Ministère de l'Enseignement Supérieure et de la Recherche ScientifiqueUniversité Abderrahmane Mira

جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

ATE Department

Faculté de Technologie

Département d'Automatique, Télécommunications et d'Electronique

Projet de Fin d'Etudes

Pour l'obtention du diplôme de Master

Filière : Télécommunications

Spécialité : Réseaux et Télécommunications

Thème

Automatisation et supervision d'un système informatique.

Préparé par :

AMIA Karim
ZIANI Madina

Dirigé par :

M. DIBOUNE Abdelhani
Mme MEHDI Sonia

Examiné par :

Mme OUALI Kahina (Président)
*M. ALLICHE Abdenour (Exa*minateur)

Année universitaire : **2022/2023**

REMERCIEMENTS

*Nous tenons tout d'abord à exprimer notre profonde gratitude envers **ALLAH**, le grand, l'infini et le tout-puissant, de nous avoir illuminées et d'avoir ouvert les portes du savoir. Nous sommes reconnaissantes d'avoir bénéficié de Sa volonté, de la santé et du courage nécessaires pour mener à bien ce travail.*

Nous souhaitons exprimer nos sincères remerciements à nos promoteurs, M. DIBOUNE Abdelhani et Mme MEHDI Sonia, pour leur soutien, leurs encouragements et leurs précieux conseils tout au long de ce travail. Leur expertise et leurs orientations nous ont été d'une grande aide pour mener à bien notre recherche.

Nous tenons également à exprimer notre reconnaissance à l'ensemble des membres du jury, en particulier à Mme OUALI Kahina et M. ALLICHE Abdenour, pour l'intérêt qu'ils ont porté à notre travail. Nous sommes reconnaissantes de leur disponibilité et de leur expertise, ainsi que de leur acceptation d'examiner notre travail et de l'enrichir par leurs propositions constructives.

Nous tenons à remercier tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail. Leur soutien, leurs encouragements et leur confiance ont été essentiels dans notre cheminement.

Nous exprimons notre profonde gratitude envers tous ceux qui ont participé à cette aventure et nous ont permis de mener à bien ce travail.

Merci à tous.

Dédicaces

Je tiens tout d'abord à exprimer ma gratitude envers Allah pour m'avoir donné la force et le courage nécessaires pour réaliser ce modeste travail.

C'est avec un immense plaisir, un cœur ouvert et une joie indescriptible que je dédie ce travail :

À mes très chers parents, Azize et Djamila, que j'aime énormément, pour leur patience, leur amour, leur encouragement et leur sacrifice tout au long de mon parcours. Que Dieu vous garde en bonne santé pour nous.

À mes chères sœurs, Djida et Cherifa, que le bon Dieu les protège et les guide dans leur chemin.

À mes anges Elina et Meriem, que Dieu vous garde

À mes frères, Lyazid, Azwaw, Ghilas et Abdelsalam, pour leur soutien et leur présence.

À mes chers tonton Mhmed et Rafik et tata Nassima, je vous aime énormément. Merci pour votre présence et votre soutien constant.

À ma meilleure amie, Melissa, et à Fatima, merci pour votre présence et votre soutien inconditionnel.

Que cette dédicace témoigne de ma profonde reconnaissance envers ceux qui ont été présents dans ma vie, m'entourant de leur amour, de leur soutien et de leur amitié précieuse. Votre soutien a été un pilier essentiel dans la réalisation de ce travail.

Merci à tous.

Madina

Dédicaces

Je tiens à exprimer ma gratitude envers Allah pour m'avoir accordé la force et le courage nécessaires pour accomplir ce modeste travail.

C'est avec un immense plaisir, un cœur ouvert et une profonde joie que je dédie ce travail :

À mes très chers parents, dont la patience, l'amour, l'encouragement et les sacrifices ont été inestimables tout au long de mes études. Aucun hommage ne pourra véritablement rendre justice à l'amour avec lequel ils ne cessent de me combler. Que Dieu leur accorde une bonne santé et une longue vie.

À mon cher frère, pour son amour et sa compréhension, qui a toujours été un modèle exemplaire pour moi.

À mes chères sœurs, pour leur aide et leurs encouragements précieux.

À mes amis qui ont rendu mon quotidien à l'université plus agréable, en partageant des moments de joie, d'apprentissage et d'entraide. Leur présence a été un soutien précieux tout au long de ce parcours.

Que cette dédicace témoigne de ma profonde reconnaissance envers tous ceux qui ont contribué à mon épanouissement personnel et académique. Que leur bienveillance et leur soutien continuent d'illuminer ma route vers de nouveaux horizons.

KARIM

Résumé

Ce mémoire a été rédigé en prenant pour inspiration le réseau d'un client de l'entreprise Campus NTS (Nouvelle Technologie et Solution) afin de créer le nôtre. L'objectif était de rechercher et d'appliquer une solution efficace, gratuite et open source pour automatiser et superviser ce réseau.

Ce mémoire se présente essentiellement comme un test de solution utilisant GNS3. Nous avons utilisé deux logiciels, Ansible pour l'automatisation et Zabbix pour la supervision. Ansible utilise des playbooks, dans lesquels les tâches à automatiser sont écrites sous forme de code YAML. Quant à Zabbix, il utilise une interface graphique qui affiche toutes les informations des équipements. Cette solution sera appliquée à des équipements d'un réseau de base tels que les pares-feux, les commutateurs, les routeurs et les serveurs. Toutes les étapes d'installation, de configuration et de test sont présentées et expliquées dans ce mémoire.

Mots clés : *Système informatique, Automatisation, Supervision, Ansible, Zabbix, SSH, SNMP*

Abstract

This thesis was written by drawing inspiration from a client's network at Campus NTS (New Technology and Solution) in order to create our own. The aim was to research and apply an efficient, free, and open-source solution to automate and supervise this network.

This thesis essentially serves as a solution test using GNS3. We utilized two software tools, Ansible for automation and Zabbix for supervision. Ansible employs playbooks, where the tasks to be automated are written in YAML code. As for Zabbix, it utilizes a graphical interface that displays all equipment information. This solution will be applied to basic network equipment such as firewalls, switches, routers, and servers. All installation, configuration, and testing steps are presented and explained in this thesis.

Keywords *: Computer system, Automation, Supervision, Ansible, Zabbix, SSH, SNMP.*

Table des matières

Liste des Figures

Chapitre I

Chapitre II

Chapitre III

Chapitre IV

Liste des playbooks

Chapitre IV

Liste des Tableaux

Chapitre I

Chapitre II

Chapitre III

Chapitre IV

Liste des abréviations

A :

API : Application Programming Interface

APT : Advanced Packages Tool

ADDS: Active Directory Domain Services

B :

BASH : Bourne Again Shell

C :

CLI : Command Line Interface

CMIP : Common Management Information Protocol

CPU : Central Processing Unit

D :

DHCP : Dynamic Host Configuration Protocol

DMZ : Demilitarized Zone

DNS : Domain Name System

E:

Eth: Ethernet

G :

GNS3 : Graphical Network Simulator 3

GIT : Global Information Tracker

GUI : Graphical User Interface

GPU: Graphical Processing Unit

H :

HTTPS : HyperText Transfer Protocol Secure

HTTP : HyperText Transfer Protocol

I :

IAB : Interactive Advertising Bureau

IP : Internet Protocol

ISO : International Organization for Standardization

IOS : Internetwork Operating System

ID: Identificateur

ICMP: Internet Control Message Protocol

J :

JSON : JavaScript Object Notation

K :

KVM : Keyboard, Video and Mouse

L :

LAN: Local Area Network

LS : List

M :

MKDIR : MaKe DIRectory

MIB : Management Information Base

N :

NMS : Network Management System

NSOT : Network Source Of Truth

O :

OID : Object Identifier

P :

PING : Packet Internet Groper

PPA : Personal Package Archive

PHP: Hypertext Preprocessor

Q :

QOS : Quality of Service

R :

RFC : Request for Comments

RSA : Rivest, Shamir, Adleman

REST API : Representational State Transfer Application Programming Interface

RAM: Random Access Memory

S :

SDN : Software Defined Networking

SNMP : Simple Network Management Protocol

SSH : Secure Shell

SUDO : Super User DO

SMTP: Simple Mail Transfer Protocol

SSMTP: Secure Simple Mail Transfer Protocol

SQL: Structured Query Language

T :

TelNet : Terminal Network

TCP: Transmission Control Protocol

U :

UDP : User Datagram Protocol

UI : Interface utilisateur (User Interface)

URL : Uniform Resource Locator

V :

VLAN : Virtual Local Area Network

VM : Virtual Machine

VDOM: Virtual Domain

W:

WIN: Windows

WAN: Wide Area Network

Y :

YAML : Yet Another Markup Language

Introduction Générale

Introduction générale

Aujourd'hui, les systèmes informatiques jouent un rôle crucial au sein des entreprises, devenant à la fois essentiels et complexes. La maintenance et la gestion de ces systèmes sont devenues des priorités absolues, car une défaillance à ce niveau peut entraîner des conséquences catastrophiques. Pour faire face à ces défis, deux concepts clés ont émergé : l'automatisation et la supervision. Ces approches offrent des solutions pratiques et intelligentes qui permettent d'améliorer considérablement la gestion et les opérations des réseaux des entreprises.

L'automatisation joue un rôle central dans l'optimisation des systèmes informatiques des entreprises. Elle consiste à mettre en place des processus automatisés pour exécuter des tâches répétitives et chronophages, qui étaient auparavant effectuées manuellement par les employés. Les entreprises utilisent des outils et des logiciels spécialisés pour automatiser des activités telles que la configuration des réseaux, la gestion des sauvegardes, les mises à jour logicielles, et bien plus encore. En automatisant ces tâches, les entreprises peuvent gagner un temps précieux, réduire les erreurs humaines et améliorer l'efficacité globale de leurs opérations informatiques.

La supervision, quant à elle, est essentielle pour surveiller et contrôler en temps réel les systèmes informatiques et les réseaux d'une entreprise. Grâce à des outils de supervision avancés, les équipes informatiques peuvent collecter et analyser des données en continu, afin de détecter les problèmes potentiels et d'y remédier avant qu'ils ne se transforment en incidents majeurs. La supervision permet de surveiller les performances du réseau, les temps de réponse, les niveaux de charge, la sécurité et bien d'autres paramètres critiques. Cela permet aux équipes informatiques d'identifier les problèmes émergents, de diagnostiquer les causes profondes et de prendre des mesures correctives de manière proactive.

L'adoption de l'automatisation et de la supervision présente de nombreux avantages pour les entreprises. Tout d'abord, elles permettent de réduire les risques d'erreurs humaines, souvent liées à la monotonie et à la fatigue. En automatisant les tâches, les entreprises peuvent améliorer la fiabilité de leurs systèmes et réduire les temps d'arrêt non planifiés. De plus, la supervision constante permet de détecter les problèmes rapidement, ce qui permet une réaction immédiate et une résolution plus rapide des incidents. Cela contribue à minimiser l'impact sur les activités de l'entreprise et à maintenir la satisfaction des clients.

Pendant notre stage chez NTS à Béjaïa, nous avons constaté que le client rencontrait plusieurs difficultés liées à la gestion de son réseau. En effet, la présence d'un grand nombre d'équipements rendait leur configuration, surveillance et gestion manuelles complexes et chronophages. Cette situation a engendré plusieurs problèmes, tels qu'une visibilité limitée sur l'état du réseau, rendant difficile la détection rapide des incidents et des pannes. De plus, les opérations manuelles répétitives nécessaires pour gérer et configurer ces nombreux

équipements ont entraîné des temps d'arrêt et ont eu un impact négatif sur la productivité des utilisateurs.

De plus, le risque d'erreurs humaines a augmenté, ce qui compromet la cohérence et la fiabilité du réseau. Enfin, l'absence d'un système de surveillance approprié a également exposé le réseau à des risques de sécurité élevés, mettant en danger la confidentialité et l'intégrité des données et des informations.

Après avoir identifié les problèmes liés à la gestion et à la configuration des équipements réseau du client, nous avons proposé la mise en place d'une solution complète pour automatiser et superviser le réseau. Cette solution est basée sur deux éléments principaux : un serveur d'automatisation, un serveur de supervision.

Ce mémoire est structuré en quatre chapitres. Le premier chapitre, intitulé "L'automatisation et la supervision des réseaux informatiques", présentera une étude sur l'automatisation et la supervision en général, ainsi que l'examen de différentes solutions avant de choisir celle qui convient le mieux à notre situation.

Le deuxième chapitre, "Présentation de l'entreprise", portera sur l'entreprise Campus NTS où nous avons effectué notre stage. Nous y présenterons sa structure hiérarchique, son réseau informatique et les protocoles utilisés. Ensuite, nous réaliserons une étude approfondie de leur problématique spécifique et proposerons une solution détaillée.

Le troisième chapitre, "Environnement de travail", se concentrera sur la mise en place d'un système d'exploitation adapté et l'installation des outils nécessaires pour créer la maquette du réseau.

Le quatrième chapitre, "Simulation", sera dédié à l'automatisation des tâches et à la supervision des différents équipements du réseau. Nous présenterons les résultats de nos simulations et évaluerons l'efficacité de la solution proposée.

Enfin, nous finirons par une conclusion générale récapitulative des points essentiels de notre travail et nous aborderons les perspectives futures.

Chapitre I : L'automatisation et supervision des systèmes informatique

Introduction

La gestion des systèmes informatiques est une tâche complexe et fastidieuse qui nécessite beaucoup de temps et d'efforts. Dans le but de simplifier et d'améliorer cette tâche, l'automatisation et la supervision des réseaux sont devenues des pratiques courantes pour les entreprises.

Dans ce chapitre, nous allons nous concentrer sur deux aspects essentiels de la gestion des réseaux : l'automatisation et la supervision.

Dans la première section, nous aborderons le sujet de l'automatisation des réseaux. Tout d'abord, nous mettrons en lumière les défis rencontrés lors de la configuration des réseaux, notamment la complexité croissante des infrastructures et la nécessité de gérer les configurations de manière cohérente et efficace. Ensuite, nous définirons le concept d'automatisation des processus et examinerons l'architecture et le fonctionnement de cette pratique. Nous expliquerons les différentes étapes nécessaires à la mise en place de l'automatisation d'un réseau informatique ainsi que les avantages et outils disponibles pour aider les entreprises à automatiser leurs processus. Enfin, nous présenterons quelques solutions pratiques qui peuvent être utilisées pour mettre en œuvre l'automatisation des réseaux dans un environnement informatique professionnel.

Dans La seconde section, nous explorerons la supervision des réseaux, en examinant le concept de la supervision, les différents types de surveillance et les actions qui y sont liées. Nous passerons en revue la norme ISO du point de vue de la gestion des réseaux, en nous concentrant sur les différentes formes de gestion, telles que la gestion des performances, la gestion des configurations, la gestion de la comptabilité, la gestion de la sécurité et la gestion des anomalies. Nous examinerons également la structure de gestion des réseaux, y compris les agents, les manageurs et la MIB (Management Information Base). Nous terminerons en discutant des protocoles de surveillance, avec une attention particulière portée sur le protocole SNMP (Simple Network Management Protocol) et nous citerons quelques outils de supervision populaires pour aider les entreprises à surveiller en temps réel la santé et la performance de leurs infrastructures informatiques.

Ce chapitre fournit les informations nécessaires pour comprendre ces deux aspects clés de la gestion des systèmes informatiques et pour mettre en place une infrastructure informatique efficace et fiable.

Automatisation des réseaux

Avant d'aborder le sujet de l'automatisation des réseaux, il est important de comprendre les défis auxquels les administrateurs réseau sont confrontés. Pour cela, nous allons commencer par évoquer les défis de la configuration réseau dans la prochaine section.

I.1.1 Les défis de la configuration réseau

La raison principale pour une organisation d'adopter l'automatisation réseau est de réduire le temps nécessaire pour maintenir et déployer des changements sur le réseau. Bien que le temps soit crucial, toutes les organisations ne choisissent pas de passer à l'automatisation réseau.

Chaque réseau est aujourd'hui unique, tout comme les appareils de réseau. Cela décourage les entreprises de passer à l'automatisation de leur réseau, car cela implique souvent une mise à niveau de l'équipement actuel ou le passage à des technologies telles que SDN, car ces dispositifs uniques sont parfois difficiles, voire impossibles, à intégrer dans un réseau automatisé [1].

Un autre obstacle pour passer à l'automatisation réseau est le manque d'un schéma normalisé sans fournisseur associé à un environnement abordable pour les tests.

Outre l'économie de temps, l'automatisation réseau aide à résoudre les problèmes sur un réseau. Les administrateurs réseau devaient configurer manuellement chaque appareil via CLI et, en cas de changement à effectuer sur tous les appareils, tels qu'un ajout d'un nouveau VLAN, ils devaient accéder à chaque appareil et le configurer.

Cela était non seulement chronophage, mais cela maximisait également la possibilité d'une erreur. De plus, il est dangereux d'appliquer des modifications sur un réseau pendant les heures de travail, et certaines entreprises travaillent tous les jours et il y a une fenêtre serrée, généralement pendant les vacances, pour appliquer les changements.

Des enquêtes menées ont montré que la raison la plus courante d'une interruption du réseau est due à des erreurs humaines. L'erreur humaine la plus courante en matière de réseau est la mauvaise configuration des appareils de réseau.

Il est courant qu'un administrateur réseau applique un fichier de configuration de mise à jour à un ensemble d'appareils réseau. Comme mentionné précédemment, il y a généralement une période de temps limitée pour effectuer cette tâche, et lorsque des tâches comme celles-ci doivent être effectuées à la main à partir d'un environnement CLI en copiant-collant la configuration et en l'appliquant sans vérification approfondie, c'est une pratique courante.

C'est là que l'automatisation réseau intervient pour aider.

I.1.2 Définition

L'automatisation des réseaux désigne le processus qui vise à simplifier et accélérer la gestion des réseaux informatiques en automatisant la configuration, la gestion, les tests, le déploiement et le fonctionnement des dispositifs physiques et virtuels. Cette approche permet d'éviter la réalisation manuelle de tâches répétitives et complexes en utilisant des technologies et des outils dédiés. Ainsi, l'automatisation des réseaux améliore l'efficacité et la sécurité des réseaux informatiques.

L'automatisation du réseau peut être mise en place dans tous les types de réseaux. Les solutions matérielles et logicielles permettent aux centres de données, aux fournisseurs de services et aux entreprises de mettre en œuvre l'automatisation du réseau pour améliorer l'efficacité, réduire les erreurs humaines et les coûts d'exploitation [2]

I.1.3 Architecture et Fonctionnement

Afin d'assurer une automatisation efficace, il est primordial d'avoir une architecture d'automatisation bien pensée, qui répond aux besoins spécifiques de l'organisation. Cette architecture doit être flexible et indépendante de tout produit spécifique, qu'il soit commercial ou open source [3].

La Figure 5 illustre les différentes phases recommandées pour la mise en œuvre d'une architecture d'automatisation.

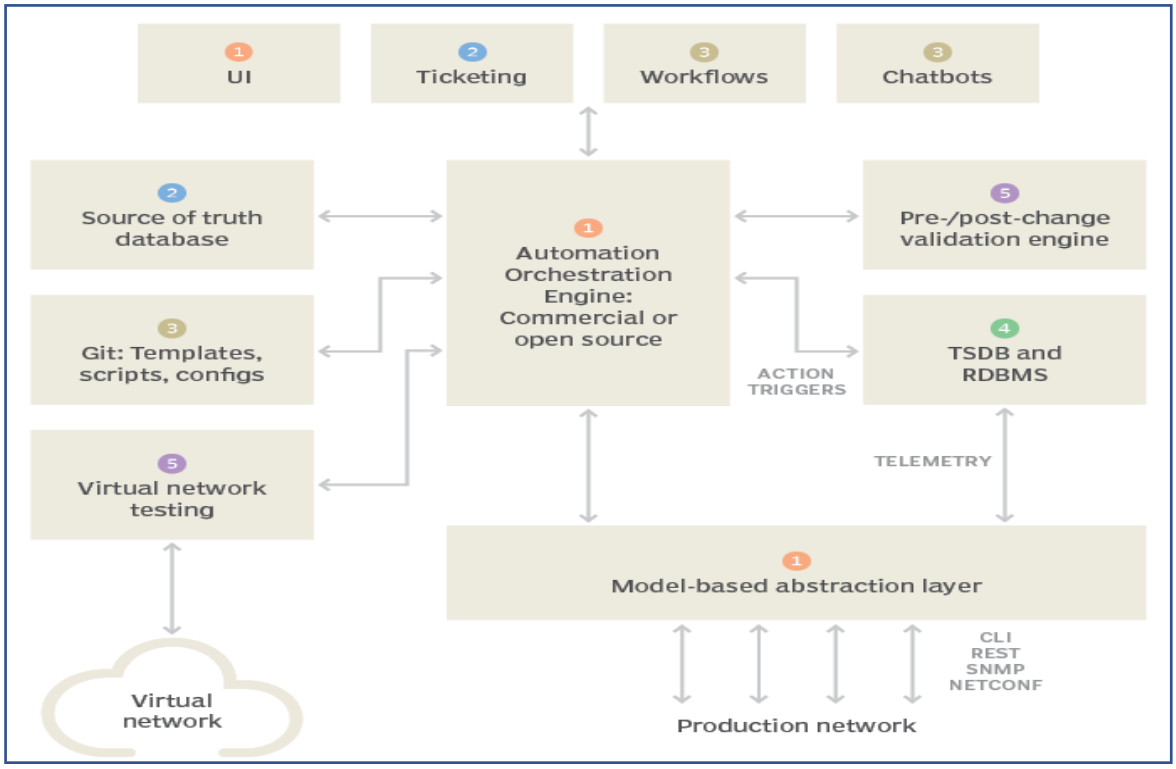


Figure I. 1 : Architecture d'automatisation.

❖ ***La phase 1*** (1)

Cette phase met en place trois fonctions clés : l'orchestration d'automatisation, l'interface utilisateur (UI) et la couche d'abstraction de périphérique.

1) Le système d'orchestration d'automatisation **(Automation Orchestration Engine)** :

C'est le moteur central qui contrôle le processus d'automatisation. Il assure la mise à l'échelle, la coordination et l'exécution des tâches d'automatisation sur l'ensemble du réseau.

Il peut s'appuyer sur des outils open source ou des produits commerciaux pour effectuer cette fonction.

2) L'interface utilisateur (UI) :

Elle fournit une interface graphique utilisateur (GUI) et une API pour contrôler le système d'automatisation. Les projets open source peuvent s'appuyer sur une interface de ligne de commande pour le contrôle administratif et une API pour le contrôle programmatique.

3) La couche d'abstraction de périphérique (Model-based abstraction layer) :

Elle fournit un modèle qui cache les différences entre les fournisseurs de périphériques. Elle simplifie considérablement l'interface de périphérique réseau en fournissant un modèle commun pour les différentes marques de périphériques.

Cette couche d'abstraction peut être intégrée à certains systèmes d'orchestration pour faciliter l'automatisation de la configuration du réseau.

❖ ***La phase 2*** 2

Cette phase ajoute une base de données Network Source of Truth (NSoT) et une interface à un système de gestion des incidents (interface to a trouble-ticketing system).

1) Le Network Source of Truth (NSoT) :

La NSoT stocke des informations sur l'état souhaité du réseau que le système d'orchestration d'automatisation utilise pour valider, et ultérieurement corriger, le fonctionnement du réseau.

Ces données peuvent inclure les affectations d'adresses, les voisins de protocole réseau, l'état opérationnel des interfaces et leur accessibilité.

2) Création automatique de tickets d'incident (Ticketing) :

Une interface vers un système de gestion des ***tickets d'incident*** permet au système d'orchestration d'automatisation de créer des tickets lorsque l'état du réseau diffère de celui enregistré dans la NSoT.

Dans un premier temps, la résolution des problèmes se fera manuellement, mais elle deviendra de plus en plus automatisée à mesure que l'organisation évolue.

✓ ***Un ticket d'incident*** :

C'est un enregistrement d'un problème ou d'un événement qui nécessite une action ou une réponse de la part de l'équipe de support technique ou de l'organisation responsable de la gestion de l'infrastructure.

Une fois qu'un ticket est ouvert, il est suivi et géré jusqu'à sa résolution, ce qui peut inclure la communication avec les parties concernées, la recherche de la cause du problème et la mise en œuvre d'une solution.

❖ ***La phase 3*** 3

Les éléments clés de cette phase comprennent un répertoire de gestion de code source, des flux de travail, des chatbots.

1) Répertoire de gestion de code source :

Un référentiel de code source permet de stocker, gérer et versionner des codes sources et des scripts utilisés pour la configuration et l'automatisation des équipements réseau.

En général, ces référentiels sont basés sur Git, qui est un système de contrôle de version distribué gratuit et open source [4], il permet aux développeurs de collaborer sur le même code et de gérer les conflits éventuels, en gardant une trace des modifications apportées au code au fil du temps.

Le référentiel de code source est étroitement intégré avec le système d'orchestration d'automatisation pour générer les configurations d'équipement réseau à partir des modèles stockés et des données NSoT. Par exemple, si nous souhaitons configurer tous les équipements réseau d'un centre de données, nous pouvons stocker les modèles de configuration, les configurations sauvegardées et les scripts dans le référentiel de code source, et utiliser l'orchestrateur d'automatisation pour générer les configurations pour tous les équipements en une seule fois.

2) Des flux de travail :

Dans la phase 3, l'accent est mis sur la transition des flux de travail manuels vers des flux de travail automatisés. Cela signifie que Les flux de travail peuvent être déclenchés à l'aide de scripts stockés dans le référentiel de code source.

Les produits commerciaux proposent souvent plusieurs mécanismes pour contrôler les flux de travail, notamment des éditeurs graphiques et des API.

3) Des chatbots :

Les chatbots jouent un rôle essentiel en permettant au système d'automatisation de transmettre des informations sur les flux de travail et l'état du réseau aux salons de discussion de communication intégrée, où les professionnels du réseau collaborent pour la mise en œuvre et le dépannage.

Cette fonctionnalité s'avère particulièrement efficace pour les équipes réseau distribuées, où les membres peuvent travailler à distance.

La phase 4 4

La phase 4 elle a pour objectif de fournir un retour d'information à partir du réseau pour aider à l'optimisation et à l'amélioration continue du système.

Jusqu'à présent, le réseau a fourni peu de commentaires, à l'exception des vérifications de validation par rapport aux données NSoT. La phase 4 introduit des mécanismes pour surveiller et collecter des données à partir du réseau.

Cela permet aux opérateurs de réseau de mieux comprendre le fonctionnement du réseau et d'identifier les problèmes potentiels. Les éléments de la phase 4 comprennent les suivants :

1) Télémétrie et surveillance :

Historiquement, la surveillance du réseau s'est appuyée sur le protocole SNMP (Simple Network Management Protocol), mais les implémentations plus modernes utilisent la télémétrie en continue (en temp réel). Les réseaux devront utiliser les deux mécanismes pendant un certain temps à venir.

La télémétrie permet aux équipes de surveillance du réseau d'avoir une vue complète de la santé du réseau et de détecter rapidement les anomalies ou les problèmes de performance.

2) Bases de données de surveillance et de gestion :

Les données de surveillance doivent être stockées dans un emplacement approprié, soit dans une base de données relationnelle pour les données de type relationnel telles que le type d'appareil et la liste des interfaces, soit dans une base de données de séries chronologiques pour les variables de performance des interfaces.

3) Déclencheurs d'action :

Pour que la surveillance du réseau soit réellement bénéfique, il est essentiel d'obtenir des réponses adéquates aux résultats obtenus. Les déclencheurs d'action utilisent des ensembles de règles ou des techniques d'apprentissage automatique pour détecter les anomalies, générer des alertes et ouvrir des tickets de dépannage.

Dans les mises en œuvre plus avancées, des workflows automatisés sont déclenchés pour initier la remédiation sans nécessiter d'intervention humaine, par exemple en redirigeant le trafic pour contourner une liaison défaillante. Cette automatisation permet d'améliorer l'efficacité des opérations réseau et de réduire les délais de résolution des problèmes.

❖ **La Phase 5** 5

C'est la dernière étape de l'architecture, elle consiste à automatiser les tests et la validation des changements. Cette phase comporte deux étapes :

1) Tests de réseau virtuel :

Cette phase consiste à tester les changements de réseau dans un environnement de laboratoire avant de les déployer dans un environnement de production réel. Cette étape utilise des dispositifs virtuels simulés par logiciel pour modéliser les paramètres clés du réseau de production.

Les changements proposés sont testés en instanciant le réseau virtuel, en exécutant des tests préalables au changement pour valider le fonctionnement du laboratoire tel qu'intentionné, en appliquant le changement et en effectuant des tests postérieurs au changement pour valider que le résultat souhaité a été obtenu.

2) Les tests de validation de changement :

Si les tests de réseau virtuel réussissent, le changement peut être appliqué au réseau de production. La mise en œuvre suit le même processus en trois étapes : la validation de l'état avant le changement, l'application du changement, puis la validation de l'état résultant après le changement.

I.1.4 Les avantages

L'automatisation des réseaux présente plusieurs avantages, notamment [5]:

❖ ***Réduction des coûts*** : L'automatisation simplifie les opérations en rendant les infrastructures moins complexes, ce qui permet de réduire considérablement le temps nécessaire pour configurer, approvisionner et gérer les services et le réseau. En consolidant

les services réseau, en optimisant l'utilisation de l'espace de travail et en mettant hors service les périphériques sous-utilisés, les entreprises peuvent réaliser des économies significatives en termes de personnel, d'énergie et de ressources.

- ***Augmentation de la main-d'œuvre stratégique*** : En automatisant les tâches répétitives sujettes aux erreurs humaines, les entreprises peuvent augmenter leur productivité, favorisant ainsi l'amélioration des activités et de l'innovation.
- ***Meilleure visibilité et contrôle du réseau*** : L'automatisation contribue à rendre les opérations informatiques plus réactives aux changements grâce à des analyses avancées. En obtenant une meilleure visibilité sur le réseau, les entreprises peuvent comprendre précisément ce qui se passe dans leur infrastructure, ce qui leur offre la capacité de contrôler et de s'adapter aux besoins changeants.

I.1.5 Types d'automatisation

Il existe différentes méthodes pour automatiser la configuration et la gestion d'un réseau informatique [6].

I.1.5.1 L'automatisation basée sur la ligne de commande (CLI) :

Au niveau le plus basique, les composants réseau peuvent être automatisés en utilisant des commandes et des arguments de ligne de commande standard. Par exemple, dans le cas du système d'exploitation Linux, les administrateurs peuvent utiliser les opérateurs Bash pour créer des chaînes d'événements conditionnels en fonction du succès ou de l'échec de la commande précédente (utilisant les opérateurs && et ||). Cela permet d'exécuter des actions spécifiques en fonction des résultats des commandes précédentes.

De plus, les utilisateurs peuvent compiler des listes de commandes dans des fichiers texte, également appelés scripts Shell. Ces scripts peuvent être exécutés de manière répétée et simultanée en une seule commande, ce qui permet d'automatiser des tâches complexes et récurrentes.

Cependant, il est important de noter que cette méthode d'automatisation requiert une connaissance approfondie des commandes et des protocoles réseau spécifiques à chaque appareil. Les administrateurs doivent être familiers avec les commandes et les fonctionnalités de chaque périphérique réseau, ce qui peut être fastidieux et source d'erreurs.

Malgré ces défis, l'automatisation basée sur la ligne de commande reste une méthode couramment utilisée dans de nombreux environnements réseau en raison de sa flexibilité et de sa capacité à automatiser des actions spécifiques de manière directe.

I.1.5.2 L'automatisation basé sur des logiciels :

Cette méthode consiste à utiliser des logiciels d'automatisation de réseau qui offrant une interface graphique pour configurer et gérer un ensemble de périphériques hétérogènes. Cette approche est plus conviviale et moins sujette aux erreurs, mais elle peut être plus complexe à

configurer et peut nécessiter des compétences supplémentaires en programmation (les taches à exécuter sous forme de code).

Les logiciels d'automatisation permettent de consolider plusieurs tâches réseau dans des programmes intégrés qu'on peut sélectionner, planifier et exécuter depuis la partie frontend de l'application.

En pratique, la plupart des entreprises utilisent une combinaison de configurations manuelles et d'outils d'automatisation pour atteindre un équilibre en termes de complexité, de contrôle et d'évolutivité.

I.1.6 Comparaison de quelques outils d'automatisation

Parmi les outils d'automatisation de réseau les plus populaires et couramment utilisés, on trouve : ***Ansible***, ***Chef***, ***Puppet.*** Ces outils offrent une solution efficace pour orchestrer et automatiser diverses activités, telles que la configuration des appareils réseau, le déploiement de services, et la résolution des problèmes. Le tableau ci-dessous résume un comparatif des solutions d'automatisation :

	Ansible	***Chef***	***Puppet***
Key Files defining actions	Playbook	Manifest	Recipe, Run-list
Communication protocole	SSH	HTTPS (Via REST API)	HTTPS (Via REST API)
Key port	22(SSH port)	8140	10002
Agent-based/Agentless	Agentless	Agent-based (Or Agentless)	Agent-based
Push/Pull	Push	Pull	Pull

Tableau I. 1 : Comparatif des solutions d'automatisation d'un réseau informatique

I.1.7 Le choix d'une solution d'automatisation

Dans le cadre de notre travail, nous avons décidé d'adopter Ansible comme solution pour l'automatisation. Nous allons donc définir et expliquer les principes de fonctionnement d'Ansible, puis exposer les raisons qui ont motivé notre choix de l'adopter.

I.1.7.1 Définition d'Ansible

Ansible est un outil Open Source d'automatisation informatique qui simplifie le provisionnement, la gestion des configurations, le déploiement des applications, l'orchestration et d'autres processus informatiques. Il fonctionne sur de multiples systèmes de type Unix et peut configurer à la fois des systèmes Unix et Microsoft Windows. Ansible propose un langage déclaratif intégré pour décrire la configuration du système.

L'une des caractéristiques clés d'Ansible est son approche sans agent, ce qui signifie qu'il n'est pas nécessaire d'installer de logiciel supplémentaire sur les nœuds gérés. Ansible se connecte temporairement à distance via SSH ou Windows Remote Management (pour exécuter des commandes à distance PowerShell) afin d'effectuer ses tâches.

