

Université Internationale de Rabat

COLLEGE OF ENGINEERING & ARCHITECTURE

École Supérieure d'Informatique et du Numérique (ESIN)

1^{ère} Année Cycle d'Ingénieur

Rapport de Stage Technique

Conception et mise en œuvre d'un laboratoire SOC virtualisé pour la détection et la supervision des cybermenaces

Stage réalisé au sein de CIH Bank



Réalisé par	: Achraf ACHOUKHANE
Filière	: Cybersecurity
Encadrants professionnels	: M. EL MOSTAFA ED-DAIYF M. Oussama AZZAM
Encadrante pédagogique	: Mme Hajar EL GADI
Période du stage	: du 22 juin 2026 au 15 août 2026

Année académique : 2025–2026

Remerciements

Je remercie **CIH Bank** pour son accueil et pour les conditions d'apprentissage offertes durant ce stage.

J'adresse mes sincères remerciements à mes encadrants professionnels, **M. EL MOSTAFA ED-DAIYF** et **M. Oussama AZZAM**, pour leurs orientations, leur disponibilité et leurs retours tout au long du projet.

Je remercie également mon encadrante pédagogique, **Mme Hajar EL GADI**, ainsi que l'**Université Internationale de Rabat** et les enseignants de la filière **Cybersecurity**, pour leur accompagnement et la formation dispensée.

Résumé

Ce rapport présente la réalisation d'un laboratoire SOC virtualisé dans le cadre d'un stage au sein de CIH Bank. L'environnement relie un pare-feu OPNsense, un serveur Wazuh, des machines Linux et Windows, ainsi qu'une machine Kali Linux réservée aux tests contrôlés.

Le laboratoire centralise les journaux, supervise les hôtes et détecte plusieurs événements : tentatives SSH, échecs d'authentification Web, scan réseau, création de processus Windows, modification de fichier et erreur IIS. Les résultats sont illustrés par les captures obtenues pendant la validation.

Mots-clés : SOC, SIEM, Wazuh, OPNsense, Suricata, Sysmon, supervision, cybersécurité.

Abstract

This report presents a virtualized SOC laboratory developed during an internship at CIH Bank. The environment connects an OPNsense firewall, a Wazuh server, Linux and Windows machines, and a Kali Linux machine used for controlled tests.

The laboratory centralizes logs, monitors endpoints, and detects several events: SSH attempts, Web authentication failures, network scanning, Windows process creation, file modification, and an IIS error. The main results are supported by screenshots collected during validation.

Keywords: SOC, SIEM, Wazuh, OPNsense, Suricata, Sysmon, monitoring, cybersecurity.

Liste des acronymes

Acronyme	Signification
ATT&CK	Adversarial Tactics, Techniques, and Common Knowledge
EVE	Format JSON utilisé par Suricata
FIM	File Integrity Monitoring
HIDS	Host-based Intrusion Detection System
IIS	Internet Information Services
NIDS	Network Intrusion Detection System
SIEM	Security Information and Event Management
SOC	Security Operations Center
SOAR	Security Orchestration, Automation and Response
SSH	Secure Shell
VM	Virtual Machine

Table des matières

Remerciements	i
Résumé	ii
Abstract	iii
Liste des acronymes	iv
1 Présentation de CIH Bank et du stage	1
1.1 Présentation de CIH Bank	1
1.2 Organisation de l'entreprise	1
1.3 Contexte du stage	2
1.4 Enjeux de sécurité	2
1.5 Objectifs du projet	3
1.6 Déroulement du stage	3
1.7 Livrables réalisés	3
1.8 Organisation du rapport	4
2 Environnement de travail et outils utilisés	5
2.1 Notions utiles	5
2.2 Fonctionnement simplifié d'une supervision	5
2.2.1 NIDS et HIDS	6
2.3 Machines virtuelles	6
2.4 Outils de sécurité et de supervision	6
2.5 Inventaire des machines	7
2.6 Réseaux virtuels et interfaces	7
3 Architecture du laboratoire	9
3.1 Topologie générale	9
3.2 Règles de circulation	9
3.3 Circulation des événements	10
3.4 Sources de journaux	11
3.5 Chaînes de détection	11
3.6 Critères de validation	11
3.7 Périmètre retenu	11
4 Mise en place du laboratoire	12
4.1 Ordre de déploiement	12
4.2 Configuration d'OPNsense	12
4.3 Installation de Wazuh	13
4.3.1 Communication entre les agents et le serveur	13
4.4 Serveur Linux et application Web	14
4.5 Serveur Windows	14

4.6	Intégration de Suricata	14
4.7	Vérifications de base	15
5	Tests et résultats	16
5.1	Principe de validation	16
5.2	Règles applicatives Wazuh	16
5.2.1	Exemple de corrélation : règle 100502	17
5.3	Autres règles Wazuh validées	18
5.4	Échecs d'authentification Web	18
5.4.1	Lecture de la règle corrélée	19
5.5	Blocages SSH par OPNsense	19
5.6	Création de processus Windows	20
5.7	Détection d'un scan SSH par Suricata	20
5.8	Surveillance de l'intégrité	21
5.9	Supervision IIS	21
5.10	Synthèse	23
6	Bilan et perspectives	24
6.1	Bilan du laboratoire	24
6.2	Interprétation technique	24
6.3	Limites	25
6.4	Perspectives	25
6.5	Recommandations	25
	Conclusion générale	26
A	Vérifications du serveur Linux	27
A.1	Identité et adressage	27
A.2	Services Apache et SSH	27
A.3	Agent Wazuh et réponse HTTP	27
B	Preuves complémentaires	28
B.1	Étapes du scénario applicatif	28
B.2	Erreur IIS côté navigateur	28
C	Lecture des événements Wazuh	29
C.1	Détail d'un événement applicatif	29
C.2	Règle de blocage du pare-feu	30
D	Commandes de validation	31
D.1	Contrôles utilisés	31
	Bibliographie	32

Table des figures

1.1	Logo de CIH Bank	1
1.2	Organigramme général de CIH Bank fourni dans les ressources du stage	2
2.1	Principales machines et plateformes du laboratoire	6
2.2	Outils utilisés pour la détection et l'analyse	6
3.1	Topologie du laboratoire SOC virtualisé	9
3.2	Workflow de détection ; le bloc Active Response/SOAR représente une évolution future non implémentée	10
4.1	Interfaces réseau configurées sur OPNsense	12
4.2	Règles de filtrage appliquées à la zone ATTACK	13
4.3	Agents Linux et Windows actifs dans Wazuh	13
4.4	Application Web utilisée pour le scénario d'authentification	14
4.5	Contrôles réalisés avant les scénarios	15
5.1	Règles applicatives de détection et de corrélation des échecs	17
5.2	Règles applicatives de connexion réussie et de limitation	17
5.3	Alertes applicatives visibles dans Wazuh	18
5.4	Champs de la règle applicative corrélée	19
5.5	Blocages SSH OPNsense corrélés dans Wazuh	19
5.6	Chaîne de supervision Sysmon vers Wazuh	20
5.7	Alerte Suricata reçue dans Wazuh	20
5.8	Modification de fichier détectée par Wazuh FIM	21
5.9	Page Web IIS du serveur Windows	22
5.10	Requête IIS retournant un code 404 visible dans Wazuh	22
A.1	Nom et adresses du serveur Linux	27
A.2	Vérification des services du serveur Linux	27
A.3	Contrôles complémentaires sur le serveur Linux	27
B.1	Éléments du scénario d'authentification	28
B.2	Page d'erreur 404 générée par IIS	28
C.1	Détail d'un événement LOGIN_FAILED reçu par Wazuh	29
C.2	Règle OPNsense bloquant SSH depuis la zone ATTACK	30

1 Présentation de CIH Bank et du stage

1.1 Présentation de CIH Bank



FIGURE 1.1. Logo de CIH Bank

CIH Bank est un établissement bancaire marocain qui propose des services aux particuliers, aux professionnels et aux entreprises. Comme toute organisation financière, la banque s'appuie sur un système d'information disponible, fiable et surveillé. La cybersécurité participe donc directement à la continuité des services et à la protection des données [CIH Bank, 2025].

TABLE 1.1. Informations générales sur le stage

Organisme d'accueil	CIH Bank
Secteur	Banque et services financiers
Période	Du 22 juin 2026 au 15 août 2026
Sujet	Mise en place d'un laboratoire SOC virtualisé
Filière	Cybersecurity

1.2 Organisation de l'entreprise

L'organigramme fourni présente les principales directions de CIH Bank. La branche dédiée aux services technologiques comprend notamment la sécurité des systèmes d'information et le pôle des systèmes d'information. C'est dans ce contexte que s'inscrit le sujet du stage.

