Modèle spécialisé dans le traitement des contextes de tests de pénétration et d'investigation dans le cadre de la sécurité informatique
Mistral Instruct (7B) [82]	Modèle général optimisé pour les contextes d'instruction
Lily Instruct (7B) [83]	Dérivé de Mistral, entrainé spécifiquement pour l'analyse de contexte de sécurité informatique
Gemma (1B) [80]	Ultraléger et rapide mais limité aux tâches préformatées et spectre de connaissance peu riche

Le choix s'est porté sur les modèles les plus performants, leur poids, ou le pré-entrainement spécifique à des tâches spécifiques de cybersécurité. Ce panel vise à comprendre les différences de performance de l'investigation selon le modèle et à exposer l'importance du modèle pour chaque agent. Il est bon de noter que l'évolution des modèles de langage est rapide et peut améliorer les performance des agents et donc de la solution multi-agents.

5.4.2 Evaluation temporelle

L'objectif de ce mémoire est le temps réel. Ainsi la performance temporelle du module multi-agent est indispensable. En effet, il est nécessaire que le temps d'investigation total par évènement soit suffisamment restreint pour suivre le flux continu d'évènements fourni par le module de détection, sans quoi le décalage temporel ne permettrait pas d'identifier un scénario menaçant avant son action décisive.

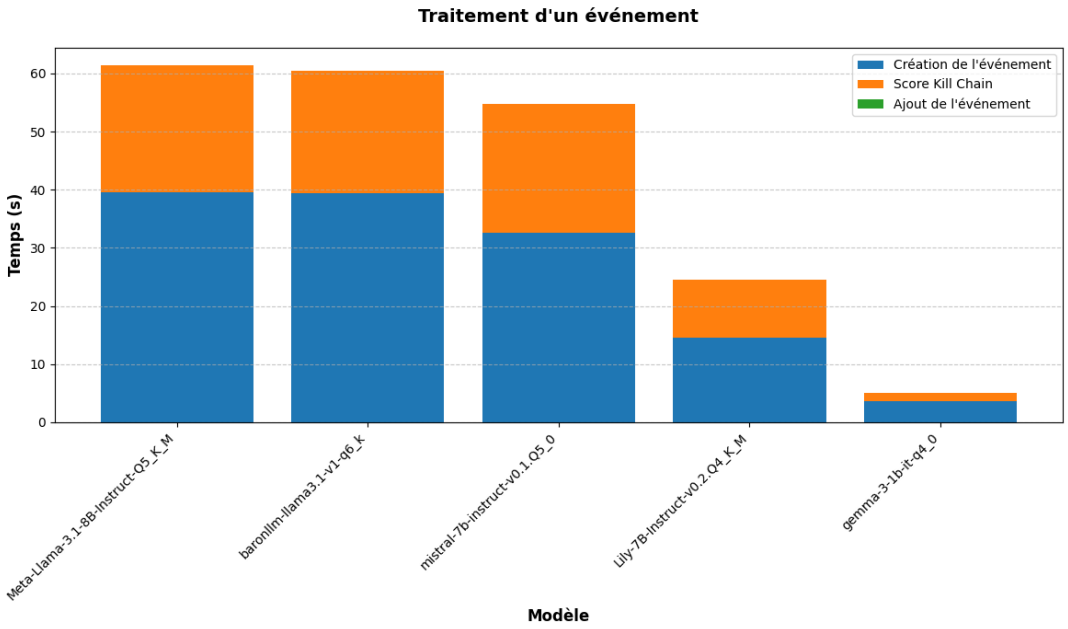

Figure 5.9 : Temps d'investigation par évènement

La Figure 5.9 expose les temps moyens de traitement d'un évènement en seconde (calculé sur 20 000 évènement analysés selon les parts présentes dans le jeu de donnée du CERT), réparti selon la part de l'agent spécifique en bleu ou de l'agent de placement dans la chaine d'attaque en orange. Une telle performance dépend essentiellement des modèles qui sont noyaux des agents, car les temps de traitement autres y sont négligeables. Les modèles de langages les plus consommateurs de temps ne sont pas forcément les plus précis, mais sont souvent ceux qui ont le volume de paramètres le plus large et les plus spécialisés comme *Lilly* ou *BaronLLM*.

Pour conclure quant à la performance temporelle brute de la solution, la capacité dans les conditions d'essai fournissent un nombre moyen d'évènement traitable par minute. Cet indicateur, présenté dans le Tableau 5.2 est essentiel pour le choix du modèle noyau des différents agents. En effet, dans une perspective de raccordement des deux modules, de détection présenté au CHAPITRE 4 et d'investigation ici décrit, il est nécessaire que le module d'investigation puisse computationnellement et temporellement suivre le flux induit par le module de détection.

Tableau 5.2 : Capacité temporelle d'investigation

Modèle noyau des agents	**Fréquence de traitement (nb/min)**
Llama (8B) [81]	0.98
BaronLLM (8B) [79]	0.99
Mistral Instruct (7B) [82]	1.10
Lily Instruct (7B) [83]	2.44
Gemma (1B) [80]	11.70

Ce tableau met en lumière ce que les modèles avec le plus de paramètres sont les plus lents et donc ceux qui permettent le traitement du moins d'évènements par minute. Comparant les modèles *Lily Instruct* et *Mistral Instruct*, nous pouvons remarquer que le modèle spécialisé (*Lily Instruct*) est plus rapide.

