

Master 2 – Cybersécurité Formation en alternance

Année 2019-2020

Mémoire de fin d'études

Etude du modèle de menaces

« MITRE ATT&CK »

Cas d'utilisation au sein d'un SOC

Réalisé par : Oussama AZRARA

Maitre d'apprentissage : Javier GONZALEZ

Tuteur université: Abdelnahen SENOUSSAOUI

Résumé

Durant de nombreuses années, les équipes de sécurité des systèmes d'informations se basaient sur le cumul des technologies, outils et les différents flux d'informations sur les menaces afin de mieux se défendre. Cette approche est malheureusement peu efficace face à la hausse du nombre d'attaques et outils qui sont de plus en plus sophistiqués et plus simple à utiliser.

Les attaquants sont maintenant plus organisés, des fois même sponsorisés par les Etats politiques, économiques ou militaires. Les outils sont de nos jours très accessibles et facile à manipuler. Face à cela, un travail de renseignements et de veille doit être fait pour accélérer le processeur de détection et de réponse à incident.

MITRE ATT&CK (Adversarial Tactics, Techniques and Common Knowledge) est l'un des derniers Framework créés pour les renseignements sur les menaces. Il analyse d'une façon profonde le comportement des attaquants et leurs actions et fournit des informations précieuses pour les analystes sécurité.

Dans ce document, nous étudions et détaillons les différents cas d'utilisation de ce modèle au sein d'un SOC et les outils qui l'implémentent et surtout comprendre l'intérêt de son utilisation.

La première partie est une présentation générale de MITRE ATT&CK, de ses fonctions, de ses composants et des nouveautés apportées cette année 2020. Ensuite, nous allons comprendre le principe du modèle qu'est-ce qui fait qu'il se démarque des autres modèles.

La deuxième partie constitue le cœur du sujet ou nous allons détailler les cas d'utilisation d'ATT&CK et des outils à disposition pour chaque cas. En effet, pour ce chapitre nous allons voir que MITRE ATT&CK s'applique dans : **les renseignements sur les menaces, l'amélioration de la couverture défensive, la chasse aux menaces et les test d'intrusion**. La conclusion de ce chapitre est un diagramme de cas d'utilisation qui résume tout ce que nous avons vu.

Enfin, comme conclusion générale nous exposons les résultats de cette étude et comment concrètement les différents membres d'un SOC peuvent utiliser MITRE ATT&CK.

Table des matières

1	Introduction	6
1.1	Contexte et motivations	6
1.2	Contribution et organisation du rapport	6
2	Au cœur du modèle	7
2.1	Introduction	7
2.2	Présentation des composants du modèle	7
2.2.1	Relation entre tactiques et techniques	7
2.2.2	Les matrice détaillées	12
2.2.3	Navigateur ATT&CK	17
2.2.4	Introduction des sous-techniques	18
2.3	Conclusion	19
3	Pyramide de la douleur	20
3.1	Introduction	20
3.2	Pourquoi s'attaquer aux TTP	20
3.3	Conclusion	21
4	Cas d'utilisation de MITRE ATT&CK	21
4.1	Introduction	21
4.2	ATT&CK et les renseignements sur les menaces	22
4.2.1	Renseignements sur un group d'attaquants	23
4.2.2	Comparer les groupes	28
4.2.3	Mapper ses propres techniques	30
4.3	ATT&CK et détection	35
4.3.1	Connaitre sa couverture défensive	36
4.3.2	Identifier les lacunes	37
4.3.3	Prioriser	41
4.4	ATT&CK et chasse aux menaces	53
4.4.1	Chasse aux processus	57
4.4.2	Recherche basée sur les ordinateurs	58
4.4.3	Recherche basée sur les connexions réseau	60
4.4.4	Listes blanches	61
4.5	ATT&CK et les tests s'intrusion	62
4.6	Conclusion	65
5	Résultats et conclusion générale	66
6	Références	68
7	Glossaire	69
8	Annexes	70
8.1	Langage EQL	70
8.2	Code source du script Python : Extraction des techniques et sources d'information qui concernent un groupe d'attaquants	71
8.3	Code source du script Python : Etude statistiques sur les sources d'information et techniques	72

Table des figures

Figure 1 - Organisation des tactiques ATT&CK	7
Figure 2 - Exemple de tactique : "Exfiltration"	8
Figure 3 - Description d'une technique	8
Figure 4 - Exemple de procédure	9
Figure 5 - Exemple de mitigation	9
Figure 6 - Exemple de recommandation pour détection	9
Figure 7 - Organisation des tactiques et techniques dans une matrice	10
Figure 8 - Exemple de groupe APT	10
Figure 9 - Relations entre les composants ATT&CK.....	11
Figure 10 - Matrice PRE-ATT&CK.....	12
Figure 11 - Matrice pour Linux.....	13
Figure 12 - Matrice Cloud	13
Figure 13 - Matrice mobile	14
Figure 14 - Matrice ICS	15
Figure 15 - Groupes connus pour attaquer les ICS.....	15
Figure 16 - Vue sur les "Assets"	16
Figure 17- Navigateur ATT&CK.....	17
Figure 18 - Sous-techniques	18
Figure 19 - Regroupement des techniques	18
Figure 20 - Sous-techniques "Phishing"	19
Figure 21 - Pyramide de la douleur	20
Figure 22 - Cas d'utilisation d'ATT&CK	21
Figure 23 - Rechercher un groupe selon un contexte	23
Figure 24 - Présentation du groupe APT33.....	23
Figure 25 - Matrice des techniques du groupe APT33.....	24
Figure 26 - Différence entre les modèles de partage dans un serveur TAXII	25
Figure 27- Schématisation du principe du script Python 1	25
Figure 28 - Diagramme de Sankey pour les techniques de l'APT33	26
Figure 29 - Dendrogramme circulaire techniques APT33	27
Figure 30 - Résultat de la comparaison des groupe APT33 et Dragonfly.....	28
Figure 31 - Comparaison des groupes avec Neo4J	29
Figure 32 - Exemple d'extraction de techniques à partir d'un texte	30
Figure 33 - Page d'accueil de TRAM.....	30
Figure 34 - Analyse d'un article avec TRAM	31
Figure 35 - Résultat d'une extraction des techniques avec TRAM	32
Figure 36 - Exemple d'une matrice de détection	33
Figure 37 - Exemple de matrice montrant les lacunes dans la détection	34
Figure 38 - Echange entre analyste et équipe détection.....	34
Figure 39 - Etapes d'application d'ATT&CK dans la détection	35
Figure 40 - Logo de DeTT&CT	36
Figure 41 - Matrice couverture défensive 1	37
Figure 42 - Matrice couverture défensive 2	38
Figure 43 - Extrait d'un fichier d'administration YAML DeTT&CT	39
Figure 44 - Matrice couverture défensive 3	40
Figure 45 - Evolution du nombre de techniques détectées	40
Figure 46 - Identification des lacunes.....	41
Figure 47 - Matrice des techniques les plus utilisées.....	42
Figure 48 - Nombre de sources de données par technique	43
Figure 49 - Nombre de techniques couverte par source de données	44
Figure 50 - Nombre de techniques par sous-données	45
Figure 51 - Variation d'exécution d'une technique	45
Figure 52 - Hypothèse CAR	46
Figure 53 - Tactiques et techniques CAR	46

Figure 54 - Informations sur la plateforme ciblée	46
Figure 55 - Exemple de pseudocode CAR	47
Figure 56 - Test unitaire CAR	47
Figure 57 - Exemple de modèle CAR	48
Figure 58 - Exemple navigation CARET	48
Figure 59 - Logo de SIGMA	49
Figure 60 - Exemple de règle SIGMA	49
Figure 61 - Carte thermiques des techniques couvertes par les règles SIGMA	50
Figure 62 - Logo d'Atomic Red Teaming	51
Figure 63 - Exemple de test atomique	51
Figure 64 - Données qu'il faut pour détection Regsvr32	52
Figure 65 - Fichier de configuration Sysmon	54
Figure 66 - Logo de l'application "Threat Hunting"	54
Figure 67 - Techniques couvertes par l'application "Threat Hunting"	55
Figure 68 - Aperçu des déclencheurs	56
Figure 69 - Détails des évènements de la technique « Credential Access »	56
Figure 70 - Détails d'un GUID	57
Figure 71 - Exploration des évènements d'un ordinateur	58
Figure 72 - Evènements enregistrés sur une machine	59
Figure 73 - Tableau de bord des connexions réseau	60
Figure 74 - Tableau des listes blanches	61
Figure 75 - Gestion des agents CALDERA	62
Figure 76 - Création d'un réseau d'agents CALDERA	62
Figure 77 - Création d'un profil d'attaquant	63
Figure 78 - Lancement de la campagne	63
Figure 79 - Détails d'une opération	64
Figure 80 - Diagramme des cas d'utilisation de MITRE ATT&CK	65
Figure 81 - Implémentation de MITRE ATT&CK dans SIRP	66
Figure 82 - Création d'une règle de détection	67
Figure 83 - Langage EQL	70

1 Introduction

1.1 Contexte et motivations

Ce mémoire marque la fin de ma formation en Master 2 Cybersécurité en alternance au niveau de l'université de Paris (anciennement connue sous le nom de Paris Descartes) et de mon contrat d'apprentissage au sein du SOC d'Engie en tant qu'analyste SOC. Le thème a été suggéré par mon maître d'apprentissage M. GONZALEZ et validé par mon tuteur université M. SENOUSAOUI.

Le but de l'étude est d'apporter le maximum d'informations sur les cas d'utilisation au sein du SOC de l'un des derniers modèles créés pour les renseignements sur les menaces cyber appelé MITRE ATT&CK.

1.2 Contribution et organisation du rapport

L'étude commence par l'explication de la philosophie et le principe derrière le modèle de MITRE ATT&CK et les différentes entités qui le composent ainsi que les récentes nouveautés apportées.

Ensuite, on explique pourquoi faut-il s'attaquer aux comportements des attaquants. Après, on énumère d'une façon détaillées les cas d'utilisation du modèle et les multiples outils qui l'utilisent ainsi que les acteurs qui peuvent le manipuler et comment ils peuvent en tirer profit.

Enfin, on aura une conclusion générale qui présente les résultats de l'étude et la manière dont MITRE ATT&CK peut s'appliquer dans une équipe de sécurité opérationnelle.

Ce rapport s'appuie sur plusieurs articles, projets et conférences qui évoquent le sujet et représente bilan de tout ce qui peut se faire avec ce modèle.

2 Au cœur du modèle

2.1 Introduction

A la différence des autres modèles de menaces qui favorisent l'étude des IoC (Indicators of Compromise) statiques , MITRE ATT&CK est centré sur le comportement dynamique des attaquants et leur manière d'agir sur un système qu'ils ciblent.

Il permet de cartographier les détails des actions lancées par l'attaquant. La succession de ses actions, les erreurs commises et les traces laissées constituent une empreinte et donne une visibilité sur le profil de l'attaquant et aide à comprendre son mode opératoire et donc de prédire ses actions et trouver la manière la plus efficace pour le stopper .

ATT&CK fournit un modèle orienté sur les **TTP (tactiques, techniques et procédures)** et donne une vue plus approfondie des IoC traditionnellement utilisés dans la sécurité opérationnelle.

2.2 Présentation des composants du modèle

ATT&CK est surtout une base de connaissance sur les cyberattaques. Le modèle se concentre sur la façon dont les outils et les logiciels malveillants interagissent avec le système pendant une opération. Ces techniques sont organisées sous forme de différentes matrices spécialisées et conçues pour de multiples systèmes et contextes.

Les matrices comprennent plus de 200 techniques et outils utilisés dans la nature par les attaquants et vus dans le passé lors des cyberattaques. Elles sont organisées en un ensemble de **tactiques** et **techniques**.

2.2.1 Relation entre tactiques et techniques

La **tactique** représente le « Pourquoi ? » d'une **technique**. Elle est un objectif que l'adversaire cherche à atteindre dans le système ciblé. On peut distinguer 12 tactiques : **accès initial, exécution, persistance, escalade de priviléges, contournement et évasion des défenses, vol d'identifiants, exploration du SI (ou discovery), mouvements latéraux, collection de données, exfiltration, C&C (commande et contrôle) et impact.**

Ces tactiques sont organisées d'une façon chronologique de gauche à droite et représentent le cycle de vie d'un acteur malveillant dans un système.

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
----------------	-----------	-------------	----------------------	-----------------	-------------------	-----------	------------------	------------	---------------------	--------------	--------

Figure 1 - Organisation des tactiques ATT&CK – source : <https://medium.com/falconforce/the-att-ck-rainbow-of-tactics-5eb6f0bddebe>

Afin d'atteindre un objectif, l'attaquant utilise une ou plusieurs « **techniques** ». Il existe donc de nombreuses **techniques** dans chaque catégorie de tactique.

Par exemple, si l'on prend la tactique « **Exfiltration** », les attaquants utilisent généralement 10 différentes techniques pour exfiltrer les données d'un système d'information. En dessous de la tactiques, on a les noms de ces techniques :

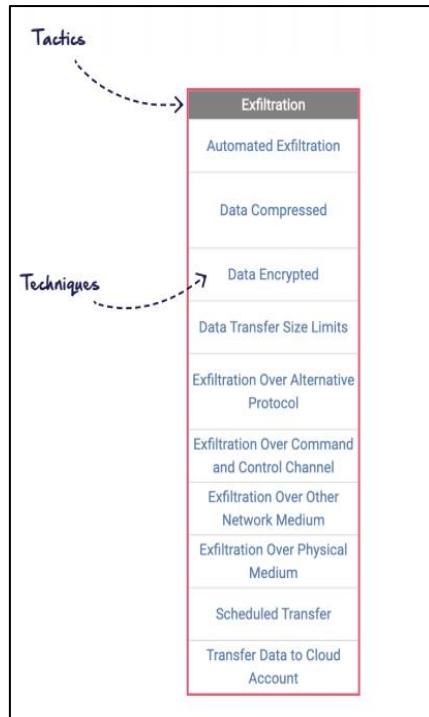


Figure 2 - Exemple de tactique : "Exfiltration"

Dans chaque détail d'une technique, on rencontre une **description**, des **exemples de procédures** pour réaliser la technique, des **recommandations** pour atténuer les risques, **méthodes de détection** et des **références** vers des articles sur le sujet.

Dans la description de la technique on retrouve un récapitulatif des informations qui concernent la technique : **ID (Identifiant sous la forme :T<CHIFFRE>)**, **nom de la tactique**, **plateformes visées**, **sources d'information requises pour détecter la technique**, **identifiant CAPEC (Common Attack Pattern Enumeration and Classification)**, **version**, **date de création** et **date de la dernière modification**.

Dans l'exemple suivant les informations qui concernent la technique « Spearphishing Attachment » :

ID: T1193
Tactic: Initial Access
Platform: Windows, macOS, Linux
Data Sources: File monitoring, Packet capture, Network intrusion detection system, Detonation chamber, Email gateway, Mail server
CAPEC ID: CAPEC-163
Version: 1.0
Created: 18 April 2018
Last Modified: 24 June 2019

Figure 3 - Description d'une technique

Les informations contenues dans « **Data Sources** » sont très importantes afin de connaître les capacités d'une organisation à détecter une technique en se basant sur les sources de d'information disponibles comme les alertes IDS/IPS.

Ces informations sont recueillies par un capteur ou un système de journalisation qui peut être utilisé pour identifier la séquence d'actions ou les résultats de ces actions par un adversaire. La liste des sources de données peut intégrer différentes variantes de la façon dont une action pourrait être effectuée pour une technique particulière. Cet attribut est destiné à être limité à une liste définie pour permettre l'analyse de la couverture des techniques en fonction des sources de données uniques. (Par exemple, "quelles techniques puis-je détecter si j'ai mis en place un contrôle des processus ?").

Les procédures se basent sur des événements déjà survenus qui montrent les multiples méthodes utilisées par certains groupes d'attaquants pour réaliser la technique et les outils employés.

Procedure Examples

Name	Description
admin@338	admin@338 has sent emails with malicious Microsoft Office documents attached. ^[92]
APT12	APT12 has sent emails with malicious Microsoft Office documents and PDFs attached. ^{[88][89]}
APT19	APT19 sent spearphishing emails with malicious attachments in RTF and XLSM formats to deliver initial exploits. ^[62]

Figure 4 - Exemple de procédure

Les groupes d'attaquants ont tendance à réaliser leurs actions d'une façon unique pour créer une marque spécifique au groupe. Donc une méthode d'accès peut constituer une signature qui contribue à l'identification de l'attaquant.

En plus des procédures, on retrouve les recommandations et les moyens qu'il faut mettre en place pour faire face à la technique et réduire l'impact en cas d'une attaque.

Mitigation	Description
Antivirus/Antimalware	Anti-virus can also automatically quarantine suspicious files.
Network Intrusion Prevention	Network intrusion prevention systems and systems designed to scan and remove malicious email attachments can be used to block activity.
Restrict Web-Based Content	Block unknown or unused attachments by default that should not be transmitted over email as a best practice to prevent some vectors, such as .scr, .exe, .pif, .cpl, etc. Some email scanning devices can open and analyze compressed and encrypted formats, such as zip and rar that may be used to conceal malicious attachments in Obfuscated Files or Information .
User Training	Users can be trained to identify social engineering techniques and spearphishing emails.

Figure 5 - Exemple de mitigation

Pour détecter une technique, il faut avoir en sa possession une ou plusieurs sources d'information. Dans la section « Détection » on a des recommandations sur les événements et informations qu'il faut superviser afin de prévenir une technique donnée.

Detection

Network intrusion detection systems and email gateways can be used to detect spearphishing with malicious attachments in transit. Detonation chambers may also be used to identify malicious attachments. Solutions can be signature and behavior based, but adversaries may construct attachments in a way to avoid these systems.

Anti-virus can potentially detect malicious documents and attachments as they're scanned to be stored on the email server or on the user's computer. Endpoint sensing or network sensing can potentially detect malicious events once the attachment is opened (such as a Microsoft Word document or PDF reaching out to the internet or spawning Powershell.exe) for techniques such as [Exploitation for Client Execution and Scripting](#).

Figure 6 - Exemple de recommandation pour détection

Les relations entre les tactiques et les techniques sont organisées dans différentes matrices ATT&CK. Les tactiques sont les noms des en-têtes de colonnes tandis que les techniques apparaissent sous chaque colonne.

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	Application Access Token	Bash History	Application Window Discovery	Application Access Token	Automated Collection	Communication Through Removable Media	Data Compressed	Data Destruction
External Remote Services	Command-Line Interface	Account Manipulation	AppCert DLLs	Binary Padding	Brute Force	Browser Bookmark Discovery	Application Deployment Software	Clipboard Data	Connection Proxy	Data Encrypted	Data Encrypted for Impact
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	BITS Jobs	Cloud Instance Metadata API	Cloud Service Dashboard	Component Object Model and Distributed COM	Data from Cloud Storage Object	Custom Command and Control Protocol	Data Transfer Size Limits	Defacement
Replication Through Removable Media	Component Object Model and Distributed COM	AppInit DLLs	Application Shimming	Bypass User Account Control	Credential Dumping	Cloud Service Discovery	Exploitation of Remote Services	Data from Information Repositories	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Content Wipe
Spearphishing Attachment	Control Panel Items	Application Shimming	Bypass User Account Control	Clear Command History	Credentials from Web Browsers	Domain Trust Discovery	Internal Spearphishing	Data from Local System	Data Encoding	Exfiltration Over Command and Control Channel	Disk Structure Wipe
Spearphishing Link	Dynamic Data Exchange	Authentication Package	DLL Search Order Hijacking	CMSTP	Credentials in Files	File and Directory Discovery	Logon Scripts	Data from Network Shared Drive	Data Obfuscation	Exfiltration Over Other Network Medium	Endpoint Denial of Service
Spearphishing via Service	Execution through API	BITS Jobs	Dylib Hijacking	Code Signing	Credentials in Registry	Network Service Scanning	Pass the Hash	Data from Removable Media	Domain Fronting	Exfiltration Over Physical Medium	Firmware Corruption
Supply Chain Compromise	Execution through Module Load	Bootkit	Elevated Execution with Prompt	Compile After Delivery	Exploitation for Credential Access	Network Share Discovery	Pass the Ticket	Data Staged	Domain Generation Algorithms	Scheduled Transfer	Inhibit System Recovery
Trusted Relationship	Exploitation for Client Execution	Browser Extensions	Emond	Compiled HTML File	Forced Authentication	Network Sniffing	Remote Desktop Protocol	Email Collection	Fallback Channels	Transfer Data to Cloud Account	Network Denial of Service
Valid Accounts	Graphical User Interface	Change Default File Association	Exploitation for Privilege Escalation	Component Firmware	Hooking	Password Policy Discovery	Remote File Copy	Input Capture	Multi-hop Proxy		Resource Hijacking

Figure 7 - Organisation des tactiques et techniques dans une matrice

En plus des tactiques et techniques, ATT&CK dispose des informations détaillées sur **93** groupes d'attaquants APT (Advanced Persistent Threat), leurs historiques d'attaques, leur façon d'opérer ainsi que des renseignements sur les différents outils et malware utilisés.

Ces informations sont utiles pour comprendre et documenter les profils des groupes d'attaquants dans une perspective de détection comportementale indépendante des outils que le groupe utilise.

APT1

APT1 is a Chinese threat group that has been attributed to the 2nd Bureau of the People's Liberation Army (PLA) General Staff Department's (GSD) 3rd Department, commonly known by its Military Unit Cover Designator (MUCD) as Unit 61398.^[1]

ID: G0006
Associated Groups: Comment Crew, Comment Group, Comment Panda
Version: 1.1
Created: 31 May 2017
Last Modified: 20 August 2019

Associated Group Descriptions

Name	Description
Comment Crew	[1]
Comment Group	[1]
Comment Panda	[5]

Techniques Used

Domain	ID	Name	Use	ATT&CK® Navigator Layers
PRE-ATT&CK	T1330	Acquire and/or use 3rd party software services	APT1 used third party email services in the registration of whois records. ^[1]	
PRE-ATT&CK	T1334	Compromise 3rd party infrastructure to support delivery	APT1 compromised a vast set of 3rd party victim hop points as part of their network infrastructure. ^[1]	
PRE-ATT&CK	T1326	Domain registration hijacking	APT1 hijacked FQDNs associated with legitimate websites hosted by hop points. Mandiant considers them to be "hijacked" since they were originally registered for a legitimate reason but are used by APT1 for malicious purposes. ^[1]	
PRE-ATT&CK	T1333	Dynamic DNS	APT1 used dynamic DNS to register hundreds of FQDNs. ^[1]	
PRE-ATT&CK	T1346	Obtain/re-use payloads	APT1 used publicly available privilege escalation tools. ^[1]	
Enterprise	T1087	Account Discovery	APT1 used the commands <code>net localgroup</code> , <code>net user</code> , and <code>net group</code> to find accounts on the system. ^[1]	

Figure 8 - Exemple de groupe APT

Tous les composants du modèle sont liés : Les groupes d'attaquants utilisent des techniques et des outils qui sont nécessaires pour la réalisation d'une tactique.

Les techniques sont détectables grâce aux différentes sources de données disponibles. Le schéma suivant résume les relations entre les composants :

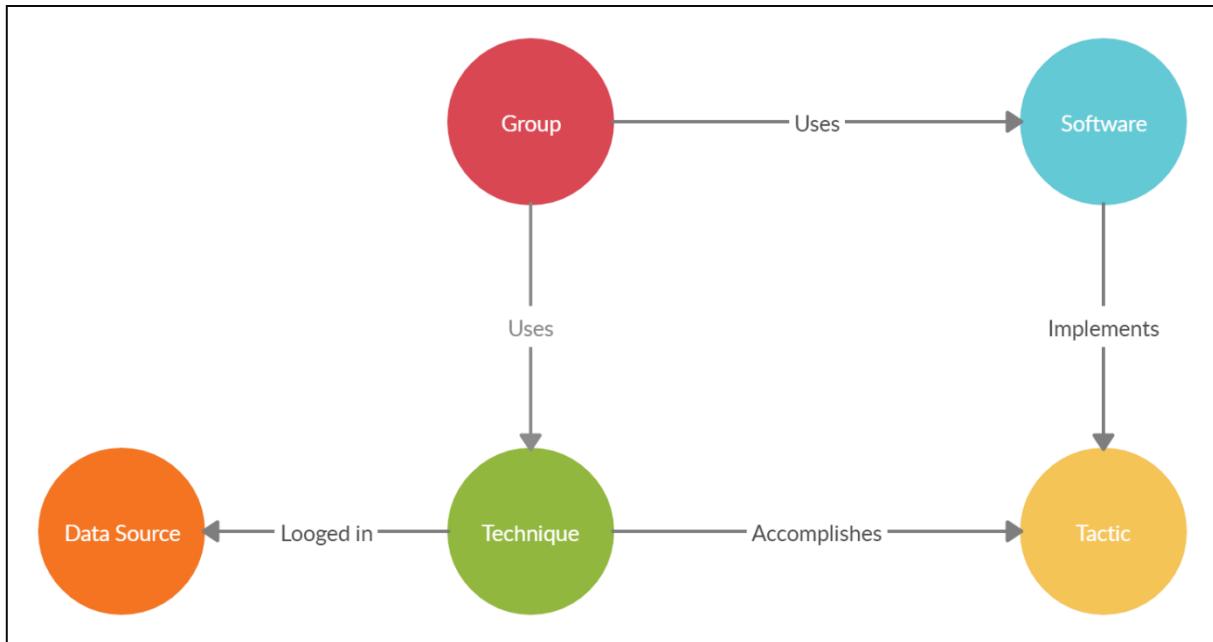


Figure 9 - Relations entre les composants ATT&CK

Enfin, on note que certaines techniques viseront exclusivement une plateforme particulière. En prenant l'exemple la techniques qui consiste à ajouter une clé d'exécution dans le registre Windows, qui entraînera l'exécution du programme au démarrage de la machine (**T1060 Registry Run keys**), on constatera que cette technique est dédiée exclusivement aux systèmes qui tournent sous Windows. Donc pour cela, MITRE a mis en place plusieurs matrices dédiées pour différentes plateformes et contextes (Windows, Linux, Cloud etc.).

Ces matrices sont en développement continu et se spécialisent de plus en plus.

2.2.2 Les matrice détaillées

Toutes les informations collectées sur les attaques sont présentées dans différentes matrices qui sont :

- **PRE-ATT&CK**
- **Entreprise**
- **Mobile**
- **ICS (industrial control system)**



2.2.2.1 PRE-ATT&CK

La première matrice est une collection de **15** catégories de tactiques axées sur les comportements qui précédent une attaque.

Cette matrice a pour but d'aider à prévenir une attaque avant que l'adversaire n'ait une chance de pénétrer dans un réseau. Par exemple, lorsqu'un potentiel attaquant lance une reconnaissance active ou passive d'un domaine ou d'une adresse IP.

Priority Definition Planning	Priority Definition Direction	Target Selection	Technical Information Gathering	People Information Gathering	Organizational Information Gathering	Technical Weakness Identification	People Weakness Identification	Organizational Weakness Identification	Adversary OPSEC	Establish & Maintain Infrastructure	Persons Development	Build Capabilities	Test Capabilities	Stage Capabilities
Assess current holdings, needs, and wants	Assign KITs, KIQs, and/or intelligence requirements	Determine approach/attack vector	Acquire OSINT data sets and information	Acquire OSINT data sets and information	Analyze application security posture	Analyze organizational skillsets and deficiencies	Analyze business processes	Acquire and/or use 3rd party infrastructure services	Acquire and/or use 3rd party infrastructure services	Build social network persona	Build and configure delivery systems	Review logs and residual traces	Disseminate removable media	
Assess KITs/KIQs benefits	Receive KITs/KIQs and determine requirements	Determine highest level tactical element	Conduct active scanning	Aggregate individual's digital footprint	Conduct social engineering	Analyze architecture and configuration posture	Analyze social and business relationships, interests, and affiliations	Analyze organizational skillsets and deficiencies	Acquire and/or use 3rd party software services	Acquire and/or use 3rd party software services	Choose pre-compromised mobile app developer account credentials or signing keys	Build or acquire exploits	Test ability to evade automated mobile application security analysis performed by app stores	Distribute malicious software development tools
Assess leadership areas of interest	Submit KITs, KIQs, and intelligence requirements	Determine operational element	Conduct passive scanning	Conduct social engineering	Determine 3rd party infrastructure services	Analyze data collected	Assess targeting options	Analyze presence of outsourced capabilities	Acquire or compromise 3rd party signing certificates	Acquire or compromise 3rd party signing certificates	Choose pre-compromised persona and affiliated accounts	C2 protocol development	Test callback functionality	Friend/Follow/Connect to targets of interest
Assign KITs/KIQs into categories	Task requirements	Determine secondary level tactical element	Conduct social engineering	Identify business relationships	Determine centralization of IT management	Analyze hardware/software security defensive capabilities					Develop social network persona digital footprint	Compromise 3rd party or closed-source vulnerability/exploit information	Test malware in various execution environments	Hardware or software supply chain implant
Conduct cost/benefit analysis	Create implementation plan	Determine strategic target	Determine 3rd party infrastructure services	Identify groups/roles	Determine physical locations	Analyze organizational skillsets and deficiencies						Create custom payloads	Test malware to evade detection	Port redirector
			Determine domain and IP address space	Identify job postings and needs/gaps	Dumpster dive	Identify vulnerabilities in third-party software libraries								
Create strategic plan		Determine external network trust dependencies	Identify people of interest	Identify business processes/tempo	Research relevant vulnerabilities/CVEs									Upload, install, and configure software/tools

Figure 10 - Matrice PRE-ATT&CK

2.2.2.2 Entreprise

Cette matrice est la pièce maîtresse du modèle. Il s'agit d'un sur-ensemble des matrices **Windows, Linux, MacOs** et, récemment ajoutée, la matrice dédiée au **Cloud**.

Elle contient 245 techniques qui sont mises à jour régulièrement avec les dernières et plus grandes techniques d'attaque que les pirates et les chercheurs découvrent dans la nature.