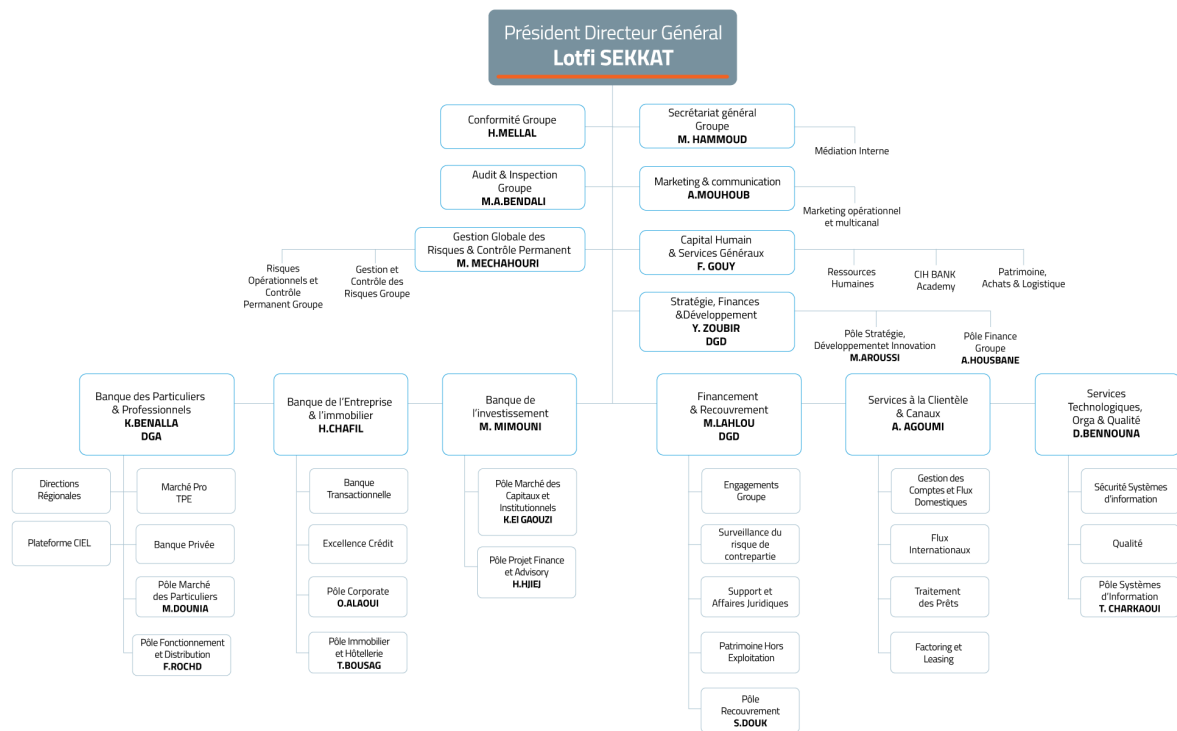


FIGURE 1.2. Organigramme général de CIH Bank fourni dans les ressources du stage

1.3 Contexte du stage

Le projet consiste à construire un environnement isolé permettant de comprendre le fonctionnement d'un SOC sans intervenir sur le réseau de production. Le laboratoire reproduit une chaîne simple :

Générer un événement → collecter le journal → détecter → analyser l'alerte

Les tests sont réalisés avec des machines virtuelles et des adresses privées. Ils concernent le réseau, un serveur Linux, un serveur Windows et une application Web de démonstration.

1.4 Enjeux de sécurité

Dans un environnement bancaire, la supervision ne consiste pas uniquement à bloquer une connexion. Elle doit aussi conserver une trace exploitable afin de comprendre l'événement et d'identifier la machine concernée. Le laboratoire reprend cette logique à une échelle réduite.

TABLE 1.2. Enjeux de sécurité représentés dans le laboratoire

Enjeu	Exemple	Réponse du laboratoire
Confidentialité	Accès non autorisé	Filtrage réseau et contrôle des connexions
Intégrité	Modification d'un fichier	Surveillance FIM
Disponibilité	Activité répétée ou scan	Blocage OPNsense et détection Suricata
Traçabilité	Recherche après incident	Centralisation des journaux dans Wazuh

À retenir. Le SOC regroupe des informations provenant de plusieurs sources afin de transformer un journal isolé en événement compréhensible.

1.5 Objectifs du projet

L'objectif principal est de mettre en place un petit laboratoire capable de centraliser et de visualiser des événements de sécurité. Les objectifs retenus sont :

- séparer le réseau de test, le réseau des serveurs et le réseau de supervision ;
- filtrer les communications avec OPNsense ;
- centraliser les journaux dans Wazuh ;
- superviser un serveur Linux et un serveur Windows ;
- détecter quelques scénarios simples et contrôlés ;
- conserver des captures permettant de vérifier les résultats.

1.6 Déroulement du stage

TABLE 1.3. Étapes principales du travail

Étape	Travail réalisé
Cadrage	Compréhension du sujet et choix du périmètre
Conception	Définition des machines, des zones et des flux
Mise en place	Installation et connexion des composants
Validation	Exécution des tests et collecte des captures
Documentation	Analyse des résultats et rédaction du rapport

1.7 Livrables réalisés

Le travail produit quatre livrables principaux :

- une architecture réseau segmentée ;
- un ensemble de machines virtuelles configurées ;
- des règles et scénarios de validation ;

— un dossier de captures et le présent rapport.

1.8 Organisation du rapport

Le rapport présente d’abord l’entreprise et l’environnement utilisé, puis l’architecture, la mise en place et les tests. Seuls les résultats visibles dans les captures sont retenus.

2 Environnement de travail et outils utilisés

2.1 Notions utiles

Le laboratoire reprend trois fonctions simples d'un SOC :

TABLE 2.1. Fonctions représentées dans le laboratoire

Fonction	Rôle dans le projet
Collecte	Récupérer les journaux réseau, Linux et Windows
Détection	Appliquer des règles et produire des alertes
Analyse	Rechercher les événements dans Wazuh Threat Hunting

Wazuh joue le rôle de plateforme SIEM et de supervision des hôtes. OPNsense filtre le réseau, tandis que Suricata inspecte le trafic. Sysmon complète les journaux Windows avec des informations sur les processus.

2.2 Fonctionnement simplifié d'une supervision

Une alerte est le résultat de plusieurs opérations successives. La source crée d'abord un journal. Celui-ci est transporté vers Wazuh, décodé, comparé à une règle, puis affiché si les conditions sont vérifiées.

TABLE 2.2. Étapes de traitement d'un événement

Étape	Élément	Résultat
Production	Pare-feu, système ou application	Ligne de journal
Transport	Agent Wazuh ou Syslog	Événement transmis
Décodage	Décodeur Wazuh	Champs identifiés
Détection	Règle Wazuh	Niveau et description
Analyse	Threat Hunting	Alerte consultable

Cette chaîne est importante : une règle ne peut fonctionner que si le journal est reçu et correctement décodé. La gestion des journaux constitue donc une base du travail d'un SOC [Kent and Souppaya, 2006].

2.2.1 NIDS et HIDS

TABLE 2.3. Complémentarité de la détection réseau et hôte

Type	Point observé	Exemple du projet
NIDS	Trafic traversant le réseau	Scan SSH détecté par Suricata
HIDS	Activité interne d'une machine	Processus Sysmon ou modification FIM

2.3 Machines virtuelles



FIGURE 2.1. Principales machines et plateformes du laboratoire

2.4 Outils de sécurité et de supervision



FIGURE 2.2. Outils utilisés pour la détection et l'analyse

TABLE 2.4. Rôle des principaux outils

Outil	Catégorie	Utilisation
OPNsense	Pare-feu	Séparation des zones et blocage de flux
Wazuh	SIEM/HIDS	Agents, règles, alertes et Threat Hunting
Suricata	NIDS	Inspection du trafic avec ET Open
Sysmon	Journal Windows	Création de processus et détails associés
Apache	Serveur Web Linux	Hébergement de l'application de test
IIS	Serveur Web Windows	Génération de journaux HTTP

Les rôles décrits correspondent aux fonctions documentées par les projets Wazuh, OPNsense, Suricata et Microsoft [Wazuh, 2026b, OPNsense, 2026b, Open Information Security Foundation, 2026, Russinovich and Garnier, 2026].

2.5 Inventaire des machines

TABLE 2.5. Machines du laboratoire

Machine	Adresse	Fonction
Kali Linux	10.10.10.10	Génération des tests
OPNsense	10.10.10.1	Passerelle de la zone ATTACK
OPNsense	192.168.56.1	Passerelle de la zone LAN
OPNsense	172.16.10.1	Passerelle de la zone SIEM
SRV Linux Web	192.168.56.20	Apache, SSH, application et FIM
SRV Windows	192.168.56.30	IIS, Sysmon et agent Wazuh
Wazuh SIEM	172.16.10.10	Centralisation et analyse

2.6 Réseaux virtuels et interfaces

VMware relie les cartes réseau virtuelles aux trois segments du laboratoire. OPNsense possède une interface dans chaque zone et devient le seul point de passage entre elles. Une machine de la zone ATTACK ne communique donc pas directement avec la zone SIEM.

TABLE 2.6. Connexion des machines aux réseaux virtuels

Machine	Zone	Connexion
Kali Linux	ATTACK	Une carte réseau de test
SRV Linux Web	LAN	Une carte réseau serveur
SRV Windows	LAN	Une carte réseau serveur
Wazuh SIEM	SIEM	Une carte réseau de supervision
OPNsense	ATTACK, LAN, SIEM	Trois interfaces de routage

Cette séparation facilite les tests : une règle peut être appliquée à une zone précise sans modifier

les autres réseaux. Elle améliore également la lecture des journaux, car l'interface d'origine indique le chemin suivi par le trafic.

À retenir. La virtualisation permet de regrouper tout le laboratoire sur une même plateforme, tout en gardant des rôles et des réseaux séparés.

3 Architecture du laboratoire

3.1 Topologie générale

L'architecture est divisée en trois zones. Cette séparation permet de contrôler les échanges et de distinguer clairement la machine de test, les serveurs supervisés et la plateforme de sécurité.

