République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Saad Dahleb de Blida

Faculté des Sciences
Département d'Informatique

Mémoire de fin d'étude
Pour l'obtention du diplôme de Master en Informatique
Option : Sécurité des systèmes d'information

Thème

Centralisation et gestion des fichiers logs (SIEM)

Réalisé par :
DEKHLI Asma **et**
HADJ SADOK Sonia

Promotrice : Mme OUKID.S
Encadreur : Mr BELLAL Arezki .M
Présidente de jury : Mme CHIKHI.I
Examinatrice : Mme HADJ HENNI.M
L'organisme d'accueil : Ooredoo

2018/2019

Dédicaces

Avec l'expression de ma reconnaissance, je dédie ce mémoire :

Aux prunelles de mes yeux : mes chers parents. Aucun hommage ne pourrait être à la hauteur de l'amour dont ils ne cessent de me donner.

Quoique je dise ou quoique je fasse je ne saurai guère les remercier comme il se doit. Leurs présences et soutiens ont toujours été ma source de force et de réussite.

Que dieux les protège et que la réussite soit toujours à ma portée pour que je puisse les combler de bonheur.

A mes deux et uniques sœurs que j'aime le plus au monde : 'Fatma zohra' et 'Khadidja' qui ont été toujours là à mes côtés, et qui n'ont jamais cessé de m'encourager.

A tous les membres de ma famille paternelle 'DEKHLI' et maternelle 'TIFOURA'.

A ma chère binôme 'Sonia' ainsi qu'à sa famille 'HADJ SADOK' et à tous mes amis et collègues, plus particulièrement 'Sylia' et 'Manel'.

A tous ceux qui ont contribué de près ou de loin pour que ce projet soit possible, je vous dis merci.

Asma.

Je dédie ce modèste travail à :

Mon très cher papa HADJ SADOK ELHADI : Autant de phrases et d'expressions aussi longues soient-elles ne sauraient exprimer ma gratitude et ma reconnaissance.

Ta patience sans fin, ta compréhension et ton encouragement sont pour moi le soutien indispensable que tu n'as cessé d'apporter.

Je te dois ce que je suis aujourd'hui et ce que je serai demain.

Que dieu le tout puissant te préserve, t'accorde santé et bonheur et te protège de tout mal.

Ma très chère maman HADJ SADOK Rabea : quoi que je dise je ne pourrai exprimer ce que mon cœur porte pour toi.

Tu as su m'inculquer le sens de responsabilité, de l'optimisme et de la confiance en soi face aux difficultés de la vie. Tes conseils ont toujours guidé mes pas vers la réussite.

Aujourd'hui cette réussite, aussi petite qu'elle soit, reflète l'image de la meilleure maman que tu es.

Merci pour tes prières qui illuminent ma vie.

Que dieu le tout puissant te préserve, t'accorde santé et bonheur et te protège de tout mal.

A ma petite sœur Latifa, à mon frère Badreddine et à ma belle-sœur Nesrine : Merci pour votre présence et votre soutien.

A mon amie et binôme DEKHLI Asma qui a partagé avec moi chaque moment de ce parcours.

Aux meilleures Amies au monde : Meriem et Khawla, je n'oublierai jamais votre soutien et votre croyance en moi. Merci de faire partie de ma vie.

A mes tantes et cousines Amina et Ihcen : Merci pour vos encouragements.

Sonia.

Remerciements

En guise de reconnaissance, nous tenons à témoigner nos sincères remerciements à toutes les personnes qui ont contribués de près ou de loin au bon déroulement de notre stage de fin d'étude et à l'élaboration de ce modeste travail.

Nos sincères gratitudes à Mrs. OUKID.S pour la qualité de son enseignement, ses conseils et son intérêt incontestable qu'elle porte à tous les étudiants.

Nous sommes profondément reconnaissantes envers notre encadreur M. BELLAL Arezki.M pour nous avoir fait le grand honneur de diriger notre projet dans les meilleures conditions, ainsi que pour son encadrement, ses conseils et sa disponibilité.

Nous tenons à remercier aussi M. BOUDEGNA Zineddine et M. ABADA Amine Pour leur bienveillance et leurs conseils précieux ainsi que leur disponibilité pour que notre formation soit possible.

Nous témoignons notre reconnaissance à tout le personnel de Ooredoo pour leur collaboration.

Nous remercions tout le corps professoral du département d'informatique de L'université Saad Dahlab de Blida pour les efforts consentis dans notre formation.

Nos sincères remerciements à tous ceux qui de près ou de loin ont contribué à l'élaboration de ce mémoire de fin d'étude.

Résumé

Ce travail propose de fournir un système de collecte et d'analyse d'évènements pour surveiller la sécurité des actifs informatiques dotés de systèmes d'exploitation Ubuntu ou Windows à l'aide d'un SIEM : un outil d'aide à la gestion centralisée des journaux avec des tableaux de bord pour surveiller les logs analysés en temps réel et générer des alertes de sécurité en cas d'anomalie.

Le système implique des agents pour collecter les logs des systèmes d'exploitation Windows et Ubuntu et envoyer ces données dans un format brut au serveur de traitement pour la normalisation et le parsing. Une fois parsés, les logs sont indexés et stockés dans une base de données No-SQL qui permettra leur exploitation par un outil de visualisation en format lisible et sous forme de graphes et de tableaux assemblés dans des tableaux de bord (Dashboard). Après la visualisation de divers logs de sécurité d'intérêt, des alertes sont créées lors de l'identification d'une tentative d'attaque tel que les attaques par force brute et par injection.

Les tests ont été effectués en générant les logs de sécurité souhaités depuis des machines virtuelles Windows 10 et Ubuntu 18 afin d'être capturés par le système conçu et mis en œuvre.

Mots clés : SIEM, centralisation, gestion de logs, évènement de sécurité, log, surveillance, alerte, incident, tableau de bord, attaque.

Abstract

This work proposes to provide an event collection and analysis system to monitor the security of IT assets running Ubuntu or Windows operating systems using a SIEM: a log management tool with dashboards to monitor the analysed logs on a real time basis and generate security alerts in case of anomalies.

The system involves agents to collect logs from the Windows and Ubuntu operating systems and send these data in a raw format to the processing server for standardisation and parsing. Once parsed, the logs are indexed and stored in a readable format in a No-SQL database that allows their use by visualization tools as graphs and tables gathered in dashboards. After viewing various security logs of interest, alerts are created when identifying an attempted attack such as brute force and injection attacks.

The tests were performed by generating the desired security logs from Windows 10 and Ubuntu 18 virtual machines in order to be captured by the designed and implemented system.

Keywords: SIEM, centralization, log management, security event, log, monitoring, alert, incident, dashboard, attack.

الملخص

يقترح هذا العمل توفير نظام لجمع الأحداث وتحليلها لمراقبة أمن أصول تكنولوجيا المعلومات التي تشغل أنظمة تشغيل أوبونتو أو أنظمة التشغيل ويندوز باستخدام سيم :أداة لإدارة السجلات مع لوحات معلومات لمراقبة السجلات التي يتم تحليلها في وقت فعلي وإنشاء تنبيهات أمنية في حالة حدوث حالات شاذة.

يشمل النظام وكلاء لجمع السجلات من نظامي التشغيل ويندوز و أوبونتو وإرسال هذه البيانات في شكل خام إلى خادم المعالجة للتوحيد والتحليل. وبمجرد تحليل السجلات ، يتم فهرستها وتخزينها في قاعدة بيانات تسمح باستخدامها بواسطة أداة تصوير في شكل قابل للقراءة وفي شكل رسوم بيانية وجداول مجمعة في لوحة القيادة. بعد الاطلاع على مختلف السجلات الأمنية ذات الأهمية ، يتم إنشاء إنذارات عند تحديد محاولة الهجوم مثل القوة الغاشمة وهجمات الحقن

وقد أجريت الاختبارات عن طريق توليد السجلات الأمنية المطلوبة من ويندوز 10 وآلات أوبونتو 18 الافتراضية من أجل حتى يتم التقاطها بواسطة النظام المصمم والمطب.

الكلمات المفتاحية: سيم، مركزية، إدارة السجل، حدث أمني، سجل، مراقبة، تنبيه، حادث، لوحة القيادة، هجوم.

Table des matières

Listes des figures

Liste des tableaux

Liste des acronymes et abréviations

A

ANSSI : L'Agence nationale de la sécurité des systèmes d'information.

ASCII : American Standard Code for Information and Interchange.

ASOC : Advanced Security Operation Center.

API : Application Programming Interface

C

CEF : Common Log Format

CLF : Common Log Format

CREST : Council of Registered Security Testers

D

DHCP : Dynamic Host Configuration Protocol

DNS : Domain Name Service

DoS : Denial of Service.

E

ELF : Extended Log Format

F

FTP : File Transfer Protocol

H

HTTP : Hypertext Transfer Protocol.

I

IDS : Intrusion Detection System

IP : Internet Protocol

IPS : Intrusion Prevention System

ISO : International Organization for Standardization

ITIL : Information Technology Infrastructure Library.

L

LDAP : Lightweight Directory Access Protocol.

LEEF: Log Event Extended format.

LFI: Local File Inclusion.

N

NIST: National Institute of Standards and Technology

NCSA : National Center for Supercomputing Applications

O

OS: Operating System.

OWASP : Open Web Application Security Project

S

SEM : Security Event Management.

SFTP : Secure File Transfer Protocol

SI : Système d'information

SIEM : Security Information and Event Management

SIM : Security Information Management

SOC : Security Operation Center

SQL : Structured Query Language.

SSH : Secure Shell

V

VPN : Virtual Private Network

X

XSS : Cross-Site Scripting.

Introduction générale

Introduction générale

Avec le développement et la croissance régulière de plusieurs technologies, la sécurité informatique est devenu un enjeu crucial pour tout type d'entreprise. Etant donné que le système d'information représente un patrimoine essentiel de l'entreprise, la sécurité de ce dernier est primordiale.

La sécurité des systèmes d'information, d'une manière générale, consiste à assurer que les ressources d'une organisation sont uniquement utilisées dans le cadre prévu. Pour faire face aux problème liés à la sécurité, la mise en place des mécanismes de sécurité permettant d'interférer avec le système et d'assembler les différentes informations en temps réel est indispensable. Pour ce faire, l'adoption du concept de centralisation et gestion des logs est le bon choix.

La centralisation repose sur le principe de localisation sur un même emplacement, les différents types de logs générés par les différentes unités du SI, afin de mieux les superviser et surveiller.

Problématique

Les systèmes informatiques de toute organisation génèrent des milliers de journaux par minute, parmi lesquels se trouvent des évènements de sécurité. Le défi consiste à identifier et filtrer ces évènements de sécurité importants qui pourraient être utile pour détecter une atteinte à la sécurité. Si nous pouvions collecter les événements de sécurité générés, et avoir une connaissance en temps réel de tous les événements qui se produisent, nous aurons alors une vue globale de la sécurité de l'organisation.
Avec l'analyse de ces données, il est plus facile pour une entreprise de déterminer sa conformité aux politiques réglementaires en matière de sécurité de l'information.

Les systèmes actuels utilisés pour l'extraction et l'analyse des événements de sécurité font face à des défis majeurs en raison de la taille énorme des fichiers logs, de la complexité de la gestion de ces logs sur plusieurs hôtes individuellement ainsi que de la difficulté de la compréhension des modèles d'attaque liés à un incident de sécurité. Cela conduit à une lente analyse des événements.

La plupart des solutions existantes sont coûteuses, standards et ne peuvent être flexibles aux besoins de la plupart des organisations, Ooredoo n'en fait pas l'exception.

Objectifs

Afin d'avoir une vision globale en terme de sécurité de l'entreprise, le service chargé de la sécurité de l'information de Ooredoo nous a convié à participer dans un projet de déploiement d'une solution de centralisation et de gestion des fichiers journaux.

Notre mission est de mettre en place une solution qui permet la bonne gestion des fichiers logs de l'entreprise afin de permettre aux analystes de maintenir une surveillance simple et efficace

Chapitre 1 : Contexte général du projet

1. Chapitre 1 : Contexte général du projet

1.1.Introduction

Ce chapitre est une introduction au contexte du travail, nous présenterons l'organisme d'accueil et le service de sécurité où s'est déroulé le stage avec des définitions et notions de base reliées à notre thème.

1.2.Présentation de l'organisme d'accueil

1.2.1. Entreprise Ooredoo

Ooredoo est une société de télécommunication d'origine qatarie. Présente en Algérie depuis le 23 décembre 2003 sous le nom de Nedjma, devenu Ooredoo le 21 novembre 2013. C'est la marque commerciale mobile de Wataniya Télécom Algérie.

Acteur essentiel du secteur des nouvelles technologiques, Ooredoo s'appuie sur les progrès rapides de la technique pour développer des services adaptés, innovants et de qualité ; Aujourd'hui Ooredoo est un opérateur universel qui répond à tous les besoins de ses clients particuliers ou entreprise en présentant une gamme d'offres et de services novateurs, en respect avec les standards internationaux.

1.2.2. Service de sécurité de l'information

Le système d'information est considéré comme le cœur de toute entreprise, il représente l'ensemble des moyens organisationnels, humains et technologiques mis en œuvre pour la gestion de l'information. Il doit être immunisé de toute faille de sécurité qui risquerait de compromettre l'information qui y circule, en terme de confidentialité, d'intégrité ou de disponibilité.
Le Service de sécurité de l'information a pour vocation d'assurer et de maintenir la sécurité des systèmes d'information d'Ooredoo, en accomplissant les missions et les responsabilités qui lui sont attribuées :

- Participer à l'élaboration et le développement des politiques de sécurité d'information.
- Participer à l'élaboration et la maintenance continue du système de gestion de sécurité de l'information et ISO27001.
- Conduire et assurer le suivi d'audits internes et avec des partenaires pour mesurer l'efficacité des processus d'organisation de la sécurité de l'information en ligne avec les exigences ISO27001/ 2.
- Faire un diagnostic et identifier les points faibles du système de sécurité informatique.
- Proposer des solutions pour sécuriser les systèmes d'information.
- Proposer des stratégies de sécurité à longs terme.
- Analyser les journaux de transaction et extraire des informations suspectes.
- Interpréter les corrélations de Logs.

- Assurer le reporting (mois, trimestre, semestre, année) etc.

1.2.2.1.Architecture du service

Le bon fonctionnement et l'efficacité de la sécurité d'un système d'information ne repose pas uniquement sur les outils de sécurité ou sur une stratégie mais également sur les équipes de ce service. À Ooredoo on en trouve trois. Chaque équipe doit accomplir les tâches qui lui sont accordées.

i. Equipe CSA (Cyber Security Architecture)

L'alignement des trois piliers : technologique, humain et organisationnel sur les objectifs de sécurité de l'entreprise est la principale mission que cette équipe effectue.
Elle prend en charge :

- La mise en place des politiques et des stratégies de sécurité et l'architecture technologique.
- Évaluation des Processus et des procédures de gestion des incidents de sécurité et aux évolutions du périmètre SOC/ SIEM.
- Validation des Projet IT sur le point de vue sécurité.
- La veille sécuritaire.
- Effectuer des recherches de produits de sécurité et d'évaluer leur pertinence pour la compagnie
- Faire évoluer la politique InfoSec et les procédures afin de rester alignés avec l'évolution technologique et les bonnes pratiques du secteur TIC et la norme ISO 27001.
- Effectuer des tests de vulnérabilité et faire le suivi de chaque vulnérabilité trouvée pour assurer l'application des correctives (Vulnerability Management, Patch Management)
- Effectuer des tests d'intrusion pour simuler l'impact des vulnérabilités sur l'entreprise et pour évaluer les contrôles de sécurité.
- Audits d'Architecture, configuration et de code applicatif.

ii. Equipe ASOC (Advanced Security Operation Center) plateforme

ASOC est l'équipe technique du service de sécurité. Les membres de cette équipe sont responsables de :

- L'administration du système (création des comptes, utilisateurs...)
- La mise à jour de la plateforme
- La maintenance de la plateforme
- L'application des use-cases créés par la CSA
- Garantir l'opérationnalité des outils 24h/24

iii. Equipe SOC (Security Operation System)

Un Centre des opérations de sécurité (SOC) est l'endroit où les systèmes d'information de l'entreprise sont surveillés, évalués et protégés. IL est typiquement dédié à la détection, à l'investigation et à la réponse des événements déclenchés par la logique de corrélation liée à la cyber sécurité.

L'équipe SOC est devisée en trois niveaux :

Niveau 1 : l'équipe de ce niveau travaille par shift, elle est chargée de surveiller le système 24h/24, détecter les menaces et y répondre rapidement.
Les membres de cette équipe sont généralement les premiers intervenants en cas d'incident. Ce sont les soldats au front qui luttent contre ces derniers. S'ils ne parviennent pas à les atténuer pendant une période de 5 à 10 minutes l'incident sera escaladé au niveau supérieur.

Niveau 2 : les membres de cette équipe sont plus expérimentés que ceux du niveau 1. Ils évaluent les incidents identifiés par les analystes de niveau 1, réalisent une analyse approfondie des informations et les investigations nécessaires afin de localiser les systèmes affectés et l'ampleur de l'incident. S'ils n'arrivent pas à résoudre la menace rencontrée, ils font appel aux niveau 3.

Niveau 3 : On les nomme « Analystes de sécurité ». Cette équipe traite les incidents critiques. Elle réalise des évaluations de la vulnérabilité et des tests d'intrusion afin d'évaluer la résilience de l'organisation et d'isoler les points faibles à surveiller. Elle examine les alertes, les informations sur les menaces et les données de sécurité. L'objectif principale de ce tiers est d'identifier les « root » causes (causes fondamentales) pour éviter de commettre les mêmes erreurs les prochaines fois.

1.2.2.2.Solutions de sécurité utilisées

Comme toute entreprise existante, Ooredoo vise à assurer la sécurité du son système d'information (SSI). Pour garantir la SSI, le service de sécurité d'Ooredoo dispose d'un ensemble de solutions permettant de maximiser ses chances de lutte contre les menaces et de réduire l'occurrence des éventuels incidents.

- **Pare feu** (Firewall en anglais) : Un pare feu est un élément du réseau informatique, il peut être logiciel, matériel ou une combinaison des deux. Le firewall interconnecte des réseaux de niveaux de sécurité différents, son rôle est de sécuriser le réseau informatique en définissant les communications autorisées et interdites selon les règles de sécurité.

- **Système de détection d'intrusion** (ou IDS, *Intrusion Detection System*) : Un IDS est un mécanisme qui a pour objectif de repérer tout type de trafic potentiellement suspect ou malveillant sur la cible analysée. En gros, il détecte les activités anormales qui s'éloignent de la norme [1].

- **Système de prévention d'intrusion** (ou IPS, *Intrusion Prevention System*) : les IPS sont similaire aux IDS mais la principale différence réside dans le fait que les systèmes de prévention des intrusions sont capables de bloquer ou d'empêcher activement les intrusions détectées [2].

- **Serveur proxy :** un serveur proxy est un serveur qui sert d'intermédiaire entre les machines de deux réseaux distincts. C'est une sorte de passerelle, utilisée pour effectuer des requêtes à la place des clients en vue de préserver leur anonymat.

- **Scanneur des vulnérabilités :** Ce sont des programmes conçus pour détecter et trouver les failles de sécurités des systèmes informatiques et des systèmes de communications, afin de les corriger avant que les pirates informatiques ne les exploitent [3].

- **Gestion d'accès à privilège :** un accès à privilèges, ou accès « root », permet de modifier les configurations d'un système, d'installer et désinstaller des programmes, de créer ou supprimer des comptes d'utilisateurs, ou encore d'accéder à des données sensibles. la gestion des accès à privilège aide les entreprises à empêcher tout usage inapproprié des données cruciales et donc de réduire les risques.

- **Antivirus :** Les virus sont des programmes informatiques capables de se dupliquer par eux-mêmes et de perturber plus ou moins gravement le fonctionnement de l'ordinateur infecté : ralentissement de la machine, suppression de fichiers, destruction des données … Un antivirus va scanner en continu les postes de travail de façon à repérer, bloquer ou éradiquer les virus en temps réel [3].

- **Antimalware :** En réalité, le « malware » est un terme générique qui englobe tous les types de logiciels malveillants : les virus, les chevaux de Troie, les vers, les ransomwares, … Les anti-malwares protègent les systèmes contre tout type de malware [3].

- **Anti spam :** Les spams sont tous des emails qu'on reçoit dans la boîte électronique sans n'avoir rien demandé. Généralement, il s'agit d'envois en grande quantité effectués à des fins publicitaires. Le logiciel anti-spam a pour objectif de faire le tri dans le courrier en plaçant tous ces emails indésirables dans un répertoire approprié [3].

- **Solutions de gestion de fichiers journaux :** Ces solutions permettent l'automatisation de l'analyse des journaux, amélioration des opérations informatiques, réduction des menaces et garantie de la conformité système d'information [3].

1.3.Notions de base

1.3.1. Information de sécurité

On appelle information de sécurité toute information contenue dans le système d'information susceptible d'avoir un impact sur la performance des fonctions de sécurité, pouvant entraîner l'échec de l'application de la politique de sécurité du système [4].

1.3.2. Evènement de sécurité

Un événement, est une action ou un fait pouvant être identifié par un programme et ayant une signification pour le matériel ou les logiciels du système. Un événement de sécurité désigne tout événement susceptible d'avoir des conséquences sur la sécurité des informations : C'est Un changement observé dans le comportement normal d'un système, soupçonné d'être un incident de sécurité, mais pas encore validé [5].

1.3.3. Fichier journal

Un fichier journal (ou log), est un fichier généré automatiquement permettant de stocker l'historique des évènements survenus dans un système. Ces évènements sont horodatés et ordonnés en fonction du temps [6]. Il a une structure ASCII qui est lisible par les humains.

Les logs ont un intérêt et une importance cruciale en informatique, car il s'agit là de savoir ce qui se passe dans les « coulisses » et si quelque chose devait se produire dans un système complexe.

Par exemple :

- **Expliquer** une erreur, un comportement anormal, un crash sur un service comme un service web.
- **Retracer** la vie d'un utilisateur, d'une application, d'un paquet sur un réseau sur les logs d'un proxy et des éléments réseau par exemple.
- **Comprendre le fonctionnement** d'une application, d'un protocole, d'un système comme les étapes de démarrage d'un service SSH [1]sous Linux.
- **Être notifié** d'un comportement, d'une action, d'une modification tel qu'une extinction ou un démarrage système.

1.3.3.1.Types des fichiers journaux

Les fichiers journaux proviennent de différentes sources, on cite les exemples suivants :

- Les fichiers log issus des systèmes,
- Les fichiers log issus des serveurs,
- Les fichiers log issus des sites web,
- Les fichiers log issus des pare-feu,
- Les fichiers log issus des systèmes de détection d'intrusion.

1.3.3.2.Formats des fichiers journaux

i. Formats standards

a. Syslog format

Les logs système sont sauvegardés dans des fichiers textes au format syslog. Il comporte trois parties :

- Un timestamp qui est la date et heure du log. Il est au format "Mmm dd hh:mm:ss".
- Un identifiant de la machine qui a généré le log. Cela peut être le nom de la machine (son hostname) ou son adresse IPv4 ou IPv6.
- Le message, qui est un message texte comportant des informations.

[1] Le service SSH (Secure Shell) : le service ssh permet aux utilisateurs de contrôler et de modifier leurs serveurs distants sur Internet [44].

```
Mar  1 06:25:43 server1 sshd[23170]: Accepted publickey for server2 from
172.30.128.115 port 21011 ssh2

Mar  1 07:28:33 server1 su: BAD SU kkent to root on /dev/ttyp2
```

Figure 1 : *Exemple de logs sshd.*

- *Mar 1 06 :25 :43* : Timestamp
- serveur1 : Nom de la machine
- *« Accepted publicket for server2 from 172.30.128.115 port 21011 ssha »* : Message

Ce log signifie qu'une connexion ssh a été acceptée pour le serveur1 ayant l'adresse 172.30.128.115 via le port 21011 utilisant le protocol ssh2 le 1er mars à 6h25m43s.

b. Web log format

Format dédié aux logs qui sont générés des serveurs web. Parmi les formats les plus répandus :

➢ CLF (Common Log Format)

Egalement connu sous le NCSA (National Center for Supercomputing Applications) Common Log Format , est un format de fichier log standard utilisé par les serveurs Web lors de la génération des logs [7].

```
161.31.132.116- - [21/Dec/2001:08:42:55 -0500] "GET /home.htm HTTP/1.0" 200 4392
```

Figure 2 : *Exemple de log CLF.*

- *161.31.132.116 :* L'adresse IP du client (hôte distante) qui a envoyé la requête.
- *[21/Dec/2001:08:42:55 -0500] :* La date est l'heure de la requête.
- *GET /home.htm HTTP/1.0 :* La méthode de requête, la page demandée et le protocole utilisé.
- *200 :* Le numéro de code de réponse de serveur.
- *4392 :* La taille de la page demandée en octets.

Ce log signifie que le client *161.31.132.116* a demandé l'accès à la page */home.*htm via le protocole *http* utilisant la methode *GET.* Sa demande a été acceptée et *4392* octets ont été transmis pour la requête le *21 Decembre 2001 à 8h42m55s.*

➢ **ELF (Extended Log Format)**

Le format de journal étendu (ELF) est un format de fichier texte normalisé, tel que le format de journal commun (CLF), utilisé par les serveurs Web lors de la génération de fichiers journaux , mais les logs ELF fournissent davantage d'informations et de flexibilité [8].

```
161.31.132.116 - - [21/Dec/2001:08:42:55 -0500] "GET /home.htm HTTP/1.0" 200 4392
"http://fr.search.yahoo.com/fr?p=peinture" "Mozilla/4.7 [en] (Win98)"
```

Figure 3 : *Exemple de log ELF.*

Le format ELF a les mêmes champs que le format CLF, avec l'ajout d'autre paramètres pour plus de détails, tel que :

- *http://fr.search.yahoo.com/fr?p=peinture* : La page de référence à partir de laquelle la requête est lancée.
- *Mozilla/4.7 [en] (Win98)* : Le navigateur et le système d'exploitation utilisés par l'utilisateur.

Ce log signifie que le client *161.31.132.116* a demandé l'accès à la page */home.htm* via le protocole *http* utilisant la methode *GET* depuis la page *http://fr.search.yahoo.com/fr?p=peinture* sous *Windows98*. Sa demande a été acceptée et *4392 octets* ont été transmis pour la requête *le 21 Decembre 2001 à 8h42m55s*.

ii. Formats propre à des organismes

a. LEEF (Log Event Extended format)

Le format LEEF (Log Event Extended Format) est un format de journal personnalisé pour IBM® Security QRadar (SIEM est une plateforme de gestion de sécurité) qui contient des événements lisibles et faciles à traiter pour QRadar [9].

b. CEF (Common Event Format)

CEF est un format de journal à base de texte développé par ArcSight ™ et utilisé par les produits HP ArcSight ™. Il s'agit d'un format extensible à base de texte conçu pour prendre en charge plusieurs types de périphériques en offrant les informations les plus pertinentes [10].