5.4.3 Évaluation qualitative

La mesure des performances d'investigation présente une réelle question de recherche sous-jacente : comment évaluer de manière rigoureuse une investigation. C'est à dire évaluer la classification d'une suite d'évènement selon une norme (ici la chaine d'attaque), à partir de données non labellisées. En effet la réponse simple à celle-ci serait de produire un jeu de donnée parallèle donc la labellisation est détaillée, jusqu'à la remise en contexte de chaque évènement au sein de ses parents et fils. Cependant, c'est une solution qui apparait peu efficace temporellement et dans le cadre de ce mémoire. De plus, dans un cas réel d'investigation au sein d'un SOC par des analystes, les données ne sont pas labellisées et l'investigation est le fruit de la réflexion d'un ou plusieurs analystes, pendant une durée pouvant être longue.

L'évaluation de notre solution nécessite une double approche : une analyse globale du résultat final, mais aussi une vérification individuelle de chaque agent. En effet, une défaillance locale (comme la surinterprétation d'événements ou la mauvaise gestion d'URLs irréalistes (un biais spécifique à notre jeu de données)) ne compromet pas nécessairement la cohérence de l'agent chargé de situer l'attaque dans la Kill Chain. Par ailleurs, les modèles de langage constituant le cœur décisionnel des agents étant déterministes (une même entrée produit une même sortie), nous pouvons valider la solution via une première évaluation qualitative basée sur un ensemble de scénarios reconstruits. Le jeu de données du CERT contient des scénarios de menace interne dont nous avons la connaissance. Il est donc possible de choisir des sessions en lien (et d'autres non) avec ces

scénarios et de juger le résultat de l'investigation, aidés de nos connaissances en cybersécurité (tel qu'un analyste en SOC le ferait manuellement).

En comparant les pourcentage de classification de la solution d'investigation en fonction des modèles choisis pour noyaux des agents, il est possible de comprendre ceux qui rendent des conclusions plus proches de ce que la réalité du jeu de donnée est. En effet, le Tableau 5.3 recense les statistiques de réponse de la solution pour les différents modèles sélectionnés, par rapport à l'analyse d'une période d'activité ne comportant pas de scénario de menace interne. L'évaluation avec le modèle *Llama* a par exemple classé 54% des évènements analysés comme Recrutement ou activité normale, 32% comme reconnaissance, 13.5% comme acquisition et enfin 0.5% comme exploitation. La période analysée ne comptant pas de menace interne, il ne devrait pas y avoir d'évènement classé en exploitation (étape 4) et très peu en acquisition (étape 3).

Cependant, la répartition des classifications produites par les 5 modèles sont plus aléatoires et peut mettre en avant la difficulté d'investigation d'un flux d'évènement sans anomalies. Dans la période d'activité normale investiguée (93.9% d'évènements web et 3.2% de connexions et 1.9% de périphériques) les activités web ont une forte majorité et ces évènements ne sont pas classés comme « exécution » mais bien (à tort) comme « acquisition » ou pré-positionnement.

Tableau 5.3 : Comparaison statistique des modèles noyaux des agents

Modèle noyau	**% de 1**	**% de 2**	**% de 3**	**% de 4**
Llama (8B) [81]	54.0	32.0	13.5	0.5
BaronLLM (8B) [79]	53.2	27.9	18.5	0.4
Mistral Instruct (7B) [82]	64.2	26.1	29.3	0.4
Lily Instruct (7B) [83]	73.9	4.5	21.2	0.4
Gemma (1B) [80]	82.4	11.1	6.1	0.4

L'évaluation de l'investigation seule implique en effet que les agents vont analyser en majorité une activité normale. Ainsi on s'attend à ce que les agents ne soulèvent peu ou aucune anomalies. Cependant, on constate que beaucoup d'évènements, en particulier des requêtes http sont classées dans la chaine d'attaque. En effet, le jeu de donnée du CERT, comporte des données générées sur

des modèles réels, mais avec une composante aléatoire. Par exemple, les url consultés ont souvent un début de chemin tout à fait réaliste, mais se concluent par des suites de caractères aléatoires. Ainsi, face à ces formats anormaux de chemins, mais aussi contenus de courriels dénués de sens, nous remarquons que notre solution multi-agent a un nombre de faux positifs très élevés, allant jusque 23% des positifs, et 38% des évènements « web » analysés sont caractérisés comme *attack-tool*, *malware*, *data-exfiltration* ou même *dark-web-access*. Sur la base des URL extraits du CERT et d'URL réels, le Tableau 5.4 compare les pourcentages d'URL selon certaines caractéristiques choisies par l'agent web.

Tableau 5.4 : Comparaison de l'analyse d'URL du CERT et réels

	URL du CERT	**URL réels**
Caractéristique	**% sur le total des évènements**	**% sur le total des évènements**
normal	22,4	72.3
personal-browsing	18,9	17.6
non-working	6.7	0.1
malware	10.6	0.0
attack-tool	10.3	0.0
data-exfiltration	5,1	0.1
dark-web-access	12,3	0.0

En analysant spécifiquement la liste d'url normaux et réels l'agent « web » a un taux de faux positifs quasiment nul, c'est-à-dire inférieur à 2% des positifs, avec 92% de tous les évènements « web » caractérisés comme *normal* ou *personal-browsing*. Cela laisse présager que dans un contexte réel, le système multi-agent ne serait pas sujet à un tel taux de faux-positifs.

