



Haute Ecole Economique et Technique

Avenue du ciseau 15,
1348 Ottignies-Louvain-la-Neuve

Système de monitoring pour PME : cas de la société Anderson

Travail de fin d'étude réalisé en vue de l'obtention du diplôme de bachelier en Informatique et
Systèmes orientation Technologie de l'informatique

Joel Cédric YEPGANG NAMENYI



Rapporteur : **Laurent Schalkwijk**

Année Académique 2019-2020



Avant -propos

La troisième année du cycle bachelier en Informatique et Systèmes est marquée au second semestre par l'élaboration d'un travail de fin d'études (TFE). Ce dernier a principalement pour objectif d'initier, de former et de familiariser l'étudiant que je suis à la réalisation d'un travail de recherche conciliant à la fois une étude théorique et un cas pratique à présenter devant un jury constitué d'enseignants et des professionnels. L'objet de ce travail de fin d'études, sa présentation et sa défense devraient permettre de prouver ma capacité à pouvoir mener à bien un projet de manière autonome et personnel et donc à m'intégrer dans un milieu socio-professionnel.

Cette unité d'enseignement (TFE) correspond à 16 ECTS.



Préface

Je ne saurai commencer cette rédaction sans toutefois expliqué les conditions dans lesquelles s'est déroulé ma vie, mon ressenti ces huit derniers mois.

Mon travail de fin d'étude devait initialement être réaliser en entreprise mais pour donner suite à l'annonce des différentes mesures barrières lié au covid-19 , l'entreprise déjà dans un premier temps m'a spécifié qu'elle ne pouvait plus m'accueillir. Après plusieurs mails et coup de fil sans réponse , j'ai finalement compris qu'elle m'avait abandonné à mon sort. Ce qui a été un énorme choc pour moi sur le coup. Car, ce dernier a entrainer un sérieux retard dans mes sprints et les différentes phases de réalisations car limité en ressources matérielles. Ma santé psychologique et mentale avait été très atteinte.

Néanmoins, je me suis référé à mon rapporteur, qui m'a été d'une grande aide et de premier support face à cette situation. Je me souviens encore qu'il me demandait de ne pas baisser les bras et de réagir un peu plus vite face aux difficultés.

L'autre facteur stressant fut l'environnement de travail car enfermé entre quatre murs à la maison avec une famille à nourrir et des factures à payer en pleine crise, j'étais vraiment engrisé et abattu au point de n'avoir aucune issue. Je n'avais jamais ressenti un mal aussi profond.

L'autre crainte durant cette période fut de devoir contracter la maladie et de ne devoir réaliser ou poursuivre mon travail.

Cependant avec le temps, je me suis ressaisir et j'ai repris confiance en smoi. La première fut de trouver une alternative pour mener à bien ce projet.



Remerciements

Avant tout développement de ce rapport, il apparaît judicieux de remercier ceux sans qui ce travail n'aurait pas été possible.

J'adresse ainsi mes remerciements à tous ceux qui ont eu la gentillesse de faire de ce travail de fin d'études un moment très intéressant. Il s'agit très spécifiquement de Mr Laurent Schalkwijk mon rapporteur pour ses conseils, la patience et le professionnalisme pédagogique donc il a fait preuve tout au long de nos multiples entretiens.

Je remercie également l'ensemble du corps professoral du Département Technologie de l'informatique de l'EPHEC pour le travail abattu durant ce cursus.

Enfin, je remercie également, ma compagne, ma famille et mes amis qui m'ont beaucoup soutenu moralement, physiquement et ont été toujours là depuis le début.



Introduction

Dans le cadre de notre parcours à l'Ephec, la dernière année est sanctionnée par le développement d'un sujet représentant notre projet de fin d'études. La finalité étant de mener à bien une mission venant d'un besoin exprimé ou non par une entreprise. Dans cette optique, l'élaboration de celui-ci nous permettra lors de nos multiples recherches d'acquérir une expérience socio-professionnelle puisqu'il intègre de la rigueur, de la méthode et de l'organisation. Le but visé par ce projet est de démontrer qu'avec nos acquis d'apprentissage accumulés durant notre cursus, nous sommes capables non seulement de comprendre un problème concret, c'est-à-dire mener une analyse, rechercher des solutions et de réaliser des cas pratiques, mais aussi de prouver sa capacité d'insertion socio-professionnelle par une réalisation autonome.

Ainsi, nous avons effectué plusieurs correspondances avec des entreprises, le but étant de trouver un client qui solliciterait nos compétences pour un projet, une problématique concrète au sein de sa structure. C'est ainsi qu'après plusieurs recherches, nous avons rencontré le responsable de la société Anderson.

La société Anderson fait face à une problématique qui affecte une des lignes de services. Elle aimerait bien utiliser un système de monitoring comme outil intégral dans la notification proactive de problèmes informatiques et ceux jusqu'en ce qui concerne la communication vers le client final.

Cette problématique nous a permis de faire ressortir la thématique de notre travail à savoir :

Système de monitoring pour PME : cas de la société Anderson

A ce jour, l'informatique est en constante évolution et les entreprises disposant d'un parc informatique doivent se doter et répondre à certains critères à l'instar de la disponibilité et la qualité de service pour son bon fonctionnement.

C'est dans cette analyse que la société Anderson veut traduire la problématique qu'elle rencontre auprès de ces clients en actions business et de justifier ses interventions.

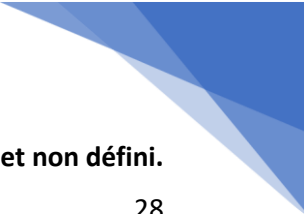
Le système de monitoring (surveillance) dans notre cas est un composant qui effectuera un contrôle d'activité important et dans le souci de mieux gérer et de contrôler l'activité de l'infrastructure, ce système non seulement garantira la disponibilité des équipements du parc informatique en cas de panne ou de baisse de performance, mais aussi permettra de répondre à un aspect proactif des éventuels problèmes pouvant survenir.