1.3.4. Vulnérabilité

Une vulnérabilité connue aussi sous le nom de « faille » est une faiblesse qui compromet la sécurité ou la fonctionnalité d'un système, qui peut être exploitée par des menaces, permettant aux pirates de contourner les politiques de sécurité et d'avoir accès non autorisé à un système informatique. [11].

1.3.5. Menace

la menace est définit dans ***Le Glossaire National 'Information Assurance'*** comme étant:

Toute circonstance ou événement susceptible d'avoir un impact négatif sur un SI par le biais d'un accès non autorisé, de la destruction, de la divulgation, de la modification de données et / ou d'un déni de service [12].

Une menace est un danger potentiel qui pourrait exploiter une vulnérabilité afin de nuire à un système et porter atteinte à sa sécurité informatique.

1.3.6. Alerte

C'est une notification prédéfinie qui se déclenche dès la survenance d'un ou de plusieurs évènements particuliers, afin d'avertir l'utilisateur des violations potentielles de la sécurité lui permettant de prendre des mesures correctives.

C'est un message décrivant une circonstance se rapportant à la sécurité réseau. Les alertes viennent souvent de systèmes de surveillance actifs sur le réseau.

1.3.7. Incident

Un Incident est défini par ***ITIL*** (*Information Technology Infrastructure Library* pour *Bibliothèque pour l'infrastructure des technologies de l'information*) comme étant :

« Tout événement qui ne fait pas partie du fonctionnement standard d'un service et qui cause ou peut causer une interruption ou une diminution de la qualité de ce service. » [13].

1.3.8. Attaque

Une attaque est le résultat de l'exploitation d'une faille d'un système informatique (Système d'exploitation, logiciel, erreur de configuration, ...etc.) à des fins non connues par l'exploitant du système et elle est généralement préjudiciable [14].

Autrement dit c'est tout type d'action offensive qui vise des systèmes d'information, des infrastructures, des réseaux informatiques ou des dispositifs informatiques personnels, en utilisant diverses méthodes pour voler, altérer ou détruire des données ou des systèmes d'information.

1.3.8.1. Types d'attaques

i. Attaques directes

Les attaques directes se produisent uniquement lorsqu'un ordinateur est connecté à Internet ou à un réseau local ou l'attaquant utilise des logiciels pour envoyer les paquets directement à partir de son ordinateur à la victime.
Dans ce cas, il est possible en général de remonter à l'origine de l'attaque, et déterminant l'identité de l'attaquant [15].

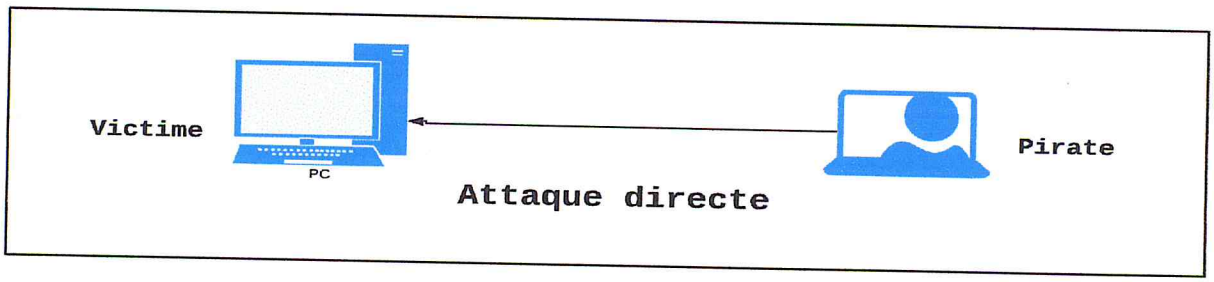


Figure 4 : *Attaque directe.*

ii. Attaques par rebond

Généralement, les pirates privilégient les attaques par rebond qui consistent à utiliser un ou des systèmes intermédiaires, participant à leur insu à l'attaque, et permettant à l'attaquant de dissimuler son identité et dans le but d'utiliser les ressources de la machine servant de rebond [16].

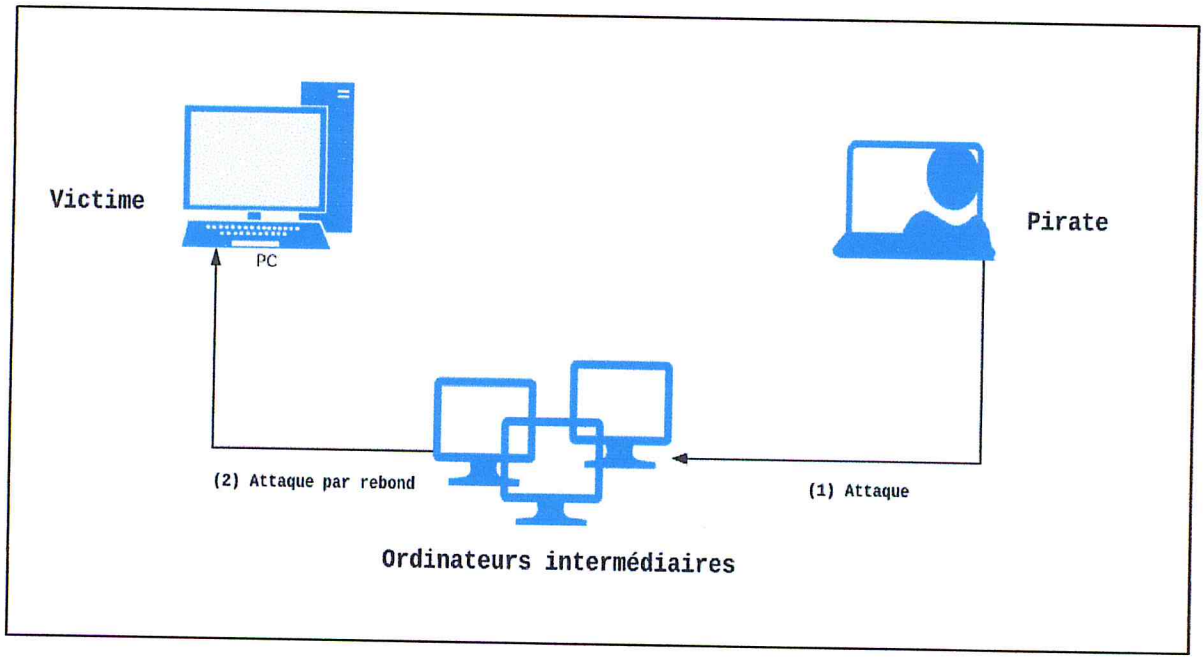


Figure 5 : *Attaque par rebond.*

iii. Attaques par réponse

Cette attaque est une alternative de l'attaque par rebond. Son principe est d'envoyer une requête à la place d'une attaque à l'ordinateur intermédiaire afin qu'il la répercute. Donc l'ordinateur victime va recevoir la réponse de cette requête [15].

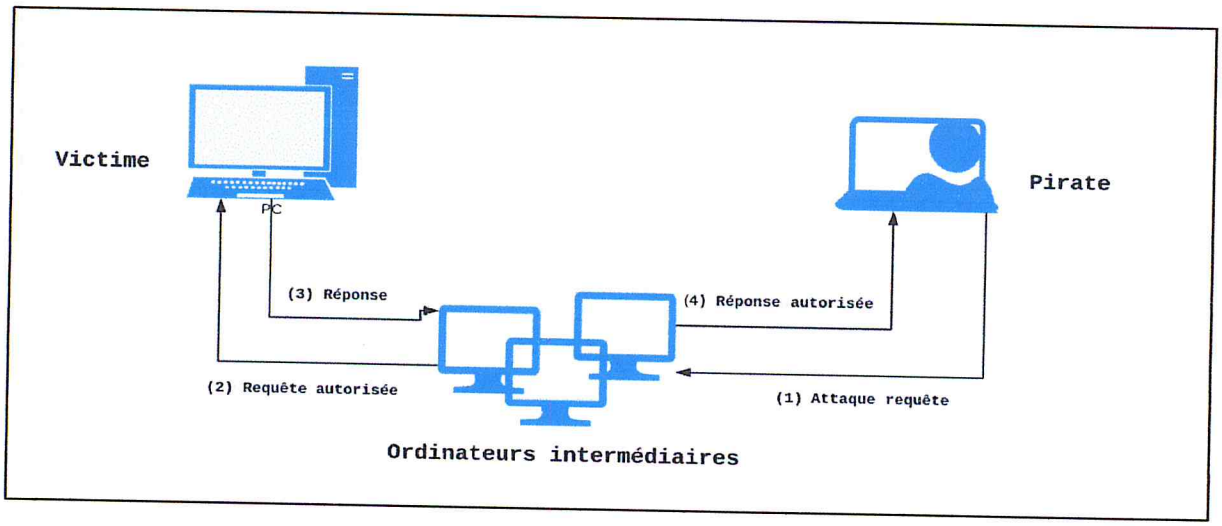


Figure 6: *Attaque par réponse.*

1.3.8.2.Les attaques les plus courantes

Comme nous l'avons défini plus haut, Une **cyberattaque** est tout type d'action offensive qui vise des systèmes, des infrastructures ou des réseaux informatiques.

Nous allons décrire les types d'attaques les plus courants :

i. Attaques par injections

a. Injection SQL

D'après l' « OWASP » les injections sont considérées comme le risque N° 1. Les attaques par injection se produisent via une application web lorsqu'une donnée non contrôlée est envoyée à un interpréteur dans le cadre d'une commande ou d'une requête. Les données malveillantes de l'attaquant peuvent leurrer l'interpréteur afin d'exécuter des commandes imprévues ou d'accéder à des données non autorisées [17].

Autrement dit, les injections consistent à détourner les requêtes en y injectant un code non prévu et pouvant compromettre la sécurité du système.

Prenons l'exemple le plus basique qui consiste à usurper une identité pour se connecter à une application Web sur des entrées client chargées d'exécuter une requête SQL :

```
String query = "SELECT * FROM accounts WHERE username='"
+request.getParameter("login") +"' AND  password = '"+ request.getParameter("pass") +"'";
```

L'attaquant modifie les champs des inputs, par exemple login égalera « *admin' or 1=1--* » :

```
SELECT * FROM accounts WHERE username='admin' or 1=1--' AND password='17101612'
```

Cela va changer le sens de la requête d'où l'utilisateur a un accès auquel il n'a pas le droit et le risque ne se résume pas seulement qu'à cela, il peut causer beaucoup de dommages dans la BD avec des requêtes imbriquées, unies … et toute la puissance de l'interpréteur (insert, drop, select …).

b. Les failles XSS (Cross-Site Scripting)

Les attaques de type Cross Site Scripting (XSS) constituent un type de problème d'injection, dans lequel des scripts hostiles sont injectés. Cette vulnérabilité permet aux attaquants d'insérer leur code côté client dans ces pages Web victime afin de détourner des sessions utilisateur, défigurer des sites Web, ou rediriger l'utilisateur vers des sites malveillants [17].

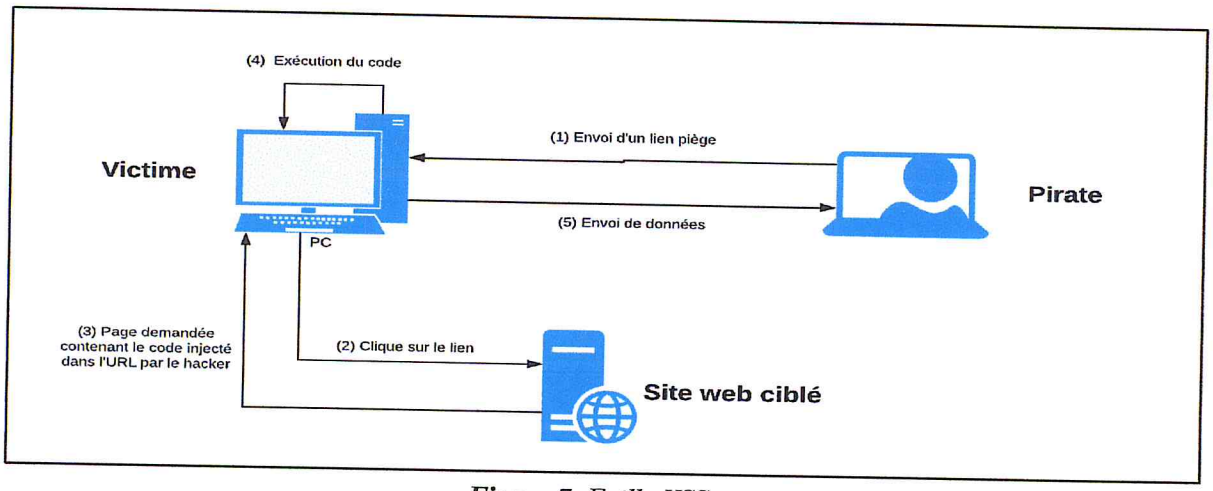


Figure 7: Faille XSS.

c. Injection de commande

L'injection de commande est également connue sous le nom d'injection de commande OS. Il s'agit d'une technique d'attaque utilisée pour exécuter des commandes sur un système d'exploitation hôte via une application Web vulnérable. Les attaques par injection de commande sont possibles lors de la transmission des données non sécurisées fournies par l'utilisateur (formulaires, cookies, en-têtes HTTP, etc.) à un shell système. Ces commandes sont exécutées avec les privilèges de l'application vulnérable [19].

Tout simplement on peut dire qu'une injection de commande est une vulnérabilité dans laquelle l'attaquant peut contrôler une ou plusieurs commandes en cours d'exécution sur un système.

ii. Attaque par mots de passe

a. Attaque par brute force

L'attaque par force brute est une méthode utilisée en cryptanalyse pour trouver un mot de passe ou une clé. Il s'agit de tester, une à une, toutes les combinaisons possibles.

b. Attaque par dictionnaire

L'attaque par dictionnaire est une méthode utilisée en cryptanalyse pour trouver un mot de passe ou une clé. Elle consiste à tester une série de mots de passe potentiels, les uns à la suite des autres, en espérant que le mot de passe utilisé pour le chiffrement soit contenu dans le dictionnaire. Si ce n'est pas le cas, l'attaque échouera.

L'attaque par dictionnaire est une méthode souvent utilisée en complément de l'attaque par force brute [18].

iii. Inclusion de fichier local (LFI)

LFI (également connu sous le nom d'Inclusion de fichier local) est le processus d'inclusion de fichiers sur un serveur via le navigateur web. En général, LFI se produit lorsqu'une application utilise le chemin d'accès à un fichier en tant qu'entrée. La vulnérabilité est alors due à l'utilisation de l'entrée sans validation adéquate. [20] En conséquence elle peut conduire à :

- L'exécution de code sur le serveur web,
- L'exécution de code sur le côté client comme Cross-site scripting (XSS),
- Déni de service (DoS),
- Extraction des données.

1.4.Conclusion :

Dans ce premier chapitre, nous avons présenté le contexte du projet, par la suite nous avons défini les différents termes utilisés tout en détaillant différents points essentiels dans la sécurité informatique : vulnérabilités, menaces, incidents et attaques.

Chapitre 2 : Supervision et Gestion de la Sécurité

2. Chapitre 2 : Supervision et gestion de la sécurité

2.1.Introduction

Dans le chapitre précédent, nous avons présenté le contexte général du travail ainsi que les notions de bases liées à notre sujet. Dans ce nouveau chapitre nous présenterons des approches de gestion de la sécurité liées à notre problème.

2.2.Supervision de la sécurité

La sécurité informatique est une condition préalable au succès durable d'une entreprise. De nombreuses mesures sont prises pour protéger les données. Cependant, les fonctionnalités étendues et complexes des systèmes informatiques offrent constamment de nouveaux points d'entrée pour les attaques. C'est pourquoi la surveillance et la supervision de manière continue et centralisées de la sécurité informatique de l'infrastructure et de ses composants est essentielle.

2.2.1. Objectifs et cadre de la supervision de la sécurité

D'après le guide de supervision de la sécurité [21], les objectifs de la supervision se résument en trois points essentiels :

- La recherche d'indicateurs d'incidents potentiels.
- Respecter les normes contractuelles ou de certification.
- Répondre aux exigences de conformité.

2.2.2. Processus de la supervision de la sécurité

Le processus de supervision de la sécurité s'appuie sur quatre phases très importantes pour assurer une supervision continue. *La figure 8* résume ces étapes de manière simple.

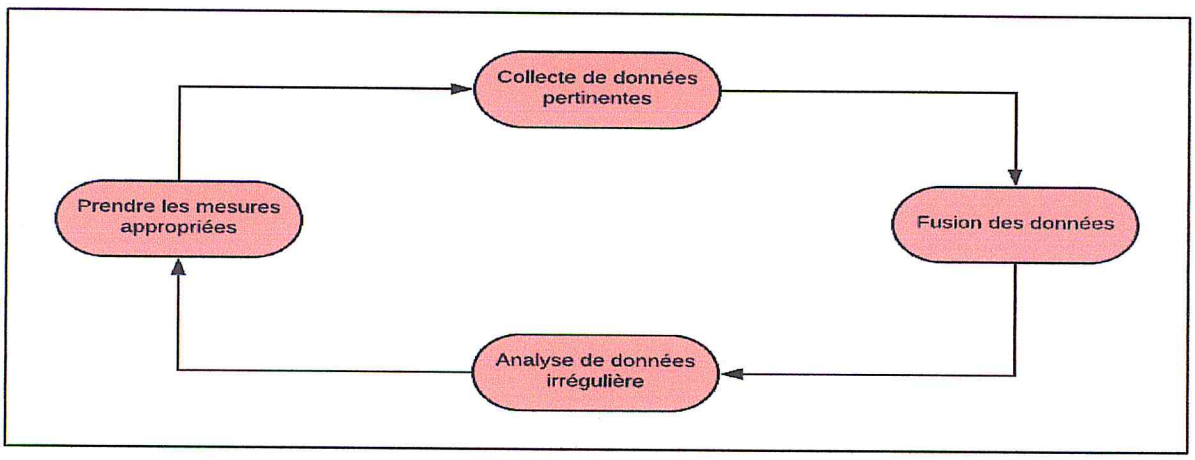


Figure 8 : *Processus de la supervision de la sécurité*

2.3.Les logs de sécurité

Les logs relatifs à la sécurité sont des logs qui contiennent des enregistrements ou évènements liés à la sécurité du système. En effet, ils sont parmi les seules informations qui permettent à la fois de mesurer le niveau de sécurité, de détecter d'éventuelles menaces et d'enclencher les éventuelles actions à entreprendre, le tout en temps réel ou pas.

2.3.1. Types de logs de sécurité

L'article de CREST [21] résume les différents types de journaux de sécurité comme suit :

Journaux du système	▪ Journaux d'activité du système ▪ Journaux des terminaux ▪ Journaux issus d'applications standards et personnalisées ▪ Journaux d'authentification ▪ Journaux de sécurité physique
Journaux issues des réseaux	▪ Journaux de courrier électronique, pare-feu VPN et Netflow.
Journaux techniques	▪ Journaux de proxy http ▪ Journaux DNS, DHCP et FTP[2] ▪ Journaux web et SQL Server
Journaux issus des outils de surveillance et de journalisation de la cyber sécurité	▪ Journaux de protection contre les logiciels malveillants (exemple : antivirus) ▪ NIDS ▪ NIPS ▪ Protection contre la perte de donnée (DLP) ▪ Autres dispositifs ou outils de gestion de la sécurité.

Tableau 2 : *Types de logs de sécurité.*

Les différents types de logs de sécurité sont détectés et traités dans la première phase de la supervision de la sécurité informatique en utilisant des outils de collecte, de parsing et d'analyse.

2.3.2. La gestion de logs

La gestion de logs fait partie intégrante d'une solution de gestion de la sécurité. Suivant le guide de gestion de log de la sécurité informatique de NIST [22], les éléments d'informations d'un incident peuvent être enregistrés à partir de plusieurs sources : routeurs, pare-feu, IDS, et les logs

[2] FTP (File Transfert Protocol) : Protocole de transfert de fichiers, comme son nom l'indique il est dédié à l'échange de fichiers sur un réseau [45].

Comme le schéma le montre, le processus de supervision de la sécurité se fait en 4 étapes. La collecte des données qui regroupe l'identification des logs à traiter, la normalisation de ses logs en des formats universels, le stockage des logs normalisés et leur filtrage en des documents partageant les mêmes caractéristiques. Après la collecte vient la fusion des données qui consiste en l'automatisation du processus et l'application des règles de corrélations tout en prenant comptes des règles de pare-feu et des signatures IDS existantes. Une fois la fusion achevée, c'est au tour de l'analyse qui est faite par des analystes et qui consiste en la vérification des règles appliquées, l'investigation et le contrôle du workflow pour assurer que tout fonctionne comme voulu. Cette étape permet d'analyser les risques en mettant en œuvre l'intelligence de cyber sécurité. La dernière étape, aussi exécutée par des analystes est la phase de prise de mesure : que ça soit en terme de remédiation, de réponse, de récupération, de gestion des incidents ou de création de rapports.

Le tableau suivant résume le paragraphe précédent :

Phase	Actions
Collecte	▪ Identification de logs ▪ Normalisation ▪ Stockage ▪ Filtrage
Fusion	▪ Automatisation ▪ Règles de corrélation ▪ Règles de pare-feux ▪ Signatures IDS
Analyse	▪ Règles prédéfinies ▪ Investigation ▪ Workflow ▪ Analyse de risques ▪ Intelligence de cyber sécurité
Prise de mesures	▪ Réponse, remédiation et récupération ▪ Mise à l'échelle ▪ Enquête ▪ Gestion des incidents ▪ Rapports

Tableau 1 : *Phases du processus de supervision de la sécurité.*

des applications et des services. Pour que les incidents soient détectés et gérés, la solution de gestion de la sécurité met en place des outils qui permettent d'analyser en temps réel, les logs, le trafic et d'autres informations de manière automatique et de générer des alertes.

2.3.2.1.Etapes de gestion de logs

La gestion de logs se fait en trois principales étapes divisées en sous-étapes que nous expliquerons comme suit :

i. Génération de logs

La génération des logs est bien évidement la première chose à faire afin que les logs méritent d'être lus et traités. Sur certains systèmes la production des logs n'est pas activée par défaut. Sur d'autres, seuls les logs critiques et supérieurs sont sauvegardés. Il peut alors, en fonction du besoin, rehausser ou abaisser la criticité des logs qui sont produits [23].

ii. Analyse et stockage de logs

a. La collecte

Les journaux sont la source idéale qui reflète l'image de ce qui se passe et ce qui fait un système en temps réel. La collecte permet de recueillir les journaux log provenant de n'importe quelle source, que ce soit de façon passive en mode écoute ou active en mettant en place des agents dédiés à la collecte qui sont capable de discerner les différents formats de logs.

b. L'agrégation

L'agrégation des journaux logs est un moyen d'assembler tous ces journaux en un seul endroit, elle permet la consolidation des différents formats de logs issus de différentes sources en un seul endroit pour faciliter les autres étapes de traitement tel que : l'analyse, la normalisation et la génération de rapports sur les informations importantes [24].

c. La normalisation

Le plus grand défi du traitement des journaux logs consiste à surmonter la diversité des formats de ces derniers. Pour permettre une interprétation efficace des données sur les différentes sources, ce processus de normalisation permet de convertir les logs originaux collectés dans un format universel, lisible et structuré. Tout en les décomposant en champs significatifs [25].

d. Le stockage

Une fois collectées, les journaux doivent être conservé stocker et archiver dans un emplacement de stockage.

iii. La supervision des logs

a) **Analyse**

L'analyse des logs permet de réduire et d'éviter les différentes menaces potentielles [26]. Elle consiste à décortiquer les journaux logs afin de savoir ce qui s'est passé et permet d'adopter en conséquence les mesures de correction ou de renforcement les plus adéquates. L'analyse peut se faire en temps réel pour un maximum de réactivité dès qu'une alerte est lancée ou à échéances

régulières. Elle se présente sous forme de visualisations et de Dashboard qui donnent une vue globale de ce qui se passe dans le système.

b) Corrélation

La corrélation consiste à relier les points et à corréler les événements des différentes sources de données. Ce travail de corrélation est basé sur des règles fournies par différents outils, prédéfinies pour différents scénarios d'attaque ou créées et ajustées par l'analyste. En termes simples, une règle de corrélation définit une séquence d'événements pouvant indiquer une atteinte à la sécurité. En fait, les règles permettent de condenser ces données en des ensembles de données plus faciles à gérer en éliminant le bruit et en pointant vers des événements potentiellement marquants [27].

c) Création des alertes

Cette fonction permet de créer des alertes qui déclenchent les notifications auprès des opérateurs ou de gestionnaires en temps réel. Leur création se base sur les règles de corrélation définies précédemment, elles se présentent sous forme d'un mail, SMS ou une notification si un problème (anomalie, attaque...) est détecté [28].

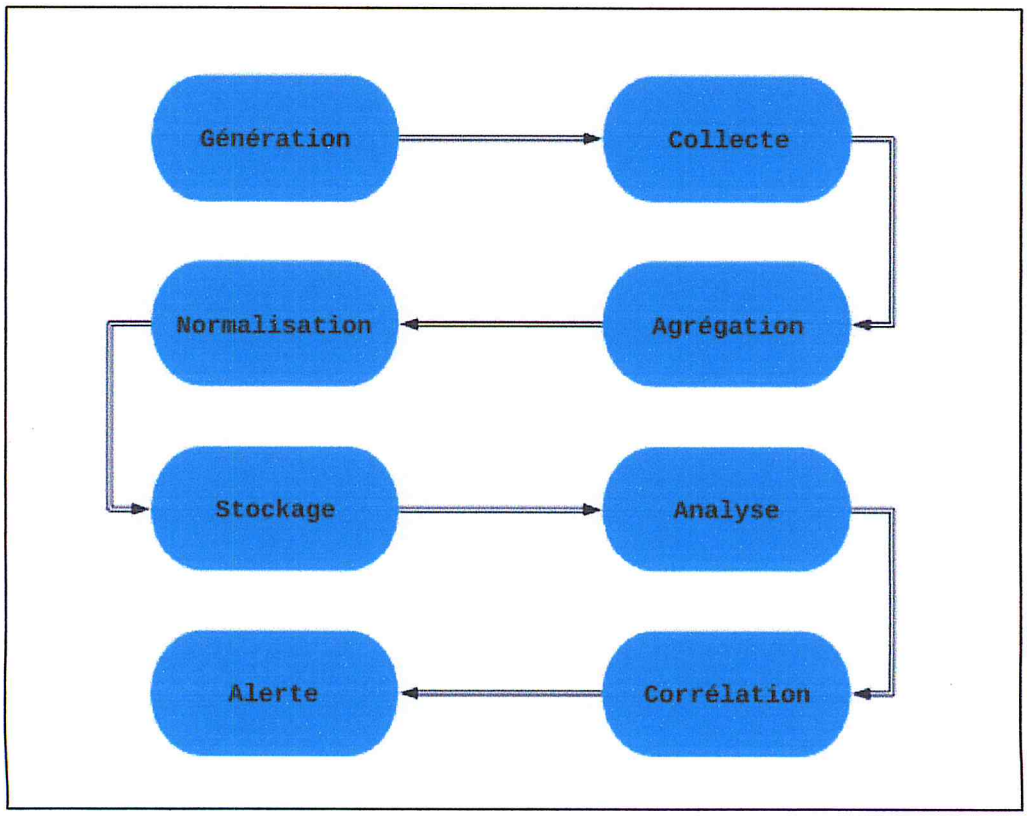


Figure 9: *Processus de gestion de logs.*

2.4.La centralisation des logs

La centralisation des logs consiste simplement à mettre sur un même système, une même plateforme, l'ensemble des logs des systèmes, applications et services des machines environnantes. Nous avons vu dans les points précédents le principe de la supervision, dont la surveillance des logs est une branche, qui consiste à centraliser les éléments de surveillance sur une plate-forme unique. [23]

La centralisation permet de :

- Avoir une vue d'ensemble d'éléments cruciaux à la bonne gestion d'un SI pour y mener des traitements
- En cas de crash ou de suppression des logs :
    - Diagnostiquer un crash
    - Garantir la survie des logs à une suppression

Une solution de centralisation et de gestion des logs rentre dans le cadre d'un SIEM (Security Information and Event Management).