Enterprise
Windows
macOS
Linux
Cloud
AWS
GCP
Azure
Office 365
Azure AD
SaaS

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Drive-by Compromise	Command-Line Interface	.bash_profile and .bashrc	Exploitation for Privilege Escalation	Binary Padding	Bash History	Account Discovery	Application Deployment Software	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal
Exploit Public-Facing Application	Exploitation for Client Execution	Bootkit	Process Injection	Clear Command History	Brute Force	Browser Bookmark Discovery	Exploitation of Remote Services	Automated Collection	Communication Through Removable Media	Data Compressed	Data Destruction
Hardware Additions	Graphical User Interface	Browser Extensions	Setuid and Setgid	Compile After Delivery	Credential Dumping	File and Directory Discovery	Internal Spearphishing	Clipboard Data	Connection Proxy	Data Encrypted	Data Encrypted for Impact
Spearphishing Attachment	Local Job Scheduling	Create Account	Sudo	Connection Proxy	Credentials from Web Browsers	Network Service Scanning	Remote File Copy	Data from Information Repositories	Custom Command and Control Protocol	Data Transfer Size Limit	Defacement
Spearphishing Link	Scripting	Hidden Files and Directories	Sudo Caching	Disabling Security Tools	Credentials in Files	Network Sniffing	Remote Services	Data from Local System	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Content Wipe
Spearphishing via Service	Source	Kernel Modules and Extensions	Valid Accounts	Execution Guardrails	Exploitation for Credential Access	Password Policy Discovery	SSH Hijacking	Data from Network Shared Drive	Data Encoding	Exfiltration Over Command and Control Channel	Disk Structure Wipe
Supply Chain Compromise	Space after Filename	Local Job Scheduling	Web Shell	Exploitation for Defense Evasion	Input Capture	Permission Groups Discovery	Third-party Software	Data from Removable Media	Data Obfuscation	Exfiltration Over Other Network Medium	Endpoint Denial of Service
Trusted Relationship	Third-party Software	Port Knocking		File and Directory Permissions Modification	Network Sniffing	Process Discovery		Data Staged	Domain Fronting	Exfiltration Over Physical Medium	Firmware Corruption
Valid Accounts	Trap	Redundant Access		File Deletion	Private Keys	Remote System Discovery		Input Capture	Domain Generation Algorithms	Scheduled Transfer	Inhibit System Recovery
	User Execution	Server Software Component		Hidden Files and Directories	Steal Web Session Cookie	Software Discovery		Screen Capture	Fallback Channels		Network Denial of Service
		Setuid and Setgid		HISTCONTROL	Two-Factor Authentication Interception	System Information Discovery			Multi-hop Proxy		Resource Hijacking
		Systemd Service		Indicator Removal from Tools		System Network Configuration Discovery			Multi-Stage Channels		Runtime Data Manipulation
		Trap		Indicator Removal on Host		System Network Connections Discovery			Multiband Communication		Stored Data Manipulation
		Valid Accounts		Install Root Certificate		System Owner/User Discovery			Multilayer Encryption		System Shutdown/Reboot
		Web Shell		Masquerading					Port Knocking		Transmitted Data Manipulation
				Obfuscated Files or Information					Remote Access Tools		
				Port Knocking					Remote File Copy		
				Process Injection					Standard Application Layer Protocol		
				Redundant Access					Standard Cryptographic Protocol		
				Rootkit					Standard Non-Application Layer Protocol		
				Scripting					Uncommonly Used Port		
				Space after Filename					Web Service		
				Timestamp							
				Valid Accounts							
				Web Service							

Figure 11 - Matrice pour Linux

La matrice cloud contient à son tour 6 sous-matrices : AWS, GCP, Azure, Office 365, Azure AD et SaaS.

Initial Access	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Impact
Drive-by Compromise	Account Manipulation	Valid Accounts	Application Access Token	Account Manipulation	Account Discovery	Application Access Token	Data from Cloud Storage Object	Transfer Data to Cloud Account	Resource Hijacking
Exploit Public-Facing Application	Create Account		Redundant Access	Brute Force	Cloud Service Dashboard	Internal Spearphishing	Data from Information Repositories		
Spearphishing Link	Implant Container Image		Revert Cloud Instance	Cloud Instance Metadata API	Cloud Service Discovery	Web Session Cookie	Data from Local System		
Trusted Relationship	Office Application Startup		Unused/Unsupported Cloud Regions	Credentials in Files	Network Service Scanning		Data Staged		
Valid Accounts	Redundant Access		Valid Accounts	Steal Application Access Token	Network Share Discovery		Email Collection		
	Valid Accounts		Web Session Cookie	Steal Web Session Cookie	Permission Groups Discovery				
					Remote System Discovery				
					System Information Discovery				
					System Network Connections Discovery				

Figure 12 - Matrice Cloud

2.2.2.3 Mobile

La matrice mobile couvre les techniques impliquant l'accès aux appareils et les effets de réseau qui peuvent être utilisés par les attaquants sans accès aux appareils. La matrice contient des informations pour les deux plateformes mobiles les plus utilisées : **Android** et **iOS**.



Initial Access	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Impact	Collection	Exfiltration	Command and Control		
Deliver Malicious App via Authorized App Store	Abuse Device Administrator Access to Prevent Removal	Exploit OS Vulnerability	Application Discovery	Access Notifications	Application Discovery	Attack PC via USB Connection	Clipboard Modification	Access Calendar Entries	Alternate Network Mediums	Alternate Network Mediums		
Deliver Malicious App via Other Means	App Auto-Start at Device Boot	Exploit TEE Vulnerability	Device Lockout	Access Sensitive Data in Device Logs	Evade Analysis Environment	Exploit Enterprise Resources	Data Encrypted for Impact	Access Call Log	Commonly Used Port	Commonly Used Port		
Drive-by Compromise	Modify Cached Executable Code	Exploit via Charging Station or PC	Disguise Root/Jailbreak Indicators	Access Stored Application Data	File and Directory Discovery		Delete Device Data	Access Contact List	Data Encrypted	Domain Generation Algorithms		
Exploit via Charging Station or PC	Modify OS Kernel or Boot Partition		Download New Code at Runtime	Android Intent Hijacking	Location Tracking		Device Lockout	Access Notifications	Standard Application Layer Protocol	Standard Application Layer Protocol		
Exploit via Radio Interfaces	Modify System Partition		Evade Analysis Environment	Capture Clipboard Data	Network Service Scanning		Generate Fraudulent Advertising Revenue	Access Sensitive Data in Device Logs	Standard Cryptographic Protocol	Standard Cryptographic Protocol		
Install Insecure or Malicious Configuration	Modify Trusted Execution Environment		Input Injection	Capture SMS Messages	Process Discovery		Input Injection	Access Stored Application Data	Uncommonly Used Port	Uncommonly Used Port		
Lockscreen Bypass	Masquerade as Legitimate Application		Install Insecure or Malicious Configuration	Exploit TEE Vulnerability	System Information Discovery		Manipulate App Store Rankings or Ratings	Capture Audio	Web Service	Web Service		
Masquerade as Legitimate Application			Modify OS Kernel or Boot Partition	Input Capture	System Network Configuration Discovery		Modify System Partition	Capture Camera				
Supply Chain Compromise			Modify System Partition	Input Prompt	System Network Connections Discovery		Premium SMS Toll Fraud	Capture Clipboard Data				
			Modify Trusted Execution Environment	Network Traffic Capture or Redirection				Capture SMS Messages				
			Obfuscated Files or Information	URL Scheme Hijacking				Data from Local System				
			Suppress Application Icon					Input Capture				
								Location Tracking				
								Network Information Discovery				
								Network Traffic Capture or Redirection				
								Screen Capture				

Network-Based Effects

Last Modified: 2019-10-24 08:29:36.078906

Network Effects	Remote Service Effects
Downgrade to Insecure Protocols	Obtain Device Cloud Backups
Eavesdrop on Insecure Network Communication	Remotely Track Device Without Authorization
Exploit SS7 to Redirect Phone Calls/SMS	Remotely Wipe Data Without Authorization
Exploit SS7 to Track Device Location	
Jamming or Denial of Service	
Manipulate Device Communication	
Rogue Cellular Base Station	
Rogue Wi-Fi Access Points	
SIM Card Swap	

Figure 13 - Matrice mobile

2.2.2.4 ICS

ATT&CK pour ICS est une base de connaissance pour décrire les actions qu'un attaquant peut prendre lorsqu'il opère au sein d'un réseau ICS. Cette base est utilisée pour mieux caractériser et décrire le comportement de l'attaquant après une compromission.

Initial Access	Execution	Persistence	Evasion	Discovery	Lateral Movement	Collection	Command and Control	Inhibit Response Function	Impair Process Control	Impact
Data Historian Compromise	Change Program State	Hooking	Exploitation for Evasion	Control Device Identification	Default Credentials	Automated Collection	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Drive-by Compromise	Command-Line Interface	Module Firmware	Indicator Removal on Host	I/O Module Discovery	Exploitation of Remote Services	Data from Information Repositories	Connection Proxy	Alarm Suppression	Change Program State	Denial of Control
Engineering Workstation Compromise	Execution through API	Program Download	Masquerading	Network Connection Enumeration	External Remote Services	Select Operating Mode	Standard Application Layer Protocol	Block Command Message	Masquerading	Denial of View
Exploit Public-Facing Application	Graphical User Interface	Project File Infection	Rogue Master Device	Network Service Scanning	Program Organization Units	Delect Program State		Block Reporting Message	Modify Control Logic	Loss of Availability
External Remote Services	Man in the Middle	System Firmware	Rootkit	Network Sniffing	Remote File Copy	I/O Image		Block Serial COM	Modify Parameter	Loss of Control
Internet Accessible Device	Program Organization Units	Valid Accounts	Spoof Reporting Message	Remote System Discovery	Valid Accounts	Location Identification		Data Destruction	Module Firmware	Loss of Productivity and Revenue
Replication Through Removable Media	Project File Infection		Utilize/Change Operating Mode	Serial Connection Enumeration		Monitor Process State		Denial of Service	Program Download	Loss of Safety
Spearphishing Attachment	Scripting					Point & Tag Identification		Device Restart/Shutdown	Rogue Master Device	Loss of View
Supply Chain Compromise	User Execution					Program Upload		Manipulate I/O Image	Service Stop	Manipulation of Control
Wireless Compromise						Role Identification		Modify Alarm Settings	Spoof Reporting Message	Manipulation of View
						Screen Capture		Modify Control Logic	Unauthorized Command Message	Theft of Operational Information
								Program Download		
								Rootkit		
								System Firmware		
								Utilize/Change Operating Mode		

Figure 14 - Matrice ICS

Le noyau du MITRE ATT&CK pour ICS est la matrice qui fournit une vue d'ensemble des TTP associés aux acteurs de la menace qui ont mené des attaques contre les systèmes ICS comme l'APT33, OilRig ou encore Dragonfly.

Group	Associated Groups	Description
ALLANITE	Palmetto Fusion ALLANITE	ALLANITE is a suspected Russian cyber espionage group, that has primarily targeted the electric utility sector within the United States and United Kingdom. The group's tactics and techniques are reportedly similar to Dragonfly / Dragonfly 2.0, although ALLANITE's technical capabilities have not exhibited disruptive or destructive abilities. It has been suggested that the group maintains a presence in ICS for the purpose of gaining understanding of processes and to maintain persistence. ^[1]
APT33	APT33 EIRN MAGNALLIUM	APT33 is a suspected Iranian threat group that has carried out operations since at least 2013. The group has targeted organizations across multiple industries in the United States, Saudi Arabia, and South Korea, with a particular interest in the aviation and energy sectors. ^[2]
Dragonfly	Energetic Bear Dragonfly	Dragonfly is a cyber espionage group that has been active since at least 2011. They initially targeted defense and aviation companies but shifted to focus on the energy sector in early 2013. They have also targeted companies related to industrial control systems. ^{[3][4][5]} A similar group emerged in 2015 and was identified by Symantec as Dragonfly 2.0. There is debate over the extent of the overlap between Dragonfly and Dragonfly 2.0, but there is sufficient evidence to lead to these being tracked as two separate groups. ^[3]
Dragonfly 2.0	Dragonfly 2.0 Beserk Bear DYMALLOY	Dragonfly 2.0 is a suspected Russian threat group which has been active since at least late 2015. Dragonfly 2.0's initial reported targets were a part of the energy sector, located within the United States, Switzerland, and Turkey. ^[6] There is debate over the extent of overlap between Dragonfly 2.0 and Dragonfly, but there is sufficient evidence to lead to these being tracked as two separate groups. ^[7]
HEXANE	Lyceum HEXANE	HEXANE is a threat group that has targeted ICS organization within the oil & gas, and telecommunications sectors. Many of the targeted organizations have been located in the Middle East including Kuwait. HEXANE's targeting of telecommunications has been speculated to be part of an effort to establish man-in-the-middle capabilities throughout the region. HEXANE's TTPs appear similar to APT33 and OilRig but due to differences in victims and tools it is tracked as a separate entity. ^[8]
Lazarus group	Guardians of Peace Lazarus group COVELLITE HIDDEN COBRA ZINC	Lazarus group is a suspected North Korean adversary group that has targeted networks associated with civilian electric energy in Europe, East Asia, and North America. ^{[9][10]} Links have been established associating this group with the WannaCry ransomware from 2017. ^[11] While WannaCry was not an ICS focused attack, Lazarus group is considered to be a threat to ICS. North Korean group definitions are known to have significant overlap, and the name Lazarus Group is known to encompass a broad range of activity. Some organizations use the name Lazarus Group to refer to any activity attributed to North Korea. ^[9] Some organizations track North Korean clusters or groups such as Bluehorror, APT37, and APT38 separately, while other organizations may track some activity associated with those group names by the name Lazarus Group.
Leafminer	Leafminer RASPITE	Leafminer is a threat group that has targeted Saudi Arabia, Japan, Europe and the United States. Within the US, Leafminer has targeted electric utilities and initial access into those organizations. ^{[12][13]} Reporting indicates that Leafminer has not demonstrated ICS specific or destructive capabilities. ^[13]
OilRig	OilRig Greenbug APT 34 CHRYSENE	OilRig is a suspected Iranian threat group that has targeted the financial, government, energy, chemical, and telecommunication sectors as well as petrochemical, oil & gas. ^{[14][15][16]} OilRig has been observed operating in Iraq, Pakistan, Israel, and the UK, and has been linked to the Sharmoon attacks in 2012 on Saudi Aramco.
Sandworm	Sandworm ELECTRUM	Sandworm is a threat group associated with the Kiev, Ukraine electrical transmission substation attacks which resulted in the impact of electric grid operations on December 17th, 2016. ^{[17][18]} Sandworm has been cited as the authors of the Industroyer malware which was used in the 2016 Ukraine attacks. ^[19]
XENOTIME	TEMPVeles XENOTIME	XENOTIME is a threat group that has targeted and compromised industrial systems, specifically safety instrumented systems that are designed to provide safety and protective functions. Xenotime has previously targeted oil & gas, as well as electric sectors within the Middle east, Europe, and North America. Xenotime has also been reported to target ICS vendors, manufacturers, and organizations in the middle east. This group is one of the few with reported destructive capabilities. ^[20]

Figure 15 - Groupes connus pour attaquer les ICS

La base de connaissances des attaques ICS comprend une catégorie "Assets" qui pourrait être utilisée par les organisations pour mieux classer le type de menaces qui pourraient avoir un impact sur les ressources de leur environnement.

Name	Description
Control Server	A device which acts as both a server and controller, that hosts the control software used in communicating with lower-level control devices in an ICS network (e.g. Remote Terminal Units (RTUs) and Programmable Logic Controllers (PLCs)). ^[1]
Data Historian	A centralized database located on a computer installed in the control system DM2 supporting external corporate user data access for archival and analysis using statistical process control and other techniques. ^[2]
Engineering Workstation	The engineering workstation is usually a high-end very reliable computing platform designed for configuration, maintenance and diagnostics of the control system applications and other control system equipment. The system is usually made up of redundant hard disk drives, high speed network interface, reliable CPUs, performance graphics hardware, and applications that provide configuration and monitoring tools to perform control system application development, compilation and distribution of system modifications. ^[3]
Field Controller/RTU/PLC/IED	Controller terminology depends on the type of system they are associated with. They provide typical processing capabilities. Controllers, sometimes referred to as Remote Terminal Units (RTU) and Programmable Logic Controllers (PLC), are computerized control units that are typically rack or panel mounted with modular processing and interface cards. The units are colocated with the process equipment and interface through input and output modules to the various sensors and controlled devices. Most utilize a programmable logic-based application that provides scanning and writing of data to and from the I/O interface modules and communicates with the control system network via various communications methods, including serial and network communications. ^[4]
Human-Machine Interface	In computer science and human-computer interaction, the Human-Machine Interface (HMI) refers to the graphical, textual and auditory information the program presents to the user (operator) using computer monitors and audio subsystems, and the control sequences (such as keystrokes with the computer keyboard, movements of the computer mouse, and selections with the touchscreen) the user employs to control the program. Currently the following types of HMI are the most common: Graphical user interfaces(GUI) accept input via devices such as computer keyboard and mouse and provide articulated graphical output on the computer monitor. Web-based user interfaces accept input and provide output by generating web pages which are transported via the network and viewed by the user using a web browser program. The operations user must be able to control the system and assess the state of the system. Each control system vendor provides a unique look-and-feel to their basic HMI applications. An older, not gender-neutral version of the term is man-machine interface (MMI).
Input/Output Server	The Input/Output (I/O) server provides the interface between the control system LAN applications and the field equipment monitored and controlled by the control system applications. The I/O server, sometimes referred to as a Front-End Processor (FEP) or Data Acquisition Server (DAS), converts the control system application data into packets that are transmitted over various types of communications media to the end device locations. The I/O server also converts data received from the various end devices over different communications mediums into data-formatted to communicate with the control system networked applications. ^[5]
Safety Instrumented System/Protection Relay	A safety instrumented system (SIS) takes automated action to keep a plant in a safe state, or to put it into a safe state, when abnormal conditions are present. The SIS may implement a single function or multiple functions to protect against various process hazards in your plant. ^[6] The function of protective relaying is to cause the prompt removal from service of an element of a power system when it suffers a short circuit or when it starts to operate in any abnormal manner that might cause damage or otherwise interfere with the effective operation of the rest of the system. ^[7]

Figure 16 - Vue sur les "Assets"

Cette base comprend actuellement 10 acteurs de la menace, 81 techniques d'attaque, 17 familles de logiciels malveillants et 7 types d'actifs (Assets).

Il y a plusieurs raisons de créer une matrice ATT&CK distincte pour les ICS :

Premièrement, les adversaires ciblent de nouveaux objectifs dans le domaine des ICS. Les trois objectifs généralement de la sécurité de l'information sont représentés par la triade **Confidentialité, Authentification, Intégrité et Disponibilité** (C.A.I.D), qui donne la priorité à la confidentialité avant les deux autres objectifs.

Dans le domaine de la sécurité des ICS, la priorité est donnée au processus industriel, car l'ensemble de l'environnement ICS n'est construit que pour maintenir le processus industriel en vie et en fonctionnement efficace. Par conséquent, la disponibilité et l'intégrité sont plus importantes, ce qui entraîne un changement d'objectifs pour les attaquants.

La deuxième raison est que le domaine ICS diffère en matière de défense et de technologie. Les dispositifs sont souvent des plateformes embarquées qui utilisent des protocoles d'automatisation de processus spécialisés et une grande variété d'applications ICS sont en fonctionnement.

Les difficultés dans cet environnement sont que la sécurité et la collecte de données pour la surveillance de la sécurité n'étaient pas au centre des préoccupations lors du développement et de l'intégration des dispositifs et des applications. En outre, les mesures d'atténuation sont confrontées à des préoccupations particulières, car la priorité des propriétaires d'actifs est la disponibilité du processus industriel.

Enfin, une attaque sur le domaine des SCI a des conséquences spécifiques. En plus de l'indisponibilité potentielle des processus et des opérations, le contrôle des processus pourrait être perdu, ce qui entraînerait des problèmes de sécurité.

2.2.3 Navigateur ATT&CK

Le navigateur ATT&CK est conçu pour fournir une navigation et une annotation de base des matrices ATT&CK, ce que les gens font déjà aujourd'hui avec des outils comme Excel.

L'outil permet notamment de : **visualiser la couverture défensive, la planification des équipes rouge/bleue, la fréquence des techniques détectées ou tout autre chose que l'on veut faire**. Le navigateur permet simplement de manipuler les cellules de la matrice (codage couleur, ajout d'un commentaire, attribution d'une valeur numérique, etc.).

La principale caractéristique du Navigateur est la possibilité pour les utilisateurs de définir des couches - des vues personnalisées de la base de connaissances ATT&CK - par exemple en montrant uniquement les techniques pour une plateforme particulière ou en mettant en évidence les techniques qu'un adversaire spécifique est connu pour utiliser.

Les couches peuvent être créées de manière interactive dans le navigateur ou générées par programme sous format JSON(JavaScript Object Notation) ou Excel et ensuite visualisées via le navigateur.

L'application peut être utilisée soit directement en ligne avec le lien :
<https://mitre-attack.github.io/attack-navigator/enterprise/>

Ou installée localement en téléchargeant son répertoire GIT et en installant les modules prérequis:

<https://github.com/mitre-attack/attack-navigator>

Exemple d'utilisation : Visualisation des techniques utilisées par le groupe d'attaquants russe connu sous le nom de TA505 (Threat Actor 505) :

MITRE ATT&CK® Navigator												
Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact	
11 items	28 items	44 items	23 items	60 items	18 items	23 items	16 items	13 items	21 items	9 items	16 items	
Drive-by Compromise	CMSTP	Accessibility Features	Access Token Manipulation	Account Manipulation	Account Discovery	Application Deployment Software	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal		
Exploit Public-Facing Application	Command-Line Interface	Account Manipulation	Accessibility Features	Binary Padding	Brute Force	Application Window Discovery	Automated Collection	Communication Through Removable Media	Data Compressed	Data Destruction		
External Remote Services	Compiled HTML File	AppCert DLLs	AppCert DLLs	BITS Jobs	Credential Dumping	Browser Bookmark Discovery	Clipboard Data	Data from Information Repositories	Data Encrypted	Data Transferred		
Hardware Additions	Component Object Model and Distributed COM	AppInit DLLs	AppInit DLLs	Bypass User Account Control	Credentials from Web Browsers	Domain Trust Discovery	Data from Local System	Custom Command and Control Protocol	Data Transfer Size Limits	Defacement		
Replication Through Removable Media	Control Panel Items	Application Shimming	Application Shimming	CMSTP	Credentials in Files	File and Directory Discovery	Exploitation of Remote Services	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocols	Disk Content Wipe		
Spearphishing Attachment	Dynamic Data Exchange	Authentication Package	Code Signing	Credentials in Registry	Network Service Scanning	Internal Spearphishing	Logon Scripts	Data from Network Shared Drive	Exfiltration Over Command and Control Channel	Disk Structure Wipe		
Spearphishing Link	Execution through API	BITS Jobs	Bypass User Account Control	Compile After Delivery	Network Share Discovery	Network Sniffing	Pass the Hash	Data from Removable Media	Data Encoding	Endpoint Denial of Service		
Spearphishing via Service	Execution through Module Load	Bootkit	DLL Search Order Hijacking	Compiled HTML File	Forced Authentication	>Password Policy Discovery	Pass the Ticket	Data from Local Protocol	Data Obfuscation	Firmware Corruption		
Supply Chain Compromise	Exploitation for Client Execution	Browser Extensions	Change Default File Association	Component Firmware	Hooking	Peripheral Device Discovery	Remote Desktop Protocol	Data Staged	Domain Fronting	Inhibit System Recovery		
Trusted Relationship	Graphical User Interface	Component Firmware	Extra Window Memory Injection	Input Capture	Permission Groups Discovery	Process Discovery	Remote File Copy	Email Collection	Domain Generation Algorithms	Network Denial of Service		
Valid Accounts	InstallUtil	Component Object Model Hijacking	Connect Proxy	Input Prompt	Remote Services	Remote Services	Input Capture	Fallback Channels	Exfiltration Over Physical Medium	Resource Hijacking		
	LSASS Driver	Create Account	Control Panel Items	Kerberos	Query Registry	Man in the Browser	Input Capture	File and Stage Channels	Scheduled Transfer	Runtime Data Manipulation		
	Mshta	DLL Search Order Hijacking	Hooking	DCShadow	LLMNR/NBT-NS Poisoning and Relay	Replication Through Removable Media	Input Capture	Multi-hop Proxys	Service Stop			
	PowerShell	Image File Execution Options Injection	Image File Execution Options Injection	Deobfuscate/Decode Files or Information	Remote System Discovery	Screen Capture	Input Capture	Multi-Stage Channels	Stored Data Manipulation			
	Regovz/Regasm	External Remote Services	New Service	Disabling Security Tools	Security Software Discovery	Shared Webcam	Video Capture	Multiband Communication	System Shutdown/Reboot			
	Regsv32	File System Permissions Weakness	Parent PID Spoofing	DLL Search Order Hijacking	Software Discovery	Taint Shared Content	Third-party Software	Multilayer Encryption	Transmitted Data Manipulation			
	Rundll32	Hidden Files and Directories	DLL Side-Loading	Private Keys	System Information Discovery	Windows Admin Shares	Remote Access Tools	Remote File Copy				
	Scheduled Task	Port Monitor	Path Interception	Steal Web Session Cookie	System Network Configuration Discovery	Windows Remote Management	Standard Application Layer Protocol					
	Scripting	PowerShell Profile	Execution Guardrails	Two-Factor Authentication Interception	System Network Connections Discovery		Standard Cryptographic Protocol					
	Service Execution	Hypervisor	Exploration for Defense Evasion	System Owner/User Discovery			Standard Non-Application Layer Protocol					
	Signed Binary Proxy Execution	Process Injection	Extra Window Memory Injection	File and Directory Permissions Modification	System Service Discovery		Uncommonly Used Port					
	Signed Script Proxy Execution	Scheduled Task	File Deletion	File System Logical Offsets	System Time Discovery		Web Service					
	Third-party Software	Service Registry Permissions Weakness	SID-History Injection	Group Policy Modification	Virtualization/Sandbox Evasion							
	Trusted Developer Utilities	Modify Existing Service	Valid Accounts	Hidden Files and Directories								
		Ntsh Helper DLL	Web Shell									

Figure 17- Navigateur ATT&CK

Le résultat est téléchargeable sous format JSON, Excel ou SVG. JSON est le format de fichier standard de gestion des techniques dans le navigateur. Ce format permettra de charger des données déjà existantes pour les visualiser, d'automatiser la transmission des techniques et de générer un nouveau fichiers avec les techniques sélectionnées.

2.2.4 Introduction des sous-techniques

En Mars 2020, MITRE a annoncé la publication d'une nouvelle forme de matrices du modèle avec des **sous-techniques** (sub-techniques). Cette version est encore phase de test et elle remplacera le modèle actuel.

Désormais à deux niveaux, les techniques et sous-techniques sont présentées en arborescence.

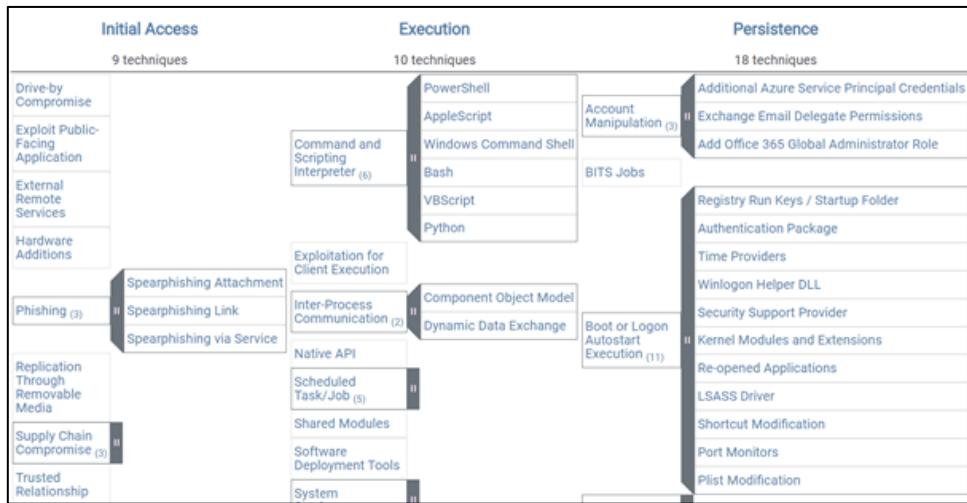


Figure 18 - Sous-techniques

Les nouvelles sous-techniques apportent une meilleure précision dans la qualification des TTP employées par un attaquant.

En prenant l'exemple de la technique « **Valid Accounts** », on s'aperçoit qu'actuellement elle regroupe des procédures un peu éloignées les unes des autres ayant comme point commun l'obtention ou l'utilisation d'un compte utilisateur pour s'introduire ou rester dans le système.

Avec le nouveau modèle, cette technique comprend 4 sous-techniques qui regroupent ce comportement : **Default accounts**, **Domain accounts**, **Local accounts** et **Cloud accounts**.

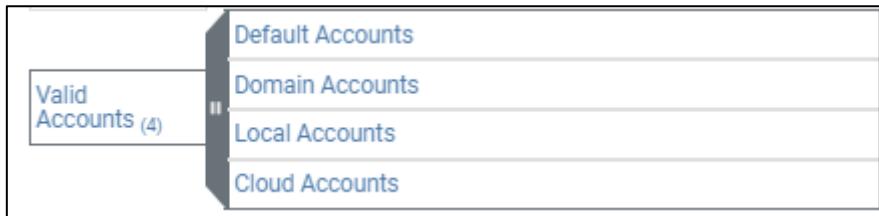


Figure 19 - Regroupement des techniques

Avant ce changement, l'évaluation de la couverture de cette technique était biaisée car nous pouvons avoir différents types de comptes dans un système d'information chacun avec son niveau de criticité et qui nécessite des surveillances différentes.

Certaines techniques ont été clustérées et groupées dans des nouvelles techniques. Comme le cas de la nouvelle techniques « Phishing » qui comportent les sous-techniques (anciennement techniques) : **Spearphishing Link**, **Spearphishing Attachment** et **Spearphising via Service**.

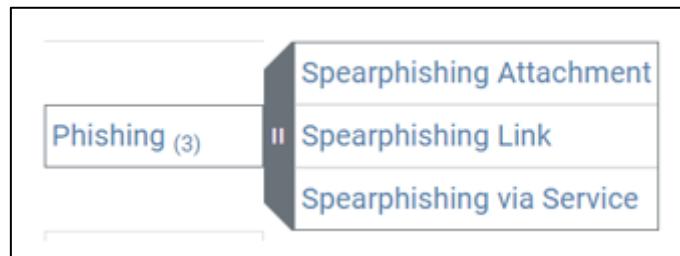


Figure 20 - Sous-techniques "Phishing"

Ce regroupement liant plusieurs techniques très proches les unes des autres rend plus simple la cartographie des modes opératoires dans la grille ATT&CK et permet d'envisager plus simplement les variantes à certaines techniques.

L'apparition des techniques « parent » dans une chaîne d'exécution d'une attaque permet de prévoir les différentes possibilités s'offrant à un attaquant et de mieux localiser d'éventuels Signle Point of Attack et repositionner des techniques qui étaient exclusives les unes des autres comme alternatives les unes aux autres.

Ces changements ne sont pas sans impact, les ID et noms des techniques ont changé et de nouvelles relations sont apparues. Cela va impacter plusieurs Framework et outils qui doivent revoir leurs implémentations avant de pouvoir intégrer le nouveau modèle.

2.3 Conclusion

Dans ce chapitre, les éléments qui composent la base de connaissance ATT&CK ont été définis et décrits. De plus, on a constaté que de grands changements sont en cours sur le modèle pour le rendre plus précis avec l'arrivée des sous-techniques et la réorganisation et regroupement de certaines techniques sous différents clusters.

Etant donné la concentration du modèle sur les TTP, il est primordiale de connaître les raisons et le principe derrière cette philosophie.

3 Pyramide de la douleur

3.1 Introduction

Ce qui rend MITRE ATT&CK formidable, c'est que toutes les tactiques, techniques et procédures sont basées sur ce qui a été observé comme attaques réalisées par des groupes d'attaquants réels. Beaucoup de ces groupes utilisent les mêmes techniques d'attaque. C'est presque comme si les groupes avaient leurs propres manuels de tactiques (playbook) lorsqu'ils ciblent des systèmes et qu'ils utilisent ce manuel pour rendre leurs nouveaux membres productifs rapidement.