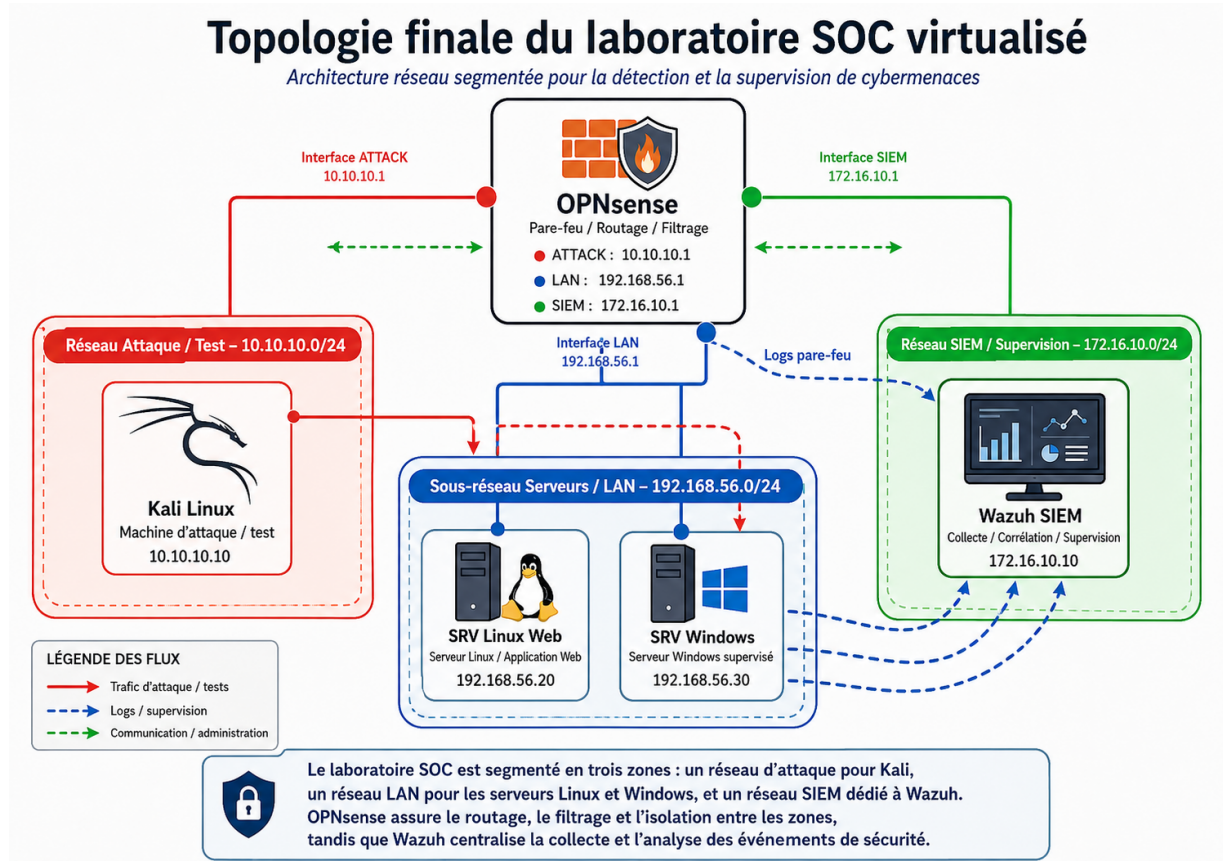


FIGURE 3.1. Topologie du laboratoire SOC virtualisé

TABLE 3.1. Segmentation réseau

Zone	Réseau	Éléments
ATTACK	10.10.10.0/24	Kali Linux et interface OPNsense
LAN	192.168.56.0/24	Serveurs Linux et Windows
SIEM	172.16.10.0/24	Serveur Wazuh

3.2 Règles de circulation

OPNsense applique un filtrage avec suivi d'état. Lorsqu'une connexion est autorisée, le pare-feu suit la session et accepte le trafic de retour correspondant. Les autres communications restent bloquées si aucune règle ne les autorise.

TABLE 3.2. Flux nécessaires au fonctionnement du laboratoire

Source	Destination	Port/service	Utilité
Kali Linux	Serveurs LAN	22, 80	Scénarios contrôlés
Agents Wazuh	Wazuh SIEM	TCP 1514	Envoi des événements
OPNsense	Wazuh SIEM	Syslog 514	Journaux du pare-feu
Suricata	Wazuh SIEM	EVE/Syslog	Alertes du NIDS
Administrateur	Interfaces Web	HTTPS	Configuration et analyse

Les flux d'administration sont séparés des actions de test. Cette distinction évite de confondre une opération normale avec un événement de sécurité.

3.3 Circulation des événements

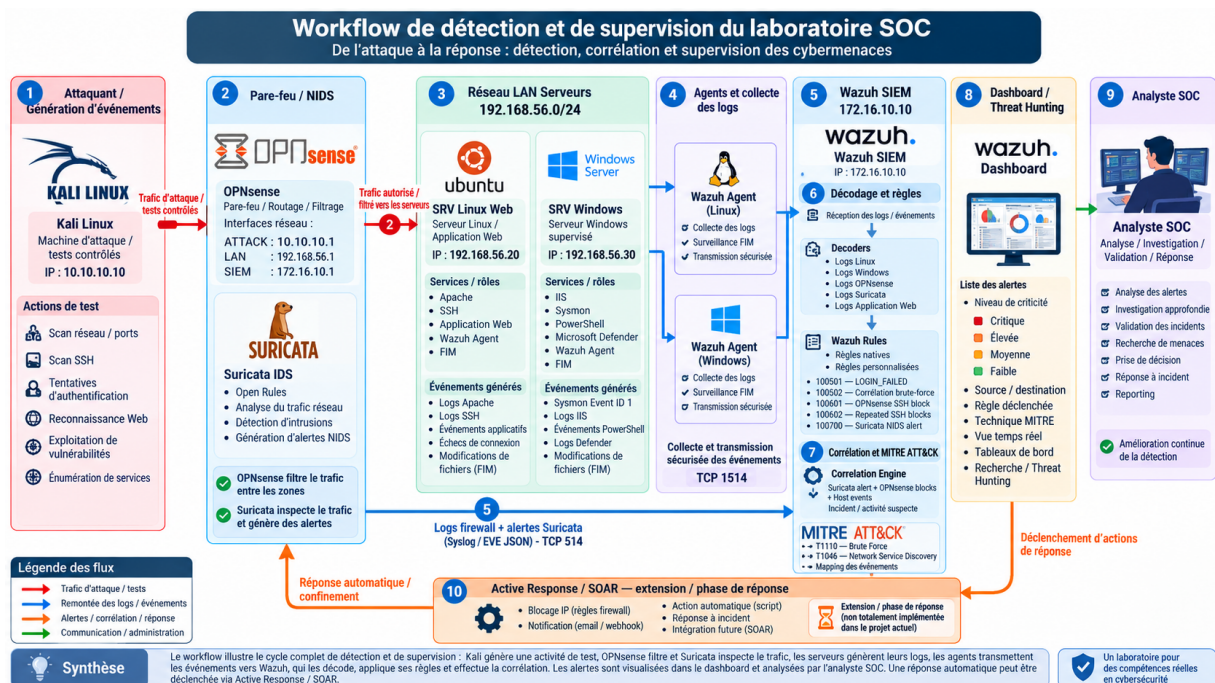


FIGURE 3.2. Workflow de détection ; le bloc Active Response/SOAR représente une évolution future non implémentée

Le fonctionnement du laboratoire peut être résumé en cinq étapes :

1. Kali Linux génère une action de test ;
2. OPNsense autorise ou bloque le trafic selon les règles ;
3. les hôtes et Suricata produisent des journaux ;
4. les agents et Syslog transmettent les événements à Wazuh ;
5. Wazuh applique les règles et affiche les alertes dans Threat Hunting.

3.4 Sources de journaux

TABLE 3.3. Journaux exploités

Source	Type de journal	Exemple d'événement
OPNsense	Pare-feu/Syslog	Connexion SSH bloquée
Suricata	EVE JSON/Syslog	Scan SSH détecté
Linux	Système et application	Authentification, FIM
Windows	Event Log et Sysmon	Création de processus
IIS	Journal HTTP	Requête retournant 404

3.5 Chaînes de détection

La même plateforme reçoit des événements de nature différente. Le chemin exact dépend de la source :

TABLE 3.4. Exemples de chaînes de détection

Scénario	Chemin suivi
Scan SSH	Kali → OPNsense → Suricata → EVE/Syslog → Wazuh
Échec Web	Application → journal JSON → agent Wazuh → décodeur → règle
Processus Windows	Sysmon → Event Log → agent Wazuh → Threat Hunting
Modification FIM	Fichier → agent Wazuh → comparaison d'intégrité → alerte

3.6 Critères de validation

Un scénario est considéré comme validé lorsque les quatre points suivants sont vérifiés :

1. l'action de test est identifiable ;
2. la source produit un journal ;
3. Wazuh reçoit un événement contenant les champs attendus ;
4. la règle et son niveau sont visibles dans l'alerte.

À retenir. Une capture de tableau de bord ne suffit pas seule : elle doit pouvoir être reliée à une action, une source et une règle.

3.7 Périmètre retenu

Le laboratoire est volontairement limité à quelques machines et scénarios faciles à reproduire. Il ne contient ni haute disponibilité ni SOAR opérationnel. Ces éléments sont présentés uniquement comme perspectives.

À retenir. L'architecture privilégie une chaîne courte et vérifiable : réseau, hôtes, collecte Wazuh, règle de détection et alerte visible.

4 Mise en place du laboratoire

4.1 Ordre de déploiement

La mise en place a suivi un ordre progressif afin d'éviter de rechercher plusieurs pannes en même temps.

TABLE 4.1. Ordre de déploiement des composants

Étape	Composant	Vérification
1	Réseaux VMware	Cartes connectées aux bonnes zones
2	OPNsense	Interfaces et passerelles accessibles
3	Serveur Wazuh	Tableau de bord disponible
4	Serveurs Linux/Windows	Services et adresses configurés
5	Agents Wazuh	Agents actifs dans le tableau de bord
6	Suricata et règles	Premiers événements reçus

4.2 Configuration d'OPNsense

OPNsense possède une interface dans chaque zone. Il assure le routage entre les réseaux et applique les règles de filtrage.

