Titre: Title:	Attribution de cyber-attaquants à partir de logs Honeypots
Auteur: Author:	Pierre Crochelet
Date:	2023
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Crochelet, P. (2023). Attribution de cyber-attaquants à partir de logs Honeypots [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/54126/

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Attribution de cyber-attaquants à partir de logs Honeypots

PIERRE CROCHELET
Département de génie informatique et génie logiciel

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

Juillet 2023

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Attribution de cyber-attaquants à partir de logs Honeypots

présenté par **Pierre CROCHELET**
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

REMERCIEMENTS

Tout d'abord, je voudrais remercier toutes les personnes avec qui j'ai eu la chance de collaborer dans le cadre de cette recherche. Nora Boulahia Cuppens et Frédéric Cuppens, mes directeurs de recherche, m'ont donné l'opportunité de m'initier à la recherche académique en travaillant sur un sujet qui m'intéresse. À l'écoute de mes demandes et besoins ainsi que présents dans mes moments de questionnement, ils ont su me diriger vers l'accomplissement de mon travail. Je tiens également à remercier toute l'équipe de Thales Digital Solutions, et particulièrement Alexandre Proulx, toujours enthousiaste et ouvert à toutes propositions avancées. Merci enfin à tous mes camarades de laboratoires et particulièrement Christopher Neal pour son soutien et son aide précieuse lors de la rédaction et de la publication des articles.

Je remercie ma famille, Sarah Crochelet et Martin Crochelet, qui ont toujours été là pour moi.

RÉSUMÉ

Devant la fréquence et la sophistication grandissantes des cyber-menaces, les analystes se retrouvent submergés par un nombre démesuré d'alertes. Ces analystes sont employés pour reconstruire les vecteurs d'attaques et comprendre les ambitions des attaquants. Il devient alors nécessaire pour les entreprises d'investir dans des solutions de renseignements de menaces pour aider ces analystes. Cette tâche de compréhension des attaquants, appelée attribution, est actuellement entièrement réalisée par des humains, la rendant lente et coûteuse. De ce fait, beaucoup d'entreprises investissent dans des solutions défensives sans comprendre le comportement des personnes qui les attaquent. Il en résulte que ces entreprises perdent beaucoup d'argent en se protégeant contre des menaces qui ne les concernent pas.

Notre revue de littérature nous a permis de réaliser que les travaux sur l'attribution étaient encore jeunes, mais semblent se décliner en trois axes principaux. On retrouve ceux utilisant des graphes d'attaques pour représenter les différents chemins que des attaquants pourraient prendre. On a aussi ceux utilisant des modèles de Markov cachés en utilisant les probabilités de transitions pour représenter les choix des attaquants. Enfin, l'axe où l'on retrouve le plus de travaux est l'utilisation de règles pour représenter l'état actuel du monde. Le principal problème des travaux actuels cependant est qu'aucune structure rigoureuse n'est utilisée. Ils définissent donc tous l'attribution de manière différente, ce qui rend leur comparaison complexe.

Cette recherche s'intéresse alors à l'automatisation de certaines parties les plus fastidieuses du processus d'attribution. Nos travaux se situent principalement dans l'axe de recherche d'utilisation de règles pour représenter l'état actuel du monde, mais en utilisant MITRE ATT&CK, une base de connaissances réputée dans le domaine pour définir clairement nos pistes d'attributions. Dans un premier temps, nous nous intéressons à une représentation des attaquants en plusieurs caractéristiques et effectuons une analyse statistique ainsi qu'une analyse par regroupement afin d'identifier les attaquants les plus malveillants et de les ordonnancer. Cela permettrait à un analyste de concentrer son attention en écartant les attaquants bénins et en examinant ceux réalisant les attaques les plus sévères. Dans un second temps, nous utilisons le cadriciel MITRE ATT&CK afin de définir des modes opératoires d'attaquants et implémentons un algorithme pour identifier ceux-ci dans les attaques observées. Cet algorithme donne donc des pistes d'attribution aux analystes qui peuvent ensuite les confirmer ou infirmer.

ABSTRACT

Faced to a growing frequency and complexity of cyber-attacks, network analysts are often overwhelmed with a huge number of alerts. Those analysts are employed to reconstruct the attack vectors and understand adversary behaviours. It becomes crucial for companies to invest in threat intelligence to help those analysts. This task of understanding adversary behaviours, generally called attribution, is currently undertaken by people, making it slow and costly. Therefore, many companies invest in defensive technologies without understanding the attackers and the attacks they face. This results in a huge waste of money as they become protected against threat they do not even face.

Through a literature review, we realised that the works tackling attribution task were still quite young. However, it seems they can be declined in three main directions. Some researches use attack graphs to derive all the possible attacks paths an attacker can take. Others use Hidden Markov Models and use the transition probabilities to represent attackers' choices when trying to breach a network. Finally, some use sets of rules to gain some insight on the current state of the world. However, we realised that a main problem with all these works is that everyone seems to define attribution in their own specific way, making it hardly possible to compare those works.

Our research tackles the attribution problem through automatising its most cumbersome parts. Our work sits primarily in the field of using sets of rules to represent the state of the world, but through using MITRE ATT&CK, a well known knowledge base to define a rigorous framework for our methods. On the one hand, we aim to represent the attackers through characteristics and perform a statistical analysis as well as a clustering analysis to identify the most malicious ones and rank them in terms of severity. This allows a network analyst to discard most of the traffic and focus his attention on the most severe attackers. On the other hand, we aim to use the MITRE ATT&CK framework to define modus operandi of attackers and we implement an algorithm that uses this information to correlate attackers and give attribution leads to an analyst. The analyst can then investigate those leads further to either confirm or refute them.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

ATTA&CK	Adversarial Tactics, Techniques and Common Knowledge
TTP	Tactics, techniques and procedures
IOC	Indicators of compromise
APT	Advanced persistent threat
IP	Internet Protocol
CTI	Cyber threat intelligence
SMOTE	Synthetic Minority Oversampling Technique
NO-DOUBT	Novel Documents Based Attribution
IDS	Intrusion Detection Systems
SIP	Session initiation protocol
UNADA	Unsupervised Network Anomaly Detection Algorithm
C&C	Command and Control
HMM	Hidden Markov Models
IT	Information technology
OT	Operational technology
CAMIAC	Cyber Attack Modeling and Impact Assessment Component
DOS	Denial of Service
CTF	Capture The Flag

CHAPITRE 1 INTRODUCTION

Que ce soit pour travailler ou se divertir, Internet fait aujourd'hui partie intégrante de notre société. Depuis 1983, il a rapidement été adopté et est aujourd'hui utilisé par plus de 5 milliards de personnes, soit 65% de la population mondiale [2]. Étant donné qu'Internet était initialement destiné à un groupe restreint de personnes, la sécurité du modèle n'était pas forcément prise en considération. Toujours basé sur ce modèle initial, l'Internet de nos jours possède beaucoup de failles de sécurité qui permettent à un utilisateur mal avisé de s'introduire dans un réseau ou sur une machine. D'autre part, il permet un détachement de l'identité physique, donnant à tous l'impression d'être anonymes. Tout cela motive les criminels à commettre des scénarios pouvant avoir de gros impacts financiers sur les personnes cibles.

Aujourd'hui, on observe un jeu du chat et la souris entre les attaquants et les défenseurs. D'un côté, les attaquants cherchent continuellement de nouvelles failles et créent de nouveaux logiciels malveillants exploitant ces failles. De l'autre côté, les individus et compagnies mettent tout en œuvre pour rester à jour et se défendre des attaques dont ils et elles sont victimes. Effectivement, d'après un rapport d'Embroker [3], la fréquence des crimes cyber a augmenté de 600% suite à la numérisation des entreprises au temps de la pandémie COVID-19. Mais, il est également mentionné que le budget destiné aux produits défensifs en cybersécurité augmente de 12% à 15% par année depuis 2017.

Comme l'a dit John Chambers, ancien président directeur général de Cisco Systems : "There are two types of companies : those that have been hacked, and those who don't know they have been hacked". Il est donc crucial d'analyser les attaques ainsi que les attaquants afin d'aider ces entreprises à s'en protéger plus efficacement. C'est dans ce contexte que se situe ce travail de recherche. Plus spécifiquement, nous nous intéressons à la problématique d'attribution, qui cherche à caractériser l'attaquant, plutôt que l'attaque, à l'inverse de la détection.

Ce premier chapitre du mémoire explique plus en détail le contexte dans lequel se déroule la recherche, la problématique considérée et les objectifs fixés. L'information contenue dans cette introduction est divisée en quatre sections. Dans la première, nous présentons les différences entre les problématiques de détection et d'attribution et motivons le besoin d'inclure l'attribution dans les technologies défensives. La deuxième section expose la problématique étudiée. Dans la troisième section, les objectifs de recherche sont clairement établis. Enfin, la dernière section détaille la structure du mémoire.

1.1 De la détection à l'attribution

La détection de nouvelles attaques s'articule en deux grands axes : en se basant sur des signatures et en recherchant des anomalies. L'approche basée sur des signatures utilise des règles spécifiques pour définir une attaque, telles que des séquences spécifiques d'octets. Par définition, il faut déjà avoir observé l'attaque afin de pouvoir définir une règle appropriée. Cette approche présente donc les faiblesses qu'elle n'est capable que de découvrir des attaques déjà connues et que sa base de données doit être mise à jour fréquemment [4].

L'approche par anomalie étudie le trafic réseau et le classifie soit comme normal, soit comme anomalie. L'utilisation d'apprentissage machine est prédominante dans ce domaine. D'un côté, l'approche supervisée obtient de bons résultats sur des attaques connues, mais requiert de grandes bases de données étiquetées. D'un autre côté, l'approche non supervisée permet de découvrir de nouvelles attaques, mais présente généralement un plus grand nombre de faux positifs. Malgré toutes les recherches dans le domaine, concevoir un système de détection d'intrusion (IDS) performant reste une tâche difficile [5].

De par leur nature, les IDS -peu importe leur type- ne découvrent que les instances malveillantes de l'attaque. Mais ils ne permettent pas de découvrir le vecteur d'attaque complet utilisé par l'attaquant. Ce point devient de plus en plus important à mesure que les attaques grandissent en complexité, comme les menaces APT (advanced persistent threat). Il devient donc de plus en plus crucial de corréler les différents indicateurs d'attaques observés et d'étudier les chemins d'attaques envisagés par les attaquants. Cette tâche est ce que l'on appelle l'attribution.

1.2 Éléments de la problématique

Le principal problème lorsque l'on considère la tâche d'attribution est la séparation entre les indicateurs observés et la personne physique. En effet, les adresses IP (Internet Protocol) ou les noms de domaines n'identifient pas une personne de la même façon qu'une empreinte digitale ou oculaire [6]. De plus, les attaquants utilisent des techniques trompeuses afin d'induire les analystes dans la mauvaise direction [7, 8]. Certains ont recourt à de l'usurpation d'identité ("spoofing"), afin de paraître comme un utilisateur légitime [9], ou encore manipulent les données dans le système pour cacher et même effacer leurs traces [10].

En plus de tout cela, on observe un phénomène de fatigue d'alerte ("alert fatigue") chez les professionnels [11]. En effet, avec la fréquence en hausse continuelle des attaques, ils reçoivent d'énormes quantités d'alertes, dont seulement une petite quantité requiert effectivement leur attention. Trouver ces alertes significatives revient alors à chercher une aiguille dans une botte

de foin, et elles peuvent facilement passer inaperçues lors d'une perte de concentration [12].

Enfin, lorsque l'on s'intéresse à l'attribution, on rencontre rapidement le problème des données. En effet, à notre connaissance, il n'existe actuellement pas de jeu de données conçu pour étudier ce problème. Suite à une entente avec Thales Digital Solutions Canada, nous avons reçu accès à un jeu de données Honeypot, qui sont notre support pour nos expériences. Cette solution est identifiée par Doynikova E. et al. [13] et adoptée dans la majorité des travaux de recherche portant sur l'attribution.

Cependant, les données telles que reçues par notre partenaire contiennent certains problèmes relatifs aux 5 Vs des mégadonnées ("big data") : volume, valeur, variété, vélocité, véracité [14]. En effet, nous recevons un grand nombre de données, dont toutes n'ont pas la même importance. Par exemple, dans certains cas il y a beaucoup de redondance dans les données, car les évènements sont enregistrés plusieurs fois. Dans d'autres, certains champs sont inutiles, car identiques pour tous les évènements. Enfin, comme le partenaire utilise plusieurs protocoles différents dans ces Honeypots, les données n'ont pas toutes le même format. Dès lors, une standardisation des données est nécessaire afin d'accélérer les capacités des algorithmes utilisés, ainsi que réduire les exigences matérielles nécessaires pour utiliser ceux-ci.

1.3 Objectifs de recherche

Dans un premier temps, il faut donc adapter le format des données reçues afin d'enlever les données inutiles, les redondances et de forcer un format commun. Ensuite, les objectifs de recherches sont de proposer des méthodes d'analyses de ces données pour aider les analystes dans leur tâche d'attribution. Plus spécifiquement, les objectifs sont les suivants :

1. Ordonnancer les logs Honeypots en fonction de la sévérité des attaquants
2. Corréler les attaquants entre eux et identifier des modes opératoires pour chaque groupe identifié

Chacun de ces objectifs a fait l'objet d'un travail lors de notre recherche, que l'on retrouve aux Sections 4 et 5. De plus, chacun de ces objectifs est articulé à partir de sous-objectifs propres. Pour l'ordonnancement des logs, on retrouve les sous-objectifs suivants :

1. Définir des caractéristiques pour représenter les attaquants
2. Identifier les attaquants aux valeurs aberrantes pour les caractéristiques choisies
3. Identifier différents groupes d'attaquants en utilisant des algorithmes d'apprentissage machine non supervisé

Et pour la corrélation des attaquants, on retrouve les sous-objectifs suivants :

1. Définir les attaquants en fonction des techniques utilisées par rapport au cadriciel MITRE ATT&CK
2. Définir différents modes opératoires par rapport au cadriciel MITRE ATT&CK
3. Corréler individuellement les attaquants et les modes opératoires obtenus
4. Définir un algorithme afin d'utiliser l'information des modes opératoires pour corréler les attaquants entre eux

1.4 Plan du mémoire

Le chapitre 2 est une revue de littérature de la tâche d'attribution et des techniques utilisées dans ce travail. Il est divisé en 3 parties : l'étude du cadriciel MITRE ATT&CK, la définition des Honeypots ainsi que l'analyse de leurs données, et enfin les différents axes de recherche autour de l'attribution. Le chapitre 3 explique la méthodologie utilisée et indique la cohérence de notre article avec le premier objectif et les sous-objectifs associés énoncés ci-dessus. Cet articles est présenté au chapitre 4. Ensuite, le chapitre 5 contient les résultats complémentaires qui répondent aux deuxième objectif de recherche et à ses sous-objectifs associés. Enfin, une discussion générale du travail effectué est présentée au chapitre 6.

CHAPITRE 2 REVUE DE LITTÉRATURE

Dans le chapitre précédent, nous avons motivé le besoin d'étudier la problématique d'attribution. Ce chapitre vise à donner les bases théoriques nécessaires à la compréhension des chapitres ultérieurs. Pour ce faire, nous nous basons sur les travaux similaires trouvés dans la littérature et identifions leurs limites, qui ont poussé nos recherches.

Nous commençons par introduire le cadriciel MITRE ATT&CK [15], base de savoir fournie en accès libre par MITRE [16]. MITRE est un organisme sans but lucratif visant à aider l'intérêt public pour la sécurité en tant que conseiller indépendant. Toutes les informations fournies dans ce cadriciel sont fondées sur des faits réels et permettent une définition rigoureuse des attaques et attaquants informatiques. Ensuite, nous définissons ce qu'est un Honeypot et jetons un œil sur les différents travaux d'analyses réalisés sur les données générées par ceux-ci. Enfin, nous identifions les différents axes de recherches pour la problématique de l'attribution et situons notre travail par rapport à ceux-ci.

2.1 MITRE ATT&CK pour la traque des menaces

Le premier modèle d'ATT&CK fut créé en 2013, à la suite d'un besoin de catégoriser systématiquement les comportements des attaquants [17]. Cette base de connaissances est séparée en trois parties : les tactiques qui décrivent les buts à court terme des attaquants, les techniques qui décrivent les moyens employés par les attaquants pour parvenir à leurs buts tactiques, et les procédures qui sont les implémentations spécifiques des attaquants pour une technique précise. Ensemble, ces trois parties forment les TTPs. En plus des TTPs, MITRE maintient une base de données sur les groupes d'attaquants les plus connus, tels que les groupes APT, et documente les techniques que ces groupes utilisent dans leurs attaques.

Pour la tâche d'attribution, toutes ces informations représentent une bonne base de travail. En effet, les groupes d'attaquants représentent une base de données de différents modes opératoires utilisés. De plus, grâce aux tactiques et aux techniques, nous avons des actions types correspondant aux comportements adverses. Ces actions représentent des indicateurs de compromission (IOC) des différents attaquants. Ces IOC sont d'autant plus intéressants qu'ils sont l'information la plus fiable que l'on peut collecter actuellement sur internet concernant un attaquant [18]. En effet, la pyramide de David J. Bianco [1] à la Figure 2.1 mentionne différents IOC et la difficulté pour un attaquant de s'en défaire. On voit par exemple que les valeurs de Hash (identifiant numériques de fichiers) ou les adresses IP ne sont pas de bons

identifiants d'attaquants, car elles sont faciles à modifier. À l'inverse, les outils ainsi que les TTPs utilisés définissent les attaquants de manière plus convaincante.

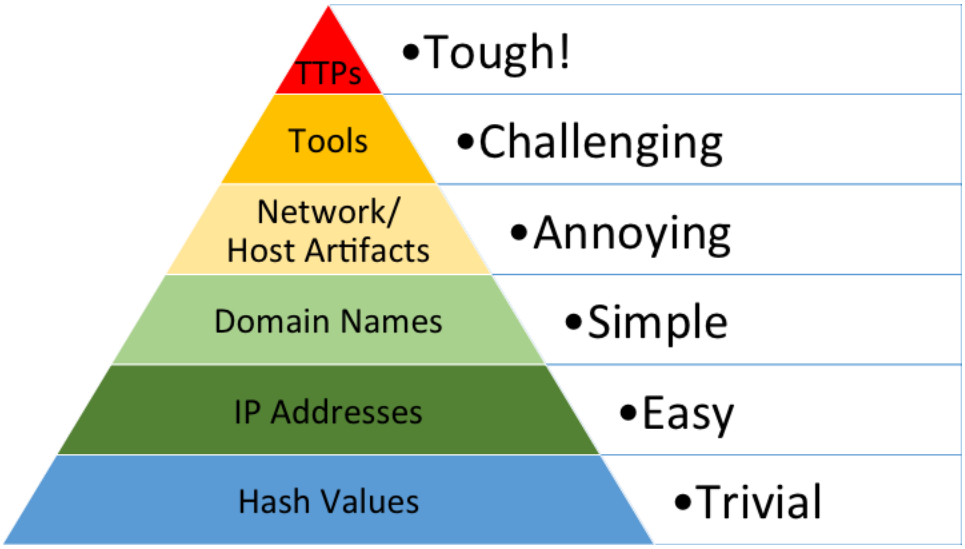


FIGURE 2.1 Pyramid of pain par David J Bianco [1]

Il est également intéressant de noter qu'utiliser le cadriciel MITRE ATT&CK rejoint les indications données par Warikoo A. [19]. Il explique que dans le profilage des criminels, deux hypothèses sont utilisées. D'une part, un criminel démontrera des comportements similaires dans tous ces crimes. D'autre part, des malveillances similaires doivent être associées à des profils criminels similaires. Pour garder ces cohérences, les criminologues se basent d'une part sur une base de données prédéfinies et d'autre part sur les données observées. Cela peut donc s'appliquer également à l'attribution des cyberattaquants en utilisant MITRE ATT&CK comme base de données prédéfinie.

Dans le contexte de l'attribution, un premier axe de recherche vise à utiliser ce cadriciel sur des rapports CTI (cyber threat intelligence), afin de découvrir l'entité à l'origine d'une attaque après une première investigation de l'attaque. Elle vise à collecter un grand nombre de rapports qui seront étiquetés grâce aux informations présentes dans le cadriciel MITRE ATT&CK, puis d'utiliser des algorithmes de traitement de langue naturelle pour effectuer un apprentissage supervisé et pouvoir dans un second temps étiqueter de nouveaux rapports.

On retrouve les travaux de Noor U. et al. [20] qui identifient encore une fois le besoin de profiler les attaquants à partir d'IOC de haut niveau tels que les TTPs. Ils observent également un problème de structuration dans les rapports CTI, car toutes les compagnies n'utilisent pas les mêmes nomenclatures pour les techniques ni pour les groupes d'attaquants. Ils proposent alors d'utiliser un système sémantique pour extraire des TTPs communs à tous ces rapports

puis d'utiliser un algorithme d'apprentissage machine supervisé pour prédire le groupe d'attaquants concerné. MITRE ATT&CK est alors utilisé pour définir une base commune de TTPs et de groupes d'attaquants pour traiter les rapports CTI. Bien que leurs résultats soient convaincants, ils n'ont évalué leur modèle que dans le domaine des technologies financières. Il est donc nécessaire d'évaluer le modèle dans d'autres domaines afin d'estimer sa capacité de généralisation et d'obtenir des métriques plus représentatives de cas réels.