La technologie SIEM a vu le jour il y a plus de 10 ans, elle est apparue pour la première fois sur le marché en 1997. C'est une approche de gestion de sécurité qui fournit une vue globale de la sécurité de l'information de l'entreprise en temps réel et éventuellement prévoir les menaces. Une solution de supervision de la sécurité réseau rentre dans la catégorie des SIEM (Security Information and Event Management). En effet, un SIEM encore appelé SIM (Security Information Management) est un outil qui permet d'avoir une vue unique sur tous les événements liés à la sécurité du réseau et des systèmes. Il comprend l'analyse et la corrélation de logs, de gestion des incidents et le reporting basé sur l'analyse d'événements. Un SIEM analyse d'autres données en plus des logs, mais la source de données primaire est le log [29].
Un SIEM agrège les données provenant des périphériques de sécurité, de réseau, des systèmes et des applications. Les données ainsi agrégées sont normalisées. Alors, un événement apparaissant plusieurs fois dans plusieurs sources différentes peut être corrélé. En plus des points cités plus haut, un SIEM fournit la capacité d'investigation et de supervision ce qui en fait « le noyau du SOC » [30].

Les fonctions d'un SIEM :

- Gestion centralisée de logs ;
- Corrélation de logs et mise en relation « cause à effet » ;
- Agrégation des événements de sécurité en une liste que l'on peut facilement gérer : classifier, catégoriser, etc.
- Permet de prévenir des dommages sur les ressources informatiques de l'entreprise ;
- Permet d'avoir un tableau de bord pour la gestion de la sécurité, l'assurance de conformité avec les politiques de sécurité, etc.

2.4.1. Solutions de centralisation propriétaires

2.4.1.1.Splunk

La plate-forme leader en matière d'intelligence opérationnelle. Cela permet aux curieux de regarder de près ce que les autres ignorent - les données de la machine - et de trouver ce que les autres ne voient jamais : des informations qui peuvent aider à rendre votre entreprise plus productive, plus rentable, plus compétitive et plus sûre. Splunk Enterprise est la principale plateforme d'intelligence opérationnelle en temps réel [31].Bien que Splunk soit une solution propriétaire, il existe une version gratuite ouverte au publique.

2.4.1.2.SolarWinds

Le logiciel de gestion d'événements d'information de sécurité SolarWinds (SIEM) est conçu pour fournir une automatisation, un soulagement et une connaissance approfondie de la situation sans la complexité des solutions concurrentes [32].

2.4.1.3.IBM QRadar

Security QRadar SIEM consolide les données d'événement de source de journal provenant de milliers de points de terminaison de périphériques et d'applications distribués sur un réseau. Il exécute immédiatement des activités de normalisation et de corrélation sur les données brutes afin de distinguer les menaces réelles des faux positifs [33].

2.4.1.4.MCAfee

La gestion performante des informations et événements de sécurité (SIEM) rassemble des données sur les événements, les menaces et les risques afin de fournir des informations de sécurité fiables, une réponse rapide aux incidents, une gestion transparente des journaux et des rapports de conformité extensibles. Enterprise Security Manager, au cœur de l'offre SIEM, consolide, corrèle, évalue et hiérarchise les événements de sécurité pour les solutions tierces et McAfee [34].

2.4.1.5.LogRhythm

C'est une plate-forme d'entreprise qui associe de manière transparente SIEM, la gestion des journaux, la surveillance de l'intégrité des fichiers et l'analyse des machines, aux analyses d'hôte et de réseau, au sein d'une plate-forme unifiée de renseignements de sécurité. Il est conçu pour faire face à un paysage en constante évolution de menaces et de défis, avec une suite complète d'outils haute performance pour la sécurité, la conformité et les opérations [35].

2.4.2. Solutions de centralisation Open Source

2.4.2.1.OSSIM

AT & T Cybersecurity propose AlienVault OSSIM qui est un outil SIEM open source basé sur leur solution AlienVault USM. Comme pour les entrées ci-dessus, AlienVault OSSIM combine plusieurs projets open source dans un même package. De plus, AlienVault OSSIM permet la surveillance des périphériques et la collecte des journaux [36].

2.4.2.2.SIEMonster

SIEMonster se situe à cheval entre SIEM gratuit et une solution payante, car il propose les deux. Comme beaucoup de solutions répertoriées, SIEMonster offre une plate-forme combinant plusieurs outils Open Source. Il offre donc une interface centralisée permettant de contrôler ces outils, la visualisation des données et les informations sur les menaces[36].

2.4.2.3.Prelude

Prelude OSS propose une version open source de la solution Prelude SIEM. Cela prend en charge une large gamme de formats de journaux et peut s'intégrer à d'autres outils de sécurité. Il offre également la normalisation des données d'événement dans un langage standard qui peut aider à prendre en charge d'autres solutions et outils de cybersécurité [36].

2.4.2.4.ELK

La pile ELK (ELK Stack) est une collection de trois produits open-source : Elasticsearch, Logstash et Kibana. Ils sont tous développés, gérés et maintenus par la société Elastic. C'est une plateforme de gestion centralisée de log qui intègre plusieurs technologies ensemble dans le but de permettre aux utilisateurs d'utiliser des données provenant de n'importe quelle source, quel que soit leur format, et de rechercher, analyser et visualiser ces données en temps réel [37].

2.4.2.5.Graylog

Graylog est une solution open source de gestion log développée en java et basée sur Elasticsearch qui permet de centraliser, monitorer et d'analyser les logs de différentes sources qui se retrouvent tous consultables au même emplacement. Il permet facilement de comparer et corréler les logs de différents services entre eux afin d'avertir en cas de rencontre d'incidents en temps réel [38].

2.4.3. Comparatif des SIEM Elk , Splunk, Graylog

Etant donné que notre approche consiste à implémenter un SIEM propriètaire à Ooredoo, ELK, Splunk et Graylog sont les outils les plus adéquats pour developper un outil SIEM flexible et qui répond aux besoins de l'entreprise.
Le tableau suivant représente une étude comparative des trois outils :

splunk>	ELK	graylog
Très simple d'installation, on créé un compte et on récupère le fichier d'installation sur le site officiel de Splunk.	L'installation est plus complexe que Splunk mais reste relativement simple grâce à la documentation en ligne.	Installation similaire à ELK (Graylog utilise également ElasticSearch), bonne documentation.
Configuration simple qui se fait depuis l'interface Web (configuration de port d'écoute, ajout de données...)	Configuration plus complexe car il faut configurer Logstash (il faut donc maîtriser un minimum les langages de script)	Configuration simple et similaire à Splunk car elle se fait là aussi depuis l'interface web.
Simple pour utilisation basique, il suffit de taper le mot clé recherché pour qu'il s'affiche en surbrillance. Recherche avancée basée sur la syntaxe de recherche SPL (Splunk Search Processing Language)	Simple également pour une basique utilisation, similaire à Splunk (mot clé = en surbrillance). Syntaxe de recherche avancée basée sur la syntaxe Lucene.	Utilisation basique simple, similaire à Splunk et ELK. Syntaxe proche de Lucene.
Les graphiques se créent depuis la recherche et grâce aux champs disponibles. Graphiques facilement réalisable et très complet.	Ils se créent aussi depuis la recherche et les champs disponibles mais cela nécessite une bonne configuration de Logstash. On peut également créer les graphiques depuis le menu « visualise » en appliquant les filtres que l'on souhaite. Graphiques légèrement moins complet que Splunk.	Graphique facilement réalisable depuis la recherche et les champs (similaire à Splunk). Graphiques cependant moins complet que Splunk et ELK.
Dashboard non interactif. Barre de recherche et temps non disponible par défaut. Il faut configurer les dashboards pour les rendre compatible avec les visualisations ce qui peut être vite contraignant. Possibilité de mettre un Dashboard en page d'accueil.	Dashboard interactif par défaut. Barre de recherche et barre de temps toujours disponible. Les dashboards s'adaptent en fonction des termes de recherches ou de la plage de temps sélectionnée. Dashboard facile à créer et à modifier. Point fort d'ELK.	Dashboard facile à créer et à modifier mais ils ne sont pas interactif et la barre de recherche / temps n'est pas disponible. Point faible de Graylog.
Nécessite la version Splunk Enterprise	Nécessite le X-Pack et donc de souscrire à un abonnement.	Alertes disponible gratuitement. Point fort de Graylog.
Nécessite la version Splunk Enterprise pour créer des utilisateurs et gérer leurs droits. Possibilité d'intégration Active Directory / LDAP.	Nécessite le X-Pack pour bénéficier de la fonction d'identification et la gestion des utilisateurs.	Gestion des utilisateurs disponible gratuitement. Intégration Active Directory / LDAP possible également. Point fort de Graylog.
Version gratuite limitée à 500 Mo de logs/jour. Nécessite également la version de Splunk Enterprise pour bénéficier des alertes, monitoring, support...	Open source sponsorisé par la société Elastic. Nécessite l'achat d'une licence (X-Pack) pour bénéficier de toutes les fonctionnalités (identification, alerting, monitoring...) et du support.	Open Source, possibilité de souscrire à un abonnement pour bénéficier d'un support. Dans ce cas, le prix varie en fonction de la quantité de donnée que l'on envoie à Graylog.

Tableau 3 : *Comparatif des outils SIEM [39]*

2.5. Discussion

- **Splunk** est l'outil le plus complet mais peut être assez complexe d'utilisation au départ. De plus il est propriétaire il faut donc impérativement une licence.
- **ELK** est simple d'utilisation et très intuitif mais sa configuration peut vite être complexe. Là aussi, pour une utilisation en entreprise, il est presque obligatoire d'adopter une licence pour bénéficier de toutes les fonctionnalités.
- **Graylog** est certainement l'outil le moins complet en termes de graphiques / plugins mais il est simple d'utilisation et de configuration. Gros point fort : il permet de gérer gratuitement les alertes et les utilisateurs (intégration Active Direcotry et LDAP). La documentation officielle est aussi très complète et détaillée.

Après comparaison des outils open-source Splunk, Graylog et ELK, nous jugeons que le meilleur choix serait de travailler avec : **ELK** en remplaçant les fonctionnalités payantes de génération d'alertes et de rapports par des modules Open-source.

2.6.Solution retenue : La pile ELK

La pile ELK (ELK Stack) est une collection de trois produits open-source : Elasticsearch, Logstash et Kibana. Ils sont tous développés, gérés et maintenus par la société Elastic.

Elle est conçue pour permettre aux utilisateurs d'utiliser des données provenant de n'importe quelle source, quel que soit leur format, et de rechercher, analyser et visualiser ces données en temps réel.

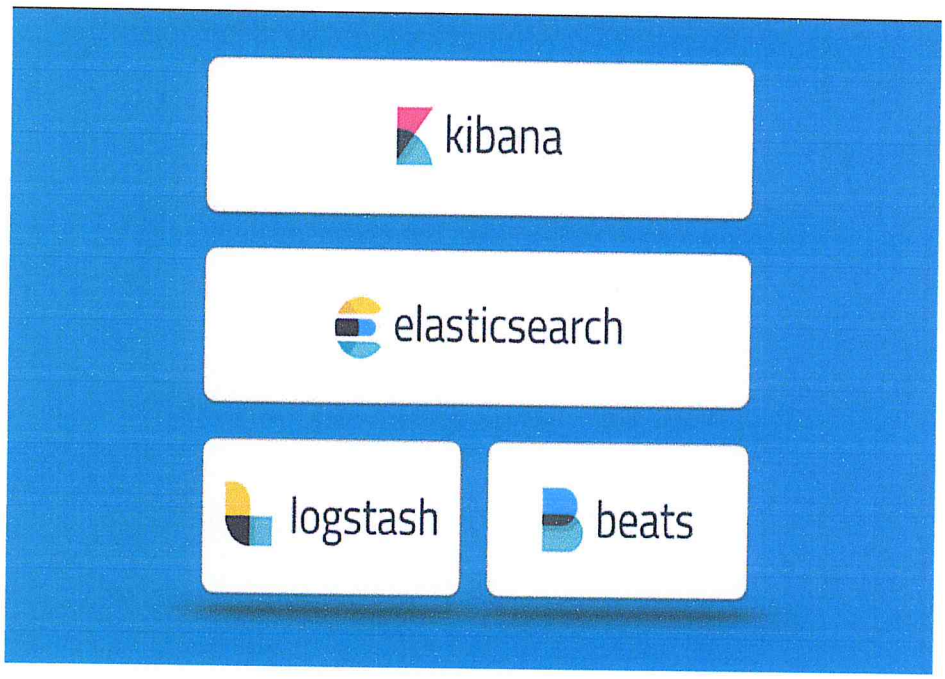


Figure 10 : *La plie ELK[40]*

Elasticsearch est un moteur de recherche et d'analyse RESTful distribué. C'est le cœur de la solution Elastic. Ce dernier est classé comme BD NoSql et est utilisé pour stocker toutes les données collectées sur l'infrastructure. Les points forts de la solution sont :

- La résilience et l'architecture hautement disponible
- La performance des requêtes grâce à l'indexation, particulièrement adapté à la recherche rapide de données de type métriques ou journaux.

Paradoxalement, c'est la couche avec laquelle on interagit le moins au début, puisqu'on passe surtout du temps à configurer en amont la collecte des logs au niveau Beats et Logstash, puis en aval avec la visualisation dans Kibana [41].

Logstash est un outil dédié au traitement des logs. Il dispose d'une grande quantité de collecteurs de donnée, ce qui en fait un outil très puissant et adapté à presque tous les besoins de récolte de logs.

Il centralise, normalise et stocke les données. Il faut le voir comme un outil qui va éclater tous les champs d'une ligne composant un log afin d'y faire des recherches complexes. Pour cela, il peut traiter, transformer les données pour nous donner une meilleure compréhension de l'information avant de les transmettre à Elasticsearch [41] .

Kibana est une interface web qui permet d'explorer/de visualiser des informations provenant d'Elasticsearch et de les afficher sous formes de graphiques. On peut ensuite consolider les graphiques en tableaux de bords (Dashboard).

Un gros point positif de la solution est la possibilité de filtrer en temps réel sur une sous-catégorie des données, soit via une barre de recherche, soit en cliquant directement sur les graphiques, en mode « drill down » [41].

Beats : Introduit depuis le renommage et la version 5 de la suite logicielle, Beats regroupe un ensemble de modules logiciels de type « agents » à déposer sur les serveurs à surveiller.

Ce sont ces agents, adaptés pour chaque contexte, qui récoltent les données brutes et les transmettent à la brique suivante (logstash). Les types d'agents Beats sont :

- Metricbeat
- Packetbeat
- Filebeat
- Winlogbeat

On peut directement envoyer les données collectées à Elasticsearch, la couche Logstash étant optionnelle, mais on perd de nombreuses fonctionnalités et il est préférable de garder Logstash [41].

SENTINL est un plugin d'application qui étend Kibana avec des fonctionnalités dynamiques d'alerte et de rapport, pour surveiller, informer et faire rapport sur les changements de la série de données en utilisant des requêtes standard, de validateurs programmables et de diverses actions configurables.

SENTINL est également conçu pour simplifier le processus de création et de gestion des alertes grâce à des observateurs « watchers ». Les watchers fournissent des alertes et des notifications basées sur les modifications des données [42].

Un Watcher se compose de quatre parties :

- Trigger : Où on programme la fréquence du lancement du watcher.
- Input : Où on spécifie la requête qui servira d'entrée pour la condition.
- Condition : Où on décide quand exécuter l'action,c'est-à-dire quand les résultats de la requête correspondent à la régle de corrélation.
- Action : Notification slack, rapport ou mail qui seront envoyés si la condition est satisfaite.

2.7.Conclusion

Dans ce second chapitre, nous avons commencé par expliquer deux principes fondamentaux de la gestion de la sécurité informatique : La supervision et la gestion de logs. Ensuite nous avons présenté une solution qui permet la gestion centralisée des fichiers journaux et les outils qui permettent de la déployer. Après une étude comparative, nous avons opté pour la solution ELK qui selon l'étude, serait le meilleur choix pour concevoir un outil SIEM flexible qui répond aux besoins de Ooredoo.

Chapitre 3 : Architecture et conception de la solution de la gestion centralisée de logs

3. Chapitre 3 : Architecture et conception de la solution de la gestion centralisée de logs

3.1.Introduction

Ce chapitre décrit l'architecture et la conception du système que nous avons conçu pour la centralisation et la gestion des fichiers logs.

3.2.Architecture du système

L'architecture de notre système peut être représentée en trois niveaux.

Nous avons la couche de présentation, la couche de traitement et la couche de stockage :

- La couche présentation permet d'exposer les données collectées et traitées aux analystes à des fin d'analyse et de prise de décision. Au niveau de cette couche, on trouve plusieurs fonctionnalités tel que la création des interfaces graphiques.
- La couche de traitement de données est utilisée pour traiter (normaliser) les données collectées par les agents et de les indexer sous un format universel lisible.
- La couche de stockage consiste en une base de données noSql qui permet de stocker les données indexées par les outils de la couche précédente.

L'architecture inclue les agents Filebeat et Winlogbeat. Filebeat est installé sur tous les serveurs Ubuntu et permet la collecte et la redirection des logs vers l'outils de traitement. Winlogbeat est installé de son coté sur les serveurs Windows pour capturer les évènements de sécurité et les envoyer à l'outil de traitement.

L'outil de traitement (Logstash) convertit les données reçues en des données indexées qui seront stockées dans la BD (Elasticsearch).

Quand un utilisateur veut visualiser les logs, il n'a qu'à se connecter à l'outil de visualisation (Kibana) qui récupère les données depuis la BD et les lui expose.

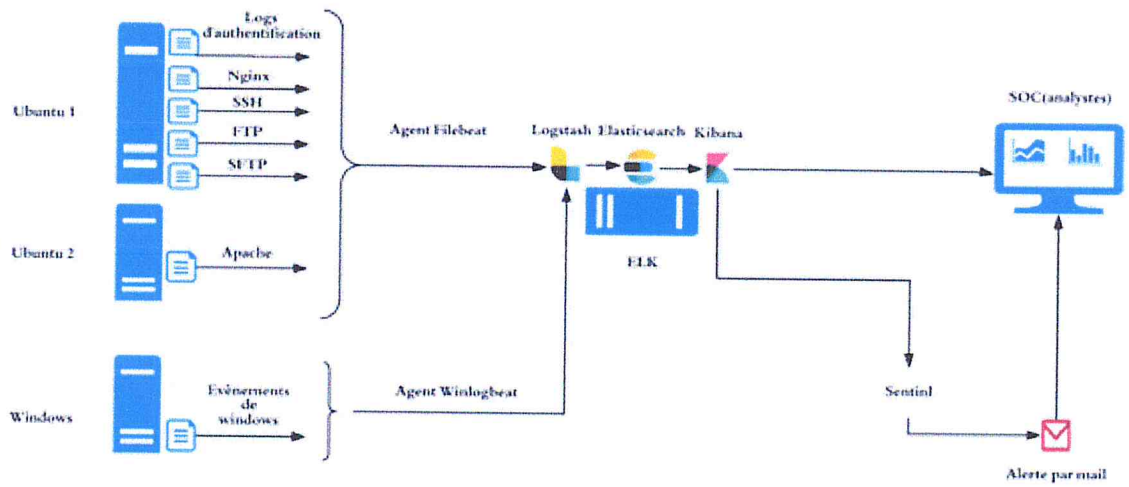


Figure 11 : Architecture de l'environnement.

3.3.Diagramme de séquence

La figure 12 montre un diagramme de séquence qui décrit les différentes interactions entre les composants : Winlogbeat, Filebeat, Logstash, Elasticsearch, Kibana et l'analyste qui utilise le système.

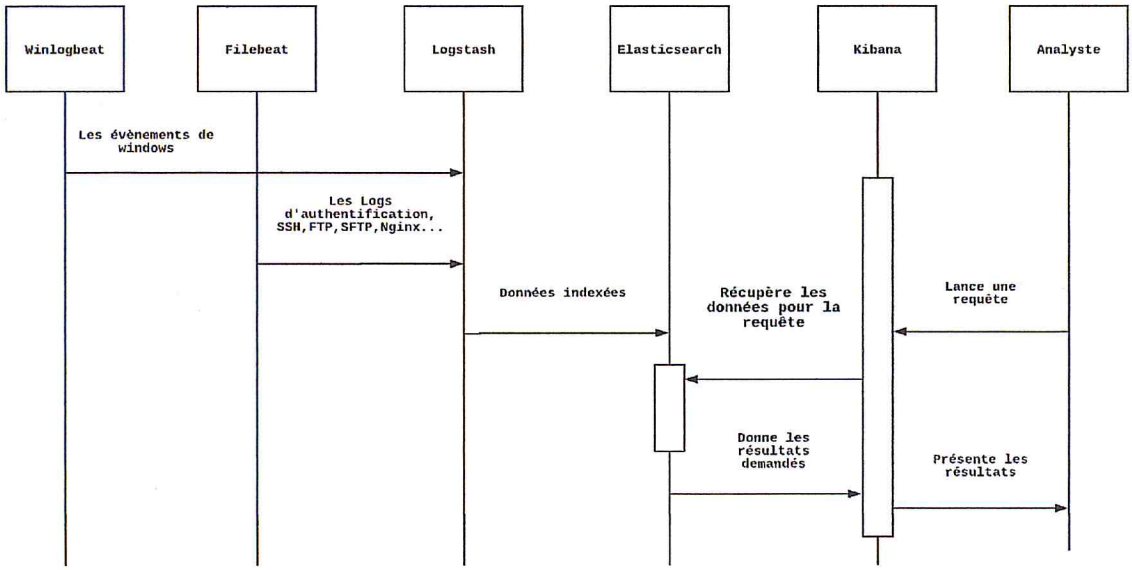


Figure 12 : *Diagramme de séquence.*

3.4.Diagramme de cas d'utilisation (Use-case)

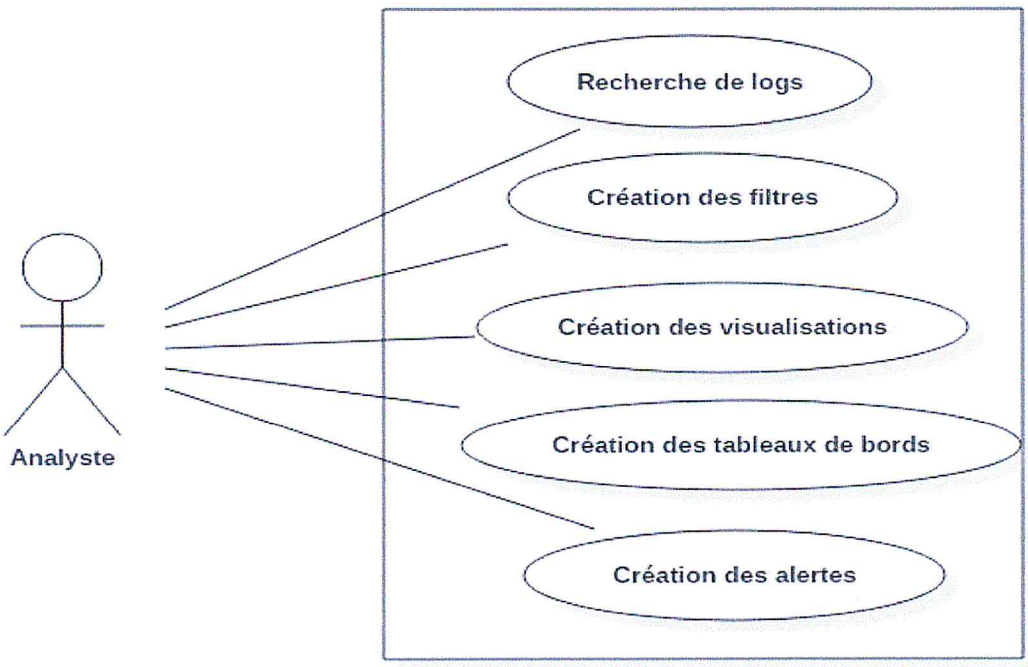


Figure 13 : *Diagramme de cas d'utilisation.*

3.4.1. Cas d'utilisation « Recherche de logs »

Ce Use-case montre comment un analyste peut effectuer des recherches dans le système afin de visualiser les logs significatifs qui sont collectés et indexés.

Nom du cas d'utilisation	Recherche de logs	
Objectif	Permettre à l'utilisateur de faire une recherche afin d'identifier un évènement.	
Acteurs	Analyste	
Précondition	Le système doit avoir reçu des données qui remplissent les critères de la recherche sinon on aura un résultat nul. Les logs doivent être traités et indexés pour une recherche efficace.	
Post-condition	Les résultats de la recherche seront exposés à l'utilisateur sur le navigateur web.	
Scénario	Acteur	Système
	L'utilisateur donne des critères de recherche sur la barre de recherche.	Le système expose les résultats qui remplissent la requête.

Tableau 4 : *Cas d'utilisation "Recherche de logs"*

3.4.2. Cas d'utilisation « Création des filtres (recherches enregistrées) »

Ce use-case décrit comment un utilisateur peut créer des filtres, ces filtres facilitent la réutilisation des recherches les plus communes.

Nom du cas d'utilisation	Création des filtres	
Objectif	Permettre à l'utilisateur de créer une recherche et l'enregistrer sous forme de filtre.	
Acteurs	Analyste	
Précondition	L'utilisateur doit donner des critères de recherche valides.	
Scénario	Acteur	Système
	L'utilisateur donne des critères de recherche sur la barre de recherche et les enregistre sous forme de filtre.	Le système stocke le filtre pour un accès facile à l'utilisateur.