WAN (wan)	em0	dhcp	192.168.9.133/24	fe80::20c:29ff:fe6f:df0/64	192.168.9.2	default 192.168.9.0/24 fe80::%em0/64
LAN (lan)	em1	static	192.168.56.1/24			192.168.56.0/24
SIEM (opt1)	em2	static	172.16.10.1/24			172.16.10.0/24
ATTACK (opt2)	em3	static	10.10.10.1/24			10.10.10.0/24
Loopback (lo0)	lo0	static	127.0.0.1/8	::1/128 fe80::1/64		10.10.10.1 127.0.0.1 172.16.10.1 192.168.9.133 192.168.56.1 ::1 fe80::20c:29ff:fe6f:df0%lo0 fe80::%lo0/64 fe80::1%lo0

FIGURE 4.1. Interfaces réseau configurées sur OPNsense

La politique de filtrage autorise uniquement les communications nécessaires au laboratoire. Les tests SSH provenant de la zone ATTACK peuvent ainsi être bloqués et journalisés.

Firewall: Rules

🔍 All rules Categories 🔍 Search 500

	Icons	Version	Protocol	Source	Port	Destination	Port	Gateway	Description	Commands
Automatically generated rules										
Floating rules										
☐				IPv4	*	SIEM network	*	*	Allow SIEM outbound during installation	
Interface rules										
☐				LAN	IPv4	*	LAN network	*	Default allow LAN to any rule	
☐				LAN	IPv6	*	LAN network	*	Default allow LAN IPv6 to any rule	
☐				SIEM	IPv4	*	*	*	TEMP_ALLOW_SIEM_DURING_INSTALLATION	
☐				ATTACK	IPv4	TCP	ATTACK network	*	test block ssh kali	
☐				ATTACK	IPv4	*	ATTACK network	*	TEMP_ALLOW_KALI_INTERNET	

Showing 1 to 27 of 27 entries

FIGURE 4.2. Règles de filtrage appliquées à la zone ATTACK

4.3 Installation de Wazuh

Le serveur Wazuh est installé dans la zone SIEM. Les agents Linux et Windows lui transmettent les événements des deux serveurs supervisés.

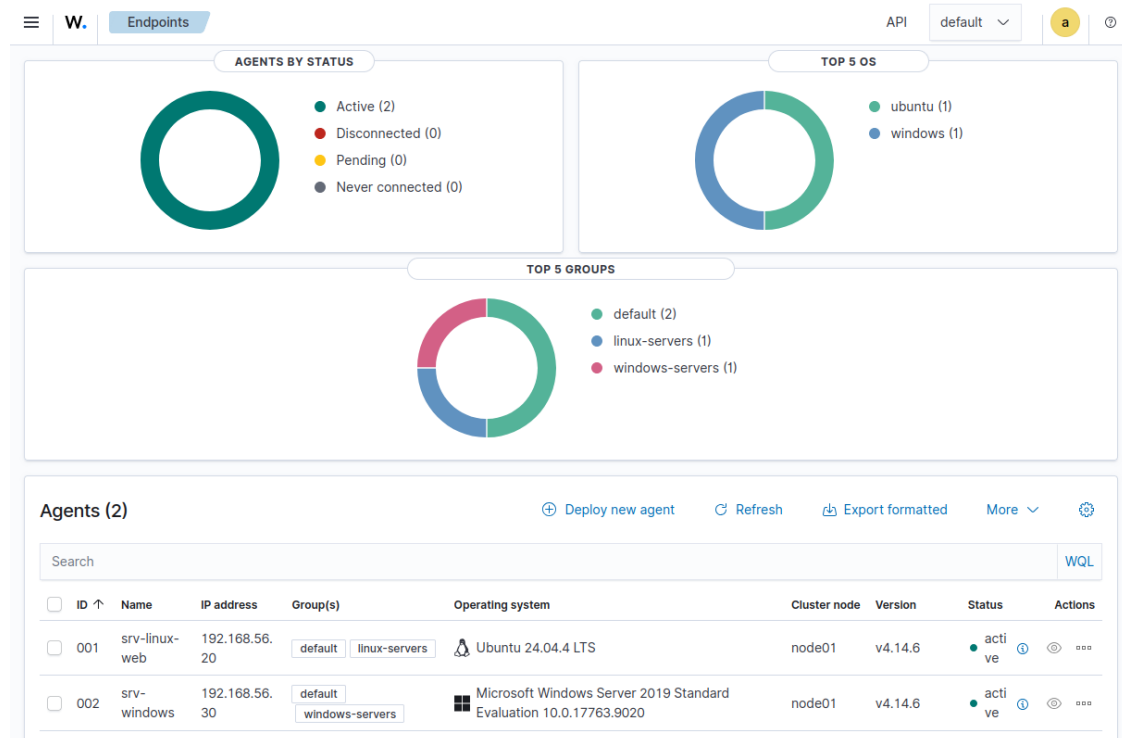


FIGURE 4.3. Agents Linux et Windows actifs dans Wazuh

4.3.1 Communication entre les agents et le serveur

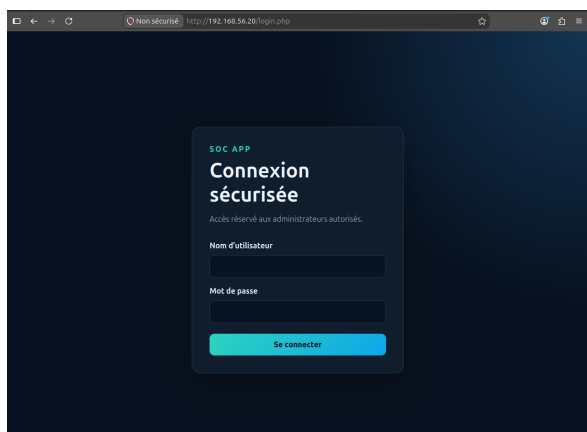
Chaque agent possède un identifiant et transmet les événements de sa machine au serveur Wazuh. Le serveur associe ensuite l'événement au nom et à l'adresse de l'agent. Cette association est visible dans les champs `agent.name`, `agent.ip` et `agent.id`.

TABLE 4.2. Événements collectés par les agents

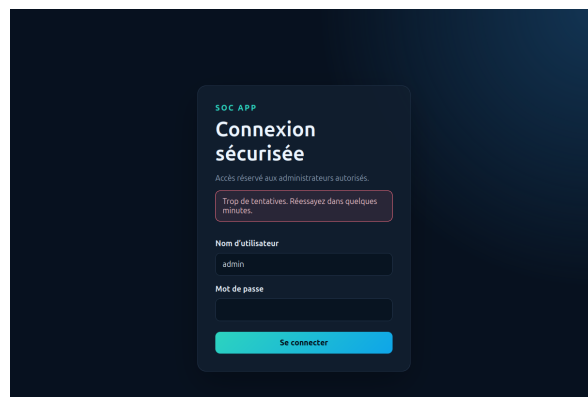
Agent	Principales données collectées
SRV Linux Web	Journaux système, application JSON et FIM
SRV Windows	Event Log, Sysmon, IIS et FIM

4.4 Serveur Linux et application Web

Le serveur Ubuntu héberge Apache, SSH, l'agent Wazuh et une application Web de test. L'application enregistre les échecs d'authentification dans un journal surveillé par Wazuh.



(a) Application hébergée par Apache



(b) Blocage après plusieurs échecs

FIGURE 4.4. Application Web utilisée pour le scénario d'authentification

4.5 Serveur Windows

Le serveur Windows reçoit IIS, Sysmon et l'agent Wazuh. IIS fournit les journaux Web, tandis que Sysmon ajoute des informations sur les processus. Le FIM de Wazuh surveille également les modifications de fichiers.

4.6 Intégration de Suricata

Suricata est activé sur l'interface ATTACK avec les règles ET Open. Ses alertes sont transmises à Wazuh afin d'être recherchées dans Threat Hunting.

TABLE 4.3. Paramètres retenus pour Suricata

Paramètre	Valeur
Interface surveillée	ATTACK
Jeu de règles	Emerging Threats Open / ET Open
Format d'événement	EVE JSON
Destination	Wazuh SIEM
Validation	Signature ET SCAN Potential SSH Scan

4.7 Vérifications de base

Avant les scénarios, l'adresse du serveur Linux et la réponse du service Apache sont contrôlées. Ces tests simples confirment que le problème ne vient pas de la configuration réseau ou du serveur Web.

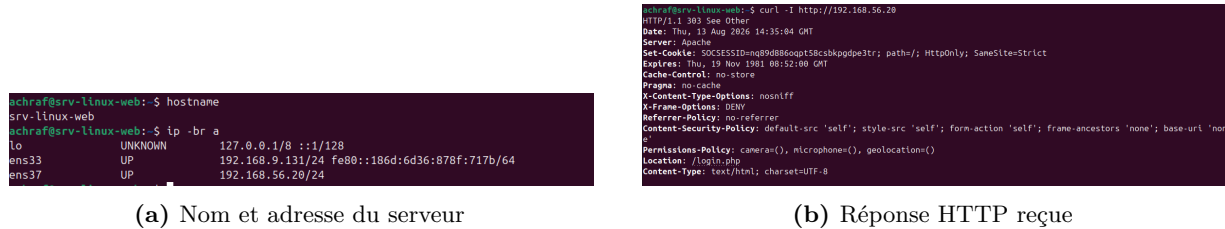


FIGURE 4.5. Contrôles réalisés avant les scénarios

TABLE 4.4. Contrôles réalisés après l'installation

Composant	Contrôle
OPNsense	Interfaces visibles et règles actives
Wazuh	Serveur accessible et agents connectés
Linux	Apache, SSH et agent Wazuh actifs
Windows	IIS, Sysmon et agent Wazuh actifs
Suricata	Alertes reçues dans Wazuh

À retenir. La mise en place est validée lorsque les machines communiquent selon les règles et que les événements sont visibles dans Wazuh.

5 Tests et résultats

5.1 Principe de validation

Chaque test suit la même logique : une action est lancée depuis une machine du laboratoire, puis le résultat est vérifié dans OPNsense, dans le journal de l'hôte ou dans Wazuh.