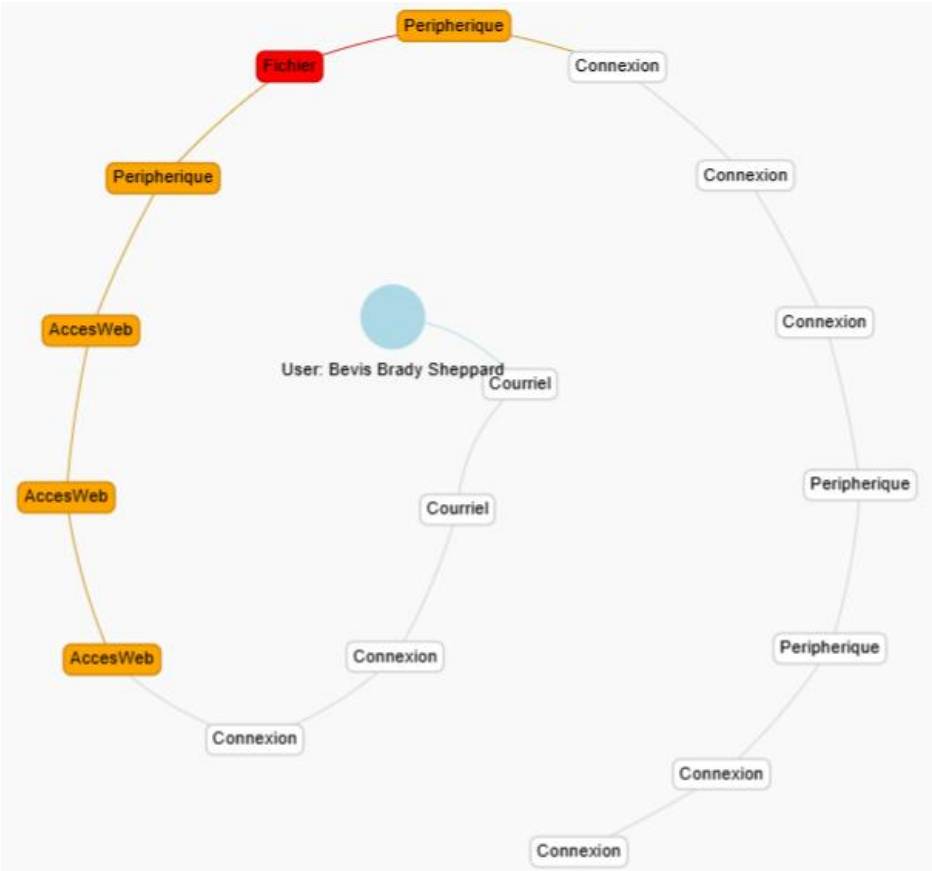

Figure 5.10 : Résultat pour un scénario de menace par « keylogging »

La Figure 5.10 présente par exemple un scénario de menace correspondant à un keylogger déployé sur le poste d'un collègue. On remarque que la solution d'investigation a bien relevé les étapes préparatoires de consultation de sites parlant de « keylogging » comme pré-positionnement (en orange), puis l'action de manipulation du fichier en tant qu'exploitation (en rouge). Cependant, on peut supposer que les connexions et utilisation de périphérique arrivant juste après la consultation du fichier de « keylogging » sont liées à cette attaque et ne sont pas relevées. Cela pourrait d'expliquer par le fait qu'il n'y ait plus de manipulation du fichier malveillant.

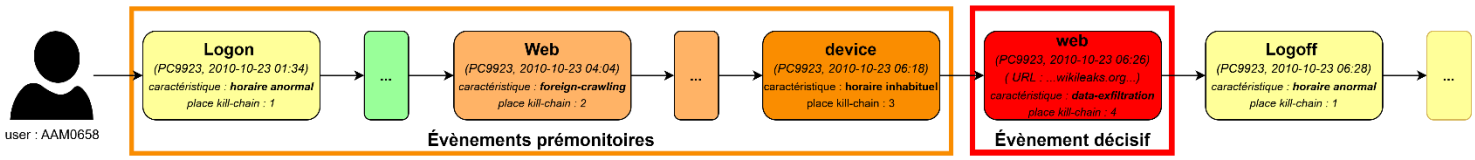

Figure 5.11 : Investigation d'un scénario d'exfiltration web

Dans les 2 autres scénarios de menace présent dans la version 4.2 du jeu de donnée du CERT (exfiltration web, dont la Figure 5.11 illustre le scénario et les évènements prémonitoire et le décisif, et via un périphérique USB précédés de contextes initiateurs) les évènements de préposition sont bel et bien identifiés comme tel (utilisation USB inhabituelle, visite fréquente de site de recrutement concurrents) et les évènements considérés comme d'exploitation sont aussi correctement identifiés.

D'autre part, l'incohérence des adresses web et autres contenus sémantiques pousse à souligner la nécessité de faire évoluer le jeu de donnée du CERT vers plus de réalisme. Enfin, cette problématique se résout partiellement lorsque le module d'investigation est placé en complément du modèle de détection, les évènements analysés ayant donc un fort potentiel d'anomalie.

CHAPITRE 6 CONCLUSION ET RECOMMANDATIONS

Ce mémoire avait pour objectif de proposer une approche innovante pour la détection et l'investigation en temps réel de la menace interne, en combinant des modèles d'apprentissage automatique et une architecture multi-agents inspirée des principes d'intelligence collaborative. Les travaux présentés ont permis de définir, d'expérimenter et d'évaluer les bases d'une solution opérationnellement viable, conçue pour assister les analystes en SOC dans leurs tâches de détection et d'analyse comportementale.

L'approche proposée repose sur la segmentation du flux d'évènements en fenêtres temporelles glissantes et sur une articulation entre détection automatique et investigation guidée par des agents cognitifs. Cette dualité rend possible une analyse continue des activités internes, tout en maintenant une réactivité conforme aux exigences d'un environnement temps réel.

Les expérimentations menées ont montré que la méthodologie adoptée permettait une identification des comportements suspects, tout en enrichissant le contexte de compréhension des incidents par une explicabilité accrue due à l'investigation. L'intégration de modèles fondés sur les graphes a permis d'ajouter une dimension relationnelle à la détection et les LLM participent quant à eux à ajouter une étude sémantique indispensable à une investigation face aux méthodes existantes essentiellement statistiques et logiques.