Ce qui amène à étudier la pyramide de la douleur (Pyramid of Pain) .

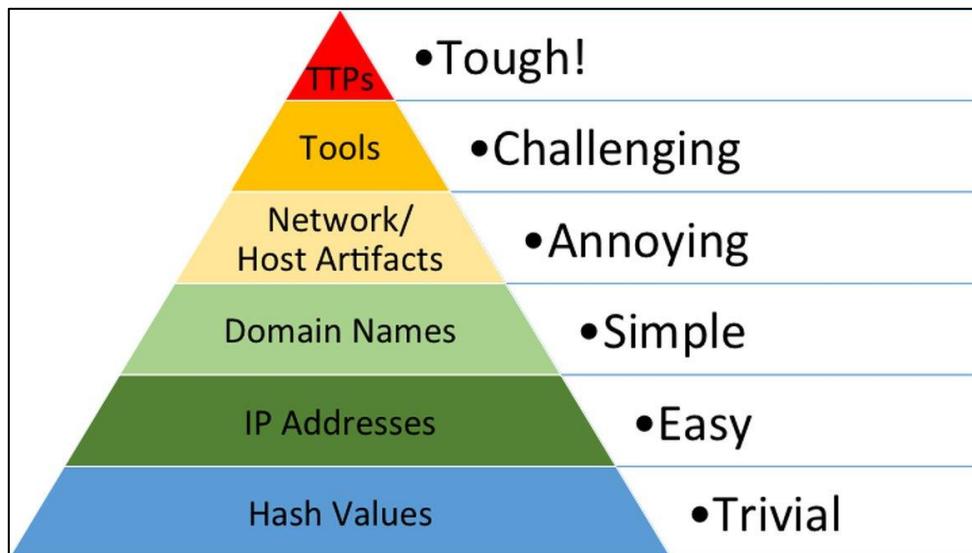


Figure 21 - Pyramide de la douleur - source :
<https://attackiq.com/blog/2019/06/26/emulating-attacker-activities-and-the-pyramid-of-pain/>

3.2 Pourquoi s'attaquer aux TTP

Cette pyramide nous apprend que de nombreux indicateurs de compromission (IoC) et indicateurs d'attaque (IoA) - qui sont sous les TTP dans la pyramide - sont similaires aux méthodes basées sur les signatures.

En basant la détection sur l'un des niveaux inférieurs au TTP, on combat les outils des attaquants. Ainsi, ils peuvent facilement contourner l'un des indicateurs en recompilant les outils ou en modifiant un fichier de configuration.

Exemple : Générer un hash différent ou changer de serveur C&C.

En effet, lorsque l'on cible les TTP d'un attaquant, on cible son comportement. Ceci est beaucoup plus difficile à changer car comprendre et détecter comment un attaquant pourrait, par exemple, faire une escalade de ses priviléges est différent de la recherche d'un fichier haché à une valeur connue pour l'outil Mimikatz (Un outil qui sert à).

Dans ce dernier cas, l'attaquant a juste besoin de recompiler son script avec des commentaires aléatoires ou du code qui ne sert à rien (junk code) pour changer la valeur du hash et contourner la détection basée sur les signatures. Cependant, trouver un compte qui devient administrateur est plus compliqué à éviter pour l'attaquant. Cela oblige l'attaquant à modifier son mode opératoire. Il est difficile pour un attaquant junior qui suit un playbook de changer son comportement. Ce qui va paralyser son opération et l'obliger à essayer son manuel ailleurs.

3.3 Conclusion

Dans la pyramide de la douleur, chaque niveau est une occasion de détecter et de prévenir divers indicateurs d'attaque. Les IoC traditionnels sont très accessibles via les flux de renseignements sur les menaces comme Threat Miner ou encore AlienVault. Néanmoins, trouver un compte qui devient administrateur est plus compliqué à éviter pour l'attaquant. Cela oblige l'attaquant à modifier son mode opératoire. Il est difficile pour un attaquant junior qui suit un playbook de changer son comportement. Ce qui va paralyser son opération et l'obligera à essayer son manuel ailleurs.

4 Cas d'utilisation de MITRE ATT&CK

4.1 Introduction

Après avoir exposé les principes et la philosophie de MITRE ATT&CK, il est temps de répondre à une question essentielle dans cette étude : **Quel est l'intérêt d'utiliser MITRE ATT&CK au sein d'un SOC?**

Le modèle ATT&CK est flexible et peut s'intégrer dans différents environnements. Donc il est important de bien cerner les domaines de son application et trouver le meilleur moyen pour l'adopter. Cette étude se concentre essentiellement sur l'intérêt de l'usage de MITRE ATT&CK au sein d'un SOC qui constitue le pilier de la surveillance et analyse de l'activité du système d'information d'une organisation.

Pour un SOC, ATT&CK peut intervenir dans : la **Threat Intelligence** (renseignements sur les nouvelles menaces et groupes d'attaquants), l'**évaluation** et l'**amélioration** des capacités de la **détection des menaces et attaques** du SOC, l'**émulation** des attaques et **test d'intrusion** automatisés et dans le **Threat Hunting** (chasse aux menaces).

Ces points seront détaillées dans ce chapitre. Le graphique suivant schématise les différents cas d'utilisation du modèle :



Figure 22 - Cas d'utilisation d'ATT&CK

4.2 ATT&CK et les renseignements sur les menaces

Se défendre contre un éventail d'acteurs malveillants avec différents niveaux de compétences est une tâche difficile. Cela nécessite de savoir qui attaque et comment.

En effet, le renseignement sur la menace est une activité millénaire permettant de répertorier et de caractériser les menaces émergentes afin d'anticiper les réponses appropriées. Le SOC est essentiellement consommateur de la « Threat Intelligence » utilisée pour enrichir les investigations et améliorer le processus de réponse à incident.

Les analystes SOC doivent être à jour des dernières menaces et doivent filtrer les informations pour le besoin de leur organisation.

L'approche traditionnelle pour faire de la veille sur les menaces est la lecture des : rapports des différents fournisseurs de sécurité (McAfee, Kaspersky, ProofPoint etc.), les tweets, les analyses internes, les articles universitaires et d'autres sources .

Après avoir traité et analysé les informations collectées, les analystes rédigent des rapports qui peuvent concerner un groupe de menaces spécifique, une vulnérabilité exploitée dans la nature ou une tendance récente d'un vecteur d'attaque.

En plus de lire et d'écrire, les analystes passent souvent une grande partie de leur journée à traiter l'autre partie du renseignement sur les cybermenaces : les indicateurs (IOC).

Les organisations utilisent des indicateurs d'activité malveillante, tels que les adresses IP, les domaines, les adresses e-mail et les certificats SSL / TLS, pour alerter et rechercher à l'appui de la défense du réseau.

Malgré les progrès du renseignement sur les cybermenaces, certains problèmes clés continuent de tourmenter les analystes. Les rapports écrits auront toujours un rôle dans CTI, mais on doit reconnaître leurs limites et envisager d'ajouter d'autres méthodes d'analyse pour remédier à ces limites. Les consommateurs peuvent ne pas avoir le temps où l'intérêt de lire les rapports. Le volume considérable de rapports et les nouveaux groupes d'attaquants menacent de submerger les analystes. Il faut des années aux analystes pour acquérir une expertise sur un groupe de menaces, et les nouveaux analystes peuvent avoir du mal à se mettre rapidement au courant.

Quant aux IOC on a appris précédemment qu'ils ne sont pas suffisants comme seul moyen d'identifier et de suivre les adversaires. Et comme vu antérieurement, la Pyramide de la douleur (Pyramid of Pain) encourage à se concentrer sur les tactiques, techniques et procédures des acteurs, qui provoquent un ralentissement considérable aux attaquants.

ATT&CK donne une méthode structurée pour décrire les TTP et les comportements des attaquants. Cette méthode peut être appliquée dans plusieurs aspects dans le cadre de cyber intelligence.

4.2.1 Renseignements sur un groupe d'attaquants

Prendre les groupes qui menacent le plus l'entreprise et son secteur d'activité et examiner leurs comportements dans ATT&CK.

Exemple : Examiner les groupes qui s'avèrent dangereux pour les entreprises du secteur de l'énergie. Il suffit de taper « Energy » dans la barre de recherche :

The screenshot shows a search results page with the query "energy". The results list several threat groups, each with a brief description and links to more information. The groups include:

- APT33, Elfin, Group G0064: An Iranian threat group targeting multiple industries since at least 2013, with a particular interest in aviation and energy sectors.
- OilRig, IRN2, HELIX KITTEN, APT34, Group G0049: An Iranian threat group targeting Middle Eastern and international victims since at least 2014, focusing on financial, government, energy, chemical, and telecommunications sectors.
- Magic Hound, Rocket Kitten, Operation Saffron Rose, Ajax Security Team, Operation Woolen-Goldfish, Newscaster, Cobalt Gypsy, APT35, Group G0059: An Iranian-sponsored threat group operating primarily in the Middle East since 2014, targeting energy, government, and technology sectors.
- APT19, Codoso, C0d0s00, Codoso Team, Sunshop Group, Group G0073: A Chinese-based threat group targeting defense, finance, energy, pharmaceutical, telecommunications, high tech, education, manufacturing, and legal services since 2010.
- Threat Group-3390, TG-3390, Emissary Panda, BRONZE UNION, APT27, Iron Tiger, LuckyMouse, Group G0027: A Chinese-based threat group targeting aerospace, government, defense, technology, energy, and manufacturing sectors since 2010.
- Dragonfly, Energetic Bear, Group G0035: A cyber espionage group active since 2011, initially targeting defense and aviation companies, then shifting to energy sectors.
- Sandworm Team, Quedagh, VOOODO BEAR, Group G0034: A Russian pro-hacktivist group targeting Ukrainian entities associated with energy, industrial control systems, SCADA, government, and media.
- Software: TSAdmin BITSAdmin is a command line tool used to create and manage BITS Jobs. BLACKCOFFEE is malware used by several Chinese groups since 2013.

Figure 23 - Rechercher un groupe selon un contexte

A partir des résultats obtenus, on sélectionne le groupe le plus actif dont les attaques sont récentes. On prend par exemple le groupe APT33 qui est un groupe iranien actif depuis 2013 et qui a ciblé plusieurs entreprises et organisations d'énergie et d'aviation en occident :

APT33

APT33 is a suspected Iranian threat group that has carried out operations since at least 2013. The group has targeted organizations across multiple industries in the United States, Saudi Arabia, and South Korea, with a particular interest in the aviation and energy sectors. [1] [2]

Figure 24 - Présentation du groupe APT33

La prochaine étape est de lister les techniques utilisées par ce groupe en utilisant le navigateur ATT&CK.

Donc on aura le résultat suivant qui montre les techniques utilisées par l'APT33 et qui ciblent les systèmes sous « Windows » :

APT33

												filters		
												stages: act		
												platforms: Windows		
Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact			
11 items	28 items	44 items	23 items	60 items	18 items	23 items	16 items	13 items	21 items	9 items	16 items			
Drive-by Compromise	CMDSTP	Accessibility Features	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	Application Deployment Software	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal			
External Remote Application	Command Line Interface	Accessibility Features	Application Shimming	Application Shimming	Brute Force	Application Discovery	Configuration and Code Injection	Cloud-to-Cloud Connection	Cloud-to-Cloud	Cloud-to-Cloud Through Removable	Cloud-to-Cloud			
External Remote Services	Component HTML File	Application Shimming	AppGet DLLs	AppGet DLLs	BITS Jobs	Behavioral Monitoring	Browser Bookmark Discovery	Cloud-based Services	Cloud-based Data	Cloud-to-Cloud	Cloud-to-Cloud			
Hardware Additions	Component Object Model and Applets	Applets	Applet DLLs	Bypass User Account Control	Credentials from Web Browsers	Domain Trust Discovery	Internal Spearphishing	Data from Information Repositories	Custom Command and Control	Data Encrypted	Data Encrypted for Impact			
Impersonation Through Removable Media	Control Panel Items	Application Shimming	Code Signing	Credentials in Files	File and Directory Discovery	File and Directory Discovery	File and Registry Discovery	Data from Local System	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Data Transfer Size Limits			
SpeakerPhishing Attachment	Dynamic Data Exchange	Authentication Package	Bypass User Account Control	Bypass User Account Control	Code Signing	Credentials in Registry	Pass the Hash	Data from Network Shared Drive	Data from Removable Media	Data Obfuscation	Decafacement	Disk Content Wipe		
SpeakerPhishing Link	Execution through API	BITS jobs	DLL Search Order Hijacking	Comprise After Delivery	Exploitation for Credential Access	Network Share Discovery	Pass the Ticket	Data from Removable Media	Data Staged	Encryption Over Command and Control	Encryption Over Other Networks	Disk Structure Wipe		
SpeakerPhishing via Service	Execution through Module Load	Bootkit	Exploitation for Privilege Escalation	Compiled HTML File	Forced Authentication	Network Sniffing	Remote Desktop Protocol	Data Transferred	Domain Fronting	Exfiltration Over Physical Medium	Endpoint Denial of Service	Endpoint Denial of Service		
Supply Chain Compromise	Execution for Client Execution	Browser Extensions	Extra Windows Memory Injection	Component Firmware	Hooking	Hooking	Password Policy Discovery	Remote File Copy	Domain Generation Algorithms	Scheduled Transfer	Transmitted System Recovery			
In-User Relationship	User Configuration	User Configuration Association	User Permissions Weakness	User Profile Hijacking	Input Hijacking	Input Hijacking	Man in the Browser	Man in the Browser	Resource Injektion					
Valid Accounts	Installlets	Component Framework	File Padding	File Padding	File System Permissions Weakness	File System Permissions Weakness	File System Permissions Weakness	File System Permissions Weakness	File System Permissions Weakness					
LSASS Driver	Component Object Model Hijacking	Image File Execution Options	Connection Privacy	Kerberos	Kerberos	Kerberos	Keyboard Intercept	Keyboard Intercept	Keyboard Intercept	Keyboard Intercept	Keyboard Intercept	Keyboard Intercept	Keyboard Intercept	Keyboard Intercept
Manta	Create Account	New Service	DCShadow	Logon Scripts	Logon Scripts	Logon Scripts	Logon Scripts	Logon Scripts						
PowerShell	DLL Search Order Hijacking	Parent PID Spoofing	Delegated Certificate/Decode Files or Scripts	Network Sniffing	Network Sniffing	Network Sniffing	Network Sniffing	Network Sniffing						
Regsvr32	External Remote Services	Path Interception	Disabling Security Tools	Password Filter DLL	Remote System Discovery	Remote System Discovery	Remote System Discovery	Remote System Discovery	Remote System Discovery					
Rundll32	File System Permissions Weakness	Port Monitors	DLL Search Order Hijacking	Private Keys	Security Software Discovery	Windows Admin Shares	Windows Admin Shares	Windows Admin Shares	Windows Admin Shares	Windows Admin Shares				
Unquoted Task	Hidden Files and Directories	PowerShell Profile	PowerShell Profile	Steal Web Session Cookie	System Information Discovery	System Information Discovery	System Information Discovery	System Information Discovery	System Information Discovery					
Execution	Hyperpreter	Process Hollowing	Process Hollowing	System Authentication Interception	System Authentication Interception	System Authentication Interception	System Authentication Interception	System Authentication Interception						
Service Execution	Image File Execution Options	Service Registry Permissions	Extra Windows Memory Injection	Extra Windows Memory Injection	File and Directory Permissions	File and Directory Permissions	File and Directory Permissions	File and Directory Permissions	File and Directory Permissions					
Signed Binary Proxy Execution	Logon Scripts	SID-History Injection	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter	File Delimiter
Signed Script Proxy Execution	LSASS Driver	Valid Accounts	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion	File Deletion
Third-party Software	Modify Existing Service	Web Shell	File System Logical Offsets	Steal Web Session Cookie	System Information Discovery	System Information Discovery	System Information Discovery	System Information Discovery	System Information Discovery					
Trusted Developer Utilities	Notch Helper DLL	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification	Group Policy Modification
User Execution	New Service	Hidden Files and Directories	Hidden Files and Directories	Indicator Removal on Host	Indicator Removal on Host	Indicator Removal on Host	Indicator Removal on Host	Indicator Removal on Host						
WMI Management	Common Application Startup	Modifying Registry	Modifying Registry	Image File Execution Options	Image File Execution Options	Image File Execution Options	Image File Execution Options	Image File Execution Options						
Windows Remote Management	Path Interception	Screen Saver	Screen Saver	Installer	Installer	Installer	Installer	Installer						
YAML Script Processing	Port Monitors	Screen Saver	Screen Saver	Indicator Removal on Host	Indicator Removal on Host	Indicator Removal on Host	Indicator Removal on Host	Indicator Removal on Host						
		PowerShell Profile	PowerShell Profile	Scripting	Scripting	Scripting	Scripting	Scripting						
		Rebootkit	Rebootkit	Session Hijacking	Session Hijacking	Session Hijacking	Session Hijacking	Session Hijacking						
		Rundll32	Rundll32	Signature-Based File Execution	Signature-Based File Execution	Signature-Based File Execution	Signature-Based File Execution	Signature-Based File Execution						
		Rootkit	Rootkit	Software Packing	Software Packing	Software Packing	Software Packing	Software Packing						
		RunDLL32	RunDLL32	Template Injection	Template Injection	Template Injection	Template Injection	Template Injection						
		Scripting	Scripting	Timestamp	Timestamp	Timestamp	Timestamp	Timestamp						
		Session Hijacking	Session Hijacking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking						
		Signed Binary Proxy Execution	Signed Binary Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution	Signed Script Proxy Execution
		SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking	SIP and Trust Provider Hijacking
		Software Packing	Software Packing	Template Injection	Template Injection	Template Injection	Template Injection	Template Injection						
		Timestamp	Timestamp	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking						
		Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking	Threat Actor Blocking
		Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities	Trusted Developer Utilities
		Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts	Valid Accounts
		Web Service	Web Service	Web Service	Web Service	Web Service	Web Service	Web Service	Web Service	Web Service	Web Service	Web Service	Web Service	Web Service
		WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing	WSL Script Processing

Figure 25 - Matrice des techniques du groupe APT33

Ensuite, pour chaque technique, collecter les informations qu'il faut pour sa détection, à savoir, le champ « **data sources** ».

Pour automatiser ce processus j'ai créé un script Python qui permet d'obtenir toutes les techniques utilisées par un groupe donné et les sources d'information liées à chaque technique.

Le script peut être téléchargé sur mon GitHub (il est aussi présenté dans la partie « Annexes ») : <https://github.com/Azara/MITRE-ATTACK-Group-CTI-automation>

Il se base sur le langage **STIX** (Structured Threat Information Expression) qui est un format de sérialisation utilisé pour échanger des renseignements sur les cybermenaces. Les échanges se font grâce au protocole **TAXII** (Trusted Automated Exchange of Intelligence Information) qui est conçu spécialement pour les communications qui concernent les menaces cyber.

TAXII permet aux organisations de partager les renseignements en définissant une API qui s'aligne sur les modèles de partage communs. Il existe deux modèles :

Collections : c'est une interface vers un référentiel logique d'objets CTI (Cyber Threat Intelligence) fourni par un serveur TAXII qui permet à un producteur d'héberger un ensemble de données CTI qui peuvent être demandées par les consommateurs : les clients et serveurs TAXII échangent des informations dans un modèle de requête-réponse (request-response model).

Canal : un canal permet aux producteurs de transmettre des données à de nombreux consommateurs et aux consommateurs de recevoir des données de nombreux producteurs : les clients TAXII échangent des informations avec d'autres clients TAXII dans un modèle de publication-abonnement.

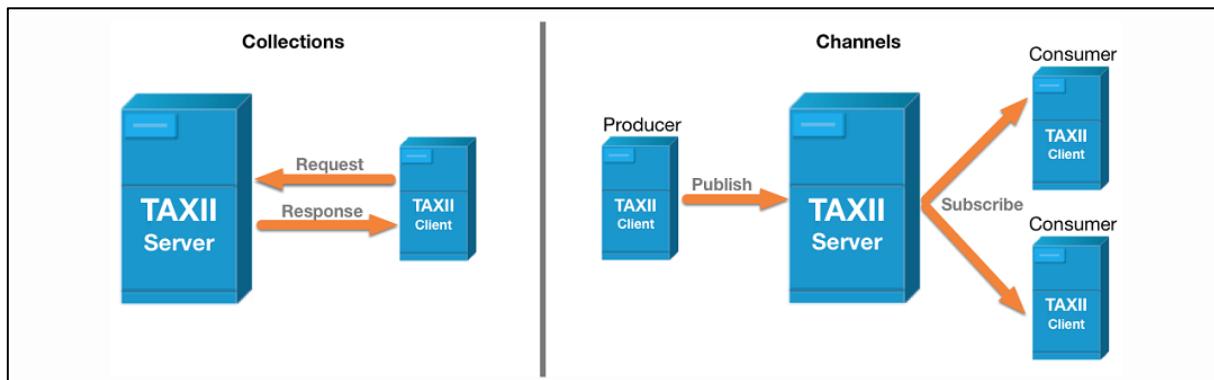


Figure 26 - Différence entre les modèles de partage dans un serveur TAXII – source : <https://oasis-open.github.io/cti-documentation/taxii/intro.html>

MITRE a choisi d'héberger ses différentes matrices et informations sur les groupes et logiciels malveillants sous forme d'un modèle « **Collection** » sur un serveur TAXII dédié. Plusieurs collections ont été mises en place : une collection par matrice de techniques. Chaque matrice est caractérisée par un identifiant.

Mon script va donc jouer le rôle d'un consommateur et requête la collection qui concerne la matrice « Enterprise » (c'est la matrice qui nous intéresse le plus) en récupérant son identifiant. MITRE avec son serveur TAXII est le producteur, et la requête envoyée permet de récupérer les techniques et sources de données qui correspondent un acteur malveillant que l'on sélectionne.

Le résultat est un fichier CSV (Comma-separated values) qui contient les techniques d'un groupe et les sources de données qui les coïncident.

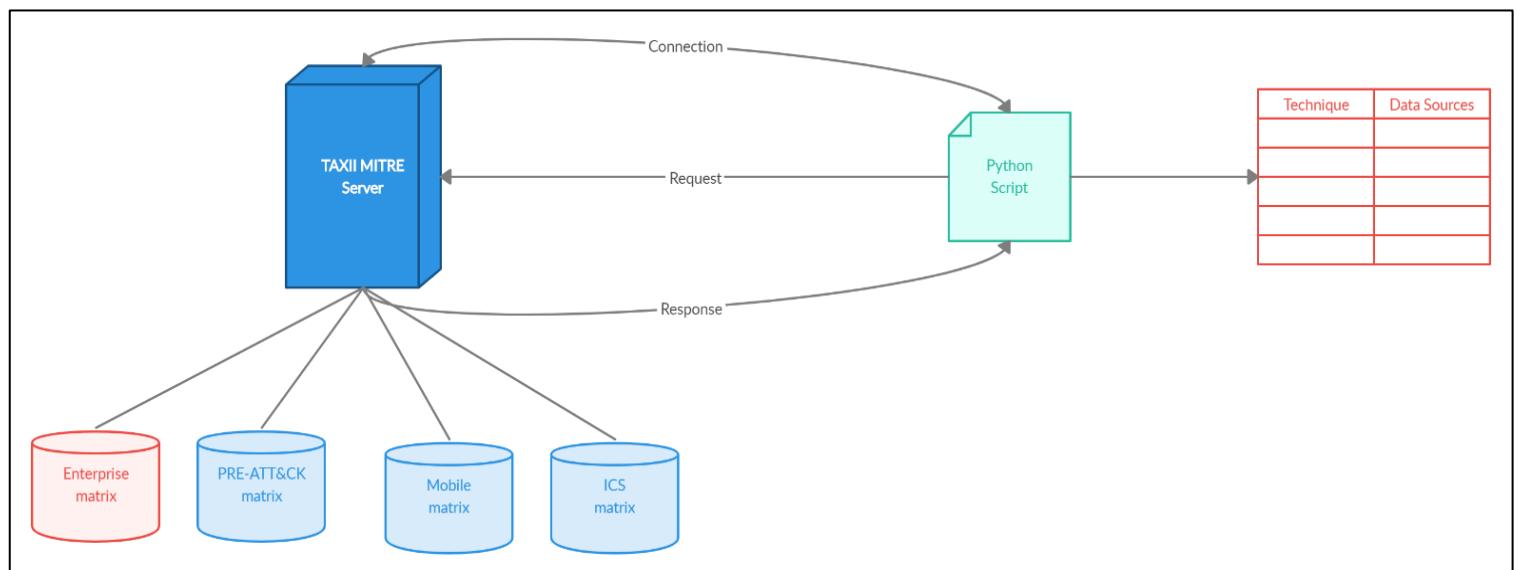


Figure 27- Schématisation du principe du script Python 1

Pour une meilleure visualisation, le fichier résultant peut être chargé sur le site « Raw Graphs » : <https://app.rawgraphs.io/>.

En reprenant l'exemple du groupe APT33 on aura le résultat avec un « diagramme de Sankey » qui montre les techniques que le groupe utilise et en face les source d'information pour chaque technique :

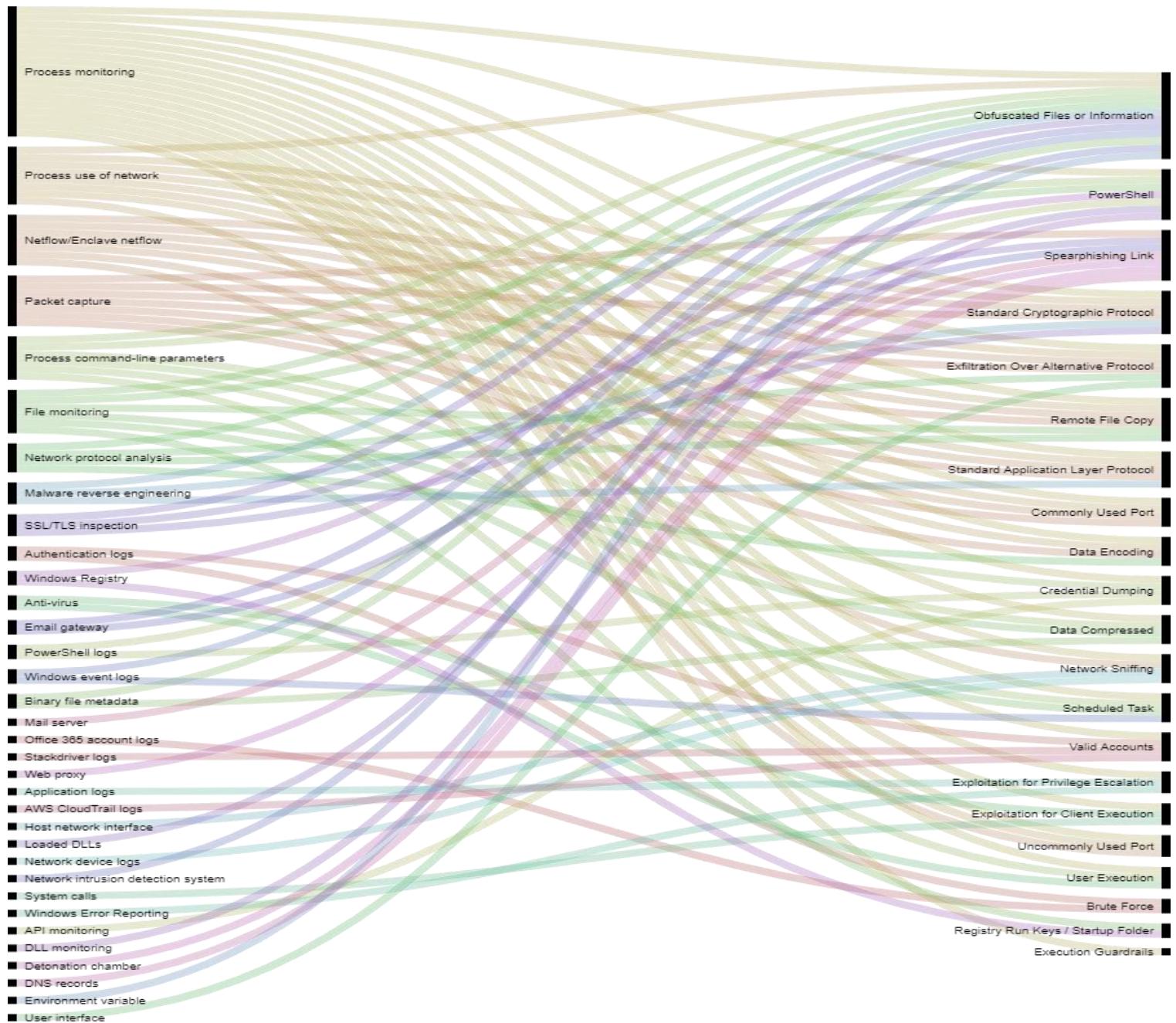


Figure 28 - Diagramme de Sankey pour les techniques de l'APT33

Un autre exemple de visualisation avec un « Dendrogramme circulaire » :

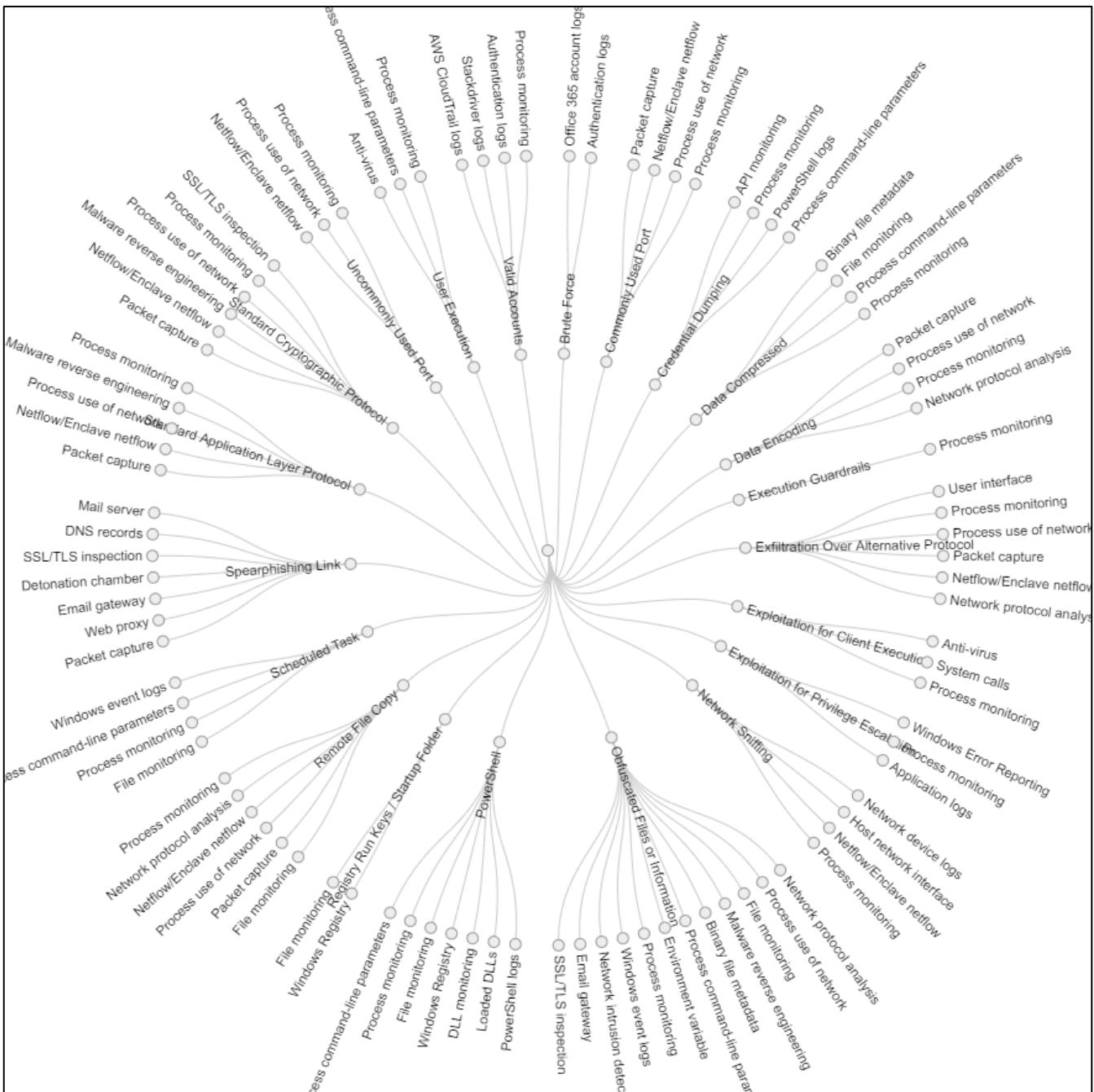


Figure 29 - Dendrogramme circulaire techniques APT33

D'après ces résultats, on remarque que la supervision des processus permet de détecter un nombre important des techniques utilisées par ce groupe.