Ce qui distingue Ansible des autres outils d'automatisation, c'est sa simplicité d'utilisation, sa sécurité et sa courbe d'apprentissage fluide. Grâce à une syntaxe intuitive et à une documentation complète, les utilisateurs peuvent rapidement maîtriser Ansible et l'utiliser pour automatiser diverses tâches opérationnelles [7].

I.1.7.2 Principe de fonctionnement

Le fonctionnement d'Ansible repose principalement sur deux types de machines :

1. ***Le serveur Ansible*** : également connu sous le nom de machine de contrôle ou de nœud de contrôle, il s'agit de la machine Ansible elle-même.
2. ***Les hôtes*** : aussi appelés nœuds, ce sont les machines sur lesquelles Ansible effectuera des tâches.

Ansible fonctionne en établissant une connexion SSH avec les hôtes ou les réseaux cibles et en y transférant de petits programmes appelés modules. Ces modules sont définis dans un fichier appelé le Playbook. Le nœud de contrôle s'appuie sur un fichier d'inventaire qui fournit la liste des hôtes sur lesquels les modules Ansible doivent être exécutés.

Lors de l'exécution, le serveur Ansible envoie les modules appropriés aux hôtes spécifiés dans le Playbook via la connexion SSH. Les modules sont alors exécutés sur les hôtes, ce qui leur permet d'effectuer les tâches définies dans le Playbook. Après l'exécution, le serveur Ansible collecte les résultats et les rapports des hôtes [7]

I.1.7.3 Le choix de cette solution

- ***Simplicité et facilité d'utilisation*** *:*

  Ansible se distingue par sa simplicité et son approche basée sur le langage YAML. Sa courbe d'apprentissage est plus douce par rapport à Chef et Puppet, ce qui facilite la prise en main et la rédaction de playbooks.

- ***Architecture sans agent*** *:*

  Contrairement à Chef et Puppet, Ansible ne nécessite pas l'installation d'un agent sur les nœuds distants. Cette architecture sans agent simplifie le déploiement initial et la configuration, tout en réduisant la complexité de gestion et les problèmes de sécurité liés à la présence d'agents supplémentaires.

- ***Large communauté et documentation complète*** *:*

  Ansible bénéficie d'une communauté active et engagée, ainsi que d'une documentation complète. On peut trouver facilement des ressources, des exemples et des réponses aux questions, ce qui facilite l'apprentissage et la résolution des problèmes éventuels.

- ***Gestion de la configuration claire et structurée :***

  Ansible permet de décrire l'état souhaité de l'infrastructure à l'aide de playbooks YAML. On peut orchestrer des tâches de configuration, de déploiement d'applications et de provisionnement d'infrastructures de manière claire et structurée, ce qui facilite la compréhension et la maintenance de l'environnement.

- ***Automatisation et extensibilité :***

  Ansible offre une large gamme de modules prêts à l'emploi qui permettent d'automatiser des tâches spécifiques telles que la gestion des serveurs, des réseaux et des bases de données. De plus, on peut étendre les fonctionnalités d'Ansible en créant ses propres modules et plugins personnalisés, ce qui offre une flexibilité et une adaptabilité accrues. [8]

La supervision des réseaux

La supervision des réseaux (ou monitoring) comprend un ensemble de protocoles matériels et logiciels informatiques permettant en temps réel de surveiller, analyser, rapporter et d'alerter les fonctionnements anormaux des systèmes informatiques.

Elle consiste à indiquer et/ou commander l'état d'un serveur, d'un équipement réseau ou d'un service software pour anticiper les plantages ou diagnostiquer rapidement une panne.

I.1.8 Le concept de la supervision :

Le but de la supervision de réseau est de surveiller le bon fonctionnement des réseaux. Ce concept est apparu dans les années 1980, lorsque la mise en place de réseaux informatiques a explosé dans les entreprises. La taille et l'hétérogénéité des réseaux posaient alors des problèmes de gestion et d'administration, augmentant les besoins en main-d'œuvre d'experts administrateurs.

C'est à cette époque que sont nées les premières réflexions sur un nouveau concept : la supervision. Cette dernière devait pouvoir s'adapter à des environnements hétérogènes, automatiser le contrôle des réseaux et générer un ensemble de statistiques offrant une meilleure vision du réseau, permettant ainsi d'anticiper ses besoins.

La supervision est définie comme l'utilisation de ressources réseau (matérielles ou logicielles) pour obtenir des informations sur l'utilisation et l'état des réseaux et de leurs composants (logiciels, matériels). Ces informations peuvent être utilisées pour gérer de manière optimale (si possible de manière automatique) le traitement des pannes et la qualité des réseaux (problèmes de surcharge). Elles permettent également d'anticiper les évolutions futures nécessaires.

La supervision est capable de diagnostiquer et de réparer la plupart des pannes. Si ce n'est pas le cas, elle alerte immédiatement les personnes concernées par l'incident. Elle est donc extrêmement réactive et représente un gain important en temps. De plus, en ayant une vision continue du réseau, elle anticipe souvent les problèmes futurs, ce qui est appelé la proactivité [9].

I.1.9 Type de surveillance et actions liées.

I.1.9.1 Types de surveillance

Les outils de supervision sont utilisés pour surveiller plusieurs aspects d'un système informatique, notamment [10] :

- ***Surveillance matérielle :***

  Il s'agit de surveiller l'activité des équipements physiques tels que les serveurs, les routeurs, les commutateurs, les disques durs, etc. Les outils de surveillance matérielle peuvent mesurer la charge, la température, l'utilisation des ressources, etc.

- ***Surveillance réseau*** :

  Il s'agit de surveiller le réseau informatique pour s'assurer qu'il fonctionne de manière efficace et sécurisée. Les outils de surveillance réseau peuvent mesurer la bande passante, la latence, le taux d'erreur, la qualité de service (QoS), les protocoles utilisés et la sécurité du réseau.

- ***Surveillance système*** :

  Il s'agit de surveiller les systèmes informatiques pour détecter les problèmes de performances, les erreurs, les pannes, etc. Les outils de surveillance système peuvent collecter des informations sur les journaux (logs) du système, les ressources système (CPU, mémoire, disque), les processus et les services en cours d'exécution.

I.1.9.2 Les actions liées

Les actions liées aux événements détectés par les outils de surveillance peuvent inclure [10] :

1. ***L'enregistrement dans un journal :***

   Chaque événement est enregistré dans un journal pour permettre une analyse ultérieure et pour aider à comprendre l'historique du système.

2. ***Le tracé graphique :***

   Les événements peuvent être tracés graphiquement pour permettre une visualisation plus claire de l'évolution des paramètres surveillés et pour faciliter l'analyse.

3. ***L'alerte :***

   Lorsqu'un événement est détecté, une alerte peut être envoyée sous forme de message électronique (e-mail, SMS, etc.) pour informer les responsables et les administrateurs système de la situation.

4. ***L'exécution de script :***

   Selon les règles prédéfinies, une action peut être exécutée automatiquement en réponse à un événement. Par exemple, un script peut être exécuté pour redémarrer un service ou effectuer une action corrective pour résoudre un problème.

I.1.10 La norme ISO du point de vue de la gestion des réseaux

L'ISO (Organisation internationale de normalisation) est impliqué dans la définition de normes pour la supervision et l'administration des systèmes depuis de nombreuses années. La norme ISO 7498/4, publiée en 1988, définit les principales fonctions que doivent implémenter les systèmes de supervision et d'administration. Ces fonctions sont les suivantes :

I.1.10.1 Gestion des performances

La gestion des performances, connue en anglais sous le nom de « *Performance Management* », est un processus qui permet d'évaluer et de mesurer le comportement des objets gérés. Ces objets peuvent être des systèmes informatiques, des réseaux ou des applications, entre autres. Le but de cette gestion est de garantir que le réseau est en mesure de répondre aux besoins des utilisateurs de manière continue.

Pour ce faire, des moyens de collecte d'informations statistiques sont mis en place. Cette collecte peut inclure des mesures telles que le trafic, le temps de réponse et le taux d'erreur. L'interprétation de ces mesures permet de calculer la charge du système et de prévoir toute défaillance ou dégradation des performances. Ces prévisions sont utilisées pour planifier les évolutions du système et pour mettre en place des outils de modélisation et de simulation.

La gestion des performances implique également le stockage et l'archivage des données collectées, afin de pouvoir les consulter ultérieurement si nécessaire. Les outils et les techniques utilisés dans ce processus sont donc essentiels pour garantir une performance optimale du système [11].

I.1.10.2 Gestion des configurations

La gestion de la Configuration, également connue en anglais sous le nom de « *Configuration Management* », est un processus qui vise à maintenir un inventaire précis des ressources logicielles et matérielles. Cela inclut les versions de logiciels, les licences associées, les fonctions, les types d'équipement, ainsi que la localisation géographique de chaque objet géré.

Un aspect important de cette gestion est d'associer un nom unique à chaque objet de l'inventaire, ce qui permet d'identifier et de suivre facilement chaque élément. Cette approche facilite également la gestion de changements et l'identification des composants à mettre à jour ou à remplacer en cas de besoin.

En outre, la gestion de la Configuration permet également de suivre les relations entre les différents composants, de sorte que les effets de tout changement puissent être évalués avant d'être mis en œuvre. Cela contribue à minimiser les risques d'erreurs et à améliorer la stabilité et la fiabilité du système [11].

I.1.10.3 Gestion de la sécurité

La gestion de la sécurité, également connue en anglais sous le nom de « *Security management* », est un domaine important qui vise à protéger les données et les objets gérés contre les menaces externes et internes, ainsi qu'à prévenir les pertes de données et les violations de la confidentialité.

Les mécanismes d'authentification des extrémités, de cryptage et de contrôle d'accès sont quelques-unes des techniques de sécurité utilisées pour assurer l'intégrité des données et des objets gérés. L'ISO, une organisation internationale de normalisation, définit cinq services de sécurité qui sont les suivants :

- ***Contrôle d'accès au réseau*** : qui permet de contrôler l'accès aux réseaux et aux systèmes d'information.
- ***Confidentialité*** : qui garantit que les données ne sont communiquées qu'aux personnes et processus autorisés.
- ***Intégrité*** : qui assure que les données ne sont pas modifiées ou détruites accidentellement ou intentionnellement.
- ***Authentification*** : qui garantit que l'entité participant à la communication est bien celle qu'elle prétend être.
- ***Non-répudiation*** : qui empêche une entité de nier sa participation à une communication.

La gestion de la sécurité implique donc des processus de surveillance, de prévention, de détection et de réponse aux incidents de sécurité. Les professionnels de la sécurité informatique utilisent une variété de techniques et d'outils pour protéger les systèmes d'information et les données, tels que les pares-feux, les antivirus, les systèmes de détection d'intrusion, les systèmes de gestion des vulnérabilités et les audits de sécurité [11].

I.1.11 Le protocole SNMP

Les réseaux informatiques se sont rapidement développés, rendant leur gestion manuelle de plus en plus difficile. Les administrateurs réseau ont donc eu besoin d'un moyen de surveiller les performances des différents équipements réseau et de gérer les erreurs et les pannes à distance. C'est ainsi qu'ils ont envisagé la création d'un protocole standardisé permettant de répondre à ces besoins en matière de gestion des équipements réseau.

I.1.11.1 Présentation

Le SNMP (Simple Network Management Protocol) est un protocole de la couche application qui permet d'échanger des informations de gestion entre les dispositifs du réseau. Il a été défini par l'IAB (Internet Architecture Board) dans la RFC 1157 pour remplacer le protocole CMIP (Common Management Information Protocol) proposé par l'ISO (International Organization for Standardization) pour la communication entre les manageurs et les agents lors de la supervision. Le protocole CMIP était considéré comme trop complexe à implémenter, c'est pourquoi l'IAB a introduit le SNMP qui se veut être le plus simple possible comme son nom l'indique [12].

I.1.11.2 Architecture et Fonctionnement du protocole SNMP

L'architecture de gestion du réseau proposée par le protocole SNMP fonctionne sur un modèle client-serveur.

Le client correspond à la station de gestion de réseau, souvent appelée Manager ou encore Network Management Station (*NMS*) par certains éditeurs. Les serveurs correspondent aux agents SNMP qui enregistrent en permanence des informations les concernant dans leur MIB. La station interroge les MIB des différents agents pour récupérer les informations qu'elle souhaite.

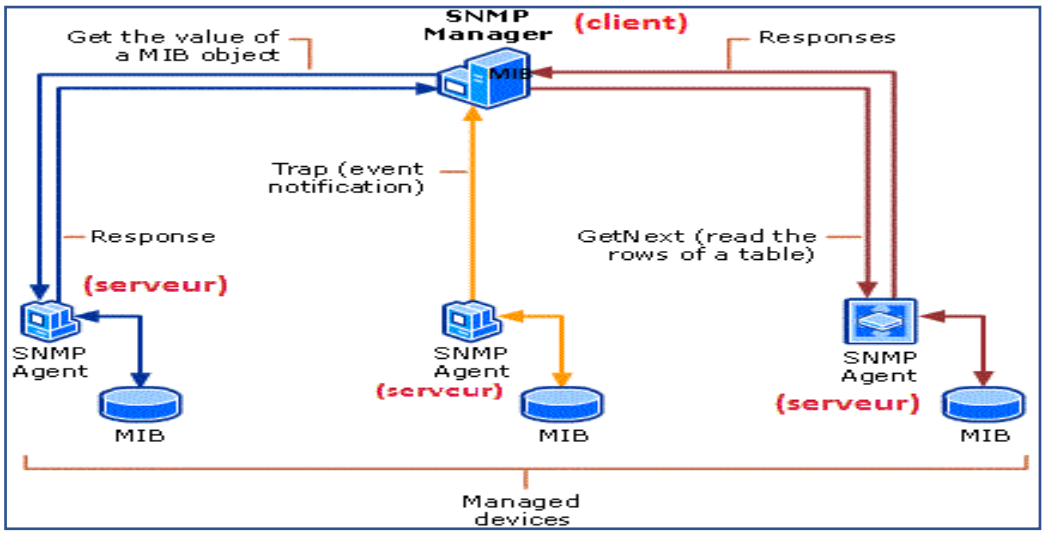


Figure I. 2 : Communication Client/serveur avec Les agents SNMP.

Un système SNMP supporte trois types de requêtes :

- **Get** : trois sous-requêtes sont comprises dans cette famille :
  - ***GetRequest*** : permet aux stations de gestion NMS (*Network Management Station)* d'interroger les objets et les variables gérées par la MIB des agents.
  - ***GetNextRequest*** : permet aux NMS de balayer les tables de la MIB.
  - ***GetResponse* :** est le message retourné par les agents aux commandes *'GetRequest'', ''GetNextRequest''* et *''SetRequest''* de la NMS.
- **Set** : la commande *''SetRequest''* permet à la NMS de modifier la valeur d'un objet de la MIB et de lancer le périphérique.
- **Trap :** c'est une alarme envoyée lors de la détection d'une anomalie. En effet, elle permet à un agent de notifier un évènement.

Le protocole SNMP sert à établir le dialogue entre les agents installés sur les machines supervisées et le client de supervision ou le manageur.

L'agent reçoit les requêtes sur le port 161 et le superviseur reçoit les alertes sur le port 162. L'échange se déroule de la manière suivante :

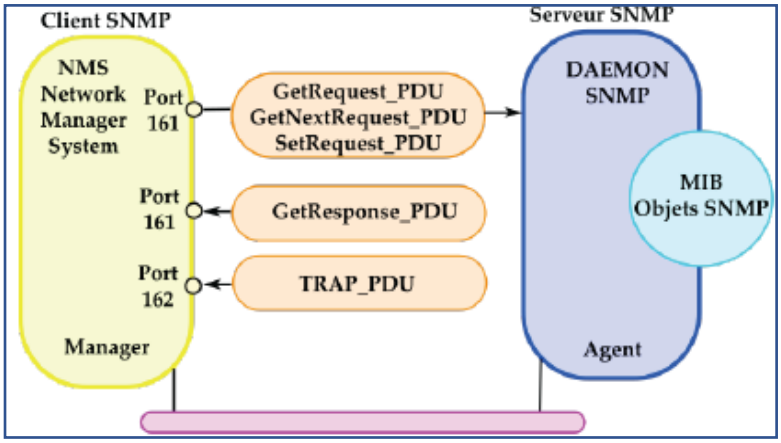


Figure I. 3 : Scenario de fonctionnement dans un système SNMP.

- Quand le manageur veut interroger l'agent ou lui donner une instruction, il envoie une requête à l'agent. Celui-ci la traite et renvoie une réponse au manageur.

- Quand un événement se produit sur l'élément du réseau surveillé par l'agent, ce dernier informe immédiatement le manageur par une alerte de type trap ou inform. Dans le cas d'un inform, le serveur envoie une réponse à l'agent émetteur.

I.1.11.3 La gestion des réseaux avec le protocole SNMP

L'ISO s'est focalisé sur la définition de la structure du système de gestion de réseau (Network Management System).

Dans ce cadre, l'ISO préconise l'installation d'un agent de gestion sur chaque machine supervisée [13], ce qui est illustré dans la figure ci-dessous :

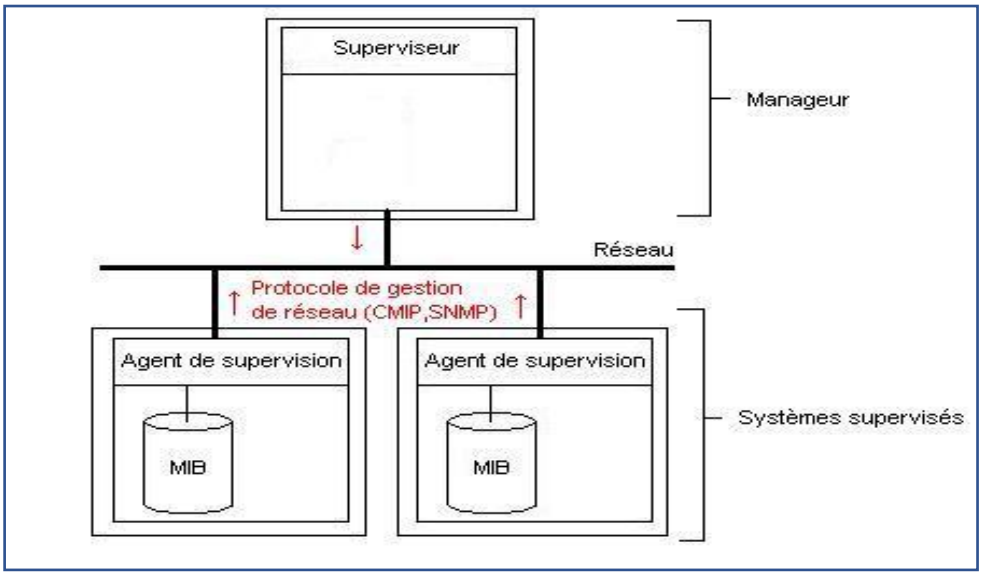


Figure I. 4 : Structure de gestion des réseaux.

Cette structure contient divers éléments à savoir les agents, les managers, la MIB et le protocole de gestion (CMIP, SNMP).

I.1.11.3.1 Les agents

Les agents sont des composants logiciels installés dans les équipements que nous souhaitons surveiller et gérer, tels que des routeurs, des ordinateurs ou des serveurs. Ils collectent périodiquement des informations sur la machine sur laquelle ils sont exécutés et les stockent localement. Si un problème est détecté, ils envoient une notification au service de gestion centralisé.

I.1.11.3.2 Le manageur

Le manageur (également appelé administrateur ou superviseur) est celui qui souhaite obtenir des informations spécifiques sur l'état du réseau. Pour ce faire, il envoie des requêtes aux agents installés sur les équipements actifs afin d'obtenir un état complet de la machine, et donc de l'ensemble du réseau.

En fonction de la réponse reçue, le manageur prend une série de décisions (actions) pour remédier à tout problème détecté. Dans de nombreux cas, ces décisions sont transmises à l'agent installé sur la machine en difficulté. L'agent exécute alors les actions correctives demandées par le manageur afin de rétablir l'état normal de la machine.

I.1.11.3.3 La MIB (Management Information Base)

La MIB (Management Information Base) représente une base de données qui stocke toutes les informations que le gestionnaire doit connaître pour superviser un équipement. Chaque agent possède sa propre MIB, ce qui signifie qu'il y a une MIB pour chaque équipement à superviser.

La MIB est organisée sous forme d'une structure arborescente (Figure I.3) avec chaque nœud étant identifié par un OID (Object Identifier) unique. La structure et les appellations des rubriques sont normalisées pour faciliter la lisibilité. Cependant, SNMP (Simple Network Management Protocol) utilise uniquement l'index numérique pour accéder à chaque niveau de la hiérarchie de la MIB. La MIB contient des informations consultables, des paramètres modifiables et des alarmes à émettre en cas de défaillance [13].

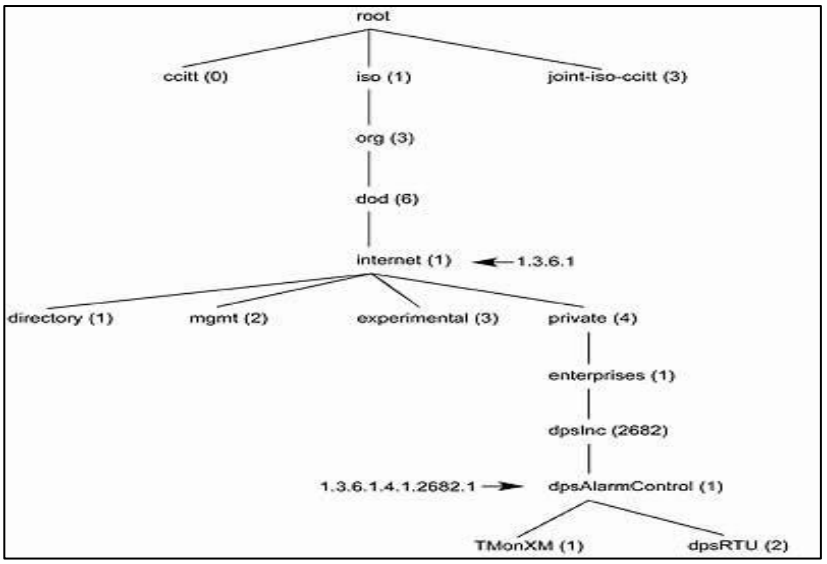


Figure I. 5 : Structure arborescente de la MIB.

Pour accéder aux variables souhaitées, on utilisera l'OID (Object Identifier) qui désigne l'emplacement de la variable à consulter dans la MIB. On aura par exemple sur une machine l'OID 1.3.6.1.4.1.2682.1.2 qui désigne la variable **dpsRTU**

I.1.11.4 Versions du protocole SNMP :

SNMPv1 est la première version du protocole SNMP. Cette version est décrite dans les RFC 1065 à 1067 et 1155 à 1157. À l'époque de son développement, les normes Internet et la sécurité ne bénéficiaient pas de la même attention qu'aujourd'hui. SNMPv1 utilise plusieurs protocoles de communication tels que UDP, IP, CLNS (ConnectionLess Network Service), DDP (AppleTalk Datagram-Delivery Protocol) et IPX (Novell Internet Packet Exchange).

Cependant, SNMPv1 présente une faiblesse en termes de sécurité, car il utilise un mécanisme d'authentification basé sur la transmission d'une "chaîne" (c'est-à-dire un mot de passe) en texte clair, ce qui est considéré comme peu sûr. Cette vulnérabilité rend SNMPv1 vulnérable aux attaques et compromet la confidentialité des informations échangées [14].

SNMPv2 est défini dans les RFC 1441 et RFC 1452. Cette version apporte plusieurs améliorations par rapport à SNMPv1, notamment en termes de performances, de sécurité et de confidentialité.

L'une des améliorations clés de SNMPv2 réside dans son mécanisme de sécurité amélioré. Il introduit un modèle de sécurité basé sur les parties, offrant un niveau supplémentaire de contrôle d'accès et de confidentialité des données. Cependant, certains utilisateurs ont perçu ce système de sécurité comme étant complexe, ce qui a limité son adoption et sa popularité. SNMPv2 propose également de nouvelles fonctionnalités pour améliorer la communication entre les gestionnaires.

L'ajout de la commande GetBulkRequest permet de récupérer de grandes quantités de données en une seule requête, ce qui évite d'avoir à effectuer des requêtes itératives avec GetNextRequest pour obtenir des données volumineuses.

Malgré ces améliorations, SNMPv2 n'a pas connu une adoption généralisée en raison de préoccupations liées à sa complexité en matière de sécurité. Cela a conduit au développement ultérieur de SNMPv3, qui a cherché à résoudre ces problèmes et à fournir un niveau de sécurité plus robuste et convivial [14].

SNMPv3 : c'est la version la plus récente et la plus avancée de SNMP. Elle offre des fonctionnalités améliorées telles que la sécurité renforcée, la confidentialité des données, la prise en charge de l'authentification de l'utilisateur et la notification de trappe en temps réel.

Cette version de SNMP est plus complexe à mettre en œuvre que les versions précédentes, mais elle offre une sécurité et une fiabilité accrues pour la surveillance du réseau.

I.1.12 Solution de supervision

Les solutions de supervision sont des logiciels ou des outils qui facilitent la mise en place d'une surveillance en collectant des données sur les performances du système et en les analysant pour détecter les problèmes.

Il existe une multitude de solutions de supervision réseau disponibles, offrant des fonctionnalités et des avantages variés en fonction des besoins spécifiques de chaque entreprise ou organisation. Parmi les options les plus populaires, on trouve ***Zabbix***, ***Nagios***, ***Observium***, ***Centreon***, et bien d'autres encore. Ces plateformes offrent des capacités étendues de surveillance et de gestion du réseau, permettant ainsi de garantir un fonctionnement optimal.

I.1.13 Le Choix d'une solution de supervision

Pour cette partie de supervision nous avons décidé d'adopter Zabbix comme solution. Nous allons donc présenter et expliquer les principes de fonctionnement de zabbix, puis exposer les raisons qui ont motivé notre choix de l'adopter.

I.1.13.1 Présentation de Zabbix

Zabbix est un outil open source de supervision offrant des vues graphiques générées par RRDtool. Le serveur Zabbix est divisé en trois parties : le serveur de données, l'interface de gestion et le serveur de traitement. Chacune de ces parties peut être installée sur une machine différente pour répartir la charge et améliorer les performances.

De plus, un agent Zabbix peut être installé sur des hôtes Linux, UNIX et Windows pour collecter des statistiques telles que la charge CPU, l'utilisation du réseau, de l'espace disque, du processeur et de la mémoire.

Cela en fait un outil complet offrant des fonctionnalités de supervision telles que des alertes sur seuil, des mesures et des actions en fonction de certaines conditions. Le logiciel prend également en charge la supervision via SNMP et permet de configurer des proxies Zabbix pour répartir la charge ou garantir une meilleure disponibilité de service. [15] [16].

I.1.13.2 Principe de fonctionnement

- **Architecture :**

  Zabbix suit une architecture client-serveur. Le serveur Zabbix est responsable de la collecte, du traitement et du stockage des données de surveillance, tandis que les agents Zabbix sont installés sur les hôtes à surveiller.

- **Configuration** :

  La première étape consiste à configurer le serveur Zabbix. Il est nécessaire de définir les hôtes à surveiller, les paramètres de surveillance souhaités, les seuils d'alerte, etc. Cette configuration se fait à travers l'interface utilisateur web de Zabbix.

- **Installation de l'agent** :

Pour surveiller les hôtes, il est nécessaire d'installer l'agent Zabbix. Cet agent a pour fonction de collecter et d'envoyer les données de surveillance vers le serveur Zabbix.

- **Collecte de données** :

L'agent Zabbix collecte des données à partir des hôtes surveillés, telles que les performances du système, les statistiques réseau, l'utilisation des ressources, etc. Ces données peuvent être personnalisées en fonction des besoins spécifiques de surveillance.

- **Envoi de données** :

L'agent Zabbix envoie les données collectées au serveur Zabbix à intervalles réguliers. Ces données sont stockées dans la base de données de Zabbix pour l'analyse ultérieure.

- **Traitement des données** :

Le serveur Zabbix traite les données reçues. Il vérifie si les seuils d'alerte sont dépassés ou si des événements anormaux se produisent.

Si c'est le cas, il déclenche des actions telles que l'envoi d'alertes ou l'exécution de scripts personnalisés.

- **Interface utilisateur** :

Zabbix offre une interface utilisateur Web conviviale qui permet de visualiser les données de surveillance, de configurer des tableaux de bord, de générer des rapports, et bien plus encore. Il est également possible de définir des actions de notification pour recevoir des alertes par e-mail, SMS, et autres moyens de communication.

I.1.13.3 Le choix de cette solution

- ***Fonctionnalités et flexibilité*** :

Zabbix offre une large gamme de fonctionnalités de surveillance, notamment la surveillance des ressources réseau, des serveurs, des applications, des bases de données, etc.

Il permet également une personnalisation avancée grâce à son langage de configuration flexible. Nagios est également une solution populaire et offre des fonctionnalités de surveillance similaires, mais sa configuration est généralement considérée comme plus complexe.

- ***Évolutivité*** :

Zabbix est connu pour sa capacité à gérer de grands environnements de surveillance avec des milliers de périphériques.

Il offre une architecture distribuée qui permet de répartir la charge entre plusieurs serveurs. Nagios peut également être mis à l'échelle, mais sa mise en place et sa configuration nécessitent souvent plus d'efforts pour les déploiements à grande échelle.

- ***Interface utilisateur*** :

Zabbix propose une interface utilisateur moderne et conviviale avec des tableaux de bord personnalisables, des graphiques interactifs et une navigation intuitive.

Nagios a une interface plus traditionnelle et nécessite souvent une configuration plus avancée pour obtenir une expérience utilisateur similaire.

- ***Communauté et support*** :

Zabbix dispose d'une communauté active et propose une documentation détaillée, des forums de discussion et un support professionnel payant.

Conclusion

Pour conclure, il est essentiel d'automatiser et de surveiller les réseaux afin d'assurer la fiabilité et les performances des systèmes informatiques. Nous avons identifié qu'en utilisant des outils tels qu'Ansible pour l'automatisation et Zabbix pour la supervision, il est possible de progresser considérablement vers cet objectif.

Ensemble, ces outils offrent une approche proactive de la gestion des réseaux, permettant aux entreprises de maximiser l'efficacité de leurs systèmes tout en réduisant les coûts et les temps d'arrêt.

Cependant, il est important de souligner que l'automatisation et la supervision des réseaux ne remplacent pas la nécessité d'une surveillance humaine continue.

Les professionnels de l'informatique doivent travailler en étroite collaboration avec les systèmes automatisés pour garantir une gestion optimale des réseaux

Chapitre II: Présentation de l'organisme d'accueil

Introduction

Nous présentons dans ce chapitre le campus NTS (New Technologies & Solutions) lieu de notre stage. Dans un premier temps, nous allons donner des généralités sur l'entreprise pour mieux comprendre sa structure et ses objectifs. Nous étudierons ensuite l'architecture réseau de cette entreprise et ses composantes afin de pouvoir suggérer d'éventuelles améliorations.

Présentation générale de l'entreprise "Campus NTS "

II.1.1 Création et évolution :

NTS qui est l'acronyme de New Technology & Solutions est une jeune entreprise a été créée en 2020 à Bejaia, axée sur la recherche, la conception et la mise en œuvre des solutions, d'intégration de système de sécurité, l'importation et la distribution des équipements et des matériels de sécurité pour les réseaux et des télécommunications, ainsi que la formation et le conseil.

II.1.2 La situation géographique de l'entreprise "Campus NTS "

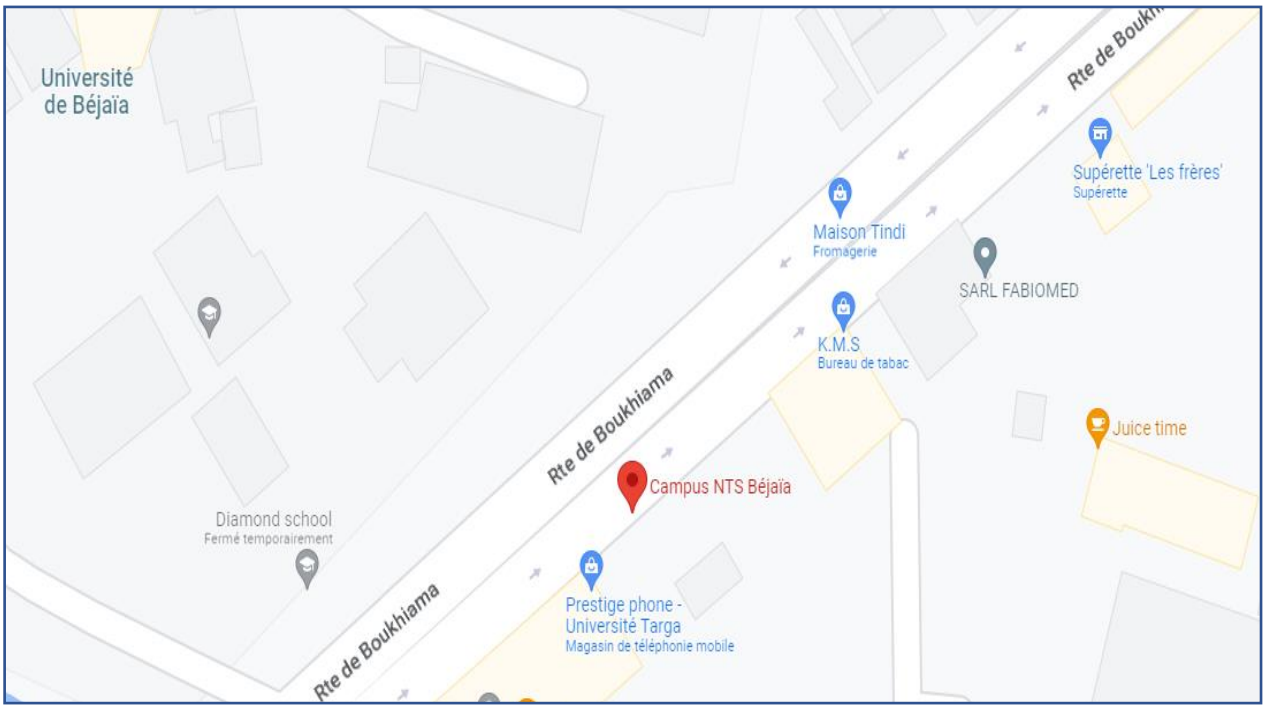


Figure II. 1 : Localisation de l'entreprise NTS.