Cette solution s'inscrit dans une logique d'aide à la décision, et non de remplacement de l'expertise humaine. Bien que l'automatisation de certaines procédures de réponse à incident pourrait être implémentée son ambition n'est pas d'automatiser, mais de fournir aux analystes des outils intelligents capables d'accélérer l'investigation et de hiérarchiser les alertes. Les résultats obtenus suggèrent qu'une telle approche peut constituer un maillon essentiel entre les systèmes UBA/UAM existants et les processus d'investigation manuelle.

6.1 Synthèse des résultats

6.1.1 Performance

Les résultats des évaluations des modules de détection et d'investigation sont respectivement détaillés dans les CHAPITRE 4 et CHAPITRE 5. Le module de détection présente des performance variables selon les paramètres choisis comme la taille des fenêtres (permettant un

contexte plus ou moins large). Le tableau rapporte une précision de 1,0 pour les trois algorithmes et un score F1 supérieur à 0.90.

Tableau 6.1 : Meilleurs métriques du module de détection selon les algorithmes

Model	Time Parameters	F1 Score	Precision	Accuracy	Recall
NLP (BERT)	All	0.9060	1.0000	0.9900	0.8300
LSTM Autoencoder	All	0.9095	1.0000	0.9900	0.8334
GNN + IF	(10, 45)	0.9986	1.0000	0.9996	0.9972

L'évaluation qualitative du module d'investigation détaillée en 5.4 montre une dépendance accrue aux modèles choisis pour les agents. La structure multi-agents présente tout de même un intérêt au vu de la cohérence de la classification des évènements sur les scénarios de menace testés et de la possibilité de naviguer dans les scénarios analysés pour comprendre les anomalies relevées par le module de détection.

Lorsque le module d'investigation agit à la suite du module de détection, seules les évènements relevés comme anormaux sont remis en contexte et analysés. Alors, les agents n'analysent que des évènement à fort potentiel de compromission, et les cas de nombreux faux-positifs relevés en 5.4.3 sont donc limités. Sur les trois types de scénarios de menace présents dans la version 4.2 du jeu de données, des périodes temporelles contenant dix de chaque ont été analysées par les deux modules mis bout à bout. Sur ces 30 scénarios, tous les évènements considérés comme décisifs ont été détecté puis investigués. Dans deux cas seulement les évènements préalables n'ont pas été identifié et placé comme « repositionnement » ou « acquisition » dans la chaine d'attaque.

Ainsi, l'analyse des scénarios de menace interne avérés présents dans le jeu de connées du CERT démontre que la détection et l'investigation de ces scénarios est fonctionnelle et permet la détection et la compréhension desdits scénarios.

6.1.2 Temps réel

Pour assurer un fonctionnement continu, et donc espérer satisfaire à la définition de temps de la partie 2.4.1, le module de détection doit être capable d'analyser une fenêtre d'évènements dans un temps au plus égal au temps de glissement choisi. La représente le temps d'analyse en fonction des paramètres des fenêtres pour chacun des algorithmes étudiés. Dans les cas d'utilisation du GNN ou du LSTM, la durée maximale d'analyse d'une fenêtre est de 17 secondes, mais atteint 3 minutes

pour le NLP. Ainsi dans les trois cas, le pas de glissement minimal est de 3 minutes afin de permettre un fonctionnement continu.

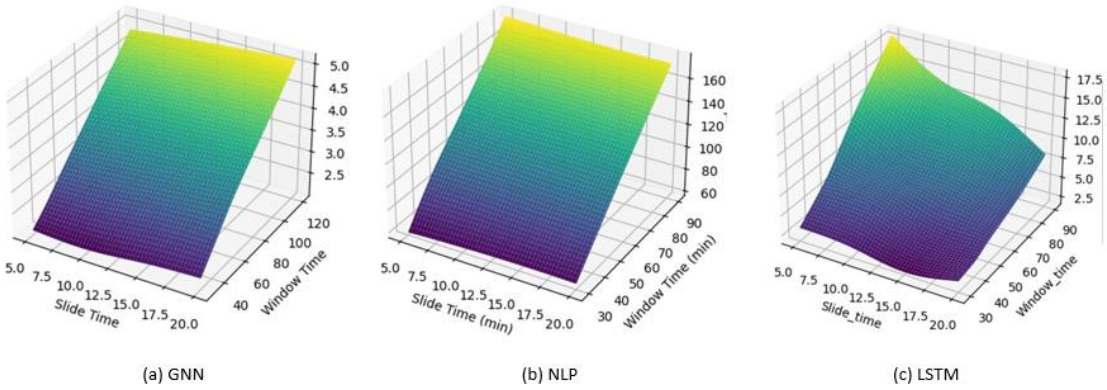

Figure 6.1 : Temps d'analyse (en secondes) par fenêtre selon les paramètres

Le module d'investigation doit quant à lui traiter les évènements d'une fenêtre dans le temps de glissement du module de détection. Il est donc nécessaire de paramétrer le fenêtrage temporel de la détection et le nombre d'anomalies remontées par fenêtre pour permettre au module d'investigation un traitement continu. Le Tableau 6.2 présente pour trois des LLM analysés le nombre maximal d'anomalies pouvant être investiguées pour une fenêtre temporelle donnée. Dans le cas de l'utilisation de *Lily Instruct*, avec une fenêtre de 45 minutes glissant toute les 10 minutes, seules 24 anomalies peuvent être investiguées.