Kim H. et al. [21] proposent une technique similaire pour attribuer les rapports CTI. En plus du travail précédent, ils observent un manque de données d'entraînement et un déséquilibre des classes dans les TTPs, ce qui empêche les modèles de généraliser convenablement. Ils proposent alors SMOTE comme algorithme pour régler ce problème en augmentant les classes minoritaires de manière synthétique. Malheureusement, l'algorithme SMOTE ne permet pas d'étendre l'espace d'observation des données, car il fonctionne avec un principe de voisins proches. Les auteurs finissent donc par mentionner le problème de généralisation comme la principale limite de leur recherche.

Finalement, on retrouve aussi NO-DOUBT, proposé par Perry L. et al. [22]. Cet algorithme projette les rapports CTI sur un espace vectoriel, qui permet ensuite une classification avec une forêt d'arbres décisionnels en utilisant l'algorithme XGBoost, qui permet en théorie une bonne généralisation. Une fois encore, ils étiquettent les rapports collectés avec les informations du cadriciel MITRE ATT&CK et évaluent ensuite leur modèle sur ces mêmes rapports étiquetés. Il est difficile de prédire correctement les performances de ce travail sur des données réelles. En fin de compte, cet axe de recherche semble toujours présenter le problème suivant : il est donc difficile de mesurer l'utilité de ces résultats, d'autant plus qu'un travail d'investigation reste nécessaire pour écrire les rapports CTI après incidents, qui sont les données d'entrée de tous ces modèles.

On observe donc un autre axe de recherche qui vise à utiliser les informations de tactiques et techniques fournies par le cadriciel MITRE ATT&CK pour représenter les attaquants en fonction de leurs actions. À partir des commandes ou des logiciels malveillants utilisés par ces attaquants, on obtient une représentation sous forme de vecteurs indiquant quelles techniques et tactiques ont été utilisées. Dans un second temps, les informations concernant les groupes d'attaquants sont croisées avec les vecteurs obtenus pour trouver des liens entre les attaquants observés et les modes opératoires définis dans le cadriciel MITRE ATT&CK. La différence avec le premier axe de recherche est que celui-ci est plus à même d'être utilisé en temps réel et cherche à faire abstraction du travail d'investigation nécessaire à la création de rapports CTI.

On retrouve par exemple le travail de Kyoungmin K. et al. [23] qui utilisent directement

les logiciels malveillants afin de trouver le groupe d'attaquants à leur origine. Ils utilisent des analyseurs de logiciels automatiques dans des bacs à sable tels que Joe Sandbox [24], Intezer [25] ou Cuckoo sandbox [26] afin d'obtenir les TTPs de MITRE ATT&CK utilisés. Pour chaque logiciel, ils obtiennent alors un vecteur de TTPs utilisés et peuvent calculer une similarité entre les logiciels malveillants et les groupes d'attaquants en termes de TTPs utilisés. Ils permettent également aux utilisateurs de leur modèle de fournir d'autres informations pour influencer les calculs de similarités. Ces informations reflètent l'intelligence qu'un utilisateur aurait a priori sur les groupes d'attaquants et les logiciels malveillants.

Lim C. et al. utilisent le cadriciel MITRE ATT&CK pour analyser les commandes d'attaquants directement, récoltées sur des Honeypots. Dans [27], ils proposent un algorithme manuel pour étiqueter les commandes des attaquants en techniques du cadriciel. Cet algorithme est assez simple, il suffit d'aller chercher manuellement la technique qui s'associe le mieux à la commande et d'étiqueter de la même façon toutes les commandes similaires. En le faisant itérativement sur toute nouvelle commande, on obtient une liste qui permet d'obtenir un vecteur de TTPs à partir des commandes des attaquants. Ensuite, dans [28], ils utilisent leur liste ainsi dérivée pour analyser les TTPs les plus fréquentes observées sur leurs Honeypots.

Ces travaux représentent une étape importante dans le domaine, car ils introduisent l'utilisation du cadriciel MITRE ATT&CK pour attribuer les attaquants. Cependant, les travaux de Lim C. et al. ne résolvent pas vraiment la problématique d'attribution, car ils s'arrêtent à une étude sommaire des techniques. D'autre part, le travail fait par Kyougmin K. et al. ne prend en compte que des logiciels malveillants visant les appareils mobiles. De plus, ces logiciels sont traités individuellement et aucune corrélation n'est faite entre les attaquants. En conclusion, on trouve encore beaucoup de place pour des travaux ultérieurs dans ce domaine.

2.2 Analyse de données Honeypot

La principale problématique lorsque l'on étudie l'attribution est le manque de données. En effet, à ce jour, aucun jeu de donnée présentant des informations sur les attaquants n'est disponible. Il n'est donc pas possible de traiter le problème de l'attribution de manière supervisée. Ce problème est identifié par Bada M. et al. dans [29] et Doynikova E. et al. dans [13], qui proposent soit de forger des attaques et d'ajouter des données d'attaquants intentionnellement, avec un générateur d'attaque, soit d'utiliser des données d'Honeypot. Dans ce travail, nous avons une entente avec une compagnie qui maintient une solution d'Honeypots et qui nous fournit ces données, à des fins d'analyses. Nous nous trouvons donc dans le deuxième cas mentionné ci-dessus.

Les Honeypots sont des serveurs dont le seul but est de récolter des informations sur les attaquants. Ils n'ont aucune fonctionnalité de production et ne devraient donc, en théorie, jamais être sollicités. Cependant, on observe que tout serveur mis à disposition sur internet finit tôt ou tard par être la cible d'attaques. Ces serveurs récoltent alors les commandes envoyées par les attaquants et, dans certains cas, les logiciels malveillants téléchargés et envoient tout cela à une unité centrale. Dans ce travail, nous avons uniquement accès aux commandes utilisées par les attaquants, qui constituent les données d'entrée de tous les algorithmes.

D'après Mokube I. et Adams M. [30], on peut distinguer trois niveaux de Honeypots selon leurs degrés d'interaction. Les Honeypots à interaction faible fournissent très peu de services aux attaquants. Cela minimise le risque que l'attaquant prenne le contrôle du Honeypot, mais permet plus facilement à un attaquant expérimenté de comprendre qu'il a affaire à un Honeypot. Ils sont principalement utilisés pour analyser les spams ou découvrir de nouveaux vers informatiques [31]. Les Honeypots à interaction moyenne fournissent plus de services, mais ne permettent toujours pas d'interagir avec un système d'exploitation complet. Ils veulent fournir une bonne illusion aux attaquants, tout en assurant un risque limité que le serveur se retrouve effectivement compromis [32]. Ce sont ces Honeypots que l'on retrouve le plus souvent sur internet, et c'est de ce type d'Honeypots que nous avons obtenu des données. Enfin, les Honeypots à interaction élevée permettent aux attaquants d'interagir avec la totalité des services d'un système d'exploitation, rien n'est simulé. Ils permettent la meilleure immersion pour les attaquants, mais présentent également le plus gros risque d'être compromis. L'avantage est que ces Honeypots permettent d'étudier le vecteur d'attaque complet et d'obtenir des données aussi réalistes que possible [33].

Il y a trois principaux axes de recherche portant sur les données Honeypots : l'amélioration des systèmes de détection d'intrusion (IDS) qui utilisent les données d'attaques pour dériver de nouvelles signatures ou de nouveaux algorithmes non supervisés de détection d'attaques, l'analyse de logiciels malveillants qui les étudie afin de créer des liens et de découvrir leurs capacités d'adaptation et l'analyse de comportements adverses qui attribue les attaquants dans différentes classes. Notre travail se situe principalement au niveau de ce dernier axe de recherche.

Amélioration des IDS

Moore C. et al. [34] font une analyse sommaire des champs trouvés fréquemment dans les Honeypots (comme les adresses IP, les ports, les protocoles utilisés), mais les considèrent tous individuellement. À partir des adresses IP, ils analysent que la Chine et le Japon sont à

l'origine de la majorité des attaques. Ils analysent également que 70% des attaques ciblent le port 5060 et donc le protocole SIP. Enfin, ils identifient des combinaisons de noms d'utilisateur et mots de passe fréquemment utilisés par les attaquants ainsi que des IPs publiques qui appartiendraient aux attaquants. Tous ces éléments pourraient être utilisés comme IOC dans des signatures pour bannir des attaquants d'un réseau.

Similairement, Sagala A. [35] propose de générer des règles d'IDS en se basant sur les commandes des attaquants. Spécifiquement, pour chaque nouveau fichier téléchargé par l'attaquant ainsi que chaque fichier existant sur la machine auquel l'attaquant tente d'accéder, une nouvelle règle est créée. da Costa Ferreira PHM. et al. [36] ainsi que Jiang CB. et al. [37] vont un peu plus loin dans leur recherche en introduisant des arbres de décision et de l'exploration de données respectivement. Les arbres de décision sont utilisés comme structure de données pour représenter les actions des attaquants et une règle correspond à un chemin entre le nœud racine et les nœuds feuilles. L'exploration de données est utilisée pour enrichir une base de règles avec des parties de commandes observées fréquemment sur un Honeypot. Pour cette exploration de données, les auteurs comparent les algorithmes Apriori et FP-Growth, en concluant que le dernier produit de meilleurs résultats.

Enfin, Owezarski P. présente UNADA [38], un nouvel algorithme non supervisé pour détecter et classifier les attaques. Cet algorithme regroupe les données en se basant sur 9 caractéristiques d'en-têtes des paquets et une corrélation des adresses IP et de l'horodatage des paquets. L'auteur propose ensuite de créer des signatures en analysant les différents regroupements identifiés par son algorithme.

Analyse de logiciels malveillants

Lingenfelter B. et al. [39] récoltent des logiciels malveillants à l'aide d'un Honeypot à interaction moyenne. Ensuite, ils les analysent manuellement afin d'étudier les similarités et variations que l'on peut retrouver dans ces logiciels. Ils concluent que Mirai, un logiciel malveillant datant de 2016 qui transforme les machines infectées en "zombies" [40], est encore actif et s'est même développé en plusieurs variantes en 2020. Deux ans plus tard, Zhu Y. et al. [41] confirment cette déclaration. À l'aide d'un Honeypot à interaction forte, ils récoltent les données échangées avec le serveur de commande et contrôle (C&C) et appliquent un algorithme de regroupement sur ces commandes directement. Ils identifient manuellement le serveur C&C de Mirai et observent que les autres serveurs C&C se comportent de façon homologue.

Analyse des comportements adverses

Bar A. et al. [42] étudient la capacité des attaquants à se propager sur un réseau. Ils collectent des données d'une multitude d'Honeypot connectés et modélisent leur études à l'aide des modèles de Markov cachés (HMM) en définissant comme état les attaques réalisées sur les Honeypots et en modélisant les probabilités de transition par les occurrences observées dans leur réseau. Pour ce faire, ils considèrent que si deux données proviennent de la même adresse IP en moins d'une heure, elles correspondent au même attaquant. Cela leur permet également de définir des communautés, groupes de serveur qui sont généralement attaqués les uns à la suite des autres. Similairement, Mehta S. et al. [43] étudient les objectifs des attaquants lors de leur traversée des systèmes de fichiers. Ils parviennent ainsi à prédire les fichiers auxquels les attaquants accéderont prochainement, en sachant quels fichiers ont été cherchés auparavant. Tout cela est fait en comptant les occurrences observées dans les données enregistrées dans leurs Honeypots. Cependant, étant donné que ces deux techniques dépendent des occurrences observées, elles ne sont pas très robustes et ne présentent pas une bonne capacité de généralisation.

Fraunholz D. et al. introduisent GAMFIS [44], un nouveau modèle d'attaquant basé sur une analyse statistique des données récoltées par des Honeypots. Dans ce modèle, ils définissent 9 types d'attaquants à partir de leurs motivations, ressources, intentions et compétences. Ensuite, dans [45], les mêmes auteurs expliquent comment utiliser ce modèle sur des données récoltées d'un Honeypot à interaction moyenne. Ils définissent alors quatre équations, une pour chaque trait, à partir de caractéristique que l'on retrouve directement dans les données. Même s'ils mentionnent que leur modèle fonctionne pour tout type de protocole, leurs équations sont spécifiques aux deux seuls protocoles considérés dans leur travail. Dans notre article, que l'on retrouve à la section 4, nous généralisons leur approche à un plus grand nombre de protocoles, aussi bien utilisés dans les technologies de l'information (IT) que dans les technologies opérationnelles (OT).

2.3 Attribution des cyberattaquants

L'importance de l'attribution est expliquée par Buchanan B. et Rid T. [46] par le fait que toute réponse à une offense requiert dans un premier temps d'identifier l'offenseur, peu importe la nature de cette offense. Ces auteurs identifient que la qualité de la tâche d'attribution dépend énormément de l'expérience de ceux qui en sont responsables. Ils mentionnent également que c'est une tâche nuancée, ce qui la rend très compliquée et qu'elle peut dépendre d'enjeux politiques. Toutes ces raisons nous poussent à rechercher des méthodes qui per-

mettront d'automatiser ce processus sans être sujet aux enjeux politiques ou aux erreurs humaines dues au manque d'expérience.

On peut identifier 3 principaux axes de recherches lorsque l'on parle de l'attribution [13]. Première, on retrouve la modélisation de graphes d'attaques, ensemble de nœuds liés représentant les actions potentielles d'un attaquant. Ensuite, on retrouve les modèles de Markov cachés (HMM) qui utilisent des états du système et des probabilités de transitions entre ces états. Enfin, on retrouve les modèles basés sur des règles, qui tirent profit d'une base de connaissances modélisant l'état actuel du monde.

Graphes d'attaques

Kotenko I. et al. [47] présentent CAMIAC (Cyber Attack Modeling and Impact Assessment Component), une approche permettant de construire un graphe d'attaque en temps quasi réel. CAMIAC fonctionne en deux phases, la première génère un graphe général à partir de la configuration du réseau et se fait hors ligne, lorsque le temps n'est pas une contrainte. La deuxième se fait en observant les chemins pris par les attaquants et modifie le graphe défini lors de la première phase. En suivant ainsi les attaquants à travers le graphe, on obtient différents profils d'attaquants. Une étude suivante de cet auteur [48] utilise la théorie bayésienne pour représenter des profils incertains. Le graphe d'attaque se voit transformé en réseau bayésien et les chemins reliés à un même profil se voient attribuer une probabilité.

GhasemiGol M. et al. [49] utilisent les graphes d'attaques afin de prédire les actions futures des attaquants. Ils introduisent un graphe d'attaque conscient d'incertitudes afin de modéliser l'incertitude des attaques. Leurs graphes d'attaques sont construits directement à partir d'alertes IDS et peuvent muter dynamiquement à mesure que de nouvelles alertes sont levées. Leur modélisation permet d'identifier quels serveurs sont les plus susceptibles d'être attaqués les uns après les autres.

Le gros problème des graphes d'attaques est qu'il faut une connaissance experte du système que l'on souhaite modéliser afin de correctement définir les différents chemins et probabilités de chemins. Tout cela rend leur utilisation difficile dans des systèmes plus conséquents.

Modèles de Markov Cachés

Katipally R. et al. [50] utilisent des HMM simples à 5 états correspondants aux actions de scan du réseau, d'énumération du réseau, d'exploitation de déni de service (DOS), d'exploitation de logiciel malveillant et de tentative d'exploitation d'accès. Ils créent également une liste de 65 règles qui leur permettent de transformer des alertes IDS en un format conforme

avec les états de leurs HMM. Ils calculent ensuite les probabilités de transitions d'état en observant les occurrences dans leurs données. Ils sont alors capables de prédire l'ordre dans lequel les phases sont les plus susceptibles d'être utilisées. Par exemple, ils montrent que la phase d'énumération est la plus susceptible d'être utilisée après la phase de scan avec une probabilité de 0.789.

Rashid T. et al. [51] utilisent les HMM en se concentrant uniquement sur la détection de menace interne, et donc un seul type d'attaquant au niveau de l'attribution. Du jeu de données de menace interne CERT [52], ils extraient les données utilisateurs telles que les sites web accédés ou les noms d'utilisateur. Ils construisent alors un profil normal à partir des 5 premières semaines de données, posant ainsi l'hypothèse discutable qu'aucune attaque ne se passe pendant ces 5 semaines. Les états du HMM sont alors les actions réalisées par ce profil normal et les probabilités de transitions sont les occurrences observées lors de ces 5 semaines. Lorsque le profil normal est défini, une anomalie est détectée quand une séquence peu probable est observée. Cela demande donc également un seuil pour séparer les séquences peu probables des autres.

Deshmukh S. et al. [53] introduisent un algorithme d'apprentissage machine semi-supervisé dérivé des HMM appelé Fusion Hidden Markov Model (FHMM). Ils partitionnent les données en K groupes non corrélés et entraînent K HMM, un par groupe. Ils combinent ensuite les prédictions des K HMM à l'aide d'un réseau neuronal qui a pour tâche de prédire la décision finale. Pour chacun de leurs HMM ils considèrent alors 19 états correspondant à 19 commandes observées des attaquants. Leur travail est cependant limité par l'hypothèse qu'il est possible de séparer les données en K sous-ensembles non corrélés et les auteurs mentionnent que leurs performances peuvent subir de graves dégradations dans le cas contraire.

Le problème principal des HMM est qu'ils sont par définition fort dépendants du jeu de données sur lequel ils ont été entraînés. Autant les états considérés que les probabilités de transitions entre les états sont difficilement extensibles à d'autres données sans modifications majeures. Il en revient alors à traiter le même problème en repartant presque de zéro à chaque changement de configuration réseau, ce qui arrive souvent en pratique.

Modèles basés sur des règles

Mallikarjunan KN. et al. [54] utilisent la logique floue et des règles floues dans leur attribution. À partir d'alertes de différents IDS, ils définissent une "méta-alerte" pour chaque alerte, qui est un regroupement des informations de chacune des autres alertes dans un format standardisé. Ils utilisent ensuite une liste de règles floues pour calculer des appartenances à chacun de 9 groupes d'attaquants considérés. Ils choisissent alors le ou les groupes qui

s'apparentent le plus à cet attaquant comme décision finale. Ils mentionnent cependant eux-mêmes la limite de se baser sur des alertes d'IDS, on ne peut pas détecter de nouvelles attaques, aussi appelées attaques "zero-day".

Nunes E. et al. [55, 56] utilisent de la logique argumentative pour représenter les intuitions d'un analyste. Ils testent leur méthode sur le jeu de données collecté au CTF DEFCON de 2013 et un ensemble de règles créées manuellement. Ces règles se basent sur les observations qu'ils font à partir du jeu de données, telles que "l'équipe X a été visée par l'exploit 1" ou "l'équipe Y a rejoué l'exploit 2". Ils introduisent aussi des règles révocables qui peuvent être vraies si aucune information contradictoire n'est trouvée. Leurs résultats ne sont pourtant pas très convaincants, avec une précision moyenne de 62%. Cela serait dû aux règles révocables qui induisent trop de confusion dans leur algorithme et qui, dans beaucoup de cas, laissent un ensemble vide de coupables potentiels.

Similairement, Karafili E. et al. [57–59] utilisent la logique argumentative pour représenter l'état du monde actuel. Les règles prennent alors la forme d'indications des conflits entre pays, les capacités économiques des groupes considérés ou même à qui appartient un serveur spécifique. En faisant le lien entre cette base de connaissances sur le monde actuel et les observations dérivées des attaques commises, ils donnent aux analystes un assistant qui les dirige dans leur tâche d'attribution. Ils mentionnent également les limites de leur approche par rapport aux attaques nouvelles.

Le problème de l'utilisation de règles spécifiques tel que fait dans ces travaux est que les résultats d'attribution vont être très dépendants des règles et donc du jeu de donnée considéré. En effet, les règles dérivées sont toujours spécifiques aux contextes étudiés et sont difficilement extensibles. Aussi, une règle peut vite devenir obsolète ou même induire l'attribution dans de mauvaises directions si l'état actuel du monde a changé.

CHAPITRE 3 DÉMARCHE DU TRAVAIL DE RECHERCHE

Le sujet de ce mémoire est l'attribution des cyber-attaquants. Nous avons décidé de séparer ce sujet en deux objectifs, comme expliqué à la section 1.3 de ce mémoire. Le premier objectif de recherche était de proposer un ordonnancement des évènements observés sur nos Honeypots afin d'identifier facilement les attaquants les plus dangereux. Pour ce faire, nous avons identifié les sous-objectifs suivants :

1. Définir des caractéristiques pour représenter les attaquants
2. Identifier les attaquants aux valeurs aberrantes pour les caractéristiques choisies
3. Identifier différents groupes d'attaquants en utilisant des algorithmes d'apprentissage machine non supervisé

Dans la Section 3.1, nous expliquons plus en détail en quoi notre article répond à cet objectif de recherche et donnons le lien entre les différentes parties de l'article et les sous-objectifs.