II.1.3 Organigramme de l'entreprise

La figure III.2 ci-dessus nous donne une vue générale sur les différents organes constituant le Campus NTS, son organigramme est défini comme suit :

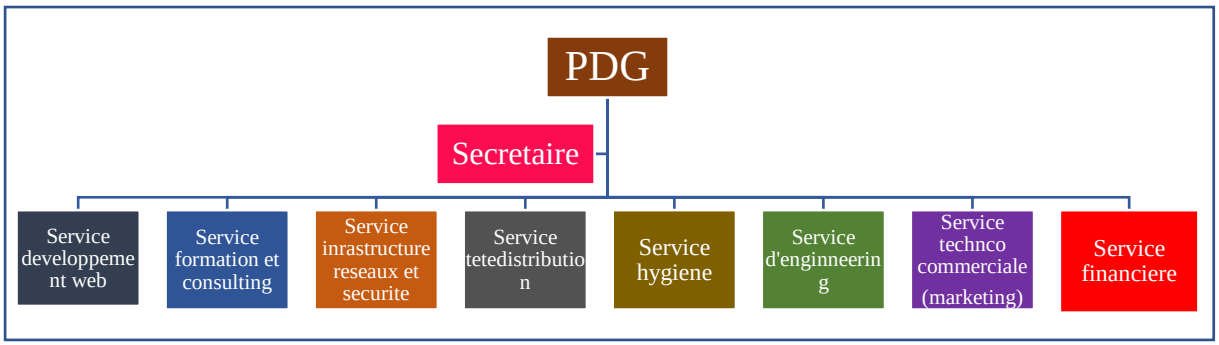


Figure II. 2 : L'organigramme de campus NTS.

II.1.4 Les activités, les missions et les objectifs de Campus NTS

II.1.4.1 Les activités

- ❖ Proximité et réactivité
- ❖ Conseil et étude
- ❖ Solution personnalisée
- ❖ Délai des réalisations optimisées
- ❖ Qualité conforme aux standards
- ❖ Support technique
- ❖ Formation après la réalisation

II.1.4.2 Les missions

Développer, exploiter et gérer des réseaux de télécommunications publics et prives.

Garantir l'installation d'une infrastructure réseau sécurité pour les clients

II.1.4.3 Les objectifs

Les clients sont satisfaits d'une production de haute qualité et d'un service sécurisé à toute épreuve avec diverses solutions et produits en un temps record et a des prix raisonnables, aussi d'atteindre, économique et social pour se :

- ✓ Maintenir durablement comme leader dans son domaine en environnement compétitif.
- ✓ Développer sa présence internationale.
- ✓ Participer à la promotion de la société algérienne de l'information.

Présentation du service d'accueil (département informatique) :

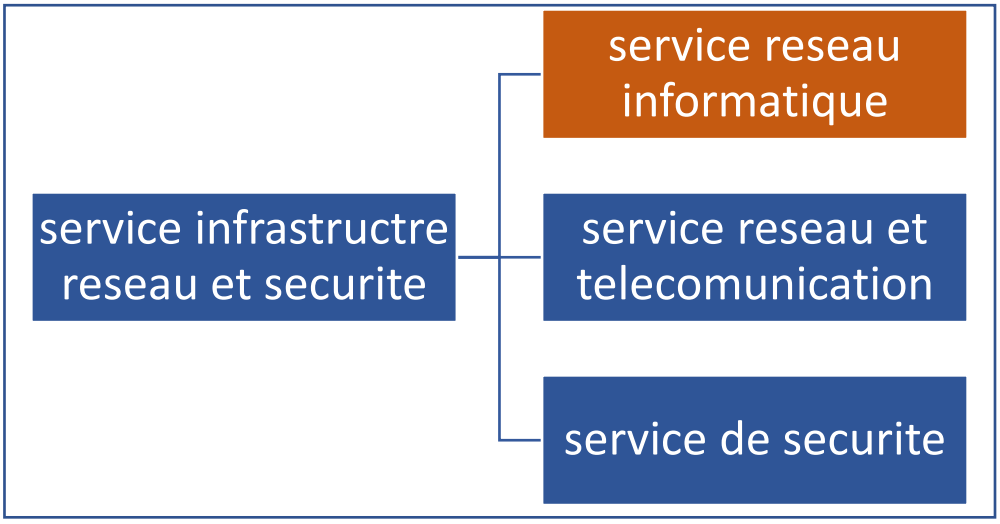


Figure II. 3 : Organigramme de service d'accueil.

❖ **Service infrastructure réseau et sécurité :**

L'infrastructure réseau est au cœur des opérations commerciales dans la plupart des industries. Il peut être considéré comme le centre sensible de toute l'organisation informatique car il centralise les données, simplifie les échanges de données et facilite la communication entre les employés.

A ce titre, il est un outil important pour le fonctionnement normal de l'entreprise et nécessite une attention constante en matière de sécurité. Ceci afin d'empêcher le nombre croissant et l'affectation des attaques externes et internes.

❖ **Service réseau informatique :**

Ce service représente tous les appareils et périphériques au sein d'une entreprise qui sont physiquement ou virtuellement connectés les uns aux autres à partir d'un wifi professionnel ou d'autre méthodes afin de partager des ressources ou des informations.

En fait, l'infrastructure réseau offre un large éventail de fonctionnalités pour les clients de services et les producteurs de services, par exemple : limitation de débit, analyse, vérification, surveillance et enregistrement et sécurisation du réseau de cette entreprise.

❖ **Service réseau et Télécommunication :**

Les services de télécommunications sont conçus pour transmettre des informations en un temps réel (synchronisation des informations) sous forme analogique ou numérique à l'exception de la radio et de la télévision. La plupart de ces services peuvent aider les clients à identifier leurs besoins en matière d'infrastructure de télécommunications. Voici des exemples de services d'infrastructure de télécommunications :

- ✓ Pose de fibre optique.
- ✓ Emplacement du site de la tour cellulaire.

- ✓ Test d'antenne radio.
- ✓ Installation d'équipements téléphoniques standards et réseau de données.

❖ **Service de sécurité :**

Cette entreprise NTS accomplit à la fois le gardiennage et l'installation des systèmes de sécurité électroniques.

Aussi elle fournit aux clients des solutions complètes et fiables pour protéger leurs ressources. Les services qu'elle réalise sont les suivants :

- ✓ Caméras de surveillance.
- ✓ Alarme anti- intrusion.
- ✓ Pointeuse et Contrôles d'accès.
- ✓ Vidéophonie.

Présentation du réseau de client

L'architecture réseau du client est basée sur un modèle hiérarchique à trois couches, conçu pour assurer des ***performances***, une ***haute disponibilité*** et une ***sécurité*** optimale de son infrastructure réseau, comme illustré dans la figure III.4.

La couche cœur est constituée de deux commutateurs de niveau 3 à haut débit, tels que le Cisco Catalyst 6800-24PS (L3). Ces deux équipements sont interconnectés par des liens d'agrégation, permettant d'améliorer la disponibilité et d'assurer la continuité du service aux utilisateurs finaux.

La deuxième couche, appelée couche de distribution, est constituée de deux commutateurs de niveau 3 capables de gérer des réseaux à grande échelle. Cette couche joue un rôle crucial en agissant comme une frontière de contrôle entre les couches d'accès et de cœur.

La couche d'accès est constituée de commutateurs de niveau 2 Cisco Catalyst 2960-24PS (L2), assurant la connectivité entre les postes de travail.

Les trois couches du réseau sont contrôlées et protégées par des équipements de niveau 4, tels que deux pares-feux de type FortiGate. Ces deux équipements sont également reliés entre eux par des liens d'agrégation, assurant ainsi la continuité du service aux utilisateurs finaux.

L'architecture réseau du client dispose également d'une zone démilitarisée (DMZ), qui est une zone isolée du réseau principal. Les équipements de la DMZ sont protégés par un autre pare-feu FortiGate, qui contrôle les accès entrants et sortants pour améliorer la sécurité du réseau et protéger les données sensibles.

Enfin, un Datacenter est également présent, représentant le noyau central du réseau de l'entreprise, où sont placés les serveurs de l'entreprise.

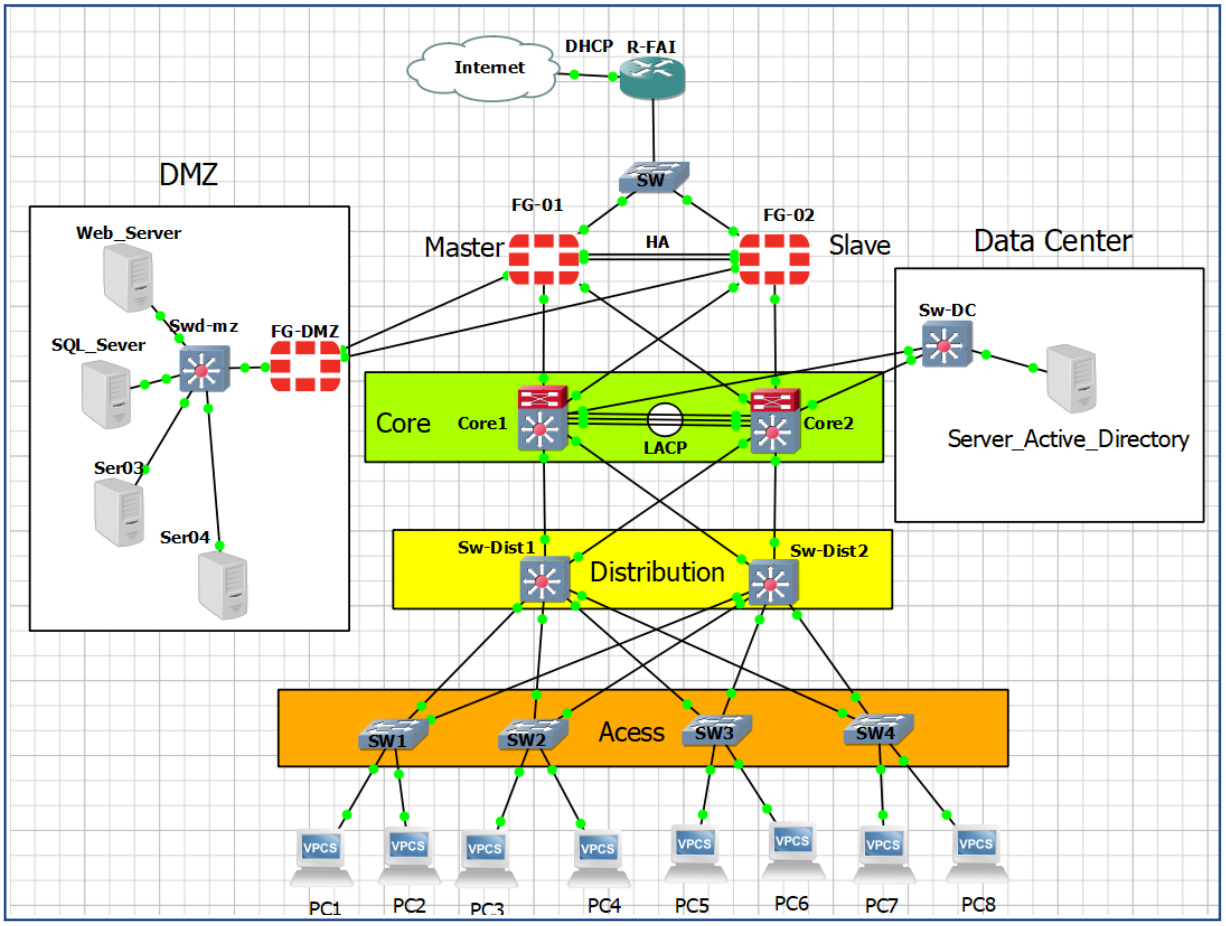


Figure II. 4 : Architecture physique du réseau de client.

❖ Le modèle hiérarchique utilisé dans le réseau du client nous a permis de garantir les services suivants :

✓ ***Agrégation de ports*** :

Il s'agit d'une technique qui permet de regrouper plusieurs ports physiques d'un équipement en un seul lien logique, ce qui permet d'augmenter la bande passante disponible entre deux équipements. Cette technique est particulièrement utile dans les liaisons entre des équipements de couche d'accès et des équipements de couche de distribution ou de cœur, où des volumes de données importants doivent être transmis rapidement.

✓ ***Performances*** :

La performance du réseau se réfère à sa capacité à transférer des données à grande vitesse et de manière fiable. Pour le réseau hiérarchique du client, les performances sont améliorées en utilisant des équipements de haute qualité tels que des commutateurs de niveau 3 à haut débit, des liens d'agrégation et des politiques de gestion de la bande passante.

Les performances d'un réseau ont un impact direct sur la qualité de service offerte aux utilisateurs finaux.

- ✓ ***Haute disponibilité*** :

La haute disponibilité se réfère à la capacité du réseau à fonctionner sans interruption, même en cas de défaillance d'un équipement ou d'une connexion.

Dans le cas du réseau du client, la haute disponibilité est assurée en utilisant des équipements redondants, des liens d'agrégation pour assurer une continuité de service en cas de panne d'un équipement.

- ✓ ***Sécurité*** :

La sécurité d'un réseau fait référence à sa capacité à protéger les données et les utilisateurs contre les attaques malveillantes, les intrusions et les violations de la confidentialité.

Dans l'architecture hiérarchique du réseau du client, la sécurité est renforcée par l'utilisation d'équipements de niveau 4 tels que des pares-feux.

Ces équipements offrent aussi des fonctionnalités de routage, telles que le routage inter-vlan, le routage statique et dynamique, ainsi que des services tels que le DHCP, Haute disponibilité, etc...

Dans le cas de notre client le routage inter-vlan, routage statique est service DHCP sont configurer au niveau des Firewall, et cela permettra d'avoir un contrôle total sur le réseau tel que les paquets entrent et sortent ver internet.

Ils assurent également une protection contre les accès non autorisés, la détection des menaces potentielles et la sécurisation des données sensibles.

- ❖ **La DMZ** :

La DMZ du client est une zone qui permet de séparer les serveurs accessibles depuis Internet de ceux qui ne le sont pas, en les isolant du reste du réseau du client.

Cette mesure de sécurité vise à réduire les risques d'attaques externes sur les données sensibles et les équipements du réseau principal.

En plaçant les serveurs accessibles depuis Internet dans une DMZ, le client peut appliquer des règles de sécurité plus restrictives et contrôler plus finement les accès entrants et sortants, améliorant ainsi la sécurité du réseau.

- ❖ **Le datacenter :**

Le datacenter du réseau du client héberge des serveurs, du stockage de données et d'autres équipements nécessaires au fonctionnement de son réseau. Et cela pour fournir des environnements contrôlés et sécurisés, avec des systèmes de refroidissement et de sécurité redondants pour garantir une disponibilité maximale des services.

- ❖ **Les Ordinateurs** :

Tous les ordinateurs du client sont configurés en mode DHCP, ce qui permet une configuration réseau rapide et facile sans nécessiter de configuration manuelle fastidieuse.

Le serveur DHCP est fourni par le pare-feu FortiGate, offrant ainsi un emplacement centralisé pour la gestion et la configuration des paramètres DHCP pour l'ensemble du réseau.

❖ **La liste des Vlan :**

Le tableau II.1 présente la liste des VLAN utilisés dans le réseau du client :

Nom de VLAN	ID de VLAN
Resource humain	100
Informatique	101
Marketing	102
Comptabilité	103
Finance	104
Management	105
VOIP	106
Datacenter	107
Native	999

Tableau II. 1 : Liste des Vlan

Présentation de l'environnement hard et soft

Le tableau II.2 résume le matériel et les logiciels utilisés dans le réseau du client.

Équipement	Le hardware (hard)	Software (soft)
Routeur	ISR 4331	IOS (Internetwork Operating System)
Pare-feu	FortiGate	Linux
Switch	Cisco Catalyst 6800-24PS (L3) Cisco Catalyst 2960-24PS (L2)	IOS (Internetwork Operating System)
Server	HP ProLiant DL380Pgeneration 10	Windows server 2022
Client	DELL IAER 35 R	Windows 10

Tableau II. 2: L'environnement hardware et le software.

Problématiques et Solutions proposées

II.1.5 Problématique

Au cours de notre stage chez l'entreprise NTS à Bejaia, nous avons constaté que le réseau de leur client présentait de nombreux équipements difficiles à gérer, à configurer et à surveiller individuellement et manuellement. Cette situation a conduit à plusieurs problèmes, notamment :

- Une visibilité limitée sur l'état du réseau, ce qui rend difficile la détection rapide des incidents, des pannes et des ralentissements. Cela a un impact négatif sur la disponibilité et les performances du réseau.
- Des opérations manuelles répétitives pour gérer et configurer les nombreux équipements du réseau, qui peuvent entraîner des temps d'arrêt et nuire à la productivité des utilisateurs.

- Des risques d'erreurs manuelles, qui peuvent compromettre la cohérence et la fiabilité du réseau.
- Des risques de sécurité élevés, car l'absence d'un système de surveillance adapté peut mettre en danger la confidentialité et l'intégrité des données et des informations.

II.1.6 Solution proposée

Après avoir identifié les problèmes liés à la gestion et à la configuration des équipements réseau du client, nous avons proposé la mise en place d'une solution complète pour automatiser et superviser le réseau. Cette solution est basée sur deux éléments principaux : un serveur d'automatisation, un serveur de supervision.

Les deux serveurs d'automatisation et de supervision seront ajoutés aux serveurs d'entreprise déjà présents à l'intérieur du datacenter, afin de bénéficier de plusieurs avantages. Comme par exemple, les serveurs seront physiquement protégés et sécurisés grâce aux mesures de sécurité en place, ce qui réduira les risques de vol, de sabotage et de dommages causés par des catastrophes naturelles telles que les incendies ou les inondations.

En outre, le datacenter offre des services tels que l'alimentation électrique de secours, la climatisation et la connectivité réseau à haute disponibilité, garantissant des performances optimales pour les serveurs et réduisant les temps d'arrêt.

La configuration du système d'exploitation Linux sur les deux serveurs permettra d'installer les outils Ansible et Zabbix sur des serveurs distincts, ce qui améliorera la disponibilité, les performances et la sécurité du réseau. En répartissant la charge de travail entre les deux serveurs, en cas de défaillance d'un serveur, l'autre pourra continuer à fonctionner sans interruption de service.

Le serveur d'automatisation sera configuré pour gérer les équipements du réseau de manière centralisée et automatique. Les tâches répétitives, telles que la configuration, les mises à jour et les sauvegardes, seront effectuées de manière plus efficace et sans risque d'erreurs manuelles. Cela réduira le temps d'arrêt et augmentera la productivité des utilisateurs. Tandis que Le serveur de supervision collectera les informations sur l'état du réseau en temps réel.

Les administrateurs pourront ainsi avoir une vue complète de l'état du réseau et recevoir des alertes automatiques en cas d'incidents, de pannes ou de ralentissements. Cette visibilité complète permettra une détection rapide et une intervention immédiate pour minimiser les temps d'arrêt et améliorer la disponibilité du réseau.

Cette solution permettra d'automatiser la gestion des équipements réseau et de superviser l'état du réseau de manière proactive.

En utilisant ces deux éléments en conjonction, nous proposons une solution complète pour améliorer la visibilité sur l'état du réseau, réduire les risques d'erreurs manuelles, augmenter la productivité et garantir la disponibilité et la sécurité du réseau.

❖ *Remarque Important :*

Nous tenons à souligner que notre travail consiste à tester cette solution sur GNS3, un simulateur réseau qui permet de créer un environnement virtuel pour effectuer des tests.

En raison des limitations matérielles, il ne sera pas possible de configurer cette solution sur tous les équipements du réseau, y compris ceux de la DMZ et de l'équipement double qui sont configurés pour la haute disponibilité, la performance et la sécurité.

De plus, étant donné que la moitié du réseau hiérarchique est utilisée pour ces mêmes objectifs, la solution ne sera appliquée qu'à une partie du réseau du client.

Pour la mise en place de ce test, nous utiliserons des ordinateurs virtuels, qui ne sont pas des ordinateurs réels, mais des machines virtuelles fonctionnant sur un ordinateur physique. Cette approche permettra de simuler les configurations sans consommer de ressources matérielles supplémentaires. Seul un ordinateur réel sera utilisé comme station de gestion.

Conclusion

Dans ce chapitre, nous avons présenté une vue d'ensemble de l'entreprise du campus NTS ainsi que le réseau de son client. Nous avons également identifié des problèmes de gestion dans ce réseau, ce qui nous a poussés à chercher une solution appropriée.

Pour tester cette solution, nous avons dû effectuer des modifications dans le réseau du client et les configurer sur GNS3.

Tous ces changements et configurations feront l'objectif du chapitre suivant, qui traitera de la préparation de l'environnement de travail.

Chapitre III: Présentation d'un nouvel environnement d'automatisation et de supervision d'un système informatique

Introduction

Le but de ce chapitre est de situer notre travail dans son cadre général. Nous commencerons par une analyse des besoins qui nous permettra d'évaluer et de critiquer le réseau du client. Ensuite, nous présenterons notre schéma proposé ainsi que les modules à automatiser et à superviser. Enfin, nous aborderons l'environnement de travail et expliquerons comment installer nos solutions.

Analyse des besoins

L'administration d'un réseau informatique implique sa gestion. On peut dire que l'administration d'un réseau informatique consiste à gérer un parc informatique. L'administrateur système effectue des tâches courantes telles que l'installation d'applications sur un ou plusieurs postes simultanément, la configuration et l'évolution du matériel, la gestion des utilisateurs et la surveillance du bon fonctionnement du réseau.

Cependant, les administrateurs système du client de l'entreprise NTS ont jusqu'à présent géré les serveurs manuellement, ce qui peut être fastidieux, en particulier lorsque plusieurs équipements nécessitent une configuration similaire. La croissance du nombre de serveurs au fil des années dans l'entreprise rend cette approche inefficace, en particulier pour les serveurs situés sur des sites distants. Ils doivent surveiller l'état des serveurs, des équipements réseau et des services logiciels pour prévenir les pannes et diagnostiquer rapidement les problèmes. Cela inclut la surveillance des performances des serveurs, des routeurs, des commutateurs, des pare-feu et d'autres périphériques réseau pour détecter les problèmes de performances, les pannes et les vulnérabilités de sécurité.

Cependant, cette méthode manuelle d'automatisation et de supervision du réseau est lourde et consomme beaucoup de temps. La principale problématique consiste à trouver une solution efficace pour effectuer les tâches d'administration et surveiller le réseau de manière plus efficiente.

La complexité de la gestion et de la surveillance du réseau conduit à la recherche de solutions Open Source simples et faciles à utiliser pour les administrateurs. Dans le chapitre 1, nous avons examiné deux outils, Ansible et Zabbix, qui se sont révélés être des solutions efficaces pour répondre à ces besoins.

Schéma proposé

Nous avons pris inspiration du réseau du client pour concevoir notre propre topologie, basée sur le modèle hiérarchique qui améliore la sécurité et la fiabilité globales de notre architecture. Pour ce faire, nous avons décidé d'utiliser quatre chambres qui correspondent à des départements spécifiques de l'entreprise. De plus, nous avons ajouté deux services au Datacenter pour automatiser et surveiller l'ensemble du réseau.
En ce qui concerne la sécurité, nous avons mis en place un pare-feu connecté au cloud afin de protéger le réseau contre les attaques externes. Afin de tester cette architecture réseau, nous allons la mettre en place dans notre LAB, comme illustré dans la figure III.1.

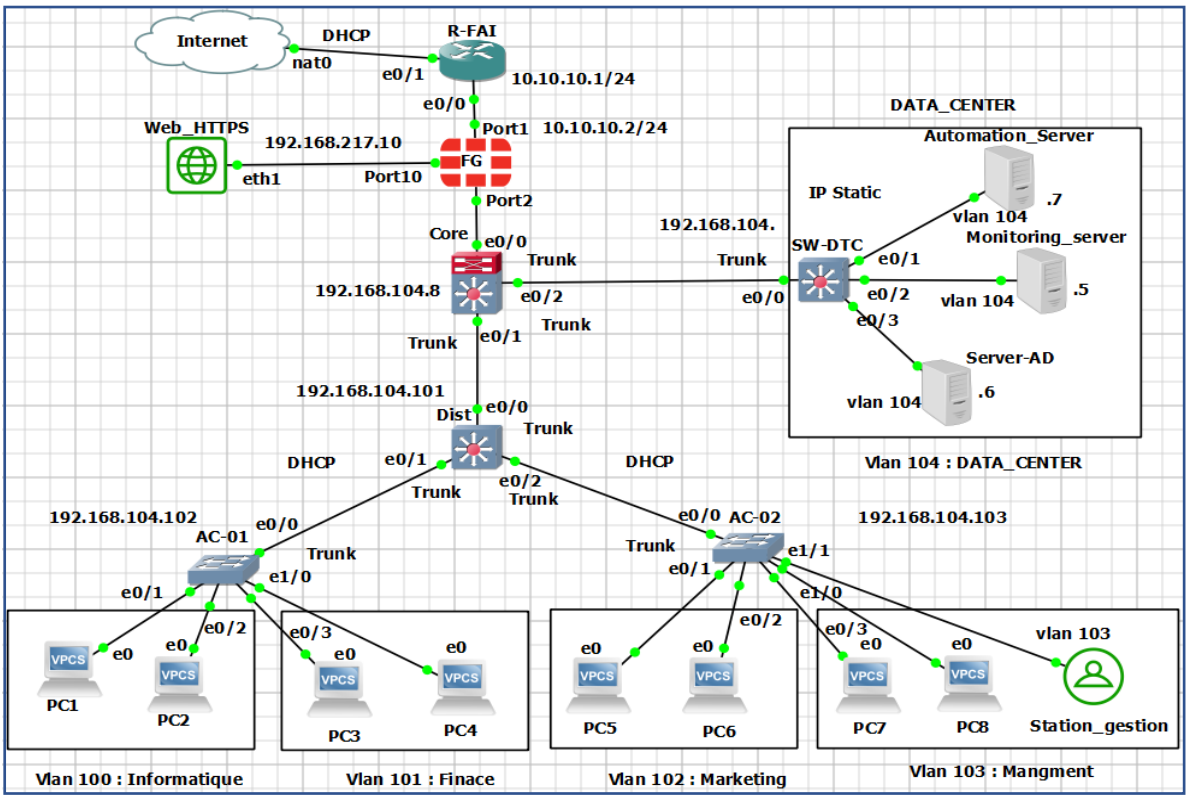


Figure III. 1 : Schéma réseau.

Modules automatisés et supervisés :

Afin de tester notre solution, nous allons sélectionner certains équipements sur lesquels nous effectuerons des tests et des configurations à l'aide d'Ansible et de Zabbix, afin d'assurer le bon fonctionnement de notre solution. Le tableau ci-dessous présente les équipements que nous devrons automatiser et superviser, ainsi que les tâches à effectuer pour l'automatisation et la supervision.

Équipements	Automatisation	Supervision
Pare-feu	• Configuration globale du FG • Création et configuration des Vlans • Config du DHCP sur une interface Vlan • Configuration d'une interface WAN • Configuration d'une route statique • Création d'une nouvelle règle internet	• Un test de redémarrage
Active Directory	• Installation des services ADDC et DNS	• Un test de stress
Switches	• Création des Vlans • Affectation des Vlans aux interfaces	• Un test pour détecter les changements de configuration effectués sur ces équipements.
Routeur	• Configuration du DHCP • Configuration et suppression du SNMP	
Routeur et Switches	• Config de Base + Banner motd	############################

Tableau III. 1 : Les modules à automatiser et à superviser.

Environnement du travail

III.1.1 GNS3

GNS3 (Graphical Network Simulator) est un simulateur de réseau gratuit, open source, multiplateforme qui permet d'émuler des réseaux complexes. Il utilise des logiciels tels que VMware ou Virtual Box pour émuler différents systèmes d'exploitation dans un environnement virtuel.

III.1.2 VMware

VMware Workstation Pro est l'hyperviseur de bureau standard de l'industrie pour l'exécution de machines virtuelles sur des PC Linux ou Windows, il peut être utilisé pour mettre la mise en place d'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.

III.1.2.1 Les machines virtuelle (Les Systèmes d'exploitation utilisés)

Une machine virtuelle (VM) est une instance de système d'exploitation virtuelle exécutée sur un hyperviseur tel que VMWare. Le système d'exploitation installé dans la machine virtuelle est stocké sous forme de fichiers sur le disque dur physique. L'objectif principal des machines virtuelles est de permettre à plusieurs systèmes d'exploitation de fonctionner simultanément sans avoir besoin de matériel physique supplémentaire.

III.1.2.2 Le serveur Active Directory

Après avoir installé VMWare Workstation 17 Pro, ou les informations sur les comptes d'utilisateurs, comme les noms, les mots de passe sont stocker. Nous allons installer le système d'exploitation Wnidows Server. Pour créer cette nouvelle machine virtuelle, nous allons ouvrir VMWare et cliquer sur "Nouvelle machine virtuelle" dans le menu "Fichier". Ensuite, nous allons suivre les étapes jusqu'à ce que nous terminions l'installation.Nous arrivons sur la fenêtre ci-dessous :

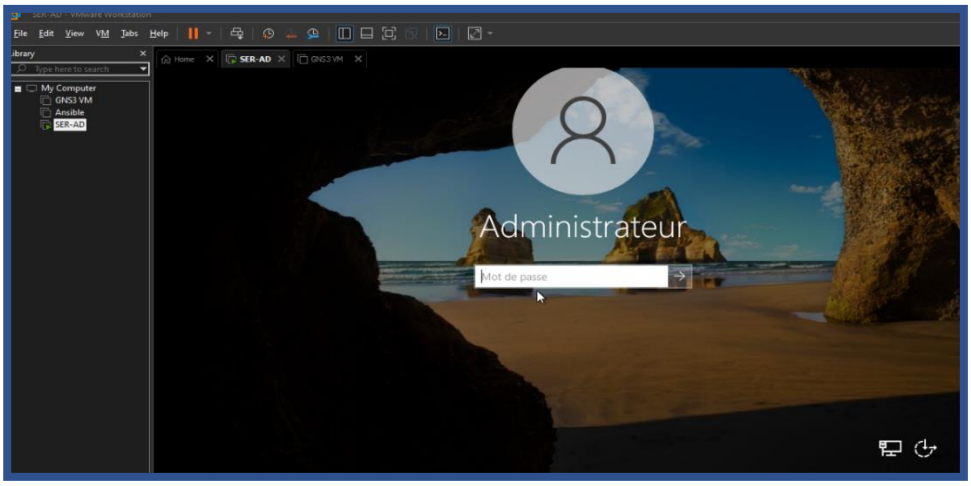


Figure III. 2 : La page d'accueil de Windows server 2020.

III.1.2.3 Serveur d'automatisation

Ubuntu est un système d'exploitation informatique de type Unix basé sur Debian et distribué en tant que logiciel libre et open source, qui est largement utilisé pour les serveurs, les ordinateurs de bureau et les périphériques IoT.

La deuxième machine virtuelle que nous avons installée est configurée pour agir en tant que serveur d'automatisation ou in a installer le système d'exploitation ubuntu. Cela signifie que la machine sera utilisée pour effectuer des tâches de manière automatisée sans intervention humaine. Dans ce cas, nous avons fait le choix d'un outil d'automatisation appelé Ansible.

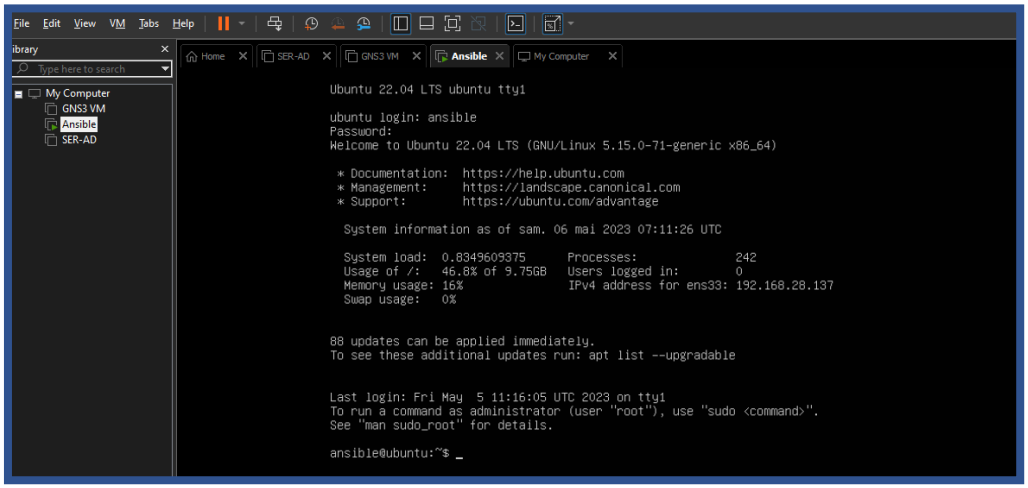


Figure III. 3 : Serveur d'automatisation

III.1.2.4 Serveur de supervision

Dans la troisième machine virtuelle nous allons configure le système d'exploitation Debian qui est un système d'exploitation basé sur Unix conçu pour la stabilité, la sécurité et la facilité d'utilisation, entièrement gratuit. Cette machine sera utilisée comme serveur de supervision sur laquelle nous installerons l'outil de supervision Zabbix.

Figure III. 4 : Serveur de supervision

Installation des outils « Ansible & Zabbix »

III.1.3 Installation et vérification d'ansible

Pour installer Ansible sur notre nœud de contrôle qu'est le serveur d'automatisation sur lequel nous avons installé le système d'exploitation Ubuntu nous exécuterons les commandes indiquées dans les figures suivantes :

III.1.3.1 Vérification et installation des mise a jours :

Avant d'installer Ansible, nous nous assurerons que notre serveur est mis à jour et mis à niveau en exécutant les deux commandes suivantes :

Figure III. 5 : Vérification des mises à jour.

Figure III. 6 : Installation des mises à jour.

III.1.3.2 Software Properties Common

Ce logiciel fournit une abstraction des dépôts apt utilisés. Il permet de gérer facilement des distributions et des logiciel indépendants, Ce package contient les fichiers communs pour les propriétés logicielles telles que D-Bus backend.