TABLE 5.1. Scénarios retenus

Scénario	Source	Résultat attendu
Échecs de connexion Web	Application Linux	Alerte et corrélation
Tentatives SSH bloquées	OPNsense	Règles 100601/100602
Création de processus	Sysmon	Event ID 1 visible
Scan SSH	Suricata	Alerte 100700
Modification de fichier	Wazuh FIM	Alerte d'intégrité
Requête IIS invalide	IIS	Code HTTP 404 collecté

5.2 Règles applicatives Wazuh

L'application Web écrit ses événements d'authentification au format JSON. Les règles personnalisées Wazuh reconnaissent ces événements, puis les corrélient selon l'adresse IP et la période observée. Cette partie présente les cinq règles applicatives utilisées dans le laboratoire.

TABLE 5.2. Explication des règles applicatives Wazuh

Identifiant	Niveau	Explication
100501	3	Reconnaît un événement <code>LOGIN_FAILED</code> . L'option <code>no_log</code> évite de créer une alerte isolée : cette règle sert de base aux corrélations.
100502	10	Détecte cinq échecs provenant de la même adresse IP en 300 secondes. Elle signale une tentative de brute force (MITRE ATT&CK T1110).
100510	3	Reconnaît un événement <code>LOGIN_SUCCESS</code> . Comme la règle 100501, elle sert de règle de base et n'est pas affichée seule.
100503	12	Signale une connexion réussie après trois échecs issus de la même adresse IP en cinq minutes. Ce comportement peut indiquer un compte compromis (MITRE ATT&CK T1078).
100504	10	Détecte l'événement <code>LOGIN_RATE_LIMIT</code> lorsque l'application limite les tentatives après plusieurs échecs.

Les règles 100501 et 100510 identifient donc les événements simples. Les règles 100502 et 100503 analysent leur enchaînement, tandis que 100504 confirme que la protection applicative a limité le rythme des tentatives.

```

achraf@wazuh-siem:~$ sudo grep -n -A 15 -B 3 '100501' /var/ossec/etc/rules/soc_app_rules.xml
1-<group name="soc_app,web_application,attack_detection,">
2-
3- <!-- Échec de connexion conservé uniquement pour la corrélation -->
4- <rule id="100501" level="3">
5-   <decoded_as>json</decoded_as>
6-   <field name="event" type="pcre2">^LOGIN_FAILED$</field>
7-   <description>SOC App: échec de connexion depuis $(src_ip)</description>
8-   <options>no_log</options>
9- </rule>
10-
11- <!-- Cinq échecs depuis la même IP en cinq minutes -->
12- <rule id="100502" level="10" frequency="5" timeframe="300" ignore="60">
13-   <if_matched_sid>100501</if_matched_sid>
14-   <same_field>src_ip</same_field>
15-   <description>SOC App: tentative de brute force Web depuis $(src_ip)</description>
16-   <group>authentication_failures,brute_force,</group>
17-   <mitre>
18-     <id>T1110</id>
19-   </mitre>
20- </rule>
21- <!-- Connexion réussie conservée uniquement comme événement parent -->
22- <rule id="100510" level="3">
23-   <decoded_as>json</decoded_as>
24-   <field name="event" type="pcre2">^LOGIN_SUCCESS$</field>
25-   <description>SOC App: connexion réussie depuis $(src_ip)</description>
26-   <options>no_log</options>
27- </rule>
28-
29- <!-- Connexion réussie après trois échecs depuis la même IP -->
30- <rule id="100503" level="12" frequency="4" timeframe="300" ignore="60">
31-   <if_sid>100510</if_sid>
32-   <if_matched_sid>100501</if_matched_sid>

```

FIGURE 5.1. Règles applicatives de détection et de corrélation des échecs

```

32-   <if_matched_sid>100501</if_matched_sid>
33-   <same_field>src_ip</same_field>
34-   <description>SOC App: connexion réussie après plusieurs échecs depuis $(src_ip) pour $(dstuser)</description>
35-   <group>authentication_success,possible_account_compromise,</group>
36-   <mitre>
37-     <id>T1078</id>
38-   </mitre>
39- </rule>
40-
41- <!-- L'application a appliqué sa limitation de tentatives -->
42- <rule id="100504" level="10">
43-   <decoded_as>json</decoded_as>
44-   <field name="event" type="pcre2">^LOGIN_RATE_LIMIT$</field>
45-   <description>SOC App: limitation déclenchée après plusieurs échecs depuis $(src_ip)</description>
46-   <group>authentication_failures,brute_force,</group>
47-   <mitre>
48-     <id>T1078</id>
49-   </mitre>
50- </rule>

```

FIGURE 5.2. Règles applicatives de connexion réussie et de limitation

5.2.1 Exemple de corrélation : règle 100502

L'extrait suivant montre la logique réellement utilisée pour la règle 100502. Elle ne recherche pas directement un mot dans le journal : elle s'appuie sur les alertes déjà produites par la règle 100501.

```

<rule id="100502" level="10" frequency="5"
      timeframe="300" ignore="60">
  <if_matched_sid>100501</if_matched_sid>
  <same_field>src_ip</same_field>
  <description>Tentative de brute force Web</description>
</rule>

```

TABLE 5.3. Lecture des paramètres de la règle 100502

Paramètre	Signification
<code>level="10"</code>	Niveau élevé attribué à l'alerte corrélée.
<code>frequency="5"</code>	Cinq alertes de base sont nécessaires.
<code>timeframe="300"</code>	Les cinq alertes doivent apparaître en cinq minutes.
<code>ignore="60"</code>	Après une alerte, les doublons sont ignorés pendant 60 secondes.
<code>if_matched_sid</code>	La corrélation porte sur la règle de base 100501.
<code>same_field</code>	Les événements doivent concerner la même adresse source.

Cette séparation entre règle unitaire et règle corrélée limite le bruit : un seul échec reste peu critique, alors qu'une répétition depuis la même source produit une alerte plus importante.

5.3 Autres règles Wazuh validées

Les autres scénarios du laboratoire s'appuient sur les règles suivantes :

TABLE 5.4. Règles utilisées pour le réseau et l'intégrité

Identifiant	Niveau	Explication
100601	7	Signale une tentative SSH bloquée et journalisée par OPN-sense.
100602	10	Corrèle plusieurs blocages SSH et augmente la criticité de l'activité répétée.
100700	7	Transforme l'alerte Suricata <i>ET SCAN Potential SSH Scan</i> en alerte Wazuh consultable.
550	7	Règle native FIM indiquant qu'un fichier surveillé a été modifié.

5.4 Échecs d'authentification Web

Une règle unitaire détecte l'événement `LOGIN_FAILED`. Une seconde règle augmente le niveau lorsque plusieurs échecs sont observés sur une courte période.

timestamp	agent.name	rule.description	rule.level	rule.id
Aug 13, 2026 @ 16:41:48.5...	wazuh-siem	Successful sudo to ROOT executed.	3	5402
Aug 13, 2026 @ 16:41:10.5...	srv-linux-web	Successful sudo to ROOT executed.	3	5402
Aug 13, 2026 @ 16:31:32.2...	srv-linux-web	Successful sudo to ROOT executed.	3	5402
Aug 13, 2026 @ 16:31:32.2...	srv-linux-web	Successful sudo to ROOT executed.	3	5402
Aug 13, 2026 @ 15:52:54.3...	srv-linux-web	SOC App: tentative de brute force Web depuis 192.168.56.20	10	100502
Jul 31, 2026 @ 12:27:22.395	srv-linux-web	SOC App: tentative de brute force depuis 192.168.56.20	10	100508
Jul 31, 2026 @ 12:27:18.440	srv-linux-web	SOC App: échec de connexion pour depuis 192.168.56.20	5	100502
Jul 31, 2026 @ 12:27:10.392	srv-linux-web	SOC App: échec de connexion pour depuis 192.168.56.20	5	100502
Jul 31, 2026 @ 12:27:04.386	srv-linux-web	SOC App: échec de connexion pour depuis 192.168.56.20	5	100502
Jul 31, 2026 @ 12:27:00.423	srv-linux-web	SOC App: échec de connexion pour depuis 192.168.56.20	5	100502
Jul 31, 2026 @ 11:48:54.909	srv-linux-web	SOC App: tentative de brute force depuis 192.168.56.20	10	100508
Jul 31, 2026 @ 11:48:52.906	srv-linux-web	SOC App: échec de connexion pour depuis 192.168.56.20	5	100502

FIGURE 5.3. Alertes applicatives visibles dans Wazuh

5.4.1 Lecture de la règle corrélée

t	rule.description	SOC App: tentative de brute force Web depuis 192.168.56.20
#	rule.firedtimes	1
#	rule.frequency	5
t	rule.groups	soc_app, web_application, attack_detection, authentication_failures, brute_force
t	rule.id	100502
#	rule.level	10
🔊	rule.mail	false
t	rule.mitre.id	T1110
t	rule.mitre.tactic	Credential Access
t	rule.mitre.technique	Brute Force

FIGURE 5.4. Champs de la règle applicative corrélée

La capture montre la règle 100502, le niveau 10 et une fréquence de cinq événements. Le groupe `authentication_failures` indique la nature du cas d'usage. L'association MITRE T1110 qualifie la technique de brute force [MITRE, 2026].

