SIEM Corrélation 2026 : Règles Avancées et Threat Hunting

Points essentiels

- Les règles vendor par défaut génèrent trop de faux positifs sans calibrage environnemental
- Sigma offre un format universel traduisible vers SPL, KQL et EQL
- Désactiver des règles sans documenter les gaps crée des angles morts ATT&CK
- La corrélation SIEM exige un cycle de vie continu, pas une configuration initiale

À RETENIR

À retenir

Les règles vendor par défaut génèrent trop de faux positifs sans calibrage environnemental

Sigma offre un format universel traduisible vers SPL, KQL et EQL

Désactiver des règles sans documenter les gaps crée des angles morts ATT&CK

La corrélation SIEM exige un cycle de vie continu, pas une configuration initiale

Les règles de corrélation SIEM sont le cœur battant de tout SOC : elles transforment des millions d'événements bruts en alertes actionnables, permettant à un analyste de détecter une intrusion avant qu'elle n'atteigne ses objectifs. Pourtant, dans la majorité des ETI françaises, le SIEM tourne encore avec les règles « out-of-the-box » du fournisseur, jamais calibrées sur l'environnement réel — d'où un taux de faux positifs écrasant et des attaques qui passent sous le radar. Maîtriser les *siem correlation rules threat hunting* suppose de dépasser la détection par signature pour construire une logique multi-sources, alignée sur MITRE ATT&CK et enrichie

par la threat intelligence. Ce guide détaille les techniques de corrélation avancée, la modélisation comportementale et les méthodes de chasse proactive qui font la différence entre un SIEM décoratif et une capacité de détection opérationnelle en 2026.

Sigma : le langage universel des règles de détection

Sigma est un format de règle de détection SIEM générique et open source, créé par Florian Roth (auteur de nombreux outils de sécurité) et maintenu par la communauté SigmaHQ sur GitHub. Une règle Sigma décrit une condition de détection en YAML indépendamment du SIEM cible, puis est "compilée" vers le langage spécifique du SIEM (SPL pour Splunk, KQL pour Sentinel, Lucene/EQL pour Elastic) via le compilateur pySigma ou sigmaclI. **L'intérêt fondamental de Sigma est la portabilité : une bibliothèque de règles Sigma peut être déployée sur n'importe quel SIEM**, ce qui évite le vendor lock-in et permet aux organisations de migrer leur corpus de règles sans repartir de zéro lors d'un changement de SIEM.

La bibliothèque officielle SigmaHQ sur GitHub contient plus de 3000 règles couvrant Windows, Linux, cloud, réseau et applications, toutes mappées avec les techniques MITRE ATT&CK. Pour les SOC français, c'est le point de départ incontournable : plutôt que de développer des règles from scratch, on part des règles Sigma existantes correspondant aux techniques ATT&CK

prioritaires identifiées lors des exercices Purple Team, on les compile vers le SIEM en place, et on les tune sur les données réelles de l'organisation pour éliminer les faux positifs spécifiques à l'environnement. Ce workflow Sigma → compilation → tuning est plus efficace et moins sujet aux erreurs que le développement de règles natives dans chaque SIEM.

Anatomie d'une règle Sigma efficace

Une règle Sigma bien structurée contient plusieurs sections obligatoires : le titre (name), la description, le statut (stable/test/experimental), les sources de log (logsource), la condition de détection (detection), le niveau de sévérité (level) et le mapping ATT&CK (tags). La section detection est la plus critique : elle définit les sélecteurs (filtres sur les champs des événements) et la condition logique qui les combine. Par exemple, une règle détectant le Kerberoasting via l'Event ID 4769 avec filtre RC4 s'écrira :

⚡ **Retour terrain**

J'ai accompagné la montée en maturité du SOC d'un groupe hôtelier pour qui chaque alerte SIEM déclenchait un ticket manuel. Le triage prenait 45 minutes par alerte, pour 300 alertes/jour. La mise en place d'un playbook automatisé pour les 15 types d'alertes les plus fréquents (principalement authentifications suspectes et connexions depuis IP étrangères) a ramené le temps de triage moyen à 4 minutes et libéré 70 % du temps analyste pour les incidents réels.

```
 title: Kerberoasting - Service Ticket Request with RC4
status: stable
logsource:
  product: windows
  service: security
detection:
  selection:
    EventID: 4769
    TicketEncryptionType: '0x17'
    ServiceName|endswith: '$'
  filter:
    ServiceName: 'krbtgt'
  condition: selection and not filter
falsepositives:
  - Legacy applications using RC4 encryption
level: high
tags:
  - attack.credential_access
  - attack.t1558.003
```

La section falsepositives est souvent négligée mais cruciale : elle documente les cas légitimes pouvant déclencher la règle, guidant l'analyste lors de l'investigation. Sans cette documentation, chaque faux positif nécessite une investigation complète pour être qualifié — une perte de temps considérable dans un SOC à faible effectif. Le niveau (low/medium/high/critical) doit être calibré sur l'environnement réel : une règle "high" dans un environnement sans RC4 légitime devient "medium" dans un environnement avec des serveurs NAS anciens utilisant NTLM/RC4 pour les partages réseau.