D'autre part, nous avons également annoncé que le deuxième objectif de recherche était de corréler les attaquants entre eux et d'identifier des modes opératoires associés. Nous avons alors identifié les sous-objectifs suivants :

1. Définir les attaquants en fonction des techniques utilisées par rapport au cadriciel MITRE ATT&CK
2. Définir différents modes opératoires par rapport au cadriciel MITRE ATT&CK
3. Corréler individuellement les attaquants et les modes opératoires obtenus
4. Définir un algorithme afin d'utiliser l'information des modes opératoires pour corréler les attaquants entre eux

De la même façon, dans la Section 3.2, nous expliquons en quoi nos résultats complémentaires répondent à cet objectif en détaillant chaque sous-objectif.

L'ordonnancement des attaquants permet d'identifier les attaquants nécessitant un travail d'attribution plus conséquent. Ces attaquants pourront alors ensuite être analysés plus amplement à l'aide de l'algorithme défini pour répondre au deuxième objectif. Cet algorithme permet alors d'obtenir des pistes d'attribution des cyber-attaquants. Ensemble, ces deux travaux permettent donc de répondre au sujet principal du mémoire.

3.1 Analyse statistique des attaquants

Dans le chapitre antérieur, nous avons montré que certains travaux se basent sur des données Honeypots afin de caractériser les comportements adverses. En particulier, Fraunholz D. et al. [44,45] introduisent un modèle d'attaquant basé sur 4 caractéristiques définies pour chaque attaquant : motivation, ressources, capacité et intention. Ils effectuent alors une analyse statistique de chaque caractéristique pour chaque attaquant, ce qui permet d'identifier les attaquants aux valeurs aberrantes. Aussi, ils définissent, de manière très empirique, 9 groupes d'attaquants à partir de régions de l'espace défini par ces caractéristiques. Cependant, même s'ils mentionnent être capables d'appliquer leur modèle à tous types de protocoles, il n'est utilisé que sur des données du Honeypot Cowrie. De plus, leurs définitions de caractéristiques restent spécifiques au protocole considéré.

Le but de cet article est donc de tester et généraliser leur méthode sur différents protocoles, aussi bien IT que OT. Aussi, nous proposons d'utiliser un algorithme d'apprentissage machine non supervisé afin d'identifier les différents groupes d'attaquants, plutôt que de les forcer de manière empirique.

La section 4.3 explique dans un premier temps la collecte des données et la répartition des données obtenues en fonction du protocole. Ensuite, nous expliquons la méthodologie de Fraunholz D. et al. que nous avons utilisée afin de répliquer et valider leurs résultats. Enfin, nous expliquons notre méthodologie généralisée, et redéfinissons de nouvelles caractéristiques pour chaque attaquant : motivation, ressources, intention, furtivité et originalité. Aussi, nous expliquons avoir ajouté une contrainte en plus à notre travail. En effet, dans leur travail initial, Fraunholz D. et al. définissent, en pratique, les attaquants par leurs adresses IP. Cependant, actuellement, à cause de la pénurie d'adresses IPv4, nous savons qu'une même adresse peut être réutilisée par une autre personne, passé un certain délai. Dès lors, nous considérons une fenêtre de temps de 1 heure lors de notre définition des attaquants. C'est-à-dire que si deux évènements proviennent de la même adresse IP en moins d'1 heure, alors ils sont considérés comme appartenant au même attaquant. Dans le cas contraire, un autre attaquant est considéré. Tout cela permet de définir un attaquant en fonction des 5 caractéristiques mentionnées précédemment et répond au premier sous-objectif identifié.

Avec ces représentations d'attaquants, nous avons dans un premier temps tenté de répliquer les résultats de Fraunholz D. et al. , ce que l'on retrouve dans la Section 4.4.1. Nous observons que la caractéristique de motivation reste similaire, si ce n'est un plus grand nombre d'attaquants ayant une faible motivation. Cela est dû à la contrainte de temps que nous avons ajoutée comme expliqué ci-dessus. Cependant, nous remarquons que les caractéristiques de

ressources et capacités sont assez différentes. En effet, les équations qu'ils ont utilisées pour définir ces caractéristiques sont propres au protocole considérés et difficilement généralisables. Enfin, bien qu'ils mentionnent considérer l'intention en théorie, ils expliquent ne pas avoir trouvé de manière de la définir en pratique. Ce sont ces raisons qui nous ont poussés à redéfinir des caractéristiques dans notre approche généralisée, présentée à la Section 4.4.2.

Dans cette Section, nous présentons, séquentiellement chacune des caractéristiques considérées ainsi que leurs définitions. Une analyse de chaque attaquant à partir de ces caractéristiques permet de trouver statistiquement les valeurs aberrantes, répondant au deuxième sous-objectif de recherche. Par exemple, à partir de la caractéristique de motivation, nous sommes capables d'identifier que certains attaquants ont ciblé jusqu'à 7 protocoles différents, alors que la plupart s'attaquent à 1 ou 2 protocoles. Une fois l'analyse statistique individuelle de chaque caractéristique effectuée et les valeurs aberrantes écartées, nous avons utilisé l'algorithme Kmeans afin de trouver différents groupements d'attaquants. Différentes métriques telles que le score Silhouette, le score Davies Bouldin et le score d'inertie nous ont permis de conclure que 13 groupes semblaient donner le meilleur regroupement. Enfin, une analyse manuelle de certains attaquants de chaque groupe nous ont permis de conclure que plus de 90% des attaquants se situent dans 3 groupes et sont relativement inoffensifs. Enfin, le dernier groupe semble contenir les 62 attaquants les plus dangereux observés sur nos données Honeypots. Cette analyse via Kmeans nous permet de répondre au troisième sous-objectif de recherche.

Enfin, l'analyse statistique et l'analyse par regroupement nous permettent de répondre à l'objectif de recherche considéré. En effet, les attaquants aux valeurs aberrantes identifiés ainsi que le dernier groupe de 62 attaquants représentent ceux qui devraient être analysés directement. Les trois groupes contenant plus de 90% des attaquants représentent les plus inoffensifs attaquants observés et devraient être analysés en dernier. Le reste des attaquants représentent alors une menace intermédiaire.

3.2 Classification et corrélation des attaquants

Lors de la revue de littérature, l'utilisation du cadriciel MITRE ATT&CK pour la traque de menace, et par conséquent l'attribution, a été abordée. Dans les résultats complémentaires présenté au Chapitre 5, nous avons utilisé les informations de ce cadriciel pour définir d'une part les attaquants et les modes opératoires des attaquants. En effet, nous utilisons ici une représentation différente des attaquants. Nous dérivons une liste, similaire à ce qui a été fait par Lim C. et al. [27] afin d'obtenir une représentation des attaquants en fonction des techniques qu'ils utilisent. D'autre part, nous utilisons les informations présentes dans le

cadriciel à propos des différents groupes de menaces connus et les techniques qu'ils utilisent habituellement afin de définir les modes opératoires.

La Section 5.3.1 explique plus en détail la dérivation de la liste de règles permettant de faire le lien entre les commandes des attaquants et les techniques utilisées. Celle-ci nous permet alors d'obtenir, pour chaque attaquant, un vecteur binaire des techniques utilisées. Ensuite, pour chaque groupe de menace (et donc mode opératoire) considéré, nous obtenons des vecteurs binaires, de la même façon. Cela nous permet de répondre aux deux premiers sous-objectifs de recherche.

Ensuite, dans la Section 5.4.2, nous utilisons un algorithme de regroupement pour trouver des similitudes entre les attaquants. Nous avons opté ici pour l'algorithme DBScan. La raison de ce choix est qu'il y a beaucoup de bruits dans notre représentation d'attaquants, qui se situent dans un espace à 27 dimensions. Un algorithme basé sur des prototypes comme Kmeans sera alors sujet à ces bruits et les regroupera seul dans des clusters. Il est plus simple d'utiliser un algorithme basé sur des densités conçu spécialement pour des données clairsemées ("sparse"), tel que DBScan. Cela nous permet alors d'identifier, pour chaque groupe d'attaquants, le mode opératoire le plus similaire. Cela répond alors au troisième sous-objectif de recherche. Cependant, analyser les attaquants individuellement ne semble pas suffisant, car pour certains, la similarité n'est que très faible, de l'ordre de 14% pour le groupe d'attaquants n'ayant utilisé que très peu de techniques.

Enfin, le dernier sous-objectif de recherche est abordé dans la Section 5.4.3. Nous dérivons alors un nouvel algorithme permettant d'utiliser l'information recueillie sur les différents modes opératoires afin de corréler les attaquants entre eux. Étant donné que nous souhaitons pouvoir généraliser notre algorithme à différents cas, nous introduisons également des paramètres afin de considérer un potentiel trafic légitime. L'algorithme n'identifie alors que les groupes d'attaquants qui sont suffisamment suspects, dont le vecteur d'attaque observé se rapproche suffisamment de l'un des modes opératoires considérés. Tout cela étant alors modulé par les paramètres introduits. Dans la Section 5.5, nous discutons également ces paramètres introduits et donnons des directives générales sur la manière de les fixer. C'est également cet algorithme qui permet de répondre au deuxième objectif de recherche considéré.

CHAPITRE 4 ARTICLE 1 : ATTACKER ATTRIBUTION VIA CHARACTERISTICS INFERENCE USING HONEYPOT DATA

authors : Pierre Crochelet, Christopher Neal, Nora Boulahia Cuppens, Frédéric Cuppens

Polytechnique Montreal, Montreal, Canada

{pierre.crochelet,christopher.neal,nora.boulahia-cuppens,frederic.cuppens}polymtl.ca

Presented at the 16th International Conference on Network and System Security (NSS), December 7, 2022

abstract. Increasingly, the computer networks supporting the operations of organizations face a higher quantity and sophistication of cyber-incidents. Due to the evolving complexity of these attacks, detection alone is not enough and there is a need for automatic attacker attribution. This task is currently done by network administrators, making it slow, costly and prone to human error. Previous works in the field mostly profile attackers based on external tools or lists of rules that need to be updated regularly. Some tackle this problem through particular methodologies that cannot be easily generalized to any data source. We focus on using a self-sufficient technique that allows us to characterize attackers through motivation, resourcefulness, stealth, intention and originality. Furthermore, we show that this technique can easily be used on several protocols by applying it to a dataset consisting of real attacks performed on several honeypots. We show that more than 90% of the recorded data is relatively harmless and only a limited number of attackers are alarming. This process enables network administrators to readily discard benign traffic and focus their attention towards high-priority attacks.

Keywords : Attacker Attribution · Information Security · Honeypot Data · Intelligent Data Analysis

4.1 Introduction

In today's world, cybersecurity has an increasing importance as cyberattacks are more and more frequent [60]. Initially, many researchers focused on the detection of these incidents through the use of Intrusion Detection System (IDS) platforms that attempt to capture the essence of the attack itself [61,62]. However, with continual advances in technology, new forms of attacks appear every day, creating a need to regularly update an IDS. Therefore, detection

alone is no longer sufficient and an IDS must consider the attribution problem as well [5]. Attribution, sometimes referred to as "attacker profiling", is the process of characterizing the attacker instead of the attack. Currently, attribution tasks are mostly done by network administrators, thus the quality of attribution will vary. There is a great need to automate, at least some, part of this process with an unbiased tool to increase the reliability and speed of this task.

Fraunholz et al. [45] propose attributes to perform attacker profiling through the use of logs collected from a honeypot. Honeypots are specific decoy servers on a network that aim to lure attackers and record their attacks [30]. As honeypots do not have any production functionality, we know that all recorded traffic on honeypots is malicious or at least suspicious. The process proposed by Fraunholz et al. is an important step in the field of attacker profiling, however, it lacks a generalization as it focuses on only one type of log from one specific honeypot.

We propose a novel method to characterize attackers from logs of various services collected by a series of low-interaction honeypots and a way to identify the actual threats from harmless attackers. As such, we define new attributes that can be induced for multiple protocols through features found in many of them. We also show that this method works not only for Information Technology (IT) protocols but also for Operational Technology (OT) ones as well. The contributions of this work are summarized as follows :

1. Validate Fraunholz et al. 's approach on a new dataset.
2. Add a time constraint when considering attacks to reflect the dynamic assignment of IP addresses.
3. Expand and improve Fraunholz et al. 's approach to consider other IT and OT protocols.
4. Propose another way to analyse the results through clustering.

The remainder of this work is organized as follows. In Section 4.2, we discuss recent developments in attacker attribution. Section 4.3 introduces the methodology in [45] and presents our proposed generalized methodology. In Section 4.4 we apply both of these methodologies to logs of real attacks from multiple honeypots. Finally, Section 4.5 discusses the results and the limitations of our approach.

4.2 Background and Related Work

When considering the attribution problem, two challenges arise. First, attackers often use deceptive techniques to avoid detection [63]. Second, there is a lack of suitable datasets to

research the relationship between an attacker's actions and the required characteristics to build a profile. However, to circumvent this second problem, Doynikova et al. [13] propose, amongst other things, to collect real attacker data through the use of honeypots, which is the focus of our work.

Nevertheless, different ways to gain intelligence about attackers have been proposed. Indeed, Bar et al. [42] use Hidden Markov Models (HMMs) to study attacker propagation in a network of honeypots. They construct communities within the network, where the majority of attacks are within the communities, and do not cross the borders of those communities. However, they do not explicitly use attacker models to identify the profiles of attackers and therefore do not outline which attackers require a deeper analysis.

Mallikarjunan et al. [54] introduce a way to build attacker profiles from IDS alerts through fuzzy rules. First, low-level alerts are grouped to generate a meta-alert for each intruder. Then, fuzzy rules are proposed to find the broad category of an attacker, which is later refined into an exact attacker profile through other specific rules. Finally, a database of attacker profiles is searched to find a match for this attacker profile. This methodology is very dependent on the set of IDS alerts generated and therefore requires an up-to-date tool to be effective. Regardless, it will not be possible to consider attackers using zero-day exploits. Similarly, Karafili et al. [59] use argumentation-based reasoning to help forensic analysts with attribution tasks. On the one hand, they use a set of rules described as *background knowledge* that needs to be updated regularly. On the other hand, they use a set of core rules that aims to mimic the analysts' reasoning during the attribution tasks. Ultimately, these two methodologies are not self-sufficient since they require human-based assistance or a high-performing tool to generate interesting results.

Deshmukh et al. [53] use real attack logs gathered from the Cowrie honeypot as a basis for attribution [64]. The authors identify the need for unsupervised or semi-supervised learning methods and propose Fusion Hidden Markov Models (FHMMs) that fall in the latter category. FHMMs are more resistant to noise and profit from all the advantages associated with ensemble learning. However, for the FHMMs, they define 19 states in terms of data specific to the Cowrie logs, making their technique hard to generalize to other types of honeypot logs.

Similarly, Fraunholz et al. [44] introduce an attacker model, GAMfIS, to characterize attackers in four attributes : Motivation (M), Skill (S), Intention (I) and Resources (R). Using those attributes, they define a Threat Rating (TR) as M+I, a Capability Rating (CR) as S+R, and finally a Total Threat Score as TR+CR. Then in [45], the authors apply their attacker model to a series of logs collected from a Cowrie Honeypot. They conclude that most attacks are coming from botnet traffic that can be mapped to rather harmless attackers. However, [45]

only use logs from a Cowrie Honeypot that implements the SSH and Telnet protocols and focuses on a statistical analysis of their results. To do this, they use features that are specific to this honeypot. Our method builds on the work of [45] but focuses on features that can be extracted from various honeypots and services. Furthermore, after the statistical analysis, our method introduces a clustering analysis to differentiate harmless and malicious attacks.

4.3 Methodology

4.3.1 Data Collection and Processing

Our data has been collected for more than a year, starting in December 2020, through a series of honeypots. In total, we have more than two million log entries. The distribution of these logs is the following ; Redis 44.9%, SSH 34.4%, Telnet 5.1%, Modbus 5.0%, HTTP 4.5%, HTTP-proxy 3.7%, DNP3 1.4%, FTP 0.9%, and BACnet 0.1%. All of these logs resemble the ones obtained from a Cowrie Honeypot. They are in a "NDJSON" format and contain information about the attacker, such as the source Internet Protocol (IP) address and port, as well as about the attack, such as the protocol, the command typed, and when it was executed. From these logs, we extract several features to classify each attacker. We include all protocols but focus on the 6 that we can easily interpret and for which we have interesting data, namely : SSH, Telnet, Redis, FTP, Modbus, and HTTP. The extracted features are shown in Tableau 4.1.

Tableau 4.1 Extracted features for each protocol

	SSH/Telnet	Redis	FTP	Modbus	HTTP	Other protocols
Number of logs	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**
List of methods	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	No
Number of scans	No	**Yes**	**Yes**	**Yes**	No	No
Number of sessions initiated	**Yes**	No	**Yes**	No	No	No
Number of malware files	**Yes**	No	No	No	No	No
Frequently used entries	**Yes**	No	No	No	No	No

4.3.2 Basic Methodology

Fraunholz et al. [45] perform the attribution on the attacks, defined as all attack sessions sharing the same IP address. In practice, this means that they group together all logs that share the same source IP, not considering the time. Then, for each attack, they propose characterizing them with :

— **Motivation :** the amount of effort the attacker puts into the attack,
— **Skill :** the degree of expertise of the attacker,
— **Resources :** the degree of automation the attacker has access to, and
— **Intention :** the attacker's goals and severity of the attack.

All these attributes are obtained through a linear combination of specific features, while ensuring they fall in the $[0, 10]$ interval. Finally, they introduce two new metrics : the threat rating, defined as the sum of the motivation and intention attributes, and, the capability rating, defined as the sum of the resource and skill attributes. From these two final values falling in the range $[0, 20]$, they plot the results in a heat map to identify where most attackers lie, and, consider only the outliers as threats.

4.3.3 Generalized Methodology

In our proposed generalized methodology, we add a time constraint to the attacks when grouping the logs together. Indeed, because of the dynamic assignment of IP addresses, even if two logs come from the same IP address, we cannot always be sure that they come from the same attacker. For this, we decide to add a time constraint of one hour as done in [42,65]. This means that if two logs share the same IP address and the time interval between these two logs is less than an hour, they are considered to come from the same attacker.

Also, in [45], not all equations are explicitly shown, in that the weight distributions are not specified. Therefore, in this work we explicitly calculate the added value of a specific feature for a characteristic as $added\_value = \frac{(given\_value - mean)}{std}$. Where, "mean" and "std" (i.e. standard deviation) are calculated for each feature independently, and "given_value" represents the value of the considered attacker for that feature. Therefore, the added value represents how far, in terms of standard deviation, the given value is from the mean. It is similar to having a linear combination where each feature is weighted corresponding to its distribution.

Finally, we change the way to characterize the attackers to make it easier to generalize to other protocols. Indeed, as will be seen in Section 4.4.1, the attributes used in [45] are sometimes too specific to the Cowrie honeypot. Therefore, we characterize the attackers in terms of :

— **Motivation :** the amount of effort the attacker puts into the attack,
— **Resourcefulness :** the degree of expertise and automation of an attacker,
— **Stealth :** the amount of effort the attacker puts into not getting caught,
— **Intention :** the attacker's goals and severity of the attack, and
— **Originality :** to identify "script kiddies" from legitimate malefactors.

4.4 Evaluation

4.4.1 Replication of Results

In practice, Fraunholz et al. [45] do not give the exact equations from which they derived the values of each attribute. Indeed, they mention the features they used for each attribute, however, do not show the weight associated to each feature. Therefore, without having the specifics, we use the previous equation to find the added value of each feature for an attribute. Then, for comparison purposes, we scale the obtained attribute values to the corresponding range. Also, Fraunholz et al. [45] mention that they could not find a way to express the attackers' intentions. Therefore, to find the threat rating they consider the intention score to be the same as the motivation score, which is what we are doing for this replication as well.

Motivation Ratings

To calculate the motivation ratings, three features are used : the number of commands entered, the number of sessions initiated, and the time spent in all these sessions. These are easily reproducible with the data at hand, except for session-less protocols like Modbus or Redis. For these protocols, we approximate the time spent in all sessions as the total time of the attack and the number of sessions initiated is considered to be already accounted for by the number of commands entered. Figure 4.1 shows the results obtained without scaling the feature. Most attackers obtain a motivation score between 0 and 20, with a few exceptions obtaining a much higher score.