Tableau 6.2 : Jeux de paramètres compatibles avec un fonctionnement en continu

Fenêtre temporelle	**Anomalies par fenêtre**	**Modèle noyau des agents**
(10,45)	24	Lily Instruct (7B) [83]
(20,120)	20	BaronLLM (8B) [79]
(5,15)	58	Gemma (1B) [80]

Ce résultat, bien qu'encourageant et montrant la possibilité d'un fonctionnement continu met aussi en avant une limite quant à la complexité de paramétrage de la solution. Dans l'optique où les fenêtres temporelles seraient dynamiques pour s'adapter à la densité du flux d'entrée, le nombre maximal d'anomalies devrait être redéfini. Il ne s'agit pas d'une limite absolue mais plus d'un élément important à prendre en compte lors du déploiement de la solution.

6.2 Limites de la recherche

Malgré ces avancées, plusieurs limites doivent être reconnues. Elles ne remettent pas en cause la validité de la démarche, mais en précisent les contours et les conditions d'étude.

Sur le plan méthodologique, les expérimentations reposent essentiellement sur des jeux de données synthétiques tels que CERT. Si ces corpus présentent un bon équilibre entre diversité et réalisme, ils ne reflètent pas pleinement la complexité et la variabilité des comportements humains réels et présentent des anomalies de synthétisation. Les biais inhérents à la simulation peuvent ainsi influencer la performance des modèles et limiter la portée des conclusions. Par ailleurs, les scénarios étudiés privilégient les comportements intentionnellement malveillants, au détriment des actions accidentelles ou négligentes, pourtant plus fréquentes dans les contextes organisationnels.

Sur le plan technique, la performance des modèles dépend étroitement du réglage de leurs paramètres et du volume de données traité. La combinaison de plusieurs modèles formant la solution globale (réseaux de neurones, forêts d'isolation, et modules sémantiques) entraîne une complexité computationnelle non négligeable, pouvant devenir un frein à l'analyse à grande échelle. Une étude et optimisation du traitement parallèle ainsi qu'une meilleure gestion du streaming de données s'imposent donc pour garantir la scalabilité de la solution mais cela repose entre autres sur le matériel computationnel disponible qui pourrait surpasser les faibles ressources utilisées pour ce mémoire.

Enfin, sur le plan architectural, la coordination des agents reste gouvernée par des règles logiques. Si cette structure assure une cohérence certaine, elle peut limiter l'autonomie et la capacité d'adaptation du système face à des contextes nouveaux. Le développement d'un agent coordinateur central, qui pourrait être chargé de la supervision et de la consolidation des inférences, constitue également un axe d'amélioration identifié.

Ces limites traduisent la complexité d'un domaine où la temporalité, la sémantique et la logique comportementale doivent cohabiter sans se contredire.

6.3 Perspectives et recommandations

Les perspectives ouvertes par ces travaux sont nombreuses et appellent à la poursuite de la recherche dans plusieurs directions complémentaires.

La première concerne la validation en environnement réel. L'évaluation de la solution au sein d'un SOC ou d'un grand groupe industriel permettrait de mesurer sa robustesse dans des conditions de production, confrontée à des volumes de données massifs et à des scénarios non simulés. Ce type de déploiement offrirait aussi l'opportunité d'intégrer d'avantages d'indicateurs organisationnels (tels que les métadonnées RH, les contextes de projet ou d'autres nouveaux) afin d'élargir la perception du risque interne au-delà de la seule sphère technique.

Au-delà de considérer des données d'entrée réelles, une mise en pratique concrète des travaux de ce mémoire permettrait d'ouvrir la réflexion sur la mise en œuvre d'une telle solution dans un contexte réel. Par exemple, déployer les modules tel un EDR, en répartissant la charge computationnelle sur l'ensemble des machines des employés, ou bien la mettre en place de manière centralisée sur des serveurs dédiés et ainsi évaluer les performance temporelles réelles.

La seconde voie consiste à renforcer l'autonomie et la spécialisation des agents. Leur évolution vers des entités capables d'apprentissage contextuel, adaptant leurs stratégies selon les environnements et leurs performances passées, représente une piste prometteuse. L'introduction d'un agent coordinateur ou d'un module de gouvernance cognitive favoriserait une meilleure cohérence entre les analyses et une hiérarchisation plus fine des signaux détectés.

Un troisième axe de développement réside dans la spécialisation des modèles d'intelligence artificielle pour la cybersécurité interne. Des LLM entraînés spécifiquement sur des corpus de journaux d'événements, d'échanges de courriels ou de politiques d'accès internes pourraient offrir une compréhension contextuelle beaucoup plus précise, réduisant les ambiguïtés d'interprétation.

Il est important de ne pas oublier que la gestion de la sécurité informatique en entreprise ne se limite pas à la partie technique, qu'elle soit autour de la détection, la protection ou l'investigation. La sécurité si elle n'est pas incluse dans une culture générale est fortement limitée. Il est donc nécessaire de réfléchir aux implémentations organisationnelles de tels sujet, tant dans la gouvernance que dans les ressources humaines et les bonnes pratiques.

Enfin, il convient de souligner les enjeux éthiques et juridiques associés à ces approches. La surveillance des activités internes, même à des fins de sécurité, doit s'accompagner de garanties strictes quant à la protection de la vie privée, à la transparence des algorithmes et à la proportionnalité des analyses. La mise en place d'une gouvernance claire et de mécanismes

d'explicabilité constitue une condition indispensable pour assurer la légitimité et l'acceptabilité de telles solutions.

À terme, la réflexion pourrait s'étendre vers des mécanismes d'automatisation raisonnée de la remédiation, dans lesquels l'IA proposerait des actions correctives sous supervision humaine, accélérant la réponse tout en préservant la responsabilité du jugement.