SPL Splunk : corrélations avancées et statistiques

Le SPL (Search Processing Language) de Splunk est l'un des langages SIEM les plus puissants grâce à ses capacités de traitement statistique en temps réel. Pour la détection de menaces, plusieurs commandes SPL sont particulièrement utiles. La commande `stats` permet de compter les événements par dimension pour détecter des anomalies volumétriques : `index=windows EventCode=4625 | stats count by src_ip, user | where count > 10` détecte les brute-force sur les comptes Windows. La commande `transaction` regroupe des événements liés dans le temps pour reconstituer des séquences : `| transaction user maxspan=10m` permet de corréler une tentative de dump LSASS (Event 10 Sysmon) suivie d'une connexion réseau atypique dans les 10 minutes suivantes.

Les **corrélations multi-sources en SPL** sont la véritable valeur ajoutée du SIEM par rapport aux alertes individuelles des outils de sécurité. Par exemple, la détection d'un lateral movement combinant une connexion RDP (Event 4624 type 10) vers une machine qui n'avait jamais reçu de connexion RDP depuis la source IP, suivie dans les 5 minutes d'un accès à un partage réseau sensible (Event 5140) depuis la même machine destination, est impossible à détecter avec des règles individuelles mais triviale à exprimer en SPL avec un join sur la dimension machine. Ces corrélations multi-hop correspondent aux Detections-as-Code modernes qui produisent des alertes à haute fidélité résistantes aux évasions individuelles.

KQL Microsoft Sentinel : threat hunting et analytics

Le *KQL* (Kusto Query Language) est le langage d'interrogation de Microsoft Sentinel, Azure Log Analytics et Microsoft 365 Defender. Sa syntaxe fonctionnelle, inspirée de LINQ et SQL, est particulièrement adaptée à l'analyse de grandes volumétries d'événements cloud. La commande `join` de KQL permet de corréler des tables différentes — par exemple, croiser les SigninLogs Entra ID avec les AuditLogs pour détecter une application qui s'authentifie puis modifie des rôles immédiatement après — avec une performance optimisée pour des volumes de téraoctets via le moteur Kusto distribué d'Azure.

Les Analytic Rules de Sentinel utilisent KQL et s'exécutent selon une fréquence configurable (5 minutes à 1 heure). Les règles de Fusion de Sentinel combinent automatiquement plusieurs signaux KQL de faible fidélité en une alerte composite de haute fidélité via un modèle ML propriétaire — par exemple, une règle de Fusion peut combiner une authentification depuis un pays inhabituel, une modification de MFA, et une tentative d'accès à un site de partage de fichiers externe pour créer une alerte "Account Compromise with Data Exfiltration" qu'aucune règle individuelle n'aurait générée. **Cette capacité de corrélation ML est l'un des différenciateurs majeurs de Sentinel par rapport aux SIEM traditionnels** pour les ETI françaises Microsoft-centriques qui souhaitent une détection avancée sans nécessiter une équipe de data scientists.

EQL Elastic Security : détection comportementale séquentielle

L'*EQL* (Event Query Language) est le langage de détection d'Elastic Security, spécialement conçu pour détecter des séquences d'événements corrélés dans le temps — sa principale différence avec SPL et KQL est la notion native de "sequence" qui permet d'exprimer des chaînes d'événements ordonnées avec des fenêtres temporelles et des conditions de liaison entre événements. Une règle EQL typique pour détecter un process injection via process hollowing s'écrit :

```
sequence by host.id with maxspan=1m
  [process where event.type == "start" and process.name == "svchost.exe"
   and process.parent.name != "services.exe"]
  [process where event.type == "start" and
   process.parent.name == "svchost.exe" and
   not process.name in ("WmiPrvSE.exe", "dllhost.exe")]
```

Cette séquence détecte un svchost.exe démarré par un parent non-standard (anomalie), suivi d'un process enfant inattendu dans la minute — un pattern caractéristique du process hollowing. **EQL est particulièrement efficace pour les détections de type "kill chain" où plusieurs étapes consécutives d'une attaque doivent être observées avant de déclencher une alerte**, réduisant drastiquement les faux positifs par rapport aux règles déclenchées sur chaque événement isolé. La bibliothèque de règles EQL d'Elastic Security contient plus de 600 règles pré-construites, maintenues en open source sur GitHub et mappées avec ATT&CK.

Threat Hunting proactif : chercher sans alerte préalable

Le threat hunting est la pratique de recherche proactive de traces d'attaquants dans les données SIEM, sans attendre qu'une alerte soit déclenchée. Contrairement à la détection réactive basée sur des règles, le threat hunting part d'une hypothèse (hypothesis-driven hunting) : "Un attaquant utilisant les TTP d'APT28 aurait probablement utilisé PowerShell avec des encodages base64 depuis un compte non-admin — cherchons ces patterns dans les 30 derniers jours." Cette approche détecte les attaquants qui ont contourné les règles existantes ou qui se cachent dans le bruit de fond sous le seuil d'alerte.

Les chasseurs de menaces (threat hunters) utilisent des techniques de pivot — commencer par un indicateur connu (une IP C2 identifiée dans un rapport CTI) puis explorer les connexions et comportements associés pour découvrir l'étendue de la compromission. En SPL, une chasse typique post-alert commence par `index=network dest_ip=185.220.x.x` pour identifier toutes les machines ayant contacté l'IP C2, puis pivote sur les process à l'origine de ces connexions, puis sur les comptes utilisateurs associés, en construisant progressivement une carte de la compromission. **Cette démarche investigative, distincte de l'alerting automatique, nécessite des analystes de niveau 2-3 avec une expertise des TTP adversariales** — une ressource rare dont la formation et la rétention sont un défi pour les ETI françaises concurrencées par les ESN et les GAFAM sur le marché des talents cybersécurité.