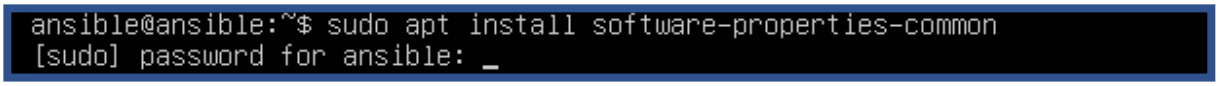


Figure III. 7 : Installation de Software properties Common.

III.1.3.3 Installation de l'outil Ansible :

L'installation d'Ansible sur Ubuntu est un processus simple qui peut être réalisé en utilisant la commande "sudo apt install ansible".

Figure III. 8 : Installation de l'outil Ansible

III.1.3.4 Installation des paquets Python :

Pip est un outil d'installation de paquets Python, cette commande permet d'installer et de gérer facilement des bibliothèques Python supplémentaires.

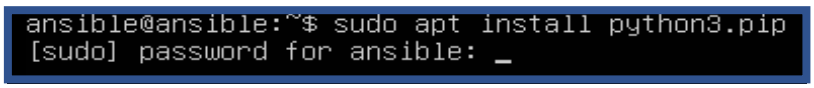


Figure III. 9 : Installation des paquets Python.

III.1.3.5 Personal package archive (PPA) :

La commande ci-dessous ajoute le référentiel PPA à la liste. Il permet aux développeurs d'applications et aux utilisateurs de Linux de créer leurs propres référentiels pour distribuer des logiciels. En utilisant les PPA, il est facile d'obtenir des versions plus récentes de logiciels ou de logiciels qui ne sont pas disponibles via les référentiels Ubuntu officiels.

```
ansible@ansible:~$ sudo apt-add-repository --yes --update ppa:ansible/ansible
```

Figure III. 10 : Mise à jour du répertoire PPA ansible.

III.1.3.6 Installation des paquets SSH-PASS

Afin de se connecter ultérieurement en SSH avec Ansible nous exécuterons la commande indiquée dans la figure

```
root@ansible:~# apt install sshpass
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
sshpass is already the newest version (1.09-1).
sshpass set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ansible:~#
```

Figure III. 11 : Installation des paquets SSH-PASS

III.1.3.7 Vérification de l'installation

Pour vérifier l'installation d'outil Ansible nous exécutons la commande illustrée par la figure

```
root@ansible:~# ansible --version
ansible [core 2.14.5]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/mod
ules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.6 (main, Mar 10 2023, 10:55:28) [GCC 11.3.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
```

Figure III. 12:Vérification de l'installation.

III.1.4 Installation et configuration du Zabbix

Pour installer et configurer le système de surveillance Zabbix sur un système Debian, on suit les étapes suivantes :

III.1.4.1 Mettre à jour le système et installations des mises à jour disponibles :

La Figure III.13 indique comment installer mis à jour du système d'exploitation Debian ;

```
zabbix@debian:~$ sudo su
[sudo] Mot de passe de zabbix :
root@debian:/home/zabbix# apt update && apt upgrade
Atteint :1 http://security.debian.org/debian-security bullseye-security InRelease
Atteint :2 http://deb.debian.org/debian bullseye InRelease
Atteint :3 http://deb.debian.org/debian bullseye-updates InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@debian:/home/zabbix#
```

Figure III. 13 : Mis à jour du système.

III.1.4.2 Installer Apache, PHP et tous les packages nécessaires

Pour exécuter l'interface utilisateur Web de Zabbix, il est nécessaire d'installer un serveur web. Dans le cas de Zabbix, le serveur web recommandé est Apache2.

L'installation d'Apache2 fournit également la prise en charge de PHP, qui est nécessaire pour exécuter l'interface utilisateur Web de Zabbix. La Figure III.14 montre l'installation d'Apache et PHP.

```
root@debian:/home/zabbix# apt install apache2 apache2-bin apache2-data apache2-u
tils libapache2-mod-php libapache2-mod-php7.4 libapr1 libaprutil1 libaprutil1-db
d-sqlite3 libaprutil1-ldap libcurl4 libgd3 liblua5.3-0 libonig5 libsodium23 libx
pm4 libxslt1.1 php php-bcmath php-common php-gd php-ldap php-mbstring php-mysql
php-xml php7.4 php7.4-bcmath php7.4-cli php7.4-common php7.4-gd php7.4-json php7
.4-ldap php7.4-mbstring php7.4-mysql php7.4-opcache php7.4-readline php7.4-xml s
sl-cert
```

Figure III. 14 : Installation de l'Apache et PHP.

Cette commande installe le serveur web Apache2, le module PHP pour Apache2, ainsi que tous les packages nécessaires pour que PHP fonctionne correctement avec Apache2. Elle installe également le certificat SSL nécessaire pour que le serveur web puisse communiquer de manière sécurisée.

- ***Contrôler l'état du serveur :*** Les commandes suivantes sont utilisées pour afficher et gérer l'état du serveur Apache2 :

 # systemctl status apache2 : affiche le statut du service Apache2 ou s'il y a des erreurs.

systemctl start apache2 : démarrer le service Apache2

systemctl stop apache2 : arrêter le service Apache2

systemctl restart apache2 : redémarrer le service Apache2

❖ ***Afficher le statut du service Apache2 pour vérifier s'il est en cours d'exécution.***

La Figure III.15 montre la vérification de l'installation correcte du service Apache

```
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enable>
     Active: active (running) since Thu 2023-05-11 16:06:45 CEST; 36s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 2069 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 2073 (apache2)
      Tasks: 6 (limit: 2264)
     Memory: 13.6M
        CPU: 56ms
     CGroup: /system.slice/apache2.service
             ├─2073 /usr/sbin/apache2 -k start
             ├─2074 /usr/sbin/apache2 -k start
             ├─2075 /usr/sbin/apache2 -k start
             ├─2076 /usr/sbin/apache2 -k start
             ├─2077 /usr/sbin/apache2 -k start
             └─2078 /usr/sbin/apache2 -k start
```

Figure III. 15 : Vérification du service Apache2.

✓ Le serveur Apache2 est en cours d'exécution.

III.1.4.3 Installer le système de gestion de bases de données Maria DB.

Maria DB est un système de gestion de base de données open-source et gratuit qui est une alternative à MySQL. Dans le contexte de l'installation de Zabbix, la commande "apt install mariadb-server mariadb-client" est nécessaire car Zabbix utilise Maria DB pour stocker les données de performance collectées par les agents Zabbix et les serveurs de supervision Zabbix. La Figure III.16 monter son installation.

```
root@debian:/home/zabbix# apt install mariadb-server mariadb-client
```

Figure III. 16 : Installer le système de gestion de bases de données Maria DB

1. Afficher le statut du service Maria DB pour vérifier s'il est en cours d'exécution

A l'aide de la commande '***systemctl status mariadb'*** présenter dans la figure III.17 nous allons vérifier le service Maria DB s'il est en cours d'exécution

```
root@debian:/home/zabbix# systemctl status mariadb
● mariadb.service - MariaDB 10.5.19 database server
     Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor prese>
     Active: active (running) since Thu 2023-05-11 16:37:08 CEST; 4min 30s ago
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
    Process: 2668 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var>
    Process: 2669 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_ST>
    Process: 2671 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && >
    Process: 2734 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_S>
    Process: 2736 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/>
   Main PID: 2719 (mariadbd)
     Status: "Taking your SQL requests now..."
      Tasks: 9 (limit: 2264)
     Memory: 72.8M
        CPU: 387ms
     CGroup: /system.slice/mariadb.service
             └─2719 /usr/sbin/mariadbd
```

Figure III. 17 : Affichage du statut du service Maria DB.

✓ Maria DB est en cours d'exécution.

2. Configurer les paramètres de sécurité de Maria DB :

Nous allons configurer les paramètres de sécurité y compris la modification du mot de passe root, la suppression des utilisateurs anonymes et la désactivation de la connexion root à distance comme la Figure III.18 indique.

```
root@debian:/home/zabbix# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
 ... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] n
 ... skipping.


Remove anonymous users? [Y/n] y
 ... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
 ... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
 ... Success!

Cleaning up...
```

Figure III. 18 : Configuration les paramètres de sécurité de Maria DB.

3. Se connecter à la console Maria DB avec le compte root :

La Figure III.19 monte comment nous allons se connecter à la console Maria DB avec la commande **myssql -u root -p**

```
root@debian:/home/zabbix# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 36
Server version: 10.5.19-MariaDB-0+deb11u2 Debian 11

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Figure III. 19 : Connexion à la console Maria DB.

4. *Création de la base de données "zabbix" avec attribution de privilèges et mot de passe "mypassword" pour l'utilisateur "zabbix".*

```
MariaDB [(none)]> create database zabbix character set utf8 collate utf8_bin;
Query OK, 1 row affected (0,001 sec)

MariaDB [(none)]> grant all privileges on zabbix.* to zabbix@localhost identified by 'mypassword';
Query OK, 0 rows affected (0,002 sec)

MariaDB [(none)]> set global log_bin_trust_function_creators = 1;
Query OK, 0 rows affected (0,000 sec)

MariaDB [(none)]> quit;
Bye
root@debian:/home/zabbix#
```

Figure III. 20 : Création de la base de données Zabbix.

5. *Téléchargement et installation du fichier de configuration 'zabbix-release.deb'*

```
root@debian:/home/zabbix# wget https://repo.zabbix.com/zabbix/6.3/debian/pool/main/z/zabbix-release/zabbix-release_6.3-1+debian11_all.deb
--2023-05-11 18:39:29--  https://repo.zabbix.com/zabbix/6.3/debian/pool/main/z/zabbix-release/zabbix-release_6.3-1+debian11_all.deb
Résolution de repo.zabbix.com (repo.zabbix.com)… 178.128.6.101, 2604:a880:2:d0::2062:d001
Connexion à repo.zabbix.com (repo.zabbix.com)|178.128.6.101|:443… connecté.
requête HTTP transmise, en attente de la réponse… 200 OK
Taille : 3672 (3,6K) [application/octet-stream]
Sauvegarde en : « zabbix-release_6.3-1+debian11_all.deb »

zabbix-release_6.3- 100%[===================>]   3,59K  --.-KB/s    ds 0s

2023-05-11 18:39:30 (13,9 MB/s) – « zabbix-release_6.3-1+debian11_all.deb » sauvegardé [3672/3672]

root@debian:/home/zabbix# dpkg -i zabbix-release_6.3-1+debian11_all.deb
Sélection du paquet zabbix-release précédemment désélectionné.
(Lecture de la base de données... 141705 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de zabbix-release_6.3-1+debian11_all.deb ...
Dépaquetage de zabbix-release (1:6.3-1+debian11) ...
Paramétrage de zabbix-release (1:6.3-1+debian11) ...
root@debian:/home/zabbix#
```

Figure III. 21 : Téléchargement et installation du fichier 'zabbix-release.deb'.

6. *Installer les package Zabbix Server, Frontend et Agent.*

```
root@debian:/home/zabbix# apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-sql-scripts zabbix-agent
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  fping libevent-core-2.1-7 libevent-pthreads-2.1-7 libmodbus5 libodbc1
  libopenipmi0 libssh-4 snmpd
Paquets suggérés :
  libmyodbc odbc-postgresql tdsodbc unixodbc-bin snmptrapd zabbix-nginx-conf
Les NOUVEAUX paquets suivants seront installés :
  fping libevent-core-2.1-7 libevent-pthreads-2.1-7 libmodbus5 libodbc1
  libopenipmi0 libssh-4 snmpd zabbix-agent zabbix-apache-conf
  zabbix-frontend-php zabbix-server-mysql zabbix-sql-scripts
0 mis à jour, 13 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 19,3 Mo dans les archives.
Après cette opération, 58,7 Mo d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

Figure III. 22 : Installation des packages Zabbix Server, Frontend et Agent.

7. Importer le script SQL de Zabbix dans la base de données Maria DB

Décompresser le fichier de script SQL et exécuter la commande MySQL pour l'importer dans la base de données spécifiée. Voire la Figure III.24

```
root@debian:/home/zabbix# zcat  /usr/share/zabbix-sql-scripts/mysql/server.sql.
gz | mysql --default-character-set=utf8mb4 -uzabbix -p'mypassword' zabbix
root@debian:/home/zabbix#
```

Figure III. 23 : Importation du script SQL.

8. Configurer le fichier de configuration de Zabbix Server

```
root@debian:/home/zabbix# nano /etc/zabbix/zabbix_server.conf
```

Figure III. 24 : Configurer le fichier de configuration de Zabbix Server

9. Modifier le fichier de configuration zabbix server

Définir le nom de l'hôte de la base de données, le nom de la base de données, l'utilisateur de la base de données et le mot de passe de l'utilisateur. Voire la Figure III.27

```
DBHost=localhost

### Option: DBName
#       Database name.
#       If the Net Service Name connection method is used to connect to Oracle
#       the tnsnames.ora file or set to empty string; also see the TWO_TASK en
#       empty string.
#
# Mandatory: yes
# Default:
# DBName=

DBName=zabbix

DBUser=zabbix

### Option: DBPassword
#       Database password.
#       Comment this line if no password is used.
#
# Mandatory: no
# Default:
DBPassword=mypassword
```

Figure III. 25 : Modification du fichier de configuration Zabbix server

III.1.4.4 Redémarrer Apache2 et Démarrer Zabbix Server et Zabbix Agent.

À l'aide de deux commandes présentées dans la figure ci-dessous, nous allons redémarrer le service : Apache2 et démarrer les services Zabbix Agent et Zabbix Server.

```
root@debian:/home/zabbix# systemctl restart apache2
root@debian:/home/zabbix# systemctl start zabbix-server zabbix-agent
root@debian:/home/zabbix#
```

Figure III. 26 : Redémarrer Apache2 et Démarrer Zabbix Server et Zabbix Agent.

III.1.4.5 Obtenir l'adresse IP du serveur

Pour afficher l'adresse de serveur Zabbix il suffit de taper la commande indiquer dans la Figure III.28

```
root@debian:/home/zabbix# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.104.5  netmask 255.255.255.0  broadcast 192.168.104.255
        inet6 fe80::20c:29ff:fe1c:bc7a  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:1c:bc:7a  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 94  bytes 8382 (8.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Boucle locale)
        RX packets 113  bytes 10864 (10.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 113  bytes 10864 (10.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figure III. 27 : Affichage du l'adresse IP du serveur.

Après avoir terminé l'installation, on peut accéder à l'interface Web de Zabbix en ouvrant le navigateur et en accédant à l'URL suivante *:* ***http://192.168.104.5/zabbix***. Le nom d'utilisateur par défaut est "Admin" et le mot de passe est "Zabbix". La Figure III.29 montre l'interface web de Zabbix.

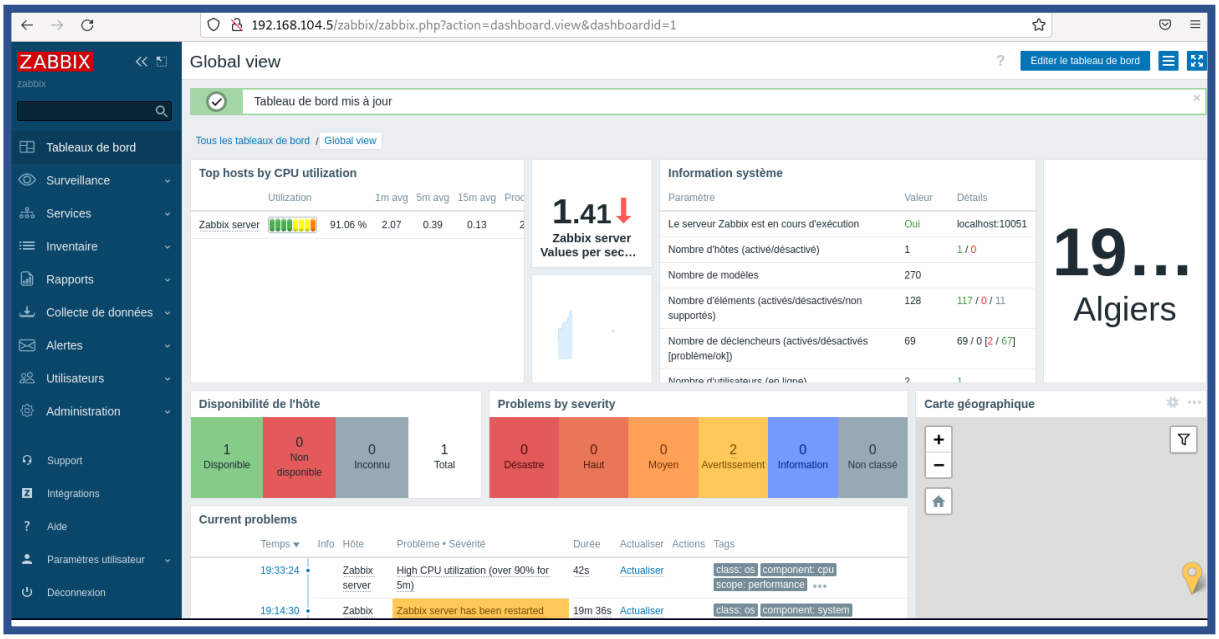


Figure III. 28 : L'interface Web de Zabbix.

Conclusion :

Dans ce chapitre, nous avons présenté les raisons qui nous ont poussés à automatiser et superviser le réseau de notre client, ainsi que le schéma proposé pour tester notre solution sur certains équipements. Nous avons également détaillé toutes les étapes nécessaires à la préparation de notre environnement.

Chapitre IV: Implémentation

Introduction

Ce chapitre examine des exemples d'automatisation avec Ansible et de supervision avec Zabbix. Notre objectif est de pouvoir utiliser efficacement les deux outils que nous avons choisi précédemment pour gérer tous les nœuds de notre réseau.

Nous allons commencer par l'automatisation en créant des fichiers ou nous allons enregistrer toutes les machines cibles que nous devons gérer. Après cela, nous créerons les Playbooks qui incluront toutes les tâches à exécuter. Une fois que nos Playbooks prêts, nous pourrons les injecter sur les nœuds cibles et tester leur efficacité. Le protocole utilisé afin d'établir une connexion sécurisée entre ces nœuds est le protocole Secure Shell « SSH ».

En parallèle, nous allons explorer la supervision avec Zabbix. Nous allons d'abord voire comment créer un compte Zabbix, puis comment ajouter les équipements à superviser sur Zabbix. Ensuite, nous créerons une carte Zabbix qui nous permettra d'avoir une représentation visuelle de la topologie du réseau ou du système surveillé. Enfin, nous configurerons des alertes Zabbix avec nos services Gmail pour être notifiés en cas d'anomalie.

Simulation avec Ansible

IV.1.1 Le protocole Secure Shell (SSH)

Secure Shell est un protocole sécurisé qui utilise le port TCP 22. Il fournit une connexion de gestion sécurisée (cryptée) à un appareil distant. SSH assure la sécurité des connexions à distance en fournissant un cryptage fort lorsqu'un dispositif est authentifié (nom d'utilisateur et mot de passe) et également pour les données transmises entre les dispositifs communicants

IV.1.1.1 Configuration du protocole SSH sur les équipements Cisco

Avant de configurer SSH, il est important de vérifier que l'équipement dispose d'un nom d'hôte unique et que les paramètres réseau sont correctement configurés. Ensuite, il convient de vérifier la prise en charge de SSH en utilisant la commande "***show ip ssh***" afin de s'assurer que les équipements prennent en charge ce protocole.

Il est à noter que si l'équipement utilise un IOS qui ne prend pas en charge les fonctionnalités cryptographiques, cette commande ne sera pas reconnue. La figure IV.1 illustre un exemple du résultat de la commande "***show ip ssh***".

```
R-FAI#show ip ssh
SSH Enabled - version 2.0
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa
Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa
Encryption Algorithms:aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,3des-cbc,aes192-cbc,aes2
56-cbc
MAC Algorithms:hmac-sha1,hmac-sha1-96
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
IOS Keys in SECSH format(ssh-rsa, base64 encoded): R-FAI.karim.dina
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCTt9qCEZahrCY/nBOOEBaqZezZcZSR80uRHqbtd6Xj
JgsFenkdk13YxEmD6B/VYZ0MwfxW4eGKxaZNuiJ+4Mu/ZZ1+Y7EfSMyimE0Ih+2Vok1s59UH0qAZOELj
VrEAk2fpl0gGaXiUBgWpDLauCZKpcFKK+kZQfUEWEJUK7PrbVFmb/DUh4E3Jwimqnjyvs7f64bHEg2n6
WF13GVaCdKo9Oh5S8l5hZXGB2IhX0wb3rW7mZ6H8PGJk9Hcf4sER56bNUT12gNIg9+7LJgvMvGcaXJ5H
ELS0dJNeQbK/AljJns+Y1GKIr3aRK2Jxeh1awUzxZLwPA2CjbLCoktmsbAFt
R-FAI#
```

Figure IV. 1 : Vérification de la prise en charge du protocol SSH.

❖ *Configuration du protocole SSH sur le routeur FAI*

La configuration de SSH sur le routeur FAI peut être réalisée en suivant les étapes illustrées à la figure IV.2. De manière similaire, le même processus sera appliqué pour configurer SSH sur les autres commutateurs.

```
R-FAI(config)#username admin priv 15 secret karim.dina
R-FAI(config)#ip domain-name cisco.com
R-FAI(config)#crypto key generate rsa
The name for the keys will be: R-FAI.cisco.com
Choose the size of the key modulus in the range of 360 to 4096 for your
  General Purpose Keys. Choosing a key modulus greater than 512 may take
  a few minutes.

How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 1 seconds)

R-FAI(config)#ip s
*May 29 20:39:38.167: %SSH-5-ENABLED: SSH 1.99 has been enabled
R-FAI(config)#ip ssh version 2
R-FAI(config)#line vty 0 15
R-FAI(config-line)#login local
R-FAI(config-line)#transport input ssh
R-FAI(config-line)#end
R-FAI#wr$
*May 29 20:40:24.475: %SYS-5-CONFIG_I: Configured from console by console
R-FAI#wr
Building configuration...

  [OK]
R-FAI#
```

Figure IV. 2 : Configuration du protocole SSH sur le routeur FAI

❖ *Vérification de la connectivité SSH à partir du serveur Ubuntu*

Avant de vérifier la connectivité SSH, il est important de s'assurer que le client Open-SSH est installé sur Ubuntu. Pour cela, il suffit de taper la commande "***ssh***" dans un terminal, comme illustré à la Figure IV.3. Si le client Open-SSH est installé, un message affichant les options disponibles pour la commande "***ssh***" devrait apparaître. En cas d'absence du client, il est possible de l'installer en utilisant la commande suivante : " ***sudo apt install openssh-client*** "

```
ansible@ansible-virtual-machine:~$ ssh
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-B bind_interface]
           [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
           [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
           [-i identity_file] [-J [user@]host[:port]] [-L address]
           [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
           [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote_tun]] destination [command]
ansible@ansible-virtual-machine:~$
```

Figure IV. 3 : Vérification de package openssh client.

Maintenant qu'Openssh Client est installé sur Ubuntu, nous pouvons vérifier la connectivité SSH à partir de cette distribution. Pour cela, il suffit de taper la commande "***ssh admin@10.10.10.1***" dans le terminal. "admin" correspond au nom d'utilisateur du routeur FAI, tandis que "***10.10.10.1***" est l'adresse IP du routeur, comme illustré dans la Figure IV.4.

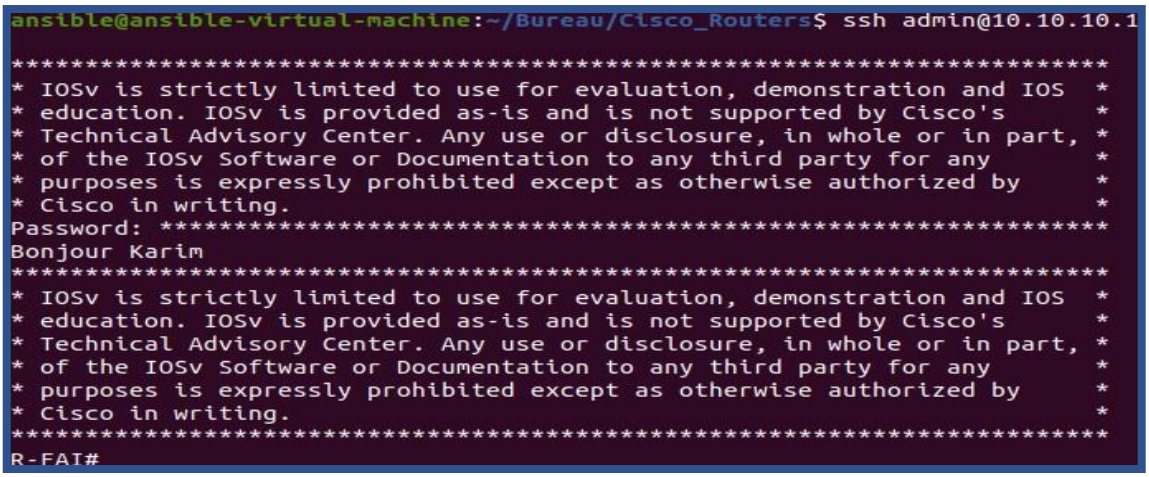


Figure IV. 4 : Connexion établie sur le routeur.

IV.1.1.2 Configuration du protocole SSH sur le serveur Windows

Dans cette partie, nous avons besoin d'installer OpenSSH Server sur le serveur Windows afin de pouvoir y accéder à partir du serveur Ubuntu en utilisant le protocole SSH pour automatiser des installations.

L'installation et la configuration d'OpenSSH Server sur Windows Server s'effectue en ouvrant Windows PowerShell et en suivant les étapes numérotées de 1 à 3, telles qu'illustrées dans la Figure IV.5.

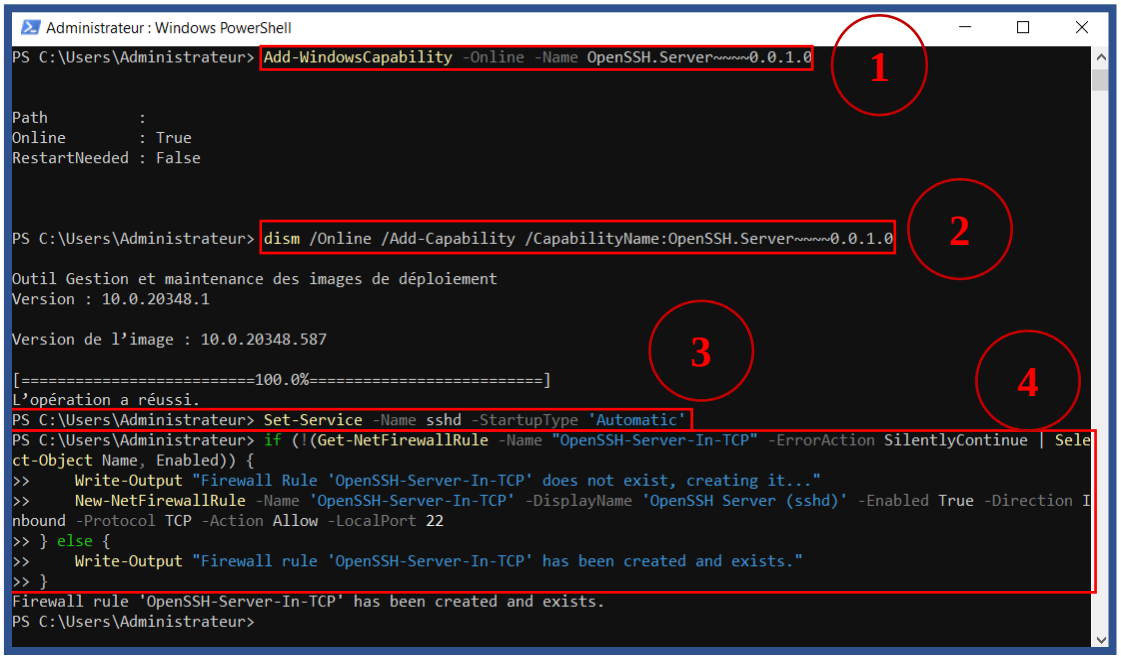


Figure IV. 5 : Installation du service OpenSSH-Server sur le Windows server.

La ***troisième*** commande est utilisée pour configurer le service du serveur Openssh afin qu'il démarre automatiquement au démarrage du système.

La quatrième commande est une structure conditionnelle "if-else" qui vérifie l'existence de la règle de pare-feu "Openssh-Server-In-TCP". Si cette règle de pare-feu n'est pas présente, la commande New-NetFirewallRule crée une nouvelle règle de pare-feu permettant le trafic

entrant sur le port TCP 22, qui est utilisé par le serveur Openssh. Si des règles de pare-feu existent déjà, la commande Write-Output affiche un message indiquant leur présence.

IV.1.2 Le fichier inventaire

Dans cette partie, nous allons créer un fichier d'inventaire appelé "***hosts.yml***". Ce fichier contiendra les informations de tous les équipements (hosts), notamment leurs adresses IP, noms et informations de connexion telles que l'authentification SSH et httpapi. Cela comprendra le nom d'utilisateur, le mot de passe et le système d'exploitation de chaque équipement. Voir la liste IV.1.

```
fortigates:
  hosts:
    FG:
      ansible_host: 192.168.104.1
      ansible_user: admin
      ansible_password: "1"
      ansible_network_os: fortinet.fortios.fortios

Windows:
  hosts:
    Windows-server:
          ansible_host: 192.168.104.6
          ansible_user: ADMINISTRATEUR
          ansible_password: Karim@Dina
          ansible_connection: ssh
          ansible_shell_type: cmd

Cisco:
  children:
    routers:
      hosts:
        R-FAI:
          ansible_host: 10.10.10.1
          ansible_ssh_user: admin
          ansible_ssh_pass: karim.dina
          ansible_network_os: ios
    switches:
      hosts:
        S-CORE:
          ansible_host: 192.168.104.8
          ansible_ssh_user: admin
          ansible_ssh_pass: karim.dina
          ansible_network_os: ios
```

Liste IV.1 : Fichier Inventaire.

IV.1.3 Les playbooks

Un Playbook est généralement structuré en sections, comprenant le nom du Playbook, la spécification des hôtes sur lesquels les tâches doivent être exécutées, la collecte ou non des faits (facts) sur les hôtes, la spécification de la connexion à utiliser, et enfin, la liste des tâches à exécuter, chacune avec son propre nom, module et paramètres.

IV.1.3.1 Structure du Playbook Ansible

Un Playbook est structuré comme suit :

- ***Nom du Playbook (name) :***

Il s'agit du nom donné au Playbook. Cela permet de donner une description concise de l'objectif ou du contenu du Playbook. Le nom est facultatif mais il est généralement recommandé de l'inclure pour faciliter la compréhension et la gestion des playbooks.

- ***Les hôtes (hosts) :***

Cela spécifie les hôtes sur lesquels les tâches du Playbook seront exécutées. Les hôtes peuvent être définis sous forme de groupes (comme "FG" dans l'exemple), de noms d'hôtes individuels ou de motifs pour faire correspondre plusieurs hôtes à l'aide d'expressions régulières. Les groupes d'hôtes peuvent être définis dans les fichiers d'inventaire d'Ansible. Voire la liste IV.1.

- ***Gather_facts :***

Ceci est un paramètre facultatif qui indique si Ansible doit collecter des faits (facts) sur les hôtes avant d'exécuter les tâches. Les faits sont des informations sur les hôtes, telles que les adresses IP, les interfaces réseau, les disques, les versions du système d'exploitation, etc. Ces faits peuvent être utilisés dans les tâches pour prendre des décisions en fonction de l'état des hôtes.

- ***Type de connexion (connection) :***

Cela spécifie le type de connexion à utiliser pour se connecter aux hôtes. Dans l'exemple donné, le type de connexion est "httpapi", ce qui indique l'utilisation d'une API HTTP pour se connecter aux hôtes FortiGate. Selon le type d'équipement ou de système distant, nous prouvent spécifier d'autres types de connexion pris en charge par Ansible, tels que SSH, Telnet, etc.

- ***Liste des tâches (tasks) :***

Il s'agit de la section principale du Playbook où on peut spécifier les tâches à exécuter. Chaque tâche est une unité d'automatisation qui effectue une action spécifique sur les hôtes cibles. Chaque tâche doit avoir un nom (name), qui décrit brièvement l'objectif de la tâche. Le nom est utilisé à des fins de référence et de lisibilité.

Chaque tâche doit également spécifier le nom du module Ansible (module_name) à exécuter. Un module est une unité d'automatisation qui implémente une fonction spécifique, comme la configuration d'un système, la gestion de fichiers, l'exécution de commandes, etc. Le module est spécifié avec ses paramètres (module_parameters), qui définissent le comportement et les options spécifiques de ce module.

Les paramètres du module varient en fonction du module utilisé. Par exemple, pour le module fortios_system_global, les paramètres spécifiques du module comprennent vdom, system_global, admin_concurrent, admin_console_timeout, etc. Chaque module a sa propre documentation qui décrit en détail les paramètres pris en charge et leur utilisation.

IV.1.3.2 Création et exécution des Playbooks sur le Firewall FortiGate

❖ ***Création d'un Playbook pour le Firewall FortiGate***

Comme le montrent les listes de 2 à 6, nous avons exécuté plusieurs tâches dans un seul Playbook. Afin de ne pas rendre le Playbook très long, nous avons décidé de le découper en plusieurs tâches pour faciliter leur explication. Chaque tâche est représentée par une liste.

***Tache 1** : Configuration globale du Firewall FortiGate* : Ce Playbook est conçu pour configurer les paramètres de base d'un pare-feu FortiGate. Il utilise une connexion httpapi pour accéder au dispositif. Les tâches incluent la configuration des paramètres globaux tels que la gestion des administrateurs, les ports d'accès (HTTP, SSH, Telnet), le nom d'hôte, le fuseau horaire et la langue, etc. Le Playbook vise à automatiser ces étapes pour simplifier la configuration initiale du pare-feu et garantir une configuration cohérente.