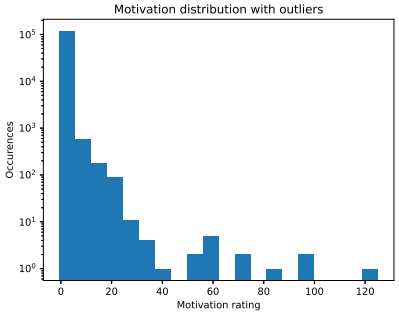
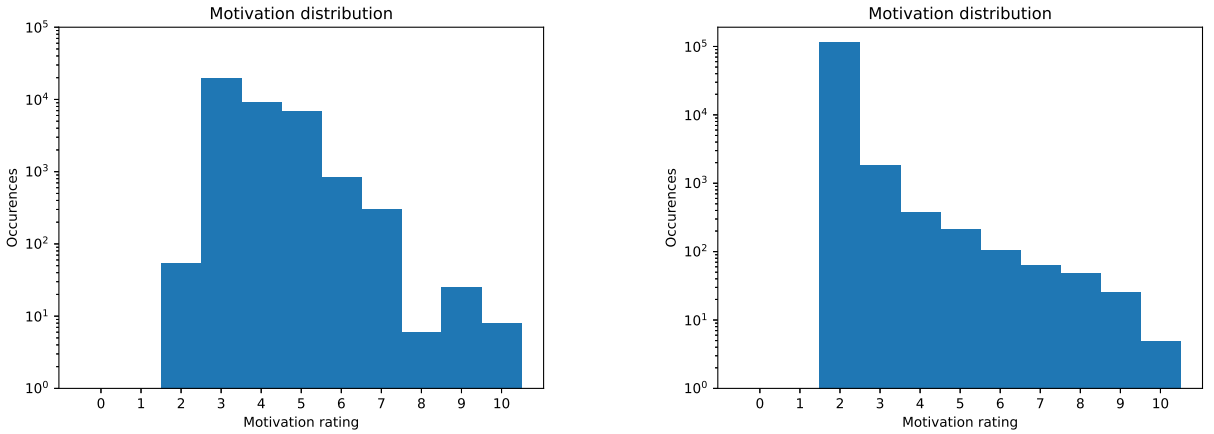


FIGURE 4.1 Motivation ratings with outliers

Looking back at the logs, we see two trends to explain those outliers. The first one is that one of the honeypots did not implement a time-to-live, and some attackers exploited that to maintain some sessions open for a up to four months. The motive isn't clear but the attackers' intent could be to cause a denial of service. Therefore the "time spent in all sessions" feature

is very high for those attackers, compared to the others. The second way to explain those outliers is through the "number of commands" feature. Indeed, where most attackers only sent between zero and a few hundreds of commands, some sent tens of thousands of commands and therefore, have a very high motivation score. To compare our results with the ones obtained in [45], we scale the values in the same range as they had. However, if we just scale the results as they are in Figure 4.1, the resulting distribution will be skewed towards those outliers. Therefore, to get a meaningful comparison, we first remove the outliers and then scale the values. These results are shown in Figure 4.2.



Motivation ratings presented in [45] Reproduction of Motivation calculation

FIGURE 4.2 Comparison of the Motivation ratings distributions

With our dataset, we obtain similar results for the motivation ratings. The major difference is that we see more attackers with a low motivation score. This is probably the result of adding the constraint on time when aggregating the logs into attacks, which gives more "small" attacks where the attackers simply send a few scans but do not act on them.

Resources Ratings

To calculate the resources ratings, only one feature can be used with the data we have. Indeed, in [45], Fraunholz et al. used the inter-arrival time of the attacker's commands as well the number of credentials tried when logging into the honeypot. However, in the honeypots we are using, the attackers do not need to log in, either because of the specification of the protocols (like Modbus) or because the login function is disabled in the honeypots. This leaves only the inter-arrival times as a suitable feature to reproduce their results. Figure 4.3 shows the results obtained without scaling the feature. Most attackers obtain a resources score between 0 and 2000, with a few exceptions.

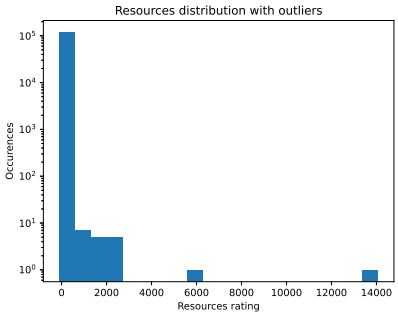
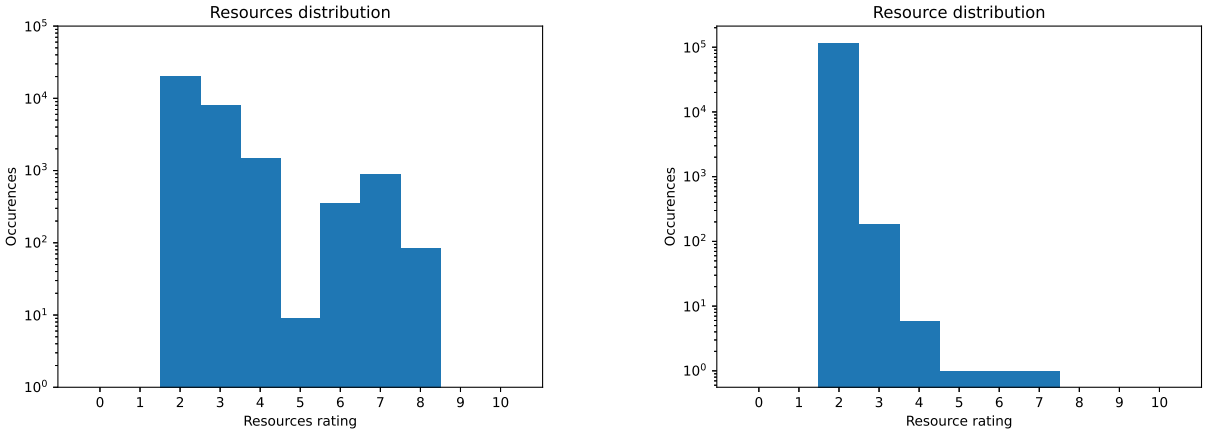


FIGURE 4.3 Resources ratings with outliers

Looking back at the logs, we see that the resources outliers correspond to attackers for which we have several observations on a small interval, usually having one new observation every 0.1 seconds. For the same reason as discussed above, we remove the outliers before scaling the values for the comparison, as shown in Figure 4.4. With our dataset, the results we get are a bit different, having very few attackers with a high resource rating. We presume this is because only one of the two features is usable with our data, which compromises the results.



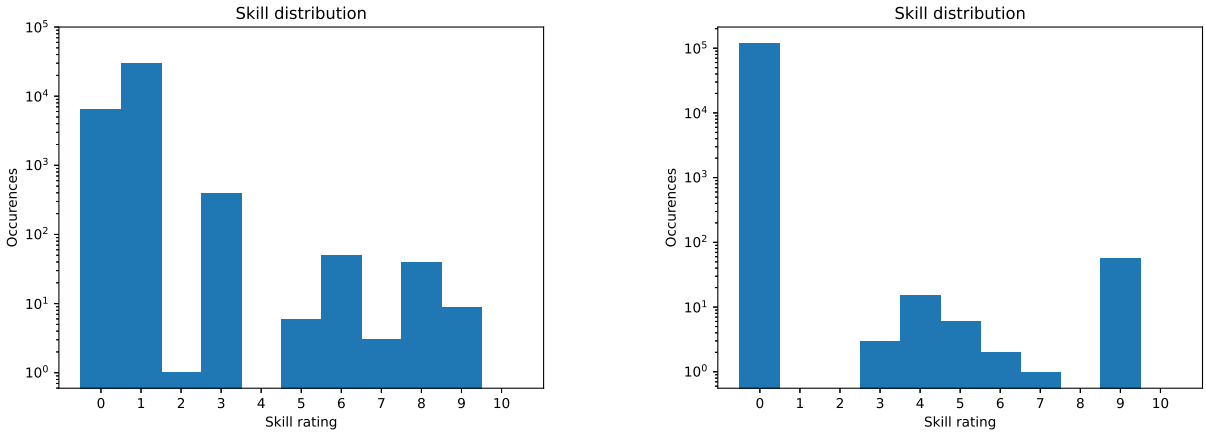
Resource ratings presented in [45] Reproduction of Resource calculation

FIGURE 4.4 Comparison of the Resources ratings distributions

Skill Ratings

The skill attribute is the most difficult to replicate. The features used are a list of commands entered by the attacker as well as a malware detection rate returned by VirusTotal [66] and a Threat level returned by Symantec [67]. However, to get those two last features, one needs access to the malware itself, which we do not have access to, at the time of writing. We only

have the command executed by the attacker (i.e. the name of the malware) and sometimes the IP address the attacker used to obtain the malware. In the end, to approximate the malware detection rate, we use the number of times that IP address was flagged as malicious by VirusTotal. However, this feature is not easy to generalize for all protocols. Therefore, for most protocols, only the number of commands entered by the attacker is used. Only for some protocols, do we also have the approached malware detection rate. Figure 4.5 shows the direct comparison between the results as there are no outliers for this attribute. However, for the reasons explained above, those results are difficult to compare.

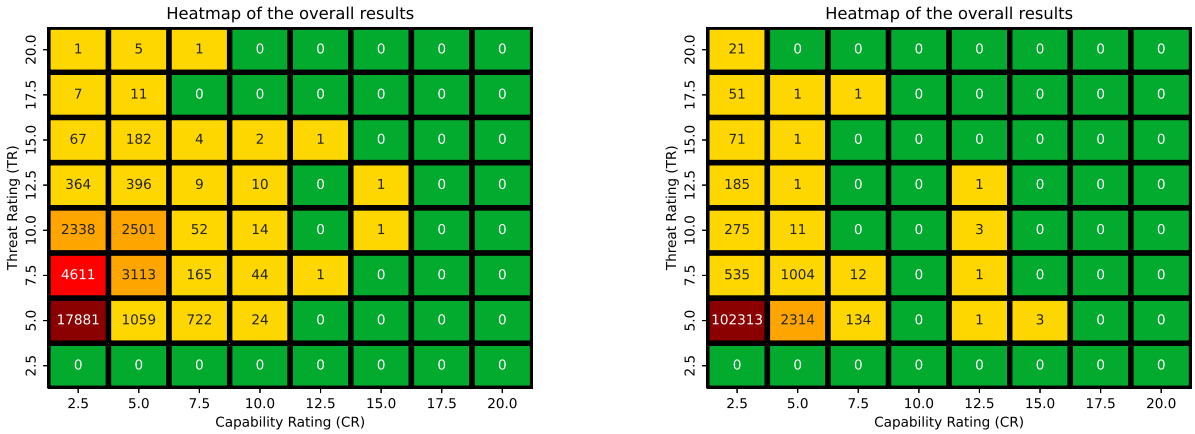


Skill ratings presented in [45] Reproduction of Skill calculation

FIGURE 4.5 Comparison of the Skill ratings distributions

Final Results

From the attributes defined above, we get the threat and capability ratings for each attacker. A comparison of these is shown in Figure 4.6. The results are relatively similar, with most of the attackers grouped in the same place on the graph, representing the fact that most attackers are rather harmless. Here again, we see the impact of adding a time constraint when aggregating the logs with a higher density of attackers in the same group. The outliers mentioned before are also easily identified on this graph as the 21 attackers on the top left (motivation outliers) and the detached group of attackers on the right (resources outliers). However, besides the outliers that were already identified before, it is difficult to identify what characterizes the attackers in each cell of the heat map. Therefore, it is difficult to find out exactly how many attackers should be considered for a more detailed analysis. This will depend on the network administrator in charge and his or her experiences. This is an issue we address in the following.



Final results presented in [45] Reproduction of final results

FIGURE 4.6 Comparison of the final results

4.4.2 Generalized Results

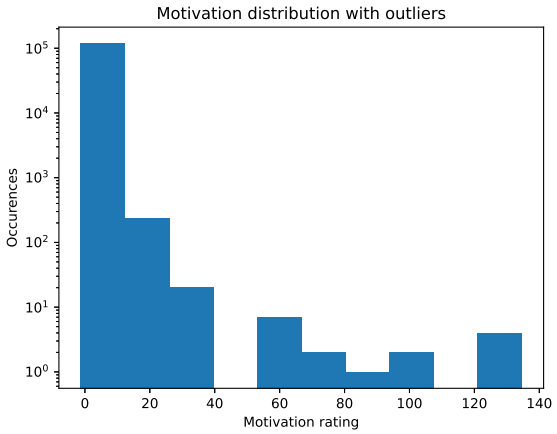
In this section, we present our proposed generalized methodology to characterize attackers through five attributes. We derive these attributes through features that can be extracted from many protocols. Furthermore, we include an OT protocol (i.e. Modbus) to show that the methodology is not only limited to IT protocols.

Motivation Ratings

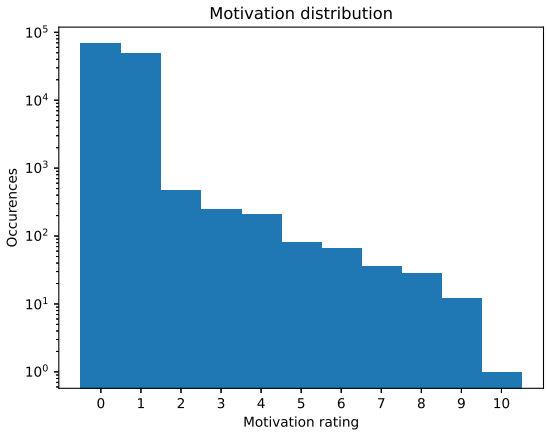
The definition of the motivation attribute is similar to what was done in Section 4.4.1. Indeed, to determine the motivation ratings, we are using the total number of commands entered, the total time of the attack, the number of sessions initiated, and the number of protocols attacked. Results resembling the ones in Section 4.4.1 are shown on the left side of Figure 4.7. There are actually a few extra outliers here from the "number of protocols" feature. Indeed, while most attackers focus on one or two protocols, some attempt to compromise up to seven different services and have, therefore, a higher motivation score. For continuity, we also show the motivation ratings scaled in the $[0, 10]$ range after removing the outliers on the right side.

Resourcefulness Ratings

For the resourcefulness ratings, we use the inter-arrival time to capture the degree of automation of an attacker. To capture the degree of expertise and knowledge of the attacker, we use the number of different commands entered and the number of downloaded files. Figure 4.8 shows the results, which at first seem similar to the ones in Section 4.4.1. However, after



Motivation ratings with outliers



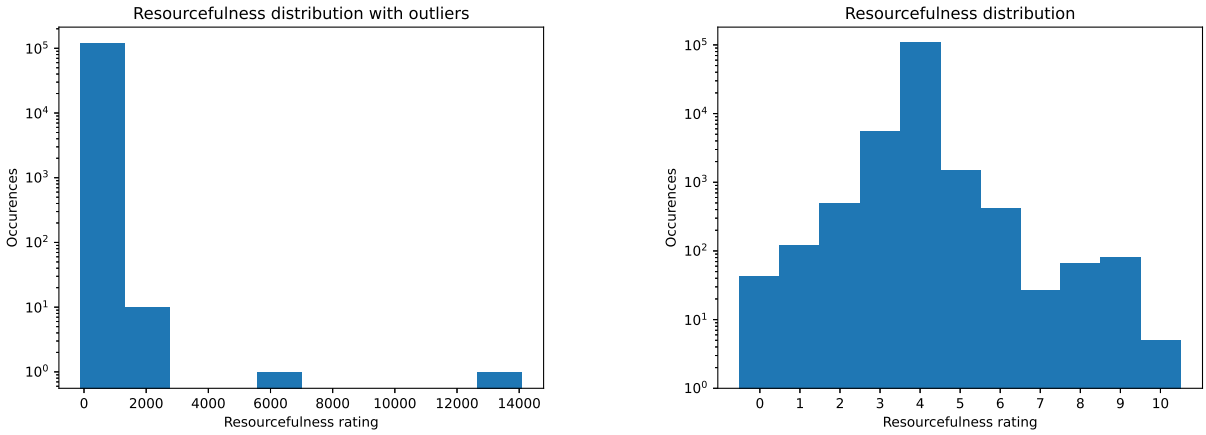
Motivation ratings

FIGURE 4.7 Motivation results

removing the outliers and scaling the values, we see a different distribution. The outliers here represent attackers who sent lots of commands in a short time interval and who sent lots of different commands. Indeed, we observe that some attackers try several commands for the same goal, which would transcribe as having more expertise and knowledge in this model. The reason why most of the attackers have a medium resourcefulness score is once again because of the missing time-to-live. Indeed, the low-resourcefulness scores represent attackers who maintain a connection open for a few weeks, but only send a few commands. Therefore, the inter-arrival time is especially low for these attackers.

Stealth Ratings

The stealth ratings are calculated from the approached malware detection rate for the downloaded malwares, as calculated in Section 4.4.1. The idea is that if the IP address from which the malware is downloaded has been flagged as malicious by VirusTotal, then it can easily be part of a banned list. The second feature we use is the number of actions the attackers take to hide their presence. Amongst others, this includes removing written files, wiping out any written data, and resetting communication links. The last feature included in the stealth attribute is the number of scans performed by the attacker. Some observations do not explicitly contain payloads and seem to represent port scans that an attacker would do. The more scans are performed, the less stealthy the attackers are. Figure 4.9 shows the results with and without outliers. The single outlier, in this case, represents an attacker who takes a lot of precautions not to get caught by resetting communication links multiple times and

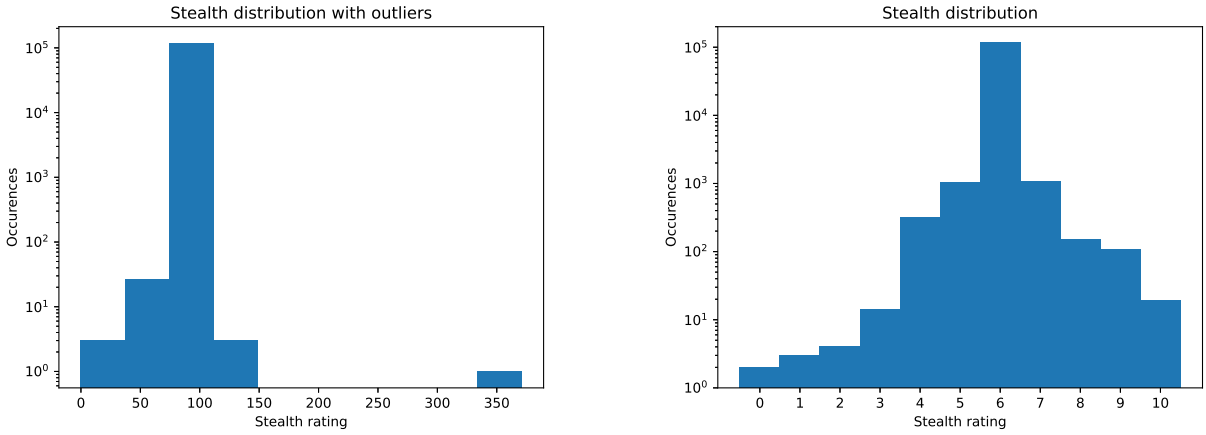


Resourcefulness ratings with outliers

Resourcefulness ratings

FIGURE 4.8 Resourcefulness results

removing any written data. On average, however, attackers do not spend much effort trying to hide themselves. The reason why the distribution of the stealth ratings is skewed on the left is that some attackers send many scans, resulting in a low stealth score. In opposition, most attackers send few scans or even no scans at all.



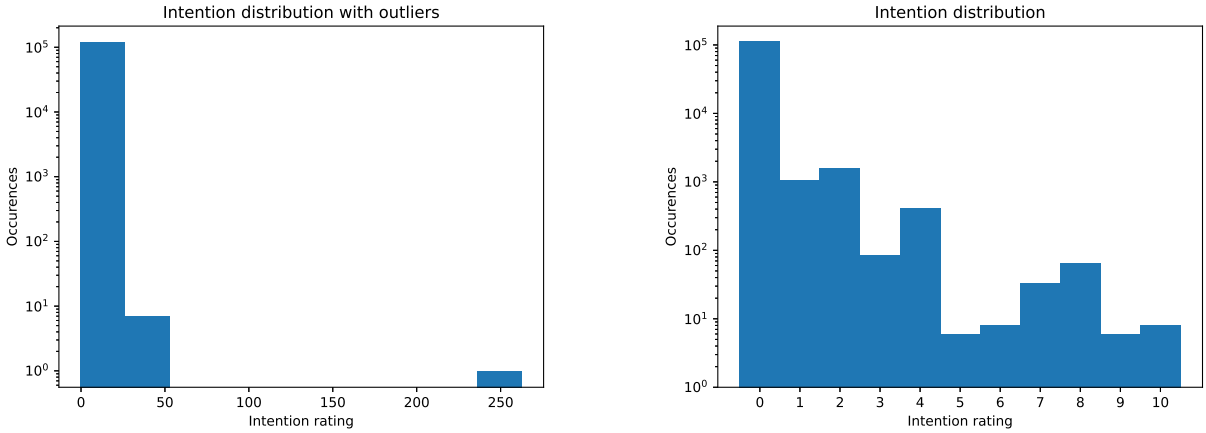
Stealth ratings with outliers

Stealth ratings

FIGURE 4.9 Stealth results

Intention Ratings

For the intention ratings, we classify each command into one of the following actions : read, write, execute, or other. Therefore, databases are created, linking each command to the corresponding action for each protocol. For example, in SSH, changing the permissions of a file using *chmod* counts as an "other" action but running a file afterwards counts as an "execute" action. This gives, for each attack, an array describing how many times the attackers try to read data, write data, or execute some code on the system. To refine this into a 1-dimensionality characteristic we calculate $intention\_score = \alpha * execute + \beta * write + \gamma * read$. Where, α, β, and γ represent the weighting of the *execute*, *write*, and *read* values, respectively. To stay general, we evaluate each action similarly and set each weight to 0.33. Figure 4.10 shows the distribution of the intention ratings. The single outlier here is an attacker whose commands could all be classified as "read" and "write" actions, while targeting exclusively the Modbus protocol for more than 1500 observations. After removing the outlier and scaling the values, we see that most attackers have low intention scores, likely since most attackers perform a few scans without acting upon them.