TABLE 5.5. Champs utiles pour interpréter la règle

Champ	Interprétation
<code>rule.id</code>	Identifiant de la règle déclenchée
<code>rule.level</code>	Niveau de criticité Wazuh
<code>rule.frequency</code>	Nombre d'occurrences corrélées
<code>rule.mitre.id</code>	Technique MITRE associée

5.5 Blocages SSH par OPNsense

Les tentatives SSH issues de 10.10.10.10 sont bloquées par le pare-feu. Wazuh reçoit les journaux et applique la règle 100601. La répétition déclenche la règle corrélée 100602, de niveau plus élevé.

timestamp	agent.name	rule.description	rule.level	rule.id
Aug 13, 2026 @ 15:07:00.6...	srv-linux-web	Integrity checksum changed.	7	550
Aug 13, 2026 @ 15:07:00.6...	srv-linux-web	Integrity checksum changed.	7	550
Aug 13, 2026 @ 15:05:37.5...	srv-windows	Wazuh agent disconnected.	3	504
Aug 13, 2026 @ 15:03:53.0...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:03:52.0...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:03:51.0...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:03:26.0...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:02:51.9...	wazuh-siem	OPNsense repeated SSH blocks/scan from 10.10.10.10 to 192.168.56.20: suspicious activity.	10	100602
Aug 13, 2026 @ 15:02:35.8...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:02:27.7...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:02:23.7...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:02:21.7...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:02:20.7...	wazuh-siem	OPNsense repeated SSH blocks/scan from 10.10.10.10 to 192.168.56.20: suspicious activity.	10	100602
Aug 13, 2026 @ 15:02:19.7...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601
Aug 13, 2026 @ 15:02:18.7...	wazuh-siem	OPNsense blocked SSH connection from 10.10.10.10 to 192.168.56.20.	7	100601

FIGURE 5.5. Blocages SSH OPNsense corrélés dans Wazuh

TABLE 5.6. Logique des règles OPNsense dans Wazuh

Règle	Niveau	Signification
100601	7	Une connexion SSH a été bloquée
100602	10	Plusieurs blocages sont corrélés

5.6 Création de processus Windows

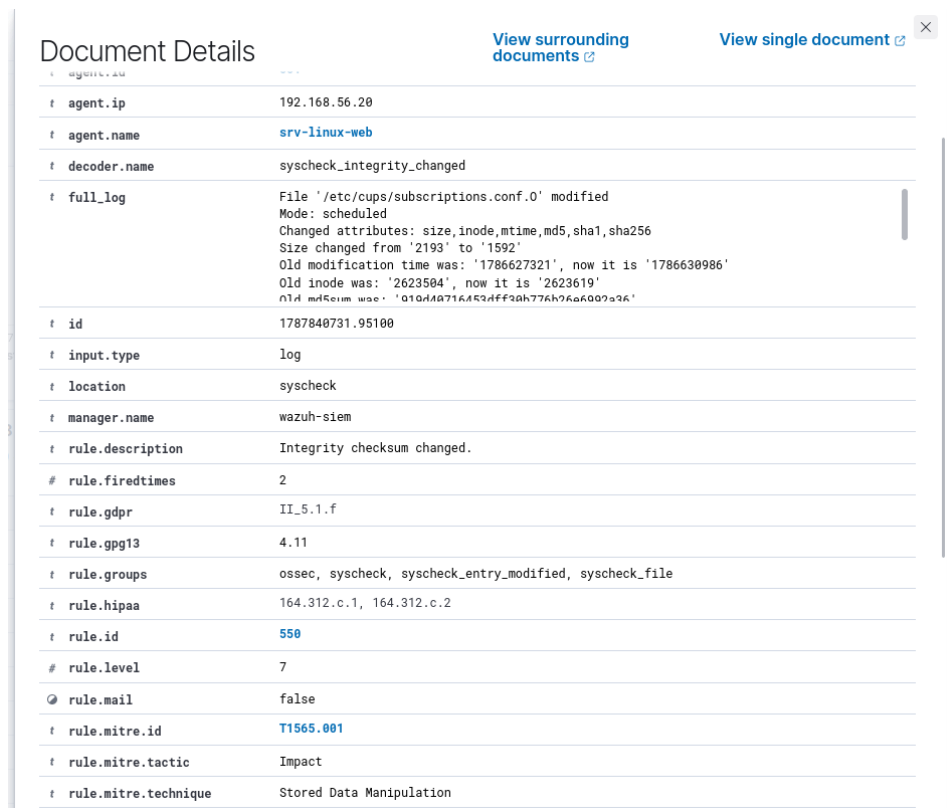
Sysmon enregistre la création d'un processus avec l'Event ID 1. L'agent Wazuh transmet ensuite cet événement au serveur de supervision.

```

C:\ProgramData\Sysmon\logs>type C:\ProgramData\Sysmon\logs\log.txt
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2
```

5.8 Surveillance de l'intégrité

Une modification effectuée sur un fichier surveillé du serveur Linux génère une alerte FIM. Wazuh affiche le fichier concerné et le type de changement.



Document Details		View surrounding documents	View single document
agent.ip	192.168.56.20		
agent.name	srv-linux-web		
decoder.name	syscheck_integrity_changed		
full_log	File '/etc/cups/subscriptions.conf.0' modified Mode: scheduled Changed attributes: size,inode,mtime,md5,sha1,sha256 Size changed from '2193' to '1592' Old modification time was: '1786627321', now it is '1786630986' Old inode was: '2623584', now it is '2623619' Old md5sum was: '010d4a9716d453d4ff30h776h76a6007a36'		
id	1787840731.95100		
input.type	log		
location	syscheck		
manager.name	wazuh-siem		
rule.description	Integrity checksum changed.		
# rule.firedtimes	2		
rule.gdpr	II_5.1.f		
rule.gpg13	4.11		
rule.groups	ossec, syscheck, syscheck_entry_modified, syscheck_file		
rule.hipaa	164.312.c.1, 164.312.c.2		
rule.id	550		
# rule.level	7		
rule.mail	false		
rule.mitre.id	T1565.001		
rule.mitre.tactic	Impact		
rule.mitre.technique	Stored Data Manipulation		

FIGURE 5.8. Modification de fichier détectée par Wazuh FIM

Le FIM calcule une référence pour les fichiers surveillés. Lorsqu'un changement est observé, Wazuh compare l'état courant à cette référence et génère un événement. La règle 550 confirme ici une modification ; elle ne signifie pas automatiquement que le changement est malveillant.

5.9 Supervision IIS

Le serveur Windows héberge une page IIS. Une requête vers une ressource inexistante produit un code 404 qui est collecté puis visible dans Wazuh.

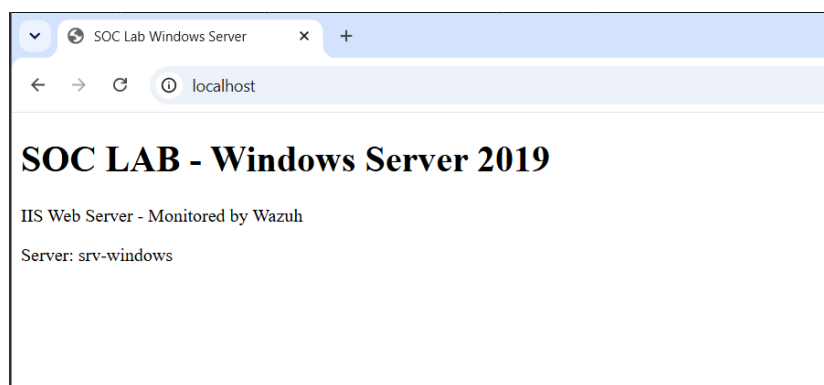


FIGURE 5.9. Page Web IIS du serveur Windows

Document Details		View surrounding documents	View single document
Table	JSON		
† _index	wazuh-alerts-4.x-2026.08.27		
† agent.id	002		
† agent.ip	192.168.56.30		
† agent.name	srv-windows		
† data.action	GET		
† data.id	404		
† data.srcip	192.168.56.30		
† data.srcport	80		
† data.url	/fichier-inexistant-test -		
† data.user_agent	Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64)+AppleWebKit/537.36+(KHTML,+like+Gecko)+Chrome/151.0.0.0+Safari/537.36		
† decoder.name	web-accesslog-iis-default		
† decoder.parent	windows-date-format		
† full_log	2026-08-13 23:59:59 192.168.56.30 GET /fichier-inexistant-test - 80 - 192.168.56.30 Mozilla/5.0+(Windows+NT+10.0;+Win64;+x64)+AppleWebKit/537.36+(KHTML,+like+Gecko)+Chrome/151.0.0.0+Safari/537.36 - 404 0 2 5278 472 4		
† id	1787841912.611073		
† input.type	log		
† location	C:\inetpub\logs\LogFiles\W3SVC1\u_ex260813.log		

FIGURE 5.10. Requête IIS retournant un code 404 visible dans Wazuh

5.10 Synthèse

TABLE 5.8. Résultats des validations

Validation	Résultat
Alertes applicatives et corrélation	Validé
Blocage SSH OPNsense et remontée Wazuh	Validé
Événement Sysmon Event ID 1	Validé
Alerte Suricata ET Open	Validé
Modification FIM	Validé
Journal IIS avec code 404	Validé

6 Bilan et perspectives

6.1 Bilan du laboratoire

Le laboratoire obtenu permet de :

- séparer les machines en trois zones réseau ;
- filtrer et journaliser les communications avec OPNsense ;
- centraliser les événements Linux, Windows et réseau dans Wazuh ;
- détecter des événements simples avec Suricata, Sysmon et le FIM ;
- corréler des répétitions d'événements ;
- rechercher les alertes dans Threat Hunting.

TABLE 6.1. Couverture obtenue

Domaine	Éléments validés
Réseau	Filtrage OPNsense et détection Suricata
Linux	Apache, SSH, application et FIM
Windows	IIS, Sysmon et agent Wazuh
Centralisation	Wazuh, règles et Threat Hunting

6.2 Interprétation technique

Les tests montrent l'intérêt de croiser plusieurs points de vue. OPNsense et Suricata observent le réseau, tandis que les agents Wazuh et Sysmon observent l'activité interne des machines.