4.2.2 Comparer les groupes

Prendre les groupes les plus menaçants pour l'entreprise et comparer leurs techniques afin d'extraire les techniques communes et d'essayer de découvrir un pattern qui aide à comprendre leur mode opératoire.

Si l'on prend l'exemple précédemment, on pourra comparer l'APT33 avec un autre groupe qui est connu pour cibler le secteur de l'énergie. Par exemple, **Dragonfly** qui est spécialisé dans les attaques contre les systèmes ICS et actif depuis 2015. De la même façon que l'APT33, on va générer sa matrice de techniques. Le navigateur ATT&CK offre la possibilité de comparer les techniques entre une ou plusieurs matrices.

Le calque des matrices donne une vision sur les techniques communes (en vert):

Figure 30 - Résultat de la comparaison des groupes APT33 et Dragonfly

En détectant les techniques communes aux groupes, on comprendra leur façon d'agir et on élargira l'angle de tir et on pourra même cibler un cartel d'attaquants spécialisé pour attaquer un secteur d'activité particulier.

La comparaison des groupes se fait aussi avec une base de données « Neo4J » où l'on peut charger toutes les informations présentes dans MITRE ATT&CK grâce au script « attack2neo » :

<https://github.com/vmapps/attack2neo>

En requérant la base, on aura les techniques communs des groupes précédents. On peut également rechercher les outils utilisés par les groupes.

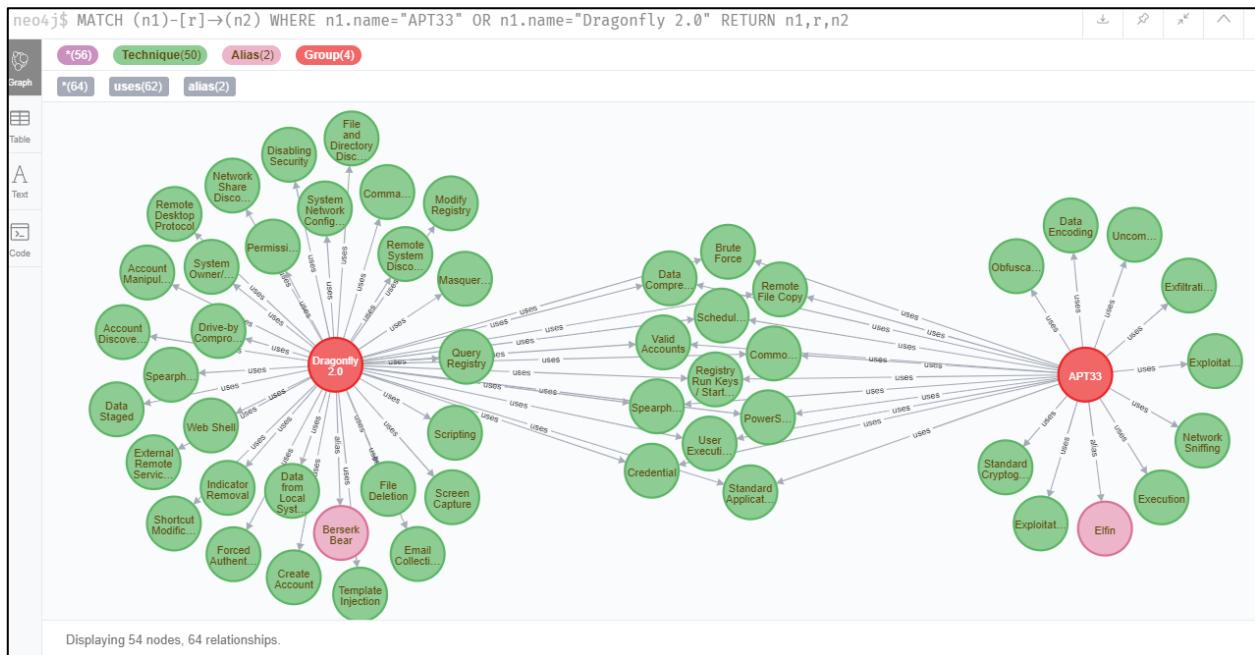


Figure 31 - Comparaison des groupes avec Neo4J

4.2.3 Mapper ses propres techniques

Il s'agit d'une action de niveau supérieur que l'on peut entreprendre et qui consiste à cartographier soi-même les renseignements sur ATT&CK plutôt que d'utiliser ce que d'autres ont déjà cartographié. Cela concerne les rapports d'incidents internes, des rapports externes qui concernent les menaces récentes non-cartographiées sur ATT&CK ou des tout simplement les articles d'actualités cyber.

The screenshot shows a list of extracted techniques from a text document. The techniques are categorized by their corresponding ATT&CK Tactic and Sub-Tactic. Some categories are highlighted in blue boxes:

- Defense Evasion | Obfuscated Files or Information (T1027)
 - The Trojan obfuscates its executable code prior to compilation, rather than packing it like most other ransomware, making it harder for researchers to reverse engineer.
 - It also obscures the links to the necessary API function, and stores hashes to strings rather than the actual strings.
 - Upon installation, the Trojan reviews the directory to execute the payload, and launches it from an 'incorrect' directory – such as a potential automated sandbox.
 - The Trojan checks if the victim's PC has a keyboard set to Cyrillic script.
 - Before encrypting files on a victim device, SynAck checks the hashes of all running processes and services against its own Impact | Data Encrypted for Impact (T1486).
 - SynAck robs virtual machines, office applications, script interpreters, database applications, backup systems, and System Service Discovery (T1007) to make it easier to seize valuable files which might otherwise be tied up into the running processes.
- Discovery | File and Directory Discovery (T1083)
- Defense Evasion | Virtualization/Sandbox Evasion (T1497)
- Defense Evasion | Execution Guardrails (T1480)
- Impact | Data Encrypted for Impact (T1486)
- Discovery | Process Discovery (T1057)
- Discovery | System Service Discovery (T1007)

Figure 32 - Exemple d'extraction de techniques à partir d'un texte

Néanmoins, cette activité peut s'avérer compliquée et il est difficile pour un analyste de se familiariser avec toutes les techniques (qui dépassent 200) et de comprendre les subtilités de la manière dont le renseignement est mis en correspondance avec elles.

Pour remédier à ce problème, MITRE a développé un outil de traitement du langage naturel appelé TRAM (Threat Report ATT&CK Mapper) qui aide à analyser les articles cyber et d'en extraire les techniques ATT&CK d'une façon automatique.

TRAM est un outil Web géré en local qui permet aux utilisateurs de soumettre une URL de page Web (Exemple : un lien vers un article qui parle d'un nouveau ransomware).

Il utilise un modèle de reconnaissance du langage naturel qui est basée sur une méthode appelée **régression logistique** (Logistic Regression) destinée pour faire des prédictions, et pour prévoir quelles techniques pourraient être utilisées dans une phrase donnée.

The screenshot shows the homepage of the Threat Report ATT&CK Mapper (TRAM). The interface is divided into several sections:

- A header bar with the title "Threat Report ATT&CK Mapper (TRAM)" and a logo.
- A left sidebar with fields for "Enter New Report": "Insert URL" (with a text input field), "Insert Title" (with a text input field), and a "Submit" button.
- Three main status boxes:
 - "NEEDS REVIEW" containing an "Example Report" section with "Source" and "Analyze" buttons.
 - "ANALYST REVIEWING" (empty box)
 - "COMPLETE" (empty box)

Figure 33 - Page d'accueil de TRAM

Une fois le lien soumis, TRAM va procéder à l'analyse du texte. Lorsque le modèle de régression logistique de TRAM prédit qu'il a trouvé une technique, il met en évidence le texte pertinent et affiche la technique prédictive dans un encadré à droite.

The screenshot shows a web-based application titled "Threat Report ATT&CK Mapper (TRAM)". The main content area displays an article about Maze ransomware, specifically its initial distribution via exploit kits and spam campaigns. Several sentences in the text are highlighted in yellow, indicating they contain detected techniques. To the right of the article, a sidebar titled "Techniques Found" lists three types of spearphishing attacks, each with "Accept" and "Reject" buttons. Below this is a section titled "Confirmed Techniques". At the bottom of the sidebar, there are buttons for "Add A Missing Technique" and "Account Access Removal", along with a "Add Technique" button.

Figure 34 - Analyse d'un article avec TRAM

L'analyste examine le résultat et « accepte » ou « rejette » la prédiction de la technique.

En coulisses, lorsque le bouton "Accepter" est cliqué, la phrase et la technique passent dans le tableau "**Vrais positifs**" de la base de données.

Lorsque le bouton "Rejeter" est cliqué, la phrase passe dans le tableau "**Faux positifs**". Ces deux tableaux aident à reconstruire le modèle et à rendre la précision plus précise lors des prochaines analyses.

En plus des prédictions, l'analyste peut également ajouter des techniques manquantes manuellement. Lorsqu'une technique manquante est ajoutée, elle sera mise dans le tableau des "**vrais négatifs**", qui est également pris en compte lors de la reconstruction des modèles.

Après la revue, les résultats peuvent être exportés en format PDF ou générés dans un fichier JSON qui peut être importé dans le navigateur ATT&CK afin d'obtenir une visualisation des techniques extraites du rapport:

Figure 35 - Résultat d'une extraction des techniques avec TRAM

TRAM est encore en phase de développement. Plusieurs fonctionnalités peuvent être ajoutées comme la possibilité d'intégrer des types de fichiers supplémentaires (.doc, .pdf ou .txt).

La question qui se pose maintenant est de savoir comment rendre toutes ces informations actionnables et quelle utilité pour le SOC ?

Les informations collectées par les analystes sont utiles pour améliorer la détection. En effet, comme les analystes, les membres de l'équipe qui s'occupe de la détection dans le SOC peuvent aussi structurer les informations sur les comportements qu'ils peuvent détecter et atténuer. En superposant les informations provenant des analystes et les celles de la détection, on pourra prioriser ce que l'on va détecter.

Si l'on reprend l'exemple de l'APT33 et en supposant que l'équipe détection a mappé les techniques qu'elle peut détecter dans ATT&CK (la méthode sera vue dans les prochains chapitres), on va alors calquer les deux matrices pour observer les fossés dans la détection qui sont les techniques que l'on ne détecte pas actuellement.

Exemple :

Matrice de détection : représente les techniques que l'on peut détecter.

filters											
stages: act platforms: Windows, Linux, macOS											
Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact
11 items	34 items	62 items	69 items	21 items	23 items	18 items	13 items	22 items	9 items	16 items	
Drive-by Compromise	AppleScript	.bash profile and .bashrc	ACCESS Token	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal	
Exploit Public-Facing Applications	CMSTP	Accessibility Features	Accessibility Features	Bash History	Application Window	Application Deployment	Automated Collection	Command-and-control	Data Compressed	Data Destruction	
Attenuate Remote Service	Command-Line Interface	Account Manipulation	AppCert DLLs	BITS Jobs	Brute Force	Bookmark Discovery	Clipboard Data	Connection Proxy	Data Encrypted for Transfer	Data Incomplete	
Hardware Additions	Compiled HTML File	AppCert DLLs	AppCert DLLs	PPAPI User Account	Browser Dumping	Domain Trust Discovery	Data from Local System	Control Protocol and Repostories	Control Protocol and Repostories	Disk Content Wipe	
Manipulating Through Configuration and Distribution C2M	Applink DLLs	Application Shimming	Clear Command History	Downloads from Web Browsers	DomainTrust	Internal Spearphishing	Data from Network	Data Encoding	Data Encryption	Disk Structure Wipe	
After-Attack	Control Panel Items	Application Shimming	PPAPI User Account	CMSSTP	Credentials in Files	Logon Scripts	Scattered Drives	Data Obfuscation	Encryption and Control	Erasure and Other	
Spearphishing Link	Dynamic Data Exchange	Authentication Package	PPAPI User Account	CMSTP	Credentials in Registry	Network Share Discovery	Pass the Hash	Data Staged	Domain Fronting	Exploit Denial of Service	
Spearphishing via Service	Execution through API	BITS jobs	Code Signing	Code Signing	EVASION	Network Sniffing	Pass the Ticket	Data Transfer Size Limits	Firmware Over Physical Medium	Firmware Corruption	
Compromised	Malicious Through Configuration and Distribution C2M	Bootkit	Dylib Hijacking	Compile After Delivery	EVASION	Pass the Hash	Pass the Ticket	File Generation	File Generation	Inhibit System Recovery	
Trusted Relationship	Exploitation for Client	Browser Extensions	Emond	Forced Authentication	EVASION	Perimeter Desktop	Email Collection	File Transfer	File Transfer	File Transfer	
Valid Accounts	Graphical User Interface	Launch Default File	Exploration for Privilege	EVASION	EVASION	Peripheral Device	Fallback Channels	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	InstallUtil	Component Firmware	EXTRA Window Memory	EVASION	EVASION	Remote File Copy	Input Capture	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Launchctl	Component Object Mode	EVASION	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Local Job Scheduling	Create Account	Hooking	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	LSASS Driver	DLL Search Order	IMAGE File Execution	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Mshta	Dvlib Hijacking	Launch Daemon	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	PowerShell	Emond	New Service	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Regsvcs\Regasm	External Remote	Parent PID Spoofing	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Regsvr32	DLL Side-Loading	Path Interception	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Rundll32	Hidden Files and Directories	Plist Modification	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Scheduled Task	Hooking	Port Monitors	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Scripting	Hypervisor	PowerShell Profile	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Service Execution	Image File Execution	Process Injection	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Signed Binary Proxy	Kernel Modules and Extensions	Scheduled Task	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	EVASION	Launch Agent	PERMISSIONS Weakness	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Source	Launch Daemon	Setuid and Setgid	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Space after Filename	Launchd (I)	SID-History Injection	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Third-party Software Addition	LDLOAD DYLIB	Startup Items	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Trap	Local Job Scheduling	Sudo	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	User Developer Utilities	Login Item	Sudo Caching	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	User Execution	Logon Scripts	Valid Accounts	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Mitigation Management	LSASS Driver	Web Shell	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	Windows Remote Management	Modify Existing Service	Indicator Blocking	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
	XSL Script Processing	Netsh Helper DLL	Indicator Removal from Application	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
		New Service	Indirect Command Execution	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	
		Office Application Startup	Install Root Certificate	EVASION	EVASION	EVASION	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	Filesystem Denial of Service	

Figure 36 - Exemple d'une matrice de détection

On va superposer cette matrice avec celle de l'APT33. Les techniques en rouge représentent les fossés que l'on a dans la détection.

Gaps			filters						legend					
			stages: act platforms: Windows, Linux, macOS											
Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact			
11 items	34 items	62 items	33 items	60 items	23 items	23 items	18 items	13 items	22 items	9 items	16 items			
Drive-by Compromise	AppleScript	bash profile and .bashrc	Access Token Manipulation	Account Manipulation	Account Discovery	AppleScript	Automated Collection	Automated Collection	Automated Command and Control	Automated Exfiltration	Account Access Removal			
Exploit Public-facing Applications	CMSTP	Accessibility Features	Access Token Features	Bash History Padding	Application Window	Application Deployment	Communication Through Removable Media	Clipboard Data	Cloud-based Proxy	Compressed Data	Data Destruction			
Extract Remote Services	Command-Line Interface	Account Manipulation	AppCn DLLs	BITS Job	Brute Force	Bookmark Discovery	Commodity Object Mode and Infrastructure	Clipboard Data	Custom Command and Control Protocol	Customized Cryptographic Protocol	Data Encrypted for Impact			
Hardware Advertisements	Compiled HTML File	AppCn DLLs	AppCn DLLs	Appress User Account	Credential Dumping	Discovery of Remote Services	Discovery of Remote Web and Directory Browsing	Data from Local System	Encryption	File Encryption	File Transfer Size Limits	Dereferencing		
Manipulating Through JavaScript and HTML/CSS/JSON	Control Panel Items	Application Shimming	Application Shimming	Clear Command History	Clearing the Browser Cache	Internal Spearphishing Scanning	Internal Spearphishing via Plugins and Scripts	Logon Scripts	Logon Scripts	Logon Scripts	Logon Scripts	Logon Scripts	Logon Scripts	Logon Scripts
Attachment	Dynamic Data Exchange	Authentication	Hubbard Order	Code Signing	Credentials in Registry	Network Share Discovery	Pass the Hash	Pass the Hash	Pass the Ticket	Data Staged	Data Staged	Data Structure Wipe	Denial of Service	Denial of Service
Spearmarking Link	Dynamic Data Exchange	Authentication	Hubbard Order	Compile After Delivery	Evil Genius	Network Sniffing	Pass the Desktop	Pass the Desktop	Pass the Desktop	Protocol	Protocol	Protocol	Protocol	Protocol
Code Shimming via Dynamic Linking	BITS Jobs	Dylib Hijacking	Dylib Hijacking	Forced Authentication	Forced Authentication	Discover Device	Remote File Copy	Remote File Copy	Remote File Copy	Remote File Copy	Remote File Copy	Resource Hijacking	Resource Hijacking	Resource Hijacking
User Change	Execution through API	Bootkit	Bootkit Execution with Persistence	Compiled HTML File	Hooking	Discover Network	Input Capture	Input Capture	Input Capture	Input Capture	Input Capture	Malicious Mechanism	Malicious Mechanism	Malicious Mechanism
Configuration	MSBuild Task	MSBuild Task	Parent PID Spoofing	Emond	Hooking	Discover Network Groups	Internal Discovery	Internal Discovery	Internal Discovery	Internal Discovery	Internal Discovery	Multi-hop Proxy	Multi-hop Proxy	Multi-hop Proxy
Trusted Relations	Execution for Client	Browser Extensions	Emond	Input Capture	Kerberosasting	Query Registry	Remote Services	Remote Services	Remote Services	Remote Services	Remote Services	Man in the Browser	Man in the Browser	Man in the Browser
Valid Accounts	Graphical User Interface	Administrator Default File	Exploitation for Privileged User	Input Prompt	DCShadow	Remote System	SSH Hijacking	SSH Hijacking	SSH Hijacking	SSH Hijacking	SSH Hijacking	Screen Capture	Screen Capture	Screen Capture
InstallUtil	Component Firmware	File System Permissions	File System Object Mode	Job Creation	Keychain	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Launchcht	Component Object Model	File System Permissions	File System Object Mode	Job Creation	Kerberosasting	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Local Job Scheduling	Create Account	Hooking	Hooking	Job Creation	DCShadow	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
LSASS Driver	Hubbard Order	Job Creation	Job Creation	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Msiha	Dylib Hijacking	Launch Daemon	Launch Daemon	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
PowerShell	Emond	New Service	New Service	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Regsvcs/Regasm	External Remote	Parent PID Spoofing	DLL Side-Loading	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Regsvr32	File System Permissions	Path Interception	Execution Guardrails	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Rundll32	Hidden Files and Directories	Plist Modification	Exploitation for Defense	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Scheduled Task	Hooking	Port Monitors	Extra Window Memory	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Scripting	Hypervisor	PowerShell Profile	File and Directory Browsing	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Service Execution	Image File Execution Monitor	Process Injection	File Deletion	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Signed Binary Proxy	Kernel Modules and Drivers	Scheduled Task	File System Logical Gatekeeper Bypass	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Signed Script Proxy	Launch Agent	PERC Registry	Gatekeeper Bypass	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Source	Launch Daemon	Setuid and Setgid	Group Policy Modification	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Space after Filename	Launchcht	SID-History Injection	Hidden Files and Directories	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Third-party Software	LLC LOAD DYLIB	Startup Items	Hidden Users	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Trap	Local Job Scheduling	Sudo	Hidden Window	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Trusted Developer	Login Item	Sudo Caching	HISTCONTROL	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
User Execution	Logon Scripts	Valid Accounts	Image File Execution Monitor	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Windows Management	LSASS Driver	Web Shell	Indicator Blocking	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
Windows Remote Management	Modify Existing Service	Netsh Helper DLL	Indicator Removal from Registry	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop
XSL Script Processing	New Service	Office Application Startup	Indirect Command	Job Creation	Job Creation	Remote System	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Discovery	Service Stop	Service Stop	Service Stop

Figure 37 - Exemple de matrice montrant les lacunes dans la détection

Ces techniques manquantes sont donc à prendre en considération et à prioriser dans la détection si l'on estime que l'APT33 est menaçante pour l'entreprise et permettent l'évolution de la couverture défensive.

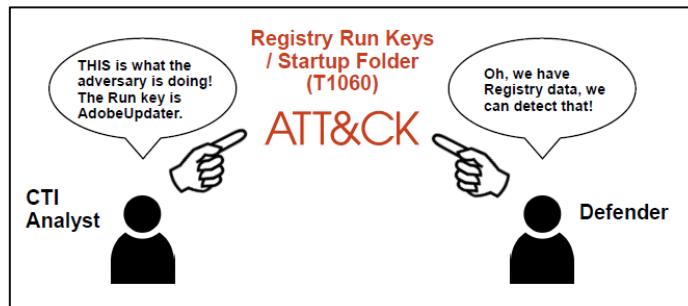


Figure 38 - Echange entre analyste et équipe détection

Enfin, on voit bien vu qu'ATT&CK est un moyen efficace pour l'échange entre les membres des différentes équipes au sein d'un SOC (Analystes, Build, RSSI etc.).

4.3 ATT&CK et détection

Un des cas d'utilisation les plus importants d'ATT&CK est l'évaluation et l'amélioration de la couverture du système de détection des menaces du SOC. Une évaluation des lacunes défensives permet à une organisation de déterminer quelles parties de son entreprise manque de défense et/ou de visibilité. Ces lacunes représentent des angles morts pour les vecteurs potentiels qui permettent à un attaquant d'accéder au système d'information sans être détecté ou atténué. ATT&CK vient donc pour évaluer les outils, la surveillance et les mesures d'atténuation des défenses existantes au sein du SOC. Les lacunes identifiées sont utiles pour hiérarchiser les investissements en vue de l'amélioration d'un programme de sécurité. Des produits de sécurité similaires peuvent également être comparés à un modèle de comportement adverse commun afin de déterminer la couverture avant l'achat.

Le processus d'évaluation/amélioration de la détection passe par 4 étapes essentielles : **Connaitre sa couverture défensive , Identifier les lacunes, prioriser et améliorer.**

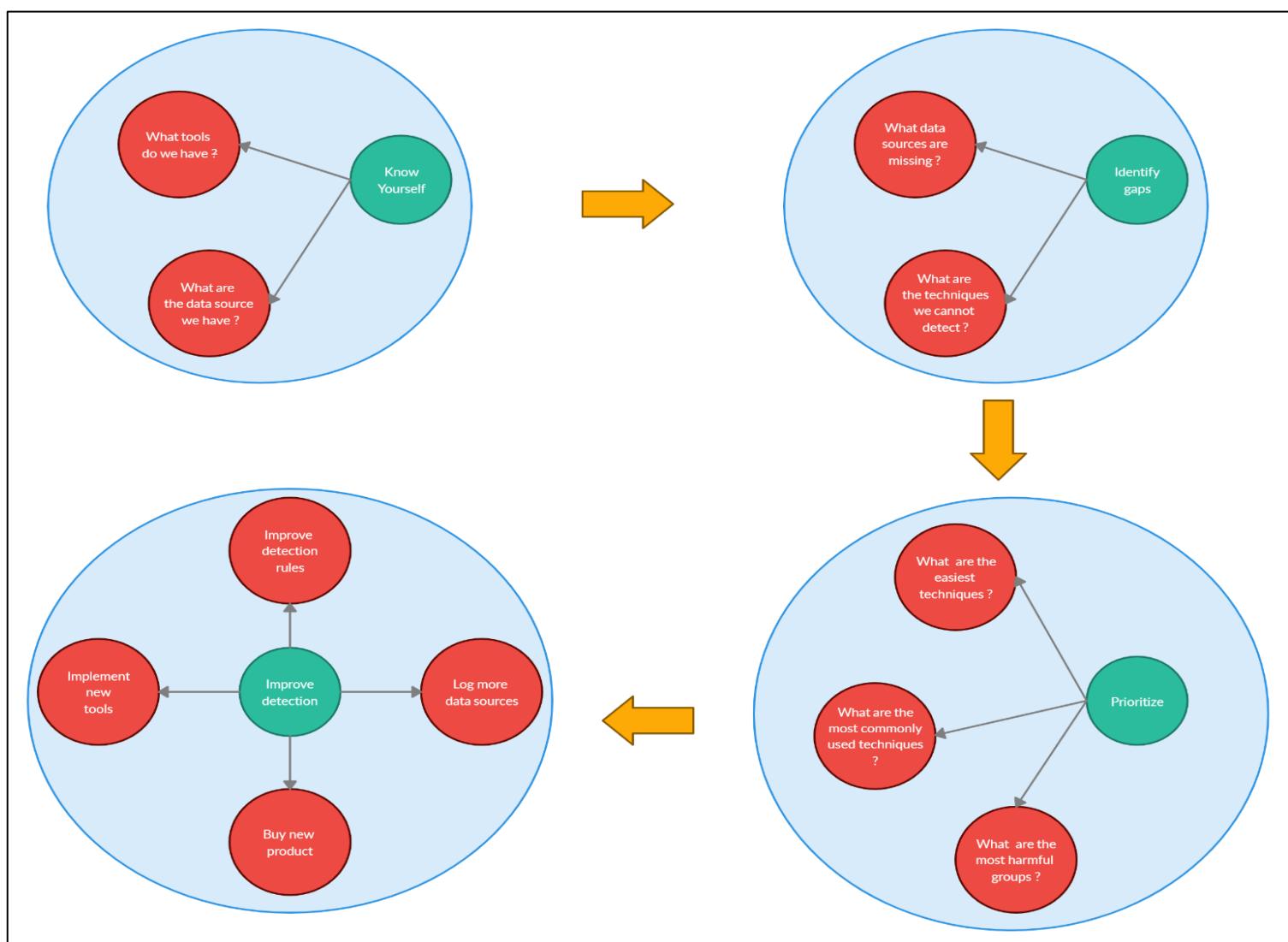


Figure 39 - Etapes d'application d'ATT&CK dans la détection

Afin d'appliquer ces étapes plusieurs outils sont disponibles en Open Source. Mais l'outil qui se démarque le plus pour ce cas d'utilisation d'ATT&CK est « **DeTT&CT** » (**DEtect Tactics, Techniques & Combat Threats**) qui axé sur l'évaluation et la comparaison de la **qualité des sources d'information disponibles, la couverture de détection et les comportements des acteurs de la menace** : <https://github.com/rabobank-cdc/DeTTECT>.

Cet outil est écrit en Python et interagit avec le module « ATT&CK Python client » qui permet d'accéder au contenu ATT&CK actualisé et disponible sur le serveur TAXII de MITRE.

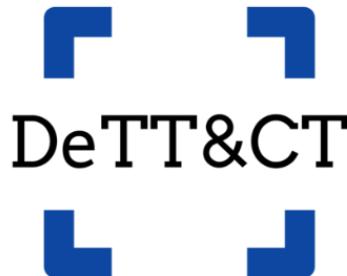


Figure 40 - Logo de DeTT&CT

4.3.1 Connaitre sa couverture défensive

Pour l'équipe bleue (Blue Team), il est crucial de savoir quelles sont les sources de données dont on dispose, quelle est leur qualité en matière de détection et si elles peuvent être utilisées pour effectuer des analyses de données. Cela va permettre de savoir quels sont les comportements d'attaquants susceptibles d'être détectés.

Donc avec ATT&CK, la première étape consiste à faire l'inventaire des informations dont on dispose grâce aux capteurs en place.

C'est la première étape de l'utilisation du Framework DeTT&CT. ATT&CK a défini environ 50 types de sources de données, qui sont incluses dans DeTT&CT. Pour chaque source de données disponible, il faut donner un score basé sur 5 critères :

- **Exhaustivité du dispositif (Device completeness)** : indique si les données requises sont disponibles pour tous les terminaux.
- **Complétude des champs de données (Data field completeness)** : Indique le degré de la disponibilité des informations requis. Et à quel point les champs disposent des données utiles dans les investigations. Par exemple, on a les logs des proxy (information requises) mais les événements n'ont pas le champs « Hôte ».
- **Actualisation (Timeliness)** : Indique quand les données sont disponibles, et dans quelle mesure les horodatages des données sont précis par rapport à l'heure réelle à laquelle un événement s'est produit.
- **Cohérence (Consistency)** : standardisation des noms et des types de champs de données.
- **Conservation (Retention)** : Indique la durée de stockage des données par rapport à la période de conservation souhaitée.

Les scores vont de 0 à 5 (**non, peu, moyen, bien, très bien, excellent**) pour chaque critère et cette administration des sources de données est stockée dans un fichier YAML (Markup Language) dédié aux sources de données.