```
---
- name: CONF DE BASE FORTIGATE
  hosts: FG
  gather_facts: false
  connection: httpapi
  tasks:
############################################################
  - name: Configuration GLOBAL du fortigate
    fortios_system_global:
      vdom: root
      system_global:
        admin_concurrent: enable
        admin_console_timeout: 300
        admin_login_max: 100
        admin_maintainer: enable
        admin_port: 80
        admin_restrict_local: disable
        admin_scp: disable
        admin_sport: 4433
        admin_ssh_grace_time: 120
        admin_ssh_password: enable
        admin_ssh_port: 22
        admin_ssh_v1: disable
        admin_telnet: enable
        admin_telnet_port: 23
        admintimeout: 120
        hostname: "FG"
        timezone: 35
        language: french
        gui_firmware_upgrade_warning: enable
```

Liste IV.2 : Playbook de la tache 1.

***Tache 2** : Création et configuration des Vlans (exemple Vlan 100) et configuration du DHCP sur cette interface (Vlan 100)* : Dans cette partie du Playbook, deux tâches sont réalisées. La première tâche consiste à créer et configurer une interface VLAN, en l'occurrence le VLAN 100. Les paramètres spécifiés incluent le type d'interface, l'adresse IP, les permissions d'accès, le nom et le rôle de l'interface. La deuxième tâche concerne la configuration du serveur DHCP sur l'interface du VLAN 100. Les paramètres comprennent l'ID du serveur DHCP, l'interface associée, le mode IP (plage d'adresses), la plage d'adresses DHCP et le masque de sous-réseau. Ces étapes visent à mettre en place un environnement réseau avec une interface VLAN configurée et un serveur DHCP opérationnel pour ce VLAN spécifique.

```
- name: Création et configuration d'un interface Vlan exemple Vlan 100
  fortios_system_interface:
    vdom: "root"
    state: "present"
    system_interface:
      vdom: "root"
      interface: "port2"
      type: "vlan"
      vlanid: "100"
      mode: "static"
      ip: "192.168.100.1 255.255.255.0"
      allowaccess: ['https','ping','http','telnet','ssh','snmp']
      name: "int_Vlan_100"
      role: "lan"
 ############################################################
- name: Configuration du DHCP sur l'interface du Vlan 100
  fortios_system_dhcp_server:
    vdom: "root"
    state: "present"
    system_dhcp_server:
      id: "1"
      interface: "int-Vlan_100"
      ip_mode: "range"
      ip_range:
      - end_ip: "192.168.100.100"
        id: "1"
        start_ip: "192.168.100.10"
      netmask: "255.255.255.0"
```

Liste IV.3 : Playbook de la tache 2.

Tache 3 *: Configuration d'une interface WAN (exemple de l'interface port1) :*

Cette partie du Playbook concerne la configuration de l'interface WAN, spécifiquement pour le port1 du FortiGate. Les paramètres configurés incluent l'algorithme de routage (L3), les permissions d'accès (https, ping, http, telnet, ssh, snmp), l'adresse IP et le masque de sous-réseau, le nom de l'interface (port1), le statut (activé), le type d'interface (physique), le rôle (WAN), le mode (statique) et un alias pour identifier cette interface (Internet_réseau_WAN). Ces étapes permettent de mettre en place une connectivité WAN en configurant l'interface appropriée avec les paramètres requis pour la communication externe.

```
- name: Configuration du WAN -> port1
  fortios_system_interface:
    vdom: root
    state: present
    enable_log: yes
    system_interface:
      algorithm: L3
      allowaccess: ['https','ping','http','telnet','ssh','snmp']
      ip: 10.10.10.2 255.255.255.0
      name: port1
      status: up
      type: physical
      role: wan
      mode: static
      alias: Internet_réseau_WAN
```

Liste IV.4 : Playbook de la tache 3.

Tache 4 : *Configuration d'une route statique par défaut (exemple pour la Gateway 10.10.10.1) :* Dans cette partie du Playbook, une tâche est effectuée pour configurer une route statique par défaut. Les paramètres spécifiés incluent le numéro de séquence, l'activation du protocole de détection de défaillance (BFD), un commentaire descriptif, la destination (0.0.0.0/0.0.0.0 pour représenter toutes les adresses IP), le dispositif d'interface de sortie (port1), la passerelle (10.10.10.1) et la distance administrative (10). Ces étapes permettent de définir une route par défaut pour acheminer tout le trafic vers la passerelle spécifiée via l'interface WAN configurée précédemment (port1).

```
- name: Configuration d'une route statique par défaut
  fortios_router_static:
    vdom:  "root"
    state: "present"
    router_static:
      seq_num: "10"
      bfd: "enable"
      comment: "route statique par défaut"
      dst: "0.0.0.0/0.0.0.0"
      device: "port1"
      gateway: "10.10.10.1"
      distance: "10"
```

Liste IV.5 : Playbook de la tache 4.

Tache 5 *: Création d'une nouvelle règle pour donner accès aux utilisateurs a internet* : Dans cette partie du Playbook, une règle de pare-feu est configurée pour permettre l'accès à Internet aux utilisateurs. Les paramètres spécifiés incluent le nom de la règle (Access to internet), l'action (accepter), l'adresse de destination (toutes les adresses), l'interface de destination (port1), la journalisation du trafic (utm), la traduction d'adresse réseau (NAT activé), l'identifiant de la politique, le service (tous les services), l'adresse source (toutes les adresses), l'interface source (port2), la planification (toujours disponible) et le délai d'expiration de la planification (désactivé). Cette règle permettra aux utilisateurs sur l'interface port2 d'accéder à Internet en acceptant tout le trafic sortant.

```
- name: Regle pour donner acces aux users a internet
  fortios_firewall_policy:
    vdom: root
    state: present
    firewall_policy:
        name: Access to internet
        action: accept
        dstaddr:
        - name: all
        dstintf:
        - name: port1
        logtraffic: utm
        nat: enable
        policyid: 2
        service:
        - name: ALL
        srcaddr:
        - name: all
        srcintf:
        - name: port2
        schedule: always
        schedule_timeout: disable
```

Liste IV.6 : Playbook de la tache 5.

❖ ***Exécution et résultat du Playbook***

Nous injectons le Playbook « playbook.yml » sur un groupe de nœuds situés dans le fichier d'inventaire « hosts.yml » en utilisant la commande « ansible-playbook » comme Illustré dans la figure IV.7.

```
ansible@ansible-virtual-machine:~/Bureau$ ansible-playbook -i hosts.yml playbook.yml
```

Figure IV. 6 : Commande d'exécution du playbook

Les résultats de cette automatisation apparaissent quelques secondes après l'exécution de la commande dans le terminal de la machine Ansible comme l'indique la figure IV.7.

Selon les résultats obtenus avec Ansible, on peut constater que 6 des 8 commandes ont été exécutées avec succès, comme indiqué par [OK]. Cependant, normalement, la première tâche, "Configuration Globale du Fortigate", aurait dû être marquée comme [changed] au lieu de [OK].

Cela est dû au fait que nous avions déjà exécuté cette tâche et les configurations globales ont déjà été modifiées. Donc Ansible indique [OK] pour cette tâche, et non [changed]. Quant aux deux commandes restantes, elles sont indiquées avec [changed], car Ansible a modifié leur configuration.

```
PLAY [CONF DE BASE FORTIGATE] ***************************************************************************

TASK [Configuration GLOBAL du fortigate] ****************************************************************
ok: [FG]

TASK [Création et configuration d'une interface Vlan exemple Vlan 100] **********************************
ok: [FG]

TASK [Configuration du DHCP sur l'interface du Vlan 100] ************************************************
ok: [FG]

TASK [Configuration du WAN -> port1] ********************************************************************
changed: [FG]

TASK [Configuration d'une route statique par défaut] ****************************************************
ok: [FG]

TASK [Regle pour donner acces aux users a internet] *****************************************************
changed: [FG]

PLAY RECAP **********************************************************************************************
FG                         : ok=6    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignore
```

Figure IV. 7 : Résultat du playbook exécuté sur FortiGate

IV.1.3.3 Création et exécution des Playbooks sur les switches

❖ ***Création des Playbooks pour les Switches***

Dans cette partie, nous allons créer deux Playbooks pour les switches. Le premier Playbook est destiné à tous les switches et vise à créer des VLANs (100, 101, 102, 103). Le VLAN 104 est déjà créé, car c'est sur ce VLAN que les switches ont été configurés avec une adresse IP. Pour le deuxième Playbook, il sera destiné au switch 'S-AC01' ou 'S-AC02'. Étant donné que les deux switches ont les mêmes configurations d'interfaces, nous allons choisir le switch 'S-AC01'. Les deux Playbooks sont créés et illustrés dans les listes 7 et 8.

Tâche du Playbook1 : Le premier Playbook est constituée d'une seule tache il vise à créer quatre VLAN (100, 101, 102, 103) sur tous les commutateurs (hosts : switches). Il utilise la

connexion network_cli pour se connecter aux commutateurs et exécuter les tâches nécessaires. Les tâches consistent à créer les VLAN avec leurs noms, identifiants, états (actifs) et statuts de désactivation (désactivés). L'état global de la tâche est "merged", ce qui signifie que les modifications seront fusionnées avec la configuration existante.

```
- name: Créatios des Vlans (100,101,102,103) sur tous les Switches
  hosts: switches
  gather_facts: false
  connection: network_cli
  tasks:
    - name: Créatios des Vlans
      ios_vlans:
        config:
        - name: Informatique
          vlan_id: 100
          state: active
          shutdown: disabled

        - name: Finance
          vlan_id: 101
          state: active
          shutdown: disabled

        - name: Marketing
          vlan_id: 102
          state: active
          shutdown: disabled

        - name: Managment
          vlan_id: 103
          state: active
          shutdown: disabled

        state: merged
```

Liste IV.7 : La Tache du Playbook1.

❖ ***Exécution et résultat du Playbook***

```
ansible@ansible-virtual-machine:~/Bureau/PFE/Cisco_Siwtchs$ ansible-playbook -i hosts.yml all_sw.yml
 -u admin -k
SSH password:

PLAY [Créatios des Vlans (100,101,102,103) sur tous les Switches] ********************************
**********

TASK [Créatios des Vlans] ********************************************************************
**********
ok: [S-AC01]
ok: [S-AC02]
ok: [S-DIST]
ok: [S-CORE]
ok: [S-DTC]

PLAY RECAP ***********************************************************************************
**********
S-AC01                     : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0
S-AC02                     : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0
S-CORE                     : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0
S-DIST                     : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0
S-DTC                      : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0
```

Figure IV. 8 : Résultat du playbook 1 exécuté sur tous les switches

Tâche du Playbook 2 : Ce Playbook a pour objectif d'affecter des VLAN aux interfaces du commutateur AC01. En se connectant au commutateur AC01 via la connexion network_cli, le Playbook exécute une tâche spécifique. L'interface GigabitEthernet0/1 est configurée en mode d'accès et associée au VLAN 100, tandis que l'interface GigabitEthernet0/2 est également configurée en mode d'accès et associée au VLAN 100. De plus, l'interface GigabitEthernet0/3 est configurée en mode d'accès et associée au VLAN 101, et l'interface GigabitEthernet1/0 est également configurée en mode d'accès et associée au VLAN 101. Enfin, l'interface GigabitEthernet0/0 est configurée en mode Trunk, autorisant les VLAN 100 à 104 ainsi que le

VLAN 1, avec une encapsulation de type dot1q. Cette tâche d'affectation des VLAN aux interfaces permet de définir la segmentation et la gestion appropriées du trafic sur le commutateur AC01.

```
---
- name: Affectation des vlans aux interfaces du switch AC01
  hosts: S-AC01
  gather_facts: false
  connection: network_cli
  tasks:
    - name: Affectation des vlans aux interfaces du switch AC01
      ios_l2_interfaces:
        config:
        - name: GigabitEthernet0/1
          mode: access
          access:
            vlan: 100

        - name: GigabitEthernet0/2
          mode: access
          access:
            vlan: 100


        - name: GigabitEthernet0/3
          mode: access
          access:
            vlan: 101

        - name: GigabitEthernet1/0
          mode: access
          access:
            vlan: 101
        - name: GigabitEthernet0/0
          mode: trunk
          trunk:
            allowed_vlans: 100-104,1
            encapsulation: dot1q
        state: merged  #, replaced, deleted
```

Liste IV.8 : La Tache du Playbook 2.

❖ ***Exécution et résultat du Playbook***

```
ansible@ansible-virtual-machine:~/Bureau/Cisco_Siwtchs$ ansible-playbook -i hosts.yml all.yml -u admin -k
SSH password:

PLAY [Affectation des vlans aux interfaces des switch AC01 ou AC02] ************************************************

TASK [Affectation des vlans aux interfaces du switch AC01 (Vlan 100 et Vlan 101)] **********************************
ok: [S-AC01]

PLAY RECAP *********************************************************************************************************
S-AC01                     : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Figure IV. 9 : Résultat du playbook 2 exécuté sur le switch S-AC01

IV.1.3.4 Création et exécution des Playbooks sur les switches

❖ ***Création du Playbook pour le Routeur FAI***

Tache à effectuer sur le routeur FAI *:* Ce Playbook a pour objectif de configurer le routeur FAI. En se connectant au routeur FAI via la connexion network_cli, le Playbook exécute plusieurs tâches. Tout d'abord, la configuration du client DHCP est effectuée sur l'interface g0/0 du routeur. Cela est réalisé en ajoutant les lignes de configuration nécessaires pour activer le client DHCP, puis en activant l'interface. Ensuite, toutes les configurations SNMP existantes

sont supprimées à l'aide du module cisco.ios.ios_snmp_server avec l'état "deleted". Enfin, la configuration SNMP est mise en place en ajoutant les lignes de configuration pour définir la communauté SNMP, spécifier l'hôte SNMP avec sa version et les autorisations en lecture seule, activer les traps SNMP, puis enregistrer les modifications de configuration. Ces tâches permettent de configurer le routeur FAI avec un client DHCP actif, de supprimer les configurations SNMP existantes et de mettre en place une nouvelle configuration SNMP pour la surveillance et la gestion du routeur. Voir la liste IV.9.

```
- name: Configuration du routeur FAI
  hosts: R-FAI
  connection: network_cli
  gather_facts: false
  tasks:
    - name: Configuration du DHCP Client sur le routeur
      ios_config:
        lines:
        - int g0/0
        - ip add dhcp
        - no shut
        - do wr

    - name: Suprimer tous les configuration SNMP
      cisco.ios.ios_snmp_server:
        state: deleted

    - name: Configuration SNMP
      ios_config:
        lines:
        - snmp-server community snmp-fai ro
        - snmp-server host 192.168.104.5 version 2c ro
        - snmp-server enable traps
        - do wr
```

Liste IV.9 : La Tache du Routeur FAI.

❖ ***Exécution et résultat du Playbook***

```
ansible@ansible-virtual-machine:~/Bureau/Cisco_Routers$ ansible-playbook -i hosts.yml playbook.yml -u admin -k
SSH password:

PLAY [Configuration du DHCP Client sur le routeur] ****************************************************

TASK [Gathering Facts] ********************************************************************************
[WARNING]: Ignoring timeout(10) for ansible.legacy.ios_facts
ok: [R-FAI]

TASK [configuration commands] *************************************************************************
[WARNING]: To ensure idempotency and correct diff the input configuration lines should be similar to how they appear
changed: [R-FAI]

TASK [Suprimer tous les configuration SNMP] ***********************************************************
changed: [R-FAI]

TASK [Configuration SNMP] *****************************************************************************
changed: [R-FAI]

PLAY RECAP ********************************************************************************************
R-FAI                      : ok=4    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Figure IV.10 : Résultat du playbook exécuté sur le routeur FAI

IV.1.3.5 Création et exécution des Playbooks sur les équipements Cisco

❖ ***Création d'un Playbook pour le routeur FAI et tous les commutateurs***

Tache à effectuer sur l'ensemble des équipements Cisco : Ce Playbook vise à configurer les paramètres de base sur tous les équipements Cisco. Il se connecte aux équipements Cisco via la connexion network_cli et exécute les tâches suivantes. La première tâche configure le

"Banner Motd" avec un message de test personnalisé. Ensuite, la configuration de base comprend la désactivation de la recherche de domaine, la configuration d'un mot de passe secret, la configuration de la console, la synchronisation des journaux, la configuration d'un délai d'inactivité infini pour la console, la configuration des lignes VTY, la synchronisation des journaux pour les connexions VTY et la configuration d'un délai d'inactivité infini pour les connexions VTY. Ces tâches permettent d'établir une configuration cohérente sur tous les équipements Cisco, améliorant ainsi la sécurité et la gestion des équipements. Voire la liste IV.10.

```
- name: Configuration de base
  hosts: Cisco
  gather_facts: false
  connection: network_cli
  tasks:
    - name: configuration de la "Banner Motd" sur tous les équipments Cisco
      ios_config:
        lines:
        - banner motd '####### Bonjour ceci est un message de test ! ***#####'
        - do wr

    - name: Config de base des équipments Cisco
      ios_config:
        lines:
        - no ip domain-lookup
        - enable secret karim.dina
        - line console 0
        - logging synchronous
        - exec-timeout 0 0
        - line vty 0 15
        - logging synchronous
        - exec-timeout 0 0
        - do wr
```

Liste IV.10 : Les Tâches des équipements Cisco (Routeur et Switches).

❖ ***Exécution et résultat du Playbook***

```
ansible@ansible-virtual-machine:~/Bureau/Cisco_Siwtchs$ ansible-playbook -i hosts.yml all.yml -u admin -k
SSH password:

PLAY [Configuration de base] ***************************************************************************

TASK [configuration de la "Banner Motd" sur tous les équipments Cisco] *********************************
[WARNING]: To ensure idempotency and correct diff the input configuration lines should be similar to how they appe
changed: [R-FAI]
changed: [S-AC02]
changed: [S-AC01]
changed: [S-DIST]
changed: [S-CORE]
changed: [S-DTC]

TASK [Config de base des équipments Cisco] *************************************************************
changed: [R-FAI]
changed: [S-AC02]
changed: [S-AC01]
changed: [S-DIST]
changed: [S-CORE]
changed: [S-DTC]

PLAY RECAP *********************************************************************************************
R-FAI                      : ok=2    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
S-AC01                     : ok=2    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
S-AC02                     : ok=2    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
S-CORE                     : ok=2    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
S-DIST                     : ok=2    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
S-DTC                      : ok=2    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Figure IV.11 : Résultat du playbook exécuté sur les équipements Cisco

IV.1.3.6 Création et exécution des Playbooks sur le serveur Windows

❖ ***Création d'un Playbook pour le Serveur Windows***

Tache à effectuer sur le serveur Windows : Ce Playbook a pour objectif d'automatiser l'installation des services AD (Active Directory) sur une machines Windows. En se connectant aux machines hôtes du groupe "Windows", le Playbook exécute les tâches suivantes. Tout d'abord, il installe la fonctionnalité des services AD en spécifiant le nom de la fonctionnalité, "AD-Domain-Services", et en incluant les outils de gestion et les sous-fonctionnalités associées. Ensuite, il crée un nouveau domaine et une nouvelle forêt en spécifiant le nom de domaine DNS et le mot de passe en mode de sauvegarde. Enfin, il crée un administrateur de domaine en spécifiant le nom de l'utilisateur, le mot de passe et en l'ajoutant au groupe "Admins du domaine". Ces tâches automatisées permettent de simplifier et d'accélérer le déploiement des services AD, facilitant ainsi la gestion des utilisateurs et des ressources dans un environnement Windows basé sur le domaine. Voire la liste IV.11.

```
- hosts: Windows-server
  gather_facts: no
  tasks:
    - name: Installer la fonctionnalité des services AD
      win_feature:
        name: AD-Domain-Services
        include_management_tools: yes
        include_sub_features: yes
        state: present

    - name: Créer un nouveau domaine et une nouvelle forêt
      win_domain:
        dns_domain_name: Karim.Madina
        safe_mode_password: Karim@Madina
        domain_netbios_name: AUTOMATI

    - name: Créer un administrateur de domaine
      win_domain_user:
        name: admin
        password: Karim@Madina
        state: present
        groups:
          - Admins du domaine
```

Liste IV.11 : Les Taches du serveur Windows

❖ ***Exécution et résultat du Playbook***

```
ansible@ansible-virtual-machine:~/Bureau$ ansible-playbook -i hosts.yml playbook.yml

PLAY [Installation de quelques services sur Windows serveur] ***************************

TASK [Installer la fonctionnalité des services AD] *********************************
ok: [Windows-server]

TASK [Créer un nouveau domaine et une nouvelle forêt] ******************************
ok: [Windows-server]

TASK [Créer un administrateur de domaine] ******************************************
changed: [Windows-server]

PLAY RECAP *************************************************************************
Windows-server             : ok=3    changed=1    unreachable=0    failed=0    skipped=0
```

Figure IV.12 : Résultat du playbook exécuté sur le serveur Windows

Simulation avec Zabbix

IV.1.4 Gestion des comptes zabbix

Lorsqu'un utilisateur se connecte à Zabbix, il doit fournir ses informations d'identification pour accéder à son compte. Les utilisateurs peuvent avoir des rôles différents, tels que : administrateur système, utilisateur avancé ou utilisateur standard, et les autorisations associées à leur compte peuvent varier en conséquence.

Les comptes Zabbix sont importants pour la sécurité et la gestion de Zabbix, car ils permettent aux administrateurs de contrôler qui peut accéder aux informations sensibles sur les systèmes surveillés. En créant des comptes avec des autorisations spécifiques, les administrateurs peuvent garantir que seules les personnes autorisées ont accès aux données de surveillance et peuvent effectuer des actions sur le système de surveillance.

Pour cela nous allons suivre les étapes suivent afin de créer un compte administrateur qui possède tous les droits en lecture-écriture.

IV.1.4.1 Création d'un compte administrateur avec le nom 'Superviseur'

Pour créer un compte administrateur sur Zabbix, les étapes numérotées de 1 à 3 dans la figure IV.13 doivent être suivies.

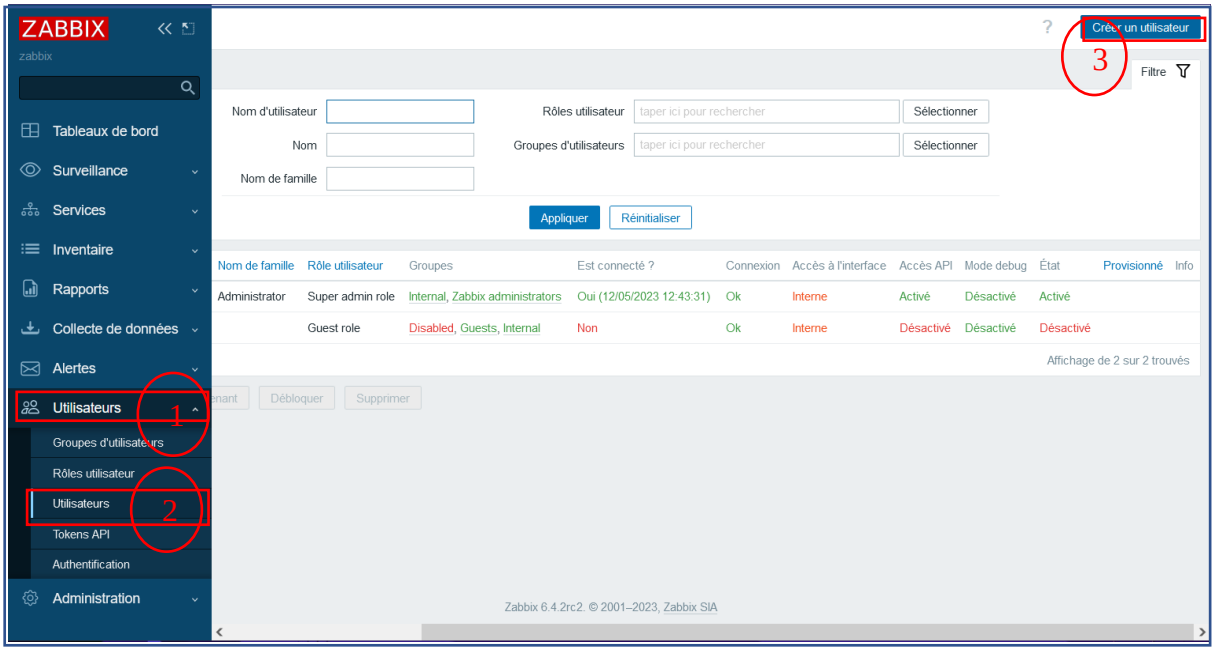


Figure IV.13 : Création d'un compte admin sur Zabbix.

IV.1.4.2 Remplir les informations du compte

Le champ le plus important c'est le Groupe si on veut avoir un utilisateur qui possède tous les droits il faut qu'il appartienne au groupe administrateur. Voir la figure IV.14.

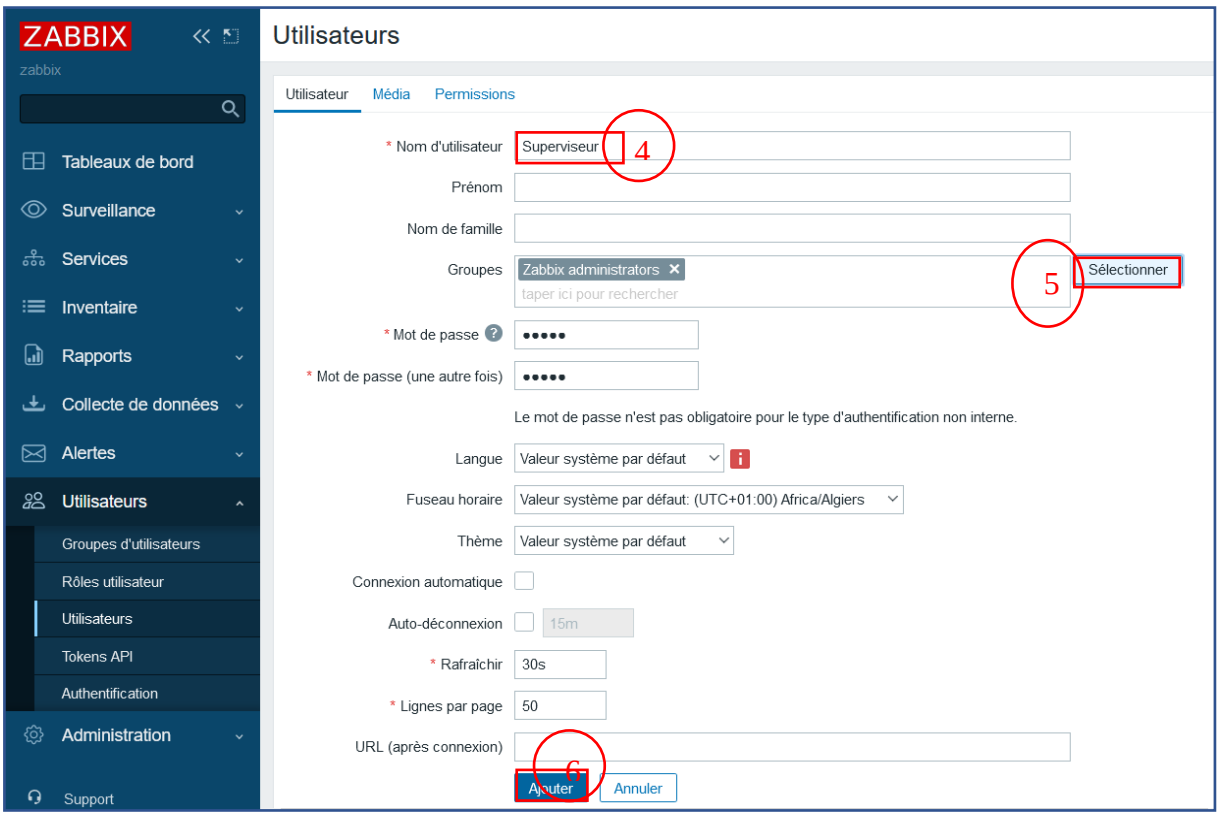


Figure IV.14 : Remplissage des informations d'utilisateur.

IV.1.4.3 Sélectionner le rôle d'utilisateur :

Dans cette partie, le rôle "Super admin" doit être sélectionné afin d'obtenir tous les droits, ce qui permet d'accéder, de visualiser et de modifier les informations. Voir la figure IV.15.

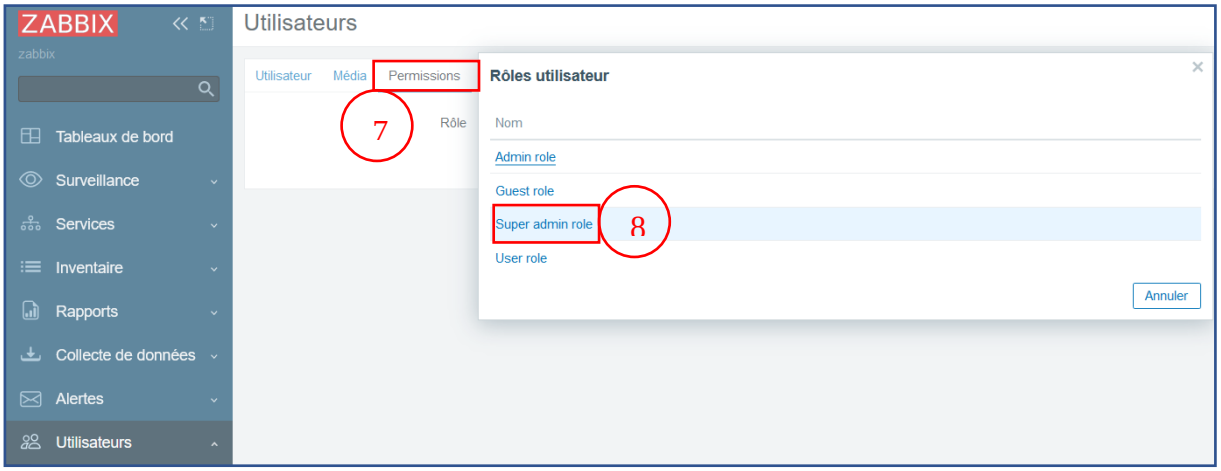


Figure IV.15 : Sélectionner le rôle d'utilisateur.

La figure IV.16 présente toutes les permissions, éléments, services, modules et actions accordés à l'utilisateur "***Superviseur***" avec le rôle "***Super admin***".

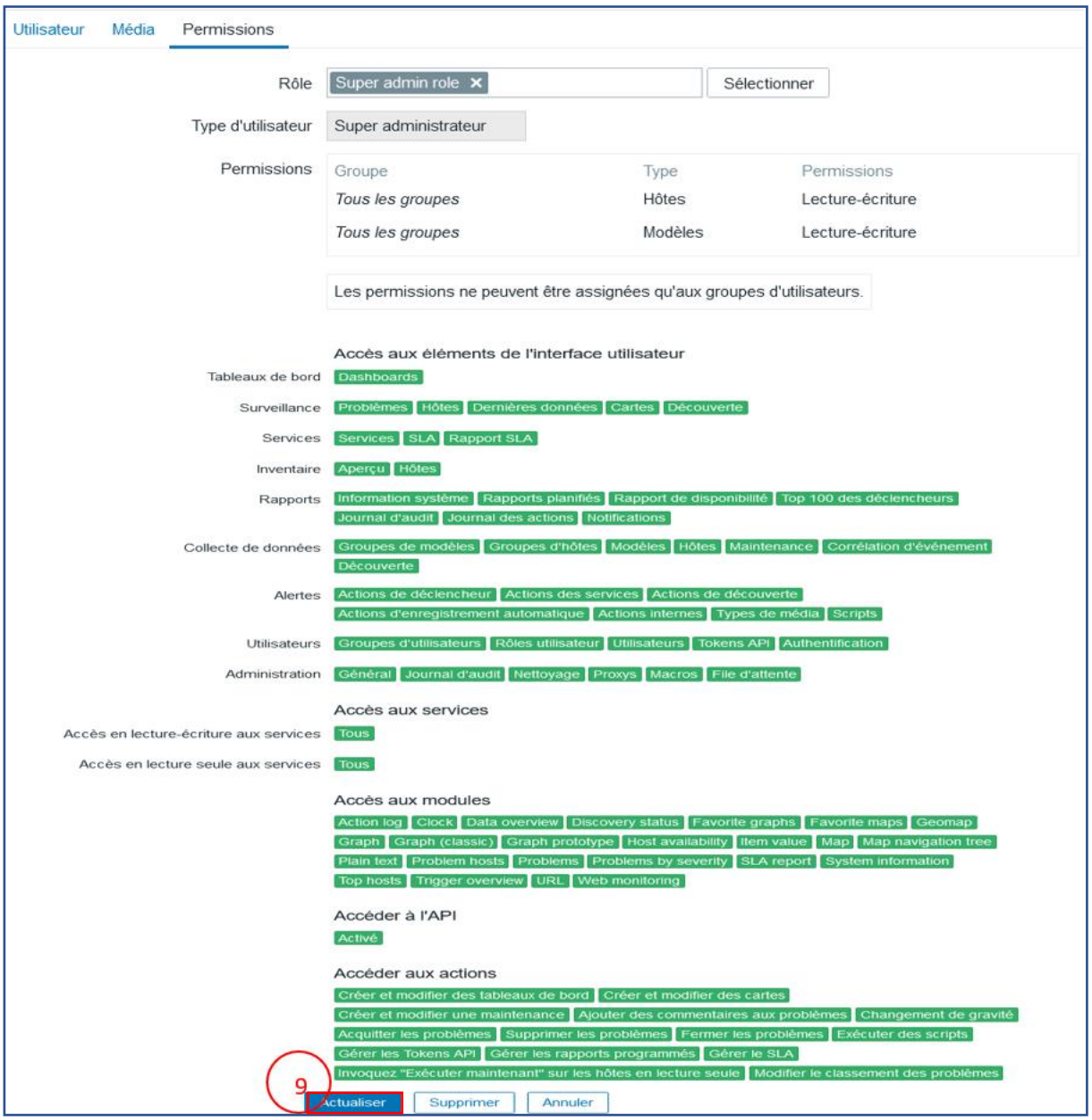
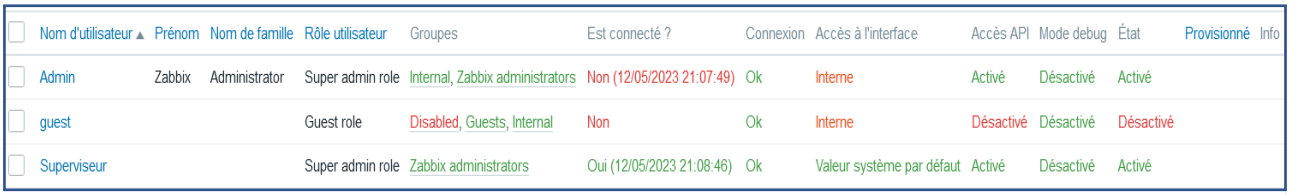


Figure IV.16 : Les permissions accordées à l'utilisateur '***Superviseur'.***

IV.1.4.4 Connexion avec le compte superviseur :

La figure suivante montre que le compte "***Superviseur***" est bien activé et connecté.