Gestion des faux positifs : l'art du tuning SIEM

Les faux positifs sont la principale raison pour laquelle les règles SIEM sont désactivées, dégradées ou ignorées par les analystes — créant les blind spots non documentés qui permettent aux attaquants de passer inaperçus. Le tuning efficace des règles SIEM repose sur trois approches complémentaires. Premièrement, l'ajout d'exclusions basées sur le contexte : exclure les IP des scanners de vulnérabilités internes des règles de détection de scanning réseau, ou exclure les comptes de service des règles de détection de comportements de connexion atypiques. Deuxièmement, l'ajout de conditions contextuelles enrichissant le signal : plutôt qu'alerter sur tout Event ID 4688 (process creation), filtrer sur les processus créés depuis un parent inhabituel ET dont le nom correspond à une liste de LOLBins (Living Off the Land Binaries).

Troisièmement, la mise en place de listes blanches dynamiques maintenues dans des tables de référence (lookups Splunk, watchlists Sentinel) qui sont mises à jour par les équipes IT au fil des changements d'infrastructure — serveurs de monitoring, IPs de jump box, comptes de service nouveaux. **Un processus formel d'enregistrement des changements IT avec mise à jour automatique des watchlists SIEM est l'une des intégrations CMDB/SIEM les plus rentables en termes de réduction des faux positifs**, car la majorité des faux positifs sont liés à des changements d'infrastructure non communiqués aux équipes SOC — un nouveau serveur de

monitoring qui commence à scanner le réseau, un nouveau compte de service qui s'authentifie depuis des IPs multiples, une nouvelle application qui génère des logs d'accès inhabituels.

Règles de détection ransomware : le cas d'usage prioritaire

La détection du ransomware via les règles SIEM doit couvrir plusieurs phases de l'attaque, pas seulement le chiffrement final. La phase de reconnaissance (SMB scanning, BloodHound), la phase de credential dumping (LSASS access, SAM dump via reg save), la phase de désactivation des sauvegardes (vssadmin delete shadows, wbadmin delete catalog), et la phase de déploiement latéral (PsExec, WMIC remote) sont toutes détectables dans les logs Windows et Sysmon avant que le chiffrement ne commence — et c'est dans ces phases précoces que la détection peut encore permettre une intervention avant l'impact. **La règle de détection "vssadmin delete shadows" (Event 4688 avec CommandLine contenant "delete shadows") est l'une des règles les plus critiques à avoir dans tout SIEM**, car la suppression des VSS (Volume Shadow Copies) est une signature quasi-universelle des ransomwares modernes qui l'effectuent systématiquement avant le chiffrement pour éliminer les possibilités de restauration rapide.

Pour les ETI françaises particulièrement exposées aux ransomwares (secteur industrie, santé, BTP), la mise en place d'un "Ransomware Detection Playbook" documentant les règles SIEM par phase d'attaque, les seuils d'alerte, et les procédures de réponse automatisée (isolation réseau via SOAR, blocage du compte via API AD) est une priorité opérationnelle. Ce playbook doit être testé régulièrement via des exercices Purple Team simulant une chaîne d'attaque ransomware complète pour valider que la détection fonctionne et que les équipes SOC connaissent les procédures de réponse — des tests que très peu d'organisations réalisent en pratique avant d'en avoir besoin face à un vrai incident.

Règles de détection Active Directory : les incontournables

Active Directory est la cible principale des attaquants dans les réseaux Windows d'entreprise, et les règles de détection AD doivent couvrir les techniques les plus courantes observées dans les compromissions réelles. Voici les règles prioritaires, toutes disponibles dans la bibliothèque Sigma :

Kerberoasting (T1558.003) : Event ID 4769 avec TicketEncryptionType = 0x17 (RC4) et ServiceName ne se terminant pas par "$" — alerte si plus de 3 demandes RC4 depuis le même compte en 5 minutes. **AS-REP Roasting** (T1558.004) : Event ID 4768 avec PreAuthType = 0 — tout utilisateur AD avec l'option "Do not require Kerberos preauthentication" est une cible de choix. **DCSync** (T1003.006) : Event ID 4662 avec Properties contenant les GUID de réplication DS-Replication-Get-Changes et DS-Replication-Get-Changes-All depuis un compte qui n'est pas un contrôleur de domaine. **AdminSDHolder abuse** : Event ID 4738 (User Account Changed) avec des modifications sur les groupes protégés par SDHolder. Ces quatre règles, correctement calibrées sur l'environnement, constituent le minimum vital pour la détection des compromissions AD les plus courantes en 2026.

Règles de détection cloud : AWS CloudTrail essentiels

Sur AWS, les règles de détection SIEM reposent sur les événements CloudTrail qui enregistrent toutes les appels API. Les règles prioritaires pour la détection d'escalade IAM et de compromission AWS couvrent : `iam:CreateAccessKey` sur un compte différent de l'appelant (création d'accès pour un autre utilisateur — technique de persistance), `iam:AttachRolePolicy` avec des policies à large périmètre (AdministratorAccess, PowerUserAccess), `sts:AssumeRole` depuis une IP hors des plages connues ou vers des rôles cross-account non habituels, et `ec2:CreateSecurityGroup` suivi immédiatement d'une règle autorisant tout le trafic entrant (0.0.0.0/0). Ces événements, filtrés sur des conditions contextuelles précises, produisent des alertes à haute fidélité avec peu de faux positifs.