Il sera question pour nous d'être informé ou d'être notifié en temps réel des possibles cas de problèmes, d'activité interrompue et de l'état de notre réseau.

Tout en présentant l'entreprise et son cahier de charges, ce rapport fera ressortir dans un premier temps une analyse réflexive du sujet, par la suite présentera l'architecture du système, puis décrira les différents processus mis en place pour assurer la collecte, l'analyse et le stockage des données.

Table des matières

Avant -propos.....	2
Préface.....	3
Remerciements	4
Introduction.....	5
Chapitre I : Cadre du projet.....	8
Introduction.....	8
I.1 Présentation de l’entreprise.....	8
I.2 Problématique.....	9
I.3 Cahier de charges	9
Scope.....	9
II. Chapitre II: Définitions des besoins.....	10
II.1 Objectif.....	10
II.2 Approche pédagogique et définition des besoins spécifiques.....	10
II.2.1 Approche pédagogique	10
II.2.2 Définition des besoins spécifiques	10
III. Chapitre III : Analyse réflexive	12
III.1 Etude Comparative outils de gestion de projet.....	12
III.2 Etude Comparative de service réseau	13
III.3 Etude Comparative Sécurité informatique.....	15
IV. Chapitre IV: Déploiement de la solution	16
IV.1 Mise en place de l’environnement de travail.....	16
IV.1.1 Architecture du système d’information	16
IV.1.2 Les systèmes d’exploitation (operating system)	17
IV.1.3 Les logiciels	17
IV.1.4 Architecture réseau vu de GNS3	18
IV.2 Mise en place du système de monitoring	19
IV.2.1 Zabbix serveur	20
IV.2.2 Zabbix-front-end.....	20
IV.2.3 Zabbix proxy	20
IV.2.4 Intégration de la base de données partagées	20
IV.2.5 Haute disponibilité	21
IV.2.6 Pacemaker	21
IV.2.7 Corosync	22
IV.3 Mise en place du SIEM	22
IV.4 Mise en place du système de sauvegarde.....	27



V.	Difficultés rencontré.....	Erreur ! Signet non défini.
VI.	Amélioration.....	28
VII.	Recommandation	28
VIII.	Conclusion	29



Chapitre I : Cadre du projet

Introduction

L'économie mondiale a eu un essor remarquable au cours des derniers décennies, elle sait vu en plein développement dans plusieurs domaines d'investissement à l'instar du domaine informatique. C'est ainsi que la prolifération des petites et moyennes entreprises, a été simultanée avec l'expansion de la technologie qui fusionne à ce jour avec qu'elle.

Dans cette partie nous allons présenter l'entreprise accueillante, la problématique donc elle fait face et son cahier de charge.

I.1 Présentation de l'entreprise

Anderson est une société spécialisée dans la gestion informatique située à Mont-Saint-Guibert . Elle aide les entreprises à maintenir la continuité de leurs activités en gérant et en maintenant leur infrastructure informatique. Elle propose des solutions adaptées à la stratégie de l'entreprise et aux besoins de ses employés.

Son champ d'action est composé de trois principaux services donc :

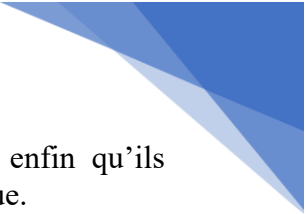
ITSI | IT INFRASTRUCTURE : Gestion intégrale de votre parc informatique; maintenance, sécurité, backup, cloud.

Son ITSI ou son infogérance, en tant que prestataire de services extérieur lui permet d'effectuer : la gestion, l'exploitation, l'optimisation et la sécurité du système d'information d'une entreprise. Elle peut l'appuyer en fournissant un support et une maintenance à ces collaborateurs à travers un responsable et son équipe dynamique pour tout éventuel problème ou question.

ITSD | IT SERVICE DESK : consiste en la centralisation du service de support informatique en un unique point de contact pour les collaborateurs, avec une gestion intégrale des demandes et incidents.

Ce service établit des contacts avec leur partenaire via un portail client qui est un lien de communication privilégié entre les collaborateurs et le support informatique. De même la gestion d'incident est prise en charge par une équipe réactive et proactive. Ce qui leur permet d'avoir un rapport d'efficacité via un reporting donnant ainsi une vue globale et spécifique sur le parc et des actions à exécuter. Ce qui permet à leur collaborateur de prendre des décisions stratégiques en ce qui concerne la gestion d'incident.

ITSP | IT PROJECT : ce service participe à la gestion de projet, elle s'appuie sur la collaboration avec ses partenaires dans la réalisation de leurs projets à travers un processus de suivi continu leur offrant comme support un audit et sécurité en adaptant au mieux le besoin de ces derniers dans le but de prévenir de l'efficacité et la fiabilité de leur système.



De même, il accompagne dans l'installation et la configuration du matériel enfin qu'ils s'adaptent à tout système en entreprise et aussi pour éviter tout problème technique.

La société est constituée d'une équipe donc un Directeur , un responsable des ressources humaines, une équipe en support online et sur site.

I.2 Problématique

La société fait face à une problématique qui affecte une des lignes de services. Elle aimerait bien utiliser un système de monitoring comme outil intégral dans la notification proactive de problèmes informatiques et ceux jusqu'à communication vers le client concerné.

I.3 Cahier de charges

Scope

Le système de monitoring sera principalement ciblé vers l'infrastructure informatique d'une PME de 10 à 50 utilisateurs informatiques. La solution comprendra trois phases critiques :

- **Outil de monitoring intégral**

Installation & configuration d'un outil de monitoring informatique capable de surveiller tout type de hardware ou software au sein d'une infrastructure.

L'outil devra être capable d'envoyer des notifications automatiques vers un système de gestion de tickets : Service Desk.

Il sera important de déterminer tous les éléments, 'assets', nécessaires d'être surveillés, de les catégoriser et ensuite classer par ordre d'importance.

- **Standardisation des process d'utilisation**

À la suite de l'analyse des assets et de leur importance, une matrice de scénarios devra être dressée afin de prévoir les procédures d'output de chaque notification reçue dans le Service Desk.