6.4 Synthèse

En définitive, ce mémoire démontre la pertinence et la faisabilité d'une approche de la détection et l'investigation de la menace interne en temps réel de manière proactive et opérable.

Les contributions proposées ouvrent la voie à une nouvelle génération de systèmes capables d'allier puissance analytique, cohérence sémantique et compréhension comportementale, au service d'une sécurité plus humaine et plus intelligente.

Si des défis subsistent, ils constituent autant d'occasions de progrès, invitant à poursuivre l'exploration d'un domaine où la technologie ne prend son sens que dans la mesure où elle renforce la confiance, la vigilance et la lucidité des organisations face à leurs propres fragilités.

RÉFÉRENCES

[1] A. Mhalla, *Technopolitique: Comment la technologie fait de nous des soldats*. Seuil, 2024.

[2] M. Maybury *et al.*, "Analysis and detection of malicious insiders," mars 2005.

[3] "The Cost of Insider Threat : Global Report," IBM Security, 2020.

[4] "Cost of Insider Risks," Ponemon Institute, 2023.

[5] J. Xiao *et al.*, "Robust Anomaly-Based Insider Threat Detection Using Graph Neural Network," *IEEE Transactions on Network and Service Management*, vol. 20, nº 3, p. 3717-3733, sept. 2023,

[6] R. B. Peccatiello, J. J. C. Gondim, et L. P. F. Garcia, "Applying One-Class Algorithms for Data Stream-Based Insider Threat Detection," *IEEE Access*, vol. 11, p. 70560-70573, 2023,

[7] A. P. Singh et A. Sharma, "A systematic literature review on insider threats." arXiv, déc. 10, 2022.

[8] A. Badaoui, "Proposition d'approche pour la détection de menace interne," masters, Polytechnique Montréal, 2021.

[9] F. Olaoye et K. Potter, "Behavioral Analytics for Insider Threat Detection," *Journal of Cybersecurity*, oct. 2024.

[10] K. Scarfone et P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," National Institute of Standards and Technology, Rapport technique NIST Special Publication (SP) 800-94 Rev. 1 (Withdrawn), juill. 2012.

[11] M. N. Al-Mhiqani *et al.*, "Insider Threat Detection in Cyber-Physical Systems: A Systematic Literature Review," *Computers and Electrical Engineering*, vol. 119, p. 109489, oct. 2024,

[12] E. Cole et S. Ring, *Insider Threat: Protecting the Enterprise from Sabotage, Spying, and Theft*, Elsevier. 2005.

[13] A. Kim *et al.*, "SoK: A Systematic Review of Insider Threat Detection," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 10, nº 4, p. 46-67, déc. 2019,

[14] J. A. Stankovic et K. Ramamritham, *Hard Real-Time Systems*. Washington, DC, USA : IEEE Computer Society Press, 1988.

[15] H. Kopetz, *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Springer Science & Business Media, 2011.

[16] G. C. Buttazzo, *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*, vol. 24. dans Real-Time Systems Series, vol. 24. Boston, MA : Springer US, 2011.

[17] M. Schonlau et M. Theus, "Detecting masquerades in intrusion detection based on unpopular commands," *Information Processing Letters*, vol. 76, nº 1, p. 33-38, nov. 2000,

[18] M. B. Salem et S. J. Stolfo, "Modeling User Search Behavior for Masquerade Detection," dans *Recent Advances in Intrusion Detection*, R. Sommer, D. Balzarotti, et G. Maier, édit.,

dans Lecture Notes in Computer Science, vol. 6961. Berlin, Heidelberg : Springer Berlin Heidelberg, 2011, p. 181-200.

[19] Athul Harilal *et al.*, "The Wolf Of SUTD (TWOS): A Dataset of Malicious Insider Threat Behavior Based on a Gamified Competition," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 9, nº 1, p. 54-85, mars 2018,

[20] G. Grinstein *et al.*, "VAST 2009 challenge: An insider threat," communication présentée à 2009 IEEE Symposium on Visual Analytics Science and Technology, 2009, p. 243-244.

[21] C. Laprade, B. Bowman, et H. H. Huang, "PicoDomain: A Compact High-Fidelity Cybersecurity Dataset." arXiv, août 20, 2020.

[22] "Insider Threat Test Dataset." Carnegie Mellon University, sept. 30, 2020.

[23] P. Pols et F. Dominguez, "Raising Resilience Against Advanced Cyber Attacks".

[24] C. Neal *et al.*, "Semantic and Graph-Based Unsupervised Learning for Insider Threat Detection Using User Activity Sequences," communication présentée à 2025 22th Annual International Conference on Privacy, Security and Trust (PST), 2025.

[25] N. Baghalizadeh-Moghadam *et al.*, "NLP and Neural Networks for Insider Threat Detection," communication présentée à 2024 IEEE 23nd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2024.

[26] Y. Gong *et al.*, "Graph-Based Insider Threat Detection: A Survey," *Computer Networks*, vol. 254, p. 110757, déc. 2024,

[27] T. Davies, "A Review of Topological Data Analysis for Cybersecurity." arXiv, févr. 16, 2022.

[28] A. Gamachchi, L. Sun, et S. Boztas, "A Graph Based Framework for Malicious Insider Threat Detection." arXiv, sept. 01, 2018.

[29] A. Gamachchi et S. Boztas, "Insider Threat Detection Through Attributed Graph Clustering," communication présentée à 2017 IEEE Trustcom/BigDataSE/ICESS, 2017, p. 112-119.

[30] N. d'Ambrosio, G. Perrone, et S. P. Romano, "Including insider threats into risk management through Bayesian threat graph networks," *Computers & Security*, vol. 133, p. 103410, oct. 2023,

[31] K. A. Saint-Hilaire *et al.*, "Attack-Defense Graph Generation: Instantiating Incident Response Actions on Attack Graphs".