4.3.2 Identifier les lacunes

Sur la base des sources de données définies et notées dans le fichier d'administration, DeTT&CT peut générer ; avec la commande qui suit ; un fichier JSON qui permet la visualisation des techniques qui correspondent aux sources de données recensées :

```
$ python detect.py ds -fd datasources.yaml -l
```

Ci-après la matrice des techniques après chargement du fichier JSON résultant:

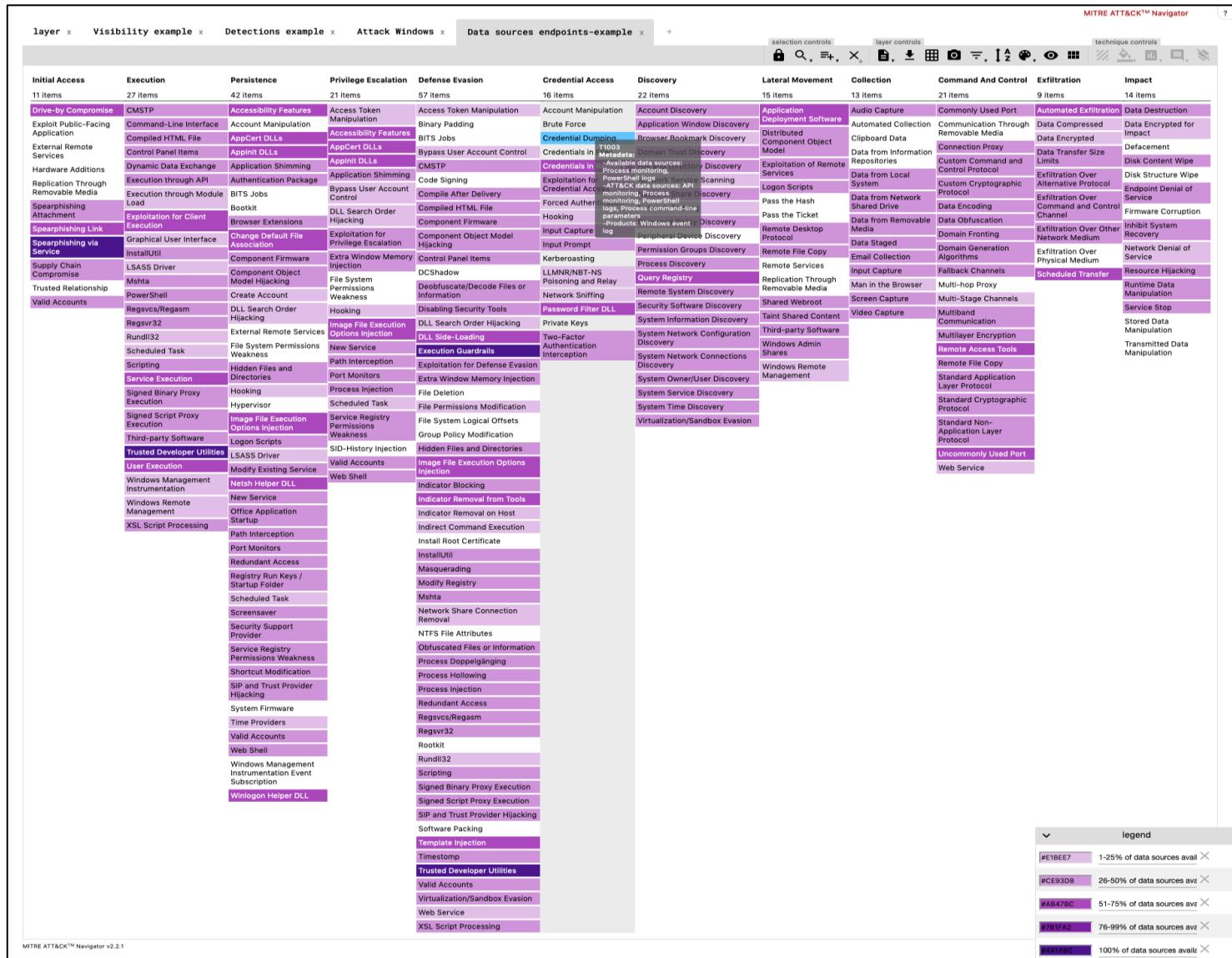


Figure 41 - Matrice couverture défensive 1

Cette matrice donne un aperçu général de la couverture défensive et révèle les techniques qui ne sont pas ou peu détectées. Elle constitue un point d'entrée pour l'abonnement de la détection.

Toutefois, cette évaluation n'est pas très précise et on a besoin d'avoir la visibilité exacte pour chaque technique. Pour réaliser cette tâche, on peut générer un autre fichier YAML **d'administration des techniques** basé sur le fichier **d'administration des sources d'informations**, qui fournira des scores de visibilité plus précis.

Par défaut, l'argument `--yaml` n'inclura dans le fichier YAML résultant que les techniques pour lesquelles le score de visibilité est supérieur à 0.

```
$ python detect.py ds -fd datasources.yaml --yaml
```

Dans le nouveau fichier YAML obtenu, on peut choisir d'ajuster manuellement le score de visibilité par technique en fonction de **l'expérience et de la connaissance** de l'environnement mais aussi de la **qualité des sources de données notées préalablement**. La notation manuelle peut être nécessaire pour plusieurs raisons :

- On peut par exemple disposer d'une source de données parmi 3 mentionnées sur une technique particulière. Mais dans certain cas, cette source de donnée unique pourrait ne pas être suffisante pour détecter la technique. C'est pourquoi le score de visibilité basé sur le nombre de sources de données disponible doit être ajusté.
- La qualité d'une source de données particulière est considérée comme trop faible pour être utile à la visibilité. Grâce à la puissance d'une requête EQL (Event Query language) – expliquée en « Annexes » -, on peut influencer les sources de données qui sont incluses dans le processus de génération automatique de scores de visibilité. Par exemple, pour exclure les sources de données dont la qualité est faible.
- On a un certain niveau de visibilité sur une technique mais c'est basé sur une source de données qui n'est pas mentionnée dans ATT&CK pour cette technique donnée.

Les scores de visibilité sont notés de 0 à 4 (**non disponible, minimale, moyenne, bien, excellente**). Il est possible d'avoir plusieurs scores par techniques qui s'appliquent sur différentes plateformes (Windows, Linux, etc.) en utilisant la propriété « `applicable_to` ».

Comme pour l'inventaire, on peut également générer un fichier JSON pour la visualisation sur le navigateur via la commande :

```
$ python detect.py v -ft techniques-administration.yaml -fd datasources.yaml
```

MITRE ATT&CK™ Navigator													
layer	Visibility example	Technique controls											
		Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact
11 items	27 items	42 items	21 items	57 items	16 items	22 items	15 items	13 items	21 items	9 items	14 items		
Drive-by Compromise	CMSTP	Accessibility Features	Access Token Manipulation	Account Manipulation	Account Discovery	Application Deployment Software	Audio Capture	Commonly Used Port	Automated Exploitation	Data Destruction			
Exploit Public-Facing Application	Command-Line Interface	Account Manipulation	Binary Padding	Brute Force	Application Window Discovery	Distributed Component Object	Clipboard Data	T1043	Communication Through Removable Media	Data Encrypted for Impact			
External Remote Services	Compiled HTML File	AppCert DLLs	Accessibility Features	BITS Jobs	Credential Dumping	Browser Bookmark Discovery	Data from Information Repositories	Connection Proxy	Communication That Is Compressed	Defacement			
Hardware Additions	Dynamic Data Exchange	Application Shimming	Bypass User Account Control	Credentials in Files	Domain Trust Discovery	File and Directory Discovery	Data from Local System	Custom Command and Control Protocol	Custom Data Sources	Disk Content Wipe			
Replication Through Removable Media	Execution through API	Authentication Package	Application Shimming	Code Signing	Forced Authentication	Network Service Scanning	Data from Network Shared Drive	File Sniffing	File Use of Network Protocol	Disk Structure Wipe			
Spearphishing Attachment	Execution through Module Load	BITS Jobs	Bypass User Account Control	Compile After Delivery	Hooking	Network Share Discovery	Pass the Hash	Fileless Executable	File Obfuscation	Endpoint Denial of Service			
Spearphishing Link	Exploitation for Client Execution	Bootkit	DLL Search Order Hijacking	Component Firmware	Input Capture	Network Sniffing	Pass the Ticket	Fileless Payload	Fileless Persistence	Firmware Corruption			
Spearphishing via Service	Graphical User Interface Association	Change Default File	Exploitation for Privilege Escalation	Component Object Model Hijacking	Input Prompt	Peripheral Device Discovery	Remote Desktop Protocol	Fileless Persistence	Inhibit System Recovery	Network Denial of Service			
Supply Chain Compromise	InstallUtil	Component Firmware	Extra Window Memory Injection	Control Panel Items	Kerberos	Process Discovery	Remote File Copy	Fallback Channels	Scheduled Transfer	Resource Hijacking			
Trusted Relationship	LSASS Driver	Component Object Model Hijacking	FAT System Permissions Weakness	DCShadow	LLMNR/NBT-NS Poisoning and Relay	Process Discovery	Email Collection	Shared Webhost	Screen Capture	Runtime Data Manipulation			
Valid Accounts	Malsp	Create Account	Deobfuscate/Decode Files or Information	File System	Man in the Browser	Realization Through Removable Media	Input Capture	Multi-hop Proxy	Video Capture	Service Stop			
Regvcs/Regasm	DLL Search Order Hijacking	Hooking	Disabling Security Tools	Private Keys	Network Sniffing	Remote System Discovery	Input Shared Content	Multi-Stage Channels	Multi-band Communication	Stored Data Manipulation			
Regvr32	External Remote Services	Image File Execution Options Injection	Password Filter DLL	System Software Discovery	NSSM	Security Software Discovery	System Information Discovery	Third-party Software	Multi-layer Encryption	Transmitted Data Manipulation			
Rundll32	File System Permissions Weakness	DLL Search Order Hijacking	Process Injection	Two-Factor Authentication Interception	NTDLL.dll	System Network Configuration Discovery	System Service Discovery	Windows Admin Shares	Remote Access Tools				
Scheduled Task Scripting	Hidden Files and Directories	Image File Execution Options Injection	File Deletion	File Permissions Modification	System Network Connections Discovery	System Network Connections Discovery	System Time Discovery	Windows Management	Standard Application Layer Protocol				
Signed Binary Proxy Execution	Hooking	DLL Side-loading	Port Monitors	Path Interception	Exploitation for Defense Evasion	System Owner/User Discovery	Virtualization/Sandbox Evasion	Windows Remote Management	Standard Cryptographic Protocol				
Signed Script Proxy Execution	Hypervisor	Image File Execution Options Injection	Extra Window Memory Injection	Process Injection	File System Logical Offsets	System Service Discovery			Standard Non-Application Layer Protocol				
Third-party Software Trusted Developer Utilities	Logon Scripts	SID-History Injection	Group Policy Modification	Hidden Files and Directories	Group Policy Modification	System Time Discovery			Uncommonly Used Port				
User Execution	LSASS driver	Valid Accounts	Image File Execution Options Injection	Web Shell	Indicator Blocking	Virtualization/Sandbox Evasion			Web Service				
Windows Management Instrumentation	Netapi Helper DLL	New Service											
Windows Remote Management	Office Application	New Service											

Figure 42 - Matrice couverture défensive 2

Dans le fichier d'administration YAML des techniques, il existe un champ appelé « **score_logbook** » qui permet de mettre à jour la visibilité d'une façon automatique. Si l'on ajoute une nouvelle source de données on peut utiliser ce champs pour mettre à jour le fichier et par la suite, avoir un historique de l'évolution de la couverture.

On peut alors choisir de mettre à jour les scores de visibilité approximatifs dans le fichier d'administration des techniques avec la commande :

```
$ python detect.py ds -ft techniques-administration.yaml -fd datasources.yaml --update
```

Enfin, il ne reste plus qu'à déterminer la couverture de détection finale en complétant le score de visibilité par un score de détection par technique dans le fichier YAML des techniques. Pour cette dernière étape, les scores vont de -1 à 5 (**non, détection absente mais les techniques sont loggées pour des fins de forensiques, basique, moyen, bien, très bien et excellent**).

Comme pour la visibilité, on retrouve le « **score_logbook** » pour le traçage et l'objet « **applicable_to** ». La notation finale de la visibilité et la détection est comme suivant :

```
- technique_id: T1204
  technique_name: User Execution
  detection:
    applicable_to: [all]
    location: [EDR]
    comment: ''
    score_logbook:
      - date: 2018-12-01
        score: 0
        comment: ''
  visibility:
    applicable_to: [all]
    comment: ''
    score_logbook:
      - date: 2019-03-01
        score: 2
        comment: ''
    auto_generated: true
```

Figure 43 - Extrait d'un fichier d'administration YAML DeTT&CT

Pour obtenir la visualisation de la matrice de détection finale :

\$ python detect.py d -ft techniques-administration.yaml -l

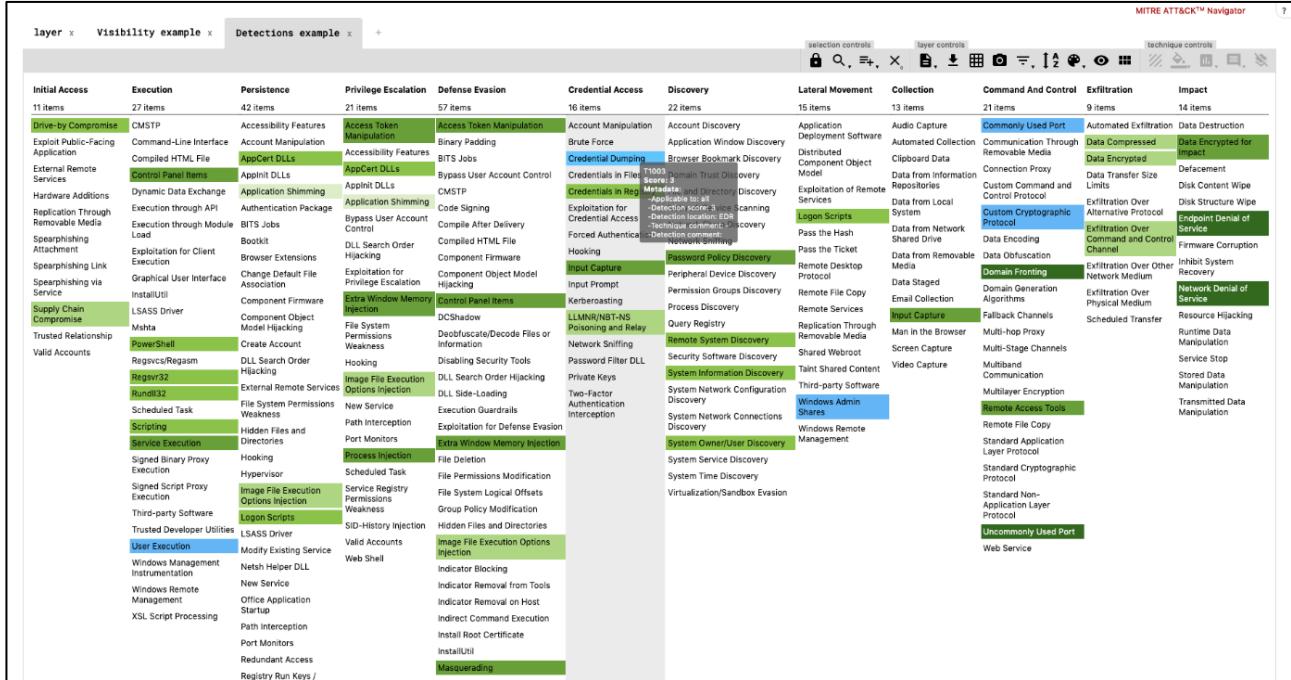


Figure 44 - Matrice couverture défensive 3

La matrice de détection obtenue est la forme la plus précise que l'on peut avoir avec le Framework DeTT&CT si la notation a été bien faite. A noter qu'il est possible de générer plusieurs matrices pour des environnements différents en spécifiant la plateforme visée.

Les dates enregistrées lors de la mise en œuvre de nouvelles détections peuvent être utilisées pour générer un graphique sur le nombre de détections ajoutées au fil du temps.

\$ python detect.py d -ft techniques-administration.yaml -g

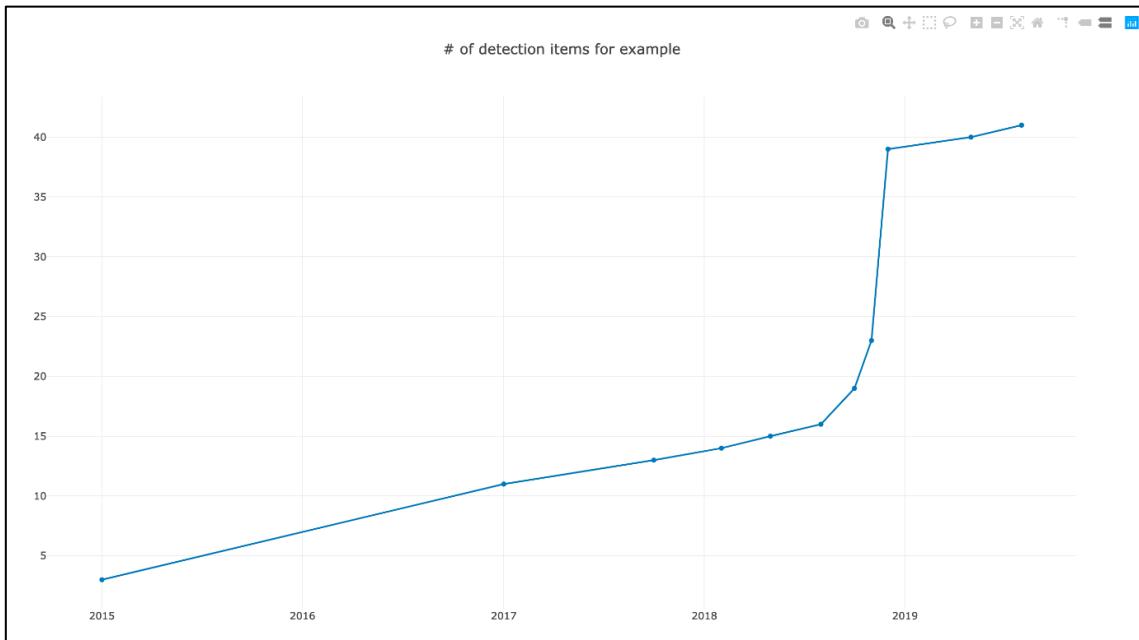


Figure 45 - Evolution du nombre de techniques détectées

4.3.3 Prioriser

Avec les informations collectées sur la détection et les lacunes trouvées et avant de commencer la mitigation, il faut prioriser les tâches et trouver le moyen le plus optimal afin de mettre en place un plan d'amélioration. Plusieurs méthodes sont possibles pour la priorisation :

4.3.3.1 Les groupes menaçants :

- Prioriser selon les techniques utilisées par les groupes APT les plus impactants pour son entreprise :

Comme vu dans précédemment, on peut affiner la détection en prenant en compte les techniques les plus utilisées par les groupes d'attaquants susceptibles d'impacter le SI. On va donc calquer la matrice de détection obtenue précédemment sur la matrice des techniques des groupes identifiées. Le résultat montrera les techniques qui sont restent sans couverture et qui sont à prioriser lors du processus d'amélioration.

Gaps													
filters													
stages: act platforms: Windows, Linux, macOS													
Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact		
Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact		
11 items	34 items	62 items	69 items	21 items	23 items	18 items	13 items	22 items	9 items	16 items			
Drive-by-Compromise	AppleScript	.bash profile and bashrc	Access Token	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Community Used Port	Automated Exfiltration	Account Access Removal			
Exploit Public-Facing	CMSTP	Accessibility Features	Accessibility Features	Binary Padding	Application Window	Application Deployment	Communication Through Port	Data Compressed	Data Destruction				
External Remote	Command-Line Interface	Account Manipulation	AppCert DLLs	Binaries	Brute Force	Browser Bookmark	Clipboard Data	Connection Proxy	Data Encrypted for Transfer	Data Recovery			
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	ByPass User Account	Credential Dumping	Domain True Discovery	Exploitation of Remote File and Directory	Data from Information Custom Command and Control	Data Transfer Size	Defacement			
Impersonation	Java Macro Script Mode	AppInit DLLs	Application Shimming	Clear Command History	Downloads from Web	Internal Spearphishing	Data from Local System	Logon Scripts	Logon Cryptography	Malicious Payload	Disk Content Wipe		
API Exploiting	Control Panel Items	Application Shimming	Portable User Account	CMSTP	Credentials in Files	Network Share Discovery	Data from Network	Data Encoding	Network and Control	Physical Overwrite	Disk Structure Wipe		
Spearphishing Link	Dynamic Data Exchange	Authentication Package	HTTP Cache Order	Code Signing	Credentials Registry	Pass the Hash	Data from Removable Media	Data Obfuscation	General Methods	Physical Denial of Service	Power Outage		
Service Exploiting	Execution through API	HTTP Cache Order	HTTP Hijacking	Compile After Delivery	Credential Access	Pass the Ticket	Data Staged	Domain Fronting	Protocol Over Physics	Power Outage	Power Outage		
Service Exploiting via Configuration Chain	BIT5 jobs	Job Scheduler	JobScheduler	Compiled HTML File	Forced Authentication	Perceived Device	Protocol Port	Remote File Copy	Protocol Port	Protocol Port	Protocol Port	Protocol Port	
Trusted Relationship	Client Certificate	Code Injection	Code Injection with Bypass	Component Firmware	Input Capture	Peripheral Device	Protocol Port	Rootkit	Rootkit	Rootkit	Rootkit	Rootkit	
Valid Accounts	Container User Interface	Container User Interface	Container User Interface	Component Object Model	Connection Proxy	Process Discovery	Protocol Port	Screen Capture	Screen Capture	Screen Capture	Screen Capture	Screen Capture	
InstallUtil	Component Firmware	Extra Window Memory	Extra Window Memory	Control Panel Items	Kerberoasting	Query Registry	Protocol Port	Shared Webroot	Video Capture	Video Capture	Video Capture	Video Capture	
LaunchCtl	Component Object Model	File System Permissions	File System Permissions	DCShadow	Keychain	Remote System	Protocol Port	Service Stop	Service Stop	Service Stop	Service Stop	Service Stop	
Local Job Scheduling	Create Account	Hooking	Hooking	DISBURSE Predecide	LLMNR/NBNS Relay	Security Software	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
LSASS Driver	DLT Search Order	Inject File Execution	Inject File Execution	DISBURSE Predecide	LLMNR/NBNS Relay	Taint Shared Content	Protocol Port	Service Stop	Service Stop	Service Stop	Service Stop	Service Stop	
Msihta	Dllvir Hijacking	Launch Daemon	Disabling Security Tools	Network Sniffing	Software Discovery	Third-party Software	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
PowerShell	Emond	New Service	DLL Search Order	Password Filter DLL	System Discovery	Third-party Software	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Regsvcs/Regasim	External Remote	Parent PID Spoofing	Parent PID Spoofing	Dll Side-Loading	Private Keys	Windows Admin Shares	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Regsvr32	File System Permissions	Path Interception	Path Interception	Execution Guards	Securirty Memory	Windows Remote Management	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Rundll32	Hidden Files and Folders	Plist Modification	Plist Modification	Exploitation for Defense	Shared Webroot	Windows Admin Shares	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Scheduled Task	Hooking	Port Monitors	Port Monitors	Exploitation for Defense	Shared Webroot	Windows Admin Shares	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Scripting	Hypervisor	PowerShell Profile	PowerShell Profile	Process Modification	Port Modification	Windows Admin Shares	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Service Execution	Process Execution	Process Injection	Process Injection	File Deletion	System Information	Windows Admin Shares	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Service Exploiting	Windows Services and Applications	Scheduled Task	Scheduled Task	File System Logical	System Network Discovery	Windows Remote Management	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Service Exploiting via Configuration Chain	Service Exploiting via Configuration Chain	Startup	Startup	Group Policy Modification	System Network Discovery	Windows Remote Management	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Source	Launch Daemon	Setuid and Setgid	Setuid and Setgid	Gatekeeper Bypass	System Network Discovery	Windows Remote Management	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Space after Filename	Logon Scripts	SD-10 History Injection	SD-10 History Injection	Group Policy Modification	System Network Discovery	Windows Remote Management	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Third-party Software	LC LOAD DLL	Startup Items	Startup Items	Histogram Users	System Network Discovery	Windows Remote Management	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Trap	Local Job Scheduling	Sudo	Sudo	Hidden Window	HISTCONTROL	Windows Remote Management	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Trusted Developer	Login Item	Sudo Caching	Sudo Caching	LC MAIN Hijacking	Masquerading	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
User Execution	Logon Scripts	Valid Accounts	Valid Accounts	Indicator Blocking	Modify Registry	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Windows Management	LSASS Driver	Web Shell	Web Shell	Indicator Removal from Tasklist	Mshta	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
Windows Helpdesk	Modify Existing Service			Indicators Removal from Tasklist	Netshell Helper DLL	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
XSL Script Processing	Netshell Helper DLL			Indicators Removal from Tasklist	Office Application Startup	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Path Interception	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Plist Modification	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Port Knocking	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Process Dupeljäning	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Process Hollowing	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Process Injection	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Redundant Access	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Re-opened Applications	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Remote Access	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Remote Access	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	ScreenSaver	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Security Support	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Server Software	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Service Repairs	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Session Hijackers	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Setuid and Setgid	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Shortcut Modification	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	SH and Trust Provider	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Startup Items	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	System Firmware	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Systemd Service	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Rundll32	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Time Providers	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Trap	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Valid Accounts	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	
				Indicators Removal from Tasklist	Winlogon Helper DLL	Windows Service	Protocol Port	Service Start	Service Start	Service Start	Service Start	Service Start	

Figure 46 - Identification des lacunes

4.3.3.2 Les techniques fréquentes

- Construire une carte thermique « Heatmap » des techniques les plus utilisées par tous les groupes :**

Cela est possible avec DeTT&CT en générant un fichier JSON basé sur les données mappées dans ATT&CK sur les groupes d'attaquants. On tape la commande :

```
$ python detect.py g
```

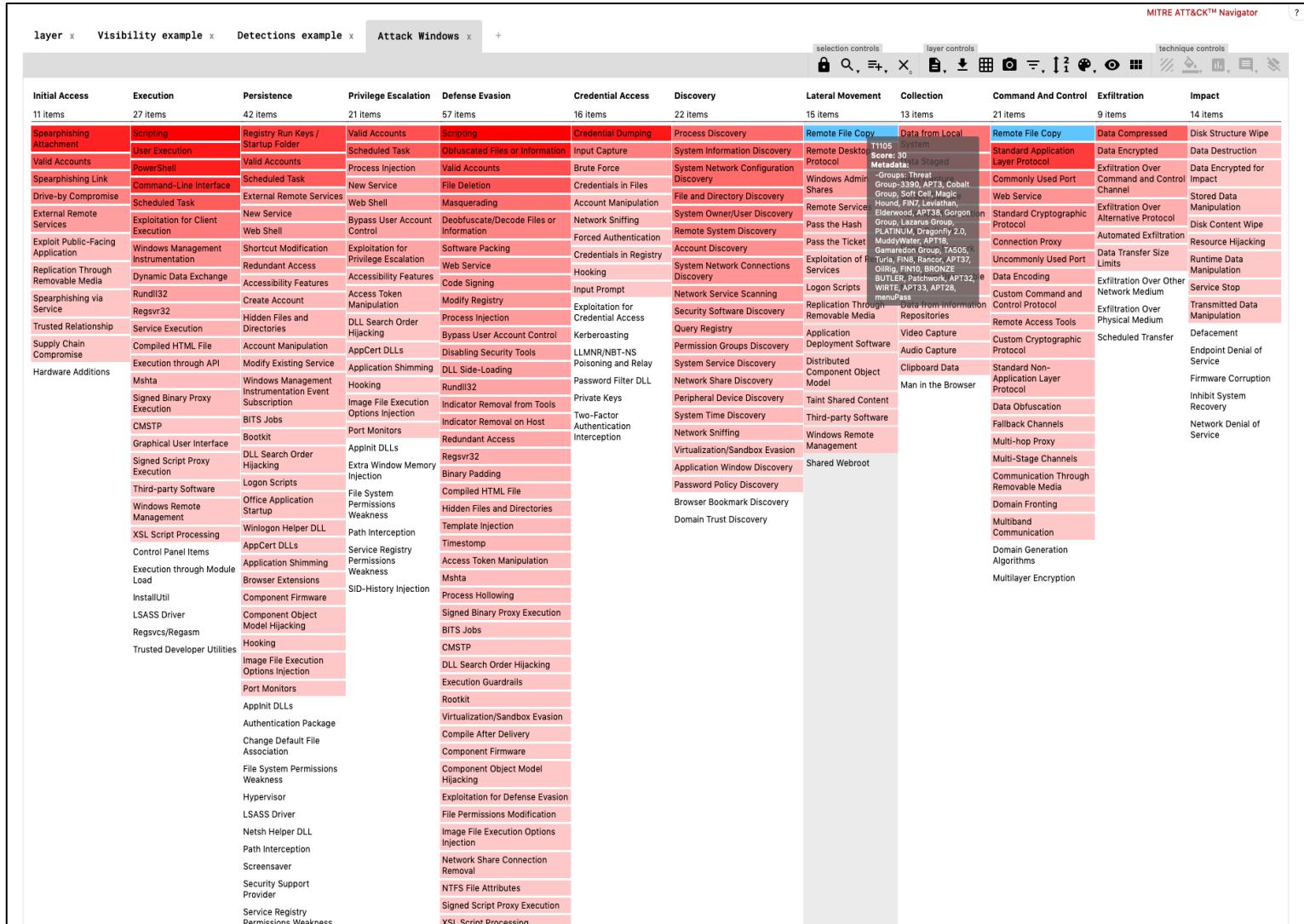


Figure 47 - Matrice des techniques les plus utilisées

Plus la couleur de la carte est foncée, plus la technique est utilisée au sein des groupes. Enfin, il ne reste plus qu'à superposer la matrice de détection sur cette matrice afin de visualiser au mieux la situation actuelle de la couverture.

4.3.3.3 Examiner les sources d'information

- **Examiner les sources d'information communes aux différentes techniques :**

J'ai créé un script Python basé sur la librairie « Attack Python Client » qui collecte les données des matrices ATT&CK à partir du serveur TAXII de MITRE. Disponible sur GitHub (également en « Annexes ») : <https://github.com/Azrara/MITRE-ATTACK-Utils>

Note : Le script reprend les « notebooks » qui viennent avec la librairie « Attack Python Client » : <https://github.com/hunters-forge/ATTACK-Python-Client/tree/master/notebooks>

Le script peut générer 3 graphiques selon l'option souhaitée :

1. Compter le nombre de source de données par technique :

Commande :

```
$ python3 datasources_techniques.py - --visualizaiton count
```

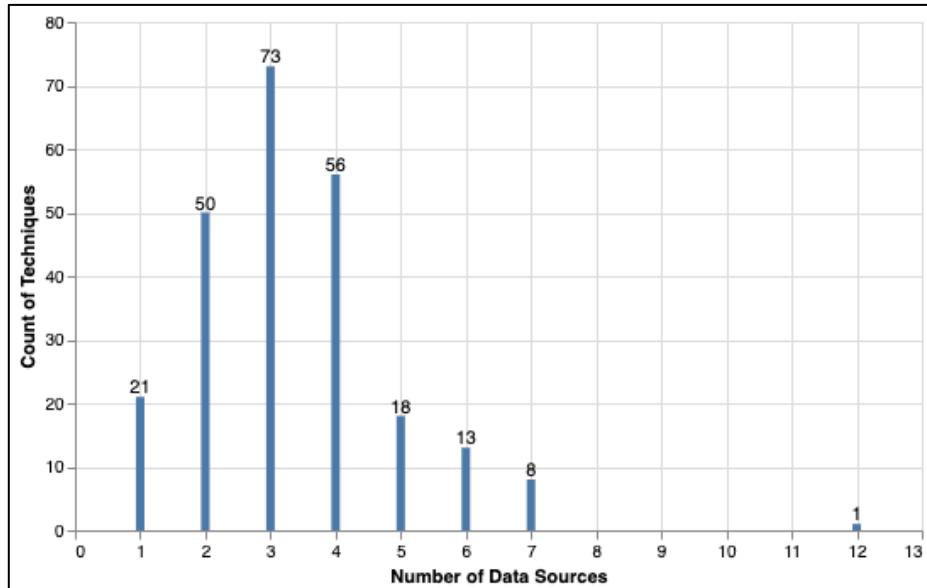


Figure 48 - Nombre de sources de données par technique

Le graphique résultant, montre le nombre de sources de données requises par technique.