- **Dashboard d'alignement client**

Afin de maintenir la communication la plus efficace et transparente possible envers le client final, toutes les notifications importantes ainsi que leur output pourront être répertoriés dans un log sous format de Dashboard d'alignement.

Conclusion : Cette première partie a permis de définir une problématique ,un périmètre et un contexte qui constitue le cadre de notre projet.

Dans le deuxième chapitre nous allons définir les objectifs et effectuer une analyse liée aux spécifications des besoins .



II. Chapitre II: Définitions des besoins

Introduction

Une bonne approche pour mener à bien un projet serait d'effectuer une bonne analyse des différents composants pouvant conduire à sa conception et sa réalisation.

Dans ce chapitre nous allons définir les objectifs et définir des besoins vu d'une approche pédagogique pour mener à bien ce projet.

II.1 Objectif

Nous nous sommes définis comme objectif d'implémenter une solution cible qui permettrait de mutualiser les objectifs poursuivis par l'entreprise(alignement business/IT) et les besoins métiers du client clé.

II.2 Approche pédagogique et définition des besoins spécifiques.


II.2.1 Approche pédagogique

Notre dévotion pour ce projet réside dans sa particularité à faire ressortir des besoins vu aux cours tels que la /le: **Gestion de projet** , **Service réseau**, **Sécurité informatique**, **SGBD**. Ces derniers nous ont permis de mieux abordé notre phase d'analyse de réflexive.

II.2.2 Définition des besoins spécifiques

Pour mieux cerner notre nous nous focaliser sur les besoins ci-dessous enfin de mieux orienter notre démarche dans les prochains chapitres.

- **Gestion de projet** : est une démarche visant à organiser de bout en bout le bon déroulement d'un projet. Une bonne gestion des facteurs opérationnels et de méthodologie font qu'un projet puisse aboutir dans un triangle représentant l'équilibre qualité-coût-délai (QCD).
- **Service réseau** : c'est une fonctionnalité assurée par un ou plusieurs serveurs ayant pour fonction la fourniture d'informations à d'autres ordinateurs via une connexion réseau normalisée.

- 
- **Sécurité informatique** : c'est une discipline qui vise la protection de l'intégrité et la confidentialité des informations stockées dans un système informatique, en outre le contrôle des utilisateurs et des droits d'accès aux données.
 - **SGBD** : c'est un système qui stocke et gère des données de façon organisée et cohérente.

Conclusion :

Cette partie nous permis de faire ressortir notre objectif de même que les besoins spécifiques que nous expliciterons dans notre analyse réflexive au travers d'une étude comparative.

III. Chapitre III : Analyse réflexive

Introduction

Nous avons vu dans les deux chapitres précédents le cadre du projet et la définition des besoins pour mieux aborder notre choix.

Pour cela nous avons spécifié ce troisième chapitre pour faire une étude comparative sur les logiciels open source à fin de justifier notre choix et savoir les fonctionnalités des logiciels libres, leurs avantages, leur fiabilité et leur l'efficacité.

III.1 Etude Comparative outils de gestion de projet

Plusieurs outils existent sur le marché parmi lesquels certaines sont plus communautaire offrant des versions non payante de base.

Nous avons pu étudier les trois meilleurs en ce moment sur le marché

Trello :

est un outil de gestion de projet en ligne, lancé en septembre 2011 et inspiré par la méthode Kanban de Toyota. Il repose sur une organisation des projets en planches listant des cartes, chacune représentant des tâches. Les cartes sont assignables à des utilisateurs et sont mobiles d'une planche à l'autre, traduisant leur avancement.


La version de base est gratuite, tandis qu'une formule payante permet d'obtenir des services supplémentaires.

- Trello version gratuite disponible
- Forfait « Business Class » à 9.99\$/mois/utilisateur
- Forfait « Enterprise » à 20.83\$/mois/utilisateur

Asana :

est un gestionnaire de communication d'équipe. Le produit prend en charge de nombreuses fonctionnalités, notamment les espaces de travail, des projets, des tâches, des étiquettes, des notes, des commentaires et une boîte de réception qui organise les mises à jour des informations en temps réel. Il est conçu pour permettre aux individus et aux équipes de planifier et gérer leurs projets et les tâches sans email. Chaque équipe reçoit un espace de travail. Les espaces de travail contiennent des projets, et les projets contiennent des tâches.

- Asana version gratuite disponible
- Forfait « Premium » à 9.99\$/mois/utilisateur
- Forfait « Enterprise » uniquement sur devis



Zoho : est une suite d'outils de productivité et d'applications SaaS en ligne pour gestion de toute une entreprise. Actuellement cette suite est payante.

- Forfait « Standard » à 240€
- Forfait « Express » à 480€
- Forfait « Premium » à 1020€
- Forfait « Enterprise » à 1500€


Le travail a été supervisé en SCRUM et les sprints ont été définis à partir d'user stories et du backlog . dans notre cas nous avons utilisé la plateforme Trello pour des raisons de collaborations simplifier et organisationnelle avec les tableaux, les listes et les cartes très simples d'utilisation.

III.2 Etude Comparative de service réseau

Le réseau informatique étant un ensemble d'équipements reliés entre eux pour échanger des informations. Il a été important pour nous de choisir un meilleur service réseau.

Notre étude nous a permis de répertorier plusieurs d'entre eux. Chacun ayant des avantages et ces inconvénients des uns et des autres.

CATEGORIES	OUTILS	AVANTAGES	INCONVENIENTS
OPEN SOURCE	Zabbix	Gratuit, facilité d'installation pour les débutants	Documentation et interface graphique complexe
	Nagios		Non réactivité du développeur sur le partage des modules
PAYANTE	PRTG	Offre un essaie de son environnement pour 30 en limité	Logiciel à but commercial
	OP5	Installation très facile	



Parmi ces solutions deux d'entre elles ont particulièrement retenu notre attention : Nagios et Zabbix.