Nom d'utilisateur ▲	Prénom	Nom de famille	Rôle utilisateur	Groupes	Est connecté ?	Connexion	Accès à l'interface	Accès API	Mode debug	État	Provisionné	Info
Admin	Zabbix	Administrator	Super admin role	Internal, Zabbix administrators	Non (12/05/2023 21:07:49)	Ok	Interne	Activé	Désactivé	Activé		
guest			Guest role	Disabled, Guests, Internal	Non	Ok	Interne	Désactivé	Désactivé	Désactivé		
Superviseur			Super admin role	Zabbix administrators	Oui (12/05/2023 21:08:46)	Ok	Valeur système par défaut	Activé	Désactivé	Activé		

Figure IV.17 : Etat du compte '***Superviseur***'

IV.1.5 Ajouter des équipements à surveiller

Maintenant que nous avons créé un compte administrateur, l'étape suivante sera d'ajouter des équipements à surveiller. Avant d'ajouter ces équipements, il est nécessaire de configurer les équipements de niveau 2 (les commutateurs) avec des adresses IP.

IV.1.5.1 Ajouter des commutateurs :

Le premier commutateur à ajouter est le commutateur Core. Sa configuration sera détaillée. Pour les autres commutateurs, le principe sera le même. Par conséquent, nous n'avons pas besoin de détailler leur configuration afin de ne pas rendre cette partie trop longue et d'éviter les répétitions.

1) Configuration du commutateur Core avec une adresse IP

Les adresses IP des commutateurs sont configurées sur Vlan 104. Voire la Figure IV.18.

```
SW-Core>en
SW-Core#config t
Enter configuration commands, one per line.  End with CNTL/Z.
SW-Core(config)#int vlan 104
SW-Core(config-if)#ip add 192.168.104.8 255.255.255.0
SW-Core(config-if)#no shut
SW-Core(config-if)#end
SW-Core#wr
Building configuration...
```

Figure IV.18 : Configuration du commutateur Core avec une adresse IP

2) Création d'un hôte sur zabbix

Avant de créer un nouvel hôte dans Zabbix, il convient de noter que le serveur Zabbix lui-même est déjà considéré comme un hôte. En effet, Zabbix surveille ses propres composants pour s'assurer qu'ils fonctionnent correctement et pour identifier tout éventuel problème de performance ou de configuration. Cette surveillance des composants internes de Zabbix permet d'obtenir une vue complète de l'état de l'environnement de surveillance. Voir la figure IV.19.

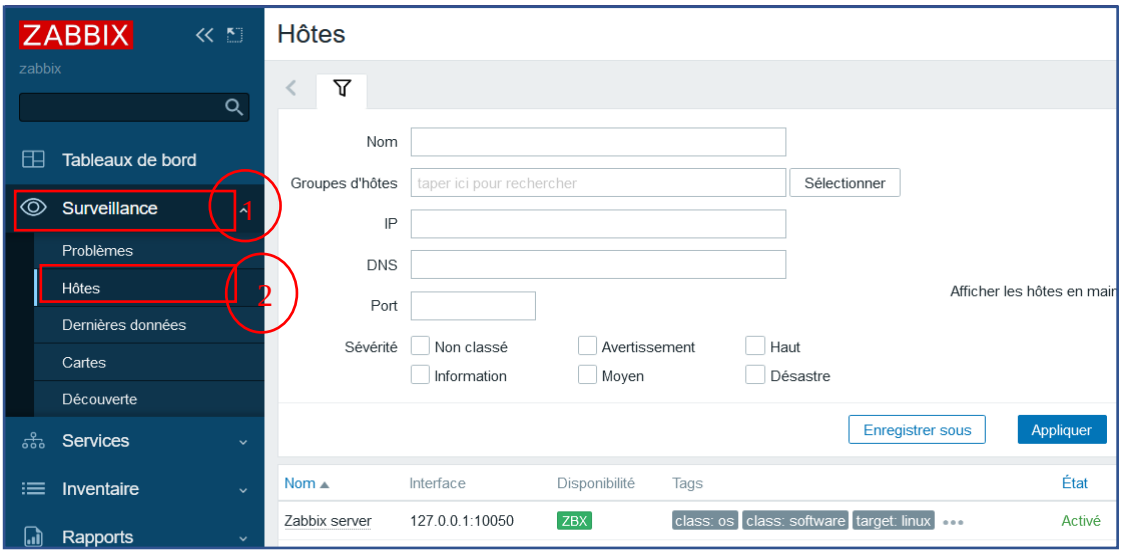


Figure IV.19 : Supervision du serveur zabbix

Pour ajouter un commutateur dans Zabbix, il est nécessaire de suivre les étapes illustrées dans les figures suivantes :

Tout d'abord, il faut remplir le nom de l'hôte, puis sélectionner le modèle approprié afin que Zabbix puisse choisir une Template spécifique pour l'équipement. Dans notre cas, il s'agit d'un équipement Cisco avec un agent SNMP. Ensuite, il est important de choisir les groupes d'hôtes correspondants pour regrouper des hôtes similaires et faciliter la gestion de la surveillance, la configuration des paramètres de surveillance de manière centralisée, la génération de rapports et de tableaux de bord pour chaque groupe d'hôtes, et la définition des permissions d'accès pour les utilisateurs de Zabbix.

Enfin, il est essentiel de configurer le protocole SNMP en indiquant l'adresse IP de l'équipement, la version du protocole et la communauté SNMP. Cette communauté est déjà configurée par défaut, il suffit donc simplement de la copier pour l'utiliser à l'étape suivante. Voir les figure IV.20 et IV.21.

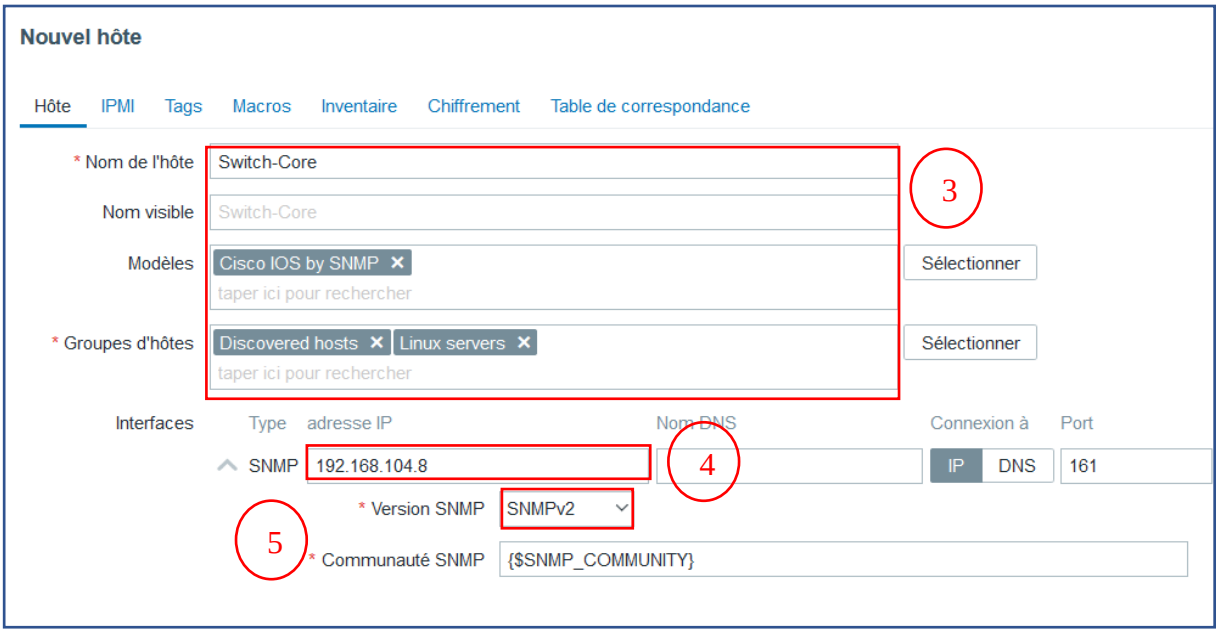


Figure IV.20 : Configuration d'un nouvel hôte

❖ ***Remarque :***

Les templates Zabbix sont des modèles de configuration prédéfinis écrits en YAML, un format de représentation de données textuelles facilement lisible par les humains et les machines. Ils peuvent être utilisés pour surveiller différents types de services et d'applications, car ils contiennent des éléments de surveillance tels que des seuils de déclenchement, des graphiques et des alertes.

En utilisant les templates Zabbix, les utilisateurs peuvent gagner du temps et simplifier la configuration de la surveillance de leurs hôtes. Les utilisateurs peuvent également créer leurs propres templates personnalisés pour répondre à leurs besoins spécifiques et les partager avec la communauté.

La prochaine étape consiste à configurer une macro avec sa valeur. Cette valeur sera utilisée lors de la configuration du protocole SNMP au niveau des commutateurs. Il s'agit en quelque sorte d'un mot de passe partagé entre le serveur Zabbix et le commutateur Core.

❖ ***Définition d'une Macro :***

Les macros dans Zabbix sont des variables qui permettent de stocker des valeurs dynamiques ou statiques, utilisées pour personnaliser et adapter le comportement du système de surveillance.

Elles sont principalement utilisées pour paramétrer des objets tels que les hôtes, les éléments, les déclencheurs, les actions, les graphiques, les modèles, etc. Une macro dans Zabbix est représentée par une chaîne de caractères encadrée par des accolades, par exemple {$MACRO}. Voir ça configuration dans la figure IV.21.

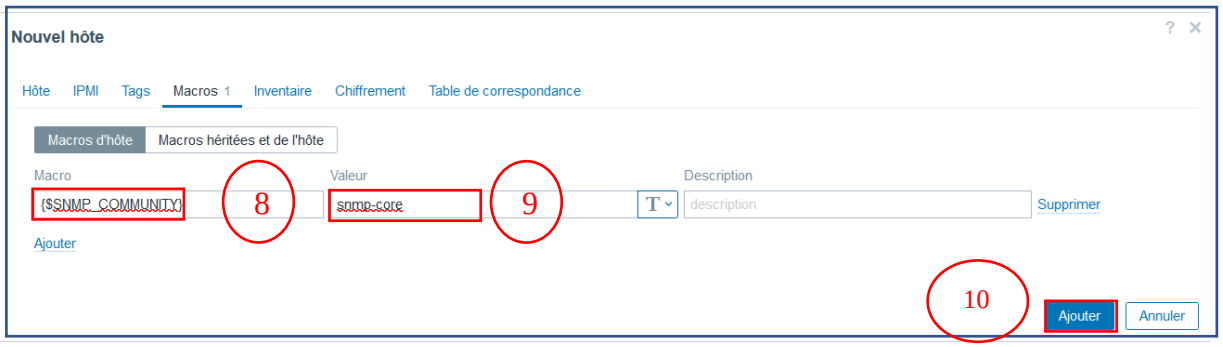


Figure IV.21 : Configuration d'une Macro

3) Configuration du protocol SNMP au niveau des Commutateurs (Core)

La configuration du SNMP sur un commutateur implique plusieurs étapes :

1. Activation du SNMP : Le service SNMP est activé sur le commutateur en utilisant la commande « ***snmp-server*** ».
2. Communautés SNMP : Les communautés SNMP sont configurées pour permettre l'accès au commutateur avec une valeur (mot de passe partagé). On définit une communauté en lecture seule (read-only) ou une communauté en lecture/écriture (read-write). Dans notre cas, nous utilisons le mot de passe partagé « ***snmp-core*** » entre le serveur et le commutateur Core.
3. Niveaux d'accès SNMP : Les niveaux d'accès SNMP sont définis pour spécifier les permissions. Nous configurons la communauté SNMP en mode « ***RO*** » (read-only) pour permettre uniquement la récupération d'informations.
4. Adresses IP autorisées : Les adresses IP autorisées à accéder au commutateur via SNMP sont spécifiées pour des raisons de sécurité. Dans notre cas, nous autorisons l'adresse IP de notre serveur de supervision Zabbix, avec l'adresse IP ***192.168.104.5***.

5. La version SNMP : L'utilisation de la version SNMP appropriée garantit une compatibilité optimale avec les systèmes de gestion de réseau et permet une communication efficace entre le commutateur et le serveur de supervision. Dans notre cas nous utilisons la « ***version 2C*** ».
6. Activation des traps SNMP : Pour recevoir des notifications d'événements importants du commutateur, il est nécessaire d'activer les traps SNMP. La commande utilisée pour cela est « ***snmp-server enable traps*** ».

```
SW-Core(config)#snmp-server community snmp-core ro
SW-Core(config)#snmp-server host 192.168.104.5 version 2c snmp-core
SW-Core(config)#snmp-server enable traps
SW-Core(config)#end
SW-Core#wr
Building configuration...
```

Figure IV.22 : Configuration du protocol SNMP sur le commutateur Core

Après avoir configuré le commutateur et le serveur Zabbix, on peut observer dans la figure IV.23 que le commutateur Core est ajouté au serveur Zabbix. Il est important de noter que le commutateur doit être ajouté au serveur avant d'activer le protocole SNMP sur celui-ci.

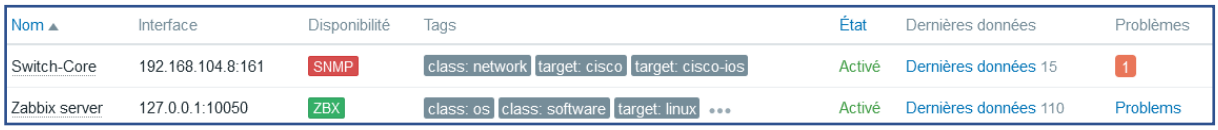


Figure IV.23 : Surveillance des hotes

Cependant, on peut remarquer que la disponibilité du SNMP sur le commutateur est affichée en rouge, ce qui indique qu'il y a un problème. Il est probable que cela soit dû au fait que le serveur Zabbix n'est pas encore connecté à la topologie sur GNS3.

Lorsque nous cliquons sur le problème, nous constatons que les requêtes ping échouent Figure IV.24. Cela confirme notre hypothèse précédente selon laquelle le serveur n'est pas connecté à la topologie sur GNS3.

Figure IV.24 : Probléme des requettes ICMP

Pour résoudre ce problème, il est nécessaire de connecter le serveur Zabbix à la topologie réseau sur GNS3.

Après avoir résolu le problème et connecté le serveur Zabbix à la topologie sur GNS3, nous pouvons effectuer un test de ping depuis le serveur Zabbix vers le commutateur Core et également vers Internet en passant par la topologie pour s'assurer que tout est correctement connecté. Voire les figures IV.25, IV.26 et IV.27.

```
zabbix@Monitorig-Server:~$ ping 192.168.104.8
PING 192.168.104.8 (192.168.104.8) 56(84) bytes of data.
64 bytes from 192.168.104.8: icmp_seq=1 ttl=255 time=1.17 ms
64 bytes from 192.168.104.8: icmp_seq=2 ttl=255 time=1.53 ms
64 bytes from 192.168.104.8: icmp_seq=3 ttl=255 time=1.13 ms
64 bytes from 192.168.104.8: icmp_seq=4 ttl=255 time=1.25 ms
64 bytes from 192.168.104.8: icmp_seq=5 ttl=255 time=1.39 ms
```

Figure IV.25 : Test Ping du serveur Zabbix ver le commutateur Core

```
SW-Core#ping 192.168.104.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.104.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
SW-Core#
```

Figure IV.26 : Test ping du commutateur Core ver le serveur Zabbix

```
zabbix@Monitorig-Server:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=2 ttl=126 time=75.7 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=126 time=79.7 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=126 time=71.9 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=126 time=69.3 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=126 time=64.6 ms
```

Figure IV.27 : Test Ping du serveur zabbix ver Internet

Après avoir effectué les tests de ping, nous constatons que les résultats sont positifs et que la connectivité fonctionne correctement. Cela est illustré dans les figures précidents où les tests de ping ont été exécutés avec succès.

Et après quelques minutes, nous constatons que le problème est résolu et que le commutateur Core est ajouté avec succès, sans aucun problème. Voir la figure IV.28.

Cela signifie que le protocole SNMP fonctionne correctement et que le serveur Zabbix peut maintenant surveiller et gérer le commutateur Core grâce à la configuration adéquate du SNMP.

Figure IV.28 : Etat du switch Core après avoir résolu le problème de connectivité

❖ Remarque

Le même principe et les mêmes étapes (de 1 à 3) doivent être appliqués pour ajouter les autres commutateurs, à savoir : ***Dist***, ***SW-DTC***, ***AC-01*** et ***AC-02***.

IV.1.5.2 Ajouter le routeur FAI

Le routeur FAI est également un équipement Cisco, tout comme les commutateurs précédents. Par conséquent, sa configuration sur Zabbix est similaire à celle des commutateurs.

Après avoir configuré le routeur FAI et suivi les mêmes étapes de configuration des commutateurs, Zabbix a détecté un problème (voir la figure IV.29) : les deux interfaces eth0/0 et eth0/1 sont configurées en mode Half-Duplex.

Figure IV.29 : problème du routeur FAI détecté par Zabbix

Pour résoudre ce problème, nous allons configurer ces deux interfaces en mode Full-Duplex. Les configurations requises sont illustrées dans la Figure IV.30.

```
R-FAI(config)#int range eth0/0-1
R-FAI(config-if-range)#duplex full
R-FAI(config-if-range)#end
R-FAI#wr
*May 20 17:41:21.433: %SYS-5-CONFIG_I: Configured from console by console
R-FAI#wr
Building configuration...
[OK]
```

Figure IV.30 : Résoudre le problème du routeur FAI

Après avoir configuré les interfaces eth0/0 et eth0/1 en mode Full-Duplex, nous constatons que le problème a été résolu, comme le montre la Figure IV.31.

Figure IV.31 : état du routeur FAI après la résolution du problème

IV.1.5.3 Ajouter le pare-feu FortiGate (FG)

Avant d'ajouter un nouvel hôte sur Zabbix, il est nécessaire de procéder à l'ajout du modèle (Template) FortiGate. Étant donné que Zabbix ne dispose pas de modèles pour tous les équipements, il est indispensable de télécharger ce modèle à partir du site officiel de FortiGate. Les modèles, comme nous l'avons déjà mentionné précédemment, sont utilisés pour surveiller différents types de services et d'applications. Ils contiennent des éléments de surveillance tels que des seuils de déclenchement, des graphiques et des alertes, etc.

1) Télécharger et importer la Template sur Zabbix :

D'abord il faut accéder au site officiel de Zabbix. Ensuite, on recherche la Template spécifique à FortiGate prenant en charge SNMP, en veillant à sélectionner la dernière version disponible pour bénéficier des fonctionnalités les plus récentes. Dans notre cas, on recommande la version "***template-fortigate_snmp_v2019***". Voire la figure IV.32.

Une fois qu'on a identifié la bonne Template, on la télécharge depuis le site officiel de Zabbix. Cette Template inclut les configurations prédéfinies nécessaires pour surveiller et collecter les données du pare-feu FortiGate, telles que les seuils de déclenchement, les graphiques et les alertes.

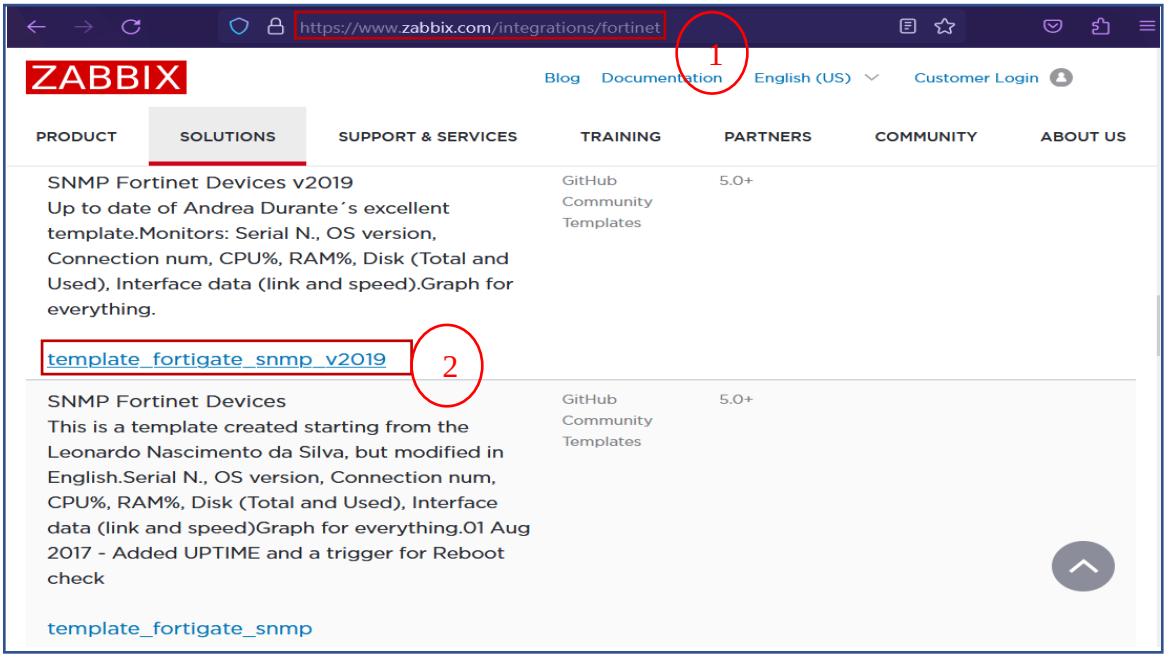


Figure IV.32 : Téléchargement de la Template FortiGate sur le site Zabbix

Une fois que la Template a été téléchargée, on peut procéder à son importation dans notre instance de Zabbix. Les étapes sont décrites dans les Figures IV.33 et IV.34.

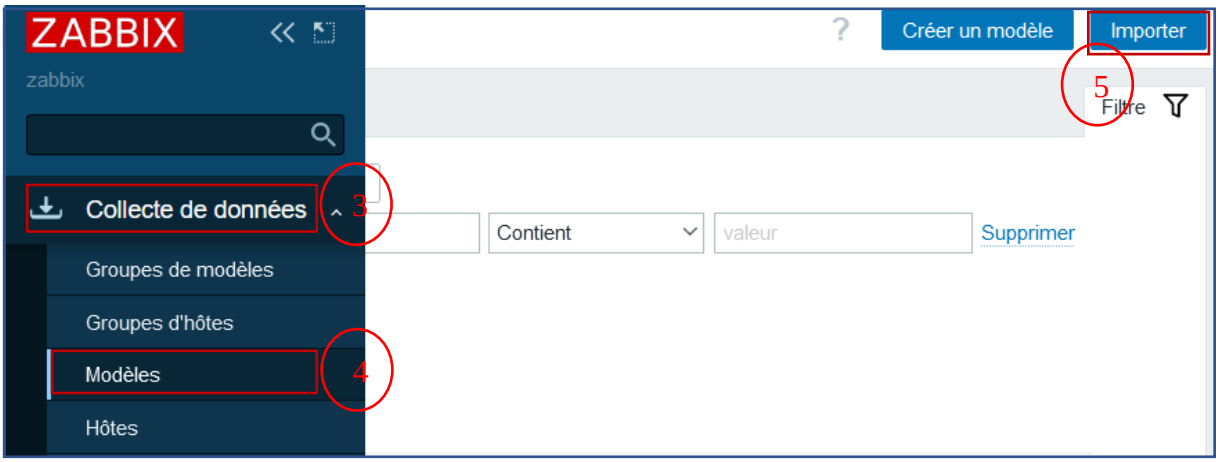


Figure IV.33 : Sectionnement de la Template

Une fois que nous avons sélectionné la Template, nous cliquons sur le bouton "Importer" pour procéder à son importation dans notre instance de Zabbix.

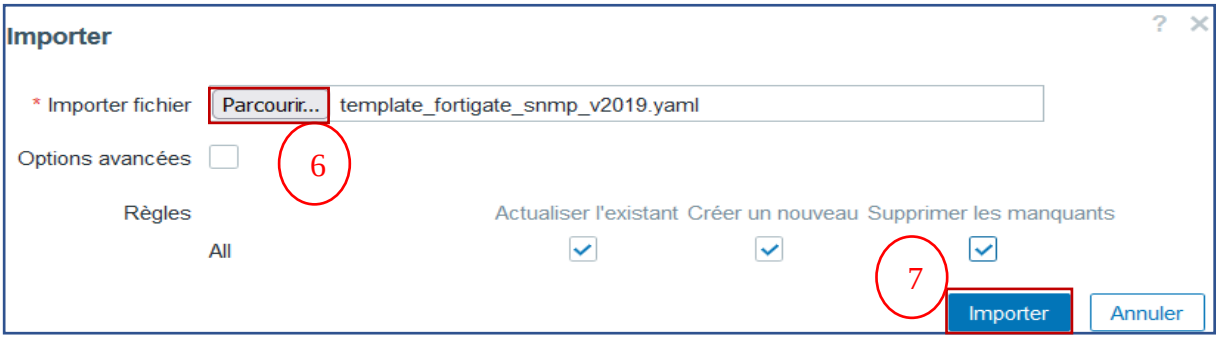


Figure IV.34 : Importation de la Template

2) Création d'un hôte et Association de la Template importé au pare-feu FortiGate

L'étape suivante consiste à associer la Template importée au pare-feu FortiGate afin de commencer la surveillance et l'analyse des performances de notre équipement. Pour cela, nous devons créer un nouvel hôte en suivant la même procédure que pour les équipements précédents. Ensuite, dans la configuration de l'hôte, nous sélectionnons le modèle (la Template importée) comme illustré dans la figure IV.35.

Enfin, nous cliquons sur le bouton "Ajouter". Cela permettra de surveiller et d'analyser les performances du pare-feu FortiGate à l'aide des paramètres prédéfinis dans la Template importée.

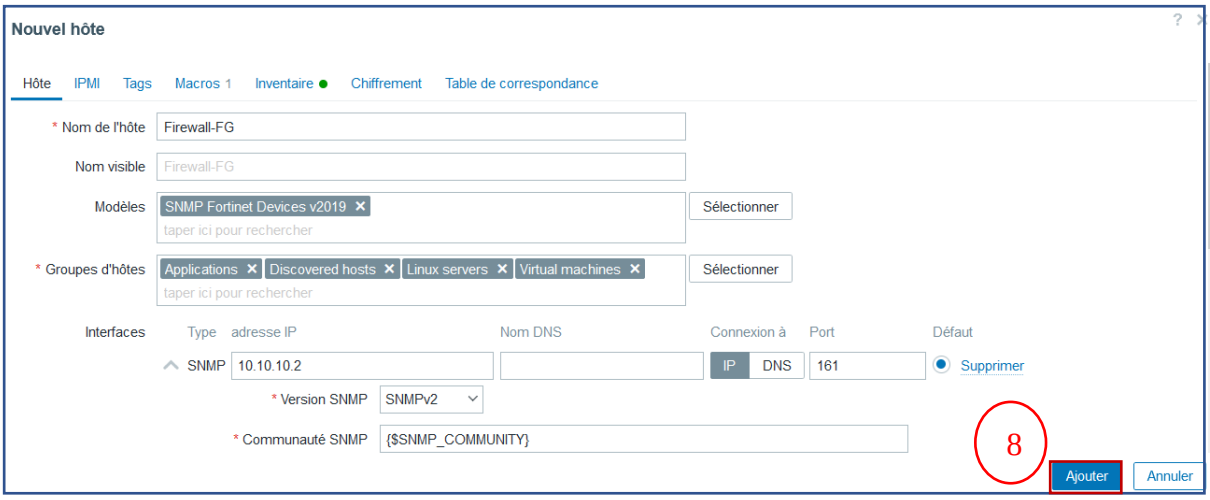


Figure IV.35 : Ajouter le pare-feu FortiGate

3) Configuration du protocole SNMP au niveau du pare-feu FortiGate

Le pare-feu FortiGate dispose d'une interface web conviviale qui facilite grandement les configurations. Pour configurer le protocole SNMP, nous devons accéder à cette interface et suivre les étapes illustrées dans les figures IV.36 et IV.37.

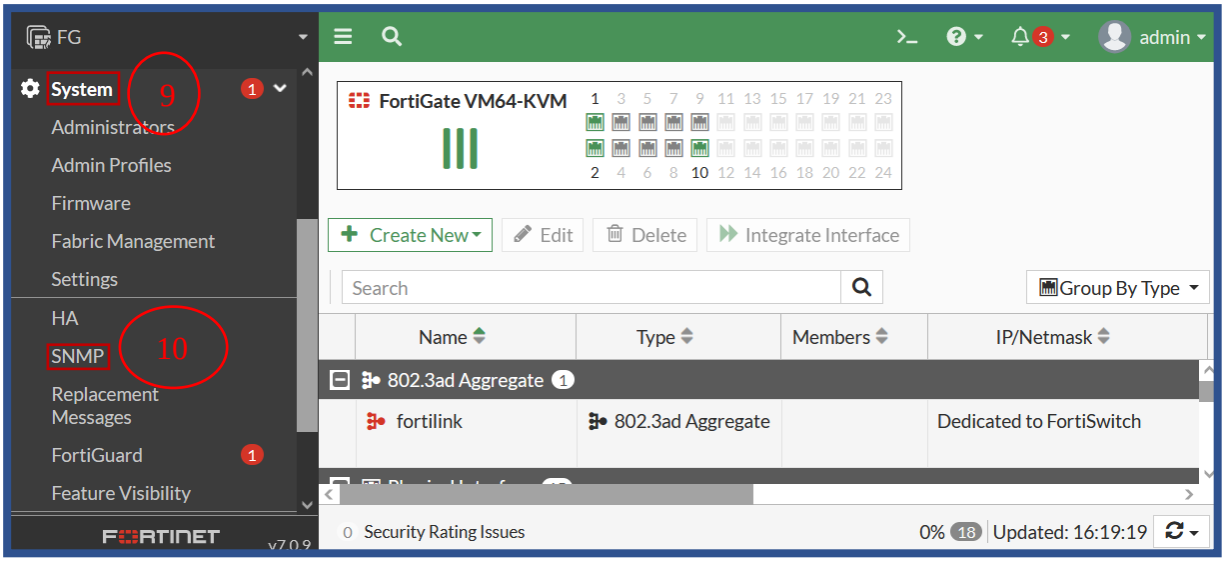


Figure IV.36 : Interface Web FortiGate

Une fois à l'intérieur de l'interface du FortiGate, nous accédons à la section SNMP et procédons à la configuration du protocole. Voire la figure IV.37

Tout d'abord, nous activons l'agent SNMP en cochant l'option correspondante. Ensuite, nous créons une nouvelle configuration SNMP en fournissant les informations nécessaires.

Dans notre cas, nous cochons la version 2 et définissons la communauté SNMP comme "***snmp-fg***". Il est également important de spécifier l'adresse IP du serveur Zabbix pour indiquer où les données SNMP doivent être envoyées. Enfin, nous enregistrons les configurations en cliquant sur le bouton "Appliquer". Ainsi, l'agent SNMP est activé sur le FortiGate et la nouvelle configuration est mise en place, permettant à Zabbix de collecter les données de performance et de surveillance de manière appropriée.

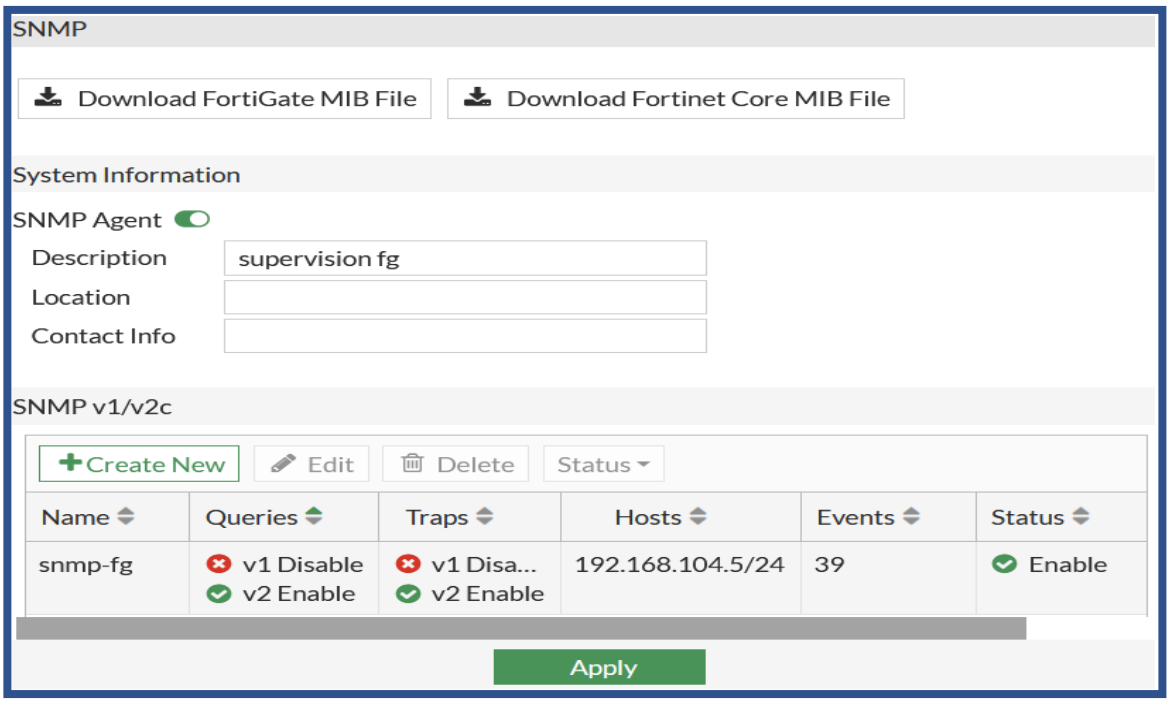


Figure IV.37 : configuration du protocole snmp sur FortiGate

Après avoir configuré le protocole SNMP sur Zabbix et sur le pare-feu, on constate que le pare-feu a été ajouté avec succès à Zabbix, sans aucun problème. Voir la figure IV.38.

Figure IV.38 : Etat du pare-feu FortiGate

IV.1.5.4 Ajouter le serveur Windows

Pour ajouter le serveur Windows à Zabbix, on doit installer et configurer le service SNMP sur le serveur.