Tableau 5 : *Cas d'utilisation « Création des filtres (recherches enregistrées) »*

3.4.3. Cas d'utilisation « Création des visualisations »

Ce cas d'utilisation explique la création des visualisations :

Nom du cas d'utilisation	Création de visualisations
Objectif	Permettre à l'utilisateur de créer des graphes et des tableaux basées sur les recherches enregistrées ou sur des agrégations des champs des indexes.
Acteurs	Analyste
Précondition	L'utilisateur a besoin d'avoir des recherches et des filtres valides.

Tableau 6 : *Cas d'utilisation « Création des visualisations »*

3.4.4. Cas d'utilisation « Création de Dashboard (tableau de bord) »

Ce use-case explique comment se fait la création des dashboard.

Nom du cas d'utilisation	Création de dashboard
Objectif	Permettre à l'utilisateur d'ajouter une collection de visualisation reliées dans un dashboard vide.
Acteurs	Analyste
Précondition	L'utilisateur Doit avoir créé les visualisations
Post-condition	Les résultats de la recherche seront exposés à l'utilisateur sur le navigateur web.

Tableau 7 : *Cas d'utilisation « Création de Dashboard (tableau de bord) »*

3.4.5. Cas d'utilisation « Création des Alertes »

Ce cas d'utilisation décrit la création des alertes selon des règles prédéfinies :

Nom du cas d'utilisation	Création des alertes	
Objectif	Permettre à l'utilisateur de créer une alerte en utilisant un observateur selon une règle de corrélation prédéfinie.	
Acteurs	Analyste	
Précondition	Les données doivent être traitées et indexées et les règles de corrélation claires et définie.	
Post-condition	En cas de satisfaction des règles et des conditions des observateurs, une alerte est envoyé via mail à l'utilisateur	
Scénario	Acteur	Système
	L'utilisateur crée un observateur qui traduit la règle de corrélation.	Le système envoie une alerte via mail en cas de satisfaction de la règle.

Tableau 8 : *Cas d'utilisation « Création des Alertes »*

3.5.Diagramme de contexte

Un diagramme de contexte est un composant de la modélisation fonctionnelle qui se révèle être un outil précieux (Burge, 2011). *La figure 14* ci-dessous représente le diagramme de contexte du système.

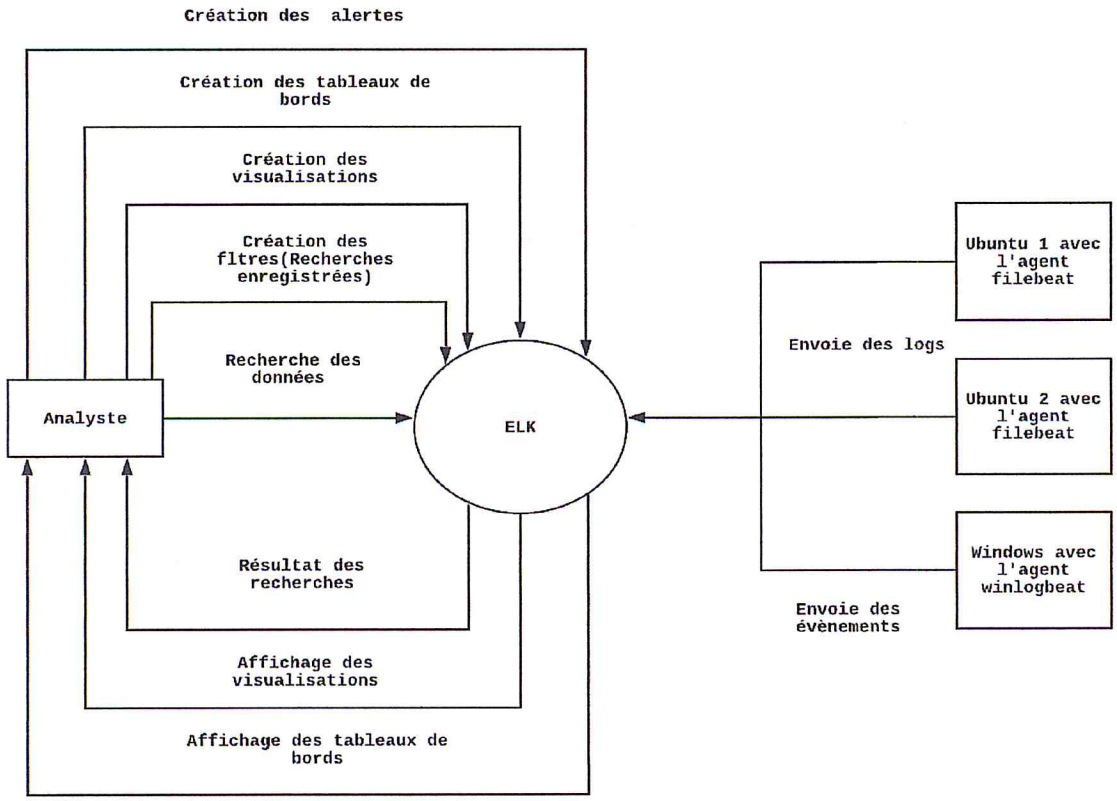


Figure 14 : *Diagramme de contexte.*

3.6.Conclusion

Dans ce chapitre nous avons donné une idée sur la conception et l'architecture de notre système tout en expliquant les différents cas d'utilisation qu'on y trouve : recherche des logs, création de filtres, création des visualisations, des tableaux de bord et des alertes.

Chapitre 4 : Déploiement de la solution de la gestion centralisée de logs

4. Chapitre 4 : Déploiement de la solution de la gestion centralisée de logs

4.1. Introduction

Ce quatrième chapitre décrit l'implémentation de solution de gestion centralisée des logs ainsi que ses caractéristiques et fonctionnalités.

4.2. Environnement du travail

Etant donné que les données de l'entreprise, surtout celles de sécurité, sont confidentielles, notre travail se fera sous une architecture virtuelle : VMware.

VMware : VMware Workstation Pro permet la création d'une ou plusieurs machines virtuelles au sein d'un même système d'exploitation (généralement Windows ou Linux), ceux-ci pouvant être reliés au réseau local avec une adresse IP différente, tout en étant sur la même machine physique (machine existant réellement). Il est possible de faire fonctionner plusieurs machines virtuelles en même temps, la limite correspondant aux performances de l'ordinateur hôte [43].

Notre environnement virtuel comprend :

1. **Serveur de la centralisation et de la gestion des logs :**
   - Serveur Ubuntu 18.10
   - 4GB de RAM
   - 80 GB de disque
   - Elasticsearch 6.6.0
   - Logstash 6.6.0
   - Kibana 6.6.0
2. **Serveur test 1 :**
   - Serveur Ubuntu 18.10
   - 1GB de RAM
   - 10 GB de disque
   - FileBeat
3. **Serveur test 2 :**
   - Serveur Ubuntu 18.10
   - 1 GB de RAM
   - 20 GB de disque
   - Filebeat
4. **Serveur test 3 :**
   - Serveur Windows 10
   - GB de RAM
   - 60 GB de disque
   - WinlogBeat

4.3.La mise en œuvre de la solution

4.3.1. La collecte des logs

4.3.1.1.La collecte des logs sous Ubuntu

Les machines Ubuntu génèrent tous les logs du système automatiquement et les stockent dans des fichiers .log.

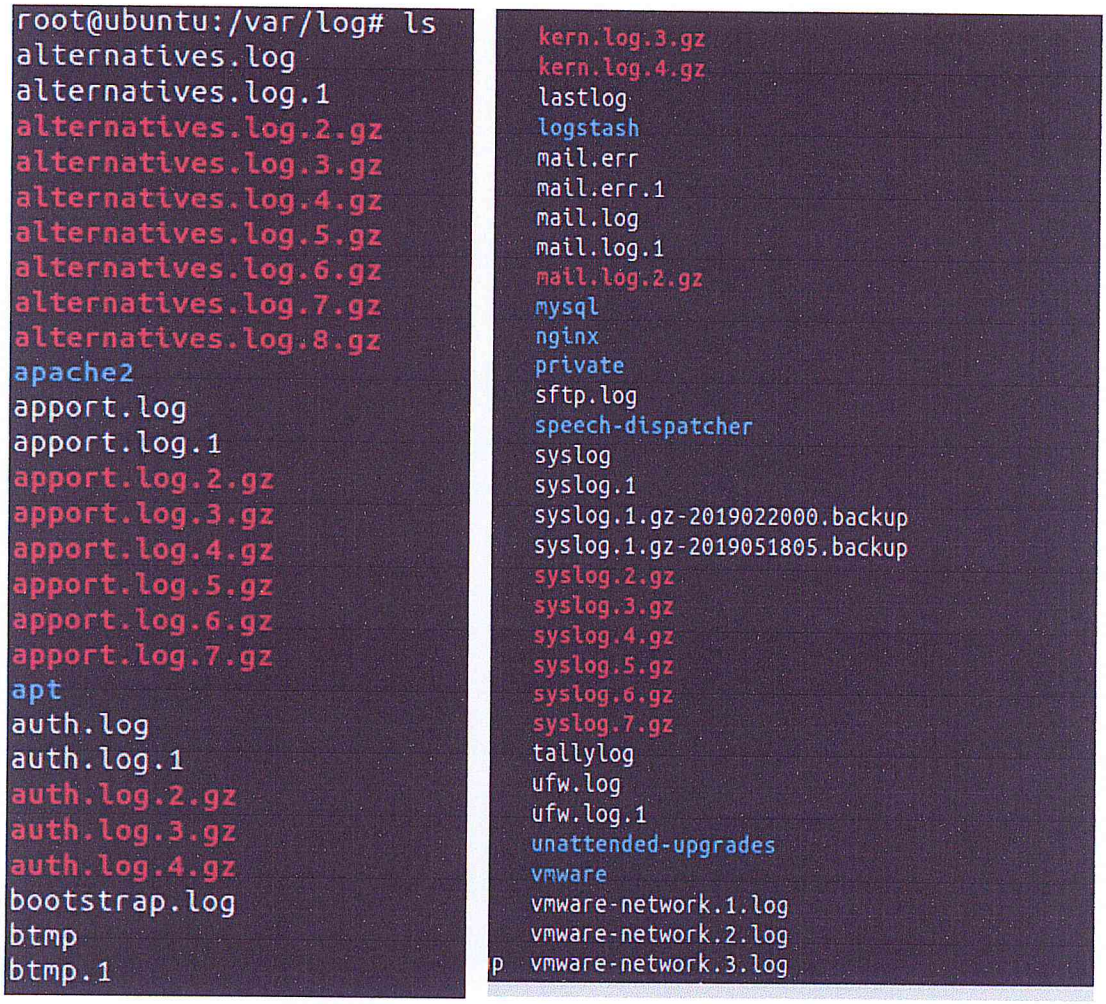


Figure 15 : *Liste des logs générés par Ubuntu.*

Comme nous pouvons le voir, une centaine de fichiers logs existent sur notre système. Par manque de ressources, nous avons choisi de travailler sur quelques types : Logs d'authentification (PAM, Systemd-logind, Sudo[3],SU..), SSH (Secure Shell), FTP (File Transfert Protocol, SFTP (Secure File Transfert Protocol) et Web (Nginx et Apache). Etant donné que les logs sont enregistrés dans des fichiers, nous avons opté pour Filebeat comme agent de collecte en le configurant pour récupérer les logs cités précédemment.

[3] Sudo (abréviation de substitute user do) est une commande permettant à l'administrateur système d'accorder à certains utilisateurs (ou groupes d'utilisateurs) la possibilité de lancer une commande en tant qu'administrateur [46].

Prenons l'exemple de la collecte des logs FTP, le reste des configurations sera donné en Annexe-A.

```
#========================== Filebeat inputs =============================

filebeat.inputs:

# Each - is an input. Most options can be set at the input level, so
# you can use different inputs for various configurations.
# Below are the input specific configurations.

#- type: log

- type: log
  paths:
    - "/var/log/vsftpd.log"
  fields:
    ftp: true
```

Figure 16 : *Collecte des logs Ftp.*

```
output.logstash:
  # The Logstash hosts
  hosts: ["localhost:5044"]
```

Figure 17 : *Les configurations de la sortie de Filebeat.*

```
Sun Jul  7 16:46:35 2019 [pid 8779] [ftpuser] OK LOGIN: Client "::ffff:192.168.102.130"
```

Figure 18 : *Log en format brut.*

La figure 18 présente un exemple d'un log en format brut collecté par filebeat et envoyé à logstash.

4.3.1.2.La collecte des logs sous Windows

La machine virtuelle Windows10 a été configurée pour capturer les évènements d'audit en activant d'abord l'audit des évènements de sécurité au niveau de l'outil des politiques de sécurité locale.

Ceci a été fait pour permettre au serveur Windows de générer des évènements lorsque des actions spécifiques sont faites par l'utilisateur. Ces actions sont : des connexions aux comptes, la gestion des comptes, les changements de politique et les évènements de système comme on le constate sur *la figure 19.*

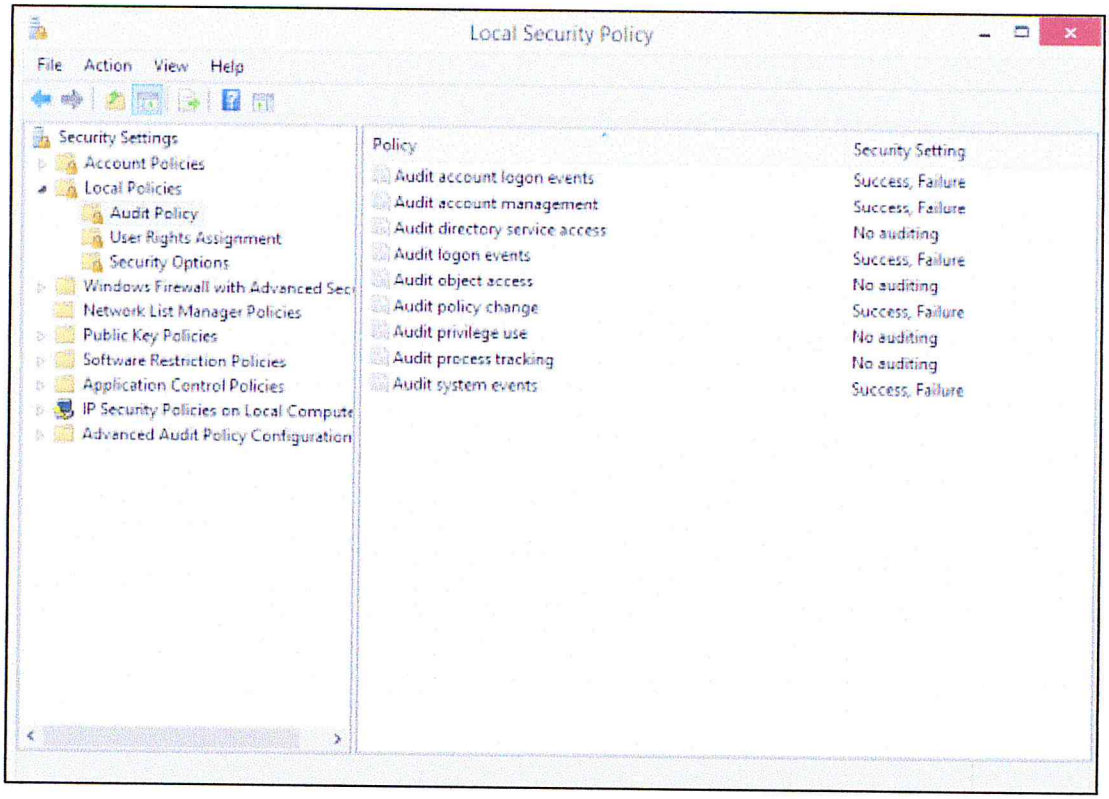


Figure 19: *Politique de la sécurité locale.*

Nous installons et configurons l'agent Winlogbeat pour la capture des évènements audités. Winlogbeat a été configuré pour la collecte des évènements de système, d'application et de sécurité de Windows et de les envoyer à logstash pour leur futur traitement ou indexation directe.

```
#========================== Winlogbeat specific options
============================

# event_logs specifies a list of event logs to monitor as well as
any
# accompanying options. The YAML data type of event_logs is a
list of
# dictionaries.
#
# The supported keys are name (required), tags, fields,
fields_under_root,
# forwarded, ignore_older, level, event_id, provider, and
include_xml. Please
# visit the documentation for the complete details of each
option.
# https://go.es.io/WinlogbeatConfig
winlogbeat.event_logs:
  - name: Application
    ignore_older: 72h
  - name: Security
  - name: System
```

Figure 20 : *Collecte des évènements windows.*

```
output.logstash:
  # The Logstash hosts
  hosts: ["192.168.102.130:5043"]
```

Figure 21 : *Configurer logstash comme sortie de winlogbeat.*

La figure 22 présente l'exemple d'évènement Windows collecté par Winlogbeat et envoyé à logstash

```
Échec d'ouverture de session d'un compte.

Sujet :
        ID de sécurité :                S-1-5-18
        Nom du compte :         DESKTOP-LB9F3V8$
        Domaine du compte :             WORKGROUP
        ID d'ouverture de session :             0x3E7

Type d'ouverture de session :                   2

Compte pour lequel l'ouverture de session a échoué :
        ID de sécurité :                S-1-0-0
        Nom du compte :         Asma
        Domaine du compte :             DESKTOP-LB9F3V8

Informations sur l'échec :
        Raison de l'échec :             Nom d'utilisateur inconnu ou mot de passe inc
orrect.
        État :                  0xC000006D
        Sous-état :             0xC000006A

Informations sur le processus :
        ID du processus de l'appelant : 0x39c
        Nom du processus de l'appelant :        C:\Windows\System32\svchost.exe

Informations sur le réseau :
        Nom de la station de travail :  DESKTOP-LB9F3V8
        Adresse du réseau source :      127.0.0.1
        Port source :           0

Informations détaillées sur l'authentification :
        Processus d'ouverture de session :              User32
        Package d'authentification :    Negotiate
        Services en transit :   -
        Nom du package (NTLM uniquement) :      -
        Longueur de clé :               0
```

Figure 22 : *Evènement Windows collecté par Winlogbeat.*

4.3.2. Traitement des fichiers logs :

Les composants individuels de la centralisation de logs ont été installé sur Ubuntu 18.10, voir l'Annexe A. Comme chaque composant est indépendant, nous les configurons de manière à ce que la communication entre eux soit possible via des ports spécifiques :

- Elasticsearch est configuré pour être accessible via le port 9200
- Logatsh est configuré pour être accessible via les ports 5044 et 5043
- Kibana est configuré pour être accessible via le port 5601

➢ La configuration de logstash

La configuration de logstash est faite en ayant des sources de données en entrée (Input) : tout log capturer depuis les ports 5044 et 5043 qui sont les ports auxquels Filebeat et Winlogbeat envoient les données.

```
input {
      beats {
        port => 5044
            }
      }
```

Figure 23 : Input capturant les données envoyées par Filebeat.

```
input {
  beats {
      port => 5043
      "type" => "winlogbeat"
        }
      }
```

Figure 24 : *Input capturant les données envoyées par Winlogbeat.*

Le filtre pour la configuration est un parseur JSON qui va capturer toutes les données en entrée et les normalisera en utilisant des patterns integrés ou de nouveau patterns créés par l'analyste. Tous les parseurs que nous avons utilisés existent dans la documentation de logstash mis à part ceux des logs FTP et SFTP que nous avons ajouté nous-même.

Ftp :

```
grok {
        patterns_dir   => "${LL_PATTERN_DIR:/etc/logstash/patterns.d}"
        match => { "message" => ['%{CONNECT}','%{RESPONS}','%{ANON}','%{FILE}']}

          }
```

Figure 25 : *Le grok pattern de ftp.*

SFTP :

```
if [program] == "sftp-server" {
grok {
        patterns_dir   => "${LL_PATTERN_DIR:/etc/logstash/patterns.d}"
        match => { "message" => ['%{CLOSE}','%{FILE}','%{DIRECTORY}','%{MODIF}','%{SESSION}','%{STATUS}']}

          }
```

Figure 26 : *Le grok pattern de sftp*

Nous donnerons les détails sur les patterns en Annexe-B.

Une fois les logs normalisés, les données traitées sont dirigées vers la configuration de sortie (output). Elles sont transférées à Elasticsearch qui les reçoit sur le port 9200.

FTP

```
if [source] == "/var/log/vsftpd.log" {
elasticsearch {
    hosts => ["localhost:9200"]
    manage_template => false
    index => "ftp-%{+YYYY.MM.dd}"
  }
}
```

Figure 27 : *Indexation de ftp.*

SFTP

```
   if [program] in ["sshd","sftp-server","su","sudo","systemd-logind"] {
    elasticsearch {
    hosts => ["localhost:9200"]
    manage_template => false
    index => "%{program}-%{+YYYY.MM.dd}"
  }}
```

Figure 28 : *Output de sftp.*

```
      output {
      if [type] == "winlogbeat" {
       elasticsearch {
          hosts => ["localhost:9200"]
          manage_template => false
          index => "%{[@metadata][beat]}-%{+YYYY.MM.dd}"
        }}

      }
```

Figure 29 : *Output des évènements Windows.*

4.3.3. Stockage des indexes

Comme mentionner précédemment, nous avons opté pour Elasticsearch comme BD. C'est-à-dire qu'après le traitement, les logs sont indexés et stockés, où chaque type de log aura son propre modèle d'indice (Index Pattern).

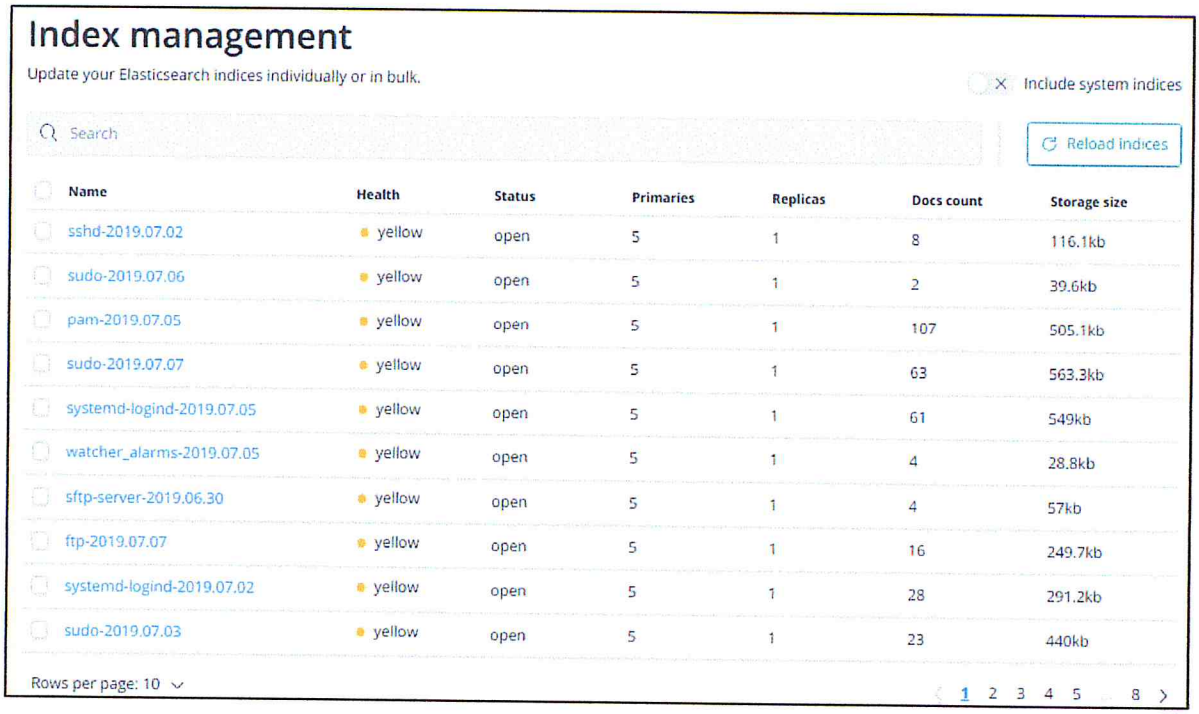


Figure 30 : *Exemple de la liste des index.*

Les données sont stockées dans des indices au niveau de Elasticsearch. Pour pouvoir les exploiter à des fins de supervision, il faut créer un modèle d'indice (Index Pattern) sur Kibana qui regroupera les indices contenant le même type de logs.

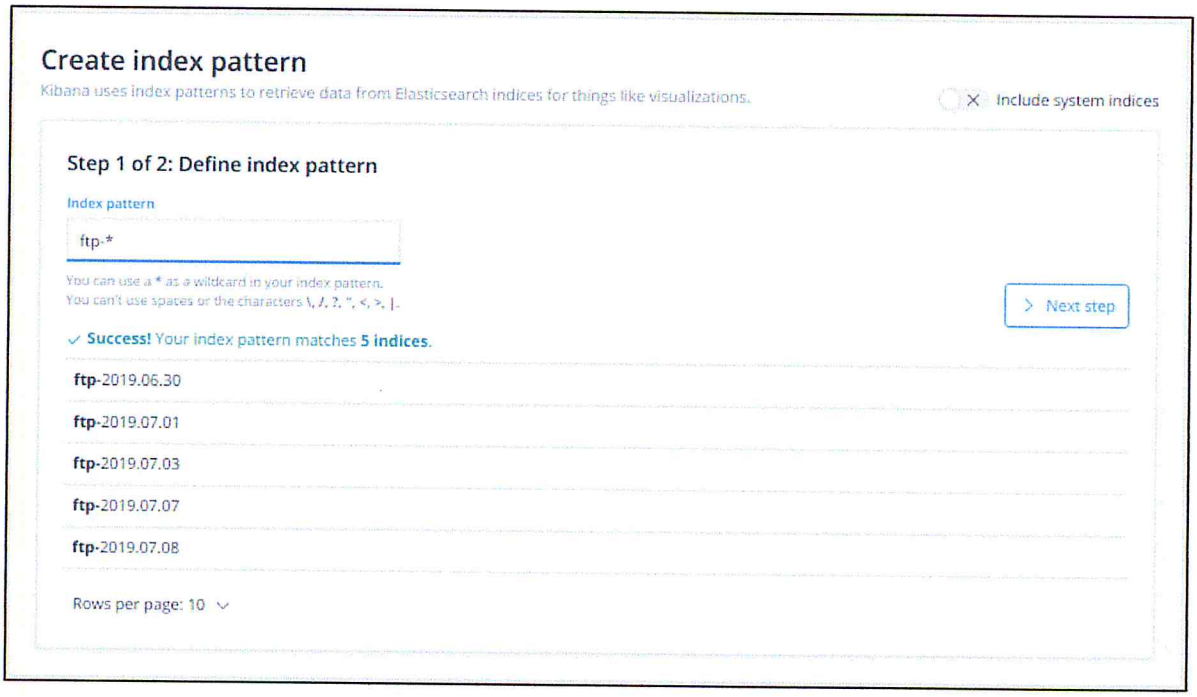


Figure 31 : *Création d'un index pattern « ftp »*

Le modèle d'indice « ftp » va regrouper tous les index « Ftp » afin les rendre exploitables par kibana : Affichage de logs, recherches, filtres, visualisations, dashboard…

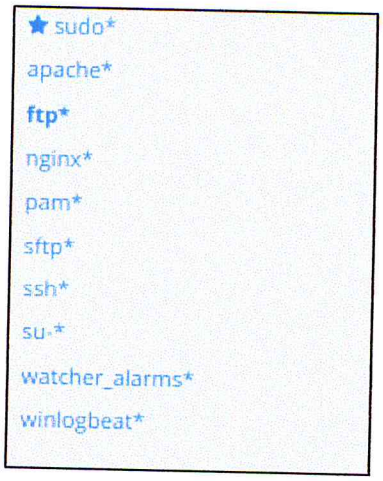


Figure 32:*Liste des « index patterns » de la solution.*

Après la création de tous nos « index patterns », la supervision des logs est possible.