Nagios :

Nagios est un outil de monitoring appliqués aux serveurs. Il permet de suivre à la trace l'état des services, et de remonter une alerte si un problème existe. On peut également le configurer pour prendre des initiatives si aucune intervention n'est effectuée après un seuil.

Zabbix :

Zabbix est une solution centralisée logiciel « audit des performances et des défaillances ». Ce logiciel est libre, il est très réputé de même que Nagios. Concrètement, le principe du monitoring est simple, on installe un serveur qui va contrôler un certain nombre de points sur d'autres serveurs et se mettre à envoyer des alertes si cela va mal.

	<i>Nagios</i>	<i>Zabbix</i>
+	<ul style="list-style-type: none">• Référence Open Source.• Excellente gestion de pannes.• Bibliothèque étendue de plugins.	<ul style="list-style-type: none">• Intégration des fonctions gestion de pannes, de performances et reporting.• Interface Web pour toutes les fonctions.• Simplicité d'utilisation.• Evolution rapide.
–	<ul style="list-style-type: none">• Configuration complexe.• Intégration difficile de fonctions configuration, gestion de performances, reporting.	<ul style="list-style-type: none">• Bibliothèque limitée de modèles d'équipements.• Reporting basique.

Le choix technique du service réseau (du moniteur de supervision) a été Zabbix, parce qu'il comporte plusieurs fonctionnalités , sa facilité de déploiement et aussi il me permettait d'implémenter le SIEM choisi.

III.3 Etude Comparative Sécurité informatique

Le SIEM (Security event management, « Gestion des événements de sécurité ») est pratiquement le logiciel le plus important dans notre projet il nous faut d'abord une étude comparative pour choisir un bon logiciel et l'implémenter ensuite dans notre architecture.

Après une recherche ciblée nous avons proposé trois logiciels concurrents

Cyberoam iView :

Cyberoam iView est une solution de log-reporting open source qui fournit aux organisations de la visibilité sur leurs réseaux à travers de nombreux dispositifs pour des niveaux élevés de sécurité, de confidentialité des données tout en satisfaisant les obligations de conformité réglementaire.

Splunk :

Splunk est une solution logicielle pouvant s'installer sur n'importe quel OS et propose une approche originale à la collecte, l'analyse et la corrélation de logs. En effet Splunk permet l'indexation universelle des logs qu'ils soient issus des applications, des logiciels de sécurité, et des serveurs.

ELK :

acronyme issu des trois composants principaux de la suite (Elasticsearch, Logstash, Kibana) Elle permet en outre :

- la collecte des données au travers de Logstash (ou des Beats comme nous le verrons plus loin) ;
- le stockage des données dans le moteur d'indexation Elasticsearch ;
- l'exploitation et l'analyse des données au travers de Kibana.

La mise en place d'un cluster est chose aisée ainsi que le déploiement dans le Cloud.

Plusieurs types d'architectures sont possibles, car nous verrons notamment que la collecte et le déversement des données peut se faire avec plusieurs types de sources ou puits de données.

Notre choix c'est rapidement porté sur la ELK qui est en forte croissance et nous permettait d'avoir une visualisation plus étendue et mieux exploiter dans notre projet et aussi pour avantage lié à sa forte communauté.

Conclusion :

Dans cette partie , nous avons effectué une analyse et une étude comparative des différentes solutions lie à nos différents besoins. Dans le chapitre suivant nous procéderons au déploiement de la solution.

IV. Chapitre IV: Déploiement de la solution

Introduction

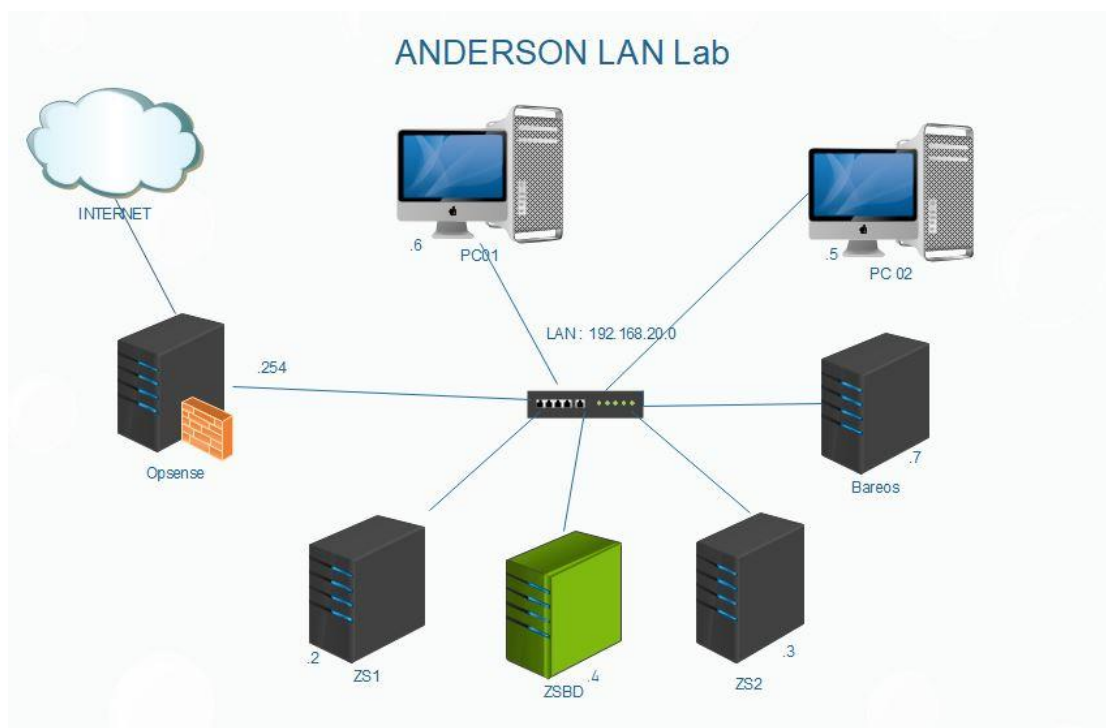
Dans ce chapitre nous définirons notre environnement de travail, ces assets (architecture du système d'information) constituant notre laboratoire de travail.