La configuration de CloudTrail elle-même est une règle de détection méta-critique : toute désactivation ou modification de CloudTrail (`cloudtrail:StopLogging`, `cloudtrail:DeleteTrail`) doit déclencher une alerte immédiate de sévérité critique, car un attaquant qui a atteint un niveau de privilège suffisant pour modifier CloudTrail cherche à couvrir ses traces avant une action destructrice ou d'exfiltration massive. **AWS Security Hub centralise automatiquement ces alertes CloudTrail dans un dashboard unifié** et peut les transmettre vers les SIEM via EventBridge → Lambda → SIEM API, une architecture d'ingestion standard recommandée par AWS pour les organisations utilisant Splunk ou Elastic en plus de Security Hub.

SOAR : automatiser la réponse aux alertes SIEM

Le *SOAR* (Security Orchestration, Automation and Response) est la couche d'automatisation qui permet de transformer une alerte SIEM en action de réponse sans intervention humaine pour les cas les plus simples. Les plateformes SOAR leaders (Palo Alto XSOAR, Splunk SOAR, Microsoft Sentinel Playbooks via Logic Apps) permettent de créer des playbooks de réponse automatisée : sur une alerte de compte bloqué par brute-force, le playbook envoie automatiquement une notification Slack à l'analyste, interroge l'AD pour récupérer les informations du compte, vérifie si l'IP source est dans les listes de réputation, et si l'IP est malveillante, bloque automatiquement le compte AD et crée un ticket Jira pour investigation — le tout en moins de 30 secondes, sans intervention humaine.

Pour les ETI françaises avec des SOC à faible effectif (2-5 analystes), le SOAR est particulièrement précieux car il multiplie la capacité de traitement des alertes sans recruter. **Microsoft Sentinel Playbooks (basés sur Azure Logic Apps) sont inclus dans la licence Sentinel et permettent d'automatiser les réponses les plus courantes sans coût SOAR additionnel**, ce qui en fait la solution de démarrage SOAR la plus accessible pour les ETI Microsoft-centriques. L'enjeu est d'éviter le "spaghetti SOAR" — une accumulation de playbooks mal documentés et non maintenus qui deviennent un problème de sécurité eux-mêmes — en appliquant les mêmes principes de DevSecOps (versionning Git, tests, documentation) aux playbooks SOAR qu'aux règles de détection.

Détection du mouvement latéral : corrélations multi-machine

Le mouvement latéral est la phase de l'attaque la plus difficile à détecter car les attaquants utilisent des credentials légitimes et des protocoles natifs. Les règles de détection efficaces reposent sur des corrélations multi-sources impossibles à exprimer dans un seul outil de sécurité périmétrique. Exemple de corrélation multi-machine pour détecter le mouvement latéral via SMB : une machine A se connecte en SMB (Event ID 5140) à une machine B qu'elle n'avait jamais contactée auparavant (sur les 30 derniers jours), depuis un compte utilisateur standard (non-admin) — ce pattern, rare dans un réseau sain, est caractéristique d'un attaquant se déplaçant avec un compte utilisateur compromis.

La mise en place de cette règle nécessite une baseline comportementale des connexions SMB habituelles (calculée sur 30 jours de données historiques) et une liste dynamique des paires machine-à-machine jamais observées auparavant. **Splunk Enterprise Security intègre le concept de "Asset and Identity Framework" qui maintient automatiquement ces baselines et permet de détecter les comportements "first seen"** — un pattern utilisateur/machine se connectant pour la première fois vers une destination — sans nécessiter de règle manuelle pour chaque paire. Cette capacité de détection basée sur l'anomalie comportementale, complémentaire aux règles de détection basées sur les signatures, constitue la detection-in-depth qui distingue un SOC mature d'un SOC basique.

Métriques d'efficacité des règles SIEM

Toutes les règles SIEM ne se valent pas : certaines génèrent 1000 alertes par jour avec 99% de faux positifs, d'autres génèrent 2 alertes par an toutes vraies. Mesurer l'efficacité des règles de détection est indispensable pour prioriser les efforts de tuning et démontrer la valeur du SOC. Les métriques clés à suivre par règle sont : le volume d'alertes (alertes/semaine), le taux de vrais positifs (% d'alertes confirmées comme malveillantes), le MTTD moyen (délai entre l'événement et l'alerte), et le ratio signal/bruit (vrais positifs / total alertes). Une règle avec moins de 20% de

vrais positifs doit être retuée ou supprimée ; une règle avec 0 alerte en 6 mois doit être vérifiée (la source de log est-elle bien ingérée ? la condition de détection est-elle correcte ?).

La mise en place d'un dashboard SIEM de "Rule Health" affichant ces métriques en temps réel pour l'ensemble du catalogue de règles est une bonne pratique de gestion du SOC encore peu répandue dans les ETI françaises. Ce dashboard permet d'identifier rapidement les règles "zombie" (jamais déclenchées), les règles "noise" (trop de faux positifs), et les règles "golden" (haute fidélité, MTTD rapide) qui méritent d'être protégées lors des changements de SIEM ou de source de logs. **Une organisation qui maintient un catalogue de règles SIEM documenté et métriqué a une visibilité sur sa propre couverture de détection que 90% des ETI françaises n'ont pas encore en 2026**, et cette visibilité est une prérequis pour s'améliorer de manière continue.