Intention ratings with outliers

Intention ratings

FIGURE 4.10 Intention results

Originality Ratings

The calculation of the originality ratings is done in two steps. First, a data mining algorithm is run on the attackers' commands to build a database of the most commonly used ones. This identifies 28 commands that are very frequent and used by a lot of different attackers. So far, we only use it on the SSH/Telnet observations as it yields the most promising results, but

we intend to do this on the other protocols as well, as we gather more data. The originality score for each attacker starts at 28, then for each sequence used in the attackers' database we subtract 1. This means that the attackers who have an originality score of 0 enter most of the 28 commonly used commands in their attack on the SSH and Telnet protocols. Figure 4.11 shows the distribution of the originality ratings obtained this way. The far left side of this figure features the attackers who would be identified as "script kiddies", reusing common scripts with similar commands. The far right side of this figure features two kinds of attackers : the ones who sent only a few commands while not really performing any attack and the attackers who perform complex attacks without using common commands.

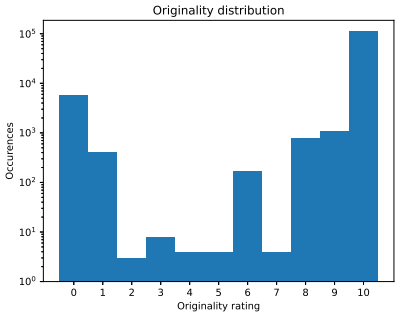


FIGURE 4.11 Originality results

Final Results

For continuity, we can still define a threat rating and a capability rating, as shown in Figure 4.12. However, we find the same problems as above ; we do not know what exactly characterizes attackers in each cell and which attackers to consider depends on one's experience and interpretation. Therefore, we instead propose to study the attackers with a clustering analysis. Through the use of the Elbow method applied to several metrics, we determine that the best number of clusters to split our data is 13 clusters. Figure 4.13 shows how many attackers each cluster contains. In the end, we have the following observations :

— The first three clusters contain more than 90% of the attackers who are identified as harmless. Indeed, these attackers usually only send a few scanning commands but do not perform any attack.

— The last cluster is the only one that contains potentially dangerous attackers. Indeed, looking back at the logs, we see that they send several hundreds or thousands of commands, which rarely use pre-built scripts. They also take several actions to hide themselves, such as removing most downloaded and written files. Finally, they try different ways to compromise the system by sending many different commands.

— The attackers who belong to the other clusters usually try a few commands like reading and writing data or exploit basic vulnerabilities such as the missing time-to-live. However, those attackers only perform basic attacks using pre-built scripts and are therefore not so alarming.

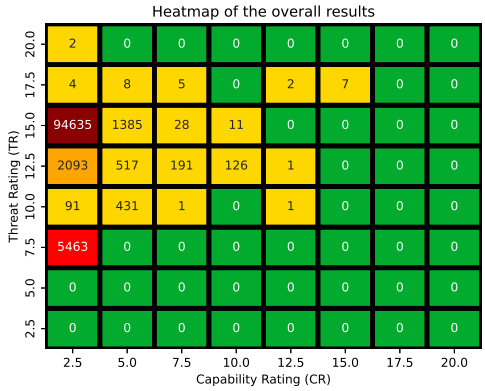


FIGURE 4.12 Final results

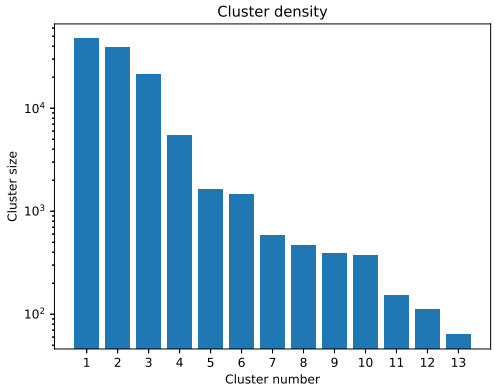


FIGURE 4.13 Cluster distribution

4.5 Discussion

As shown in Section 4.4.2, the generalized approach allows us to extend the methodology introduced by Fraunholz et al. [45] for any combination of protocols as it uses common features. Anyone who wants to analyse logs from one or several honeypots at once could therefore use this approach. Furthermore, we show that this methodology can be used on any kind of honeypot as it is shown working on low-interaction honeypots and only needs a recording of the attackers' commands. However, higher interaction honeypots will keep the attackers' interests for a longer time and might result in more attackers identified as alarming. We also propose a new way to analyse the results that returns the specific attackers that network administrators would need to consider, independently of their experience. Indeed, only the attackers who are identified as outliers through the attribute analysis and the attackers who belong to the dangerous cluster need to be considered for a more detailed analysis. This means that out of a few million logs, more than one hundred thousand attackers are identified, and out of those attackers, only 111 (49 outliers and 62 from the dangerous cluster) represent a real threat and need to be studied more thoroughly by a network administrator. However, our methodology only analyses the attackers separately and does not consider evasion techniques. Indeed, attackers who perform their attacks using multiple IP addresses are not correlated together. This is a simple evasion technique that many attackers perform and is a main limitation of the presented approach.

4.6 Conclusion

In this paper, we tackle the problem of cyber attacker attribution. Cyber attacker attribution tries to gain insight into attackers instead of the attacks. We expand upon an already established idea to characterize the attackers through different attributes. We generalize this approach so that it can be used on any protocol and apply it to honeypot logs collected from real attacks on several services, both IT and OT. We demonstrate that this methodology would greatly help a network administrator to identify a fixed number of attackers that need to be investigated more carefully, independently of that network administrator's experience and interpretation. Furthermore, this methodology is self-sufficient in that it does not depend on other tools and does not need human-based help to identify those attackers. In future work, our major focus will be towards considering attackers using evasion techniques. We aim to associate each attacker to a threat group as described in the MITRE ATT&CK framework [68] and link attackers who belong to the same threat group together. Finally, we will also validate this methodology on new datasets as they are gathered from the honeypots.

Acknowledgements We would like to thank Thales Digital Solutions for their generous support to enable this work.

CHAPITRE 5 RÉSULTATS COMPLÉMENTAIRES

5.1 Introduction

Le renseignement de menace ainsi que l'attribution s'intéressent aux auteurs des attaques afin de comprendre leurs comportements et motifs. La collecte et l'étude de ces informations, qui sont cruciales pour protéger une organisation, manquent actuellement de cadre générique et d'outils automatisés pour aider les entreprises. Un rapport de Mandiant en 2023 [69] explique que beaucoup d'organisations ne considèrent pas l'étude en renseignement de menaces lors de prises de décisions sur les stratégies défensives à acheter et utiliser. Dès lors, ces organisations utilisent souvent des stratégies défensives qui ne sont pas adaptées à leurs besoins et qui ne les défendent pas toujours des acteurs qui les attaques effectivement. De cela, résulte une grande perte de temps et d'argent [69]. Pour palier à cela, les organisations devraient dédier plus de temps à comprendre qui sont les acteurs derrière les attaques et quels sont leurs buts.

Dans [29], Bada M. et al. mentionnent que l'attribution, aussi appelée profilage de cyberattaquants, est un sujet de recherche assez récent avec un intérêt grandissant. Cependant, l'attribution manque d'une définition commune et d'une approche systématique. Dans ce chapitre, nous utilisons la base de connaissance MITRE ATT&CK [15] comme soutien pour notre travail. Cette base de connaissance fournit un aperçu détaillé de différentes tactiques et techniques que les attaquants peuvent utiliser lors de leurs attaques. Ces tactiques et techniques représentent alors le "pourquoi" et le "comment" ces attaques surviennent. Pour aider à l'attribution, cette base de connaissance fournit également des informations sur des groupes d'attaques connus et leurs techniques préférentielles, représentant une sorte de mode opératoire pour ces groupes d'attaques.

Une première tentative dans le domaine, faite par Lim C. et al. [27], introduit un algorithme manuel de catégorisation de menaces pour collecter des informations sur les attaquants d'un Honeypot. Ces informations sont obtenues sous forme de techniques utilisées du cadriciel MITRE ATT&CK.

Nous nous basons sur l'algorithme proposé dans [27] pour étiqueter manuellement un trafic collecté d'un Honeypot Cowrie [64] en termes de techniques définies dans MITRE ATT&CK. Cet algorithme propose de passer séquentiellement à travers toutes les commandes observées dans le jeu de données et de trouver les techniques correspondantes, le cas échéant. Cela permet de représenter un attaquant uniquement via les techniques qu'il emploie. Avec cette représentation, nous considérons deux façons d'analyser ces attaquants. Premièrement, nous

regroupons les attaquants individuellement en utilisant l'algorithme DBSCAN [70], adapté à notre type de donnée, comme expliqué plus tard dans ce chapitre. Ensuite, nous définissons notre propre algorithme de corrélation pour corréler les attaquants pendant la phase de regroupement. Cette corrélation est induite de liens sémantiques apportés par les modes opératoires récupérés du cadriciel MITRE ATT&CK.

De ce fait, nous proposons une nouvelle méthode pour regrouper et corréler les attaquants de notre Honeypot, puis lions ces attaquants à des modes opératoires définis dans MITRE ATT&CK. Nous dérivons un algorithme de regroupement basé sur une similarité entre les attaquants et les modes opératoires et sur une métrique mesurant le taux de corrélation entre les attaquants regroupés. Cela nous permet d'utiliser l'information comprise dans MITRE ATT&CK comme information a priori pour le regroupement. Ces informations nous permettent alors une influence sur la direction du regroupement vers des résultats compréhensibles et interprétants. Nous nous intéressons donc aux points suivants :

- Proposer une méthode pour représenter les attaquants comme des vecteurs binaires en termes de techniques utilisées de MITRE ATT&CK.
- Proposer un regroupement individuel des attaquants, en rassemblant ceux utilisant des techniques similaires.
- Utiliser les groupes d'attaques de MITRE ATT&CK comme interprétation sémantique des groupes observés.
- Proposer un algorithme non supervisé pour corréler et regrouper les attaquants.
- Fournir une certaine compréhension du contexte des cybermenaces actuel et des techniques utilisées pour attaquants de notre Honeypot.

Dans ce chapitre, on retrouve les aspects théoriques et travaux similaires dans la Section 5.2. La Section 5.3 introduit la collection des données et la méthodologie appliquée. Dans la Section 5.4, nous appliquons notre méthodologie sur les données collections et démontrons ces résultats. Enfin, dans la Section 5.5, nous discutons des résultats et des limites de notre approche.

5.2 Aspects théoriques

L'attribution cherche à découvrir des informations sur les auteurs des attaques. Récemment, différentes façons de collecter et d'analyser ces informations ont été proposées. Certaines recherches étudient l'utilisation de graphes d'attaques [47, 49] pour représenter tous les chemins possibles d'un système initial à un système piraté. Cependant, la construction de tels graphes demande une connaissance experte du système considéré, ce qui est difficile à appliquer à des systèmes réels. D'autres utilisent les modèles de Markov cachés (HMMs) [42, 53],

qui discrétisent le vecteur d'attaque en états et modélisent les actions des attaquants par des transitions entre ces attaquants. Par définition, les états et les probabilités de transition d'un HMM sont fort dépendants du jeu de données considéré, ce qui rend ces modèles difficiles à généraliser et à étendre. Enfin, certains chercheurs utilisent des ensembles de règles, floues, [54] ou argumentatives [59], qui offrent une représentation actuelle du monde en plus des évènements observés. Ces études introduisent généralement leurs propres ensembles de règles, qui tendent à traîner derrière les réalités du monde qui évoluent rapidement. Inévitablement, lorsque l'ensemble de règles utilisées ne reflète plus précisément l'état actuel du monde, les résultats d'attribution deviennent désuets.

Tous ces travaux travaillent sur l'attribution de leur propre façon, en donnant chacun leur propre définition de l'attribution, des attaquants et des catégories d'attaquants. Bada M. et al. [29] identifient ce problème et suggèrent l'adoption d'une définition commune et d'une approche commune pour s'intéresser à ce problème. C'est dans ce contexte que se situe ce travail, en suivant leur recommandation par la définition de nos règles et définitions en utilisant MITRE ATTA&CK, un organisme dont le seul but est de fournir un tel cadriciel au monde pour informer et aider les tâches de renseignements de menaces telles que l'attribution.

Avec l'intérêt grandissant envers l'attribution, les chercheurs ont réalisé que cette tâche requiert une certaine base d'informations, sous forme d'indicateurs tels que des noms de domaines, des adresses de protocole internet (IP), des tactiques, techniques et procédures (TTPs) ou des artéfacts réseau. Cependant, tous ces indicateurs n'ont pas la même valeur. Warikoo explique dans [71], par l'utilisation de la "pyramid of pain" de David J. Bianco [1], que les valeurs de hash et les adresses IP ne sont pas de bons indicateurs pour l'attribution étant donné que les attaquants peuvent facilement changer ces valeurs. D'un autre côté, les TTPs sont de bons indicateurs pour l'attribution étant donné "qu'elles réfèrent au comportement et à l'artisanat des attaquants", qui sont plus difficiles à changer.

Ces TTPs sont présentes dans la matrice MITRE ATT&CK que les chercheurs ont commencé à utiliser dans leurs processus d'attribution, et sont une bonne façon de définir l'approche commune dont nous avons besoin. Certains l'utilisent pour acquérir des connaissances dans des rapports en source ouverte tels que les rapports de renseignements de menaces (CTI) [72] ou simplement pour récupérer toutes les informations accessibles sur internet [73] et calculer des similarités entre différents groupes d'attaques. Cependant, ces recherches présentent le même problème qui est qu'elles ne peuvent être utilisées en temps réel, car elles demandent de l'information générée après qu'un incident ait eu lieu et que quelqu'un l'ait déjà enquêté. D'autres recherches utilisent MITRE ATT&CK pour corréler des informations collectées par les alertes de systèmes de détection d'intrusions (IDS) [74] ou par les logiciels malveillants eux-

mêmes [23] pour reconstruire l'incident entier et attribuer l'attaque à un groupe spécifique. Notre travail suit un processus similaire, mais en travaillant directement sur les commandes entrées par les attaquants sur notre Honeypot. Nous réduisons donc le besoin d'un logiciel tiers tel qu'un IDS ou d'un bac à sable pour étudier les logiciels malveillants.

Dans [27], Lim C. et al. proposent un algorithme pour étiqueter les commandes des attaquants collectés sur un Honeypot Cowrie et présentent un sous-ensemble de leur liste d'étiquetage. Puis, dans [28], les mêmes auteurs utilisent cette liste et présentent une vue d'ensemble de ce qu'il se passe sur leur Honeypot, en montrant les fréquences d'usage de chaque technique identifiée. Cependant, cette analyse n'utilise pas d'information fournie par MITRE ATT&CK concernant les différents groupes d'attaques. Dès lors, leur analyse ne réalise pas de tâche d'attribution. Nous pensons que ces informations de groupes d'attaques peuvent grandement aider à définir un cadre rigoureux pour l'attribution, ce qui est l'accent porté par notre travail.

5.3 Méthodologie

5.3.1 Collection des données

Trouver un jeu de données adapté pour travailler sur l'attribution n'est pas une chose facile. Un manque de données dans le domaine a été identifié comme le défaut principal par les recherches dans le domaine [13, 29]. En effet, au meilleur de nos connaissances, il n'existe pas de jeu de données en source ouverte contenant de l'information sur les attaquants, qui serait adapté pour les études en renseignement de menace et en attribution. Cependant, Doynikova E. et al. [13] proposent d'utiliser des données collectées d'Honeypots pour séparer les problèmes d'attribution et de détection étant donné que les Honeypots n'ont aucune fonctionnalité de production. Cela nous donne la propriété intéressante que tout trafic récolté est suspicieux tout du moins. Aussi, ces chercheurs mentionnent que la collection de données à partir de Honeypots est ce que l'on a de plus proche actuellement d'un trafic réel. Cependant, étant donné que les Honeypots sont manufacturés pour récolter de l'information sur les attaquants, ils sont généralement moins défendus qu'un serveur réel. Il ne serait donc pas sage de penser qu'aucun attaquant ne se rend compte qu'il attaque un Honeypot. Tout de même, c'est l'approche que l'on a considérée dans notre étude, dû à un manque de meilleure option.

Nous avons utilisé un Honeypot Cowrie accessible publiquement depuis internet. Les détails complets d'implémentation de cet Honeypot sont omis ici pour préserver la propriété intellectuelle de notre partenaire industriel. Nous avons collecté des données sur plus de 2 mois, à partir de décembre 2021 et avons collecté plus d'un million d'entrées. Ces entrées sont au

format “NDJSON” et contiennent l’information dans des champs clé-valeur, tels que l’adresse IP de l’attaquant, l’heure de l’attaque, les cookies de session, les commandes, etc. Une version simplifiée de ces entrées peut être trouvée au Tableau 5.1. Avec ce Honeypot déployé publiquement, sur des adresses IP inutilisées, nous avons collecté diverses attaques : certaines plus simple où l’attaquant ne fait que collecter des informations sur le serveur et d’autres, plus élaborées, où des attaquants téléchargent et exécutent plusieurs logiciels malveillants après une phase initiale de reconnaissance. De ces entrées, l’information nécessaire pour ces travaux sont les adresses IP des attaquants, les heures d’attaques, et les commandes utilisées, i.e. les charges utiles des paquets que l’on va ensuite transformer en techniques du cadriciel MITRE ATT&CK.

D’un autre côté, nous avons également besoin de dériver une liste d’étiquetage pour associer les commandes utilisées par les attaquants aux techniques MITRE ATT&CK. Cela est fait de la manière proposée dans [27] mais avec une procédure plus automatisée. En effet, les techniques telles que présentées par MITRE ATT&CK contiennent une description de commandes qui peuvent être utilisées pour réaliser cette technique. Par exemple, pour la technique de découverte de fichiers et dossiers, il est mentionné qu’un attaquant peut la réaliser en utilisant les commandes “dir”, “tree”, “ls”, “find”, et “locate”. Nous reprenons ces informations et formons notre liste d’étiquetage sur base de celles-ci, générant ainsi une liste de plus de 100 règles. Cette liste associe à certaines commandes, la ou les techniques MITRE ATT&CK visées, similairement à ce qui est fait dans les travaux de Lim C. et al. [27]. En revenant sur l’exemple précédent au Tableau 5.1, l’attaquant essaie d’accéder aux informations de l’unité centrale de traitement (CPU) et cela équivaut à la technique “$T1082$: Découverte d’information système”.

5.3.2 Représentation des attaquants

En utilisant la liste d’étiquetage définie à l’étape précédente, nous dérivons un vecteur d’attaque à partir des commandes entrées par l’attaquant, représentant les techniques utilisées.

Tableau 5.1 Simplified example of a log

Attribute	**Value**
Session	233ed93a0c46
Timezone	Central Standard Time
Source IP	157.245.101.31
Command	cat /proc/cpuinfo | grep name | wc -l
Timestamp	2021-12-27T19 :26 :09.727623Z

Cependant, considérer tous les logs de notre Honeypot de façon individuelle rend nos représentations très clairsemées et les résultats assez limités. Similairement à ce qui est fait au chapitre précédent, nous considérons que des logs venant de la même adresse IP en moins d'une heure proviennent du même attaquant. Cette hypothèse est aussi utilisée dans d'autres ouvrages de la littérature [42, 65] pour obtenir une représentation tangible d'une attaque.

De cette façon, nous identifions plus de 10000 attaquants dans nos données. Pour chaque attaquant, nous obtenons un vecteur binaire avec un 1 à la position i si la technique i a été utilisée par l'attaquant et un 0 sinon. Ces vecteurs sont de taille T, avec T le nombre de techniques de la matrice MITRE ATT&CK. Pour ce travail, nous considérons $T = 26$, le nombre de techniques observées dans nos données Honeypot. Au moment de la rédaction de ce mémoire, le nombre total de techniques présentes dans MITRE ATT&CK est de 185.

5.3.3 Aperçu des techniques utilisées

Nous avons d'abord cherché à répliquer le processus exposé dans [28] sur nos données. Ce processus donne un aperçu à haut niveau des techniques utilisées sur notre Honeypot. Il identifie toutes les techniques utilisées puis les classe en fonction de leur fréquence. Les résultats de cette procédure sont présentés dans la Section 5.4.1. Dans [28], les auteurs expliquent chaque technique en détail, ce que nous nous abstenons de faire ici étant donné que le lecteur intéressé peut trouver ces informations dans leur article ou sur le site de MITRE ATT&CK directement [15].

5.3.4 Analyse individuelle des attaquants

Une manière directe d'analyser les attaquants est d'utiliser un algorithme de regroupement pour trouver des attaquantes similaires, qui utilisent les mêmes techniques. Avec notre méthodologie, on voit facilement que le nombre de représentations d'attaquant est de 2^T. Dès lors, nos représentations sont fort clairsemées et on peut y trouver beaucoup de bruit. Cela perturbe un algorithme de regroupement basé sur des prototypes comme K-means, c'est pourquoi nous avons choisi d'utiliser DBSCAN, un algorithme de regroupement basé sur les densités. Cet algorithme fonctionne bien dans ces conditions, comme expliqué dans [70]. Il regroupe ensemble les points qui sont une à une distance inférieure à un paramètre, nommé ϵ. Les données regroupées ne sont ensuite considérées comme un regroupement que si au moins min_points données sont proche. Dès lors, ϵ et min_points sont des paramètres de l'algorithme qui vont influencer les résultats et le nombre de regroupements trouvés. Nous utilisons cet algorithme et réglons les paramètres comme indiqué dans [75]. Le paramètre min_points est réglé à $(2 * dim) - 1$, où dim représente la dimension des données, soit T,

le nombre de techniques dans notre cas. Le paramètre ϵ est évalué en utilisant un graphique de K-distance, qui peut être trouvé à la Figure 5.1. Cette figure montre que l'on peut former nos regroupements en réglant ce paramètre ϵ à 0. Cela veut dire que deux attaquants seront considérés dans le même regroupement s'ils ont utilisé exactement les mêmes techniques.