IV.1 Mise en place de l'environnement de travail

Pour la réalisation de ce projet nous nous sommes permis d'utiliser et d'exploiter environnements et logiciels bien connus.

IV.1.1 Architecture du système d'information

L'architecture suivante représente l'asset constituant le laboratoire tel que définie pour notre projet



IV.1.2 Les systèmes d'exploitation (operating system)

- Os Windows utilisés pour l'installation des logiciels de simulation réseau.
Windows 10 Edge (version trial)
- Os Debian ou Ubuntu 18.04 pour l'installation des services et logiciels de contrôle.

IV.1.3 Les logiciels

- GNS3 ;
- VMWare ;
- Edraw max.

Gns3, pour Graphical Network Simulator est un logiciel libre permettant l'émulation ou la simulation de réseaux informatiques. Publié pour la première fois en 2008, Il permet la combinaison de périphériques virtuels et réels, utilisés pour simuler des réseaux complexes. Il utilise le logiciel d'émulation Dynamips pour simuler Cisco IOS . Il est utilisé par de nombreuses grandes entreprises, notamment Exxon , Walmart , AT & T et NASA , et est également populaire pour la préparation des examens de certification professionnelle en réseau. En 2015, le logiciel avait été téléchargé 11 millions de fois.

Raison du choix

- Il utilise des fonctions issues du monde de Linux. Sous Windows ou Mac, ses fonctions peuvent être mal interprétées. Le serveur étant sous Ubuntu, cela permet à GNS3 d'exécuter ces fonctions sans problème.
- Il utilise la puissance des logiciels de VMware Workstation et son rendement est supérieur à la simple émulation.
- Il permet d'effectuer une simulation concrète de son réseau avant un déploiement définitif en environnement réel ou physique.
- Il permet l'installation de périphériques autres que les routeurs Cisco. Notons qu'en fonction de la taille du réseau à mettre en place , cette dernière pourra demander plus de ressource. Son utilisation dans ce projet nécessite que l'on crée une liaison réseau à la VMware Workstation, ce qui permet la sauvegarde de toutes manipulations effectuées sur gns3 vers Workstation.

VMware Workstation : Ce logiciel est un outil de virtualisation de poste de travail créé par la société VMware, il peut être utilisé pour mettre en place un environnement de test pour développer de nouveaux logiciels, ou pour tester l'architecture complexe d'un système d'exploitation avant de l'installer réellement sur une machine physique.

Nous avons opté pour une version premium enfin de bénéficier de tout l'environnement qu'offre cette plateforme.

Avantages de cet hyperviseur

- **Partitionnement** : Il exécute plusieurs systèmes d'exploitation et repartit les ressources entre les différentes machines virtuelles.
- **Isolation** : Il assure la protection de la sécurité au niveau matériel et les performances accrues tout en déployant le contrôle des ressources.
- **Encapsulation** : Il enregistre des fichiers donnant états des différentes machines virtualisés.

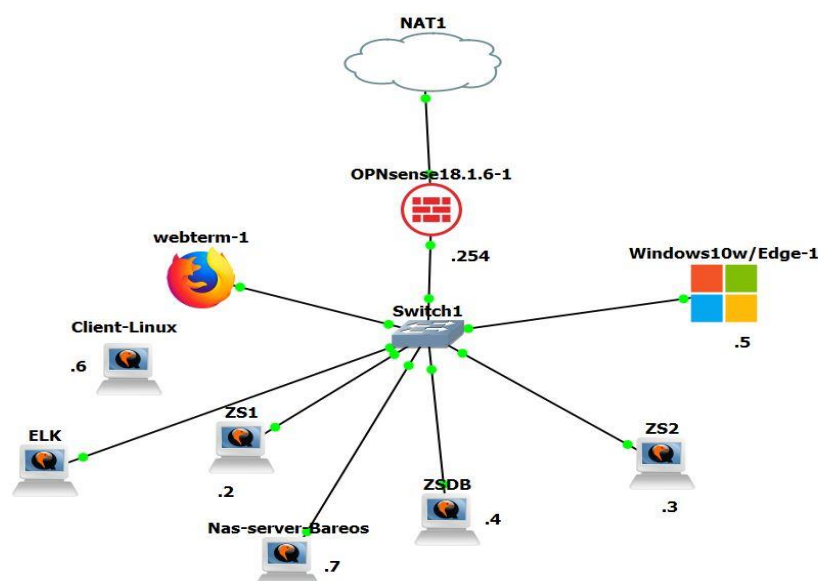
Pour exécuter VMware Workstation, le système hôte et le système d'exploitation hôte doivent satisfaire à des exigences matérielles et logicielles spécifiques. Pour permettre à gns3 d'être hypervisé car celui-ci est gourmand en ressources. Ainsi, Pour pouvoir installer ces logiciels, nous nous sommes procuré une machine faisant office de serveur hôte. Cette **machine hôte** disposera des ressources suivantes

- Système d'exploitation Windows 10 Enterprise;
- 12 Go de mémoire RAM (random Access memory) ;

Processeur Intel Core I5 3.5Ghz et 400 Go de disque dur ssd

IV.1.4 Architecture réseau vu de GNS3

la figure ci-dessous reprend les Appliance installé sur gns3.



IV.2 Mise en place du système de monitoring

Le monitoring ou monitoring est une activité de surveillance et de mesure d'une activité informatique. On parle aussi de supervision.

Les raisons de cette pratique peuvent être variées notamment pour des raisons de :

- Performance, en termes de temps de réponse par exemple ;
- Disponibilité, indépendamment des performances ;
- D'intégrité, l'état des processus sur une machine linux par exemple.

Dans le contexte actuel , travaillant dans un environnement de test, nous nous sommes servis d'un système d'exploitation Ubuntu 18.4 enfin de déployer notre solution.