Questions fréquentes sur les règles de corrélation SIEM

Peut-on utiliser les règles Sigma directement sans les compiler ?

Non — les règles Sigma sont un format source qui doit être compilé vers le langage natif du SIEM cible (SPL, KQL, EQL, Lucene) avant d'être déployées. L'outil pySigma (anciennement sigma-cli) effectue cette compilation avec des "backends" spécifiques à chaque SIEM. Certains SIEMs comme Microsoft Sentinel et Elastic Security proposent des convertisseurs natifs Sigma intégrés dans leur interface, simplifiant le déploiement pour les équipes qui ne souhaitent pas gérer la pipeline de compilation.

Combien de règles de détection actives faut-il dans un SIEM ?

La quantité n'est pas le critère pertinent — la qualité et la couverture ATT&CK sont. Un SIEM avec 50 règles bien tunées et documentées, couvrant les 25 techniques ATT&CK les plus critiques pour le secteur, est plus efficace qu'un SIEM avec 2000 règles "out-of-the-box" non validées et générant 500 faux positifs par jour. La recommendation est de partir avec les 50-100 règles Sigma de niveau "stable" les plus pertinentes pour l'environnement, de les valider via

exercice Purple Team, puis d'en ajouter de nouvelles progressivement avec validation systématique.

Quelles sources de logs prioriser pour alimenter le SIEM ?

Par ordre de priorité : 1) Windows Security Event Logs avec Sysmon (couverture AD et endpoints Windows), 2) logs réseau/proxy/DNS (détection des exfiltrations et C2), 3) logs EDR (comportements process et fichiers), 4) logs cloud (CloudTrail AWS, Activity Log Azure pour les environnements hybrides), 5) logs applicatifs des applications critiques (ERP, VPN, pare-feu). L'activation de Sysmon avec la configuration SwiftOnSecurity est recommandée comme priorité absolue pour tout Windows SIEM, car elle multiplie par 10 la visibilité sur les comportements process et réseau sans coût additionnel de licence.

Tableau comparatif des langages SIEM 2026

Critère	Sigma	SPL (Splunk)	KQL (Sentinel)	EQL (Elastic)
Portabilité	Maximale (source)	Splunk uniquement	Azure/Microsoft uniquement	Elastic uniquement
Courbe apprentissage	Faible (YAML)	Moyenne	Moyenne	Moyenne
Corrélations temporelles	Limitée	Transaction/Join	Join/MV-Expand	Sequence native
Bibliothèque communautaire	3000+ règles GitHub	Splunk Security Essentials	Sentinel GitHub 1000+	600+ règles intégrées
Mapping ATT&CK	Natif (tags YAML)	Via Security Essentials	Natif (metadata)	Natif (tags)
Coût licence	Gratuit (open source)	Élevé (data volume)	Moyen (par GB/jour)	Moyen (nœuds)

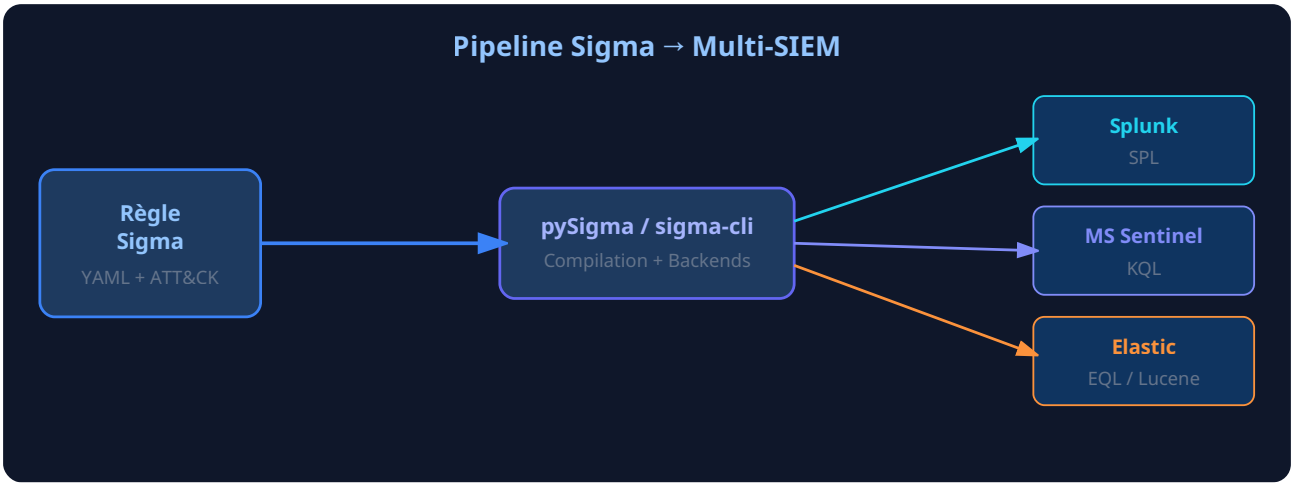
Opinion : les SIEM "magic quadrant" ne font pas les SOC efficaces

L'industrie de la sécurité française souffre d'une confusion persistante entre acheter un SIEM performant et avoir une capacité de détection efficace. Des dizaines d'ETI ont dépensé 200 000 à 500 000 euros en licences Splunk Enterprise Security ou Microsoft Sentinel, pour déployer au final les mêmes règles de détection génériques qu'elles auraient pu utiliser avec un ELK Stack à 0 euro de licence, avec les mêmes taux de faux positifs et les mêmes blind spots. Le SIEM n'est qu'un moteur — c'est la qualité des règles de corrélation, la couverture des sources de logs, et les compétences des analystes qui déterminent l'efficacité de la détection. Investir 50 000 euros dans la formation Sigma et threat hunting d'une équipe SOC produit généralement plus de valeur de détection qu'investir 200 000 euros supplémentaires dans des licences SIEM premium. Ce n'est pas une opinion populaire chez les vendors, mais c'est ce que montrent les métriques MTTD des organisations qui ont investi dans leurs équipes plutôt que dans leurs outils.