TABLE 6.2. Apport des différentes sources

Source	Ce qu'elle montre	Limite
OPNsense	Flux autorisé ou bloqué	Peu de détails sur l'hôte
Suricata	Signature détectée dans le trafic	Dépend du trafic visible
Sysmon	Processus créé sur Windows	Limité à l'hôte configuré
FIM	Fichier modifié	Ne détermine pas seul l'intention
Application/IIS	Action HTTP et résultat	Dépend de la qualité des journaux

La corrélation apporte un niveau supplémentaire : plusieurs événements faibles peuvent devenir une alerte plus importante lorsqu'ils sont répétés dans une même période.

6.3 Limites

TABLE 6.3. Principales limites

Limite	Conséquence
Environnement virtualisé	Résultats limités au laboratoire
Peu de machines	Volume d'événements réduit
Architecture centralisée	Pas de haute disponibilité
Scénarios contrôlés	Ne couvrent pas tous les cas réels

6.4 Perspectives

Les améliorations possibles sont :

- ajouter d'autres postes et serveurs ;
- créer de nouveaux cas d'usage et tableaux de bord ;
- sauvegarder les configurations et automatiser le déploiement ;
- étudier une plateforme SOAR pour la réponse aux incidents.

Le SOAR ne fait pas partie de la réalisation actuelle. Le principal résultat reste une chaîne de supervision complète, de la génération d'un événement jusqu'à son affichage dans Wazuh.

6.5 Recommandations

Pour conserver un laboratoire stable et réutilisable, il est recommandé de sauvegarder la configuration OPNsense, les règles Wazuh et les fichiers de configuration des agents. Les horloges des machines doivent également rester synchronisées afin de comparer correctement les événements.

- conserver un instantané propre de chaque machine virtuelle ;
- documenter les ports ouverts et les règles ajoutées ;
- tester une règle après chaque modification ;
- archiver les captures avec la date et le scénario ;
- séparer les essais de validation des essais exploratoires.

Conclusion générale

Ce stage a permis de concevoir et de mettre en place un laboratoire SOC virtualisé autour d'OPNsense, Wazuh, Suricata, Linux et Windows. L'architecture sépare les machines de test, les serveurs et la supervision.

Les validations montrent que les événements réseau et hôte peuvent être centralisés dans Wazuh. Les scénarios retenus couvrent les tentatives SSH, les échecs d'authentification Web, Sysmon, Suricata, le FIM et IIS.

Le laboratoire reste volontairement limité, mais il constitue une base claire pour ajouter de nouvelles machines, de nouvelles règles et des fonctions de réponse aux incidents.

A Vérifications du serveur Linux

A.1 Identité et adressage

```
achraf@srv-linux-web:~$ hostname
srv-linux-web
achraf@srv-linux-web:~$ ip -br a
lo                UNKNOWN      127.0.0.1/8 ::1/128
ens33             UP            192.168.9.131/24 fe80::186d:6d36:878f:717b/64
ens37             UP            192.168.56.20/24
```

FIGURE A.1. Nom et adresses du serveur Linux

A.2 Services Apache et SSH

```
achraf@srv-linux-web:~$ sudo systemctl status apache2 --no-pager
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-08-13 14:06:29 +01; 1h 24min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 1208 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 1208 (apache2)
    Tasks: 6 (limit: 1160)
   Memory: 2.1M (peak: 33.5M swap: 8.8M swap peak: 8.8M)
      CPU: 601ms
   CGroup: /system.slice/apache2.service
           └─1208 /usr/sbin/apache2 -k start
             └─1319 /usr/sbin/apache2 -k start
               └─1320 /usr/sbin/apache2 -k start
                 └─1321 /usr/sbin/apache2 -k start
                   └─1322 /usr/sbin/apache2 -k start
                     └─1324 /usr/sbin/apache2 -k start

 août 13 14:06:28 srv-linux-web systemd[1]: Starting apache2.service - The Apache HTTP Server...
 août 13 14:06:29 srv-linux-web systemd[1]: Started apache2.service - The Apache HTTP Server.
```

(a) Service Apache actif

```
achraf@srv-linux-web:~$ sudo systemctl status ssh --no-pager
● ssh.service - OpenSSH Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-08-13 14:06:29 +01; 1h 25min ago
     Docs: man:ssh(8)
   Process: 1216 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Main PID: 1230 (sshd)
    Tasks: 1 (limit: 1160)
   Memory: 160.0K (peak: 2.3M swap: 1.1M swap peak: 1.1M)
      CPU: 39ms
   CGroup: /system.slice/ssh.service
           └─1230 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

 août 13 14:06:28 srv-linux-web systemd[1]: Starting ssh.service - OpenSSH Secure Shell server...
 août 13 14:06:29 srv-linux-web sshd[1230]: Server listening on 0.0.0.0 port 22.
 août 13 14:06:29 srv-linux-web sshd[1230]: Server listening on :: port 22.
 août 13 14:06:29 srv-linux-web systemd[1]: Started ssh.service - OpenSSH Secure Shell server.
```

(b) Service SSH actif

FIGURE A.2. Vérification des services du serveur Linux

A.3 Agent Wazuh et réponse HTTP

```
achraf@srv-linux-web:~$ sudo systemctl status wazuh-agent --no-pager
● wazuh-agent.service - Wazuh agent
   Loaded: loaded (/usr/lib/systemd/system/wazuh-agent.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-08-13 14:06:37 +01; 1h 27min ago
     Docs: https://wazuh.com/docs
   Process: 1249 ExecStart=/usr/bin/env /var/ossec/bin/wazuh-control start (code=exited, status=0/SUCCESS)
  Main PID: 1249 (wazuh-agent)
    Tasks: 22 (limit: 1160)
   Memory: 11.5M (peak: 938.0M swap: 21.5M swap peak: 22.0M)
      CPU: 1min 11.068s
   CGroup: /system.slice/wazuh-agent.service
           └─1249 /var/ossec/bin/wazuh-execd
             └─1515 /var/ossec/bin/wazuh-agentd
               └─1514 /var/ossec/bin/wazuh-syscheckd
                 └─1586 /var/ossec/bin/wazuh-logcollector
                   └─1683 /var/ossec/bin/wazuh-modulesd
```

(a) Agent Wazuh actif

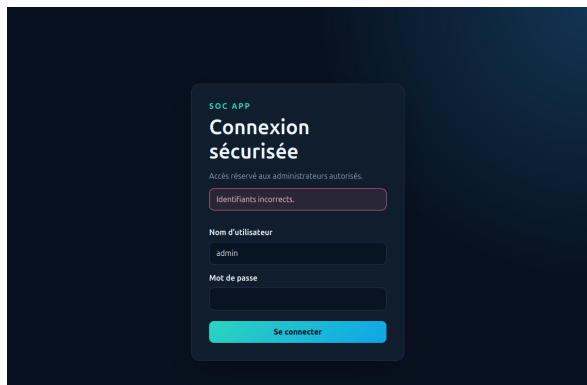
```
achraf@srv-linux-web:~$ curl -I http://192.168.56.20
HTTP/1.1 303 See other
Date: Thu, 13 Aug 2026 14:35:04 GMT
Server: Apache/2.4.18
Set-Cookie: SESSIONID=mgP4866ogpt5Bccbhpgdp3tr; path=/; HttpOnly; SameSite=Strict
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store
Pragma: no-cache
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
Referer-Policy: no-referrer
Content-Security-Policy: default-src 'self'; style-src 'self'; form-action 'self'; frame-ancestors 'none'; base-uri 'none'
Permissions-Policy: camera=(), microphone=(), geolocation=()
Location: /login.php
Content-Type: text/html; charset=UTF-8
```

(b) Réponse HTTP du serveur

FIGURE A.3. Contrôles complémentaires sur le serveur Linux

B Preuves complémentaires

B.1 Étapes du scénario applicatif



(a) Échec de connexion

```
tail -f /var/log/soc-app/security.json | grep -m 10 'LOGIN_FAILED' | tail -n 20
/var/log/soc-app/security.json:17:{"timestamp": "2026-07-31T10:36:00", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:18:{"timestamp": "2026-07-31T10:36:02", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:19:{"timestamp": "2026-07-31T10:36:04", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:20:{"timestamp": "2026-07-31T10:36:06", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:21:{"timestamp": "2026-07-31T10:36:08", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:22:{"timestamp": "2026-07-31T10:40:44", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:23:{"timestamp": "2026-07-31T10:40:47", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:24:{"timestamp": "2026-07-31T10:40:49", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:25:{"timestamp": "2026-07-31T10:40:52", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:26:{"timestamp": "2026-07-31T10:40:54", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:33:{"timestamp": "2026-07-31T11:26:59", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
/var/log/soc-app/security.json:34:{"timestamp": "2026-07-31T11:27:03", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "u
ser": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:152.0) Gecko/2
6100181 Firefox/152.0", "username": "admin", "reason": "invalid_credentials"}
```