Lors de notre première élaboration de solution, nous envisagions de mettre dans l'un des réseaux locaux un serveur qui devait faire du monitoring et un autre en externe enfin de faire du failover.

Mais après réévaluation des possibles risques et de perte d'information ou d'indisponibilité de ce serveur ,nous avons recouru à une autre solution qui fut de mettre en place deux serveurs qui nous permettraient de faire du basculement entre les deux si jamais l'un tombait. Cependant, enfin de pousser un peu plus l'idée , il fut judicieux de mettre entre ces deux serveurs une solution appelée pacemaker que nous développerons dans la section suivante. Cela étant, et à des fins plus amélioratives, nous nous sommes posé la question de savoir ce qui se passerait si jamais l'un des serveurs présents en local était stoppé et qu'advientra-t-il de ce dernier une fois qu'il sera rétabli. D'où, l'idée d'une base de données partagée entre les deux serveurs.

Dans la suite de ce rapport nous développerons les aspects importants et pertinents en ce qui concerne la mise en place d'un tel système de surveillance.

Il convient d'avoir sur chacune des machines que nous nommerons Zabbix serveur1 (ZS1) et Zabbix serveur2 (ZS2) des os (operating system) Ubuntu 16.04 identiques. La version Xénial de Ubuntu et la base de données MySQL a été utilisées sur celle-ci.

Mais avant d'entrer dans le vif de l'installation, il convient de faire une présentation de l'utilité de chaque logiciel installé dans les manipulations qui couvriront la supervision d'un telle infrastructure réseau.

L'architecture Zabbix repose sur quatre composants donc un Zabbix serveur, Zabbix agent, Zabbix frontend, Zabbix proxy et MySQL ou MariaDb.

IV.2.1 Zabbix serveur

Il tourne exclusivement sur linux, et il est chargé de collecter les données, les analyser, réaliser des graphes, les stocker dans une base de données.

Zabbix-agent

C'est un logiciel installé sur les équipements qui devront être surveillés. Il est possible de l'installer sur des systèmes d'exploitation tels que Windows, linux, mac etc. Il interagit, collecte, envoie les données récupérées et écoute sur le port 10050 par défaut.

Notons que la collecte d'information se fera auprès des différents périphériques c'est l'exemple pour un switch ou d'un routeur où nous utiliserons le protocole SNMP pour les monitorer

IV.2.2 Zabbix-front-end

Ce logiciel va permettre à travers une interface web de mettre en forme les données (graphiques), consulter les dernières mesures, aussi d'administrer.

IV.2.3 Zabbix proxy

Enfin de décharger les serveurs zabbix(ZS1 OU ZS2), de surveiller les emplacements distants mais surtout de simplifier la maintenance du réseau distribué , nous avons mis en place sur le réseau distant un serveur zabbix proxy enfin de collecter des données de performances et de disponibilité pour le compte des serveurs zabbix .

Nous avons pu remarquer ainsi que l'utilisation d'un proxy est un moyen plus simple d'implémenter une surveillance centralisée et distribuée. Car toutes les données récoltées sont stockées localement avant d'être transmises aux serveurs principaux.


Ainsi, aucune donnée n'est perdue en raison de problèmes de communication temporaires avec le serveur. Les paramètres ProxyLocalBuffer et ProxyOfflineBuffer dans le fichier de configuration du proxy contrôlent la durée de conservation locale des données.

Il est bien de noter que , le proxy Zabbix est un collecteur de données. Il ne calcule pas les déclencheurs, ne traite pas les événements et n'envoie pas d'alertes.

IV.2.4 Intégration de la base de données partagées

Pour des raisons de disponibilités d'informations, il est ici installé dans un serveur différent des deux autres une base de données donc le moteur utilisé ici est MYSQL server.

Sur le serveur nommé zabbix data base (Zsdb) dédié au partage de la base de données , la collecte et la sauvegarde de l'état de la BD des différents serveurs zabbix1 et zabbix2. Jusqu'ici



se faisait en local. Nous avons ainsi procédé à l'installation d'une base de données secondaire ou de backup nommée mariadb.

`apt Install mariadb-server.`

Nous l'avons sécurisée en créant un nouvel utilisateur et un mot de passe pour se connecter à la base de données. Il fallut ensuite se rendre sur l'un des serveurs zabbix de surveillance enfin de transférer le script du fichier de base de données local vers la base de données partagée.

Bien évidemment, une connexion cliente ssh est nécessaire pour ce type de transfert car certaines informations d'authentification seront demandées pour se connecter au serveur distant.

IV.2.5 Haute disponibilité

En effet, le failover dans notre cas est un dispositif mis en place entre nos deux serveurs formant des nœuds où chacun fonctionne indépendamment de l'autre. Il aura pour objectif, d'assurer la haute disponibilité et fonctionne selon ce principe ; quand un nœud tombe en panne ou est indisponible pour une quelconque raison, un autre nœud du cluster prend la relève et assure le relais pour maintenir la disponibilité du serveur en attendant le rétablissement de l'autre nœud.

Pour le mettre en pratique dans notre situation, il fallut procéder à l'installation de pacemaker et de corosync.

IV.2.6 Pacemaker

C'est un gestionnaire de cluster haute disponibilité. Il est chargé de démarrer, arrêter et de superviser les ressources d'un cluster. Il a pour rôle de basculer entre deux ou plusieurs machines à travers une IP de basculement préalablement configuré. En outre, c'est cette IP de basculement qui est renseigné dans la configuration des deux Zabbix serveurs permettant ainsi la liaison pacemaker Zabbix.

Pour qu'une telle installation fonctionne, nous avons référencé ces champs dans zabbix enfin de pouvoir identifier et de renseigner chaque basculement.

- Zabbix1 : premier système de basculement (anderson.example.com).
- Zabbix2 : deuxième système de basculement (anderson.example.com) .
- Zabbix - IP de basculement qui est une adresse IP flottante du réseau sur lequel, sont situés mes deux serveurs.