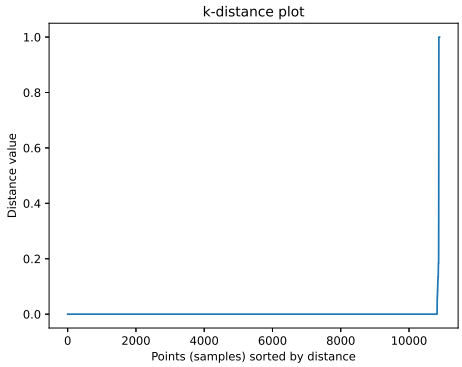


FIGURE 5.1 k-distance plot

Après avoir identifié les groupes, nous les interprétons en utilisant l'information fournie par les groupes d'attaque de MITRE ATT&CK. Nous regardons les similarités entre les attaquants de chaque groupe et les groupes d'attaques obtenus. Il est intéressant de noter que cet algorithme n'essaie pas d'associer chaque donnée à un groupe, mais définit un groupe de "bruit" pour toutes les données qui n'appartiennent à aucun regroupement. Les résultats de cette méthode sont présentés à la Section 5.4.2.

Avec ce genre de regroupement, nous ne traitons les attaquants qu'individuellement et utilisons l'information de MITRE ATT&CK qu'a posteriori pour interpréter les regroupements trouvés. Cependant, au vu de la manière dont nous considérons les attaquants, basés sur l'adresse IP et une contrainte de temps, il est plus que probable que l'on puisse trouver des corrélations, un attaquant unique réalisant son attaque à partir de plusieurs adresses IP ou même un groupe d'attaquants réalisant une attaque ensemble. En effet, une manière relativement simple d'implémenter de la déception dans une attaque est de la réaliser à partir de plusieurs adresses IP pour brouiller les pistes [76]. Nous proposons donc ensuite un algorithme de regroupement qui tient profit de l'information des groupes d'attaques dans MITRE ATT&CK pour corréler les attaquants lors de la phase de regroupement.

5.3.5 Analyse des attaquants par corrélation

Dans l'algorithme 1, nous présentons notre procédure qui boucle à travers tous les attaquants d'un ensemble "S" pour trouver des corrélations satisfaisant un seuil de confiance "ct" et

maximisant un seuil de similarité "ft" envers un mode opératoire considéré "g". Les détails de l'algorithme sont expliqués ci-après.

Le seuil de confiance est un paramètre régulant le nombre de tentatives de l'algorithme pour trouver des liens entre les attaquants. Lorsque l'algorithme ne trouve plus d'attaquants qui satisfont au seuil de confiance ou qui améliorent la similarité envers le mode opératoire considéré, il s'arrête. Si la similarité est plus grande que le seuil de similarité spécifié, on obtient ainsi une piste d'attribution. Le seuil de similarité est le second paramètre qui permet de réguler les sorties de l'algorithme en séparant les comportements effectivement malveillants de ceux jugés plus bénins. Les deux paramètres sont compris dans l'intervalle $[0, 1]$.

Cette approche est gourmande en ce qu'elle essaie de maximiser la similarité à chaque itération. On pourrait aussi regarder aux corrélations de tout groupe possible d'attaquants satisfaisant le seuil de confiance donné, mais cela ferait exploser la complexité temporelle. L'algorithme utilise cette approche gourmande comme un compromis entre complexité et complétude. De cette façon, la complexité temporelle est $\mathbf{O}(n^2)$ en moyenne, avec n le nombre d'attaquants considérés (par Gauss [77]). Cet algorithme peut ensuite être appliqué à chaque mode opératoire que l'on souhaite étudier, ce qui élève la complexité temporelle à $\mathbf{O}(gn^2)$, avec g le nombre de groupes d'attaques considérés. Dans le cas où l'on considère plusieurs groupes d'attaques différents, si un attaquant est retenu pour plus d'un groupe, on peut soit garder celui avec la similarité la plus grande ou bien obtenir des probabilités d'appartenance dictées par les scores de similarités.

Nous définissons deux formules pour cet algorithme. Premièrement, pour exprimer la similarité entre un mode opératoire et un groupement observé d'attaquant, nous utilisons l'équation 5.1, où s représente le nombre de techniques en commun entre le mode opératoire et le groupe observé d'attaquant. o représente le nombre de techniques présentes dans le groupe observé d'attaquant, mais qui ne sont pas présentes dans le mode opératoire. Enfin, N est le nombre total de techniques utilisées par le mode opératoire, qui agit en temps que valeur de normalisation.

$$similarity = (s - o)/N \tag{5.1}$$

Ensuite, pour définir la corrélation entre attaquants observés, nous utilisons l'équation 5.2, où t représente le nombre de jours entre les activités des attaquants (avec un minimum de 1) et m représente une correspondance entre les adresses IP utilisées. Cette correspondance se base sur le nombre de bits de poids fort en commun. Nous utilisons aussi un facteur $\alpha \in [0, 1]$ qui régule à quel point la correspondance d'adresses IP influence le calcul de corrélation.

Algorithm 1: Correlate and Cluster

input: Attackers S, threatGroup g, fitThreshold ft, confidenceThreshold ct

for *each attacker* a_i *in* S **do**
- `// Initialisation de variables`
- confidenceScore ← 1
- grouped ← $\{a_i\}$
- lastSimilarity ← -1
- similarity ← FindSimilarity(grouped, g)
- `// Loop while a better similarity is found`
- **while** *similarity* > *lastSimilarity* **do**
  - `// Find next candidate and update variables`
  - Find attacker a_j in S that maximises similarity to g such that confidenceScore * correlation > ct
  - confidenceScore ← confidenceScore * correlation
  - grouped ← grouped + $\{a_j\}$
  - lastSimilarity ← similarity
  - similarity ← FindSimilarity(grouped, g)
- **end**
- **if** *similarity* > ft **then**
  - S ← S - grouped
  - Assign cluster associated with group g to all attackers in grouped
- **end**

end

Dans nos expériences, nous l'avons fixé à 0.125 car nous souhaitons que les corrélations se fassent principalement en fonction du temps. Cependant, dans le cas de différence temporelle sensiblement similaire, nous préférons que les attaquants provenant du même réseau aient une corrélation supérieure. Cette valeur de α s'est montrée efficace à cet égard. On peut facilement vérifier que les valeurs de corrélation tombent toujours dans l'intervalle $[0, 1]$.

$$correlation = (1/\sqrt{t}) - \alpha\frac{32 - m}{32} \tag{5.2}$$

D'une part, l'algorithme est régulé par ces deux formules. D'autre part, il est régulé par les deux seuils qu'on lui donne, qui auront un impact significatif sur les résultats. Le seuil de similarité permet de réguler comment les liens sémantiques entre attaquant et modes opératoires sont fixés. Si ce paramètre est à 0, l'algorithme acceptera toute similarité trouvée entre attaquants et modes opératoires. Dans ce cas, nous obtiendrons beaucoup de faux positifs étant que même l'utilisation d'un utilisateur légitime pourrait être en partie associée à un mode opératoire. Si ce paramètre est mis à 1, nous ne trouverons en sorte que les liens entre attaquants et modes opératoires qui résultent en une similarité parfaite. Cependant, cette similarité parfaite peut être rare et cela empêchera donc l'algorithme de faire une grande partie de pistes d'attribution presque parfaites. Dès lors, ce paramètre devrait être fixé en accord avec les besoins et les modes opératoires considérés.

Le seuil de confiance régule les corrélations faites par l'algorithme. S'il est mis à 0, tous les attaquants pourront être corrélés ensemble. Il en résultera probablement de fausses corrélations. Cependant, le mettre à 1 est très strict et empêche l'algorithme de faire la plupart des corrélations. Ce paramètre a également un impact sur la complexité temporelle de l'algorithme, car le fixer à 1 empêche l'algorithme de passer dans la boucle et le fixer à 0 force l'algorithme à épuiser toute corrélation possible dans la boucle. Dès lors, il est nécessaire de régler cette valeur pour obtenir les meilleurs résultats.

Nous appliquons notre algorithme aux données collectées de notre Honeypot dans la Section 5.4.3.

5.4 Évaluation

5.4.1 Aperçu des techniques

Nous commençons par réaliser l'analyse globale des techniques utilisées par les attaquants, similairement à ce qui a été fait par Lim C. et al. [28], en comptant les fréquences d'utilisation. Les résultats des 10 techniques les plus fréquentes sont montrées dans la Figure 5.2. Nous

remarquons que les attaquants sont en général intéressés de comprendre la machine dans un premier temps, cherchant à découvrir des informations sur les systèmes, les identifiants ou les logiciels. Après avoir récolté ces informations, ils commencent leurs attaques. Étant donné que nous n'avons pas la totalité de la liste d'étiquetage de Lim C. et al. , cette étape nous permet dans une certaine mesure de valider notre liste étant donné que nous obtenons des conclusions similaires à celles de ces auteurs.

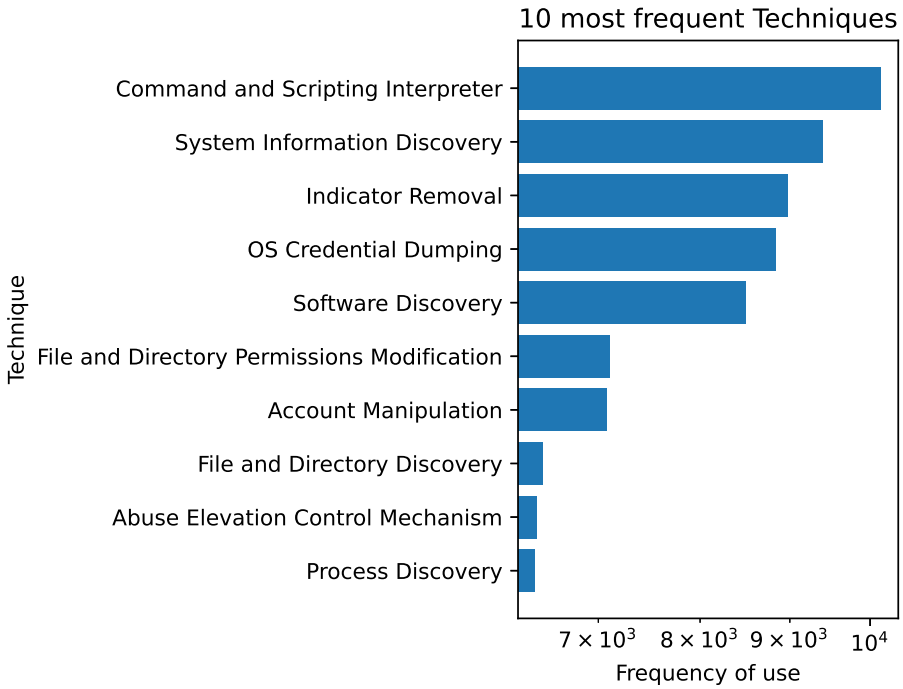


FIGURE 5.2 Frequently used techniques on our Honeypot

5.4.2 Analyse individuelle des attaquants

Avec nos données, 17 groupes sont identifiés et 587 attaquants sont labélisés comme "bruit". Cela veut dire qu'il y a 587 attaquants utilisant des vecteurs d'attaques différents de la majorité des attaquants. Dans un scénario réel, ceux-ci devraient être identifiés et considérés pour une analyse manuelle complémentaire. Il est également intéressant de noter que l'on retrouve tous les attaquants identifiés au chapitre 4 dans ces 587 attaquants. Les résultats de ce regroupement sont présentés à la Figure 5.3. Les différents groupes sont représentés par leurs modes opératoires les plus similaires. Ici, cette similarité est simplement définie par le nombre de techniques qu'ils ont en commun. Cela explique pourquoi certains modes opératoires, tels que "FIN6" sont présents plusieurs fois dans la Figure 5.3. Ils correspondent

à différents regroupements et ont des similarités différentes. Les similarités de chaque regroupement peuvent être trouvées au Tableau 5.2.

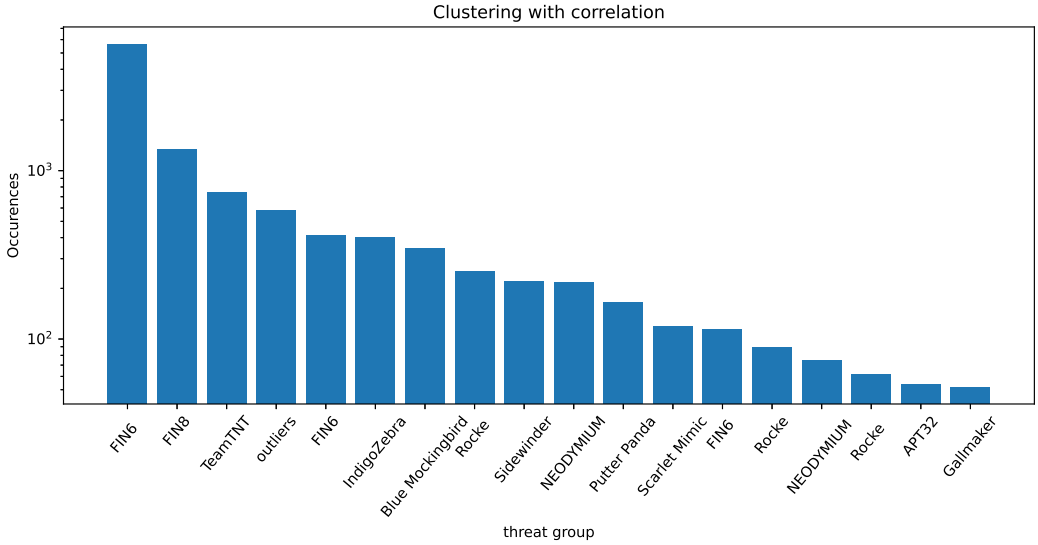


FIGURE 5.3 DBSCAN Clustering

Ensuite, on peut aller un pas plus loin et tenter de prédire les actions futures des attaquants en cherchant les techniques présentes dans le mode opératoire mais que l'on a pas encore observé dans nos données. Un exemple de ceci, pour l'un des regroupements, est montré au Tableau 5.3. Ceci correspond à un regroupement de 165 attaquants dont le mode opératoire se rapproche du groupe "Putter Panda" [78]. Nous pouvons donc nous attendre à observer ces techniques dans le futur : $T1055$, $T1057$, $T1071$, $T1083$ et $T1555$. Pour plus de détail sur ces techniques, le lecteur intéressé peut visiter le site web de MITRE ATT&CK [15]. Ces 5 techniques sont utilisées dans le mode opératoire, mais on ne les retrouve pas dans nos données. Par concision, nous n'incluons dans le Tableau 5.3 que les techniques utilisées soit par les attaquants, soit dans le mode opératoire.

Comme on peut le voir dans le Tableau 5.2, la plupart des similarités sont faibles (en dessous de 50%). Ceci peut être expliqué par le fait que ces attaquants sont considérés individuellement, basé sur les adresses IP. En pratique, les attaquants utilisent rarement une adresse IP unique lors de leurs attaques. Cela nous a conduits à créer l'algorithme 1 qui permet de corréler les attaquants ensemble lors de la phase de regroupement. Nous trouvons ainsi plusieurs attaquants et donc plusieurs adresses IP comme faisant partie d'une même attaque, ce qui renforce les similarités trouvées et résout ce problème. Les similarités résultantes sont

Tableau 5.2 Interpretation of DBSCAN clusters with MITRE ATT&CK threat groups

Cluster	**Number of attackers**	**Similarity**
FIN6	5646	0.33
FIN8	1335	0.25
TeamTNT	744	0.5
FIN6	413	0.14
IndigoZebra	402	0.37
Blue Mockingbird	345	0.33
Rocke	253	0.61
Sidewinder	222	0.67
NEODYMIUM	217	0.56
Putter Panda	165	0.45
Scarlet Mimic	119	0.33
FIN6	115	0.4
Rocke	89	0.57
NEODYMIUM	75	0.27
Rocke	62	0.31
APT32	54	0.55
Gallmaker	52	0.58

Tableau 5.3 Prediction of attackers' future actions

Technique	**Clustered Attackers**	**"Putter Panda"**
$T1027$: Obfuscated Files of Information	X	X
$T1055$: Process Injection		X
$T1057$: Process Discovery		X
$T1059$: Command and Scripting Interpreter	X	X
$T1070$: Indicator Removal	X	X
$T1071$: Application Layer Protocol		X
$T1082$: System Information Discovery	X	X
$T1083$: File and Directory Discovery		X
$T1547$: Boot or Logon Autostart Execution		X
$T1552$: Unsecured Credentials	X	X
$T1562$: Impair Defences	X	X

toutes au-delà de 0.6 et certaines arrivent même à 0.1, soit une similarité parfaite avec le mode opératoire considéré. Cette analyse fait l'objet de la Section suivante.

5.4.3 Analyse des attaquants par corrélation

Regrouper les attaquants en utilisant l'algorithme DBSCAN revient à ne les considérer que comme des données individuelles, sans faire de liens entre eux. Nous appliquons donc notre algorithme 1 pour trouver des corrélations entre les attaquants en même temps que le regroupement est calculé. Pour utiliser cet algorithme, nous utilisons les deux formules définies plus tôt pour calculer les corrélations entre attaquants et les similarités entre les attaquants et les modes opératoires.

Les résultats de cet algorithme sont présentés à la Figure 5.4. Pour obtenir ces résultats, nous avons mis le seuil de confiance ct à 0.8 et le seuil de similarité ft à 0.6. La valeur de ct est choisie pour être assez restrictive sur les corrélations que l'algorithme peut trouver. En effet, cela force l'algorithme à se concentrer sur les attaquants qui ont réalisés leur attaque en moins de 1.6 jours lors de recherches de corrélations. Cette valeur peut être trouvée en égalant l'équation 5.2 et en posant $m = 32$ (meilleur cas). De cette façon, l'algorithme regroupe les attaquants par 2 ou 3 en moyenne, avec les groupes les plus nombres étant formés de 5 attaquants. La valeur de ft a été décidée empiriquement, en commençant avec la valeur la plus haute de 1 et en la réduisant jusqu'à ce que l'algorithme puisse trouver certains résultats. Une procédure plus détaillée sur la manière de fixer ces valeurs est expliquée à la Section 5.5.

On peut observer à la Figure 5.4 que la plupart des attaquants ne sont pas regroupés (ils sont dans le groupe "unclustered"). Ceux-ci sont des attaquants où l'algorithme ne trouve pas de similarité suffisamment grande avec un mode opératoire, peu importe les corrélations effectuées. Avec les paramètres choisis ($ct = 0.8$ et $ft = 0.6$), l'algorithme ne parvient pas à effectuer de corrélations qui permettent de passer le seuil de similarité donné. Dans notre cas, nous utilisons des données Honeypot, qui ont la propriété de toutes être au moins suspicieuses, car le Honeypot n'a aucune fonctionnalité de production. À la suite d'un examen manuel, on observe que se trouvent dans ce groupe "unclustered", tous les attaquants qui n'ont pas vraiment effectué d'attaque. On y retrouve des utilisateurs qui se sont connectés au Honeypot et ont entré des commandes telles que "this is a test", ou simplement navigué dans la structure de dossiers à l'aide des commandes "ls" ou "cd". Dans un scénario réel, nous pensons donc que notre algorithme fera la distinction entre des utilisateurs légitimes qui se trouveront dans le groupe "unclustered" et des acteurs malveillants qui se trouvent dans les groupes trouvés.