À RETENIR

Points clés à retenir

- Sigma est le format universel à adopter pour écrire des règles portables indépendantes du SIEM
- SPL, KQL et EQL offrent chacun des capacités de corrélation avancées — choisir selon l'écosystème existant
- Le tuning des règles (exclusions contextuelles, watchlists dynamiques) est aussi important que leur création
- La détection du ransomware doit couvrir toutes les phases pré-chiffrement, pas uniquement l'impact final
- Un dashboard "Rule Health" avec métriques de fidélité par règle est indispensable pour gérer un catalogue SIEM
- SOAR + SIEM multiplie la capacité de réponse des équipes SOC réduites des ETI françaises



Ressources et liens pour développer ses règles SIEM

Pour les équipes souhaitant approfondir le développement de règles de corrélation, plusieurs ressources de référence sont indispensables. La bibliothèque officielle SigmaHQ sur GitHub est le point de départ pour tout nouveau projet de règles — les 3000+ règles stables représentent des années d'expertise distillées. La documentation MITRE ATT&CK Enterprise est la référence pour mapper les techniques aux sources de données et aux detections. En France, les capacités ML dans les SIEM modernes complètent les règles statiques par des détections comportementales. L'article threat hunting avec Microsoft 365 et Sentinel approfondit les cas d'usage KQL spécifiques à l'environnement Microsoft. Les connexions entre SIEM et exercices Purple Team et règles de corrélation forment le cycle vertueux de la détection continue.

Règles de détection DNS dans le SIEM

La surveillance DNS via le SIEM est un vecteur de détection crucial souvent négligé par les équipes SOC qui se concentrent sur les logs Windows et les pare-feux périmètriques. Les règles de détection DNS couvrent trois catégories principales d'anomalies. Premièrement, le DNS tunneling : des requêtes vers des sous-domaines de longueur anormale (plus de 50 caractères) ou

d'entropie élevée (calculée sur le label avant le TLD), des fréquences de requêtes vers un même domaine anormalement élevées (plus de 100 requêtes par minute), ou des types d'enregistrements inhabituels (TXT, NULL, PRIVATE) utilisés pour l'exfiltration. Deuxièmement, les requêtes vers des DGA (Domain Generation Algorithm) : des domaines à haute entropie dont aucun historique de résolution n'existe dans les logs, souvent reconnaissables par un ratio consonnes/voyelles aberrant ou une suite de caractères statistiquement improbable dans une langue naturelle.

Troisièmement, les Fast Flux et Domain Shadowing : des domaines dont l'IP associée change plusieurs fois par heure (Fast Flux — technique des botnets pour masquer l'infrastructure C2), ou des sous-domaines d'un domaine légitime compromis utilisés comme points de rendez-vous C2 (Domain Shadowing, observé dans des campagnes APT ciblant des organisations françaises). **L'intégration des logs DNS dans le SIEM (via les logs du résolveur DNS interne, ou via des captures réseau Zeek/NetworkMiner parsées en JSON) est une prérequis pour ces règles** — et c'est un prérequis encore absent dans 40% des ETI françaises qui n'ont pas de visibilité sur leur flux DNS interne dans leur SIEM. L'article DNS tunneling et exfiltration de cette série détaille les techniques de détection DNS avancées.

Règles de détection d'exfiltration de données

L'exfiltration de données est l'objectif final de la majorité des attaques APT et est paradoxalement l'une des phases les moins détectées dans les SOC français, car les canaux d'exfiltration sont souvent des protocoles légitimes (HTTPS vers des services cloud, DNS, SMTP) que les règles de détection classiques ne challengent pas. Les règles d'exfiltration efficaces reposent sur la détection d'anomalies volumétriques par rapport à la baseline comportementale de chaque utilisateur ou machine : un poste de travail qui télécharge 2 Go vers OneDrive en une heure alors que sa moyenne habituelle est de 100 Mo/jour est un signal fort d'exfiltration, même si le protocole HTTPS vers Microsoft 365 est totalement légitime.

La mise en place de ces règles d'anomalie volumétrique nécessite deux composantes techniques : premièrement, l'ingestion des logs proxy/DLP avec les volumes de données par session et par destination, et deuxièmement, un calcul de baseline comportementale par utilisateur et par destination. Splunk ML Toolkit et Elastic Machine Learning (inclus dans la licence Platinum)

offrent des modèles de dérive temporelle (time series forecasting) qui calculent automatiquement ces baselines et alertent sur les déviations significatives. **Sans ces logs proxy/DLP dans le SIEM, la détection d'exfiltration reste aveugle aux transferts de données massifs via des canaux légitimes** — une lacune critique pour les organisations manipulant des données sensibles soumises au RGPD, dont la notification CNIL sous 72h en cas de violation est une obligation légale.