[32] M. Blay, *Dictionnaire des concepts philosophiques*, Larousse. dans In Extenso. 2013.

[33] S. E. Parkin, A. van Moorsel, et R. Coles, "An information security ontology incorporating human-behavioural implications," communication présentée à Proceedings of the 2nd international conference on Security of information and networks, dans SIN '09. New York, NY, USA : Association for Computing Machinery, 2009, p. 46-55.

[34] K. Saint-Hilaire *et al.*, "Automated Enrichment of Logical Attack Graphs via Formal Ontologies," dans *ICT Systems Security and Privacy Protection*, N. Meyer et A. Grocholewska-Czuryło, édit., dans IFIP Advances in Information and Communication Technology, vol. 679. Cham : Springer Nature Switzerland, 2024, p. 59-72.

[35] D. L. Costa *et al.*, "An Insider Threat Indicator Ontology".

[36] J. von der Assen *et al.*, "The Danger Within: Insider Threat Modeling Using Business Process Models." arXiv, sept. 03, 2024.

[37] D. C. Le, N. Zincir-Heywood, et M. I. Heywood, "Analyzing Data Granularity Levels for Insider Threat Detection Using Machine Learning," *IEEE Transactions on Network and Service Management*, vol. 17, nº 1, p. 30-44, mars 2020,

[38] D. C. Le et N. Zincir-Heywood, "Exploring Anomalous Behaviour Detection and Classification for Insider Threat Identification," *International Journal of Network Management*, vol. 31, nº 4, p. e2109, juill. 2021,

[39] D. Lin (janv. 20, 2017) A User and Entity Behavior Analytics Scoring System Explained. [En ligne]. Disponible : https://www.exabeam.com/blog/ueba/a-user-and-entity-behavior-analytics-scoring-system-explained/

[40] B. Predovich et D. Gopal, "Market Guide for Insier Risk Managment," Gartner, mars 2025.

[41] A. Khraisat *et al.*, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, nº 1, p. 20, juill. 2019,

[42] A. Zargar, A. Nowroozi, et R. Jalili, "XABA: A zero-knowledge anomaly-based behavioral analysis method to detect insider threats," communication présentée à 2016 13th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC), 2016, p. 26-31.

[43] H. Teymourlouei et V. E. Harris, "Preventing Data Breaches: Utilizing Log Analysis and Machine Learning for Insider Attack Detection," communication présentée à 2022 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA : IEEE, 2022, p. 1022-1027.

[44] D. C. Le et A. N. Zincir-Heywood, "Machine Learning Based Insider Threat Modelling and Detection".

[45] X. Bao, T. Xu, et H. Hou, *Network Intrusion Detection Based on Support Vector Machine*. 2009, p. 4.

[46] F. Janjua *et al.*, "Handling Insider Threat Through Supervised Machine Learning Techniques," *Procedia Computer Science*, vol. 177, p. 64-71, janv. 2020,

[47] P. Manoharan *et al.*, "Insider Threat Detection Using Supervised Machine Learning Algorithms," *Telecommunication Systems*, vol. 87, nº 4, p. 899-915, déc. 2024,

[48] L. Breiman *et al.*, *Classification and Regression Trees*. Taylor & Francis, 1984.

[49] R. Abdulhammed *et al.*, "Deep and Machine Learning Approaches for Anomaly-Based Intrusion Detection of Imbalanced Network Traffic," *IEEE Sensors Letters*, vol. 3, nº 1, p. 1-4, janv. 2019,

[50] H. Park *et al.*, "BGP Dataset-Based Malicious User Activity Detection Using Machine Learning," *Information*, vol. 14, nº 9, p. 501, sept. 2023,

[51] G. Gavai *et al.*, "Supervised and Unsupervised Methods to Detect Insider Threat from Enterprise Social and Online Activity Data".

[52] D. Karev, C. McCubbin, et R. Vaulin, "Cyber Threat Hunting Through the Use of an Isolation Forest," communication présentée à Proceedings of the 18th International Conference on

Computer Systems and Technologies, dans CompSysTech ’17. New York, NY, USA : Association for Computing Machinery, 2017, p. 163-170.

[53] N. K. Nanamou *et al.*, "From Traits to Threats: Learning Risk Indicators of Malicious Insider Using Psychometric Data," communication présentée à Information Systems Security, V. T. Patil, R. Krishnan, et R. K. Shyamasundar, édit., Cham : Springer Nature Switzerland, 2025, p. 180-200.

[54] S. A.-D. Qawasmeh et A. A. S. AlQahtani, "Beyond Firewall: Leveraging Machine Learning for Real-Time Insider Threats Identification and User Profiling," *Future Internet*, vol. 17, nº 2, p. 93, févr. 2025,

[55] T. Rashid, I. Agrafiotis, et J. R. C. Nurse, "A New Take on Detecting Insider Threats: Exploring the Use of Hidden Markov Models," communication présentée à Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats, dans MIST ’16. New York, NY, USA : Association for Computing Machinery, 2016, p. 47-56.

[56] A. Ambre et N. Shekokar, "Insider Threat Detection Using Log Analysis and Event Correlation," *Procedia Computer Science*, vol. 45, p. 436-445, janv. 2015,

[57] E. T. Axelrad *et al.*, "A Bayesian Network Model for Predicting Insider Threats," communication présentée à 2013 IEEE Security and Privacy Workshops, San Francisco, CA : IEEE, 2013, p. 82-89.