On peut remarquer qu'il y a 73 techniques qui nécessitent 3 sources de données comme contexte suffisant pour valider leur détection. On a aussi une source de données qui couvre 21 techniques.

2. Compter le nombre de techniques couvertes par chaque source d'information :

Commande :

```
$ python3 datasources_techniques.py -v visualization grouping
```

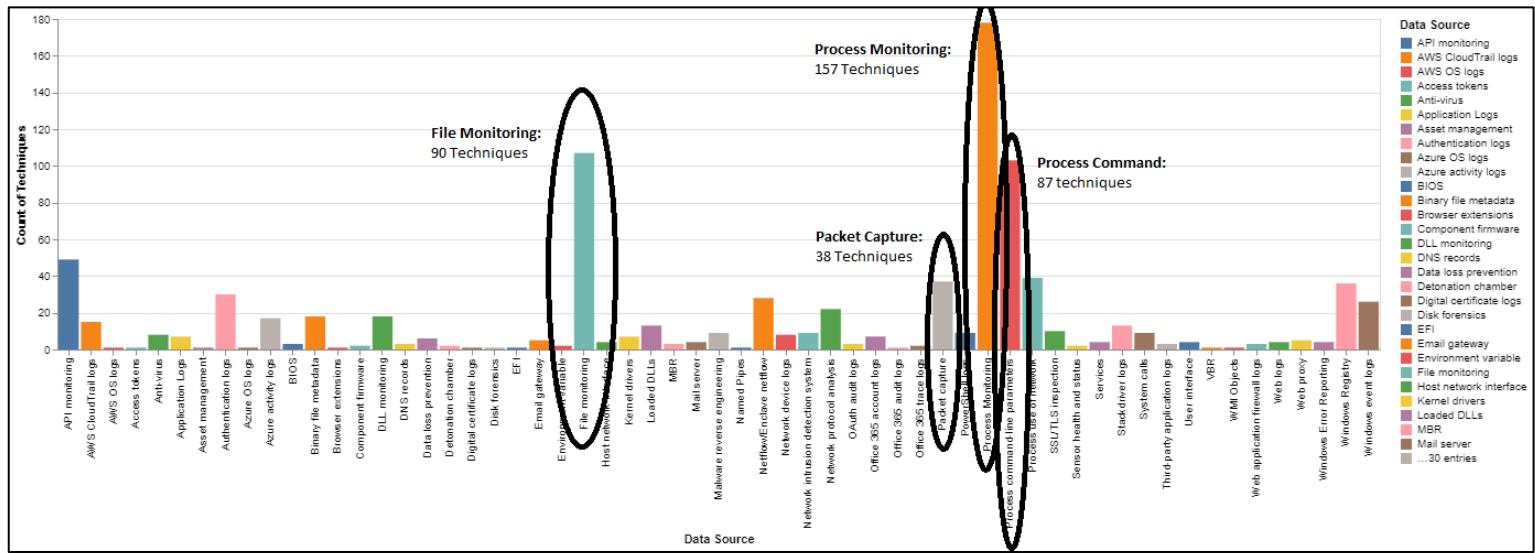


Figure 49 - Nombre de techniques couverte par source de données

Le graphique montre que plusieurs sources sont précieuses pour détecter un grand nombre de techniques qui sont le monitoring :

- Des **processus, fichiers et registres** qui peut couvrir 80% des techniques.
- Des différents logs d'**authentification**.
- Des **paquets réseaux**.

3. Regrouper les sources d'information :

Presque toutes les sources de données de haut niveau mentionnées dans ATT&CK sont constituées de sous-données (différentes formes de cette source de données) qui peuvent être classées dans des groupes.

Par exemple, on peut grouper la supervision des paquets réseaux avec les processus faisant des communications réseaux.

Commande :

```
$ python3 datasources_techniques.py -visualizaiton subset
```

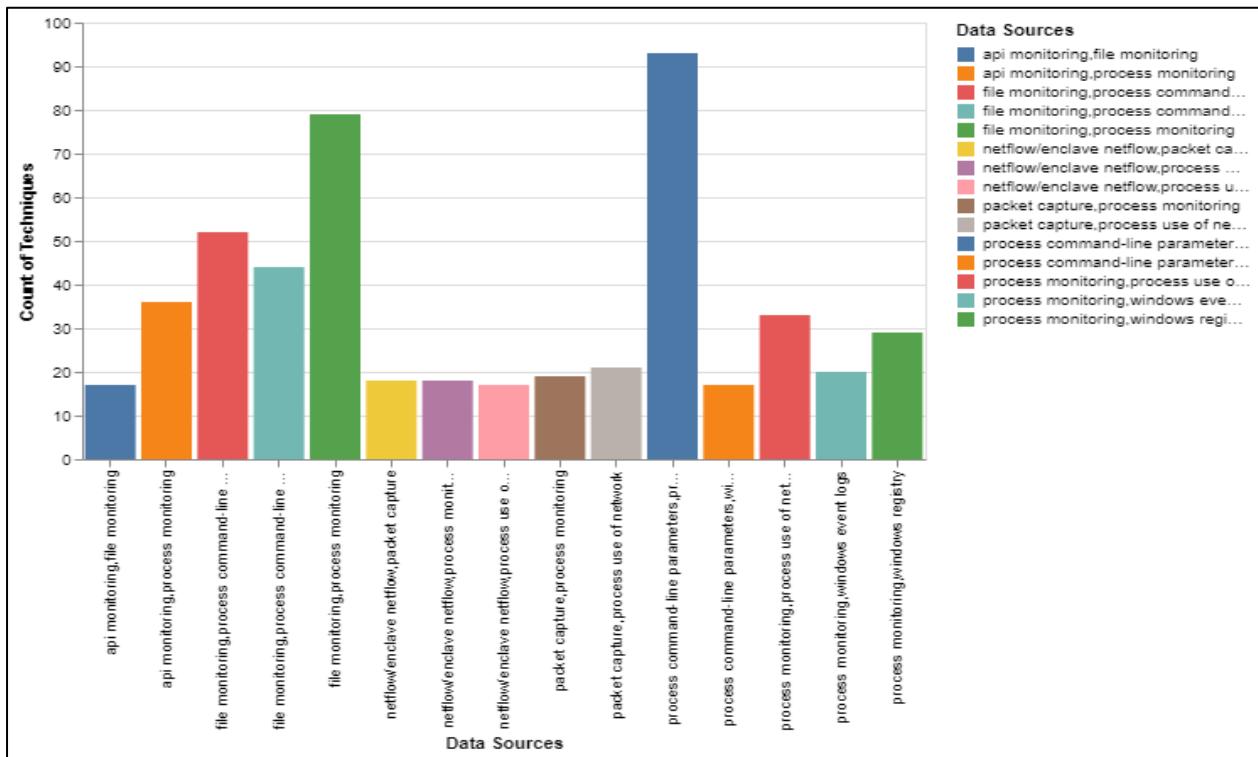


Figure 50 - Nombre de techniques par sous-données

On en déduit que le groupe : **Process Monitoring – Process Command-line parameters** est le groupe de sources de sous-données ayant le plus grand nombre de techniques. **Ce groupe est suggéré pour chasser plus de 90 techniques.**

Néanmoins, surveiller les processus et les fichiers peut très vite générer un grand volume de logs spécialement lorsqu'il s'agit d'un grand parc informatique. C'est pourquoi il faut optimiser la collecte de logs en se concentrant uniquement sur les processus les plus pertinents et qui peuvent couvrir les techniques les plus dangereux pour un système d'information donné.

Par exemple, cela peut dépendre des variations dans la façon de l'exécution des techniques. La technique « Overpass-the-hash » peut être exécutée et détectée à l'aide de deux méthodes différentes : **Mimikatz** et **Rubeus**.

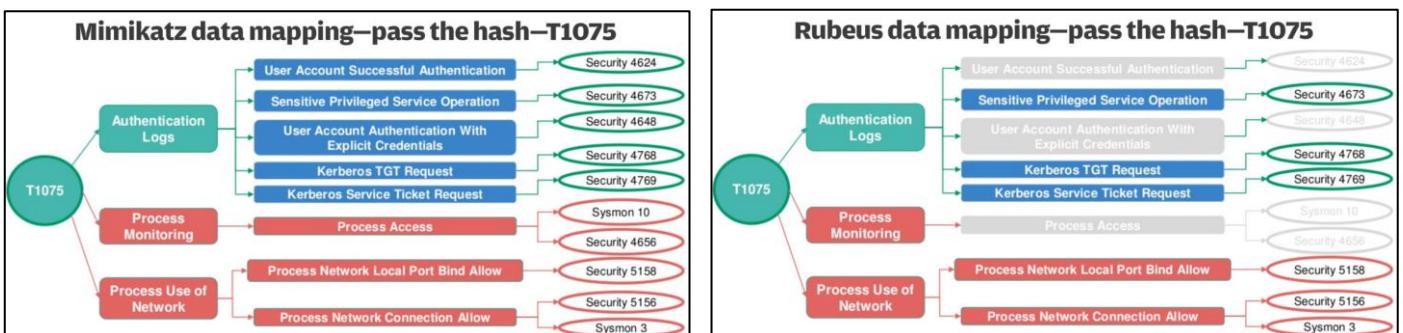


Figure 51 - Variation d'exécution d'une technique source : <https://github.com/Cyb3rWard0g/presentations/tree/master/ATTACKcon>

L'intérêt de montrer le sous-ensemble des sources de sous-données est que l'on peut se concentrer sur une version plutôt que sur l'autre si les renseignements sur la menace indiquent que l'une d'entre elles est plus pertinente pour l'environnement de son entreprise. Cela permettra de faire sortir les détections plus efficacement.

Enfin la feuille de calcul suivante montre davantage de sources de données ainsi que leurs sous-composants : [ATT&CK Data Modeling](#)

4.3.3.4 Améliorer

Une fois que l'on sait quelles sont les données dont on dispose, et que l'on a récolté celles manquantes, on doit les rassembler dans un SIEM (Security Information and Event Management) afin d'effectuer des analyses de données. La prochaine étape est de mettre en place des règles de détection.

MITRE fournit un répertoire d'*analytics* et de pseudocodes qui peuvent contribuer à la constitution des règles de détection. Ce répertoire est appelé CAR (Cyber Analytics Repository) et il est mis à jour régulièrement. Il comporte un grand nombre de règles : <https://car.mitre.org/>

Les *analytics* stockées dans CAR contiennent les informations suivantes :

- Une hypothèse qui explique l'idée derrière l'analyse.

CAR-2019-07-001: Access Permission Modification	
<p>Adversaries sometimes modify object access rights at the operating system level. There are varying motivations behind this action - they may not want some files/objects to be changed on systems for persistence reasons and therefore provide admin only rights; also, they may want files to be accessible with lower levels of permissions.</p> <p>Note - this analytic references file permissions, which are not currently in the CAR data model.</p>	<p>Submission Date: 2019/07/08 Information Domain: Host Data Subtypes: File Analytic Type: Situational Awareness Applicable Platforms:</p>

Figure 52 - Hypothèse CAR

- Des informations sur la plateforme ciblée, domaine d'information et le type de source de données.

Submission Date: 2019/07/08
Information Domain: Host
Data Subtypes: File
Analytic Type: Situational Awareness
Applicable Platforms: Windows, Linux, macOS
Contributors: Meric Degirmenci, MITRE

Figure 53 - Tactiques et techniques CAR

- Les références aux techniques et tactiques ATT&CK que l'analyse détecte.

Technique	Subtechnique(s)	Tactic(s)	Level of Coverage
File and Directory Permissions Modification	Windows File and Directory Permissions Modification, Linux and Mac File and Directory Permissions Modification	Defense Evasion	Moderate

Figure 54 - Informations sur la plateforme ciblée

- Une description pseudocodée ou qui cible un SIEM particulier de la manière dont l'analyse pourrait être mise en œuvre.

Windows - Pseudocode (Pseudocode)

Windows environment logs can be noisy, so we take the following into consideration:

- We need to exclude events generated by the local system (subject security ID "NT AUTHORITY\SYSTEM") and focus on actual user events.
- When a permission modification is made for a folder, a new event log is generated for each subfolder and file under that folder. It is advised to group logs based on handle ID or user ID.
- The Windows security log (event ID 4670) also includes information about the process that modifies the file permissions. It is advised to focus on uncommon process names, and it is also uncommon for real-users to perform this task without a GUI.

```
log_name == "Security" AND
event_code == "4670" AND
object_type == "File" AND
subject_security_id != "NT AUTHORITY\SYSTEM"
```

Windows - Splunk (Splunk)

Splunk version of the above pseudocode.

```
index=__your_windows_security_log_index__ EventCode=4670 Object_Type="File" Security_ID!="NT AUTHORITY\SYSTEM"
```

Figure 55 - Exemple de pseudocode CAR

- Un test unitaire qui peut être effectué pour déclencher l'analyse.

Unit Tests

Test Case 1

For Windows - right click on any file and change its permissions under properties. Or, execute the following command: `icacls "C:\<fileName>" /grant :F`

Test Case 2

For Linux - execute the following command: `chmod 777 "fileName"`

Figure 56 - Test unitaire CAR

En plus des analyses, CAR contient également des modèles de données pour les données observables utilisées pour l'exécution des analyses et les capteurs qui sont utilisés pour collecter ces données.

Le modèle de données est une organisation d'objets qui peuvent être surveillés dans une perspective d'hôte ou de réseau. Chaque objet peut être identifié avec deux dimensions : ses **actions** et ses **champs**.

Lorsqu'ils sont appariés, les 3 champs (**objet**, **actions**, **champs**) agissent comme des coordonnées et décrivent les propriétés et les changements d'état de l'objet qui peuvent être enregistrés par un capteur (exemple : Windows autoruns).

Si l'on prend l'exemple de l'objet « fichier » on aura le modèle suivant :



Figure 57 - Exemple de modèle CAR

Il est possible de naviguer ces données et d'avoir une vision sur leur impact sur les techniques et les groupes grâce à CARET : <https://mitre-attack.github.io/caret/#/>. Il s'agit d'une application web développée à la base pour le gouvernement américain.

Un exemple sur l'analyse des activités RPC (Remote Procedure Call) et les techniques et groupes couverts :

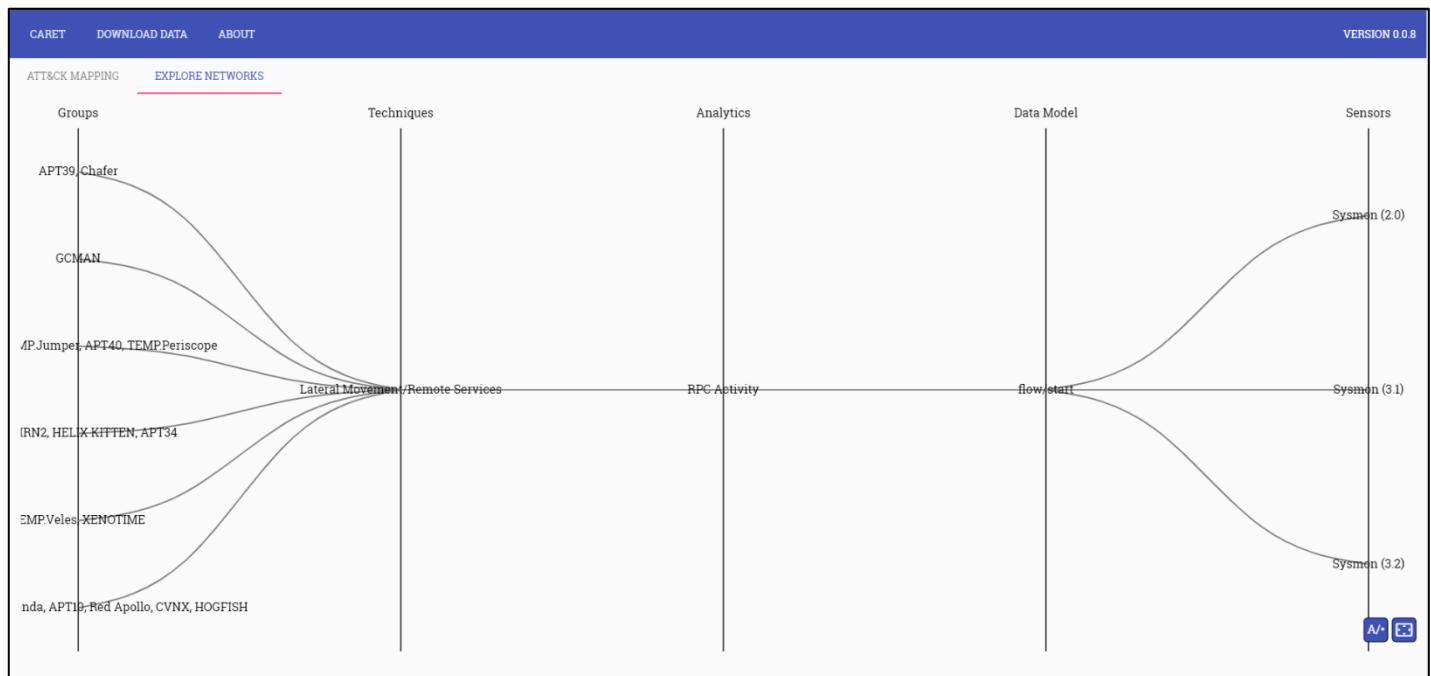


Figure 58 - Exemple navigation CARET

Afin de mettre en place une analyse il faut se concentrer sur le pseudocode donné dans chaque analyse CAR. Ce pseudocode est à traduire en un langage de recherche qui correspond au SIEM utilisé en s'assurant que les noms des champs des données sont corrects.

La traduction n'est pas tout le temps évidente, pour cela il existe un langage open source appelé SIGMA : <https://github.com/Neo23x0/sigma>



Figure 59 - Logo de SIGMA

Sigma est un format de signature générique qui permet de décrire de manière simple les logs. Il s'agit d'un répertoire contenant des règles écrites en YAML qui est un format très simple à écrire et à comprendre par un humain.

Si l'on prend l'analyse **CAR-2016-03-002** qui concerne la création de processus via WMIC (Windows Management Instrumentation) utilisé par les attaquants pour effectuer des mouvements latéraux, on pourra trouver son équivalence dans les règles SIGMA :

```
title: Suspicious WMI Execution
id: 526be59f-a573-4eea-b5f7-f0973207634d
status: experimental
description: Detects WMI executing suspicious commands
references:
  - https://digital-forensics.sans.org/blog/2010/06/04/wmic-draft/
  - https://www.hybrid-analysis.com/sample/4be06ecd234e2110bd615649fe4a6fa95403979acf889d7e45a78985eb50acf9?environmentId=1
  - https://blog.malwarebytes.com/threat-analysis/2016/04/rokku-ransomware/
author: Michael Haag, Florian Roth, juju4
date: 2019/01/16
logsource:
  category: process_creation
  product: windows
detection:
  selection:
    Image:
      - '*\wmic.exe'
    CommandLine:
      - '/NODE:*process call create *'
      - '* path AntiVirusProduct get *'
      - '* path FirewallProduct get *'
      - '* shadowcopy delete *'
  condition: selection
fields:
  - CommandLine
  - ParentCommandLine
tags:
  - attack.execution
  - attack.t1047
  - car.2016-03-002
falsepositives:
  - Will need to be tuned
  - If using Splunk, I recommend | stats count by Computer,CommandLine following for easy hunting by Computer/CommandLine.
level: medium
```

Figure 60 - Exemple de règle SIGMA

A partir de ce format, on peut générer des règles pour différents types de SIEM (QRadar, Splunk etc.) en utilisant l'outil de conversion fourni avec SIGMA appelé « **sigmac** ».

Pour lancer la conversion de la règle précédente on doit spécifier le SIEM ciblé (option **-t**), la plateforme (option **-c**) et la règle en YAML.

On va supposer que l'on veut convertir la règle vers un forma de règle Splunk.

```
$ python tools/sigmac -t splunk -c splunk-windows win_susp_wmi_execution.yml
```

On aura le résultat suivant qui correspond à la traduction de la règle SIGMA dans le langage SPL (Search Processing Language) utilisé requêter dans Splunk :

```
((Image="*\lwmic.exe") (CommandLine="*/NODE:*process call create **" OR CommandLine="* path AntiVirusProduct get *" OR CommandLine="* path FirewallProduct get *" OR CommandLine="* shadowcopy delete *"))
| table CommandLine,ParentCommandLine
```

Une carte thermique ATT&CK peut être générée à partir d'un répertoire contenant des règles SIGMA en exécutant la commande :

```
$ python sigma2attack
```

Cette commande prendra les règles par défaut et créera un fichier JSON qui comporte les techniques couvertes par les règles. On pourra également spécifier des règles personnalisées en ajoutant l'option « **-rules-directory** » et en donnant le répertoire qui contient les règles.

Cela permettra d'évaluer rapidement le niveau de couvertures des règles créées ou qui fournies par défaut par SIGMA.

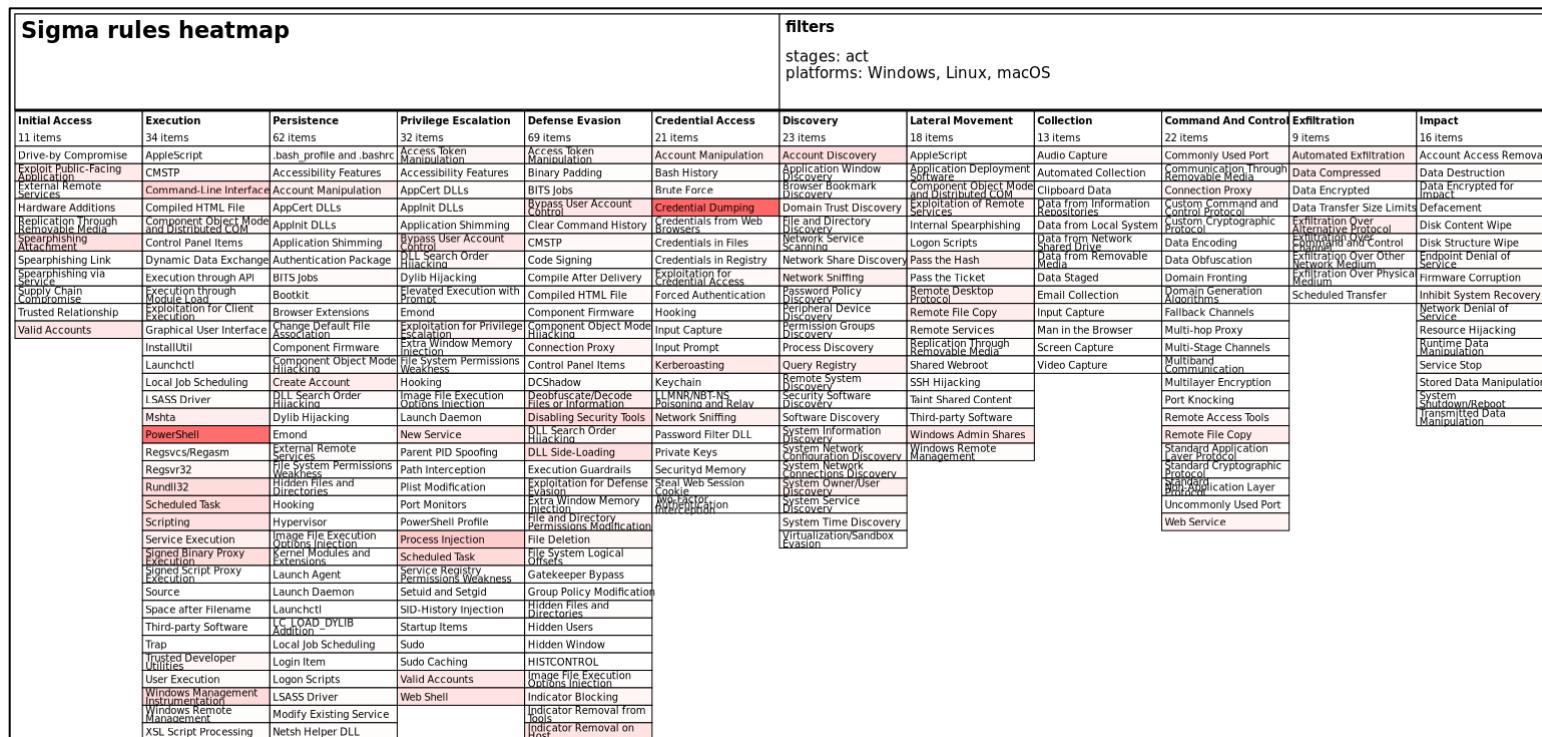


Figure 61 - Carte thermiques des techniques couvertes par les règles SIGMA

Enfin, pour tester les règles il existe un outil appelé « Atomic Red Teaming » qui exécute de simples « test atomiques » qui font appel aux mêmes techniques utilisées par les attaquants.

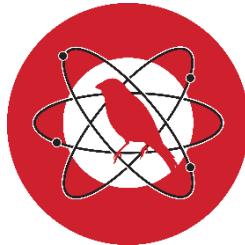


Figure 62 - Logo d'Atomic Red Teaming

Les tests passent par 5 étapes :

- Sélectionner un test
- Exécuter le test
- Recueillir des preuves
- Développer la détection
- Mesurer le progrès

On commence par sélectionner les tests qui correspondent aux règles créées.

Un grand panel de tests est offert par Atomic Red Teaming :

[https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/Indexes-
Markdown/index.md](https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/Indexes-Markdown/index.md)

Ensuite, on exécute un test choisi. Par exemple, on prend la technique T1117 «Regsvr32 »¹ :

T1117 - Regsvr32

Description from ATT&CK

Regsvr32.exe is a command-line program used to register and unregister object linking and embedding controls, including dynamic link libraries (DLLs), on Windows systems. Regsvr32.exe can be used to execute arbitrary binaries. (Citation: Microsoft Regsvr32)

Adversaries may take advantage of this functionality to proxy execution of code to avoid triggering security tools that may not monitor execution of, and modules loaded by, the regsvr32.exe process because of whitelists or false positives from Windows using regsvr32.exe for normal operations. Regsvr32.exe is also a Microsoft signed binary.

Regsvr32.exe can also be used to specifically bypass process whitelisting using functionality to load COM scriptlets to execute DLLs under user permissions. Since regsvr32.exe is network and proxy aware, the scripts can be loaded by passing a uniform resource locator (URL) to file on an external Web server as an argument during invocation. This method makes no changes to the Registry as the COM object is not actually registered, only executed. (Citation: LOLBAS Regsvr32) This variation of the technique is often referred to as a "Squiblydoo" attack and has been used in campaigns targeting governments. (Citation: Carbon Black Squiblydoo Apr 2016) (Citation: FireEye Regsvr32 Targeting Mongolian Gov)

Regsvr32.exe can also be leveraged to register a COM Object used to establish Persistence via [Component Object Model Hijacking](#). (Citation: Carbon Black Squiblydoo Apr 2016)

Atomic Tests

- [Atomic Test #1 - Regsvr32 local COM scriptlet execution](#)
- [Atomic Test #2 - Regsvr32 remote COM scriptlet execution](#)
- [Atomic Test #3 - Regsvr32 local DLL execution](#)

Figure 63 - Exemple de test atomique

¹ Une commande pour exécuter les DLL sur Windows utilisée par les attaquant pour légitimer l'exécution de leurs fichiers malicieux.

On va alors émuler l'exécution de cette technique et tester les capacités du système de détection à capter ce comportement :

```
$ regsvr32.exe /s /u /i: https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1117/RegSvr32.sct scrobj.dll
```

Après exécution, on surveille la réaction du SIEM et des règles de détection en observant (pour cet exemple) :

- Une modification de fichier dans le profil de l'utilisateur.
- Les connexions réseau effectuées par regsvr32.exe vers une IP externe.
- Une entrée dans les journaux du proxy.
- Le chargement de scrobj.dll sur Windows.

Ou on peut ne pas observer de comportement sur l'hôte ou le réseau. Dans ce cas, il est temps d'essayer de détecter cet événement en améliorant les règles ou en collectant les données manquantes pour la détection.

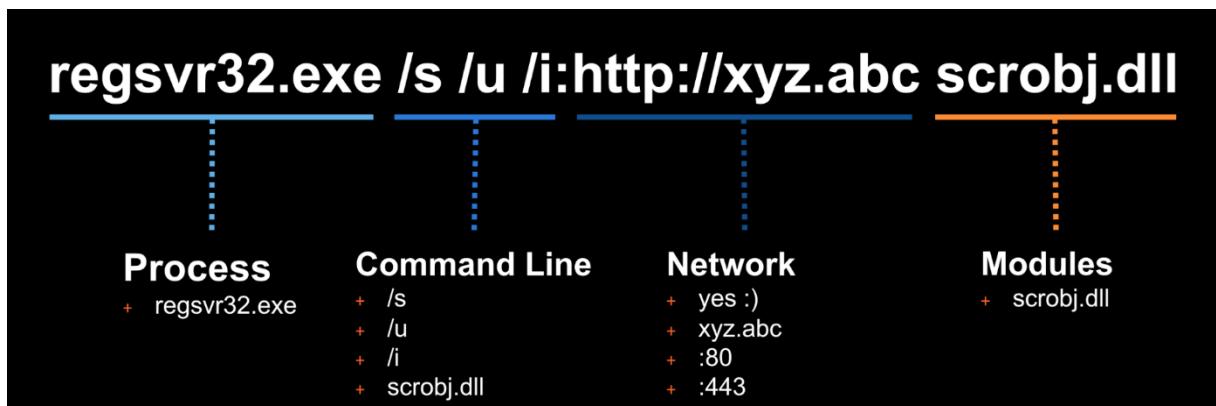


Figure 64 - Données qu'il faut pour détection Regsvr32

4.4 ATT&CK et chasse aux menaces

MITRE ATT&CK intervient aussi dans le chasses aux menaces (Threat Hunting). En effet, mettre en place des règles de détection s'avère insuffisant et les angles morts peuvent exister. C'est pour cela qu'il faut partir à la chasse des menaces en analysant les comportements anormaux qui suscitent une analyse approfondie.

L'analyse des sources de données et leur impact sur la détection a montré que si l'on surveille les processus et fichiers on couvrira un grand nombre de techniques. Donc, il est intéressant de fouiller dans ces informations en complément des règles afin d'identifier des événements anormaux que la détection a pu rater.

Pour logger les activités liées aux processus et registres dans un système Windows, Microsoft fournit Sysmon (System Monitor) qui est un service capable d'enregistrer l'activité du système dans le journal d'événements Windows :

<https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon>

En fonction de la configuration fournit, Sysmon est en mesure de logger les évènements suivants :

- Création de processus : avec ligne de commande complète et hachage.
- Arrêt des processus.
- Accès à la mémoire des processus.
- Connexions réseau.
- Activités liées aux répertoires/fichiers.
- Evénements de la WMI.
- Accès aux registres.