Détection des menaces internes : l'angle mort des SIEM

Les menaces internes — employés malveillants, sous-traitants négligents, comptes compromis utilisés par des attaquants — sont les plus difficiles à détecter car les actions effectuées utilisent des accès légitimes qui ne déclenchent pas les règles basées sur des comportements anormaux de type "attaque externe". La détection des insider threats dans un SIEM repose sur des techniques d'analyse comportementale plutôt que sur des règles de signature : UEBA (User and Entity Behavior Analytics), qui établit des profils comportementaux normaux pour chaque utilisateur et alerte sur les déviations — accès à des ressources jamais consultées auparavant, volumes de téléchargement inhabituels, connexions en dehors des heures normales de travail, accès depuis une localisation géographique inhabituelle.

Les solutions UEBA intégrées dans les SIEM modernes (Splunk UBA, Microsoft Sentinel UEBA, Exabeam) utilisent des modèles ML non supervisés pour détecter ces anomalies comportementales sans nécessiter de règles manuelles pour chaque cas d'usage. **La limitation principale de l'UEBA est le besoin d'une période d'apprentissage de 30 à 90 jours** avant que les baselines comportementales soient suffisamment représentatives pour générer des alertes fiables — une période pendant laquelle l'organisation reste aveugle aux comportements anormaux initiaux. Pour les ETI françaises, la mise en place d'UEBA doit être planifiée suffisamment en avance par rapport aux periodes à risque (fusions-acquisitions, restructurations, licenciements) qui augmentent statistiquement le risque de menace interne.

DevSecOps pour les règles SIEM : versionner et tester ses détections

Les règles de détection SIEM sont du code — et comme tout code, elles doivent être versionnées, testées et revues avant déploiement en production. L'approche *Detection-as-Code* applique les principes DevSecOps au développement de règles SIEM : les règles sont stockées dans Git (avec historique complet des modifications et auteurs), des tests unitaires valident que chaque règle détecte les événements malveillants simulés et n'alerte pas sur les événements légitimes connus, et un pipeline CI/CD compile les règles Sigma et les déploie automatiquement dans le SIEM de staging pour validation avant promotion en production. Cette discipline, encore rare dans les SOC français en 2026, élimine les régressions courantes (une modification de règle qui casse la détection existante, ou qui crée une avalanche de faux positifs) et crée un audit trail des changements de règles précieux lors des investigations post-incident.

Des outils comme `detection-rules` d'Elastic et `sigma-tests` de la communauté SigmaHQ permettent d'écrire des tests unitaires de règles en Python, validant que la règle détecte bien un événement malveillant simulé (test "true positive") et n'alerte pas sur un événement légitime similaire (test "false positive"). L'intégration de ces tests dans GitHub Actions ou GitLab CI crée un pipeline Detection-as-Code complet qui réduit le temps de déploiement d'une nouvelle règle de plusieurs semaines (review manuelle, déploiement SIEM manual) à quelques heures (review pair, tests automatiques, déploiement automatique). **Les équipes SOC qui adoptent Detection-as-Code reportent une réduction de 70% des régressions de règles et une multiplication par trois de leur vélocité de déploiement de nouvelles détections** — un gain d'efficacité particulièrement précieux face à l'évolution rapide des techniques adversariales.

Optimiser l'ingestion des logs Windows : Sysmon et audit policy

La qualité des règles de détection SIEM dépend directement de la richesse des logs disponibles — et les logs Windows par défaut sont largement insuffisants pour la détection des menaces avancées. Deux configurations sont indispensables pour maximiser la visibilité sur les endpoints Windows. Premièrement, l'activation d'une audit policy avancée (Windows Advanced Audit

Policy) couvrant les catégories Account Logon, Account Management, Detailed Tracking, Logon/Logoff, Object Access (pour les partages réseau et les fichiers sensibles), Policy Change et Privilege Use. Ces catégories d'audit, désactivées par défaut, sont configurables via GPO (Computer Configuration → Windows Settings → Security Settings → Advanced Audit Policy Configuration) et génèrent les Event IDs critiques pour la détection des attaques AD et des mouvements latéraux.

Deuxièmement, le déploiement de Sysmon (System Monitor de Sysinternals) avec une configuration robuste comme SwiftOnSecurity ou Olaf Hartong's Modular Sysmon Config ajoute des dizaines d'Event IDs supplémentaires couvrant la création de processus avec hashes (Event ID 1), les connexions réseau par process (Event ID 3), les chargements de drivers (Event ID 6), les accès au process LSASS (Event ID 10), la création de fichiers exécutables (Event ID 11), les modifications de valeurs de registre sensibles (Event ID 13), et les connexions WMI (Event ID 19-21). **Avec Sysmon déployé et une audit policy avancée activée, la couverture de détection ATT&CK pour les endpoints Windows passe de 15% à plus de 60%** — une transformation de la visibilité SOC sans coût de licence additionnel, uniquement avec de la configuration. Cette combinaison est désormais le prérequis minimum recommandé par l'ANSSI pour tout SOC prétendant à une détection des menaces avancées sur les endpoints Windows.