1) Installer le service SNMP sur le serveur Windows

Pour installer le service SNMP, on suit les étapes illustrées dans les figures suivantes :

D'abord On clique sur "Ajouter des rôles et des fonctionnalités" sur le serveur Windows.

Figure IV.39 : Gestionnaire de serveur

Ensuit on sélectionne notre serveur (SER01) avec son adresse IP (192.168.104.6) et on clique sur "Suivant".

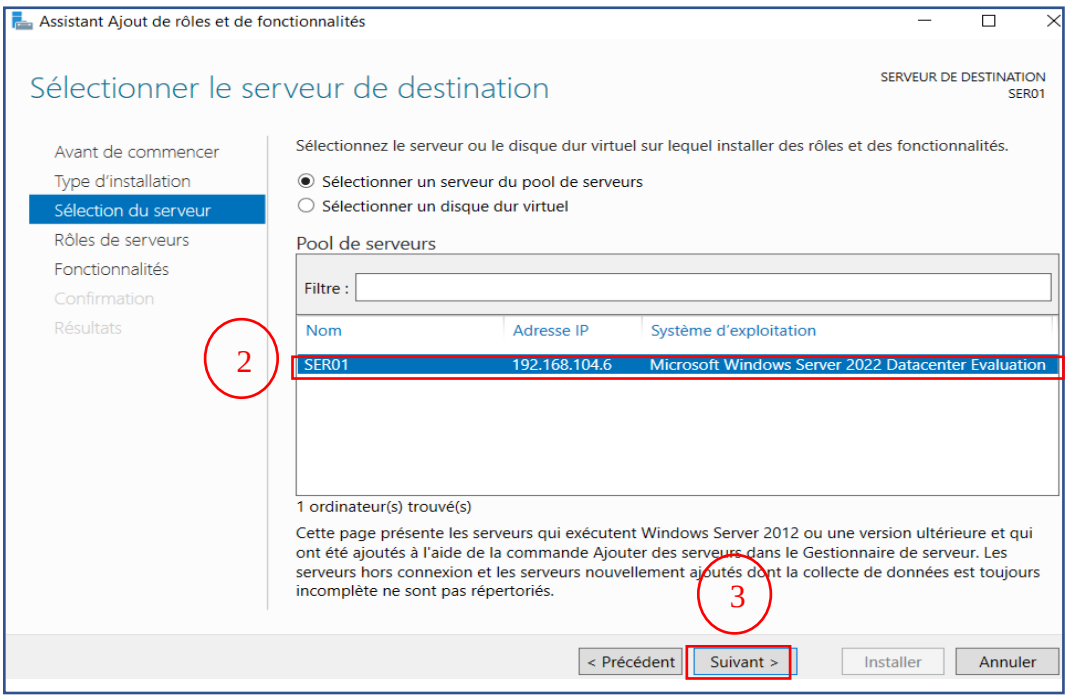


Figure IV.40 : Sélectionner le serveur de destination.

On coche la case "SERVICE SNMP" dans la liste des fonctionnalités à installer, puis on clique sur "Suivant".

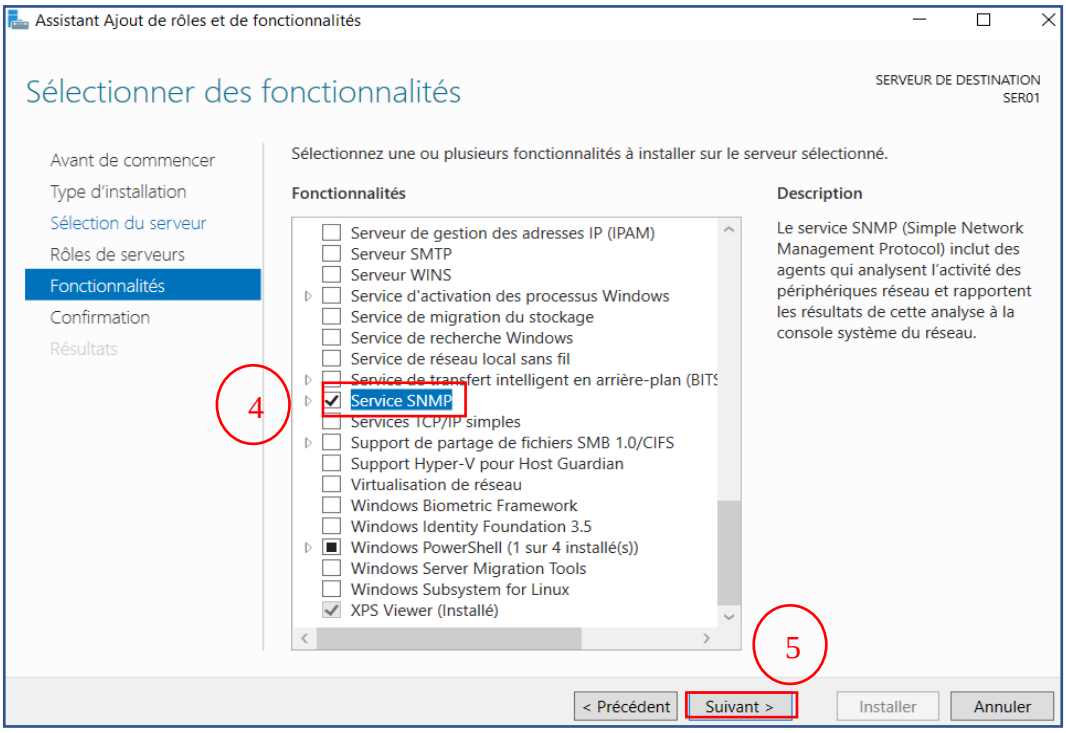


Figure IV.41 : Sélectionner le service à installer

Enfin, on clique sur "Installer" pour commencer l'installation du service SNMP.

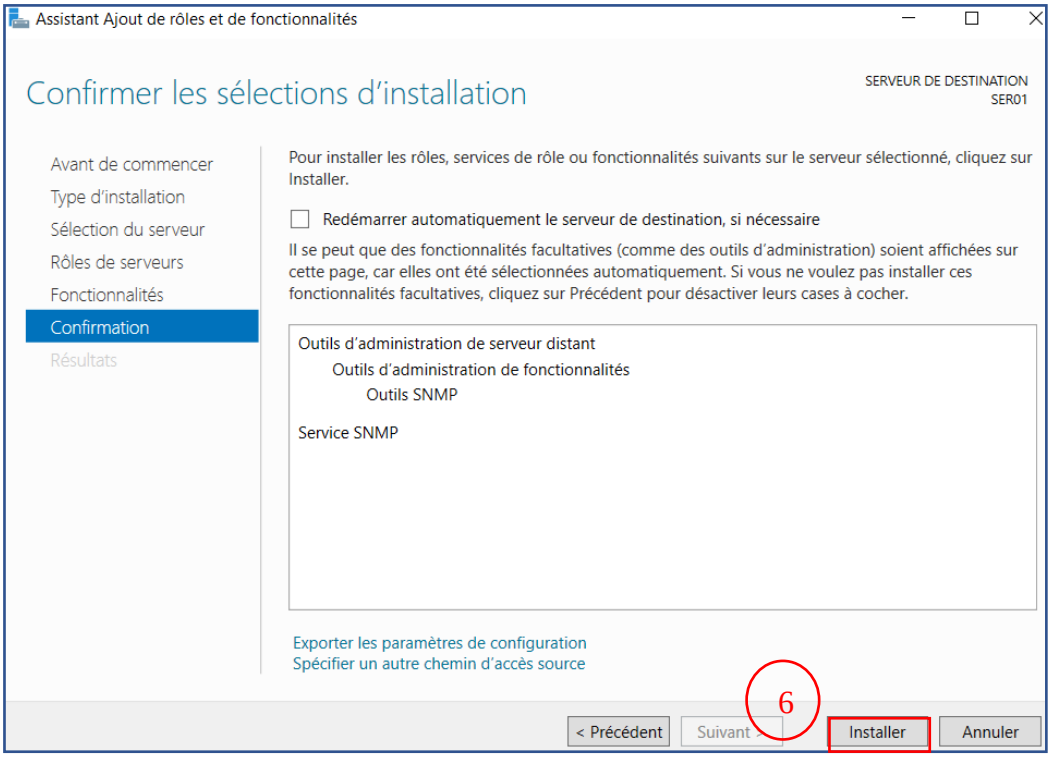


Figure IV.42 : Confirmer les sélections d'installation

2) Configuration du service SNMP sur le serveur Win 2022

Après l'installation du service SNMP, il est nécessaire de le configurer. Pour cela, dans le tableau de bord, on accède à l'option « ***Outils*** » puis on sélectionne « ***Services*** ». Une fois la fenêtre des services ouverte, on recherche le service SNMP. Voir la figure IV.43.

Après ça on effectue un clic droit sur ce service et on choisit l'option « ***Propriétés*** ».

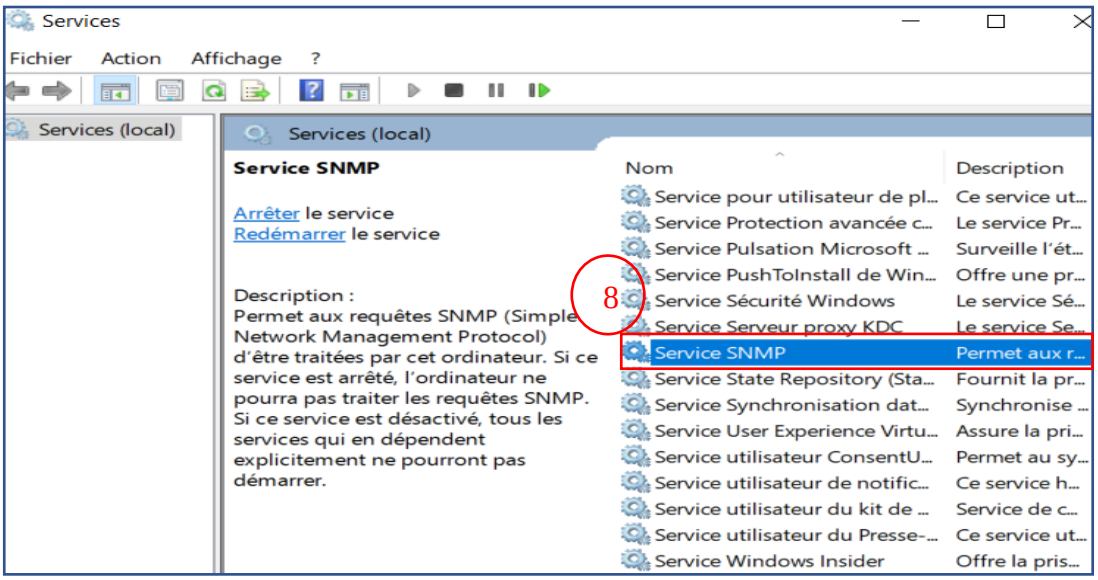


Figure IV.43 : Modifier le service SNMP

Ensuite, on accède à l'onglet « ***Agent SNMP*** » où on peut renseigner les informations de contact, d'emplacement et de service liées au service SNMP. Voir la figure IV.44.

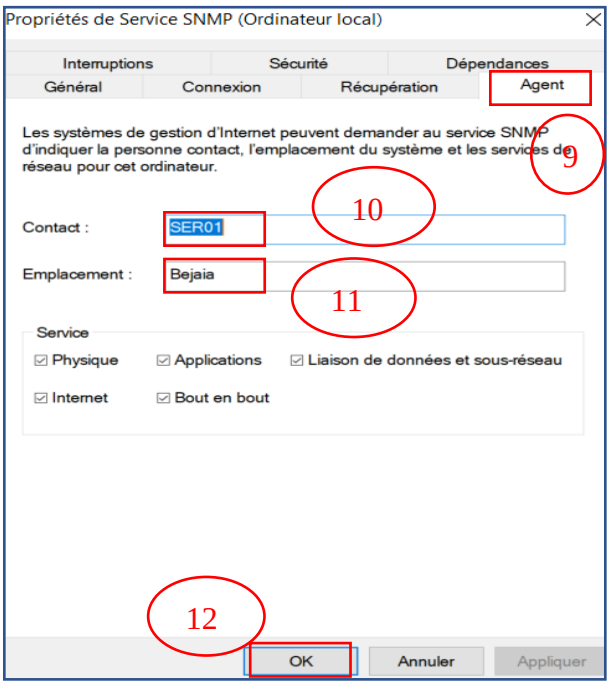


Figure IV.44 : Configuration de l'agent SNMP

Enfin, on passe à l'onglet « ***Sécurité*** » pour ajouter la communauté SNMP, définir les droits associés à cette communauté et spécifier l'adresse IP du serveur Zabbix. Cette étape permet de sécuriser l'accès au service SNMP et de configurer les paramètres de communication avec le serveur Zabbix. Voir la figure IV.45.

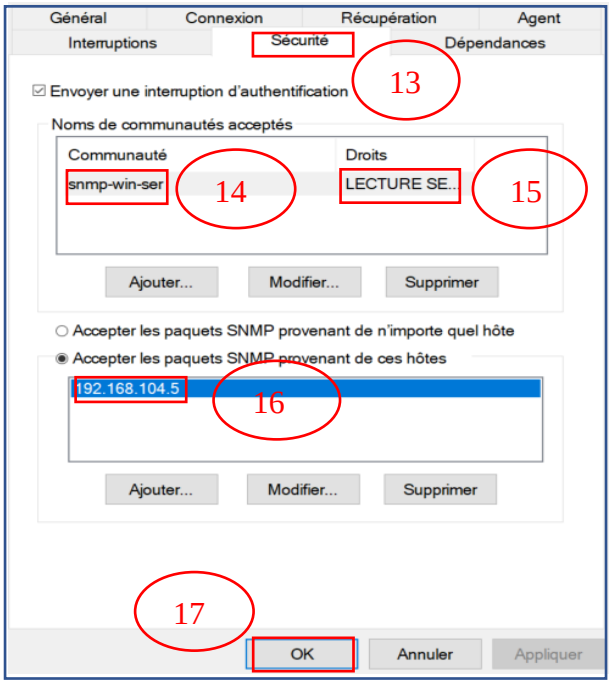


Figure IV.45 : Configuration de la sécurité SNMP

3) Configuration du protocole SNMP sur le serveur Zabbix

Pour ajouter le serveur Windows à Zabbix, les configurations à effectuer sont similaires à celles que nous avons déjà utilisées pour les hôtes précédents. Il suffit de télécharger la Template correspondante à partir du site officiel de Zabbix, puis de l'importer dans Zabbix, comme nous l'avons fait précédemment avec le pare-feu FortiGate.

Après avoir configuré avec succès tous les équipements et les avoir ajoutés à Zabbix, nous pouvons confirmer que la surveillance est opérationnelle. Dans la Figure IV.46, nous pouvons voir que tous les équipements sont présents dans Zabbix, prêts à être surveillés et à détecter d'éventuels problèmes.

Firewall-FG	10.10.10.2:161	SNMP		Activé
Routeur-FAI	10.10.10.1:161	SNMP	class: network target: cisco target: cisco-ios	Activé
SER01	192.168.104.6:161	SNMP	class: os target: windows	Activé
Switch-AC-01	192.168.104.102:161	SNMP	class: network target: cisco target: cisco-ios	Activé
Switch-AC-02	192.168.104.103:161	SNMP	class: network target: cisco target: cisco-ios	Activé
Switch-Core	192.168.104.8:161	SNMP	class: network target: cisco target: cisco-ios	Activé
Switch-Dist	192.168.104.101:161	SNMP	class: network target: cisco target: cisco-ios	Activé
Switch-DTC	192.168.104.104:161	SNMP	class: network target: cisco target: cisco-ios	Activé
Zabbix server	127.0.0.1:10050	ZBX	class: os class: software target: linux ...	Activé

Figure IV.46 : Vue d'ensemble de la surveillance des équipements sur Zabbix

Voici le tableau représentant les mots de passe partagés entre les équipements et Zabbix, ainsi que leurs adresses IP :

Les équipements ajoutés sur Zabbix	Les Valeur des Macros (mot de passe partagé)	Les adresses IP des équipements
Switch-Core	snmp-core	192.168.104.8
Switch-Dist	snmp-dist	192.168.104.101
Switch-DTC	snmp-dtc	192.168.104.104
Switch-AC-01	snmp-ac01	192.168.104.102
Switch-AC-02	snmp-ac02	192.168.104.103
Routeur-FAI	snmp-fai	10.10.10.1
Firewall-FG	snmp-fg	10.10.10.2
SER01	snmp-win-ser	192.168.104.6

Tableau IV.1 : Les adresses IP et mots de passe partagés pour la configuration SNMP

IV.1.6 Ajouter une carte sur Zabbix

Ajouter une carte sur Zabbix offre une représentation visuelle de la topologie réseau ou système surveillée, permettant ainsi de visualiser les éléments surveillés et leurs performances de manière graphique.

Pour ajouter une carte sur Zabbix, on suit les étapes illustrées dans les figures suivantes :

D'abord, on commence par cliquer sur l'onglet "***Surveillance***" dans Zabbix. Ensuite, on sélectionne l'option "***Cartes***" dans le menu. Pour créer une nouvelle carte, on clique sur le bouton "***Créer une carte***". On peut se référer à la figure IV.47 pour voir à quoi cela ressemble.

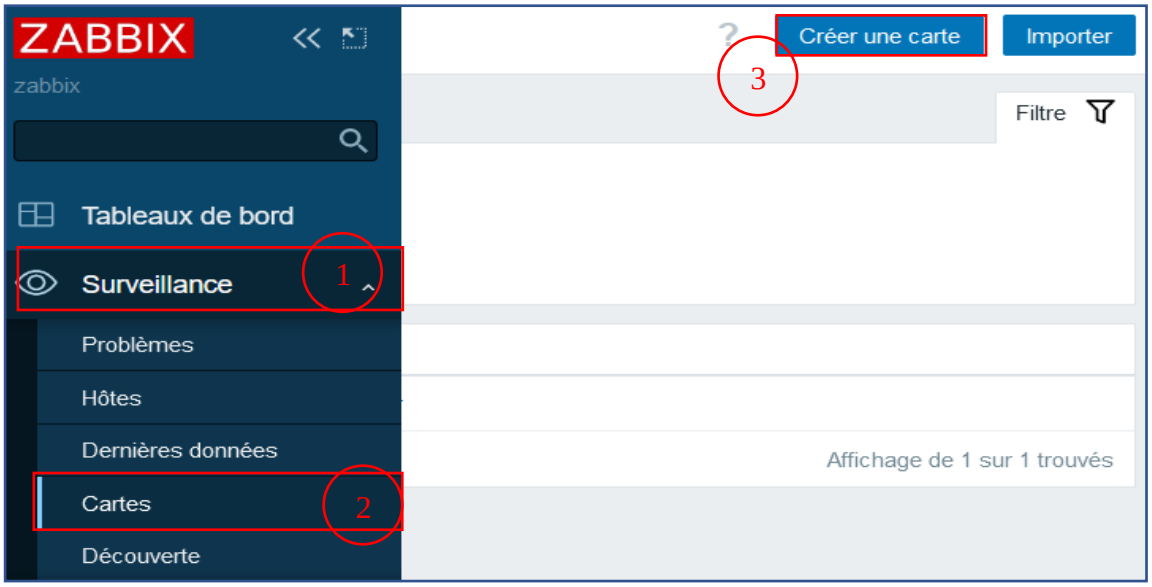


Figure IV.47 : Création d'une nouvelle carte

Ensuite, il faut remplir les informations de la carte telles que le Propriétaire, le Nom, la Largeur, etc. Une fois toutes les informations remplies, on clique sur le bouton "***Ajouter***" pour ajouter cette carte. Voir la figure IV.48.

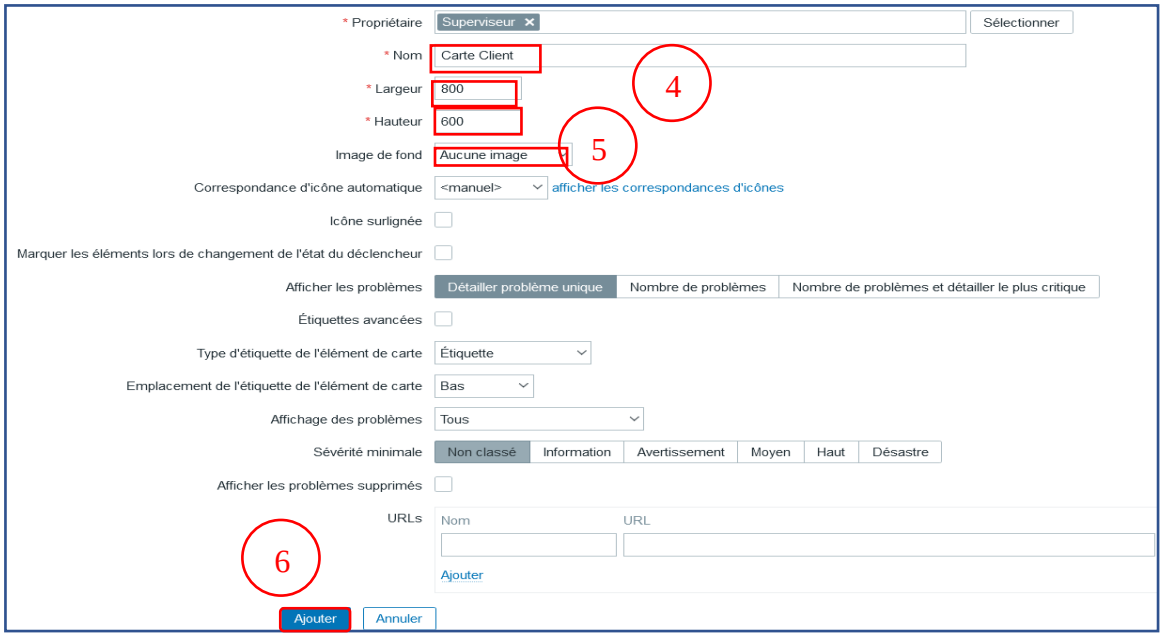


Figure IV.48 : Remplissage des informations de la carte sur Zabbix

Après avoir créé cette carte, on clique sur le bouton "***Ajouter***" pour ajouter un équipement à la carte. Ensuite, on remplit les différents éléments à l'intérieur tels que le Type, l'Étiquette, la Sélection de l'icône, etc. On peut se référer à la figure IV.49 pour visualiser cette étape. Le champ le plus important est le champ "***Hôte***", où il faut sélectionner l'équipement physique correspondant à ajouter à la carte. Une fois toutes les informations remplies, on clique sur le bouton "***Appliquer***" pour valider les modifications.

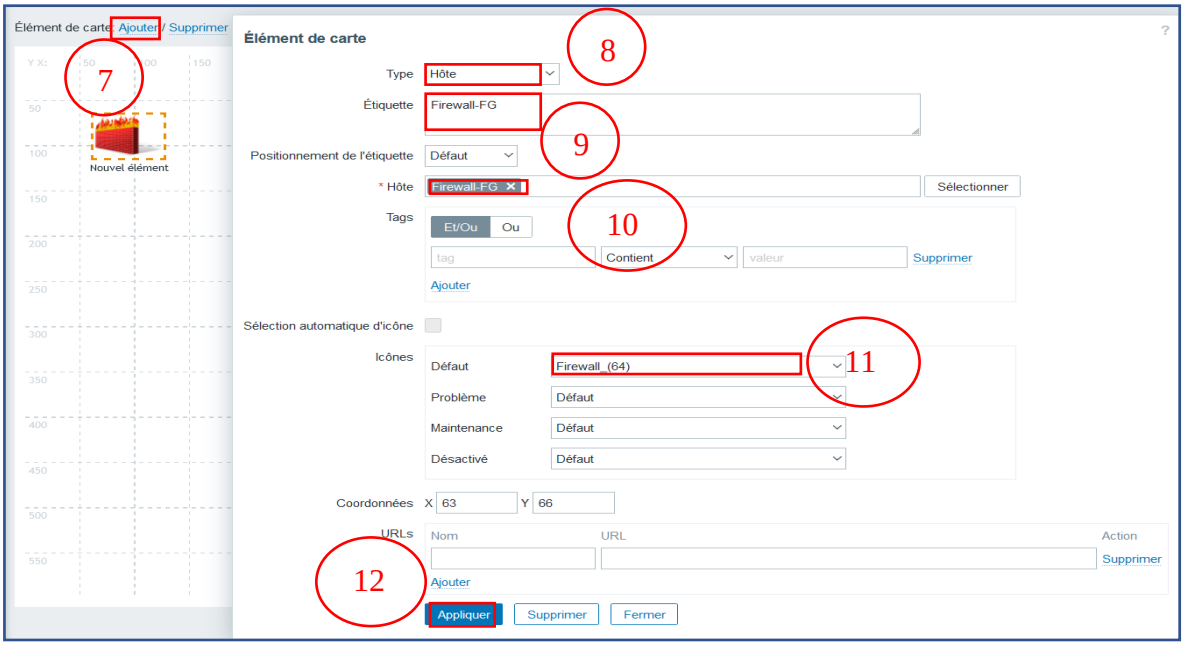


Figure IV.49 : Ajout d'un équipement à la carte et configuration des éléments

Le même principe est appliqué pour ajouter les autres équipements sur la carte. Il suffit de modifier l'hôte et l'icône de l'hôte pour obtenir le résultat illustré dans la figure IV.50.

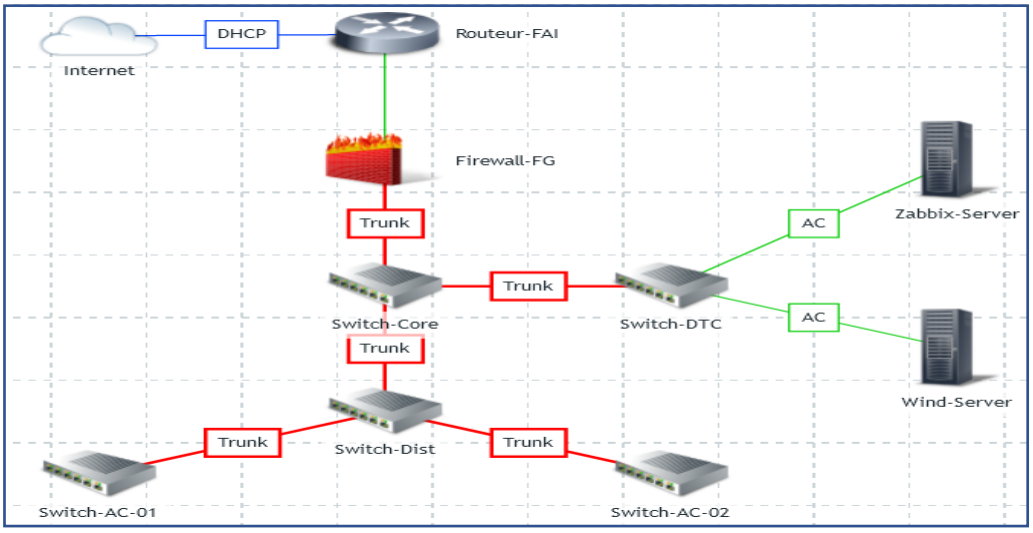


Figure IV.50 : Vue finale de la carte avec les équipements ajoutés

Maintenant, nous allons ajouter cette carte au tableau de bord Zabbix afin de visualiser les problèmes détectés par Zabbix.

Pour effectuer cette tâche, on se rend sur la barre de navigation de Zabbix, puis on clique sur "***Tableau de bord***". Ensuite, on clique sur "***Éditer le tableau de bord***". On peut se référer à la figure IV.51 pour visualiser cette étape.

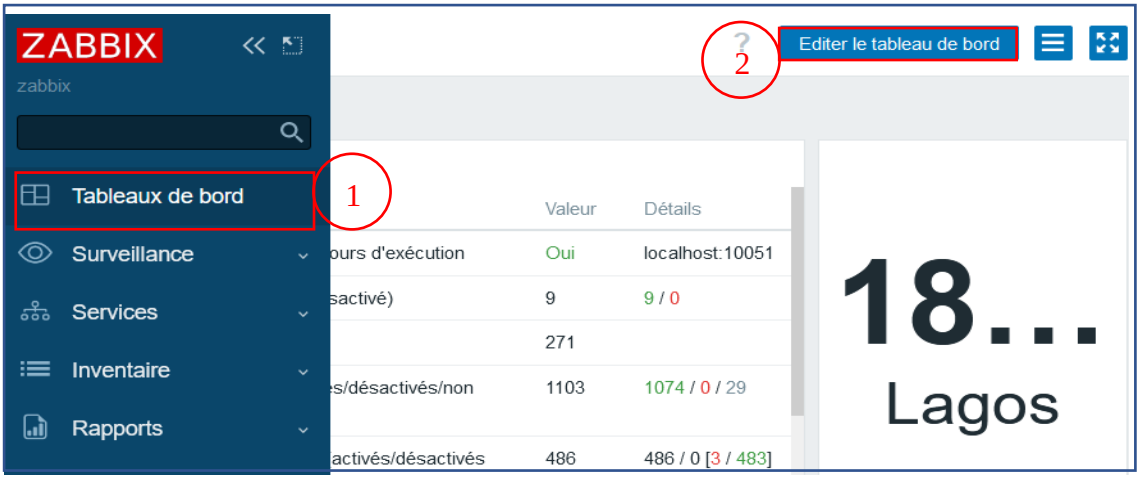


Figure IV.51 : Édition du tableau de bord Zabbix

Une fois que nous avons cliqué sur le bouton "***Éditer le tableau de bord***", nous cliquons sur un endroit vide du tableau de bord. Cela ouvrira la fenêtre "***Ajouter un widget***", comme illustré dans la Figure IV.52. Dans la section "***Type***", nous sélectionnons "***Carte***", puis nous cliquons sur "***Ajouter***" pour l'ajouter au tableau de bord.

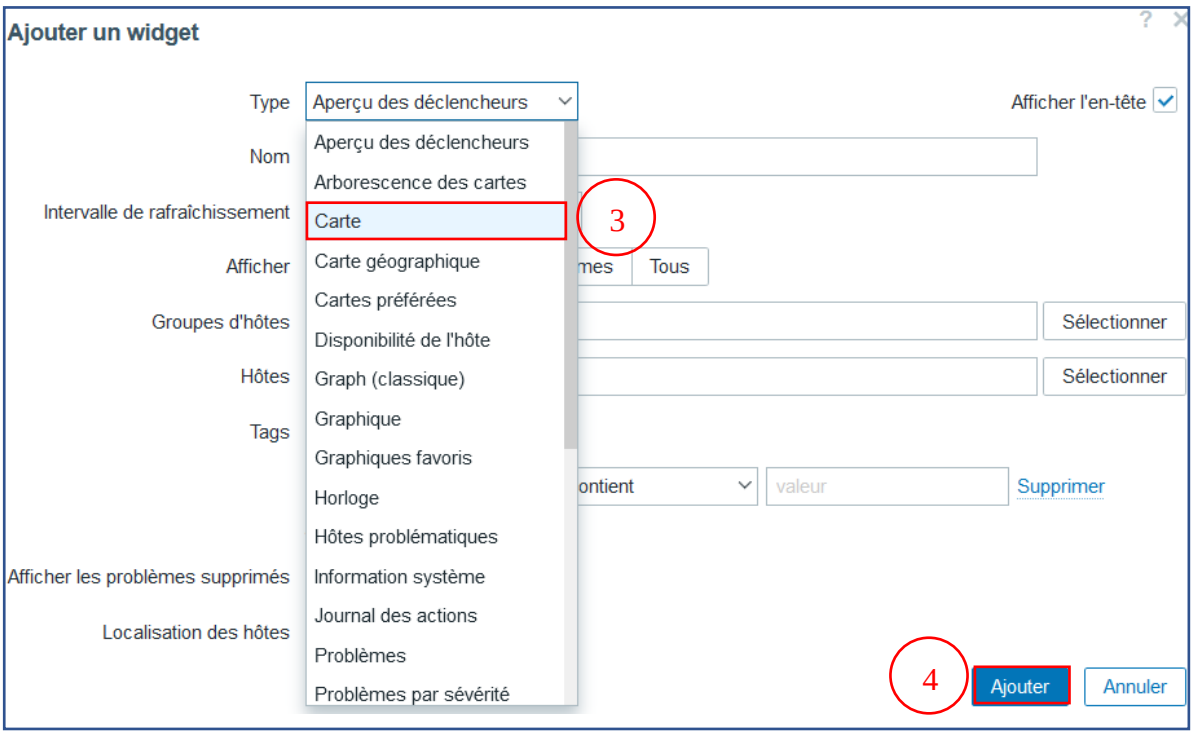


Figure IV.52 : Ajout d'un widget de carte sur le tableau de bord Zabbix

Dans la Figure IV.53, on peut observer que la carte a été ajoutée et qu'il n'y a aucun problème dans la topologie du réseau pour le moment. Cependant, dans les prochaines étapes, nous allons tester cette carte ainsi que d'autres fonctionnalités telles que l'envoi d'alertes par e-mail.