4.3.4. Supervision

4.3.4.1.Recherche de logs

Quand un utilisateur accède au système depuis le navigateur il est directement mis face à l'interface Kibana où il peut effectuer des recherches spécifiques par texte en utilisant des requêtes DSL (Domain Specific Language). *La figure 33* montre un exemple de recherche de l'utilisateur 'ftpuser ' sur la machine Ubuntu.

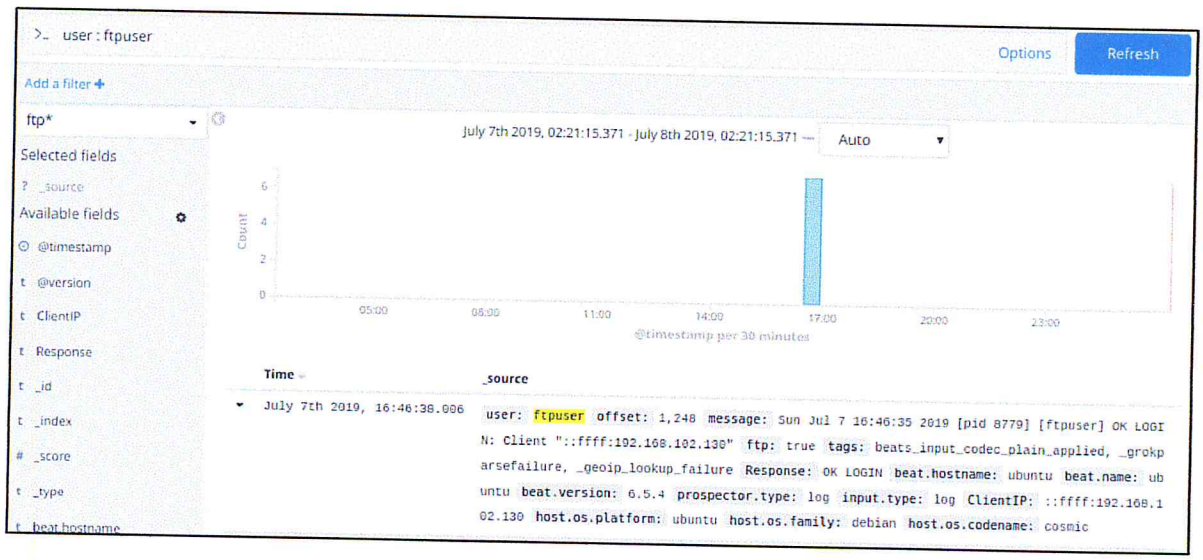


Figure 33 : *Recherche de l'utilisateur 'Ftpuser'.*

4.3.4.2.La création des recherches enregistrées (Filtres)

Les recherches enregistrées sont des requêtes qui sont enregistrées sur Kibana pour une réutilisation dans le futur. Elles sont généralement conçues pour les utiliser dans la création des visualisations qui ont besoin de filtrer des données. *La figure 34* montre la liste de nos recherches enregistrées.

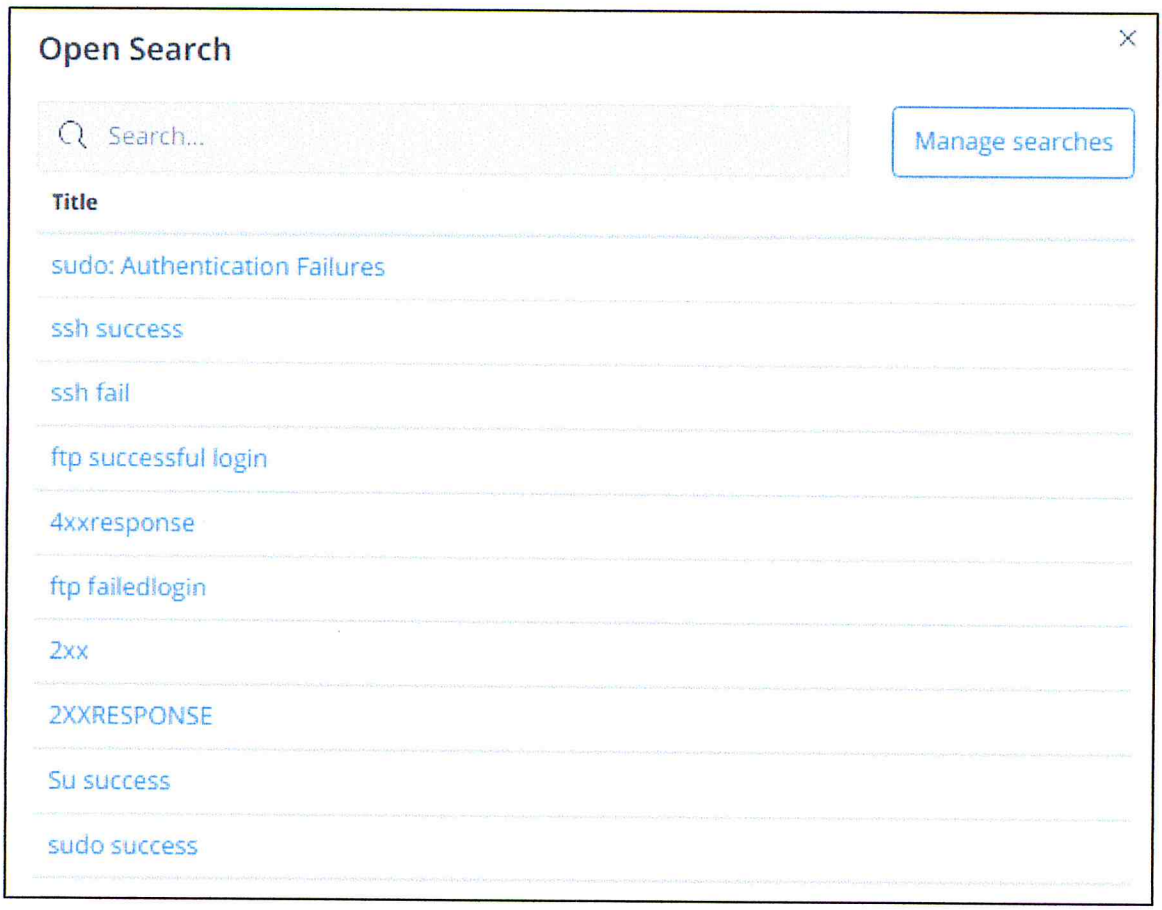


Figure 34: *Liste des recherches enregistrées.*

Une fois la recherche enregistrée, il suffit de la relancer pour avoir les résultats voulus. *La figure 35* en donne un exemple.

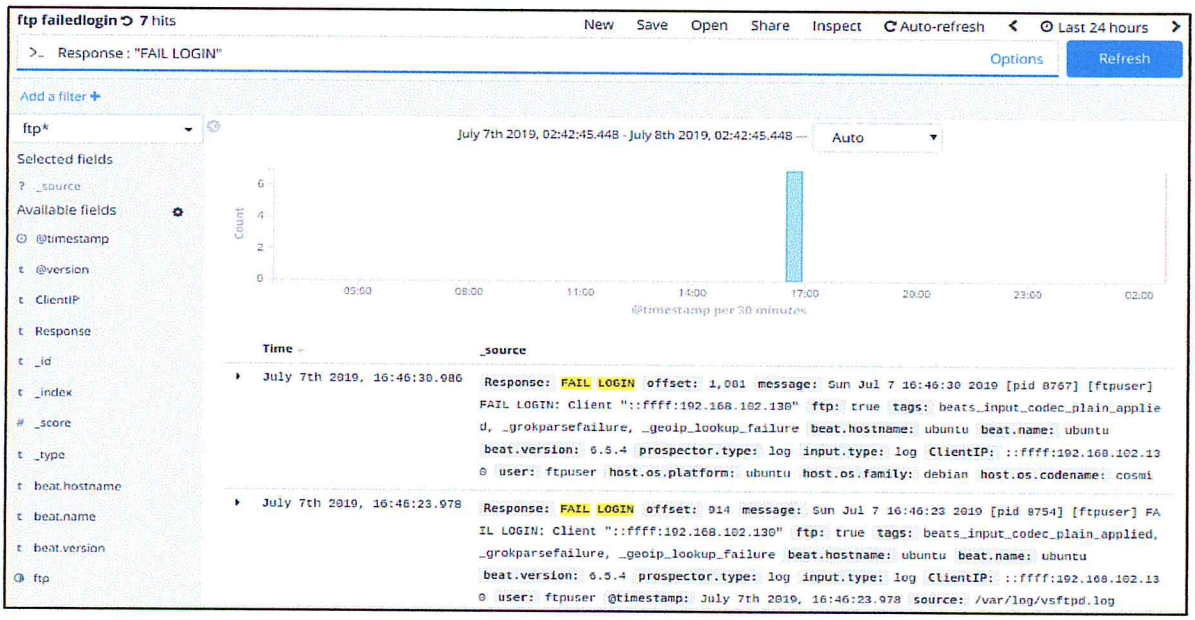


Figure 35 : *Recherche Enregistrée lancée pour le terme 'Fail Login'.*

4.3.4.3.La création de visualisation

Les visualisations aident les analystes de sécurité à avoir une bonne vue des évènements collectés par le système. Elles sont créées à partir des recherches enregistrées ou des modèles d'indices en utilisant des agrégations des champs obtenus par les filtres de logstash et la normalisation. *La figure 36* montre un exemple de visualisation (tableaux) contenant les top 10 tentatives de connexion échouées.

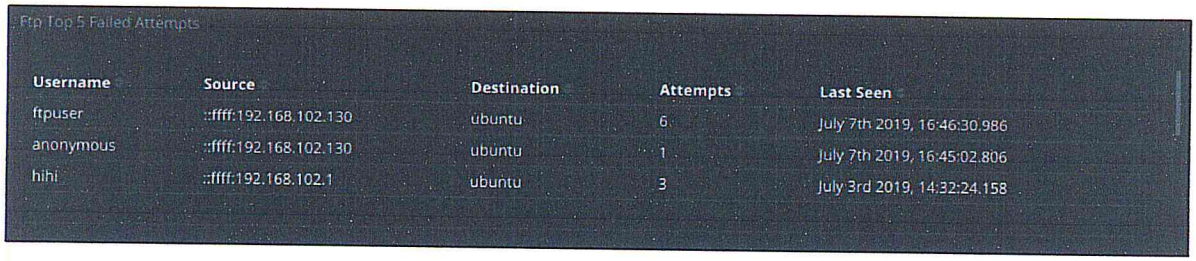


Figure 36 : *Visualisation.*

4.3.4.4.Création des tableaux de bord

Un dashboard est une collection de plusieurs visualisations ayant des points d'intérêts communs.

- Le tableau de bord de FTP

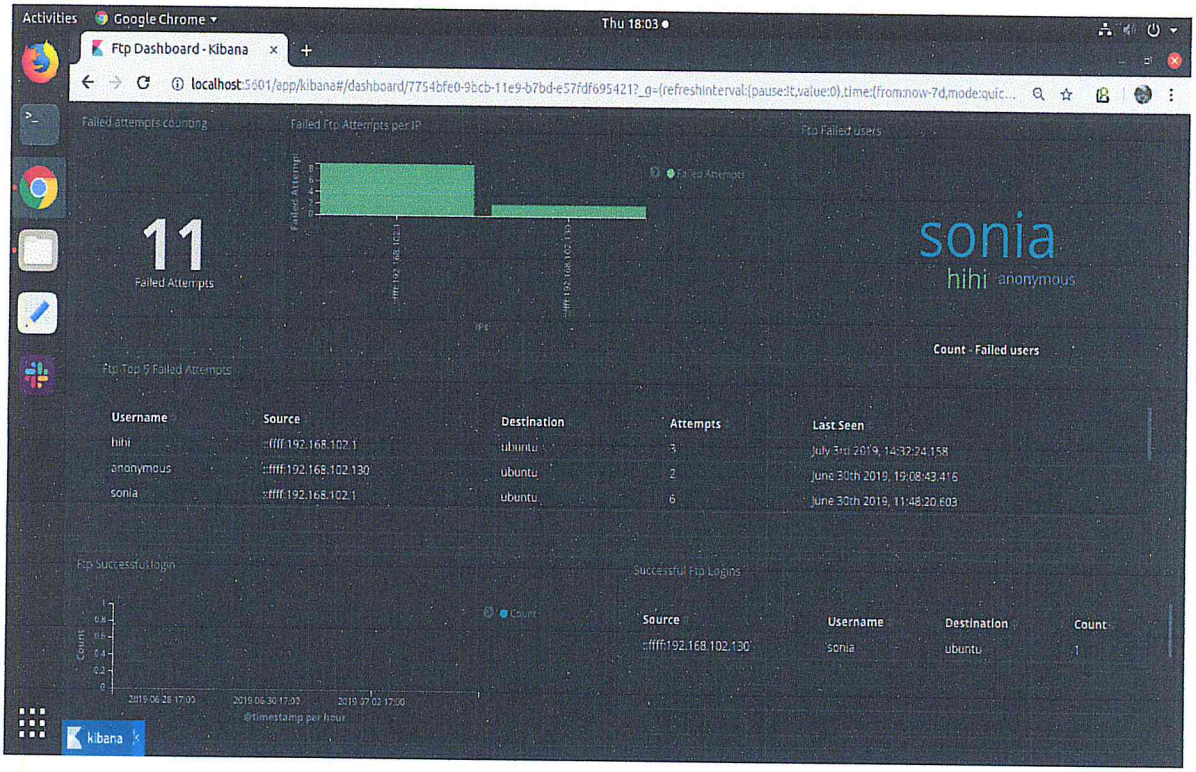


Figure 37 : *Tableau de bord FTP.*

- Le tableau de bord de Windows

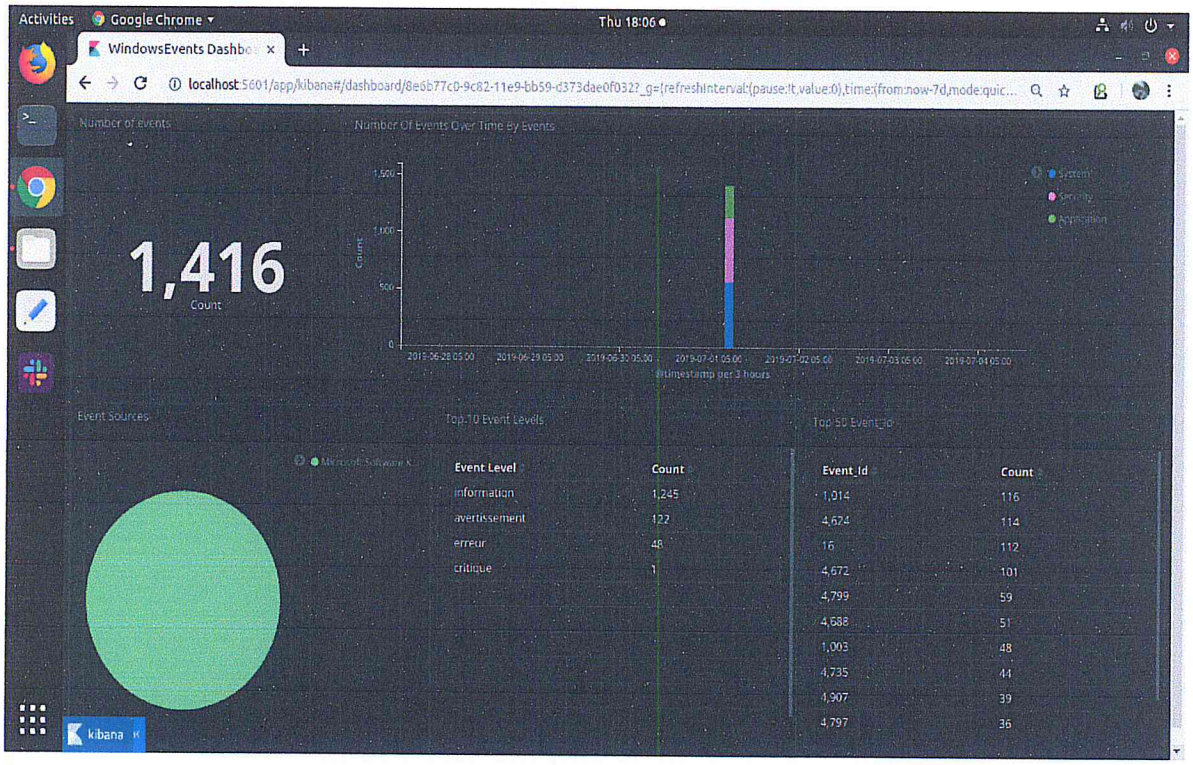


Figure 38 : *Tableau de bord de windows..*

4.3.4.5.Corrélation

i. Création des règles de corrélation

Une fois les tableaux de bords faits, c'est au tour de la corrélation des évènements.

Les règles de corrélation permettent de faire un lien entre différents logs pour détecter une tentative d'attaque ou un incident.

Voici les règles que nous avons défini selon les besoins du service de sécurité de l'entreprise :

a) **Authentification SSH**

- Échecs de 5 tentatives de connexion suivis par succès par une même adresse IP vers une même destination.
- Échecs de multiples tentatives de connexion par une même adresse IP vers une même destination.
- Échec de connexion.

b) **Authentification FTP**

- Échecs de 5 tentatives de connexion suivis par succès par une même adresse IP sur un même utilisateur.
- Détection de multiple tentatives de connexion échoué.
- Détection de connexion anonyme.

c) Authentification Windows

- Détection des événements d'échec d'authentification répétés au minimum 3 fois sur un seul hôte Windows en moins d'une minute.
- Force brute : alerte lorsqu'un nom d'hôte a plus de 3 échecs de connexion suivis par un succès en 3 minutes.

d) Authentification Sudo

- Détecter l'échecs de multiples tentatives de connexion.

e) Injection de SQL

- Détection des caractères spéciaux tel que (', ",--, #....) ou les mots clé (select, union, exec....) dans les champs de saisie.

f) Injection XSS

- Détection des caractères spéciaux ayant des significations particulières dans l'interpréteur visé.

4.3.4.6.Création des alertes

Une alerte est une notification qui est envoyée lors de la rencontre de certaines conditions : les règles de corrélation.

Nous créons nos alertes en utilisant l'outil Sentinl. Sur Sentinl, ou même sur x-Pack d'Elastic une alerte est programmée en utilisant un observateur « Watcher ».

La création du watcher consiste à se baser sur une règle de corrélation, la traduire en requête DSL et de filtrer les résultats selon une condition. Si la condition est satisfaite, l'action de création d'alerte sera exécutée.

Les watchers créés :

- **« SSH : One failed login » une tentative de connexion SSH échoué**

  Dès que ce watcher détecte le terme 'invalid_user' provenant d'une seule adresse IP source avec un seul utilisateur dans les 3 dernières minutes, il lance une alerte : **« SSH : One failed login ».**

- **« SSH : failed login » plusieurs échecs de connexion SSH**

  Dès que le watcher détecte le terme 'invalid_user' au moins 2 fois provenant d'une seule adresse IP source avec un seul utilisateur et accédant au même hôte de destination dans les 3 dernières minutes, il lance une alerte : **« SSH : failed login »**

- **« SSH brute force success » force brute réussie de SSH**

  Dès que le watcher détecte le terme 'fail' au moins 5 fois suivi d'un 'success' au moins une fois provenant d'une seule adresse IP source, avec le même utilisateur ssh et accédant

au même hôte de destination dans les 3 dernières minutes, il lance une alerte : **« SSH brute force success »**

- **« Sudo incorrect attempts » tentatives incorrectes de sudo**

  Dès la détection 'incorrect password attempts' dans le champ'sudo_message' au moins 1 fois survenu d'une même adresse IP source, accédant la même hôte de destination durant les dernières 3 minutes, le watcher lance une alerte : **« Sudo incorrect attempts»**

- **« SQL injection attempt » Injection SQL**

  Dès que le watcher détecte les caractères d'échappements et les mot-clé dans le champs 'request' durant les dernière 48h, il lance une alerte : **« SQL injection attempt »**

- **« XSS attempt » tentative de XSS**

  Dès que le watcher détecte les balises et les tags dans le champs 'request' durant les dernière 48h, il lance une alerte : **« XSS attempt »**

- **« FTP anonymous login » connexion anonyme de FTP**

  Dès l'identification des connexions réussies d'un utilisateur anonyme avec la même adresse IP source dans les dernières 2 minutes, le watcher lance une alerte : **« FTP anonymous login »**

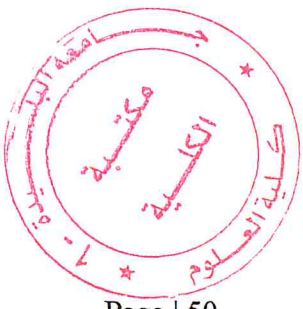
- **« FTP : failed logins » connexions échouées de FTP**

  Dès la détection du terme 'FAIL LOGIN' au moins 5 fois survenu d'une même adresse IP source et d'un même utilisateur, durant les dernières 2 minutes, le watcher lance une alerte : **« FTP : failed logins »**

- **« FTP brute force success » force brute réussie de FTP**

  Dès que le watcher détecte 'FAIL LOGIN' au moins 5 fois suivi d'un 'OK LOGIN' au moins une fois, provenant d'un même utilisateur et d'une même adresse IP, il lance une alerte : **« FTP brute force success »** comme *la figure 39* le montre.

  Le reste de nos watchers seront mis en Annexe C.



```
  "actions": {
    "email_admin": {
      "email": {
        "to": "soniahdjs@gmail.com",
        "from": "soniahdjs@gmail.com",
        "subject": "SENTINL ALARM {{ payload._id }}",
        "priority": "high",
        "body": "Series Alarm "
      }
    }
  },
  "input": {
    "search": {
      "request": {
        "index": "<ftp-{now/d}>",
        "body": {
          "size": 0,
          "query": {
            "bool": {
              "filter": [
                {
                  "range": {
                    "@timestamp": {
                      "from": "now-3m"
                    }
                  }
                },
                {
                  "terms": {
                    "Response.keyword": [
                      "FAIL LOGIN",
                      "OK LOGIN"
                    ]
                  }
                }
              ]
            }
          },
          "aggs": {
            "failed_ip": {
              "terms": {
                "field": "ClientIP.keyword"
              },
              "aggs": {
                "users": {
                  "terms": {
                    "field": "user.keyword"
                  },
                  "aggs": {
                    "login": {
                      "terms": {
                        "field": "Response.keyword"
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    },
    "condition": {
      "script": {
        "script": "payload.aggregations.failed_ip && payload.aggregations.failed_ip.buckets.length > 0"
      },
      "compare": {
        "payload.aggregations.failed_ip.buckets.0.users.buckets.0.doc_count": {
          "gte": 5
        }
      },

    "compare_array": {
      "payload.aggregations.failed_ip.buckets.0.users.buckets.1.doc_count": {
        "gte": 1
      }
    }
  },
  "trigger": {
    "schedule": {
      "later": "every 2 minutes"
    }
  },
  "disable": true,
  "report": false,
  "title": "FTP : Brute force success",
  "save_payload": false,
  "spy": false,
  "impersonate": false
}
```

Figure 39 : *Watcher « FTP : Brute force success »*

- **« Windows failed logins » Tentative de connexion windows échouées**

  Dès que le watcher détecte 'winlog.event_id = 4625' qui signifie 'connexion échouée' et 'Echec de l'audit' en même temps au moins 3 fois durant les dernières 24h, il lance une alerte **« Windows failed logins »**

- **« Windows succsseful brute force » force brute réussie de windows**

  Dès que le watcher détecte le terme 'authentication_failure' au moins 5 fois suivi d'un 'authentication_success'' au moins une fois, provenant d'un même utilisateur dans les 8 dernières heures, il lance une alerte : **« Windows succsseful brute force »**

```
"query": {
  "bool": {
    "filter": [
      {
        "range": {
          "@timestamp": {
            "from": "now-8h"
          }
        }
      },
      {
        "terms": {
          "event.type.keyword": [
            "authentication_failure",
            "authentication_success"
          ]
        }
      },
      {
        "match": {
          "winlog.channel": "Security"
        }
      }
    ]
  }
},
"aggs": {
  "Account": {
    "terms": {
      "field": "user.name.keyword"
    },
    "aggs": {
      "event": {
        "terms": {
          "field": "event.type.keyword"
        }
      }
    }
  }
}
},
"condition": {
  "script": {
    "script": "payload.aggregations.Account && payload.aggregations.Account.buckets.length > 1 "
  },
  "compare": {
    "payload.aggregations.Account.buckets.0.event.buckets.0.doc_count": {
      "gte": 3
    }
  },
  "compare_array": {
    "payload.aggregations.Account.buckets.0.event.buckets.1.doc_count": {
      "gte": 1
    }
  }
```

Figure 40 : *Watcher « Windows succsseful brute force »*

4.4.Conclusion

Dans ce chapitre, nous avons expliqué le déploiement de notre solution de gestion centralisée des fichiers journaux. Nous avons commencé par décrire l'environnement du travail dans lequel la mise en œuvre de la solution a été établie tout en expliquant les étapes suivies : collecte des logs sur ubuntu et windows, le traitement de ces logs avec l'outil logstash en utilisant les parseurs adéquats, le stockage sur la BD elasticsearch et la supervision sur Kibana qui englobe des recherches, des visualisations, des tableaux de bords et des alertes.

Chapitre 5 : Tests et Résultats

5. Chapitre 5 : Tests et résultats

5.1.Introduction

Après l'implémentation de notre solution, nous passons dans ce chapitre aux différents scénarios de tests afin de vérifier l'efficacité et le bon fonctionnement de notre système.