Ces événements sont enregistrés dans le journal d'événements dans : **Microsoft-Windows-Sysmon/Operational**.

Sysmon n'enregistre pas tous les événements par défaut, c'est pourquoi son fichier de configuration doit être modifié et personnalisé. Le fichier de configuration est écrit en XML et contient un ensemble de règles qui filtrent les données à logger. Il faut tuner le fichier au mieux afin de réduire les faux positifs et ne pas générer une grande quantité de logs.

Ci-après un fichier de configuration simplifié :

```
<sysmon schemaversion="1.0">
<configuration>
    <!-- Capture MD5 Hashes -->
    <hashing>MD5</hashing>
    <!-- Enable network logging -->
    <network />
</configuration>
<rules>
    <!-- Log all drivers except if the signature -->
    <!-- contains Microsoft or Windows -->
    <driverLoad default="include">
        <signature condition="contains">microsoft</signature>
        <signature condition="contains">windows</signature>
    </driverLoad>
    <!-- Do not log process termination -->
    <processTerminate />
    <!-- Exclude certain processes that cause high event volumes -->
    <processCreate default="include">
        <image condition="contains">chrome.exe</image>
    </processCreate>
    <!-- Do not log file creation time stamps -->
    <fileCreateTime />
    <!-- Do not log network connections of a certain process or port -->
    <networkConnect default="include">
        <image condition="contains">chrome.exe</image>
        <destinationPort>123</destinationPort>
    </networkConnect>
</rules>
</sysmon>
```

Figure 65 - Fichier de configuration Sysmon

Le projet « sysmon-config » de SwiftOnSecurity sur GitHub fait gagner beaucoup de temps dans le développement de ce fichier de configuration et propose une version assez optimisée : <https://github.com/SwiftOnSecurity/sysmon-config>

L'installation se fait avec la commande suivante en spécifiant le fichier de configuration :

```
$ sysmon.exe -accepteula -i sysmonconfig-export.xml
```

Une fois installé, il est temps d'exploiter les informations collectées par Sysmon. Il existe une application qui s'ajoute au SIEM Splunk et qui joue le rôle d'un EDR (Endpoint Detection & Remediation) basique.

L'application « Threat Hunting app » a été créée par le chercheur en sécurité Olaf Hartong, il est possible de la télécharge sur le store de Splunk ou bien en important le répertoire GIT :

Store Splunk: <https://splunkbase.splunk.com/app/4305/>

Répertoire GIT : <https://github.com/olafhartong/ThreatHunting>



Figure 66 - Logo de l'application "Threat Hunting"

Cette application couvre actuellement 117 techniques. Ça donne le résultat suivant dans le navigateur :

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command And Control
10 items	27 items	42 items	21 items	53 items	15 items	20 items	15 items	13 items	9 items	19 items
Drive-by Compromise	CMSTP	Accessibility Features	Access Token Manipulation	Account Manipulation	Account Discovery	Application Deployment Software	Audio Capture	Automated Exfiltration	Commonly Used Port	
Exploit Public-Facing Application	Command-Line Interface	Account Manipulation	Binary Padding	Brute Force	Application Window Discovery	Distributed Component Object Model	Automated Collection	Data Compressed	Communication Through Removable Media	
Hardware Additions	Compiled HTML File	AppCert DLLs	BITS Jobs	Credential Dumping	Browser Bookmark Discovery	Exploitation of Remote Services	Clipboard Data	Data Encrypted	Custom Command and Control Protocol	
Replication Through Removable Media	Control Panel Items	AppInit DLLs	Bypass User Account Control	Credentials in Registry	File and Directory Access	Data from Information Repositories	Data Transfer Size Limits	Data Staged	Custom Cryptographic Protocol	
Spearphishing Attachment	Dynamic Data Exchange	Application Shimming	Applinit DLLs	CMSTP	Forced Authentication	Data from Local System	Exfiltration Over Alternative Protocol	Exfiltration Over Other Network Medium	Data Encoding	
Spearphishing Link	Execution through API	Authentication Package	Bypass User Account Control	Code Signing	Network Service Scanning	Data from Network Shared Drive	Exfiltration Over Command and Control Channel	Fallback Channels	Data Obfuscation	
Spearphishing via Service	Execution	Load	BITS Jobs	Compiled HTML File	Pass the Hash	Data from Removable Media	Exfiltration Over Physical Medium	Exfiltration Over Network Medium	Domain Fronting	
Supply Chain Compromise	Exploitation for Client Execution	Bootkit	DLL Search Order Hijacking	Component Firmware	Pass the Ticket	Remote Desktop Protocol	Exfiltration Over Other Network Medium	Exfiltration Over Physical Medium	Fallback Channels	
Trusted Relationship	Graphical User Interface	Browser Extensions	Exploitation for Privilege Escalation	Component Object Model Hijacking	Network Share Discovery	Remote File Copy	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Multi-hop Proxy	
Valid Accounts	InstallUtil	Change Default File Association	Control Panel Items	Input Capture	Network Sniffing	Email Collection	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Multi-Stage Channels	
LSASS Driver	Component Firmware	Extra Window Memory Injection	Kerberoasting	Network Sniffing	Peripheral Device Discovery	Input Capture	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Multiband Communication	
PowerShell	Component Object Model Hijacking	DCShadow	LMNR/NBT-NS Poisoning	>Password Filter DLL	Permission Groups Discovery	Man in the Browser	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Multilayer Encryption	
Regsvcs/Regasm	Create Account	File System Permissions Weakness	Deobfuscate/Decode Files or Information	Private Keys	Process Discovery	Screen Capture	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Remote Access Tools	
Regsvr32	DLL Search Order Hijacking	Hooking	DLL Search Order Hijacking	Two-Factor Authentication Interception	Query Registry	Video Capture	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Remote File Copy	
Rundll32	External Remote Services	Image File Execution Options Injection	DLL Side-Loading	Remote System Discovery	Windows Admin Shares	Taint Shared Content	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Standard Application Layer Protocol	
Scheduled Task	File System Permissions Weakness	New Service	Exploitation for Defense Evasion	Security Software Discovery	Windows Remote Management	Third-party Software	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Standard Cryptographic Protocol	
Scripting	Hidden Files and Directories	Path Interception	Extra Window Memory Injection	System Information Discovery		Windows Admin Shares	Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Standard Non-Application Layer Protocol	
Service Execution	Hooking	Port Monitors	File Deletion	System Network Configuration Discovery			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Uncommonly Used Port	
Signed Binary Proxy Execution	Hypervisor	Scheduled Task	File Permissions Modification	System Network Connections Discovery			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium	Web Service	
Signed Script Proxy Execution	Image File Execution Options Injection	Logon Scripts	File System Logical Offsets	Indicator Blocking			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium		
Third-party Software	Logon Scripts	LSASS Driver	Image File Execution Options Injection	Indicator Removal from Tools			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium		
Trusted Developer Utilities	Modify Existing Service	Web Shell	Service Registry Permissions Weakness	Indicator Removal on Host			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium		
User Execution	Ntsh Helper DLL	New Service	SID-History Injection	Indirect Command Execution			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium		
Windows Management Instrumentation	New Service	Office Application Startup	Valid Accounts	Install Root Certificate			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium		
Windows Remote Management	Path Interception	Port Monitors	Web Shell	InstallUtil			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium		
XSL Script Processing				Masquerading			Exfiltration Over Physical Medium	Exfiltration Over Physical Medium		

Figure 67 - Techniques couvertes par l'application "Threat Hunting"

Avant de pouvoir utiliser l'application, on doit installer certaines applications requises, créer l'index « Threat Hunting » et ajuster les macros en fonction des index déjà en place.

La configuration par défaut est fournie avec le type de source :

"sourcetype="XMLWinEventLog:Microsoft-Windows-Sysmon/Operational"

Les applications suivantes sont requises pour la gestion et la création des différents tableaux de bord de supervision :

- Punchcard Visualization: <https://splunkbase.splunk.com/app/3129/>
- Force Directed Visualization: <https://splunkbase.splunk.com/app/3767/>
- Sankey Diagram Visualization: <https://splunkbase.splunk.com/app/3112/>
- Lookup File Editor: <https://splunkbase.splunk.com/app/1724/>

Enfin, il faut créer des fichiers « lookups » pour whitelister les évènements anodins (faux positifs). Ils peuvent être téléchargés avec le lien suivant :

<https://github.com/olahartong/ThreatHunting/raw/master/files/ThreatHunting.tar.gz>

On obtient un décompte de tous les déclencheurs par catégorie ATT&CK ainsi que les techniques les plus déclenchées et les hôtes les plus touchés :

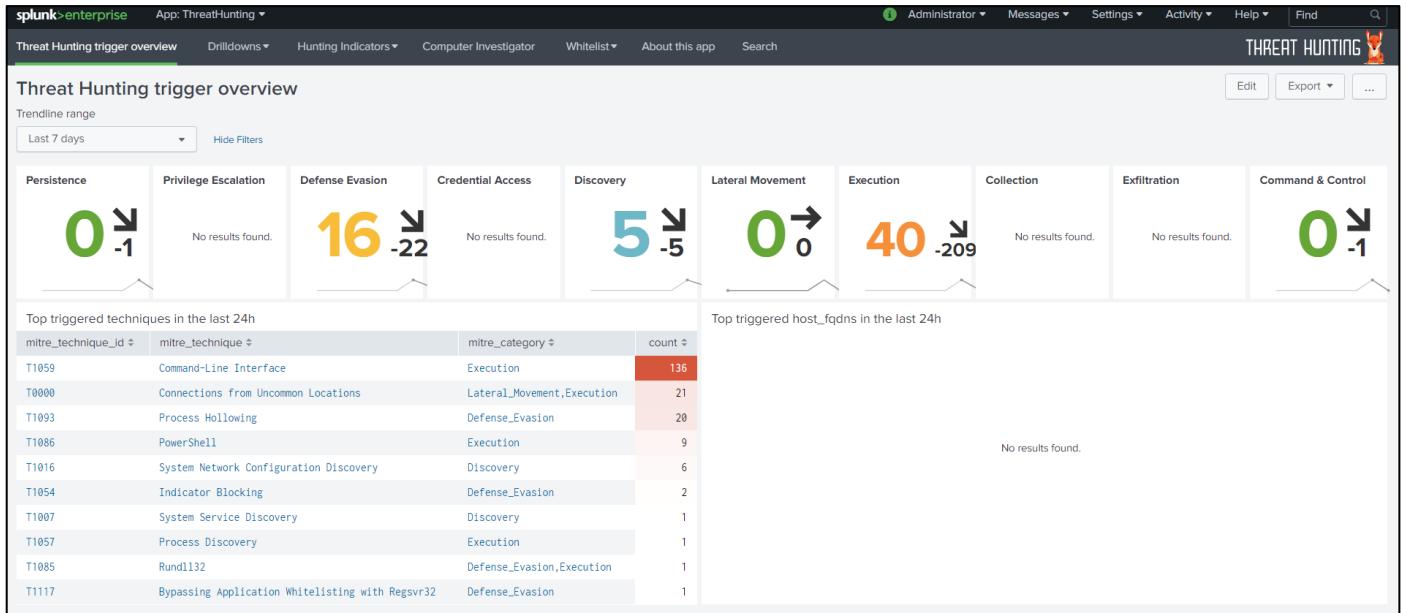


Figure 68 - Aperçu des déclencheurs

On peut aller en détail dans chaque technique pour visualiser les événements Sysmon qui ont déclenché des indicateurs par le biais des rapports.

Dans l'exemple suivant, on a les événements liés à la technique « Credential Access » :

The details view shows three tables of events:

- Process Create:**

_time	ID	Technique	Category	Trigger	host_fqdn	user_name	process_parent_path	process_path	process_parent_command_line	process_command_line	process_parent_guid
2018-12-04 14:40:53	T1003	Credential Dumping	Credential_Access	Reg dump SAM/System db	alice.insecurebank.local	NT AUTHORITY\SYSTEM	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	/C reg save hklm\sam sam	reg save hklm\sam sam	{\$1789B85-91F5-5C06-0000-0019740F0909}
- Process Access:**

_time	ID	Technique	Category	Trigger	host_fqdn	process_path	target_process_path	process_granted_access	target_process_guid	process_id	target_process_id
2018-12-04 15:11:08	T1003	Credential Dumping	Credential_Access	Potentially Mimikatz	dave.insecurebank.local	C:\Windows\system32\rundll32.exe	C:\Windows\system32\lsass.exe	0x1010		2192	484
2018-12-04 15:11:17	T1003	Credential Dumping	Credential_Access	Potentially Mimikatz	edward.insecurebank.local	C:\Windows\system32\rundll32.exe	C:\Windows\system32\lsass.exe	0x1010		1628	492
2018-12-04 15:11:26	T1003	Credential Dumping	Credential_Access	Potentially Mimikatz	fred.insecurebank.local	C:\Windows\system32\rundll32.exe	C:\Windows\system32\lsass.exe	0x1010		1268	484
2018-12-04 15:11:44	T1003	Credential Dumping	Credential_Access	Potentially Mimikatz	bob.insecurebank.local	C:\Windows\system32\rundll32.exe	C:\Windows\system32\lsass.exe	0x1010		3860	468
2018-12-04 15:10:56	T1003	Credential Dumping	Credential_Access	Potentially Mimikatz	charles.insecurebank.local	C:\Windows\system32\rundll32.exe	C:\Windows\system32\lsass.exe	0x1010		1300	480
2018-12-04 15:09:33	T1003	Credential Dumping	Credential_Access	Potentially Mimikatz	DC1.insecurebank.local	C:\Windows\system32\rundll32.exe	C:\Windows\system32\lsass.exe	0x1010		3992	468
- Image Loaded:**

_time	ID	Technique	Category	Trigger	host_fqdn	process_path	driver_loaded	driver_is_signed	driver_signature	driver_signature_status	process_id	process_guid
2018-12-04 15:00:38	T1003	Credential Dumping	Credential_Access	Probably Mimikatz	dev_server.insecurebank.local	C:\Windows\System32\rundll32.exe	C:\Windows\System32\WinCard.dll	true	Microsoft Windows	Valid	3992	{6E0C83B8-968E-5C06-0000-0010A7433308}

Figure 69 - Détails des évènements de la technique « Credential Access »

4.4.1 Chasse aux processus

Les GUID (Globally Unique Identifier) sont très puissants pour l'identification des processus, ce sont des identifiants uniques pour un processus qui permettent de suivre de manière fiable les processus et de rechercher ses parents et ses enfants. L'utilisation des PID (Process ID) est moins fiable car ils sont réutilisés assez rapidement sur un système actif, ce qui peut fausser l'investigation.

En cliquant sur ces GUID, on accédera à un tableau de bord ciblé indiquant le niveau de parent et 1 niveau d'enfants :

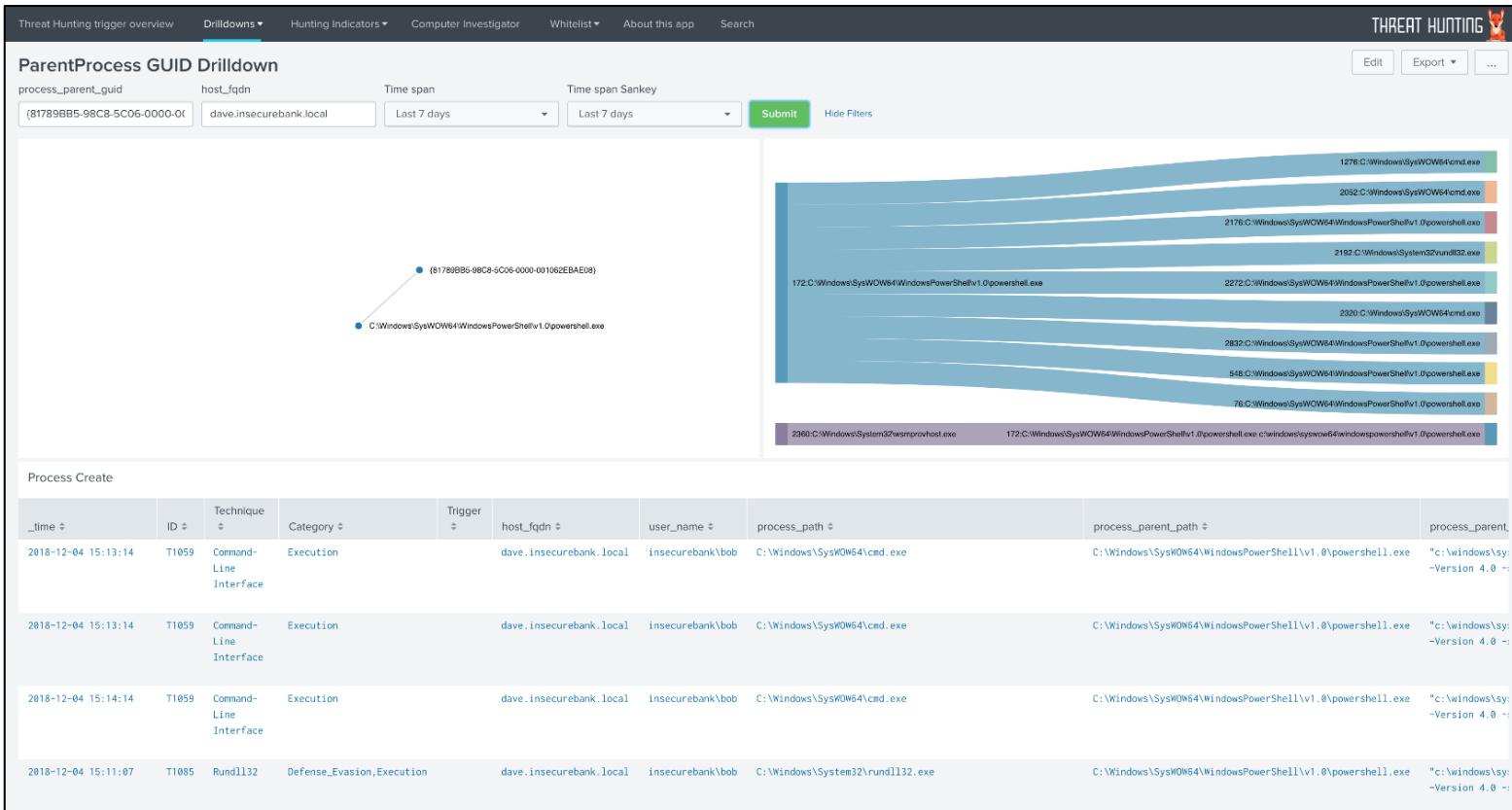


Figure 70 - Détails d'un GUID

Cette page donne un aperçu des événements dans de nombreux cas. Le graphique de gauche montre les processus liés au GUID, il ne devrait y en avoir qu'un seul. **Dans le cas de certaines techniques d'injection de processus, il pourrait y en avoir un deuxième qu'il faudra vérifier.**

Le graphique sur la droite montre un arbre de processus. Il recherche un parent et un enfant du GUID utilisé. Dans ce cas, on voit le lancement de PowerShell. En plus, de nombreux enfants sont nés, qui sont visibles dans la section "**Process Create**" et d'autres types d'événements qui auraient été déclenchés.

Avec ce tableau, on détectera les injections de processus et les processus qui sont potentiellement malicieux qui se cachent derrière des processus légitimes. Comme « svchost » utilisé par les malware pour échapper aux antivirus.

4.4.2 Recherche basée sur les ordinateurs

En basant la recherche sur le nom d'un ordinateur (**host_fqdn**), on sera dirigé vers l'exploration de l'ordinateur (Computer Drilldown). Ce tableau de bord comporte une chronologie des événements déclenchés sur une période choisie :

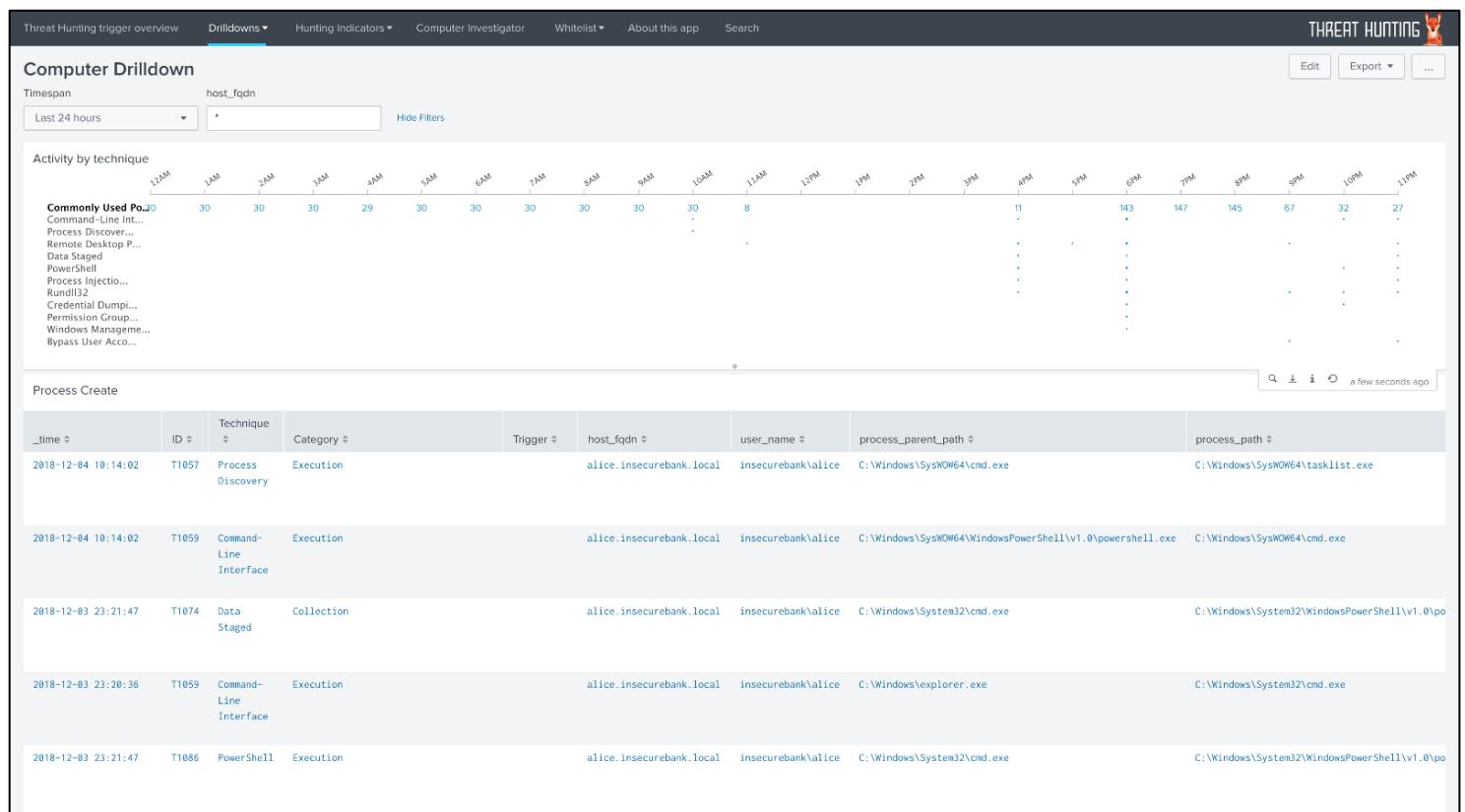


Figure 71 - Exploration des évènements d'un ordinateur

On peut également pousser l'investigation d'un ordinateur en visitant le tableau de bord « **Computer Investigator** ».

Ce tableau fournira une vue d'ensemble de tous les événements qui sont produis sur de la machine recherchée :

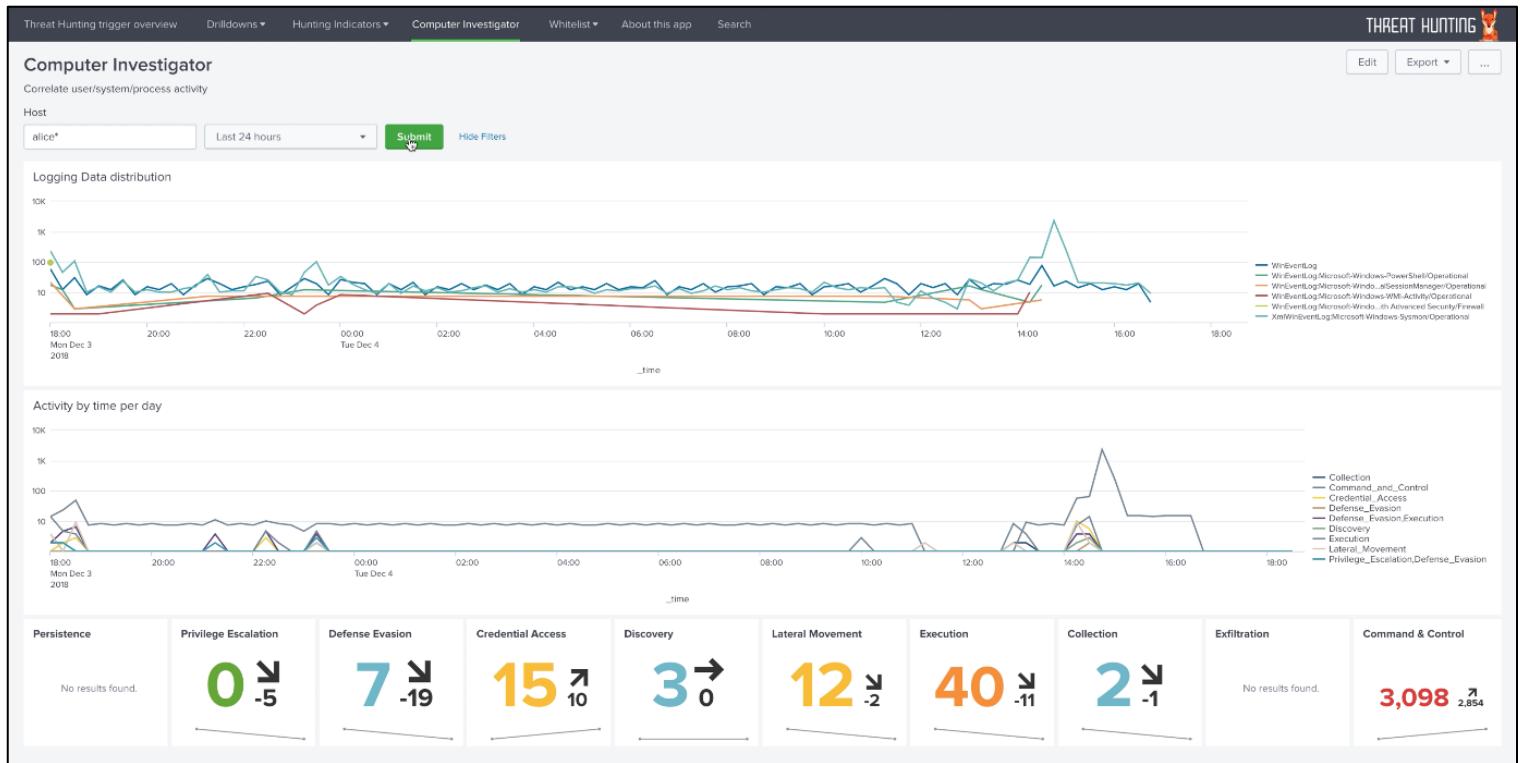


Figure 72 - Evènements enregistrés sur une machine

Cela aidera à repérer les périodes intéressantes ou des choses suspectes qui ont pu se produire sur la machine. On peut également s'en servir pour des fins de forensiques.

4.4.3 Recherche basée sur les connexions réseau

En cliquant sur une adresse IP source ou destination on se redirigé vers un tableau de bord des connexions réseau ou l'on aura les toutes les connexions pertinentes vers ou depuis cette adresse IP :

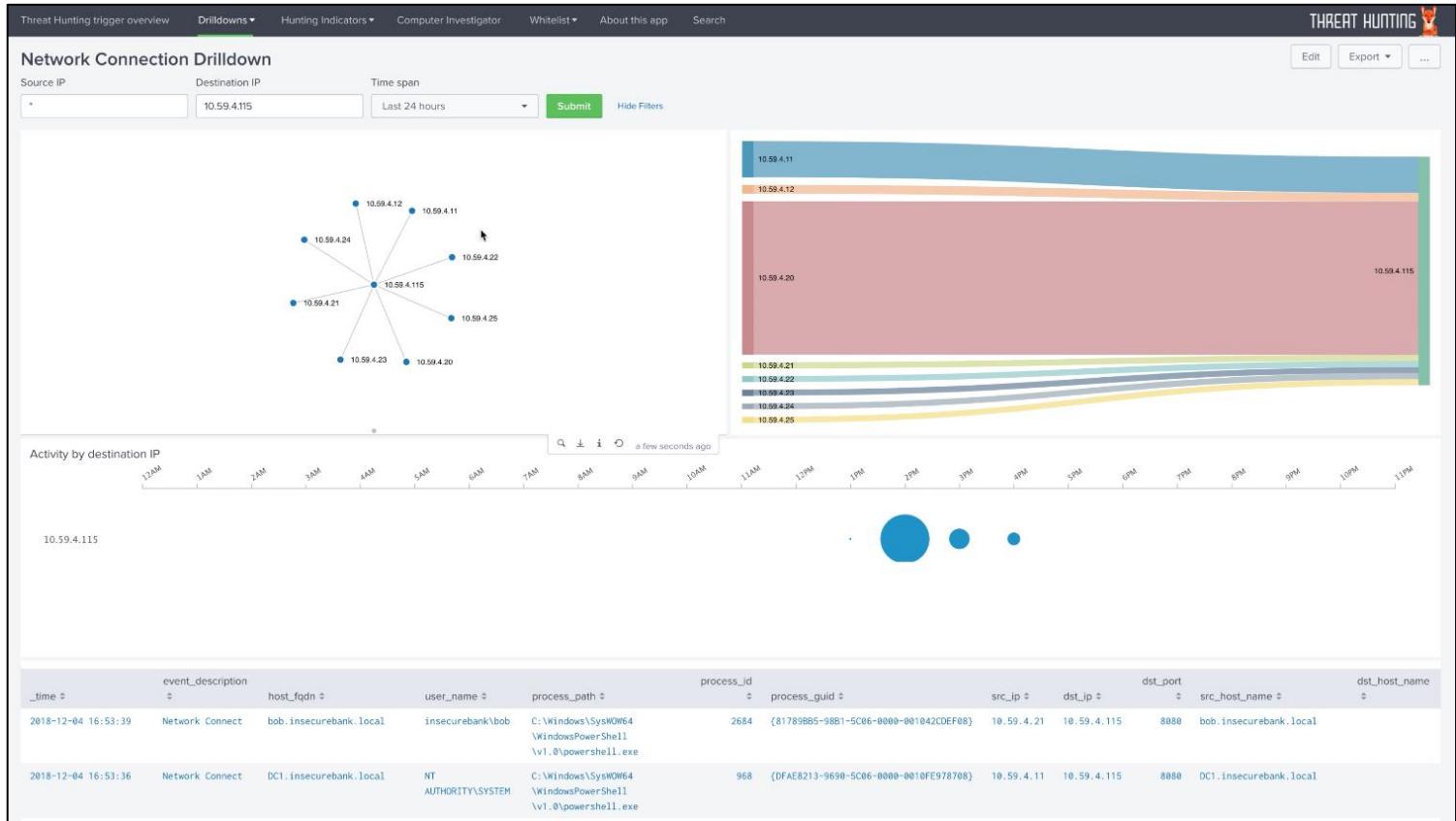


Figure 73 - Tableau de bord des connexions réseau

Le graphique de gauche montre l'entité recherchée au milieu des connexions autour de celle-ci. A droite, un diagramme de Sankey indiquant le poids ou la quantité de connexions (en rendant les barres plus épaisses) pour chaque IP.