Politique de rétention des logs : équilibrer détection et coût

La rétention des logs dans le SIEM est un sujet directement lié à la capacité de détection et à la conformité réglementaire, mais souvent abordé uniquement sous l'angle du coût. Les recommandations de l'ANSSI et de NIS 2 convergent vers une rétention minimale de 12 mois pour les logs de sécurité des organisations critiques — ce qui permet de détecter des attaques "low-and-slow" qui s'étendent sur plusieurs mois, d'avoir suffisamment de données historiques pour établir des baselines comportementales significatives, et de satisfaire les exigences forensiques en cas d'incident nécessitant une investigation rétrospective. La réalité dans les ETI françaises est souvent une rétention de 30 à 90 jours imposée par les contraintes de stockage SIEM, insuffisante pour ces cas d'usage.

La solution architecturale standard pour concilier rétention longue et coût raisonnable est le tiering : les logs "chauds" (30 derniers jours) restent dans l'index SIEM principal pour des recherches rapides, les logs "tièdes" (31-180 jours) sont archivés dans un stockage moins cher (S3, Azure Blob) avec possibilité d'interrogation différée via Frozen Tier (Elastic) ou Archive Tier (Sentinel), et les logs "froids" (180 jours - 7 ans) sont compressés et archivés pour conformité uniquement. **Ce tiering trois-niveaux réduit le coût de rétention de 60 à 70% par rapport à une rétention uniforme dans l'index principal**, tout en maintenant la capacité de forensique rétrospective sur 12 mois et la conformité légale sur 7 ans exigée par certains référentiels sectoriels (santé, finance). La mise en place de cette architecture doit être planifiée dès le déploiement initial du SIEM, car une migration de données existantes vers un schéma tiered est coûteuse et complexe après coup.

Retour terrain : la règle désactivée qui aurait tout changé

Lors d'une réponse à incident dans une PME industrielle bretonne en mars 2025, l'équipe d'investigation a découvert que l'attaquant (un groupe de ransomware affilié à BlackBasta) avait opéré dans le réseau pendant 23 jours avant le chiffrement. En analysant les logs SIEM rétrospectivement, la règle de détection "vssadmin delete shadows" — qui aurait alerté sur l'action de suppression des VSS effectuée au jour 22 — avait été désactivée six mois auparavant par un analyste SOC qui la jugeait trop bruyante à cause de faux positifs générés par un logiciel de backup utilisant vssadmin de manière légitime. La correction du faux positif aurait pris 30 minutes (ajouter une exclusion sur le process parent du logiciel de backup) ; la non-correction a coûté à l'entreprise 340 000 euros de rançon plus un arrêt de production de 4 jours. Cette histoire illustre que désactiver une règle sans l'avoir préalablement tunée est systématiquement une mauvaise décision — quelles que soient les contraintes opérationnelles du moment.

La maturité d'un SOC se mesure autant à sa capacité à créer de nouvelles règles de détection qu'à maintenir la qualité opérationnelle des règles existantes : un catalogue de détection vivant, versionné et régulièrement challengé par des exercices Purple Team est l'actif de sécurité le plus précieux qu'une organisation puisse constituer, bien au-delà des licences logicielles et des équipements réseau dont la valeur se déprécie rapidement face à l'évolution constante des

techniques adversariales ciblant les entreprises françaises. Les organisations qui traitent leurs règles SIEM comme un produit logiciel — avec un owner, un backlog de fonctionnalités, des tests automatisés et des métriques de performance — atteignent des niveaux de détection systématiquement supérieurs aux organisations qui les considèrent comme une configuration statique initiale.

Intégrer les feeds CTI dans le SIEM pour enrichir les alertes

L'enrichissement des alertes SIEM avec des données de Cyber Threat Intelligence (CTI) transforme une alerte générique "connexion vers une IP externe" en une alerte contextualisée "connexion vers une IP C2 connue du groupe APT28, associée à la campagne Fancy Bear Q1 2026 ciblant le secteur aérospatial européen". Cet enrichissement contextuel réduit drastiquement le temps d'investigation des alertes et améliore la priorisation : une connexion vers une IP classée "high confidence malicious" par MISP ou OpenCTI est traitée immédiatement, tandis qu'une connexion vers une IP inconnue peut attendre.

L'intégration technique des feeds CTI dans le SIEM se fait via des lookups enrichis : pour Splunk, des CSV tables ou KV Stores mis à jour automatiquement par des scripts Python récupérant les IoC depuis les APIs TAXII/STIX (MISP, OpenCTI, Sekoia.io, CERT-FR) ; pour Sentinel, les Threat Intelligence feeds intégrés dans l'interface et les ThreatIntelligenceIndicator tables interrogeables en KQL ; pour Elastic, le module Threat Intelligence qui ingère automatiquement les feeds CTI dans des index dédiés. **L'alimentation automatique des IoC CERT-FR dans le SIEM via TAXII est une intégration disponible gratuitement pour toutes les organisations françaises**, permettant de détecter en temps quasi-réel les menaces actives identifiées par le CERT-FR comme ciblant des organisations françaises — un enrichissement particulièrement pertinent pour les ETI soumises à NIS 2 qui doivent démontrer une veille active sur les menaces sectorielles.

Conclusion

Ce sujet s'inscrit dans un contexte de menaces en constante évolution. La meilleure protection combine veille active, audits réguliers et sécurité by design. Pour approfondir ou évaluer votre exposition, consultez nos experts.

Vous souhaitez en savoir plus ou évaluer votre exposition ?

Contacter nos experts ou contactez-nous directement.

© Ayi NEDJIMI Consultants — Tous droits réservés