[58] S. Bertrand, J. Desharnais, et N. Tawbi, "Unsupervised User-Based Insider Threat Detection Using Bayesian Gaussian Mixture Models," communication présentée à 2023 20th Annual International Conference on Privacy, Security and Trust (PST), 2023, p. 1-10. [En ligne]. Disponible : https://ieeexplore.ieee.org/document/10320169/?arnumber=10320169

[59] A. Zhou *et al.*, "Research on User Behavior Credibility Evaluation Model in Trusted Network," communication présentée à Signal and Information Processing, Networking and Computers, Y. Wang, Y. Liu, J. Zou, et M. Huo, édit., Singapore : Springer Nature, 2023, p. 911-919.

[60] B. Bin Sarhan et N. Altwaijry, "Insider Threat Detection Using Machine Learning Approach," *Applied Sciences*, vol. 13, nº 1, Art. nº 1, janv. 2023,

[61] J. Jiang *et al.*, "Anomaly Detection with Graph Convolutional Networks for Insider Threat and Fraud Detection," communication présentée à MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM), 2019, p. 109-114. [En ligne]. Disponible : https://ieeexplore.ieee.org/document/9020760

[62] F. R. Alzaabi et A. Mehmood, "A Review of Recent Advances, Challenges, and Opportunities in Malicious Insider Threat Detection Using Machine Learning Methods," *IEEE Access*, vol. 12, p. 30907-30927, 2024,

[63] E. Bingham et H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," communication présentée à Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, dans KDD ’01. New York, NY, USA : Association for Computing Machinery, 2001, p. 245-250.

[64] M. Mouyart, G. Medeiros Machado, et J.-Y. Jun, "A Multi-Agent Intrusion Detection System Optimized by a Deep Reinforcement Learning Approach with a Dataset Enlarged Using a

Generative Model to Reduce the Bias Effect," *Journal of Sensor and Actuator Networks*, vol. 12, nº 5, p. 68, oct. 2023,

[65] C. Song *et al.*, "Audit-LLM: Multi-Agent Collaboration for Log-based Insider Threat Detection." arXiv, août 12, 2024.

[66] A. Ferraro, G. M. Orlando, et D. Russo, "Generative Agent-Based Modeling with Large Language Models for insider threat detection," *Engineering Applications of Artificial Intelligence*, vol. 157, p. 111343, oct. 2025,

[67] Welcome to The Apache Software Foundation | Apache Software Foundation. [En ligne]. Disponible : https://www.apache.org/

[68] J. Kreps, N. Narkhede, et J. Rao, "Kafka : a Distributed Messaging System for Log Processing," 2011.

[69] P. Carbone *et al.*, "Apache Flink™: Stream and Batch Processing in a Single Engine".

[70] M. Stonebraker, U. Çetintemel, et S. Zdonik, "The 8 requirements of real-time stream processing," *SIGMOD Rec.*, vol. 34, nº 4, p. 42-47, déc. 2005,

[71] Z. Zhan et S.-K. Kim, "Versatile Time-Window Sliding Machine Learning Techniques for Stock Market Forecasting," *Artificial Intelligence Review*, vol. 57, nº 8, p. 209, juill. 2024,

[72] H. Yhdego, C. Paolini, et M. Audette, "Toward Real-Time, Robust Wearable Sensor Fall Detection Using Deep Learning Methods: A Feasibility Study," *Applied Sciences*, vol. 13, nº 8, Art. nº 8, janv. 2023,

[73] B. Böse *et al.*, "Detecting Insider Threats Using RADISH: A System for Real-Time Anomaly Detection in Heterogeneous Data Streams," *IEEE Systems Journal*, vol. 11, nº 2, p. 471-482, juin 2017,

[74] X. Cai *et al.*, "LAN: Learning Adaptive Neighbors for Real-Time Insider Threat Detection," *IEEE Transactions on Information Forensics and Security*, vol. 19, p. 10157-10172, 2024,

[75] T. Leblanc *et al.*, "Real-Time Anomaly Detection for Event-Based Insider Threat Hunting," communication présentée à Network and System Security, J. Chen, D. He, et W. Xie, édit., Singapore : Springer Nature, 2026, p. 116-132.

[76] A. V. Singh *et al.*, "Hierarchical Multi-agent Reinforcement Learning for Cyber Network Defense." arXiv, sept. 05, 2025. [En ligne]. Disponible : http://arxiv.org/abs/2410.17351

[77] B. R. K. Mantha, M. S. Sonkor, et B. Garcia De Soto, "Investigation of the Cyber Vulnerabilities of Construction Networks Using an Agent-Based Model," *Developments in the Built Environment*, vol. 18, p. 100452, avr. 2024,

[78] Y. Sasaki *et al.*, "Systematic Literature Review of Prompt Engineering Patterns in Software Engineering," communication présentée à 2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC), 2024, p. 670-675.

[79] AlicanKiraz0/Cybersecurity-BaronLLM_Offensive_Security_LLM_Q6_K_GGUF · Hugging Face. [En ligne]. Disponible : https://huggingface.co/AlicanKiraz0/Cybersecurity-BaronLLM_Offensive_Security_LLM_Q6_K_GGUF

[80] (sept. 12, 2025) google/gemma-3-1b-it · Hugging Face. [En ligne]. Disponible : https://huggingface.co/google/gemma-3-1b-it

[81] (déc. 06, 2024) meta-llama/Llama-3.1-8B-Instruct · Hugging Face. [En ligne]. Disponible : https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

[82] mistralai/Mistral-7B-Instruct-v0.3 · Hugging Face. [En ligne]. Disponible : https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3

[83] segolilylabs/Lily-Cybersecurity-7B-v0.2 · Hugging Face. [En ligne]. Disponible : https://huggingface.co/segolilylabs/Lily-Cybersecurity-7B-v0.2