Ensuite, un graphique en pointillé montrant les connexions vers ou depuis l'IP sur une période donnée. Le tableau en bas contient tous les déclencheurs pertinents et tous les événements enregistrés.

Ces événements peuvent être défilés pour fournir plus d'informations tirées des différentes sources de « Threat Intelligence ».

4.4.4 Listes blanches

Une des caractéristiques essentielles de cette application est l'option de liste blanche (whitelist). Certains rapports ont des paramètres génériques ce qui engendre beaucoup de faux positifs.

Tout champ **_time** de n'importe quel tableau est cliquable et amènera à un éditeur de liste blanche correspondant au type d'événement respectif. Le formulaire sera rempli avec tous les champs pertinents de la ligne du tableau, ce qui permettra de les modifier pour ajouter des listes blanches.

added_date	contact	mitre_technique_id	reason	host_fqdn	user_name	process_path	src_ip	dst_ip	dst_port
2018-12-04	admin	T1043	This is fine, i trust this host	alice.insecurebank.local	NT AUTHORITY\SYSTEM	C:\Windows\System32\spoolsv.exe	10.59.4.20	10.59.4.115	8080

Figure 74 - Tableau des listes blanches

Cela sera particulièrement utile pour les listes blanches pour plusieurs machines ou lorsqu'une ligne de commande contient une valeur dynamique. La seule chose requise est d'ajouter une raison. Après l'envoi, l'événement sera ajouté à la liste de recherche, avec l'ajout des données et de l'utilisateur qui a ajouté l'entrée de la liste blanche pour référence ultérieure.

Enfin, cette application est encore en voie de développement mais elle fournit déjà les bases d'un EDR dans un environnement Splunk. On peut imaginer qu'elle pourra être déployée uniquement sur des systèmes jugés critiques pour éviter la génération d'une grande quantité de logs qui pourra impacter la facturation de la licence Splunk.

4.5 ATT&CK et les tests s'intrusion

MITRE ATT&CK s'applique également dans les test d'intrusion. En effet, on a vu que l'on peut tester les règles en exécutant des tests atomiques qui reproduisent les techniques des attaquants avec Atomic Read Teaming.

Cet exercice peut être automatisé grâce à **CALDERA (Cyber Adversary Language and Decision Engine for Red Team Automation)**, un outil qui permet de simuler les différentes décisions qui pourraient être prises par un véritable attaquant :

<https://github.com/mitre/caldera>

CALDERA est une application web écrite en Python et qui s'installe localement. Pour lancer les opérations, il faudra installer des agents CALDERA sur les machines ciblées par les tests. On peut retrouver la liste des agents connectés sur l'interface d'administration :

The screenshot shows the CALDERA administration interface. At the top, there are navigation tabs: Threat, Networks, Operations, and Debug. On the right, there are links for Script Editor, Settings, and admin (Admin). The main area is titled "Connected Agents" and displays a table with 5 rows. The columns are IP and Hostname. The data is as follows:

IP	Hostname
192.168.56.200	win7x01.mountainpeak.local
192.168.56.201	win7x02.mountainpeak.local
192.168.56.202	win7x03.mountainpeak.local
192.168.56.203	win7x04.mountainpeak.local
192.168.56.254	win2012xdc.mountainpeak.local

Figure 75 - Gestion des agents CALDERA

Les test d'intrusion sont appelés ici « campagnes ». Afin de lancer une campagne, on crée un réseau de machines. On ajoutera toutes les machines ciblées dans l'onglet « Networks » :

The screenshot shows the CALDERA administration interface with the "Networks" tab selected. On the left, there is a diagram of five red circles representing hosts, labeled win7x01, win7x02, win7x03, win7x04, and win2012xdc. To the right of the diagram is a table titled "Network mountainpeak.local" showing host status. Below the table is a section for adding new hosts.

hostname	Status
win7x01	active
win7x04	active
win7x02	active
win7x03	active
win2012xdc	active

Add a New Host

Figure 76 - Création d'un réseau d'agents CALDERA

Avant de lancer une opération, on doit ajouter un profil d'attaquant. Le profile est ajouté dans l'onglet « Adversary » et il porte un nom et surtout une liste de techniques à réaliser appelées « Steps » :

Figure 77 - Crédit d'un profil d'attaquant

Ensuite, on crée une opération en spécifiant le profil de l'attaquant à partir des profiles créés, le réseau ciblé, la première machine à infecter et le processus parent.

Maintenant, on lance la campagne. CALDERA va exécuter les techniques disponibles dans le profile et va essayer d'infecter tout le réseau victime en commençant par la première machine désignée dans la configuration :

Figure 78 - Lancement de la campagne

On peut voir le détail de chaque pas exécuté avec les noms des techniques ATT&CK réalisées et les commandes lancées sur la machine :

The screenshot shows a detailed view of a specific step in an ATT&CK operation. At the top, a green header bar indicates the step number (1) and the action: "Copying an implant from win7x03.mountainpeak.local to win7x04.mountainpeak.local". Below this, the "Step" section shows the name "copy_file". Under "ATT&CK Tags", two tags are listed: "Remote File Copy (Lateral Movement)" and "Execution through API (Execution)". The "Commands" section contains the following details:
Hostname: win7x03
Command Line: cmd /c copy "C:\commander.exe" "\\\win7x04.mountainpeak.local\C\$\commander.exe"
StdOut:
1 file(s) copied.

Figure 79 - Détails d'une opération

Se lancer dans l'émulation APT n'est pas une tâche facile, mais CALDERA rend tout cela beaucoup plus intuitif. C'est un excellent outil pour tester les capacités des plates-formes de défense par rapport à la matrice MITRE ATT&CK et voir ce que l'on peut, ou ne peut pas détecter, ainsi que pour mieux comprendre comment certaines de ces techniques sont réellement exécutées ; ce qui permettra d'améliorer massivement l'analyse et la posture de défense.

4.6 Conclusion

Pour conclure ce chapitre, le diagramme de cas d'utilisation synthétise tout ce qui a été discuté et présente les acteurs qui peuvent utiliser MITRE ATT&CK et les besoins auxquels le modèle répond :

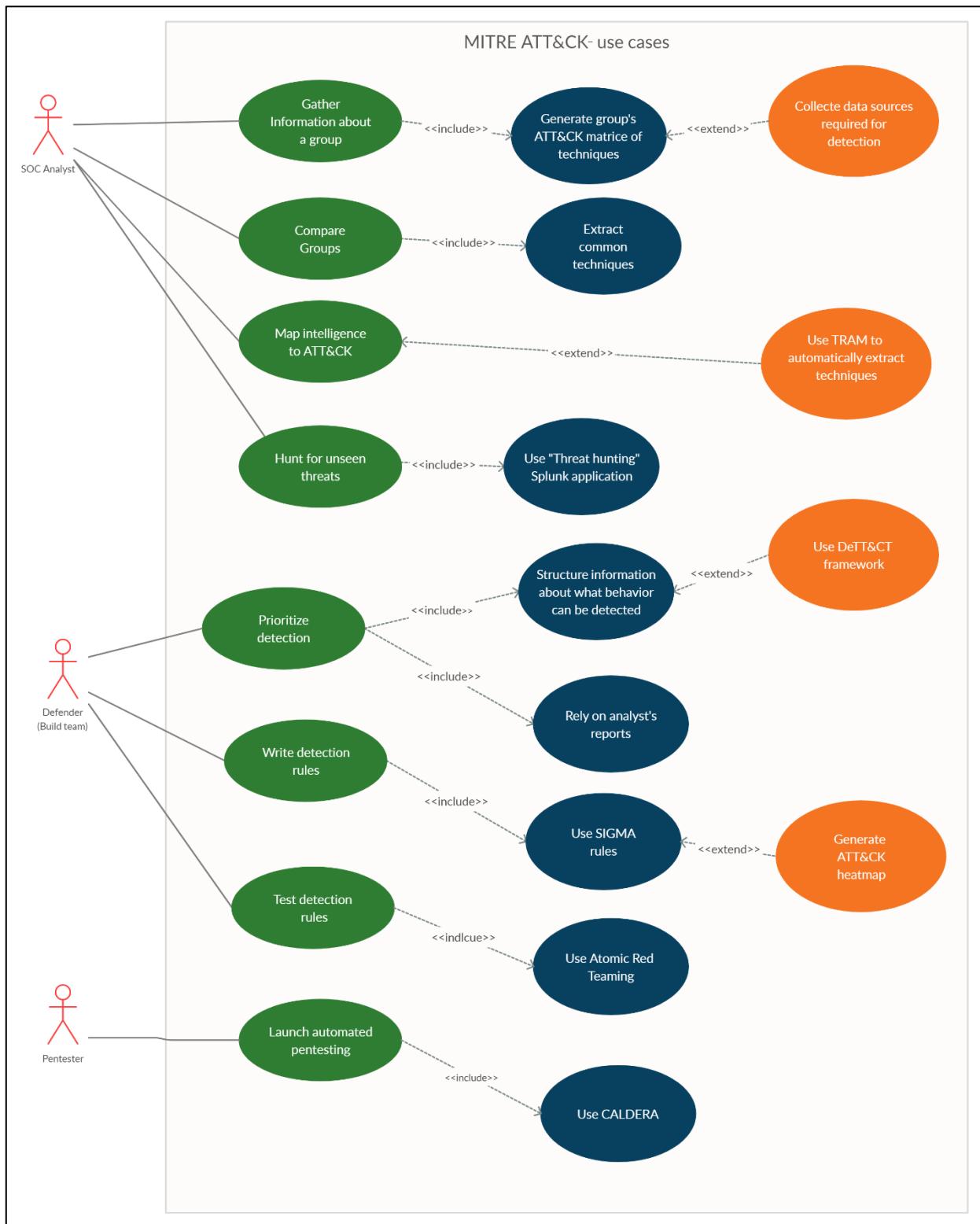


Figure 80 - Diagramme des cas d'utilisation de MITRE ATT&CK

5 Résultats et conclusion générale

Cette étude a montré que MITRE ATT&CK peut devenir un langage de communication entre les membres des différentes équipes d'un SOC et il n'est pas fait uniquement pour les chercheurs en CTI.

a. Threat Intelligence & Réponse à incident :

Les analystes peuvent s'appuyer sur les différentes matrices MITRE ATT&CK afin de récolter des informations sur une technique donnée ou sur un groupe connu pour être menaçant envers les entreprises d'énergie. Les informations qui se trouvent dans la partie « Références » de chaque groupe permettent d'obtenir rapidement une liste d'IOC ou TTP qui constituent un point de pivot dans la recherche d'une potentielle compromission non vue par le SIEM.

Afin d'optimiser la collecte des TTP, les analystes peuvent utiliser TRAM l'outil qui permet d'automatiser l'extraction des techniques à partir d'un rapport. Le script que j'ai créé intervient également pour avoir des informations sur les techniques associées aux sources de données requises pour la détection. On a vu également que les données enregistrées dans les matrices peuvent être chargées dans Neo4J et requêtées afin de comparer entre les groupes ou d'étudier un groupe particulier.

MITRE ATT&CK a des implémentations dans différents outils de gestion d'incidents. Par exemple, IBM a mis en place un module qui s'intègre dans son outil de gestion d'incidents SIRP.

The screenshot shows a ticket in the IBM Resilient platform for a 'Template Injection' technique. The ticket details include:

- Description:** No description.
- Owner:** Unassigned
- Creator:** testerfirst testeralast
- Date Initiated:** 03/13/2019 15:42
- Instructions:** Microsoft's Open Office XML (OOXML) specification defines an XML-based format for Office documents (.docx, .xlsx, .pptx) to replace older binary formats (.doc, .xls, .ppt). OOXML files are packed together ZIP archives comprised of various XML files, referred to as parts, containing properties that collectively define how a document is rendered. (Citation: Microsoft Open XML July 2017) Properties within parts may reference shared public resources accessed via online URLs. For example, template properties reference a file, serving as a pre-formatted document blueprint, that is fetched when the document is loaded. Adversaries may abuse this technology to initially conceal malicious code to be executed via documents (i.e., Scripting) (<https://attack.mitre.org/techniques/T1064>). Template references injected into a document may enable malicious payloads to be fetched and executed when the document is loaded. These documents can be delivered via other techniques such as [Spearphishing Attachment] (<https://attack.mitre.org/techniques/T1193>) and/or [Taint Shared Content] (<https://attack.mitre.org/techniques/T1080>) and may evade static detections since no typical indicators (VBA macro, script, etc.) are present until after the malicious payload is fetched. (Citation: RedoxBlue Remote Template Injection) Examples have been seen in the wild where template injection was used to load malicious code containing an exploit. (Citation: MalwareBytes Template Injection OCT 2017) This technique may also enable [Forced Authentication] (<https://attack.mitre.org/techniques/T1187>) by injecting a SMB/HTTPS (or other credential prompting) URL and triggering an authentication attempt. (Citation: Anomali Template Injection MAR 2018) (Citation: Talos Template Injection July 2017) (Citation: ryanshon phishery SEPT 2016)
- Description:** Microsoft's Open Office XML (OOXML) specification defines an XML-based format for Office documents (.docx, .xlsx, .pptx) to replace older binary formats (.doc, .xls, .ppt). OOXML files are packed together ZIP archives comprised of various XML files, referred to as parts, containing properties that collectively define how a document is rendered. (Citation: Microsoft Open XML July 2017) Properties within parts may reference shared public resources accessed via online URLs. For example, template properties reference a file, serving as a pre-formatted document blueprint, that is fetched when the document is loaded. Adversaries may abuse this technology to initially conceal malicious code to be executed via documents (i.e., Scripting) (<https://attack.mitre.org/techniques/T1064>). Template references injected into a document may enable malicious payloads to be fetched and executed when the document is loaded. These documents can be delivered via other techniques such as [Spearphishing Attachment] (<https://attack.mitre.org/techniques/T1193>) and/or [Taint Shared Content] (<https://attack.mitre.org/techniques/T1080>) and may evade static detections since no typical indicators (VBA macro, script, etc.) are present until after the malicious payload is fetched. (Citation: RedoxBlue Remote Template Injection) Examples have been seen in the wild where template injection was used to load malicious code containing an exploit. (Citation: MalwareBytes Template Injection OCT 2017) This technique may also enable [Forced Authentication] (<https://attack.mitre.org/techniques/T1187>) by injecting a SMB/HTTPS (or other credential prompting) URL and triggering an authentication attempt. (Citation: Anomali Template Injection MAR 2018) (Citation: Talos Template Injection July 2017) (Citation: ryanshon phishery SEPT 2016)
- Detection:** Analyze process behavior to determine if an Office application is performing actions, such as opening network connections, reading files, spawning abnormal child processes (ex: [PowerShell]) (<https://attack.mitre.org/techniques/T1086>), or other suspicious actions that could relate to post-compromise behavior.
- Mitigation:** Consider disabling Microsoft Office macros/active content to prevent the execution of malicious payloads in documents (Citation: Microsoft Disable Macros), though this setting may not mitigate the [Forced Authentication] (<https://attack.mitre.org/techniques/T1187>) use for this technique. Because this technique involves user interaction on the endpoint, it's difficult to fully mitigate. However, there are potential mitigations including training users to identify social engineering techniques and spearphishing emails. Network/Host intrusion prevention systems, antivirus, and detonation chambers can be employed to prevent documents from fetching and/or executing malicious payloads. (Citation: Anomali Template Injection MAR 2018)

Figure 81 - Implémentation de MITRE ATT&CK dans SIRP

Cette application va aider notamment dans l'enrichissement des tickets d'incidents et spécialement la partie **mitigation** où l'on peut très vite trouver les informations et les conseils de remédiation des techniques et malware.

b. Détection :

L'équipe détection (Build) utilisera les renseignements sur les menaces pour évaluer le niveau de maturité de la détection. L'évaluation peut se faire avec l'outil **DeTT&CT** qui permet d'hiérarchiser les menaces en se basant sur les résultats du Framework.

On a vu que la supervision des processus et fichiers couvre un grand nombre de techniques. Cependant, cela génère une quantité importante de logs qui peuvent impacter la licence du SIEM en place. Une des solutions, est de superviser les processus identifiés pertinents ou de réaliser la surveillance uniquement sur des systèmes jugés critiques.

Du côté de l'écriture des règles, l'équipe en charge de la détection peut se baser sur les analyses CAR qui offre un panel de règles avec des pseudocodes qui y correspondent. Ces pseudocodes peuvent être réécrits avec SIGMA qui convertira à son tour les règles en langage SPL ou tout autre langage de traitement des recherches.



Figure 82 - Création d'une règle de détection

c. Threat Hunting :

Pour repérer les faux négatifs, les analystes peuvent utiliser l'application Splunk "**Threat Hunting app**" qui requiert la disponibilité des logs Windows Sysmon. Pour permettre une utilisation optimale sans excéder la licence du SIEM en place, le BUILD de son côté doit trouver le fichier de configuration le plus efficace et qui remonte uniquement les événements les plus pertinents.

Deux solutions sont disponibles :

- **Sysmon-modular** : <https://github.com/olafhartong/sysmon-modular>
- **Sysmon-config** : <https://github.com/SwiftOnSecurity/sysmon-config>

d. Red Teaming :

Les règles mises en place sont à tester avec **Atomic Red Teaming** qui fournit de multiples tests atomiques pour chaque technique ATT&CK. Les tests peuvent être automatisés avec **CALEDRA**.

6 Références

6.1 Articles en ligne :

- Katie Nickels (2018), *Using ATT&CK to advance cyber threat intelligence* :
<https://medium.com/mitre-attack/using-att-ck-to-advance-cyber-threat-intelligence-part-1-c5ad14d59724>
- Olaf Hartong (2018), *Endpoint detection superpowers on the cheap, Threat Hunting app* :
<https://medium.com/@olafhartong/endpoint-detection-superpowers-on-the-cheap-threat-hunting-app-a92213f5e4b8>
- Barbara Louis-Sidney (2018), *MITRE ATT&CK : et si on s'attaquait au haut de la pyramid of pain ?* :
<https://medium.com/cyberthreatintel/mitre-att-ck-et-si-on-sattaquait-au-haut-de-la-pyramid-of-pain-1-2-9d05f707c9f0>
- Barbara Louis-Sidney (2020), *MITRE ATT&CK et ses sous-techniques, analyse* :
<https://medium.com/cyberthreatintel/mitre-att-ck-et-ses-sous-techniques-analyse-ba3ef4b1e1d9>
- David J. Bianco (2014), *Pyramid of pain* :
<http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html>
- Roberto Rodriguez (2018), *Categorizing and Enriching Security Events in an ELK with the Help of Sysmon and ATT&CK* :
<https://cyberwardog.blogspot.com/>
- Blake Strom (2020), *ATT&CK with sub-techniques* :
<https://medium.com/mitre-attack/attack-sub-what-you-need-to-know-99bce414ae0b>
- Blake Strom (2019), *Getting Started with ATT&CK: Adversary Emulation and Red Teaming*:
<https://medium.com/mitre-attack/getting-started-with-attack-red-29f074ccf7e3>
- Brianne Fahey (2020), *Defending with Graphs: Create a Graph Data Map to Visualize Pivot Paths* :
<https://www.sans.org/reading-room/whitepapers/logging/paper/39030>
- Ruben Bouman (2019), *Mapping your Blue Team to MITRE ATT&CK™* :
<https://www.siriussecurity.nl/blog/2019/5/8/mapping-your-blue-team-to-mitre-attack>

6.2 Livres

- Blake E. Strom, Andy Applebaum, Doug P. Miller, Kathryn C. Nickels, Adam G. Pennington, Cody B. Thomas (2018), *MITRE ATT&CK : Design and Philosophy*.
- Don Murdoch (2018), *Blue Team Handbook : SOC, SIEM and Threat Hunting use cases*.

7 Glossaire

ATT&CK	Adversarial Tactics, Techniques and Common Knowledge
C&C	Command and Control
CALDERA	Cyber Adversary Language and Decision Engine for Red Team Automation
CAR	Cyber Analytics Repository
CSV	Comma-separated values
CTI	Cyber Threat Intelligence
DeTT&CT	DEtect Tactics, Techniques & Combat Threats
EDR	Endpoint Detection and Response
EQL	Event Query language
GUID	Globally Unique Identifier
ICS	Industrial Control System
IoA	Indicator of Attack
IoC	Indicator of Compromise
JSON	JavaScript Object Notation
PID	Process ID
RPC	Remote Procedure Call
SIEM	Security Information Event Management
SOC	Security Operations Center
SPL	Search Processing Language
STIX	Structured Threat Information Expression
Sysmon	System Monitor
TAXII	Trusted Automated Exchange of Intelligence Information
TRAM	Threat Report ATT&CK Mapper
TTP	Tactics, Techniques and Procedures
WMI	Windows Management Instrumentation
YAML	Markup Language

8 Annexes

8.1 Langage EQL

EQL est un langage qui permet de : correspondre des événements, générer des séquences, empiler des données, construire des agrégats et effectuer des analyses.

Il est sans schéma et peut prendre en charge plusieurs base de données. Les requêtes sont lisible et expriment facilement ce que l'on souhaite retrouver.

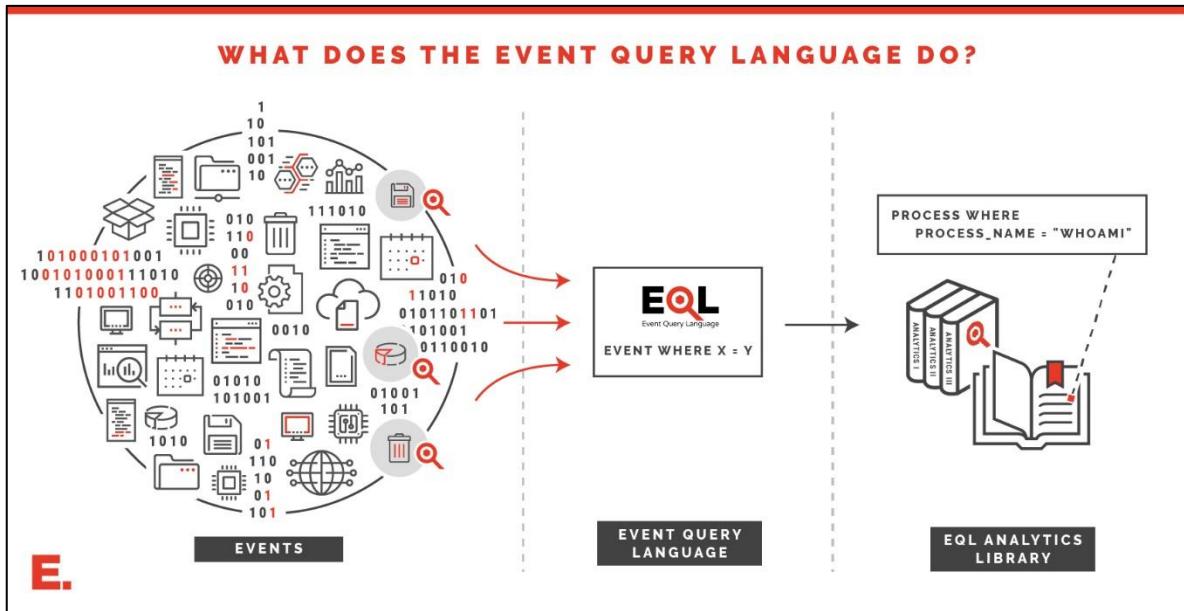


Figure 83 - Langage EQL source: <https://eql.readthedocs.io/en/latest/>

Par exemple : On veut savoir quelles sont les cinq principales connexions réseau sortantes qui ont transmis plus de 100 Mo :

```
network where total_out_bytes > 100000000
| sort total_out_bytes
| tail 5
```

8.2 Code source du script Python : Extraction des techniques et sources d'information qui concernent un groupe d'attaquants

```
from stix2 import Filter,TAXIICollectionSource
from taxii2client import Collection
import argparse
import csv

#Se connecter au serveur TAXII de mitre et récupérer la matrice Enterprise
collection = Collection("https://cti-taxii.mitre.org/stix/collections/95ecc380-afe9-11e4-9b6c-751b66dd541e/")
fs = TAXIICollectionSource(collection)

techniques_data_sources = {}

#Retrouver le groupe avec son alias
def get_group_by_alias(src, alias):
    return src.query([
        Filter('type', '=', 'intrusion-set'),
        Filter('aliases', '=', alias)
    ])

#Extraire toutes les techniques du groupe
def get_technique_by_group(src, stix_id):
    relations = src.relationships(stix_id, 'uses', source_only=True)
    return src.query([
        Filter('type', '=', 'attack-pattern'),
        Filter('id', 'in', [r.target_ref for r in relations])
    ])

#Extraire tous les groupes
def get_all_groups(src):
    groups = []
    q = src.query([Filter('type', '=', 'intrusion-set')])
    for item in q:
        groups.append(item.name)
    return groups

#Récupération du choix de l'utilisateur et test si le groupe entré existe
parser = argparse.ArgumentParser(description='From Group to techniques & data source')
parser.add_argument('--group','-g', type=str, help='ATT&CK Group',required=True)
args = parser.parse_args()
group = args.group

if group not in get_all_groups(fs):
    print("Invalid Group")
    exit()

group = get_group_by_alias(fs, group)[0]
techniques = get_technique_by_group(fs, group)

#Pour chaque technique, ajouter les sources d'information qui correspondent
for item in techniques:
    techniques_data_sources.update({item.name:item.x_mitre_data_sources})

#Enregistrer les données dans un fichier CSV
with open("techniques_datasource.csv", "w", newline="") as fd:
    wr = csv.writer(fd)
    wr.writerow(['technique','data source'])
    for k,v in techniques_data_sources.items():
        for x in v:
            wr.writerow((k,x))
```

8.3 Code source du script Python : Etude statistiques sur les sources d'information et techniques

```
#Ce script s'inspire des notes de la librairie : ATTACK PYTHON CLIENT:  
#https://github.com/hunters-forge/ATTACK-Python-Client/blob/master/notebooks/ATT%26CK_DataSources.ipyn  
  
from attackcti import attack_client  
from pandas import *  
from pandas.io.json import json_normalize  
import altair as alt  
import argparse  
  
def subs(l):  
    res = []  
    for i in range(1, len(l) + 1):  
        for combo in itertools.combinations(l, i):  
            res.append(list(combo))  
    return res  
  
alt.renderers.enable('altair_viewer')  
lift = attack_client()  
  
all_techniques = lift.get_techniques(stix_format=False)  
techniques_normalized = json_normalize(all_techniques)  
  
techniques =  
techniques_normalized.reindex(['matrix','platform','tactic','technique','technique_id','data_sources'], axis=1)  
techniques_with_data_sources=techniques[techniques.data_sources.notnull()].reset_index(drop=True)  
  
techniques_data_source=techniques_with_data_sources  
attributes_3 = ['data_sources']  
  
for a in attributes_3:  
    s = techniques_data_source.apply(lambda x: pandas.Series(x[a]),axis=1).stack().reset_index(level=1,  
drop=True)  
    s.name = a  
    techniques_data_source = techniques_data_source.drop(a, axis=1).join(s).reset_index(drop=True)  
  
techniques_data_source_2 =  
techniques_data_source.reindex(['matrix','platform','tactic','technique','technique_id','data_sources'],  
axis=1)  
techniques_data_source_3 = techniques_data_source_2.replace(['Process monitoring','Application logs'],['Process  
Monitoring','Application Logs'])  
  
parser = argparse.ArgumentParser("Relations between ATT&CK techniques and data sources")  
parser.add_argument("--visualization","-v",type=str,help="Specify a visualisation option : grouping, count or  
subset",required=True)  
args = parser.parse_args()  
option = args.visualization  
  
if option == "grouping":  
    data_source_distribution = pandas.DataFrame({  
        'Data Source': list(techniques_data_source_3.groupby(['data_sources'])['data_sources'].count().keys()),  
        'Count of Techniques':  
    })  
    techniques_data_source_3.groupby(['data_sources']).count().tolist())  
    bars = alt.Chart(data_source_distribution, width=800, height=300).mark_bar().encode(x='Data Source', y='Count  
of Techniques', color='Data Source').properties(width=1200)  
    bars.show()  
  
elif option == "count":  
    data_source_distribution_2 = pandas.DataFrame({  
        'Techniques': list(techniques_data_source_3.groupby(['technique'])['technique'].count().keys()),  
        'Count of Data Sources':  
    })  
    techniques_data_source_3.groupby(['technique']).count().tolist())  
    data_source_distribution_3 = pandas.DataFrame({  
        'Number of Data Sources': list(data_source_distribution_2.groupby(['Count of Data Sources'])['Count of  
Data Sources'].count().keys()),  
        'Count of Techniques': data_source_distribution_2.groupby(['Count of Data Sources'])['Count of Data  
Sources'].count().tolist())  
    })  
    bars = alt.Chart(data_source_distribution_3).mark_bar().encode(x='Number of Data Sources', y='Count of  
Techniques').properties(width=500)  
    bars.show()
```

```

elif option == "subset":
    df = techniques_with_data_sources[['data_sources']]
    for index, row in df.iterrows():
        row["data_sources"]=[x.lower() for x in row["data_sources"]]
        row["data_sources"].sort()
    df['subsets']=df['data_sources'].apply(subs)
    techniques_with_data_sources_preview = df

    attributes_4 = ['subsets']
    for a in attributes_4:
        s = techniques_with_data_sources_preview.apply(lambda x:
pandas.Series(x[a]),axis=1).stack().reset_index(level=1, drop=True)
        s.name = a
        techniques_with_data_sources_preview = techniques_with_data_sources_preview.drop(a,
axis=1).join(s).reset_index(drop=True)

    techniques_with_data_sources_subsets =
techniques_with_data_sources_preview.reindex(['data_sources','subsets'], axis=1)

techniques_with_data_sources_subsets['subsets_name']=techniques_with_data_sources_subsets['subsets'].apply(lambda x: ','.join(map(str, x)))

techniques_with_data_sources_subsets['subsets_number_elements']=techniques_with_data_sources_subsets['subsets'].str.len()

techniques_with_data_sources_subsets['number_data_sources_per_technique']=techniques_with_data_sources_subsets[
'data_sources'].str.len()

subsets = techniques_with_data_sources_subsets
subsets_ok=subsets[subsets.subsets_number_elements != 1]
subsets_graph =
subsets_ok.groupby(['subsets_name'])['subsets_name'].count().to_frame(name='subsets_count').sort_values(by='subsets_count',ascending=False)[0:15]
subsets_graph_2 = pandas.DataFrame({
    'Data Sources': list(subsets_graph.index),
    'Count of Techniques': subsets_graph['subsets_count'].tolist()})
bars = alt.Chart(subsets_graph_2).mark_bar().encode(x ='Data Sources', y ='Count of Techniques',
color='Data Sources').properties(width=500)
bars.show()

else:
    print("Option does not exist")
    exit()

```