5.2.Outils utilisés

Pour effectuer nos tests nous utilisons :

- Terminal GNU/Ubuntu
- **Damn Vulnerable Web App** (DVWA) qui est une application Web PHP / MySQL extrêmement vulnérable. Ses principaux objectifs sont d'aider les professionnels de la sécurité à tester leurs compétences et leurs outils dans un environnement juridique, à aider les développeurs Web à mieux comprendre les processus de sécurisation des applications Web et à aider les enseignants / étudiants à enseigner / apprendre la sécurité des applications Web dans un environnement de salle de classe.
- **MobaXterm** qui est une boîte à outils ultime pour l'informatique à distance. Dans une seule application Windows, il fournit une multitude de fonctions sur mesure pour les programmeurs, les webmasters, les administrateurs informatiques et à peu près tous les utilisateurs qui doivent gérer leurs tâches distantes de manière plus simple. MobaXterm fournit tous les outils réseau distants importants (SSH, X11, RDP, VNC, FTP, MOSH, ...) et les commandes Unix (bash, ls, cat, sed, grep, awk, rsync, ...) au bureau Windows.
- **Machine virtuelle Kali/Linux** : Kali Linux est une distribution GNU/Linux basée sur Debian. L'objectif de Kali Linux est de fournir une distribution regroupant l'ensemble des outils nécessaires aux tests de sécurité d'un système d'information,

5.3.Scénarios de tests

Afin de tester notre solution, nous avons tenté quelques attaques et observé le comportement du système face à elles.

5.3.1. Tests sur le système d'exploitation Ubuntu

5.3.1.1.Détection d'attaques sur le service FTP

- **Détection de connexion anonyme**

Un serveur hôte qui héberge un service FTP peut dans certains cas autoriser les connexions anonymes. Ceci ne veut pas dire que tout accès anonyme est légitime et sans risque. D'où, être

au courant des connexions anonymes est important pour la sécurisation des transferts de fichiers via Ftp.

Pour pouvoir effectuer une connexion anonyme, nous devons d'abord autoriser l'accès au service pour les utilisateurs anonymes. *La figure 41* montre comment cette autorisation se fait.

Open vsftpd.conf /etc

```
# Example config file /etc/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
#
# Run standalone?  vsftpd can run either from an inetd or as a standalone
# daemon started from an initscript.
listen=NO
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=YES
#
```

Figure 41: *Autorisation d'accès aux utilisateurs anonymes.*

Une fois la connexion anonyme autorisée, nous accédons au serveur ftp installé sur notre machine Ubuntu dont l'adresse IP est 192.168.102.130 avec le nom d'utilisateur « anonymous » et un mot de passe vide, le tout se fait en utilisant l'outil MobaXtrem. *La figure 42* représente cette étape.

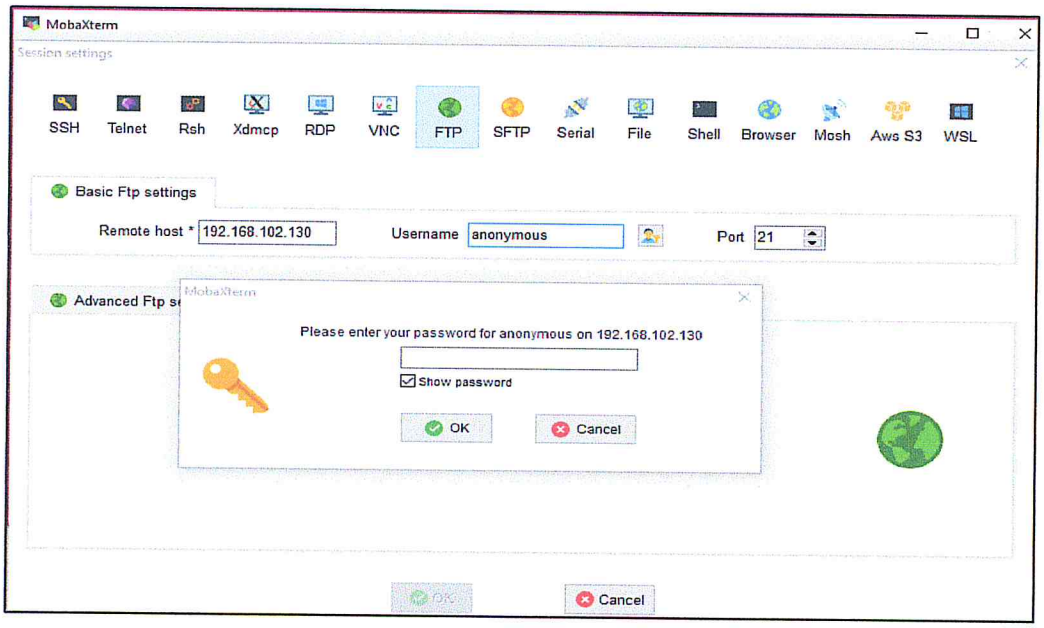


Figure 42: *Entrée d'un mot de passe vide.*

La connexion anonyme donne un accès limité aux fichiers du système sauf si l'administrateur change les politiques d'accès. *La figure 43* décrit une connexion anonyme réussie.

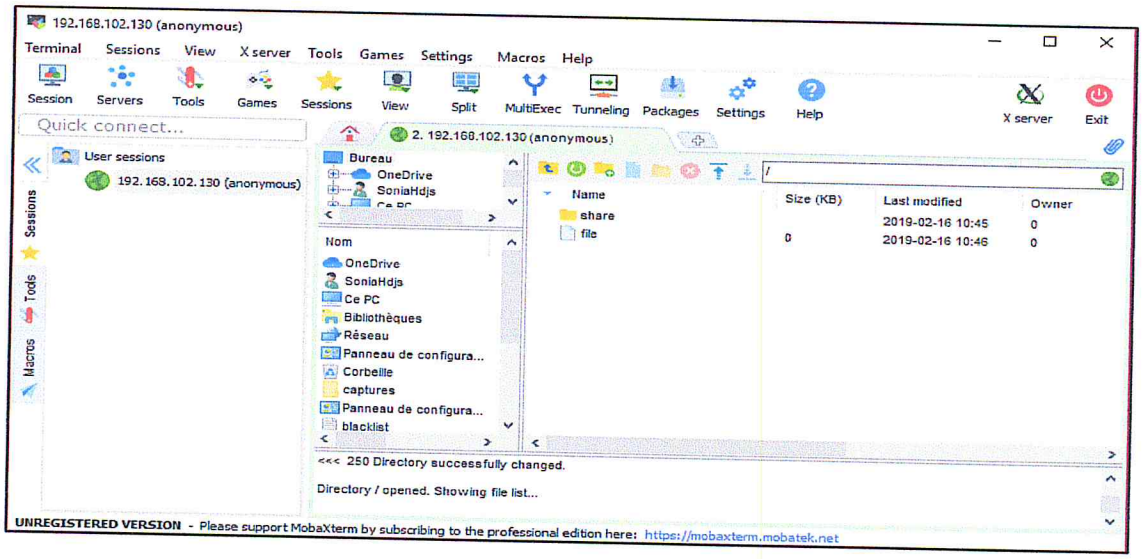


Figure 43 : *Connexion anonyme réussie.*

Le log généré par cet évènement apparait sur la plateforme Kibana sous sa forme normalisée, un utilisateur anonyme prend automatiquement le nom d'utilisateur « ftp » comme le montre *la figure 44.*

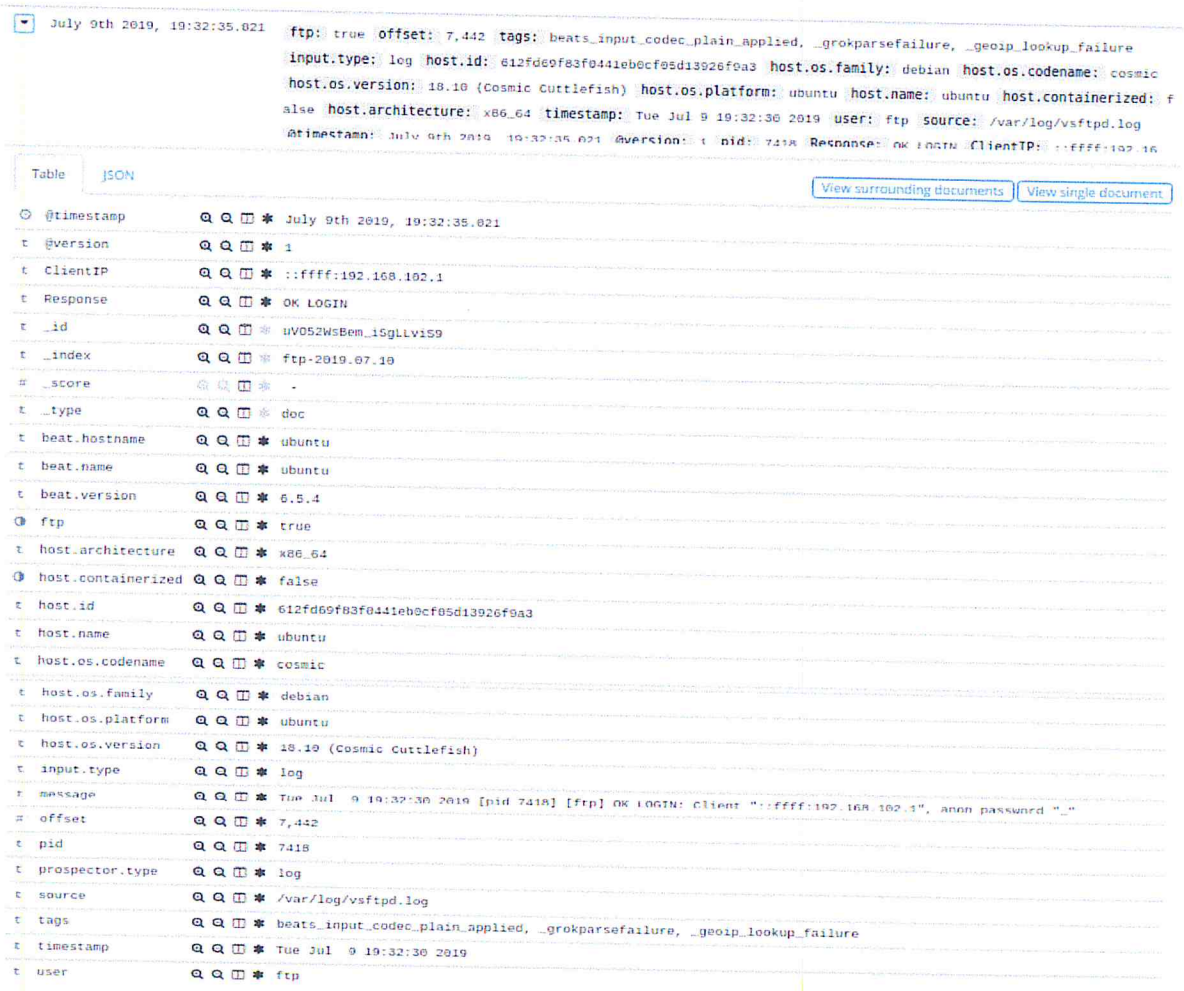


Figure 44 : *Log normalisé d'une connexion anonyme.*

Cette connexion est détectée par le watcher « FTP : Anonymous Login » présenté en Annexe-C et une alerte est tout de suite générée pour la notification de cette connexion. Les figures suivantes représentent l'alerte générée ainsi que le mail envoyé pour la notifié.

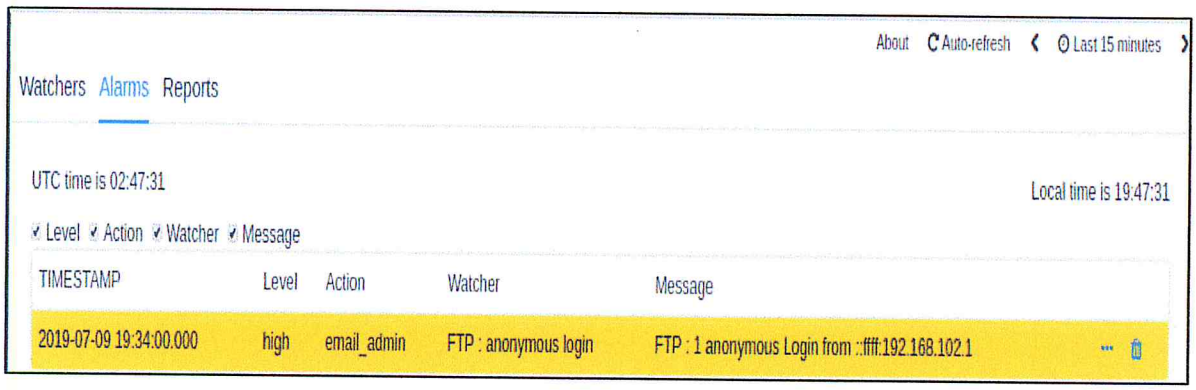


Figure 45 : *Alerte « FTP : Anonymous Login » .*

Figure 46 : *Email Correspondant à l'alerte.*

- **Détection de multiples connexions FTP échouées**

Le test de détection de multiples tentatives échouées de connexion à un serveur ftp consiste à essayer d'accéder aux fichiers de la machine cible en utilisant un nom d'utilisateur existant avec la saisie d'au moins 3 faux mots de passe afin que l'alerte soit générée.

Pour simuler ce test, nous utiliseront l'outil MobaXtrem pour tenter d'accéder aux fichiers de la machine ayant l'adresse IP 192.168.102.130 par le nom d'utilisateur « ftpuser » comme le montre *la figure 47.*

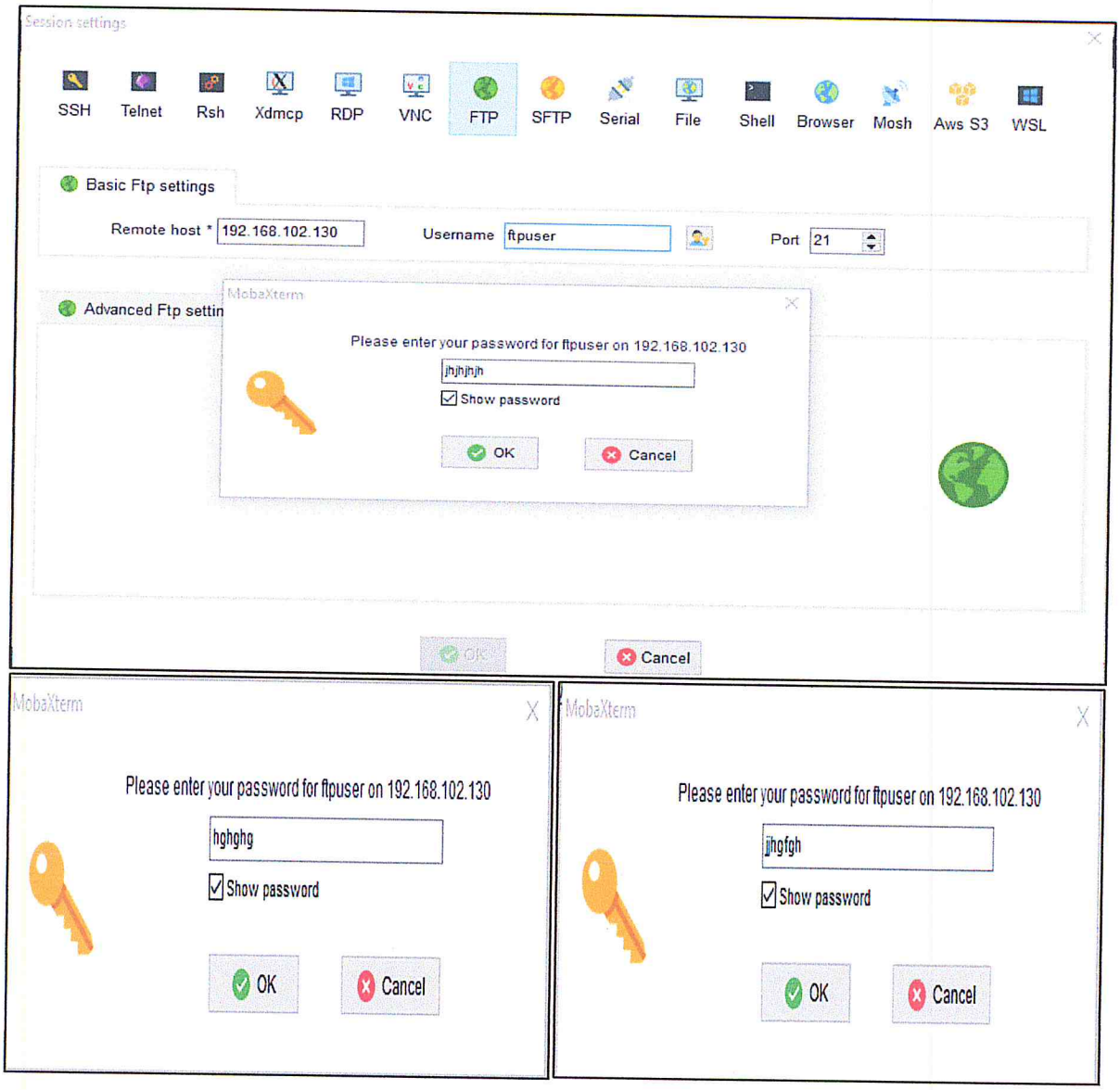


Figure 47 : *La saisie de 3 faux mots de passe.*

Trois logs identiques sont générés par cet évènement et apparaissent sur la plateforme Kibana sous leur forme normalisée. Nous pouvons voir sur *la figure 48* que l'utilisateur « ftpuser » n'a pas pu se connecter et a un « Fail login » comme réponse.

Figure 48 : *Log normalisé de multiples échecs de connexion.*

Cette connexion est détectée par le watcher « FTP : Failed Login » présenté en Annexe-C et une alerte est tout de suite générée pour la notification de cette connexion suspecte. Les figures suivantes représentent l'alerte générée ainsi que le mail lui correspondant.

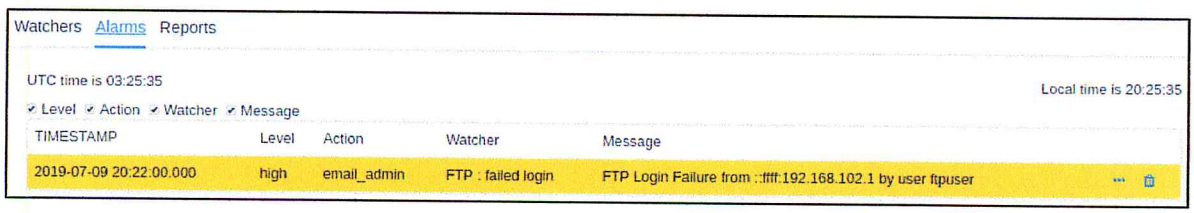


Figure 49 : *Alerte « FTP : Failed Login ».*

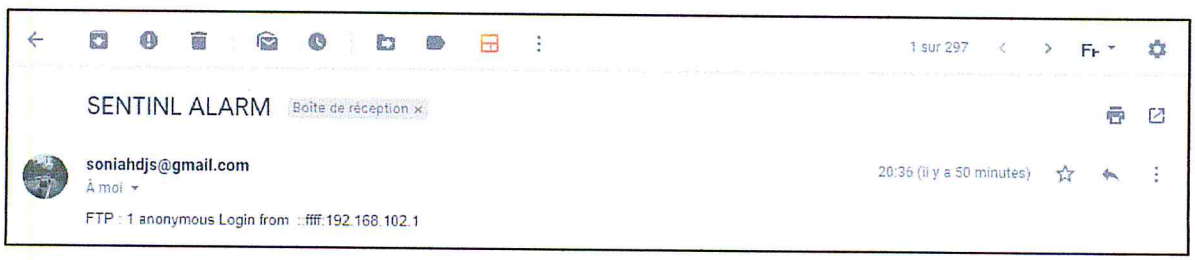


Figure 50 : *Email Correspondant à l'alerte.*

➢ Détection force brute sur service FTP

Ce test sera effectué à partir d'un serveur Kali Linux.

Pour ce test, nous avons lancé une attaque par force brute ftp sous Hydra[4] pour cracker le mot de passe d'une session distante de notre serveur Ubuntu dont nous connaissons le login, pour ce faire nous avons ajouté les mots de passe possibles au dictionnaire nommé FastTrack.txt qui contient les mots de passe possibles pour hacker cette session (une chaine de caractères par ligne).

On lance l'attaque en tapant la commande décrite dans la figure suivante :

```
root@kali:/usr/share/wordlists# hydra -t 1 -l ftpuser -P fasttrack.txt 192.168.1
02.130 ftp
Hydra v8.8 (c) 2019 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-07-10 10:02:
15
[DATA] max 1 task per 1 server, overall 1 task, 223 login tries (l:1/p:223), ~22
3 tries per task
[DATA] attacking ftp://192.168.102.130:21/
[21][ftp] host: 192.168.102.130   login: ftpuser   password: 16121995
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-07-10 10:02:
31
root@kali:/usr/share/wordlists#
```

Figure 51 : *Exécution de l'attaque force brute sur FTP.*

5 logs ont été générés suite à 5 tentatives échouées, le 6 ème mot de passe était le bon et a généré un log de connexion aboutie. *Les figure 52 et 53* représentent les 5 logs d'authentifications échouées et l'authentification réussie respectivement.

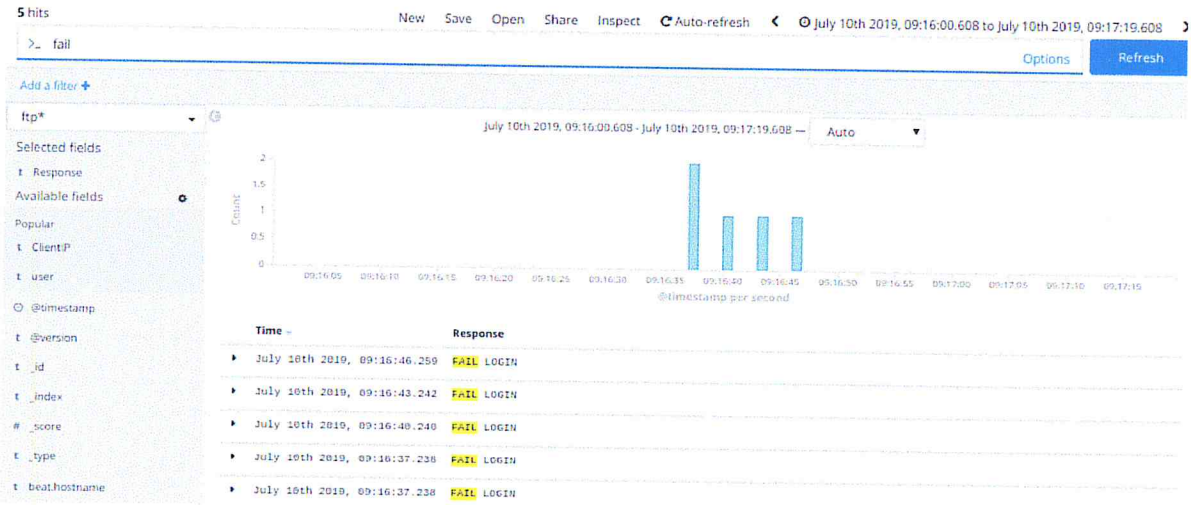


Figure 52 : *Génération de 5 logs suite à 5 tentatives échouées.*

[4] Hydra : un logiciel de craquage de mots de passe capable de cracker des mots de passe sur un grand nombre de protocoles ou de bases de données différentes : HTTP, HTTPS, SSH, FTP, MYSQL etc…

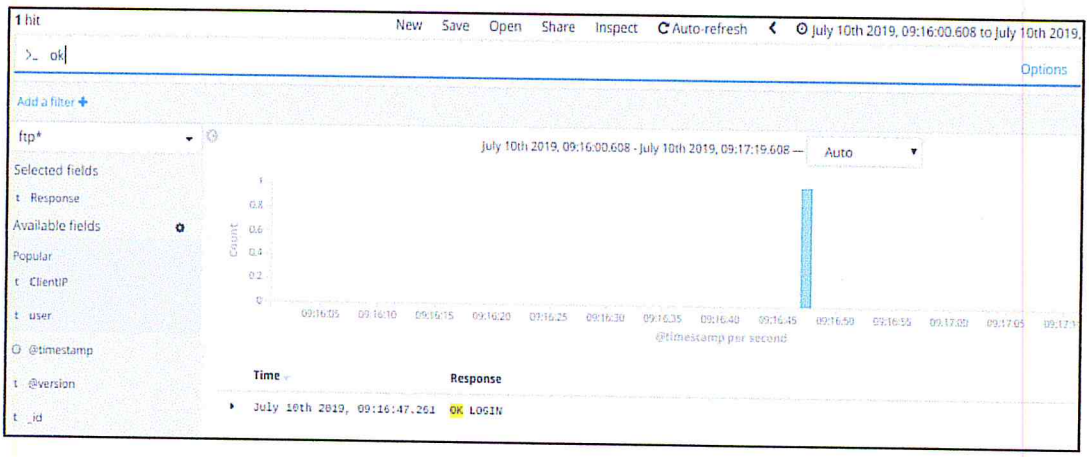


Figure 53 : *Log généré suite à une authentification réussie.*

Le watcher « Ftp Brute Force Success » a detecté cette attaque et généré l'alerte correspondante en notifiant l'utilisateur par email. *Les figures 54 et 55* représentent l'alerte envoyée.

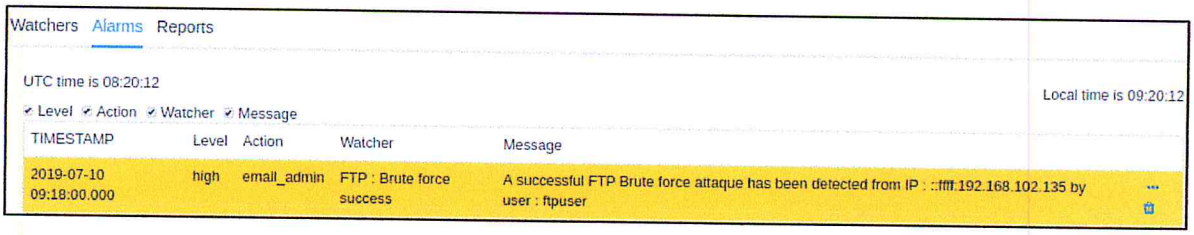


Figure 54 : *Alerte « Ftp Brute Force Success ».*

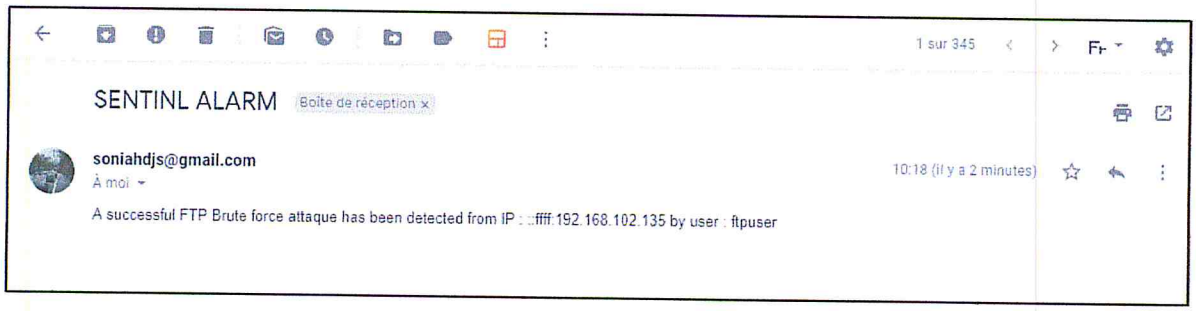


Figure 55 : *Email Correspondant à l'alerte.*

5.3.1.2.Détection d'attaques sur le service SSH

- **Détection d'un échec de connexion**

La détection d'un échec de connexion ssh permet à un analyste de prévenir une tentative d'attaque pour accéder au système via le protocole SSH.