IV.2.7 *Corosync*

L'installation de pacemaker s'appuyant sur d'autres logiciels tels heartbeat ou corosync pour agir et surveiller des postes, il nous fallut permettre aux deux machines de s'envoyer des informations d'où l'usage de corosync.

IV.3 Mise en place du SIEM

Elastic Stack - anciennement connu sous le nom de ELK Stack - est une collection de logiciels open source produits par Elastic qui vous permet de rechercher, d'analyser et de visualiser les journaux générés à partir de n'importe quelle source dans n'importe quel format, une pratique connue sous le nom de journalisation centralisée. La journalisation centralisée peut être très utile lorsque vous essayez d'identifier des problèmes avec vos serveurs ou applications, car elle vous permet de rechercher dans tous vos journaux en un seul endroit. Il est également utile car il vous permet d'identifier les problèmes qui s'étendent sur plusieurs serveurs en corrélant leurs journaux pendant une période donnée.

L'Elastic Stack comprend quatre composants principaux:

Elasticsearch : un moteur de recherche RESTful distribué qui stocke toutes les données collectées.

Logstash : le composant de traitement des données d'Elastic Stack qui envoie les données entrantes à Elasticsearch.

Kibana : une interface web pour rechercher et visualiser les logs.

Beats : des expéditeurs de données légers et à usage unique qui peuvent envoyer des données de centaines ou de milliers de machines à Logstash ou Elasticsearch.

Remarque : lors de l'installation d'Elastic Stack, vous avons utilisé la même version sur l'ensemble de la pile.

Les étapes d'installation :

Étape 1 - Installation et configuration d'Elasticsearch

Les composants Elastic Stack ne sont pas disponibles dans les référentiels de packages par défaut d'Ubuntu. Ils peuvent cependant être installés avec APT après avoir ajouté la liste des sources de paquets d'Elastic.

Pour commencer, exécutez la commande suivante pour importer la clé GPG publique Elasticsearch dans APT:

```
• wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key  
add -
```

• Ensuite, ajoutez la liste des sources Elastic aux sources. List. Répertoire, où APT recherchera de nouvelles sources :

```
• echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee  
-a /etc/apt/sources.list.d/elastic-7.x.list
```

• Ensuite, mettez à jour les listes de packages afin qu'APT lise la nouvelle source Elastic:

```
• sudo apt update
```

• Ensuite, installez Elasticsearch avec cette commande:


```
• sudo apt install elasticsearch
```

• Une fois que l'installation est terminée ElasticSearch, utilisez votre éditeur de texte préféré au fichier de configuration principal de modifier ElasticSearch, elasticsearch.yml. Ici, nous utiliserons nano:

```
• sudo nano /etc/elasticsearch/elasticsearch.yml
```

Elasticsearch écoute le trafic de partout sur le port 9200. Vous souhaitez restreindre l'accès extérieur à votre instance Elasticsearch pour empêcher des tiers de lire vos données ou d'arrêter votre cluster Elasticsearch via l'API REST. Trouvez la ligne qui spécifie network.host, décommentez-la et remplacez sa valeur par localhost pour qu'elle ressemble à ceci:

```
/etc/elasticsearch/elasticsearch.yml  
  
. . .  
network.host: localhost  
. . .
```



Enregistrez et fermez elasticsearch.yml en appuyant sur CTRL+X, suivi de Yet ensuite ENTER si vous utilisez nano. Ensuite, démarrez le service Elasticsearch avec systemctl:

- `sudo systemctl start elasticsearch`

- Ensuite, exécutez la commande suivante pour permettre à Elasticsearch de démarrer à chaque démarrage de votre serveur:

- `sudo systemctl enable elasticsearch`

- nous pouvons tester si notre service Elasticsearch est en cours d'exécution en envoyant une requête HTTP:

- `curl -X GET "localhost:9200"`

la réponse suivante apparaîtra

Output

```
{
  "name" : "ElasticSearch",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "SMYhVWRiTWS1dF0pQ-h7SQ",
  "version" : {
    "number" : "7.6.1",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "aa751e09be0a5072e8570670309b1f12348f023b",
    "build_date" : "2020-02-29T00:15:25.529771Z",
    "build_snapshot" : false,
    "lucene_version" : "8.4.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

Maintenant qu'Elasticsearch est opérationnel, installons Kibana, le prochain composant d'Elastic Stack.

Étape 2 - Installation et configuration du tableau de bord Kibana

Selon la documentation officielle, vous ne pouvez installer Kibana qu'après l'installation d'Elasticsearch. L'installation dans cet ordre garantit que les composants dont dépend chaque produit sont correctement en place.

Étant donné que nous avons déjà ajouté la source du package Elastic à l'étape précédente, vous pouvez simplement installer les composants restants d'Elastic Stack en utilisant apt:

- `sudo apt install kibana`

Ensuite, activez et démarrez le service Kibana:

- `sudo systemctl enable kibana`

- `sudo systemctl start kibana`

Étant donné que Kibana est configuré pour écouter uniquement localhost, nous devons configurer un proxy inverse pour autoriser l'accès externe à celui-ci. Nous utiliserons Nginx à cette fin, qui devrait déjà être installé sur votre serveur.

Tout d'abord, utilisez la `openssl` commande pour créer un utilisateur administratif Kibana que vous utiliserez pour accéder à l'interface Web de Kibana. À titre d'exemple, nous nommerons ce compte `kibanaadmin`, mais pour assurer une plus grande sécurité, nous vous recommandons de choisir un nom non standard pour votre utilisateur qui serait difficile à deviner.

La commande suivante créera l'utilisateur et le mot de passe administratifs Kibana, et les stockera dans le fichier `htpasswd.users`. Vous allez configurer Nginx pour exiger ce nom d'utilisateur et mot de passe et lire ce fichier momentanément:


- `echo "kibanaadmin:`openssl passwd -apr1`" | sudo tee -a /etc/nginx/htpasswd.users`

Entrez et confirmez un mot de passe à l'invite. Rappelez-vous ou prenez note de cette connexion, car vous en aurez besoin pour accéder à l'interface Web de Kibana.

Ensuite, nous allons créer un fichier de bloc de serveur Nginx. À titre d'exemple, nous ferons référence à ce fichier comme `anderson.com`, bien que vous puissiez trouver:

- `sudo nano /etc/nginx/sites-available/anderson.com`

Ajoutez le bloc de code suivant dans le fichier, en veillant à ce que la mise `anderson.com` à jour corresponde au nom de domaine complet ou à l'adresse IP publique de notre serveur. Ce code configure Nginx pour diriger le trafic HTTP de votre serveur vers l'application Kibana, qui écoute `localhost:5601`. De plus, il configure Nginx pour lire le `htpasswd.users` et exiger une authentification de base.