Il est aussi important de mentionner que nous utilisons les informations reprises du cadriciel MITRE ATT&CK comme renseignement sur les diverses modes opératoires utilisés pour

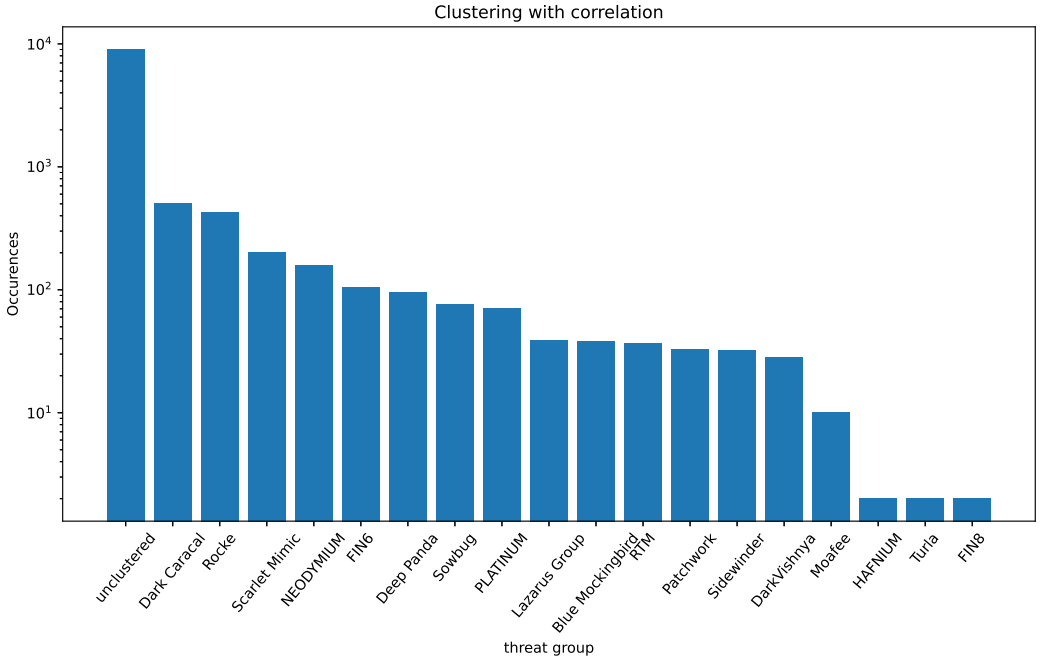


FIGURE 5.4 Clustering with Algorithm 1

pénétrer dans un réseau. Cependant, il est à noter que ces résultats ne signifient pas nécessairement que ces groupes d'attaques ont été observés, simplement que certains attaquants ont un mode opératoire similaire à ces groupes d'attaque.

5.5 Discussion

Comme expliqué dans la Section 5.3.1, la base de notre travail est de trouver une façon de représenter les attaquants par les techniques qu'ils utilisent. Pour y arriver, nous dérivons une liste associant les techniques aux commandes utilisées. Cette liste influence forcément nos résultats et une certaine attention doit y être portée. Nous avons décidé d'utiliser l'information comprise dans la description des techniques MITRE ATT&CK, mais il existe probablement d'autres façons de dériver cette liste qui pourraient être considérées. Comme nous considérons notre recherche comme une preuve de concept, nous avons décidé de garder la liste relativement simple. Cependant, même avec cette liste simple, nous pouvons corréler les attaquants et identifier des modes opératoires.

Comme montré dans la Section 5.4, il existe différentes façons d'utiliser l'information du cadriciel MITRE ATT&CK. D'une part, on peut étudier seulement les techniques utilisées par les attaquants, comme fait dans la Section 5.4.1 et plus en détail dans [28]. D'autre part,

on peut tenter d'associer les attaquants aux modes opératoires de façon individuelle, comme fait dans la Section 5.4.2. Enfin, on peut essayer d'induire des corrélations entre attaquants dictées par des liens sémantiques dérivés de ces modes opératoires, comme fait dans la Section 5.4.3.

Regarder les techniques donne une idée de ce que font la plupart des attaquants sur le réseau lorsqu'ils essaient de le pirater. Cela peut donner des conseils sur ce qu'il faut considérer lorsqu'on évalue les stratégies défensives des systèmes et des décisions d'achats.

Regrouper les attaquants individuellement demande de trouver un algorithme adapté aux particularités du jeu de données. Avec notre représentation des attaquants, nous utilisons l'algorithme DBSCAN, régulé par deux paramètres : ϵ, la distance entre deux points pour qu'ils soient dans le même groupe, et min_points, le nombre de points minimum qui doivent être proches pour qu'ils forment un groupe. Le nombre de groupe n'est pas un paramètre de cet algorithme, car il est induit des deux paramètres précédents. Le paramètre min_points est défini par la dimension des données, tel qu'expliqué dans [75]. Le paramètre ϵ a été dérivé à partir du graphe K-distance comme expliqué dans la Section 5.4.2 et montré dans la Figure 5.1. Pour confirmer ces choix, nous considérons aussi le score Silhouette et le score Davies Bouldin avec différentes valeurs de ϵ.

Le score Silhouette mesure la similarité des objets dans un même groupe. Il prend des valeurs dans l'intervalle $[-1, +1]$, où $+1$ est le meilleur score signifiant que tous les objets dans un même groupe sont identiques. Nous montrons les scores Silhouette calculés pour différentes valeurs de ϵ dans la Figure 5.5. On y voit que pour des valeurs très faibles de ϵ, le score Silhouette atteint sa meilleure valeur de 1. Cela fait du sens, car une valeur de 0 pour ϵ signifie que tous les attaquants dans un même groupe utilisent les mêmes techniques, et donc le même mode opératoire.

Le score Davies Bouldin mesure la séparation des groupes. Ces valeurs sont toujours positives avec 0 qui est le meilleur score. On voit dans la Figure 5.6 que des valeurs plus petites de ϵ résultent en une meilleure séparation des groupes étant donné que les scores Davies Bouldin sont proches de 0. Cela est donc conforme aux résultats obtenus avec le score Silhouette et avec le graphe K-distance.

Après avoir fait le regroupement, nous utilisons les modes opératoires comme des étiquettes pour effectuer une tâche d'attribution simple. Celle-ci fonctionne dans le cas où un attaquant réalise toute son attaque à partir d'une même adresse IP, mais reste limitée dans la majorité des cas. En effet, dans des scénarios réels, il est rare que des attaquants n'utilisent qu'une adresse IP. La plupart en utilisent plusieurs, et répartissent l'attaque sur un certain temps.

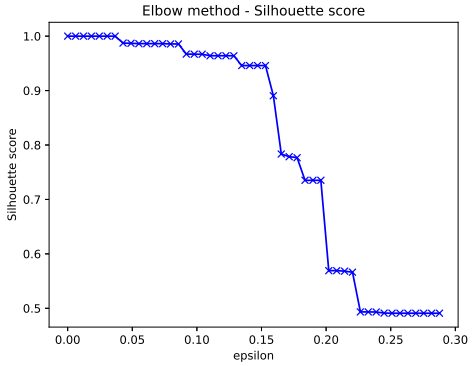


FIGURE 5.5 Silhouette score

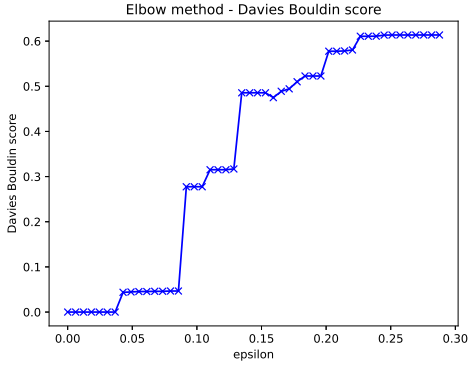


FIGURE 5.6 Davies Bouldin

En faisant de la corrélation d'attaquants, nous visons à régler ce problème d'adresses IP multiples. L'effort mis dans les tentatives de corrélation est régulé par notre paramètre de seuil de confiance (ct). En pratique, ce seuil, tout comme le seuil de similarité (ft), devrait être choisi judicieusement, et en fonction des résultats que l'on souhaite lors de l'utilisation de l'algorithme 1. La meilleure façon de régler ces paramètres serait par apprentissage supervisé, en comparant les résultats de l'algorithme à une base de vérité. Cependant, au meilleur de nos connaissances, il n'existe pas de jeu de données permettant de travailler sur l'attribution de façon supervisée actuellement. Dès lors, on peut également définir un intervalle de temps dans lequel les corrélations devraient être faites et trouver le seuil de corrélation correspondant à l'aide de l'équation 5.2. Ensuite, le seuil de similarité contrôle le nombre de pistes d'attribution en sortie de l'algorithme. Comme suggéré dans la Section 5.4.3, nous conseillons de le mettre à une valeur haute au départ et de le réduire jusqu'à obtenir un nombre de résultats satisfaisant.

Il est également à noter que l'algorithme ne remplace pas totalement le facteur humain dans la tâche d'attribution. Plutôt, il donne des conseils aux analystes, en fournissant un

support pour la tâche d'attribution. Il peut aider à automatiser des tâches lourdes et aider à concentrer l'attention des analystes vers les attaquants les plus alarmants. Il identifie des pistes prometteuses, qui devraient ensuite être considérées par les analystes qui peuvent ensuite modifier les paramètres au fur et à mesure de leurs investigations pour mieux diriger leur attention.

Enfin, nous utilisons des données Honeypot dans notre travail comme suggéré dans [13] mais il est important de garder en tête que les données collectées sont tout de même différentes de données réelles, étant donné que les Honeypot ne sont pas protégés de la même façon que les serveurs réels. Cependant, ces données réelles ne sont pas accessibles à des fins académiques. Tout de même, notre méthode basée sur les commandes entrées n'est pas limitée à la technologie utilisée pour la tester, soit les Honeypots. Elle peut être utilisée avec d'autres mécanismes de défense également, tel que des IDS.

5.6 Conclusion

Ce chapitre recherche comment le cadriciel MITRE ATT&CK peut être utilisé pour le problème de l'attribution à partir de logs Honeypot suspicieux. Nous utilisons des données collectées de Honeypots accessible publiquement sur Internet. Nous caractérisons d'abord les attaquants individuellement en nous basant sur les commandes entrées, puis corrélons et groupons ces attaquants par un algorithme créé dans ce travail. Nous démontrons comment les groupes d'attaquants trouvés peuvent être directement associés à des modes opératoires connus, publiés dans par MITRE. Notre approche peut être utilisée par des administrateurs réseau pour aider à automatiser certaines tâches et donner des conseils dans la tâche d'attribution.

CHAPITRE 6 DISCUSSION GÉNÉRALE

L'attribution est une tâche qui vise à caractériser les cyber attaquants plutôt que les attaques elles-mêmes. De par sa nature, l'espace cyber rend cette tâche complexe en créant un détachement entre les indicateurs observés, telles que les adresses IP, et les personnes physiques. L'attribution est alors généralement effectuée par des analystes, ce qui prend du temps et la rend sujette aux erreurs.

Actuellement, l'axe de recherche le plus étudié dans le domaine est celui de la détection. Celle-ci ne permet cependant que d'identifier certains comportements malveillant mais n'est généralement pas suffisante pour recréer le vecteur d'attaque au complet. Une meilleure compréhension des comportements adverses permettrait donc d'identifier plus précisément ces attaques.

Par ailleurs, ces mécanismes de détection présentent souvent des failles. D'un côté, les IDS basés sur signature et ceux basés sur l'apprentissage machine supervisé ne permettent pas l'identification de nouvelles attaques. De l'autre, ceux basés sur l'apprentissage machine non supervisé présentent de grand nombre de faux positifs, une faille étroitement liée au problème de fatigue d'alertes.

Ce travail vise donc à concevoir des outils qui aident les analystes dans leur travail. D'une part, il permet d'ordonnancer les différents évènements observés sur le réseau, afin de concentrer l'attention des analystes au bon endroit. D'autre part, il donne des pistes d'attribution en identifiant des modes opératoires observés sur le réseau. Ce chapitre présente premièrement une analyse de la valeur des données obtenues, puis une analyse sur les deux méthodologies adoptées et leurs pertinences avec les objectifs fixés.

6.1 Utilisation de données Honeypots

Étant donné le manque de jeu de données pour cette tâche, nous avons réalisé toutes nos expérimentations sur des données collectées par des solutions d'Honeypots, comme réalisé dans de nombreux travaux similaires. Même si ce sont actuellement les données les plus proches de cas réels que nous sommes capables de récolter, il est à noter qu'elles présentent tout de même quelques simplifications.

Premièrement, comme les Honeypots n'ont aucune fonctionnalité de production, nous savons que toutes les données récoltées sont malveillantes, ou suspicieuses tout au moins. Cette propriété est intéressante, car elle nous permet de clairement séparer les problèmes de détection

et d'attribution. Cependant, dans un contexte réel, les entreprises reçoivent également une grande partie de trafic légitime qu'il ne faut pas perturber. Ce point est à garder en tête lorsque l'on définit des outils d'attribution, ce que nous avons fait dans notre recherche. En effet, dans notre article qui traite de l'ordonnancement des évènements pour identifier les attaquants les plus dangereux, nous remarquons qu'une grande majorité (90%) des évènements se retrouvent dans 3 regroupement. Nous avons ensuite identifié manuellement que ces attaquants étaient les plus bénins, ceux-ci ne réalisant pas d'attaques très sérieuses. Il est alors naturel de penser que les clients légitimes, ayant généralement des comportements similaires, se retrouveront de la même façon dans un ou plusieurs regroupements jugés "innofensifs". Pour le deuxième objectifs de recherche, abordé dans le chapitre de résultats complémentaires, nous proposons un algorithme qui donne des pistes concrètes d'attribution en suggérant des modes opératoires définis dans la base de connaissance MITRE ATT&CK. La première version de cet algorithme proposait, pour chaque groupe d'attaquants, le meilleur mode opératoire trouvé. Cependant, dans certains cas, il était peu précis, par exemple lorsque le groupe d'attaquants trouvé effectuait peu d'opérations sur le Honeypot. C'est pourquoi nous avons ajouté, dans la version finale, des paramètres permettant de moduler les inférences de cet algorithme. En effet, dans cette version finale, l'analyste peut choisir un seuil de similarité et l'algorithme ne retournera que les évènements qui obtiennent une similarité avec le mode opératoire trouvé supérieur à ce seuil. Appliqué à un trafic légitime, l'algorithme ne trouvera alors pas de similarité suffisante entre ce trafic légitime et les modes opératoires considérés. En effet, certaines techniques définies dans le cadriciel MITRE ATT&CK pourraient être identifiables dans un trafic légitime, tel que les phases de reconnaissance comme la recherche de site web et de domaines accessibles. Cependant, la grande majorité des techniques est spécifique aux attaques telles que les exécutions arbitraires de code, ou l'élévation de privilèges. Les clients légitimes pourraient donc obtenir une similarité faible avec certains modes opératoires, mais rarement suffisante pour dépasser un seuil judicieusement choisi.

Deuxièmement, les solutions de Honeypots enregistrent le comportement des attaquants sur la machine ainsi que quelques indicateurs sur ces attaquants telles que des adresses IP, des noms de domaines ou encore dans certains cas une localisation géographique. Cependant, ces solutions ne peuvent pas induire d'informations concrètes sur les profils d'attaquants telles que des domaines d'expertise, la familiarité avec le réseau ou encore les ressources disponibles. Combiné à une absence de jeu de données étiquetées, il n'est donc actuellement pas possible d'évaluer l'efficacité précise des méthodes envisagées. En effet, les articles de recherche dans le domaine (ainsi que nos travaux), ne présentent pas de métriques telles que la précision, score F1 ou autres. Dès lors, les travaux actuels se positionnent plutôt comme des preuves de concept, des outils ayant pour but d'assister les analystes en automatisant les tâches les

plus fastidieuses, sans pour autant les remplacer totalement. C'est également dans ce but que nos travaux se situent, en examinant des travaux déjà réalisés et acceptés dans le domaine et en les poussant plus loin. Effectivement, notre article cherche à étendre et améliorer les travaux de Fraunholz D. et al. présentés dans [44, 45]. Les auteurs affirment avoir défini un modèle applicable à tous protocoles, mais nous nous sommes vite rendu compte que ce n'était pas le cas. C'est pourquoi nous l'avons d'abord généralisé puis étendu avec une analyse par regroupement, qui permet d'obtenir des résultats moins empiriques que leur proposition initiale. Cette approche peut ensuite être utilisée par un analyste pour découvrir rapidement les évènements les plus susceptibles d'être intéressants et éviter le problème de fatigue d'alerte. Pour les travaux complémentaires, nous nous sommes d'une part basés sur les travaux de Lim C. et al. [27, 28] et d'autre part sur ceux de Kyoungmin K. et al. [23] qui utilisent chacun le cadriciel MITRE ATT&CK dans leur analyse d'attaquants. Encore une fois, nous avons poussé leurs travaux plus loin en introduisant un nouvel algorithme permettant de corréler les attaquants et de donner des pistes d'attribution qui peuvent ensuite être confirmées par un analyste réseau. De plus, en réglant les paramètres de cet algorithme, l'analyste réseau peut facilement interagir avec l'algorithme en fonction de ses besoins.

Enfin, il est à retenir que les solutions de Honeypot proposées sont des solutions à interaction moyenne. Notre partenaire nous assure alors faire tout ce qu'il peut pour que cette solution de Honeypot soit la plus transparente possible du point de vue des attaquants. Il ne serait cependant pas sage de penser qu'aucun d'entre eux ne se rende compte qu'ils attaquent un Honeypot. Il est donc fort possible que l'on ne retrouve qu'une petite partie de certains vecteurs d'attaques dans nos données. Cependant, comme mentionné précédemment, utiliser un jeu de donnée issu de solutions Honeypot est ce que nous avons de plus réaliste actuellement.

6.2 Ordonnancer les logs Honeypots en fonction de la sévérité des attaquants

Le premier objectif de recherche était d'ordonnancer les évènements observés sur le réseau pour identifier ceux méritant d'être examinés plus amplement. Pour ce faire, nous avons créé 5 caractéristiques à partir d'éléments facilement observables dans tout type de logs, tels que le nombre de commandes ou l'intervalle de temps moyen entre les commandes. Le but était alors de définir un modèle applicable à tous protocoles, aussi bien IT que OT. Il est à noter que pour obtenir une représentation suffisamment intéressante des attaquants, nous avons considéré que deux logs venant de la même adresse IP en moins d'une heure appartenaient au même attaquant, comme fait dans d'autres travaux de recherche dans le même domaine. Il est alors possible qu'une perte d'information résulte de cette représentation si un attaquant conserve une adresse IP pendant plus d'une heure, sans spécialement envoyer de commandes

à intervalle régulier. Cependant, traiter chaque log individuellement ne donnait que trop peu d'informations sur les attaquants. Aussi, considérer que tous les logs provenant d'une même adresse IP proviennent du même attaquant parait exagéré aujourd'hui avec la pénurie et la réutilisation des adresses IPv4.

Une fois ces représentations d'attaquants obtenues, nous effectuons une analyse statistique sur chacune des caractéristiques individuellement et identifions les valeurs aberrantes. Celles-ci représentent des attaquants fort différents des autres et donc une première partie d'attaquants à examiner. Ensuite, une analyse par regroupement en utilisant l'algorithme Kmeans permet d'identifier 13 groupes d'attaquants, dont l'un d'entre eux nous semble plus préoccupants et qui constitue donc la deuxième partie des attaquants à examiner. Ensuite, un analyste pourrait envisager d'étudier les attaquants d'autres groupes identifiés. Nous avons utiliser l'algorithme Kmeans ici, car nous avions déjà écarté les valeurs aberrantes qui pourraient bouleverser l'algorithme. Nous nous retrouvons alors avec des vecteurs à 5 valeurs toutes entre 0 et 10, ce qui rend un algorithme basé sur des prototypes tels Kmeans performants.

L'analyse statistique et celle par regroupement nous donnent alors un ordonnancement des attaquants. Un analyste peut effectivement analyser les valeurs aberrantes en premier, suivi de chaque groupe identifié en ordre croissant par rapport au nombre d'attaquants qu'ils contiennent. Une telle méthodologie permettra à l'analyste de diriger son attention vers ces attaquants qui sont relativement différents du reste. En observant les groupes les plus denses à la fin, un analyste obtiendra une compréhension du comportement majoritaire sur son réseau.

6.3 Corréler les attaquants entre eux et identifier des modes opératoires pour chaque groupe identifié

Le deuxième objectif de recherche était de corréler les attaquants sur la base d'informations génériques et d'identifier des modes opératoires. Pour ce faire, nous avons choisi d'utiliser le cadriciel MITRE ATT&CK, base de connaissance de comportements adverses tirée d'observations réelles. Le choix de cette base de connaissance a été motivé par le fait qu'il s'agit d'une référence dans le domaine, utilisée par nombres de professionnels et entreprises [79]. Grâce à celle-ci, nous obtenons dans un premier temps de l'information sur des modes opératoires utilisés par différents groupes d'attaques. D'autre part, cela nous permet de créer une liste liant les commandes des attaquants observées sur le réseau aux différentes techniques définies. Ici encore, afin d'obtenir une représentation suffisamment intéressante des attaquants, nous avons considéré que deux commandes venant de la même adresse IP en moins d'une heure appartenaient au même attaquant. On peut alors discuter ce point de la même façon

que précédemment. Aussi, étant donné qu'il s'agit principalement d'une preuve de concept, la liste créée liant les commandes des attaquants aux techniques considérées est assez sommaire. À mesure que ces techniques sont exploitées sur des cas plus complexes, des experts du domaine pourraient prendre le temps de réaliser une liste plus exhaustive. Cependant, cette liste est suffisante dans notre étude pour montrer le potentiel de la méthode considérée.

À l'aide de ces attaquants et de ces modes opératoires, nous avons d'abord analysé les attaquants de manière individuelle. Étant donné que nous travaillons sur des vecteurs binaires de grande dimension, un algorithme de regroupement basé sur les densités est plus approprié. C'est pourquoi nous utilisons DBScan et identifions 17 groupes, ou ensembles de techniques utilisées fréquemment par les attaquants. On obtient aussi un groupe de bruit contenant des attaquants utilisant des ensembles de techniques différents. Il est également intéressant de constater que l'on retrouve dans ce groupe de bruit, la plupart des attaquants qui ont été identifiés comme malveillant par la première technique. Cependant, l'utilisation de DBScan de cette façon ne permet que de considérer les attaquants individuellement, nous avons donc conçu un algorithme permettant de corréler les attaquants entre eux et d'identifier les modes opératoires associés. C'est principalement cet algorithme qui répond à la deuxième question de recherche, en donnant des pistes d'attribution à un analyste qui pourra alors les confirmer ou infirmer. D'autre part, l'algorithme et ses résultats sont contrôlés par deux paramètres, qui peuvent être fixés par un analyste pour mieux répondre à ces besoins.