Pour simuler ce test, nous utiliseront deux machines Ubuntu. La machine à l'adresse IP 192.168.102.129 sera utilisée pour accéder au Ubuntu18 dont l'adresse IP est 192.168.102.130 par l'utilisateur « sonia ».

La figure 56 représente la demande de connexion avec un faux mot de passe.

```
serveur2@ubuntu:~$ ssh sonia@192.168.102.130
sonia@192.168.102.130's password:
Permission denied, please try again.
sonia@192.168.102.130's password:
```

Figure 56 : *Demande de connexion avec un faux mot de passe.*

Un log est tout de suite généré par cet évènement et apparait dans Kibana sous sa forme normalisée comme le montre *la figure 57.*

@timestamp	July 9th 2019, 22:13:52.591
@version	1
_id	It1N2msB2mTL6JwLZcXb
_index	sshd-2019.07.10
_score	-
_type	doc
beat.hostname	ubuntu
beat.name	ubuntu
beat.version	6.5.4
host.architecture	x86_64
host.containerized	false
host.id	612fd69f83f0441eb0cf05d13926f9a3
host.name	ubuntu
host.os.codename	cosmic
host.os.family	debian
host.os.platform	ubuntu
host.os.version	18.10 (Cosmic Cuttlefish)
input.type	log
logsource	ubuntu
message	Failed password for sonia from 192.168.102.129 port 52436 ssh2
offset	69,956
pid	4338
program	sshd
prospector.type	log
source	/var/log/auth.log
ssh_authmethod	password
ssh_authresult	fail
ssh_client_ip	192.168.102.129
ssh_client_port	52436
ssh_failreason	invalid_user
ssh_protocol	ssh2
ssh_user	sonia
tags	beats_input_codec_plain_applied, _geoip_lookup_failure, SSH_AUTHFAIL_WRONGUSER
timestamp	Jul 9 22:13:51

Figure 57: *Log normalisé d'une tentative de connexion échouée.*

Cet échec de connexion est détecté par le watcher « SSH : One failed login » présenté en Annexe-C et une alerte est tout de suite générée pour la notification de cette connexion. Les figures suivantes représentent l'alerte générée ainsi que le mail lui correspondant.

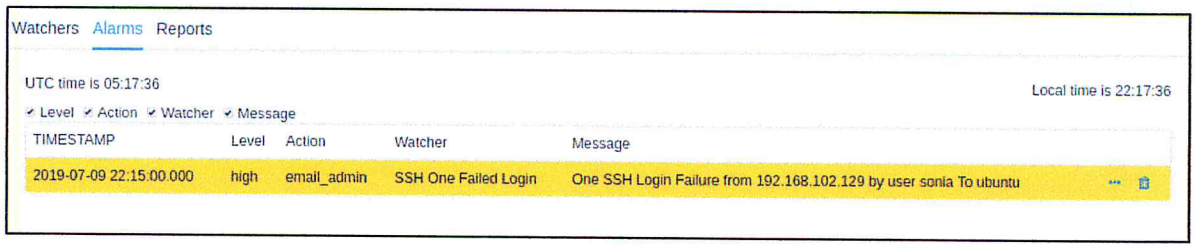


Figure 58 : *Alerte « SSH : One failed login ».*

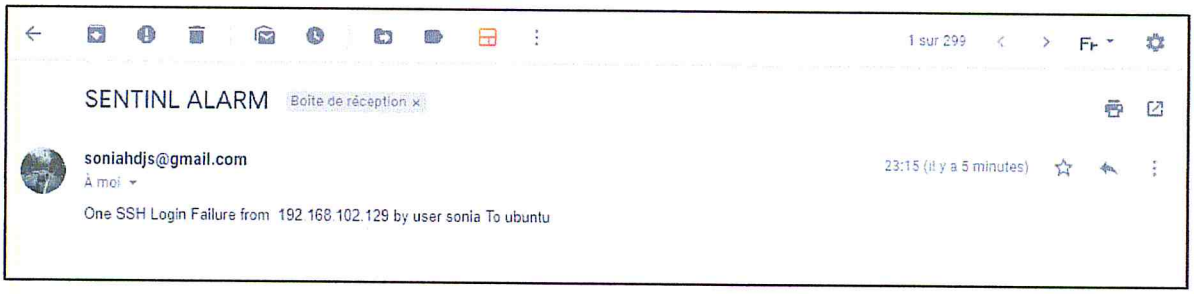


Figure 59 : *Email Correspondant à l'alerte.*

➢ **Détection de multiples échecs de connexion SSH**

Le test de détection de multiples tentatives échouées de connexion ssh consiste à détecter plus d'un échec de connexion sur le protocole ssh.

Nous continuons notre test en utilisant le terminal ubuntu précédent avec le même scénario mais en répétant l'action de demande de connexion avec 3 faux mots de passe successifs. *La figure 60* représente ces tentatives de connexion.

```
serveur2@ubuntu:~$ ssh sonia@192.168.102.130
sonia@192.168.102.130's password:
Permission denied, please try again.
sonia@192.168.102.130's password:
Permission denied, please try again.
sonia@192.168.102.130's password:
sonia@192.168.102.130: Permission denied (publickey,password).
```

Figure 60: *Demande de connexion avec 3 faux mots de passe successifs.*

Trois logs identiques sont générés suite aux connexions échouées tel que *la figure 61* le montre.

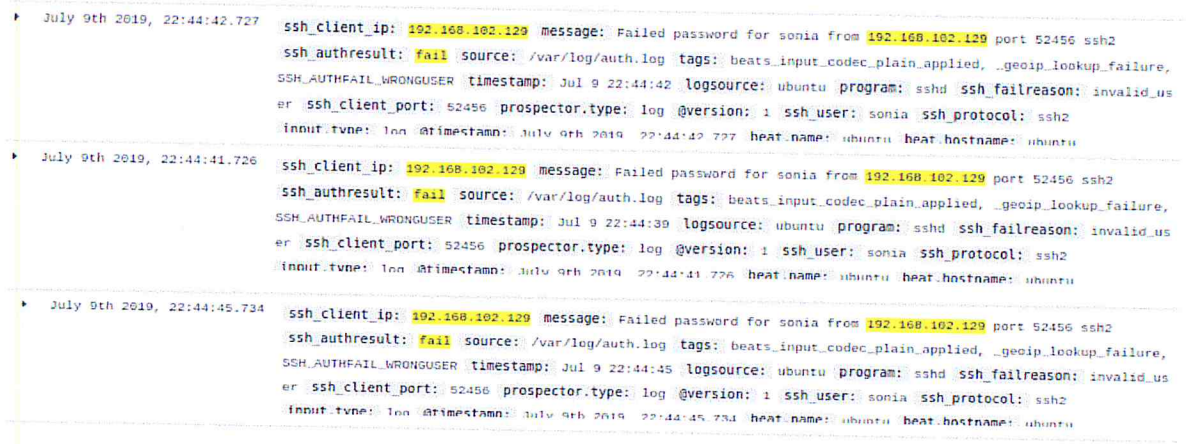


Figure 61: *les 3 logs générés suite à la saisie de 3 faux mot de passe successifs.*

Cet échec connexion est détectée par le watcher « SSH : Failed login » présenté en Annexe-C et une alerte est tout de suite générée pour la notification de cette connexion. Les figures suivantes représentent l'alerte générée ainsi que le mail lui correspondant.

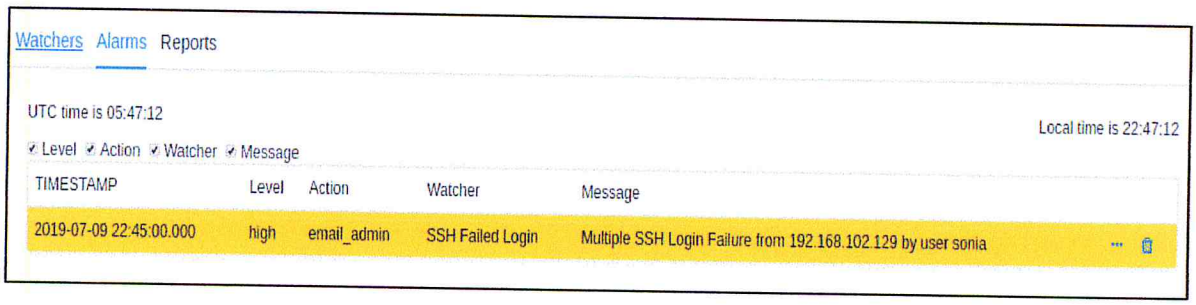


Figure 62 : *Alerte « SSH : Failed login » .*

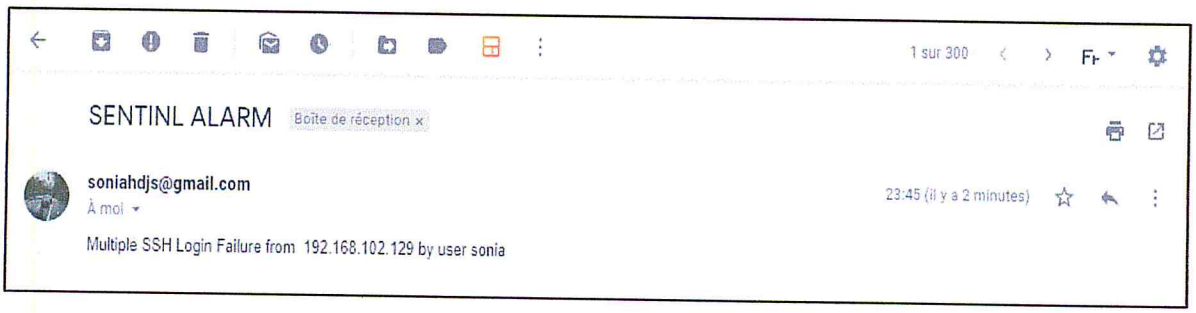


Figure 63 : *Email Correspondant à l'alerte.*

- **Détection de force brute sur le service SSH**

Ce test sera effectué à partir d'un serveur Kali Linux.

Pour ce test, nous avons lancé une attaque par force brute SSH sous Hydra pour cracker le mot de passe d'une session distante de notre serveur Ubuntu dont nous connaissons le login. Pour ce faire nous avons ajouté les mots de passe possible au dictionnaire nommé FastTrack.txt qui

contient les mots de passe possibles pour hacker cette session (une chaine de caractères par ligne).

On lance l'attaque en tapant la commande décrite dans la figure suivante :

```
root@kali:/usr/share/wordlists# hydra 192.168.102.130 ssh -l sonia -P fasttrack.
txt -s 22 vV
Hydra v8.8 (c) 2019 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-07-10 08:58:
16
[WARNING] Many SSH configurations limit the number of parallel tasks, it is reco
mmended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 223 login tries (l:1/p:223),
 ~14 tries per task
[DATA] attacking ssh://192.168.102.130:22/vV
[22][ssh] host: 192.168.102.130   login: sonia   password: 16121995
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 6 final worker threads did not complete u
ntil end.
[ERROR] 6 targets did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-07-10 08:58:
19
root@kali:/usr/share/wordlists#
```

Figure 64: *Exécution de l'attaque force brute sur SSH*

Un log est généré pour chaque échec de connexion, Hydra indique avoir fait 16 tentatives et donc 15 échecs et un succès. *La figure 65* un des logs obtenus montre que 15 logs d'échecs de connexion ont été reçus.

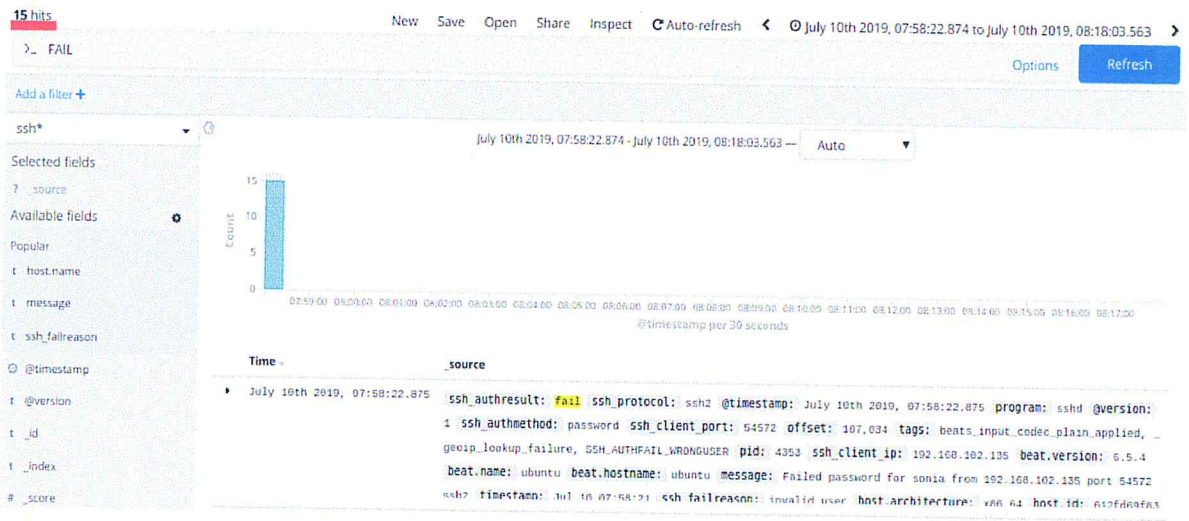


Figure 65 : *Génération de 15 logs d'échecs de connexion.*

La figure 66 quant à elle illustre le log généré par l'aboutissement de la connexion.

@timestamp		July 10th 2019, 07:58:22.873
@version		1
_id		D-at2msBcnEaIqJ9GOuh
_index		sshd-2019.07.10
_score		-
_type		doc
beat.hostname		ubuntu
beat.name		ubuntu
beat.version		6.5.4
host.architecture		x86_64
host.containerized		false
host.id		612fd69f83f0441eb0cf05d13926f9a3
host.name		ubuntu
host.os.codename		cosmic
host.os.family		debian
host.os.platform		ubuntu
host.os.version		18.10 (Cosmic Cuttlefish)
input.type		log
logsource		ubuntu
message		Accepted password for sonia from 192.168.102.135 port 54574 ssh2
offset		102,766
pid		4354
program		sshd
prospector.type		log
source		/var/log/auth.log
ssh_authmethod		password
ssh_authresult		success

Figure 66 : *log généré par l'aboutissement de la connexion.*

Le watcher « SSH Brute Force Success » a détecté l'attaque et l'alerte correspondante a été générée. Les figures 67 et 68 représentent l'alerte.

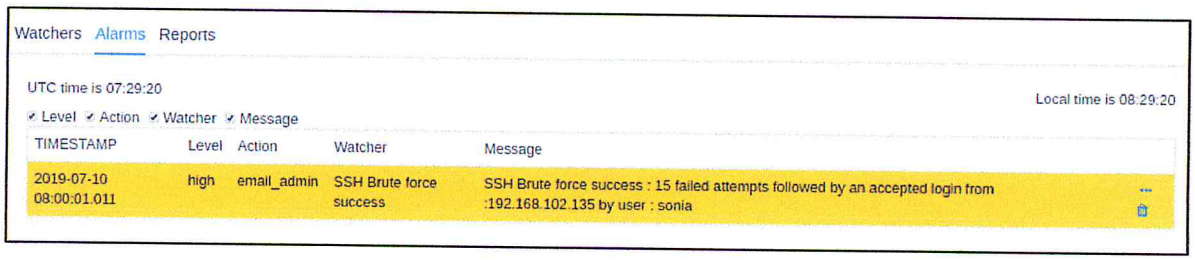


Figure 67 : *Alerte « SSH Brute Force Success » .*

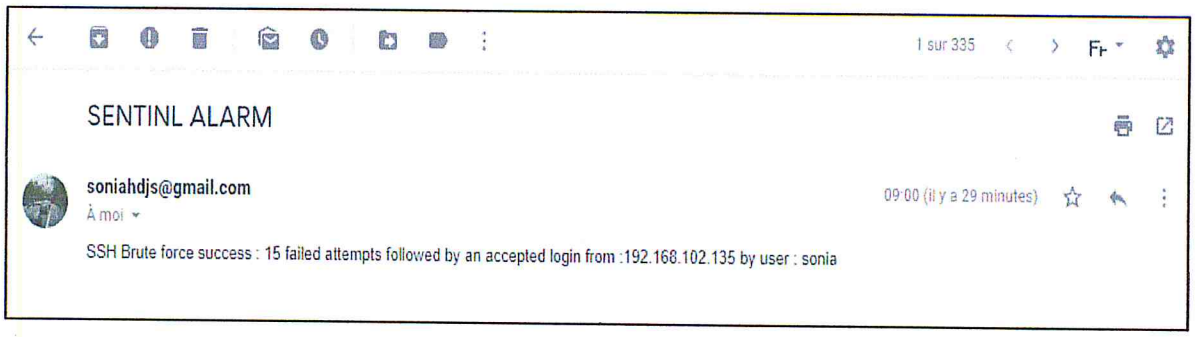


Figure 68 *: Email Correspondant à l'alerte.*

5.3.1.3.Détection de tentatives d'accès aux droits de super utilisateur (Sudo)

- **Détection de multiples tentatives d'accès échouées**

Avoir le droit de super utilisateur veut dire avoir tous les droits sur la machine. Pour éviter qu'un simple utilisateur ne puisse accéder à cela, la détection des tentatives échouées de la commande sudo est primordial.

Le test se fait sur une seule machine comme la commande est utilisée localement. 3 faux mots de passe seront saisi pour la génération d'une alerte. *La figure 69* représente un essai de connexion avec un faux mot de passe.

```
sonia@ubuntu:~$ sudo gedit /etc/logstash/conf.d/ftp.conf
[sudo] password for sonia:
Sorry, try again.
[sudo] password for sonia:
Sorry, try again.
[sudo] password for sonia:
sudo: 3 incorrect password attempts
```

Figure 69 : *Essai de connexion avec un faux mot de passe.*

Un log est immédiatement généré traité et transféré à Kibana. *La figure 70* montre le log correspondant à ce scénario de test.

@timestamp		July 9th 2019, 23:22:18.911
@version		1
_id		dN6M2msB2mTL6JwLDXWR
_index		sudo-2019.07.10
_score		-
_type		doc
beat.hostname		ubuntu
beat.name		ubuntu
beat.version		6.5.4
host.architecture		x86_64
host.containerized		false
host.id		612fd69f83f0441eb0cf05d13926f9a3
host.name		ubuntu
host.os.codename		cosmic
host.os.family		debian
host.os.platform		ubuntu
host.os.version		18.10 (Cosmic Cuttlefish)
input.type		log
logsource		ubuntu
message		sonia : 3 incorrect password attempts ; TTY=pts/0 ; PWD=/home/sonia ; USER=root ; COMMAND=/usr/bin/gedit /etc/logstash/conf.d/ftp.conf
offset		72,634
program		sudo
prospector.type		log
source		/var/log/auth.log
sudo_message		incorrect password attempts
sudo_pwd		/home/sonia
sudo_runas		root
sudo_tty		pts/0
sudo_user		sonia, root
tags		_grok_sudo_success

Figure 70 : *Log normalisé de 3 tentatives de connexion échouées.*

Cet échec de connexion est détecté par le watcher « Sudo : Incorrect attempts » présenté en Annexe-C et une alerte est tout de suite générée pour la notification de cette connexion. Les figures suivantes représentent l'alerte générée ainsi que le mail lui correspondant.

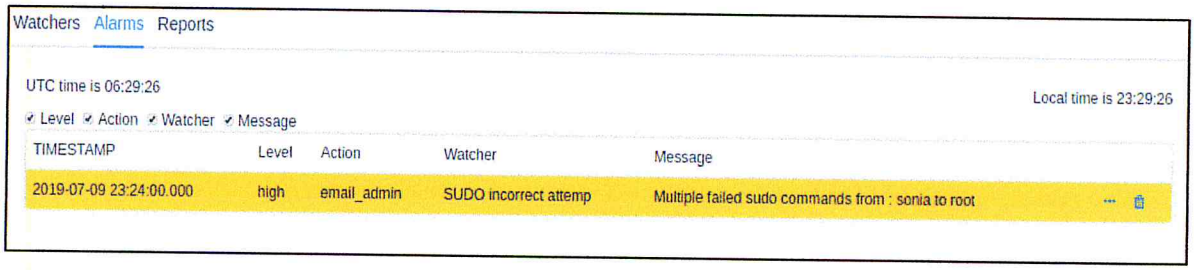


Figure 71: *Alerte « Sudo : Incorrect attempts ».*

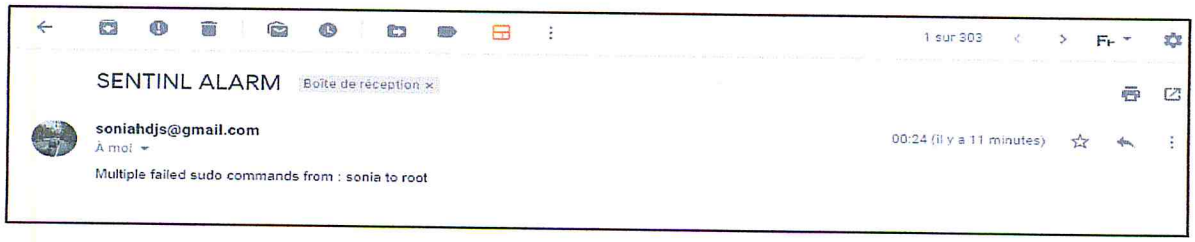


Figure 72 : *Email Correspondant à l'alerte.*

5.3.2. Tests sur système d'exploitation Windows10

- **Détection de multiple connexions échouées**

Le test de détection de multiples tentatives de connexion à un compte windows consiste à essayer de se connecter à un compte Windows avec un faux mot de passe 3 fois de suite.

Nous testerons cela sur le compte « Asma » de notre machine Windows10.

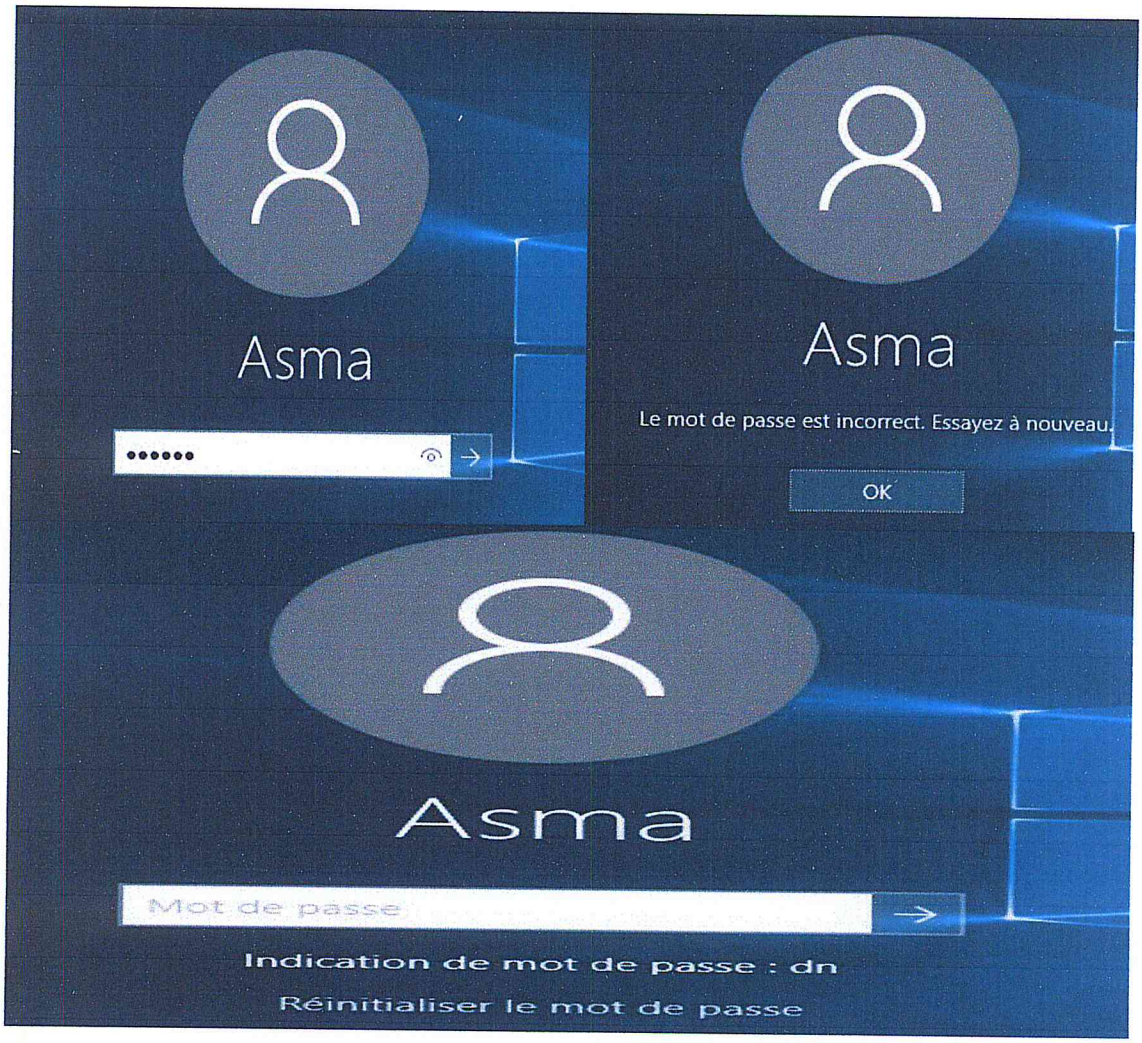
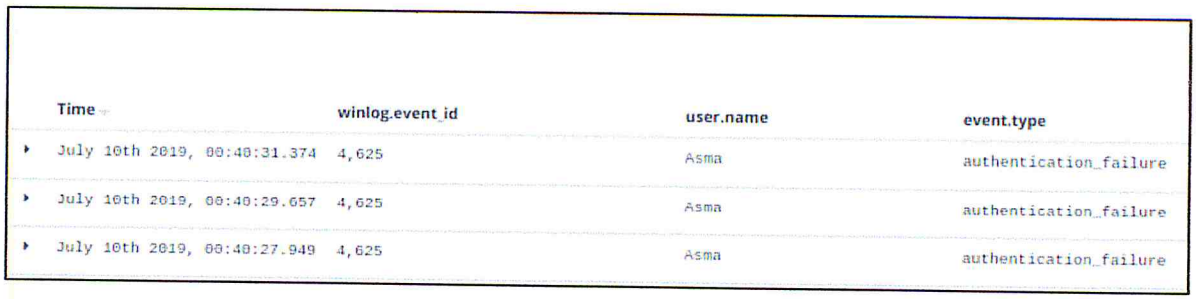


Figure 73 : *Saisie d'un faux mot de passe 3 fois de manière successive .*

Trois evenements windows d'ID 4625 (Echec d'authentification) sont alors ,traité et envoyé à Kibana. Les évènements normalisés sont présentés dans *la figure 74.*



Time	winlog.event_id	user.name	event.type
July 10th 2019, 00:40:31.374	4,625	Asma	authentication_failure
July 10th 2019, 00:40:29.657	4,625	Asma	authentication_failure
July 10th 2019, 00:40:27.949	4,625	Asma	authentication_failure

Figure 74 : *les 3 logs générés suite à la saisie de 3 faux mot de passe successifs.*

Ces échecs de connexion sont détectés par le watcher « Windows possible bruteforce » présenté en Annexe-C et une alerte est tout de suite générée pour la notification de cette connexion. Les figures suivantes représentent l'alerte générée ainsi que le mail lui correspondant.