```
/etc/nginx/sites-available/example.com
```

```
server {  
    listen 80;  
  
    server_name anderson.com;  
  
    auth_basic "Restricted Access";  
    auth_basic_user_file /etc/nginx/htpasswd.users;  
  
    location / {  
        proxy_pass http://localhost:5601;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection 'upgrade';  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
    }  
}
```

Lorsque vous avez terminé, enregistrez et fermez le fichier.

Ensuite, activez la nouvelle configuration en créant un lien symbolique vers le sites-enabled répertoire. Si vous avez déjà créé un fichier de bloc serveur avec le même nom dans le prérequis Nginx, vous n'avez pas besoin d'exécuter cette commande:

```
• sudo ln -s /etc/nginx/sites-available/example.com /etc/nginx/sites-enabled/example.com
```

```
•
```

Vérifiez ensuite la configuration pour les erreurs de syntaxe:

```
• sudo nginx -t
```

```
•
```


Si des erreurs sont signalées dans votre sortie, revenez en arrière et vérifiez que le contenu que vous avez placé dans votre fichier de configuration a été ajouté correctement. Une fois que vous voyez syntax is ok dans la sortie, continuez et redémarrez le service Nginx:

```
• sudo systemctl restart nginx
```

Pour autoriser les connexions à Nginx, nous pouvons ajuster les règles en tapant:

```
• sudo ufw allow 'Nginx Full'
```

```
•
```



Kibana est désormais accessible via votre FQDN ou l'adresse IP publique de votre serveur Elastic Stack. Vous pouvez vérifier la page d'état du serveur Kibana en accédant à l'adresse suivante et en entrant vos informations de connexion lorsque vous y êtes invité:

```
http://your_server_ip/status
```

Étape 3 - Installation et configuration de Logstash

Bien qu'il soit possible pour Beats d'envoyer des données directement à la base de données Elasticsearch, il nous a été recommandé d'utiliser Logstash pour traiter les données. Cela nous permettra de collecter des données de différentes sources, de les transformer dans un format commun et de les exporter vers une autre base de données.

Installez Logstash avec cette commande:

- ```
sudo apt install logstash
```

## IV.4 Mise en place du système de sauvegarde

**Bareos** est une solution de sauvegarde basée sur Bacula. Dans cet article, je vais vous montrer comment l'installer et comment réaliser des sauvegardes et des restaurations. Nous allons aussi installer l'interface Web de l'outil.

### Fonctionnement

Bareos repose sur quatre composants principaux:


Le director (programme de contrôle central): le director est utilisé pour planifier les sauvegardes et restaurer les fichiers. Il supervise toutes les opérations de sauvegarde, de restauration, de vérification et d'archivage.

La console : sert à communiquer avec le director.

Le storage (SD): programme qui va effectuer des lectures et des écritures sur les périphériques de stockage utilisés pour les sauvegardes. Il gère les médias où sont écrites les données. Dans le cas d'une restauration, il est chargé de trouver les données et de les envoyer au démon de fichier Bareos.

Le file (FD): gère la partie cliente et est installé sur toutes les machines à sauvegarder. Le composant FD est un logiciel qui fournit au serveur Bareos, l'accès aux données qui seront sauvegardées.

### Installation Bareos



Entrez la commande ci-dessous pour installer **Bareos**, le lien de base de données et l'interface web:

```
apt-get install bareos bareos-database-mysql bareos-webui
```

## V. Difficultés rencontrées

La réalisation de ce projet à fait surgir plusieurs problèmes parmi lesquelles :

- L'établissement la sauvegarde au niveau de notre serveur de stockage ;
- Le manque de ressource matérielle conséquente ce qui a engendrer des bug continuelle sur notre pc
- La mise en place de Elk sur zabbix.

## VI. Amélioration

Bien que le système soit mis en place pour répondre au cahier de charge, il reste bon nombre de fonctionnalité à incorporer dans celui-ci. Il s'agit par exemple de :

- La mise en place d'un système rapport fournissant un état sur les statistiques du réseau tous les mois ou semaines ;
- La réflexion sur la mise en place d'un failover avec un nœud sur un serveur externe(cloud) ;
- L'orchestration de tout le réseau l'une des pistes serait ansible.

## VII. Recommandation

Pour toutes les procédures d'installation, il est impératif de regarder le lien GitHub suivant :

<https://github.com/projet2019/TFE>

## VIII. Conclusion

Le sujet portait sur l'Implémentation **Système de monitoring pour PME : cas de la société Anderson**. Ceci permettant de réduire le temps d'intervention lors de certaines défaillances systèmes. Et d'aligner son corps business de l'entreprise à celui de ces clients A travers les outils tel que Zabbix, GNS3, VMWARE, COROSYNC, PACEMAKER, ce thème nous a permis de mettre en place un système d'information (SI) au travers des différentes phases.

Dans l'optique de notifier les personnes chargées de l'administration du réseau, un système de mailing a été mis en place permettant de nous alerter lorsque survient une défaillance système.

En définitif, bien que ceci réponde à la ligne directrice fixée en introduction, il reste encore à ce jour un grand travail à effectuer au niveau du système d'alerte qui est celui de la mise en place d'un système de backup (sauvegarde automatique) de notre système de monitoring vers le serveur NAS prévus pour la contenance de ces fichiers.