CHAPITRE 7 CONCLUSION

Dans le chapitre précédent, nous avons fait un récapitulatif des objectifs de recherche et expliqué comment nos méthodologies permettaient d'atteindre ces objectifs. Dans ce chapitre, nous concluons le travail réalisé en synthétisant les travaux effectués. Ensuite, nous expliquons les limitations des approches utilisées et finissons par introduire des pistes d'améliorations futures, objets de potentielles prochaines recherches dans le domaine.

7.1 Synthèse des travaux

Dans ce travail, nous nous sommes intéressés à la caractérisation des attaquants plutôt que de l'attaque elle-même. Pour ce faire, nous avons établi deux objectifs, qui étaient d'ordonnancer les attaquants observés dans le réseau en termes de sévérité, puis d'identifier des modes opératoires utilisés par ceux-ci. Pour atteindre ces objectifs, nous avons émis l'hypothèse qu'un attaquant conserve la même adresse IP pendant au moins une heure et nous avons formulé deux représentations de ces attaquants : en termes de 5 caractéristiques que nous avons définies, et en termes de techniques utilisées telles que décrites dans le cadriciel MITRE ATT&CK.

Pour répondre au premier objectif, nous avons effectué une analyse statistique sur les caractéristiques des attaquants. Nous sommes capables d'identifier ceux aux valeurs aberrantes, qui paraissent donc préoccupants et devraient être examinés plus amplement. Ensuite, une analyse par regroupement sur ces mêmes caractéristiques permet d'identifier 13 groupes d'attaquants dont 3 de ceux-ci contiennent plus de 90% des attaquants qui semblent être assez inoffensifs. Cela permet donc d'écarter une grande partie du réseau et facilite grandement la tâche d'un analyste réseau. Enfin, nous identifions également un de ces 13 groupes comme étant constitué des attaquants les plus malveillants que l'on trouve sur le réseau, qui tentent de s'y introduire de plusieurs manières différentes, en téléchargeant plusieurs logiciels malveillants et en tentant d'effacer leurs traces. En donnant la priorité aux attaquants aux valeurs aberrantes et à ceux du groupe le plus malveillant et en l'enlevant de ceux faisant partie des groupes inoffensifs, on obtient un ordonnancement tel que recherché.

Nous avons ensuite défini un algorithme permettant de corréler les attaquants sur la base des informations contenues dans la base de connaissances MITRE ATT&CK pour répondre au deuxième objectif de recherche. Les modes opératoires sont définis par une suite de techniques utilisées et les attaquants sont représentés par les techniques qu'ils utilisent selon les

observations faites sur le réseau. En sortie de cet algorithme, un analyste obtient des pistes d'attribution en fonction de la similarité entre les attaquants et les modes opératoires. De plus, deux paramètres peuvent être fixés par les analystes afin d'affiner les résultats à ses besoins.

7.2 Limitations de nos travaux

Comme expliqué dans le chapitre précédent, la principale limitation du travail nous vient des données. En effet, il n'existe actuellement pas de jeu de donnée en libre accès pour s'atteler à la problématique d'attribution. Nous utilisons donc un jeu de donnée généré par des solutions Honeypot, qui est la solution la plus fidèle au monde réel dont nous disposons actuellement. Cependant, avec de telles données, nous n'obtenons que du trafic malveillant, car aucun utilisateur légitime ne se connecte sur un Honeypot, ce qui peut biaiser les méthodes envisagées. Également, nous n'obtenons aucune information sur les attaquants eux-mêmes et tous les travaux du domaine en sont donc actuellement au stade de preuve de concept, sans réelle validation.

Aussi, il est à mentionner que nos solutions ne remplacent pas complètement les analystes réseau. Cependant, elles assistent les analystes en donnant des conseils, des pistes à suivre qui permettent aux analystes de concentrer leur attention plus efficacement. En effet, elles permettent d'automatiser les parties les plus fastidieuses de la tâche d'attribution de façon à ce que les analystes puissent investir leur temps et énergie plus efficacement.

7.3 Améliorations futures

Le principal axe de recherche qui nous paraît intéressant pour continuer dans ce domaine est la génération d'un jeu de données qui permettrait de tester et valider les différentes méthodologies introduites. Pour ce faire, plusieurs démarches sont envisageables et discutées, mais représentent un travail conséquent à elles seules.

Premièrement, on pourrait envisager de récupérer différents logiciels malveillants utilisés par les attaquants. En analysant manuellement ces logiciels, on identifierait différents taux de complexité et donc différents profils d'attaquants. Ensuite, en rejouant ces logiciels dans un environnement contrôlé, on pourrait obtenir un trafic malveillant, étiqueté par le logiciel qui a généré ce trafic et donc par le profil attribué à ce logiciel.

Une autre idée qui a été discutée pendant cette recherche est de faire appel à des compagnies de "bug bounty". Ces compagnies emploient des experts de sécurité pour tenter de pénétrer

dans des réseaux spécifiques. Ils possèdent donc des informations sur les experts employés tels que leur domaine d'expertises, leur expérience dans le domaine, ou leur proactivité pour trouver de nouvelles failles. Ces informations pourraient alors être utilisées pour définir différents profils d'attaquants et permettraient d'étiqueter le trafic généré par ces experts.

Une dernière piste pour créer un tel jeu de donnée serait d'organiser une compétition spécifique, un CTF ("Capture The Flag"). Dans ces compétitions, des joueurs ont accès à un environnement spécifique ou ils doivent tenter de s'introduire dans différents réseaux. En faisant remplir un formulaire, on pourrait alors recueillir des informations similaires que dans la solution de "bug bounty" et définir différents profils pour étiqueter le trafic ainsi généré.

Toutes ces solutions permettraient alors d'obtenir un jeu de données étiqueté. Cela permettrait d'une part de trier les différents travaux réalisés à ce jour en validant ou infirmant les résultats. D'autre part, cela permettrait d'ouvrir de nouvelles portes de recherches en considérant également des approches supervisées, qui sont absentes du domaine actuellement.

RÉFÉRENCES

[1] D. J. Bianco, "Pyramid of pain," http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html, 2014.

[2] Statista, "Number of internet and social media users worldwide as of january 2023," https://www.statista.com/statistics/617136/digital-population-worldwide/.

[3] Embroker, "2023 must-know cyber attack statistics and trends," https://www.embroker.com/blog/cyber-attack-statistics/.

[4] J. L. Leevy et T. M. Khoshgoftaar, "A survey and analysis of intrusion detection models based on cse-cic-ids2018 big data," *Journal of Big Data*, vol. 7, n°. 1, p. 1–19, 2020.

[5] A. Nisioti, A. Mylonas, P. D. Yoo et V. Katos, "From intrusion detection to attacker attribution : A comprehensive survey of unsupervised methods," *IEEE Communications Surveys & Tutorials*, vol. 20, n°. 4, p. 3369–3388, 2018.

[6] K. M. van den Dool (2018), "Cyber attribution : Problem solved ?" [Master's thesis, Universiteit Leiden]. https://studenttheses.universiteitleiden.nl/access/item%3A2663968/view.

[7] C. Kwon, W. Liu et I. Hwang, "Security analysis for cyber-physical systems against stealthy deception attacks," dans *2013 American control conference*. IEEE, 2013, p. 3344–3349.

[8] A. Teixeira, G. Dán, H. Sandberg et K. H. Johansson, "A cyber security study of a scada energy management system : Stealthy deception attacks on the state estimator," *IFAC Proceedings Volumes*, vol. 44, n°. 1, p. 11 271–11 277, 2011.

[9] N. C. Rowe et E. J. Custy, "Deception in cyber attacks," dans *Cyber warfare and cyber terrorism*. igi Global, 2007, p. 91–96.

[10] Z.-H. Pang, L.-Z. Fan, H. Guo, Y. Shi, R. Chai, J. Sun et G.-P. Liu, "Security of networked control systems subject to deception attacks : a survey," *International Journal of Systems Science*, vol. 53, n°. 16, p. 3577–3598, 2022.

[11] A. Bushby, "How deception can change cyber security defences," *Computer Fraud & Security*, vol. 2019, n°. 1, p. 12–14, 2019.

[12] E. Segal, "'alert fatigue' can lead to missed cyber threats and staff retention/recruitment issues : Study," https://www.forbes.com/sites/edwardsegal/2021/11/08/alert-fatigue-can-lead-to-missed-cyber-threats-and-staff-retentionrecruitment-issues-study/?sh=517164d935c9, 2021.

[13] E. Doynikova, E. Novikova et I. Kotenko, “Attacker behaviour forecasting using methods of intelligent data analysis : A comparative review and prospects,” *Information*, vol. 11, n°. 3, p. 168, 2020.

[14] Teradata, “What are the 5 v’s of big data ?” https://www.teradata.com/Glossary/What-are-the-5-V-s-of-Big-Data, April 2023.

[15] MITRE ATT&CK, https://attack.mitre.org/, Februrary 2023.

[16] “Mitre,” https://www.mitre.org, February 2023.

[17] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington et C. B. Thomas, “Mitre att&ck : Design and philosophy,” dans *Technical report*. The MITRE Corporation, 2018.

[18] F. Skopik et T. Pahi, “Under false flag : Using technical artifacts for cyber attack attribution,” *Cybersecurity*, vol. 3, p. 1–20, 2020.

[19] A. Warikoo, “Proposed methodology for cyber criminal profiling,” *Information Security Journal : A Global Perspective*, vol. 23, n°. 4-6, p. 172–178, 2014.

[20] U. Noor, Z. Anwar, T. Amjad et K.-K. R. Choo, “A machine learning-based fintech cyber threat attribution framework using high-level indicators of compromise,” *Future Generation Computer Systems*, vol. 96, p. 227–242, 2019.

[21] H. Kim, H. Kim *et al.*, “Comparative experiment on ttp classification with class imbalance using oversampling from cti dataset,” *Security and Communication Networks*, vol. 2022, 2022.

[22] L. Perry, B. Shapira et R. Puzis, “No-doubt : Attack attribution based on threat intelligence reports,” dans *2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE, 2019, p. 80–85.

[23] K. Kim, Y. Shin, J. Lee et K. Lee, “Automatically attributing mobile threat actors by vectorized att&ck matrix and paired indicator,” *Sensors*, vol. 21, n°. 19, p. 6522, 2021.

[24] “Joe sandbox,” https://www.joesecurity.org, February 2023.

[25] “Intezer,” https://www.intezer.com, February 2023.

[26] “Cuckoo sandbox,” https://cuckoosandbox.org, February 2023.

[27] C. Lim et K. E. Silaen, “Xt-pot : Exposing threat category of honeypot-based attacks,” dans *Proceedings of the 2021 International Conference on Engineering and Information Technology for Sustainable Industry*, 2020, p. 1–6.

[28] R. Djap, C. Lim, K. E. Silaen et A. Yusuf, “Xb-pot : Revealing honeypot-based attacker’s behaviors,” dans *2021 9th International Conference on Information and Communication Technology (ICoICT)*. IEEE, 2021, p. 550–555.

[29] M. Bada et J. R. Nurse, “Profiling the cybercriminal : a systematic review of research,” dans *2021 international conference on cyber situational awareness, data analytics and assessment (CyberSA)*. IEEE, 2021, p. 1–8.

[30] I. Mokube et M. Adams, “Honeypots : concepts, approaches, and challenges,” dans *Proceedings of the 45th annual southeast regional conference*, 2007, p. 321–326.

[31] A. M. Nasution, M. Zarlis et S. Suherman, “Analysis and implementation of honeyd as a low-interaction honeypot in enhancing security systems,” *Randwick International of Social Science Journal*, vol. 2, nº. 1, p. 124–135, 2021.

[32] G. Wicherski, “Medium interaction honeypots,” *German Honeynet Project*, 2006.

[33] M. Knöchel et S. Wefel, “Analysing attackers and intrusions on a high-interaction honeypot system,” dans *2022 27th Asia Pacific Conference on Communications (APCC)*. IEEE, 2022, p. 433–438.

[34] C. Moore et A. Al-Nemrat, “An analysis of honeypot programs and the attack data collected,” dans *Global Security, Safety and Sustainability : Tomorrow’s Challenges of Cyber Security : 10th International Conference, ICGS3 2015, London, UK, September 15-17, 2015. Proceedings 10*. Springer, 2015, p. 228–238.

[35] A. Sagala, “Automatic snort ids rule generation based on honeypot log,” dans *2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE)*. IEEE, 2015, p. 576–580.

[36] P. H. M. da Costa Ferreira et L. N. de Castro, “Extracting ids rules from honeypot data : A decision tree approach,” dans *The Proceedings of the International Conference in Information Security and Digital Forensics, Thessaloniki, Greece*, 2014.

[37] C.-B. Jiang, I.-H. Liu, Y.-N. Chung et J.-S. Li, “Novel intrusion prediction mechanism based on honeypot log similarity,” *International Journal of Network Management*, vol. 26, nº. 3, p. 156–175, 2016.

[38] P. Owezarski, “A near real-time algorithm for autonomous identification and characterization of honeypot attacks,” dans *Proceedings of the 10th ACM symposium on information, computer and communications security*, 2015, p. 531–542.

[39] B. Lingenfelter, I. Vakilinia et S. Sengupta, “Analyzing variation among iot botnets using medium interaction honeypots,” dans *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2020, p. 0761–0767.

[40] Cloudflare, “What is the mirai botnet ?” https://www.cloudflare.com/en-gb/learning/ddos/glossary/mirai-botnet/, April 2023.

[41] Y. Zhu, Z. Chen, Q. Yan, S. Wang, E. Li, L. Peng et C. Zhao, “Mining function homology of bot loaders from honeypot logs,” *arXiv preprint arXiv :2206.00385*, 2022.

[42] A. Bar, B. Shapira, L. Rokach et M. Unger, “Identifying attack propagation patterns in honeypots using markov chains modeling and complex networks analysis,” dans *2016 IEEE international conference on software science, technology and engineering (SWSTE)*. IEEE, 2016, p. 28–36.

[43] S. Mehta, D. Pawade, Y. Nayyar, I. Siddavatam, A. Tiwart et A. Dalvi, “Cowrie honeypot data analysis and predicting the directory traverser pattern during the attack,” dans *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*. IEEE, 2021, p. 1–4.

[44] D. Fraunholz, S. D. Anton et H. D. Schotten, “Introducing gamfis : A generic attacker model for information security,” dans *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. IEEE, 2017, p. 1–6.

[45] D. Fraunholz, D. Krohmer, S. D. Anton et H. D. Schotten, “Yaas-on the attribution of honeypot data.” *Int. J. Cyber Situational Aware.*, vol. 2, n°. 1, p. 31–48, 2017.

[46] T. Rid et B. Buchanan, “Attributing cyber attacks,” *Journal of strategic studies*, vol. 38, n°. 1-2, p. 4–37, 2015.

[47] I. Kotenko et A. Chechulin, “A cyber attack modeling and impact assessment framework,” dans *2013 5th International Conference on Cyber Conflict (CYCON 2013)*. IEEE, 2013, p. 1–24.

[48] E. Doynikova et I. Kotenko, “Enhancement of probabilistic attack graphs for accurate cyber security monitoring,” dans *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, 2017, p. 1–6.

[49] M. GhasemiGol, A. Ghaemi-Bafghi et H. Takabi, “A comprehensive approach for network attack forecasting,” *Computers & Security*, vol. 58, p. 83–105, 2016.

[50] R. Katipally, L. Yang et A. Liu, “Attacker behavior analysis in multi-stage attack detection system,” dans *Proceedings of the seventh annual workshop on cyber security and information intelligence research*, 2011, p. 1–1.

[51] T. Rashid, I. Agrafiotis et J. R. Nurse, “A new take on detecting insider threats : exploring the use of hidden markov models,” dans *Proceedings of the 8th ACM CCS International workshop on managing insider security threats*, 2016, p. 47–56.

[52] C. M. University, “Insider threat test dataset,” https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=508099, April 2023.

[53] S. Deshmukh, R. Rade, D. Kazi *et al.*, “Attacker behaviour profiling using stochastic ensemble of hidden markov models,” *arXiv preprint arXiv :1905.11824*, 2019.

[54] K. N. Mallikarjunan, S. M. Shalinie et G. Preetha, “Real time attacker behavior pattern discovery and profiling using fuzzy rules,” *Journal of Internet Technology*, vol. 19, n°. 5, p. 1567–1575, 2018.

[55] E. Nunes, P. Shakarian et G. I. Simari, “Toward argumentation-based cyber attribution,” dans *30th AAAI Conference on Artificial Intelligence, AAAI 2016*. AI Access Foundation, 2016, p. 177–184.

[56] E. Nunes, P. Shakarian, G. I. Simari et A. Ruef, “Argumentation models for cyber attribution,” dans *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 2016, p. 837–844.

[57] E. Karafili, A. C. Kakas, N. I. Spanoudakis et E. C. Lupu, “Argumentation-based security for social good,” *arXiv preprint arXiv :1705.00732*, 2017.

[58] E. Karafili, L. Wang, A. C. Kakas et E. Lupu, “Helping forensic analysts to attribute cyber-attacks : an argumentation-based reasoner,” dans *PRIMA 2018 : Principles and Practice of Multi-Agent Systems : 21st International Conference, Tokyo, Japan, October 29-November 2, 2018, Proceedings 21*. Springer, 2018, p. 510–518.

[59] E. Karafili, L. Wang et E. C. Lupu, “An argumentation-based reasoner to assist digital investigation and attribution of cyber-attacks,” *Forensic Science International : Digital Investigation*, vol. 32, p. 300925, 2020.

[60] “Cyber Security Breaches Survey 2022,” https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2022/cyber-security-breaches-survey-2022.

[61] A. Khraisat, I. Gondal, P. Vamplew et J. Kamruzzaman, “Survey of intrusion detection systems : techniques, datasets and challenges,” *Cybersecurity*, vol. 2, n°. 1, p. 1–22, 2019.

[62] A. L. Buczak et E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” *IEEE Communications Surveys & Tutorials*, vol. 18, n°. 2, p. 1153–1176, 2016.

[63] R. K. Goutam, “The problem of attribution in cyber security,” *International Journal of Computer Applications*, vol. 131, n°. 7, p. 34–36, 2015.

[64] “Michel Oosterhof (2022), Cowrie,” https://www.cowrie.org.

[65] M. Nawrocki, M. Wählisch, T. C. Schmidt, C. Keil et J. Schönfelder, “A survey on honeypot software and data analysis,” *arXiv preprint arXiv :1608.06249*, 2016.

[66] “VirusTotal (August 2022),” https://www.virustotal.com/gui/home/upload.

[67] “Symantec (August 2022),” https://securitycloud.symantec.com/cc/landing.

[68] “MITRE ATT&CK, Groups (August 2022),” https://attack.mitre.org/groups/.

[69] Mandiant, “The Majority of Business Cyber Security Decisions are Made Without Insight into the Attacker, According to New Mandiant Report.” https://www.mandiant.com/company/press-releases/mandiant-security-perspectives-report, 2023, February 13.

[70] M. Ester, H.-P. Kriegel, J. Sander et X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” dans *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD’96. AAAI Press, 1996, p. 226–231.

[71] A. Warikoo, “The triangle model for cyber threat attribution,” *Journal of Cyber Security Technology*, vol. 5, n°. 3-4, p. 191–208, 2021.

[72] P. S. Charan, P. M. Anand et S. K. Shukla, “Dmapt : Study of data mining and machine learning techniques in advanced persistent threat attribution and detection,” dans *Data Mining-Concepts and Applications*. IntechOpen, 2021.

[73] Y. Shin, K. Kim, J. J. Lee et K. Lee, “Art : automated reclassification for threat actors based on att&ck matrix similarity,” dans *2021 world automation congress (WAC)*. IEEE, 2021, p. 15–20.

[74] H. M. Soliman, G. Salmon, D. Sovilj et M. Rao, “Rank : Ai-assisted end-to-end architecture for detecting persistent attacks in enterprise networks,” *arXiv preprint arXiv :2101.02573*, 2021.

[75] T. Caliński et J. Harabasz, “A dendrite method for cluster analysis,” *Communications in Statistics-theory and Methods*, vol. 3, n°. 1, p. 1–27, 1974.

[76] R. K. Goutam, “The problem of attribution in cyber security,” *International Journal of Computer Applications*, vol. 131, n°. 7, p. 34–36, 2015.

[77] University of Cambridge, “Clever carl,” https://nrich.maths.org/2478, 2012.

[78] Mitre Attack, “Putter panda,” https://attack.mitre.org/groups/G0024/, February 2023.

[79] D. Walkowski, “Mitre attck : What it is, how it works, who uses it and why,” https://www.f5.com/labs/learning-center/mitre-attack-what-it-is-how-it-works-who-uses-it-and-why, 2021.