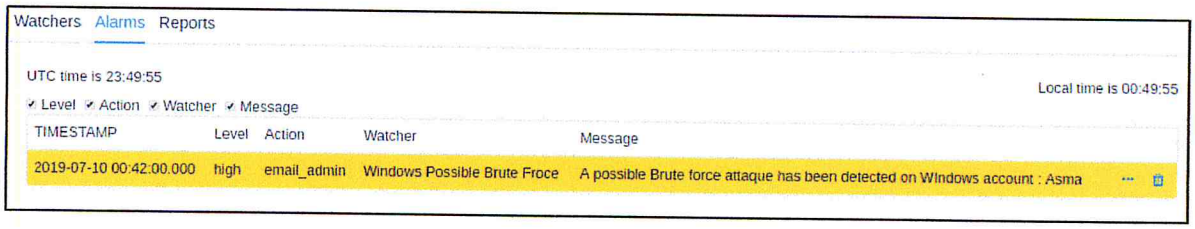


Figure 75 : *Alerte « Windows possible bruteforce ».*

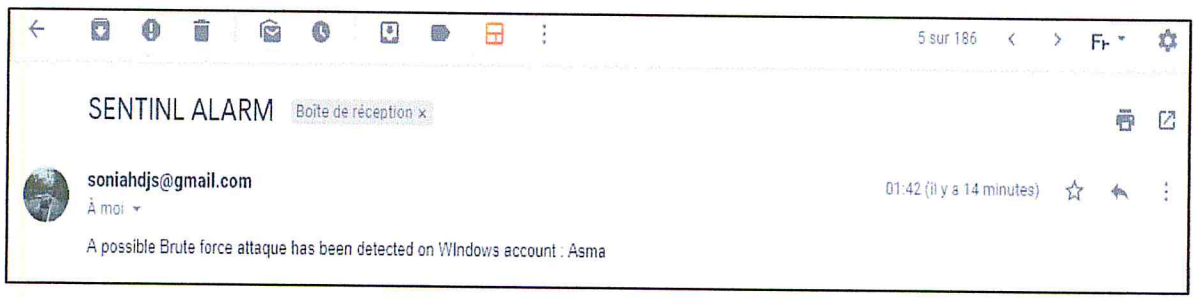


Figure 76: *Email Correspondant à l'alerte.*

5.3.3. Tests sur application web

Les tests sur application web seront effectué sur l'application web DVWA spécialement conçue pour l'exploitation des vulnérabilités web.

- **Détection d'attaque par injection SQL**

Pour commencer le test on doit d'abord se connecter à l'application comme le montre *la figure 77.*

Figure 77 : *Connexion à l'application DVWA.*

Une fois connectés, nous accédons au menu SQL Injection. Un champ de saisie est disponible pour inclure notre requête (injection).
La déclaration de selection PHP que nous allons exploiter est :

$ getid = "SELECT nom, dernier nom FROM utilisateurs WHERE id_utilisateur = ' **$ id** '";

Le principe est de remplacer le **$ id** par une entrée qui aura toujours une valeur Vraie. L'injection SQL la plus courante est ' OR '0'='0.

L'injection de ce code se traduira par la requête SQL comme suit :

mysql> SELECT nom_nom, nom_nom FROM utilisateurs WHERE id_utilisateur = ' **' OR '0' = '0'** ;

- '' ne sera égal à rien et sera donc faux.
- '0'='0' est égal = vrai car 0 sera toujours égal à 0.

La requête renverra toujours un « true » et donc la récupération des données se fera automatiquement. *La figure 78* décrit le scénario précédent.

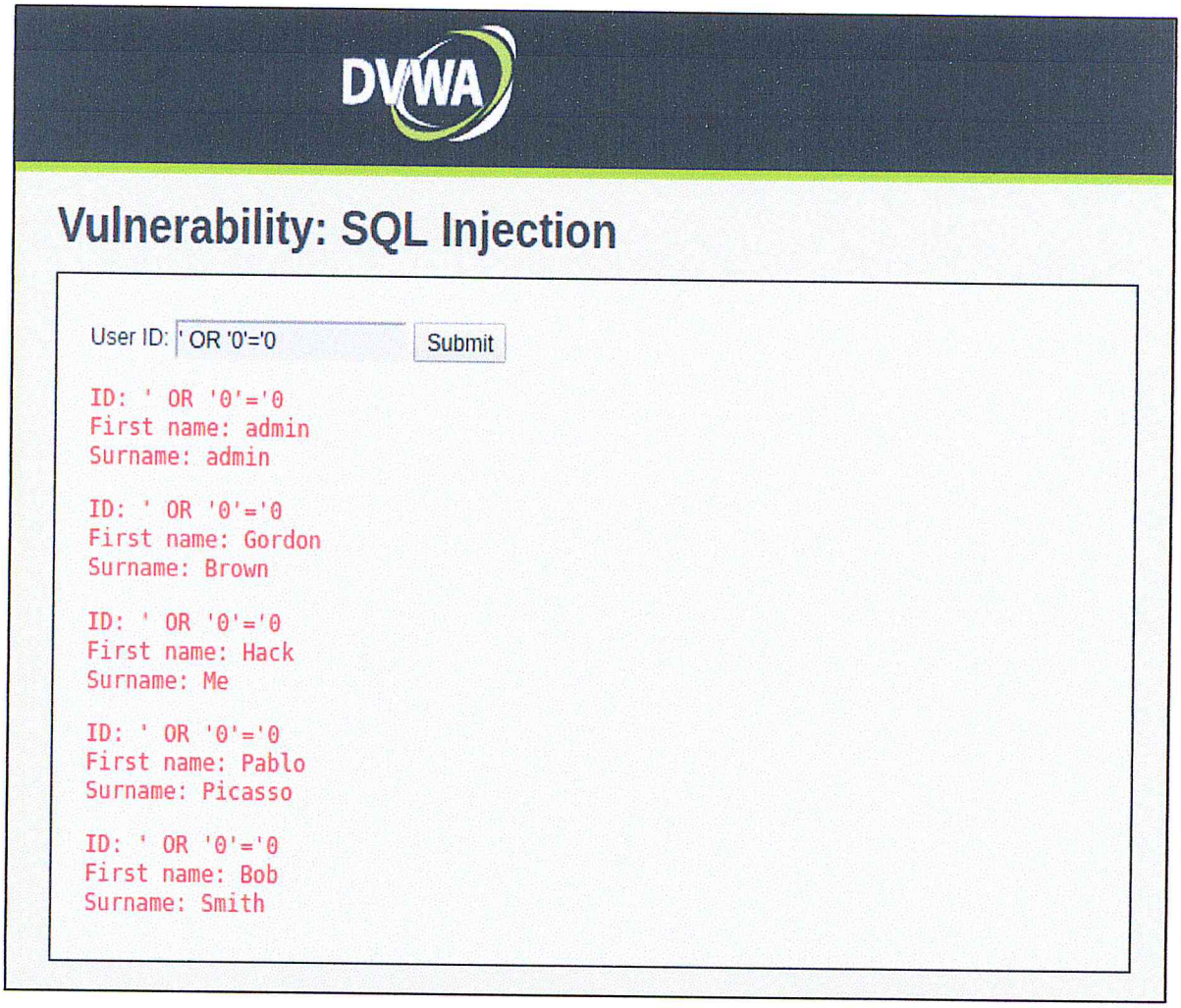


Figure 78 : *Injection SQL réussie.*

Ceci est visible au niveau du log généré par cet évènement et nous pouvons voir que le champ « Request » contient des valeurs hexadécimales car effectivement, les caractères spéciaux sont remplacés par la valeur équivalente en hexadécimale dans les logs apaches.

La figure 79 représente le log généré.

t	@timestamp	July 10th 2019, 06:07:16.000
t	@version	1
t	_id	PZVH2msBLvz_sx6WbHTE
t	_index	apache-2019.07.10
#	_score	-
t	_type	doc
t	agent	"Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36"
t	auth	-
t	beat.hostname	ubuntu
t	beat.name	ubuntu
t	beat.version	6.6.2
#	bytes	1,855
t	clientip	192.168.102.130
t	host.architecture	x86_64
◑	host.containerized	false
t	host.id	bfeb751b854b4ea9bc7ca7f7cacfa952
t	host.name	ubuntu
t	host.os.codename	cosmic
t	host.os.family	debian
t	host.os.name	Ubuntu
t	host.os.platform	ubuntu
t	host.os.version	18.10 (Cosmic Cuttlefish)
t	httpversion	1.1
t	ident	-
t	input.type	log
t	log.file.path	/var/log/apache2/access.log
t	message	192.168.102.130 - - [09/Jul/2019:22:03:42 -0700] "GET /vulnerabilities/sqli/?id=%27+OR+%270%27%3D%270&Submit=Submit HTTP/1.1" 200 1855 "http://192.168.102.129/vulnerabilities/sqli/" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36"
#	offset	17,702
t	prospector.type	log
t	referrer	"http://192.168.102.129/vulnerabilities/sqli/"
t	request	/vulnerabilities/sqli/?id=%27+OR+%270%27%3D%270&Submit=Submit
t	response	200
t	source	/var/log/apache2/access.log
t	tags	beats_input_codec_plain_applied, _grokparsefailure, _geoip_lookup_failure
t	timestamp	09/Jul/2019:22:03:42 -0700
t	useragent_build	
t	useragent_device	Other
t	useragent_major	71
t	useragent_minor	0
t	useragent_name	Chrome
t	useragent_os	Linux
t	useragent_os_name	Linux
?	useragent_patch	⚠ 3578
t	verb	GET

Figure 79 : *Log normalisé d'une tentative d' injection SQL.*

Ces injections de code sont détectées par le watcher « SQL Injection Attempt » présenté en Annexe-C et qui consiste en la détection des délimiteurs et des caractères d'échappement en utilisant des expressions régulières. Une alerte est tout de suite générée pour notifier l'utilisateur de cette attaque. Les figures suivantes représentent l'alerte générée ainsi que le mail lui correspondant.

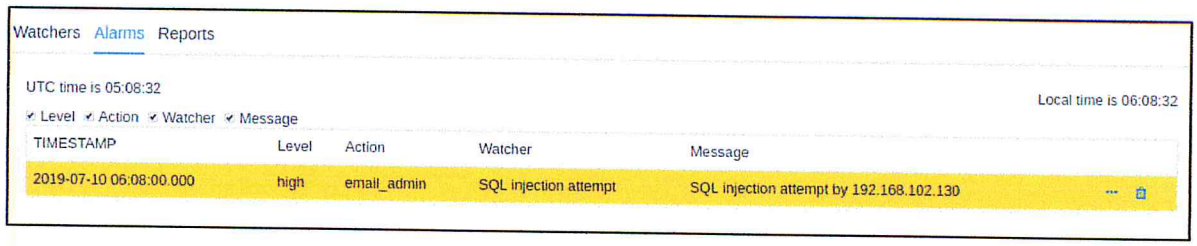


Figure 80 : *Alerte « SQL Injection Attempt » .*

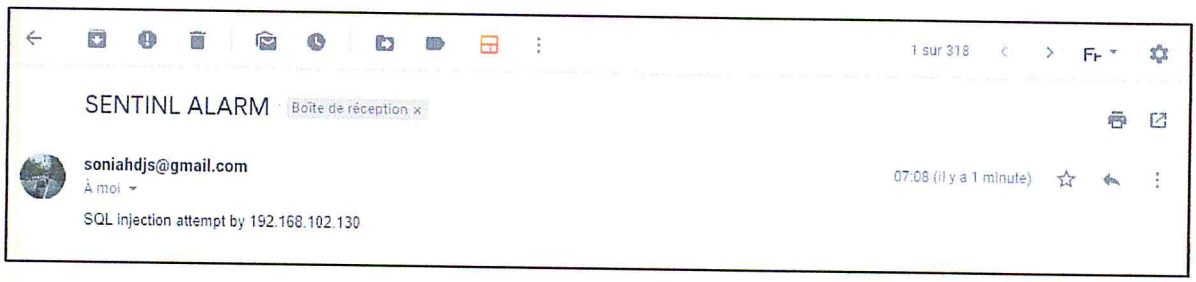


Figure 81: *Email Correspondant à l'alerte.*

➢ Détection d'attaque par injection XSS

Une attaque XSS sont réalisées en incorporant des balises de scripts. Nous testerons une des attaques les plus populaires :

- ***<script> alerte ("Ceci est un test d'exploitation XSS") </script>*** **:** ce script affiche une boite de dialogue contenant une alerte JavaScript. La boite de dialogue sera affichée à tout utilisateur utilisant le champ de texte que nous avons utilisé.
  La figure 82 présente la réalisation de l'attaque décrite.

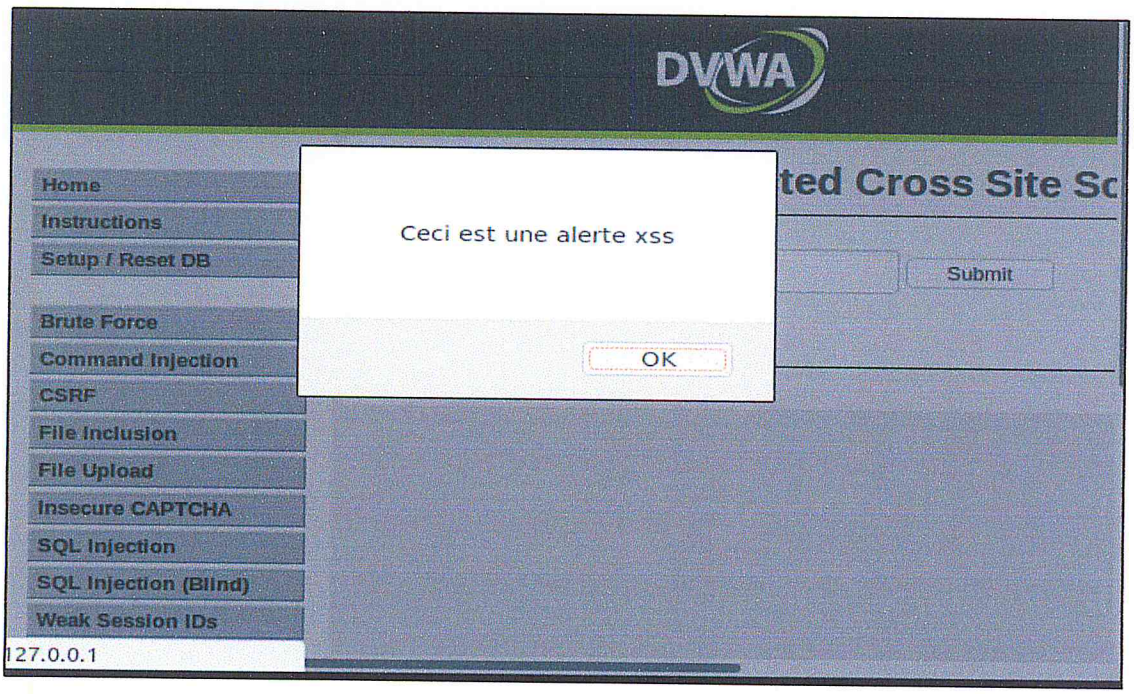


Figure 82: *Attaque XSS réussie (1).*

Un log est tout de suite généré et nous pouvons voir encore une fois ce qui a été injecté au niveau du « Request », voir *la figure 83.*

@timestamp		July 10th 2019, 06:14:58.000
@version		1
_id		a5VO2msBLvz_sxOWgYXo
_index		apache-2019.07.10
_score		-
_type		doc
agent		"Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36"
auth		-
beat.hostname		ubuntu
beat.name		ubuntu
beat.version		6.6.2
bytes		1,797
clientip		192.168.102.130
host.architecture		x86_64
host.containerized		false
host.id		bfe0751b854b4ea9bc7ca7f7cacfa952
host.name		ubuntu
host.os.codename		cosmic
host.os.family		debian
host.os.name		Ubuntu
host.os.platform		ubuntu
input.type		log
log.file.path		/var/log/apache2/access.log
message		192.168.102.130 - - [09/Jul/2019:22:14:58 -0700] "GET /vulnerabilities/xss_r/?name=%3Cscript%3Ealerte+%28%22Ceci+est+un+test+d%27exploitation+XSS%22%29%3B%3C%2Fscript%3E HTTP/1.1" 200 1797 "http://192.168.102.129/vulnerabilities/xss_r/?name=%3Cscript%3E+alerte+%28%22Ceci+est+un+test+d%27exploitation+XSS%22%29%3B%3C%2Fscript%3E" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36"
offset		24,572
prospector.type		log
referrer		"http://192.168.102.129/vulnerabilities/xss_r/?name=%3Cscript%3E+alerte+%28%22Ceci+est+un+test+d%27exploitation+XSS%22%29%3B%3C%2Fscript%3E"
request		/vulnerabilities/xss_r/?name=%3Cscript%3Ealerte+%28%22Ceci+est+un+test+d%27exploitation+XSS%22%29%3B%3C%2Fscript%3E
response		200
source		/var/log/apache2/access.log
tags		beats_input_codec_plain_applied, _grokparsefailure, _geoip_lookup_failure
timestamp		09/Jul/2019:22:14:58 -0700
useragent_build		
useragent_device		Other
useragent_major		71
useragent_minor		0
useragent_name		Chrome
useragent_os		Linux
useragent_os_name		Linux
useragent_patch		3578
verb		GET

Figure 83 : *Log normalisé d'une attaque XSS.*

Le watcher « XSS Attempt » détectera cette attaque en utilisant des expressions régulières et génèrera l'alerte correspondante tout en envoyant un email de notification. Les figures suivantes représentent l'alerte et le mail lui correspondant.

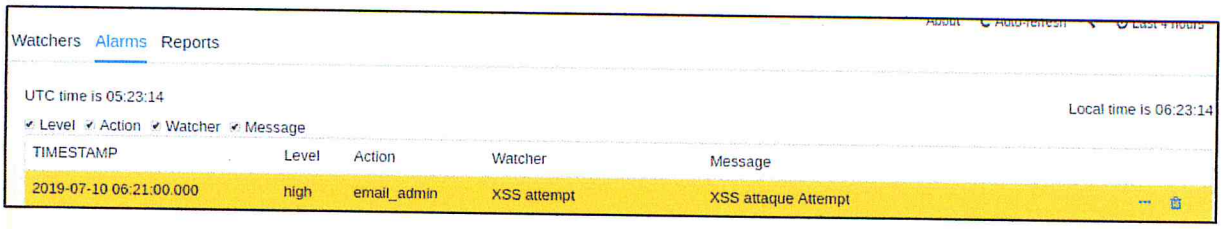


Figure 84 : *Alerte « XSS Attempt » .*

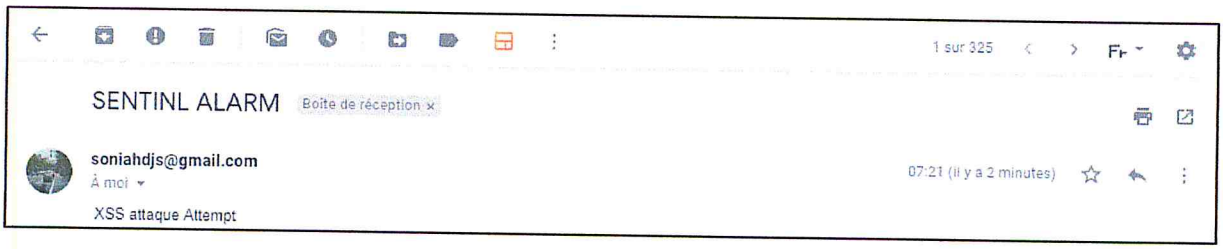


Figure 85 : *Email correspondant à l'alerte.*

5.4. Conclusion

Dans ce dernier chapitre nous avons effectué les différents scénarios d'attaques afin de voir le comportement et l'efficacité de notre solution. Les tests ont abouti car toutes les alertes ont été généré en temps quasi-réel.

Conclusion générale

Conclusion générale

Ce projet de fin d'études porte sur le développement d'un système pour la surveillance et la gestion de logs de sécurité et d'audit pour les systèmes Windows et Ubuntu pouvant être utilisé par Ooredoo pour la surveillance de ses réseaux.

Notre travail a été conçu avec des outils open-sources qui permettent d'avoir une meilleure flexibilité et adaptation aux besoins de l'entreprise.

En prenant compte de ce qui a précédé, nous pouvons dire que les objectifs fixés au début de ce projet ont été atteints. Nous avons conçu un système qui permet de collecter, parser, stocker, analyser et superviser les fichiers logs en générant des alertes dès qu'une attaque par mot de passe ou pas injection est réalisée ou même tentée. Ces alertes se basent sur des règles de corrélation précises qui permettent de détecter l'anomalie en temps réel. Une fois détecté, l'incidents est directement prit en charge par un analyste de l'équipe SOC.

Les tests émis ont donné des résultats satisfaisants car nous avons pu détecter toute tentative d'attaque testée. Cela ne veut pas dire que notre système est parfait car la sécurité de l'information n'est jamais garantie à 100%.

Notre solution pourrait être enrichie par la collecte et l'analyse des logs générés par des logiciels et des équipements de sécurité pour avoir une meilleure corrélation et donc une meilleure détection.

Ce travail a été l'occasion pour nous en tant qu'étudiantes en Sécurité des Système d'information de voir concrètement une partie du monde professionnel et de mettre en œuvre les connaissances théoriques acquises durant notre cursus.

Références

[1] https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_d%C3%A9tection_d%27intrusion

[2] https://www.techopedia.com/definition/15998/intrusion-prevention-system-ips

[3] https://www.savtec.fr/blog/posts/antivirus-anti-malware-anti-spam-quelles-differences

[4] Dictionnaire de termes juridiques, spécifiques à l'industrie et peu communs. «Defined Term » https://definedterm.com/security_relevant_information

[5]Dictionnaire de termes juridiques, spécifiques à l'industrie et peu communs. «Defined Term » https://definedterm.com/security_relevant_event_0

[6] https://whatis.techtarget.com/fr/definition/Log

[7] "Formats de fichier journal: NCSA commun"

[8] Format de fichier journal étendu

[9]https://www.ibm.com/support/knowledgecenter/SS42VS_DSM/c_LEEF_Format_Guide_intro.html

[10] https://ldapwiki.com/wiki/Common%20Event%20Format

[11]https://fr.wikipedia.org/wiki/Vuln%C3%A9rabilit%C3%A9_(informatique)

[12] https://en.wikipedia.org/wiki/Threat_(computer)

[13] https://fr.wikipedia.org/wiki/Gestion_des_incidents

[14] Jean-Christophe GALLARD. Sécurité et réseaux,v 2.0.

[15] https://www.securiteinfo.com/attaques/hacking/typesattaques.shtml

[16] http://info.haas-avocats.com/droit-digital/cybers%C3%A9curit%C3%A9-les-attaques-par-rebond-cibl%C3%A9es-par-lanssi

[17]https://dakarhacking.blogspot.com/2015/01/les-top-10-des-failles-de-securite-qui.html

[18] https://tpesecuriteinformatique.wordpress.com/les-differentes-attaques-informatiques/

[19] https://www.owasp.org/index.php/Command_Injection

[20] https://www.acunetix.com/blog/articles/local-file-inclusion-lfi/

[21]https://www.crest-approved.org/wp-content/uploads/2015/05/Cyber-Security-Monitoring-Guide

[22] https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-92.pdf

[23] https://ogma-sec.fr/centralisation-des-logs-un-outil-pour-la-securite/

[24] https://stackify.com/log-aggregation-101/

[25]https://www.microfocus.com/media/whitepaper/the_complete_guide_to_log_and_event_management_wp_fr.pdf

[26] https://www.techopedia.com/definition/31756/log-analysis

[27] https://fr.wikipedia.org/wiki/Security_information_management_system

[28] https://documentation.observeit.com/user_guide/monitoring_alerts_.htm

[29] S. Gupta et L. D. Kees, «Logging and Monitoring to Detect Network Intrusion and Compliance Violations in the Environment, » SANS Institute InfoSec Reading Room, p. 44,2012

[30] https://www.mitic.gov.py/application/files/7115/5646/3619/The_SOC_Project_for_Paraguay.pdf

[31] https://www.splunk.com/fr_fr/software/enterprise-security.html

[32] https://www.solarwinds.com/fr/security-event-manager

[33] https://www.ibm.com/fr-fr/marketplace/ibm-qradar-siem

[34] https://www.mcafee.com/enterprise/fr-fr/products/siem-products.html

[35] https://logrhythm.com/solutions/security/siem/

[36] https://solutionsreview.com/security-information-event-management/the-10-best-open-source-siem-tools-for-businesses/

[37] https://www.elastic.co/fr/elk-stack

[38] https://buzut.net/analysez-vos-logs-graylog/

[39] https://maximepiazzola.files.wordpress.com/2018/02/tableau-elk-splunk-graylog-1.pdf

[40] https://github.com/nosql-bootcamp/elastic-stack-101

[41] https://blog.zwindler.fr/2017/10/03/elasticstack-kibana-elasticsearch-logstash-beats/

[42] https://sentinl.readthedocs.io/en/latest/

[43]https://my.vmware.com/en/web/vmware/info/slug/desktop_end_user_computing/vmware_workstation_pro/15_0

[44]https://blog.garamotte.net/posts/2018/01/07/fr-limit-brute-force-attacks-on-the-ssh-service.html

[45] https://zone.votresite.ca/-/jVM0GRLJgp/

[46] https://www.linuxtricks.fr/wiki/sudo-utiliser-et-parametrer-sudoers