(b) Journal de l'application

FIGURE B.1. Éléments du scénario d'authentification

B.2 Erreur IIS côté navigateur

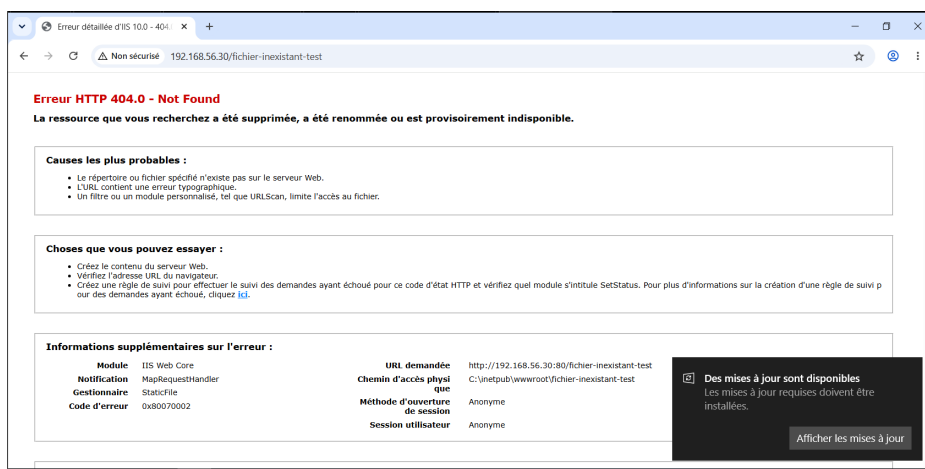
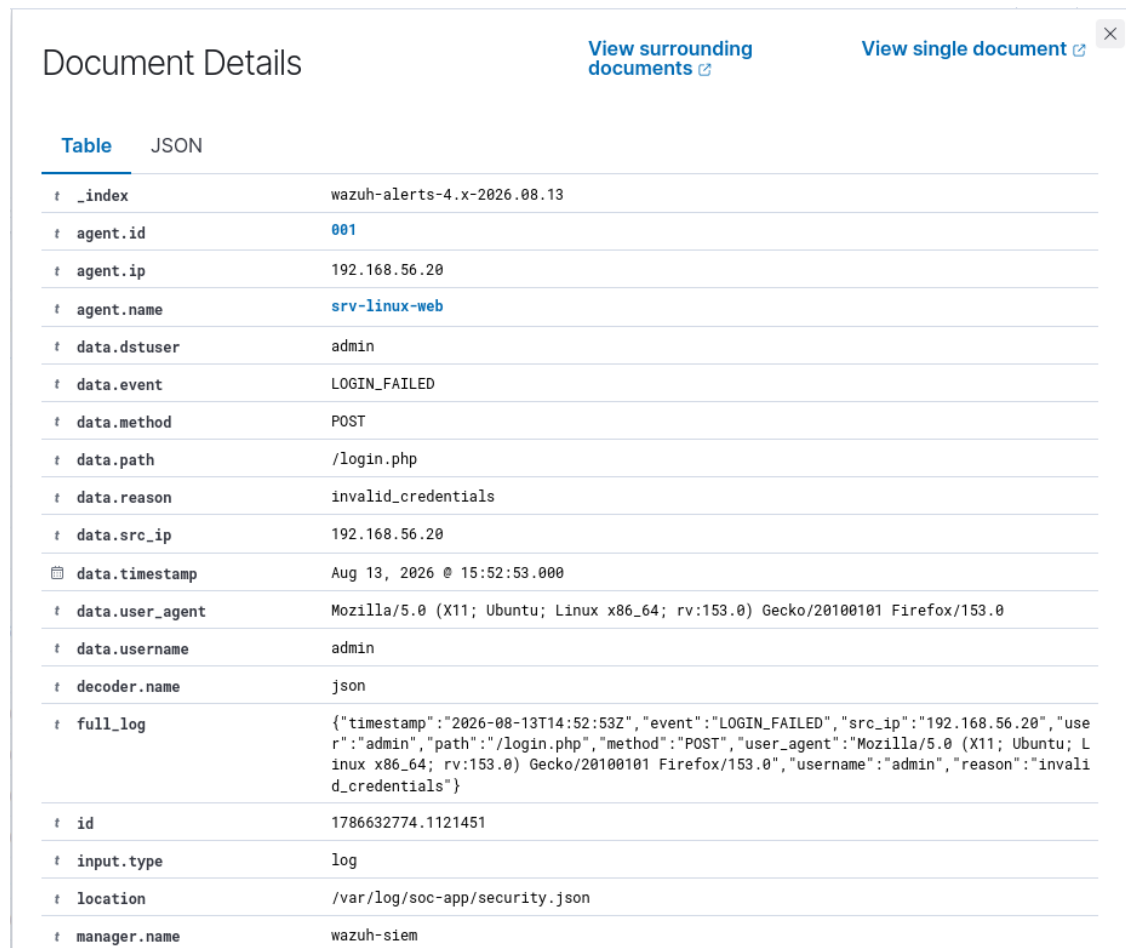


FIGURE B.2. Page d'erreur 404 générée par IIS

C Lecture des événements Wazuh

C.1 Détail d'un événement applicatif



The screenshot shows the 'Document Details' page in the Wazuh interface. It features a table with various fields related to a security event. The table is titled 'Table' and 'JSON'. The fields include agent information, event details, and a full log entry.

<code>_index</code>	wazuh-alerts-4.x-2026.08.13
<code>agent.id</code>	001
<code>agent.ip</code>	192.168.56.20
<code>agent.name</code>	srv-linux-web
<code>data.dstuser</code>	admin
<code>data.event</code>	LOGIN_FAILED
<code>data.method</code>	POST
<code>data.path</code>	/login.php
<code>data.reason</code>	invalid_credentials
<code>data.src_ip</code>	192.168.56.20
<code>data.timestamp</code>	Aug 13, 2026 @ 15:52:53.000
<code>data.user_agent</code>	Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:153.0) Gecko/20100101 Firefox/153.0
<code>data.username</code>	admin
<code>decoder.name</code>	json
<code>full_log</code>	{ "timestamp": "2026-08-13T14:52:53Z", "event": "LOGIN_FAILED", "src_ip": "192.168.56.20", "user": "admin", "path": "/login.php", "method": "POST", "user_agent": "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:153.0) Gecko/20100101 Firefox/153.0", "username": "admin", "reason": "invalid_credentials" }
<code>id</code>	1786632774.1121451
<code>input.type</code>	log
<code>location</code>	/var/log/soc-app/security.json
<code>manager.name</code>	wazuh-siem

FIGURE C.1. Détail d'un événement LOGIN_FAILED reçu par Wazuh

TABLE C.1. Champs principaux de l'événement

Champ	Utilité
<code>agent.name</code>	Machine qui a transmis le journal
<code>data.event</code>	Type d'événement applicatif
<code>data.src_ip</code>	Adresse associée à l'action
<code>location</code>	Fichier de journal surveillé
<code>full_log</code>	Ligne complète reçue par Wazuh

C.2 Règle de blocage du pare-feu



FIGURE C.2. Règle OPNsense bloquant SSH depuis la zone ATTACK

TABLE C.2. Lecture de la règle OPNsense

Élément	Valeur observée
Interface	ATTACK
Protocole	TCP/IPv4
Source	Réseau ATTACK
Destination	Alias du serveur Linux
Port	SSH
Action	Blocage et journalisation

La journalisation est nécessaire pour que Wazuh reçoive la trace du blocage. Sans journal du pare-feu, la règle réseau serait appliquée mais aucune alerte ne pourrait être corrélée dans le SIEM.

D Commandes de validation

D.1 Contrôles utilisés

```
ip -br a
systemctl status apache2 --no-pager
systemctl status ssh --no-pager
systemctl status wazuh-agent --no-pager
curl -I http://192.168.56.20
nmap -sS -p 22 192.168.56.30
```

TABLE D.1. Objectif des commandes

Commande	Vérification
ip -br a	Interfaces et adresses IP
systemctl status	État des services Linux
curl -I	Code de réponse HTTP
nmap -sS	Génération du test de scan SSH

Ces commandes sont exécutées uniquement dans le laboratoire. Elles permettent de confirmer l'état de l'environnement avant de rechercher les alertes dans Wazuh.

Bibliographie

- Broadcom. Installing VMware Workstation Pro, 2026. URL <https://knowledge.broadcom.com/external/article/387947/installing-vmware-workstation-pro.html>. Guide officiel d'installation, consulté le 27 août 2026.
- Canonical. Basic installation - Ubuntu Server, 2026. URL <https://ubuntu.com/server/docs/tutorial/basic-installation/>. Documentation officielle, consultée le 27 août 2026.
- CIH Bank. Rapport annuel 2024. Technical report, CIH Bank, 2025. URL https://www.cihbank.ma/sites/default/files/2025-06/Rapport%20annuel%202024_compressed.pdf. Consulté le 27 août 2026.
- Kali Linux. Installing Kali Linux, 2025. URL <https://www.kali.org/docs/installation/hard-disk-install/>. Documentation officielle, consultée le 27 août 2026.
- Karen Kent and Murugiah Souppaya. Guide to computer security log management. Technical Report NIST SP 800-92, National Institute of Standards and Technology, 2006. URL <https://csrc.nist.gov/pubs/sp/800/92/final>.
- Microsoft. Install Windows Server from installation media, 2026. URL <https://learn.microsoft.com/en-us/windows-server/get-started/install-windows-server>. Microsoft Learn, consulté le 27 août 2026.
- MITRE. Brute force, technique t1110, 2026. URL <https://attack.mitre.org/techniques/T1110/>. MITRE ATT&CK, consulté le 27 août 2026.
- Open Information Security Foundation. Suricata user guide : Eve json output, 2026. URL <https://docs.suricata.io/en/latest/output/eve/eve-json-output.html>. Consulté le 27 août 2026.
- OpenAI. Assistant IA : ChatGPT, 2026. URL <https://learn.chatgpt.com/docs/use-chatgpt>. Documentation officielle de ChatGPT, consultée le 27 août 2026.
- OPNsense. Initial Installation and Configuration, 2026a. URL <https://docs.opnsense.org/manual/install.html>. Documentation officielle, consultée le 27 août 2026.
- OPNsense. Firewall rules, 2026b. URL <https://docs.opnsense.org/manual/firewall.html>. Consulté le 27 août 2026.
- Mark Russinovich and Thomas Garnier. Sysmon – sysinternals, 2026. URL <https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon>. Microsoft Learn, consulté le 27 août 2026.
- Wazuh. Quickstart installation, 2026a. URL <https://documentation.wazuh.com/current/quickstart.html>. Documentation officielle, consultée le 27 août 2026.
- Wazuh. Architecture – getting started with wazuh, 2026b. URL <https://documentation.wazuh.com/current/getting-started/architecture.html>. Consulté le 27 août 2026.