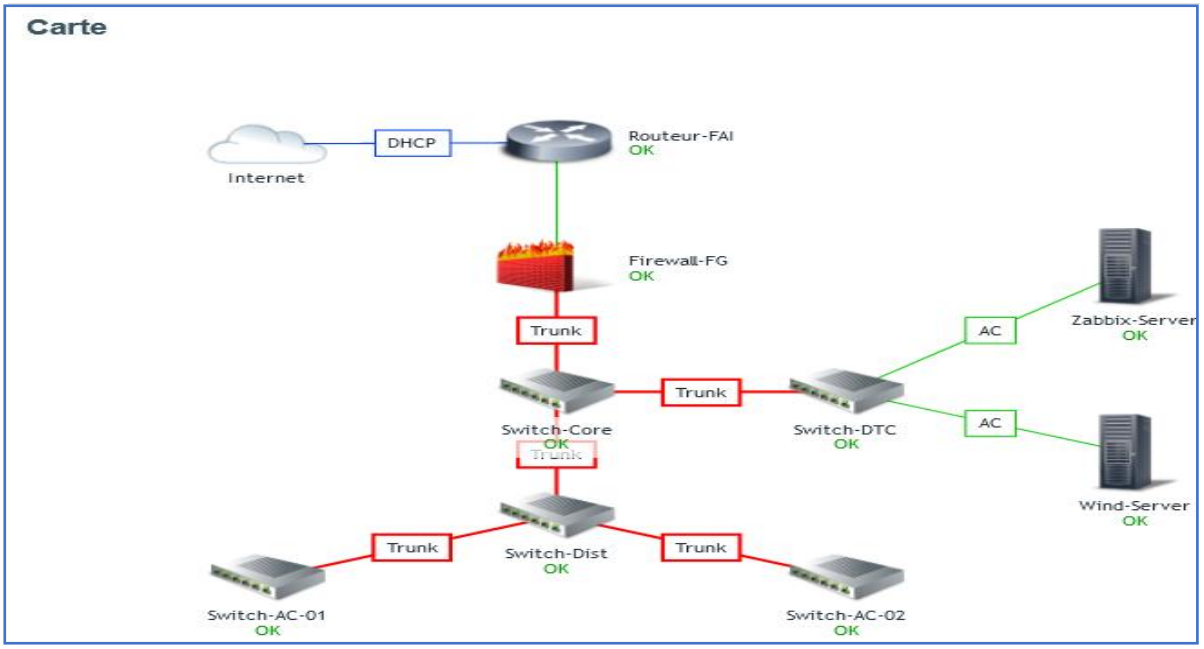


Figure IV.53 : Carte ajoutée sur le tableau de bord Zabbix

IV.1.7 Configuration des alertes Zabbix avec le service Gmail

La configuration des alertes Zabbix avec le service Gmail permet d'envoyer des notifications par e-mail via un compte Gmail lorsqu'il y a des événements importants, tels que des problèmes ou des pannes, dans une infrastructure surveillée. Grâce à cette configuration, les utilisateurs recevront des alertes immédiates par courrier électronique chaque fois qu'un incident critique se produit, leur permettant ainsi de prendre rapidement des mesures correctives. Cette intégration entre Zabbix et Gmail garantit une communication rapide et fiable des notifications, assurant ainsi une surveillance proactive de l'infrastructure.

1) Configuration de la fonctionnalité ‘’Validation en deux étapes ‘’ du compte Gmail

Nous allons configurer la fonctionnalité de "Validation en deux étapes" ou "Connexion en deux étapes" pour générer un code ou un mot de passe supplémentaire. Ce code sera utile lorsque nous configurerons des alertes dans Zabbix et que nous souhaiterons utiliser un compte Gmail pour recevoir les notifications par e-mail. Pour générer ce code en suit les étapes illustrées dans les figures suivantes

D'abord, on accède à notre compte Gmail et on se connecte. Ensuite, on clique sur notre avatar ou notre photo de profil dans le coin supérieur droit de la page, puis on sélectionne "***Gérer votre compte Google***". Dans le menu de gauche, on clique sur "***Sécurité***". On fait défiler vers le bas jusqu'à trouver la section "***Validation en deux étapes***" et on clique sur le bouton "***Activer la validation en deux étapes***". Voir la Figure IV.54.

On suit les instructions fournies pour configurer la connexion en deux étapes. Il se peut qu'on nous demande de vérifier notre identité en utilisant un code de vérification envoyé par SMS à notre téléphone ou via une application d'authentification.

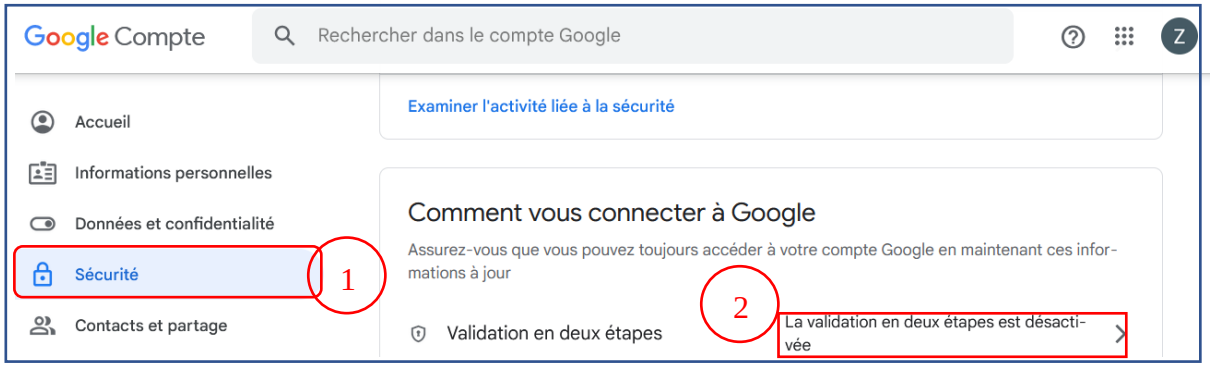


Figure IV.54 : Configuration de la fonctionnalité validation en deux étapes

Après avoir suivi les instructions pour activer la fonctionnalité "***Validation en deux étapes***" dans notre compte Gmail, on vérifie les paramètres de sécurité pour confirmer son activation. Voir la figure IV.55.

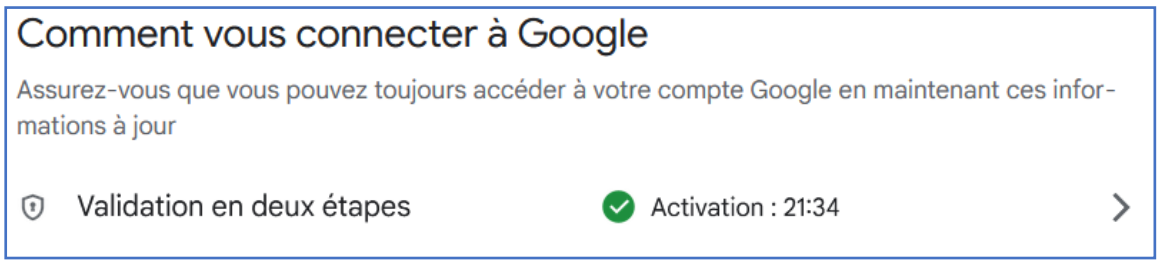


Figure IV.55 : Activation réussie de la Validation en deux étapes dans Gmail

Une fois la connexion en deux étapes activées dans notre compte Gmail, on doit générer le code d'authentification spécifique à l'application pour permettre à Zabbix d'accéder à notre compte de manière sécurisée. Pour générer ce code, on accède aux paramètres de sécurité de notre compte Gmail et on recherche la section "***Mots de passe des applications***". Ensuite, on clique sur le lien correspondant pour générer un mot de passe spécifique à l'application. Voir la figure IV.56.

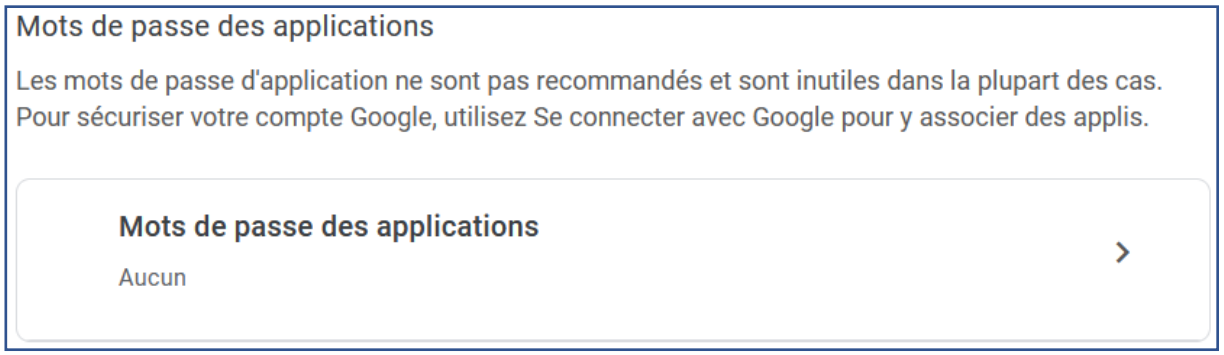


Figure IV.56 : Mot de passe des applications

Lors de la génération, on sélectionne "Autre (personnalisé)" comme type d'application et on lui donne un nom significatif comme "***Zabbix***" dans notre cas. Voir la figure IV.57

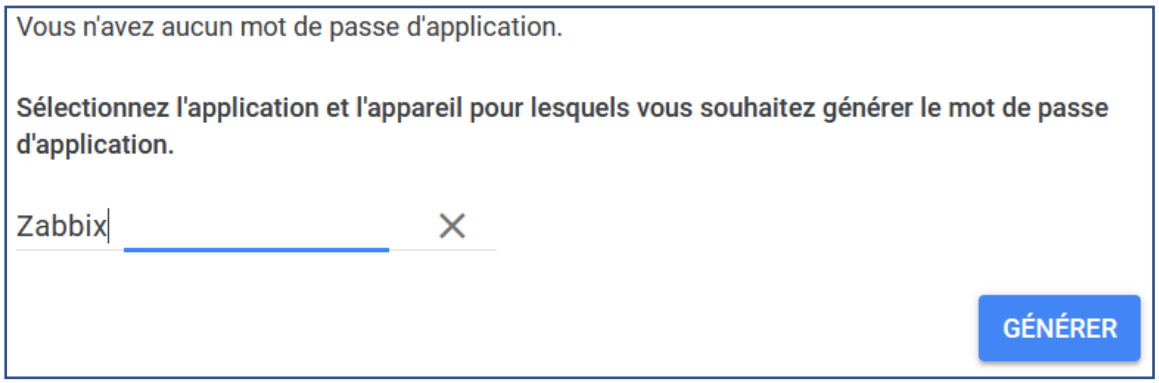


Figure IV.57 : Sélectionner l'application Zabbix

Une fois le mot de passe spécifique à l'application généré (voir la figure IV.58)**,** on le note soigneusement, car on en aura besoin lors de la configuration de Zabbix pour permettre l'envoi d'alertes via notre compte Gmail

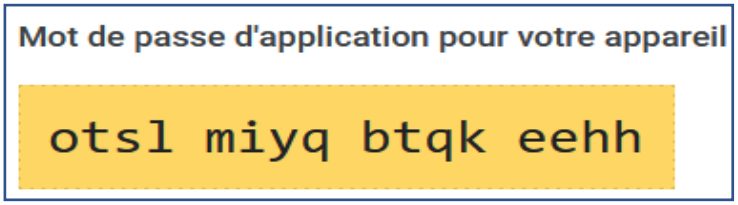


Figure IV.58 : Mot de passe généré

2) Installation et configuration du service SSMTP sur le serveur Debian

L'objectif de l'installation de SSMTP sur le serveur Debian est de configurer un serveur SMTP local qui agira comme un relais pour les e-mails sortants depuis le serveur Debian vers le serveur SMTP de Gmail. Voir la figure IV.59.

Figure IV.59 : Instalation du service SSMTP

Cette étape consiste à modifier le fichier de configuration de SSMTP afin que notre service SSMTP puisse se connecter à Gmail et envoyer des e-mails. Pour effectuer des modifications dans le fichier de configuration SSMTP, nous allons utiliser la commande illustrée dans la figure IV.60.

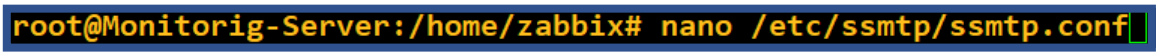


Figure IV.60 : Configuration du service SSMTP

Maintenant nous devons fournir des informations d'authentification, telles que le nom d'utilisateur (dans notre cas, "***zabbix.alert.send@gmail.com***") et le mot de passe généré par Gmail. De plus, nous devons spécifier le "***mailhub***" comme étant "***smtp.gmail.com***" et modifier le port pour le configurer sur "***465***". Voire la figure IV.61

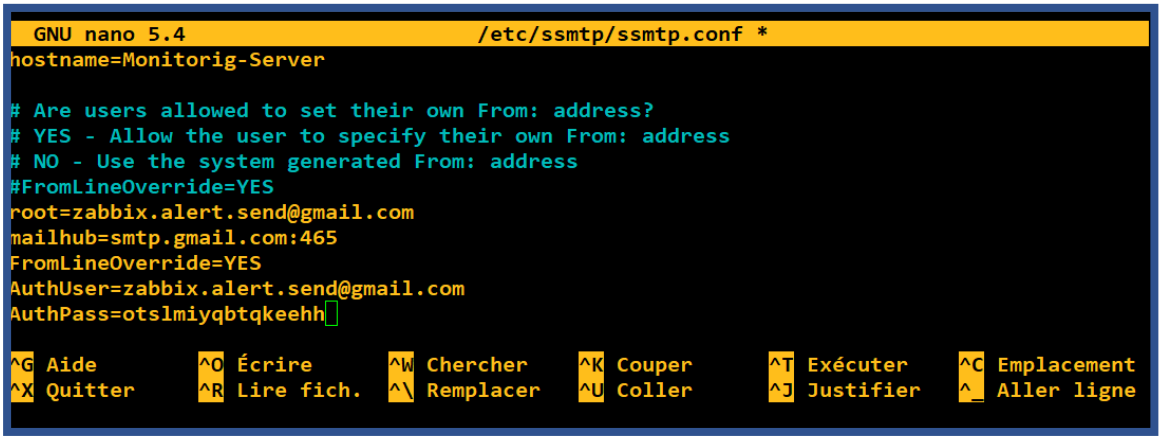


Figure IV.61 : modification du fichier de configuration SSMTP

Le "mailhub" représente l'adresse du serveur SMTP vers lequel SSMTP enverra les e-mails sortants. Quant au port, il s'agit du canal de communication utilisé pour établir la connexion avec le serveur SMTP. En configurant correctement ces paramètres dans le fichier de configuration de SSMTP, nous permettons à SSMTP d'établir une connexion sécurisée avec le serveur SMTP de Gmail et d'envoyer les e-mails avec succès.

Maintenant que nous avons configuré SSMTP sur le serveur Debian, nous allons effectuer un test en envoyant un e-mail via SSMTP. Pour cela, nous allons utiliser la commande illustrée dans la Figure B pour envoyer un message contenant le texte "Test". Cette commande permettra de vérifier si la configuration de SSMTP fonctionne correctement en envoyant un e-mail de test. Voir la figure IV.62.

```
root@Monitorig-Server:/home/zabbix# echo "Test" | ssmtp zabbix.alert.send@gmail.com
```

Figure IV.62 : Envoi d'un message de test avec SSMTP

Comme le montre la figure IV.63, le message a été envoyé avec succès. Cela confirme que la configuration de SSMTP sur le serveur Debian est fonctionnelle.

Figure IV.63 : Capture d'écran de confirmation d'envoi de l'e-mail via SSMTP.

3) Configuration et activation du service SMTP sur Zabbix

Dans cette étape, nous allons configurer Zabbix pour qu'il utilise le serveur SMTP local configuré via SSMTP. Cela permettra de faire transiter les e-mails sortants de Zabbix par le biais du serveur SMTP local, qui se chargera ensuite de les envoyer à Gmail. Le service SSMTP installé sur Debian agit en tant qu'interface pour l'envoi des e-mails vers un serveur SMTP distant.

Tout d'abord, nous cliquons sur l'onglet "Alertes" dans le menu principal de Zabbix. Ensuite, nous sélectionnons "Types de média" dans la section des alertes. Une fois dans la liste des types de média, nous repérons l'option "Gmail" et cliquons sur le bouton "Activer" correspondant. Cette étape permet d'activer le support de Gmail en tant que type de média pour les alertes par e-mail dans Zabbix. Voir la figure IV.64.

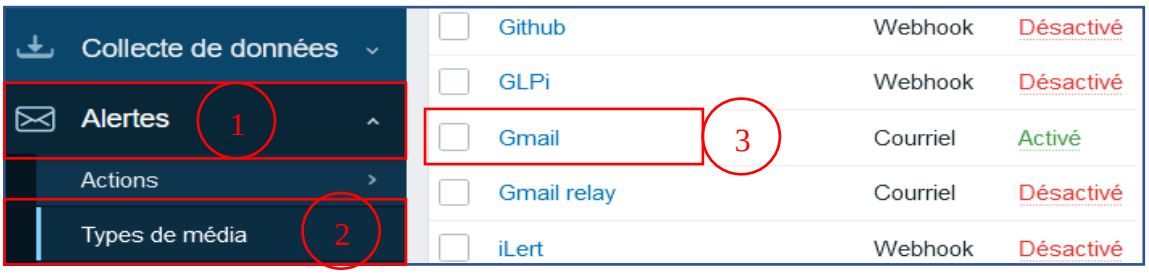


Figure IV.64 : Capture d'écran de l'activation du type de média Gmail dans Zabbix

Une fois que le type de média Gmail est activé, nous cliquons sur l'icône correspondante, ce qui nous permet d'accéder à la page de configuration. Sur cette page, nous remplissons les informations requises telles que le nom, le type, etc., et enfin pour finir nous allons cliquer sur Actualiser, comme illustré dans la figure IV.65.

Il est important de noter que les informations relatives au serveur SMTP doivent être identiques à celles que nous avons utilisées lors de la configuration du service SSMTP. Cela inclut le serveur SMTP, le port du service SMTP, le nom d'utilisateur, le mot de passe, et autres

informations nécessaires pour établir la connexion avec le serveur SMTP. En s'assurant que les informations sont cohérentes, Zabbix pourra utiliser le même serveur SMTP configuré par SSMTP pour envoyer les e-mails.

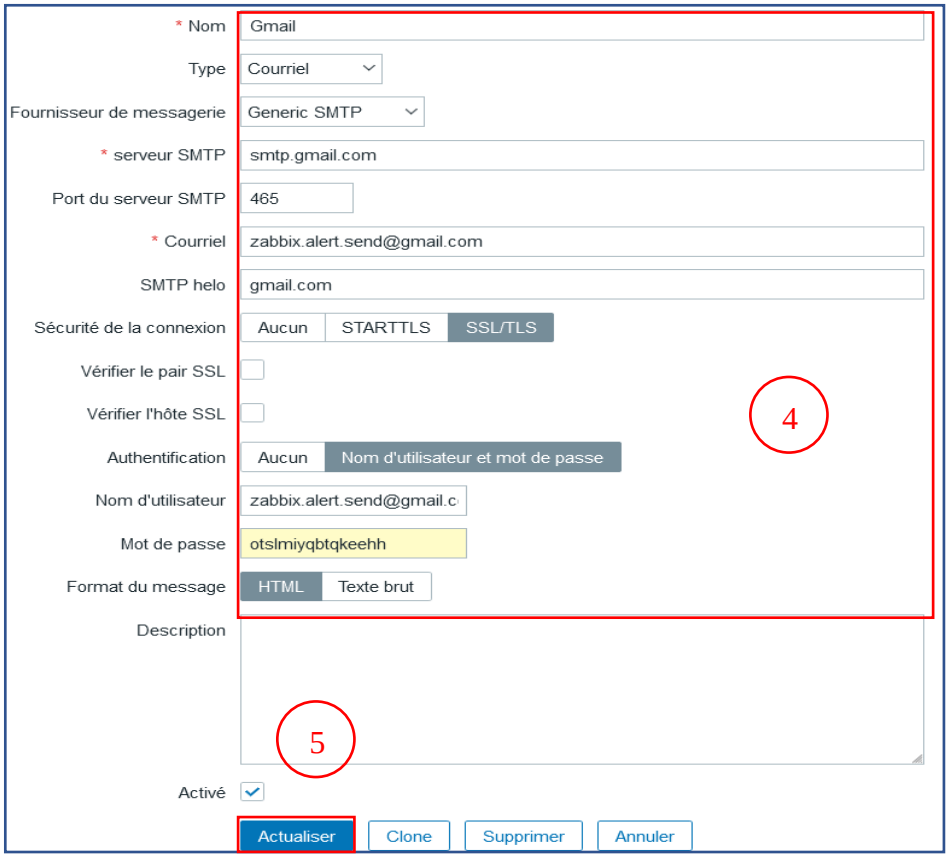


Figure IV.65 : configuration du type de média Gmail dans Zabbix

On aurait pu envisager de configurer directement le service SMTP sur Zabbix sans passer par SSMTP installé sur Debian. Cependant, dans certains cas, cette configuration directe peut rencontrer des problèmes de compatibilité ou de sécurité, ce qui peut empêcher l'envoi réussi des e-mails. C'est pourquoi nous avons utilisé SSMTP comme une solution alternative pour faciliter la communication entre Zabbix et le serveur SMTP distant, en assurant une configuration plus fiable et sécurisée.

❖ ***Effectuer un Test sur Zabbix :***

Pour tester le bon fonctionnement de l'envoi d'e-mails depuis Zabbix, nous procédons à une étape de test. Pour effectuer cette vérification, nous pouvons nous appuyer sur la figure IV.66 qui fournit une capture d'écran détaillant le processus de test d'envoi d'e-mails depuis Zabbix.

Il suffit de saisir l'adresse e-mail à laquelle nous souhaitons envoyer le test, puis de rédiger un message. Ensuite, il nous suffit de cliquer sur le bouton "***Test***" pour envoyer cet e-mail de test. Ce processus simple permet de vérifier rapidement et facilement la capacité d'envoi d'e-mails de Zabbix, en s'assurant que les notifications sont reçues avec succès.

Figure IV.66 : Capture d'écran du test d'envoi d'e-mail depuis Zabbix

Après avoir envoyé le courrier, on peut voir dans la figure IV.67 que le message a été réceptionné avec succès. Cela confirme l'efficacité de la configuration réalisée à cette étape, démontrant ainsi que le système est capable d'envoyer des alertes via des e-mails.

Figure IV.67 : Réception réussie du message envoyé depuis Zabbix"

4) Configuration du média au niveau du groupe utilisateur

Dans cette étape, nous allons configurer le média au niveau du groupe utilisateur dans le but de s'assurer que seuls les utilisateurs autorisés recevront les alertes. Nous allons définir les paramètres de notification spécifiques pour le groupe utilisateur, tels que l'adresse e-mail de destination, les seuils d'alerte, etc. Cette configuration permet de personnaliser les notifications en fonction des besoins de chaque groupe utilisateur, garantissant ainsi que seuls les utilisateurs autorisés recevront les alertes pertinentes.

4.1) Configuration du groupe d'utilisateurs pour la réception des alertes

Pour configurer les groupes d'utilisateurs afin qu'ils puissent recevoir les alertes, nous allons suivre les étapes présentées dans les figures suivantes :

D’abord nous commençons par accéder à la barre de navigation de Zabbix. Ensuite, nous cliquons sur l'option "***Utilisateurs***". Dans le menu déroulant qui apparaît, nous sélectionnons "***utilisateurs***". Une liste des utilisateurs disponibles s'affiche. Nous recherchons et cliquons sur le groupe spécifique auquel nous souhaitons apporter des modifications, dans notre cas c’est

"***Zabbix administrateur***". En cliquant sur ce groupe, nous accédons à la page de configuration du groupe d'utilisateurs. Voir la figure IV.68.

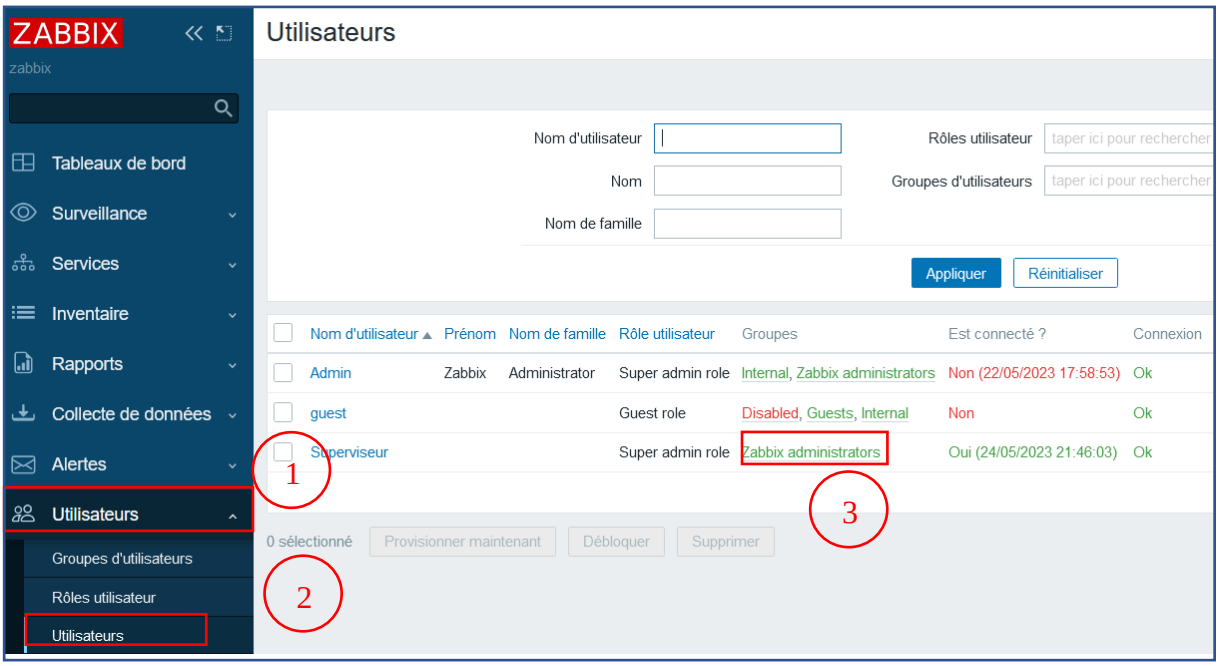


Figure IV.68 : Accéder au groupe Zabbix administrateurs

Une fois que nous accédons à la page de configuration du groupe d'utilisateurs, nous naviguons vers l'onglet "***Média***". Là, nous cliquons sur le bouton "***Ajouter***" pour ajouter un nouveau média. Cela nous redirige vers la page de configuration du média. Voir les Figures IV.69 et IV.70.

Figure IV.69 : Ajouter un Média

Dans cette étape, nous sélectionnons le type de média approprié pour notre configuration, à savoir "***Gmail***". Ensuite, nous saisissons les adresses e-mail du groupe "***Zabbix administrateur***" dans les champs correspondants. Nous cocherons également la case "Activer" pour activer ce média. Une fois que nous avons rempli toutes les informations nécessaires, nous cliquons sur le bouton "***Ajouter***" pour ajouter le média. Ensuite, nous pouvons cliquer sur le bouton "***Actualiser***" pour mettre à jour les paramètres. Les étapes détaillées sont illustrées dans les Figures IV.69 et IV.70.

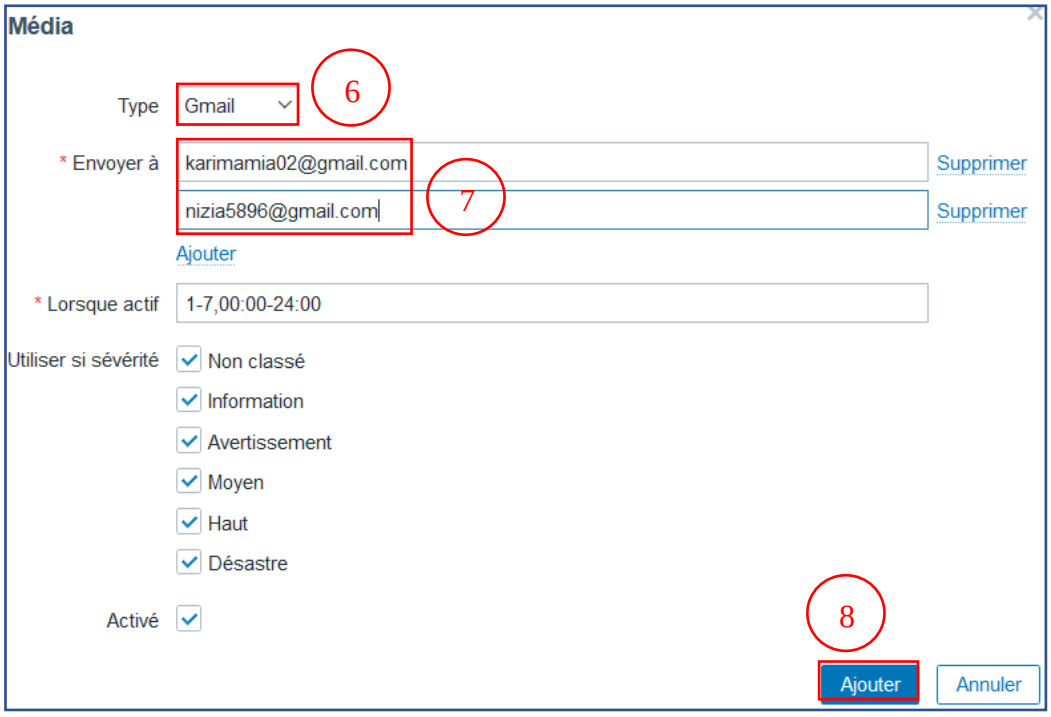


Figure IV.70 : Configuration du Média

Après avoir effectué la configuration du média pour le groupe d'utilisateurs "***Zabbix administrateur***", nous pouvons observer dans la figure IV.71 que le média a été correctement configuré et activé. Cela signifie que les utilisateurs appartenant à ce groupe seront en mesure de recevoir les alertes via ce média spécifié.

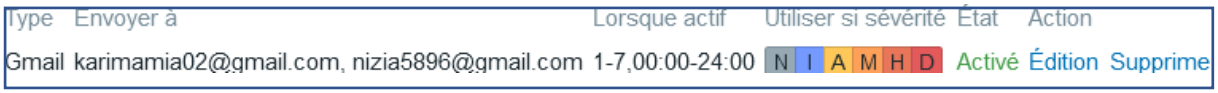


Figure IV.71 : Confirmation de l'activation du média pour le groupe Zabbix administrateur

4.2) Configuration des Actions de Déclencheur Pour l'envoi des alertes

Dans cette section, nous allons configurer les actions de déclencheur dans Zabbix afin d'envoyer des alertes. Les actions de déclencheur déterminent les actions à effectuer lorsque des déclencheurs sont activés comme l'utilisation du processeur élevée, Espace disque faible, etc... Pour faire ça dans Zabbix en suit les étapes illustrées dans les Figures suivants :

Dans la barre de navigation de Zabbix, nous cliquons sur "***Alertes***" puis sur "Actions". Ensuite, nous cliquons sur "***Actions de déclencheur***" et enfin sur le bouton "***Créer une action***". Voir la figure IV.72.

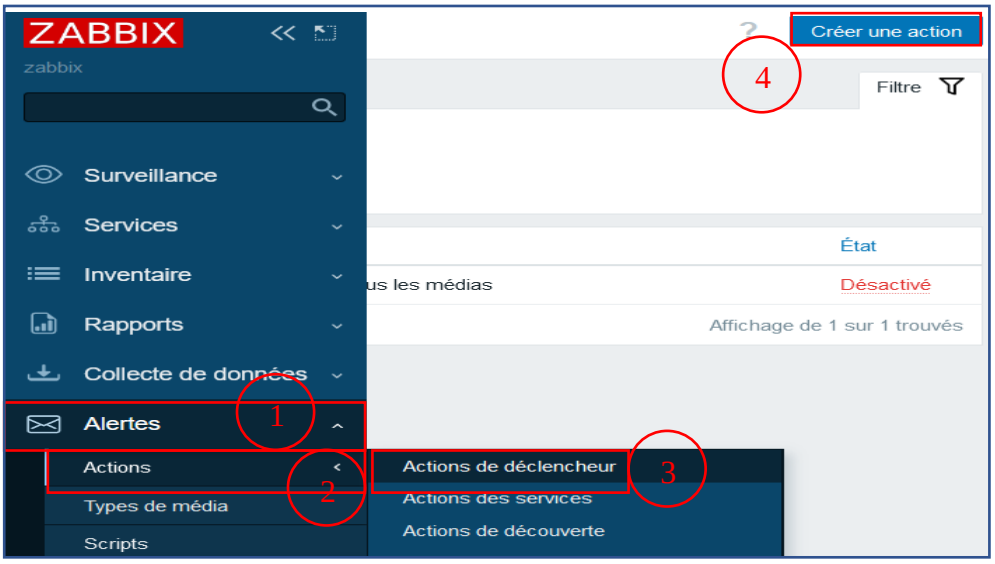


Figure IV.72 : Accéder au Actions de Déclencheur

Pour configurer cette action, nous commençons par spécifier un nom pour l'action. Ensuite, nous activons l'action en cochant la case correspondante. Pour ajouter une condition à l'action, nous cliquons sur le bouton "***Ajouter***". Voir la figure IV.73.

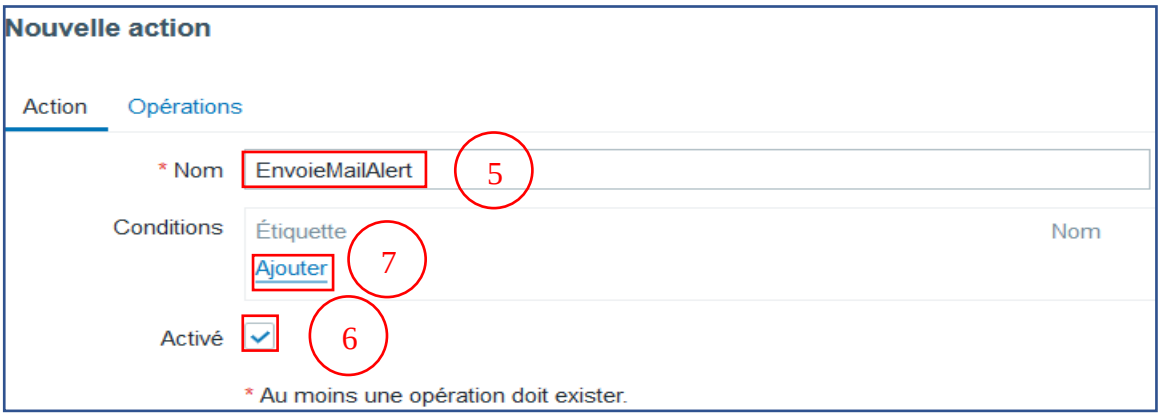


Figure IV.73 : Configuration d'une nouvelle action

Après avoir cliqué sur le bouton "***Ajouter***", une nouvelle fenêtre apparaît pour ajouter une nouvelle condition. Dans cette fenêtre, nous commençons par sélectionner le type de condition souhaité. Ensuite, nous choisissons l'opérateur "***égal***" dans la case correspondante. Pour la source du déclencheur, nous la configurons sur "***Hôte***". Enfin, nous sélectionnons le déclencheur en cliquant sur le bouton "***Sélectionner"***. Voir la figure IV.74.

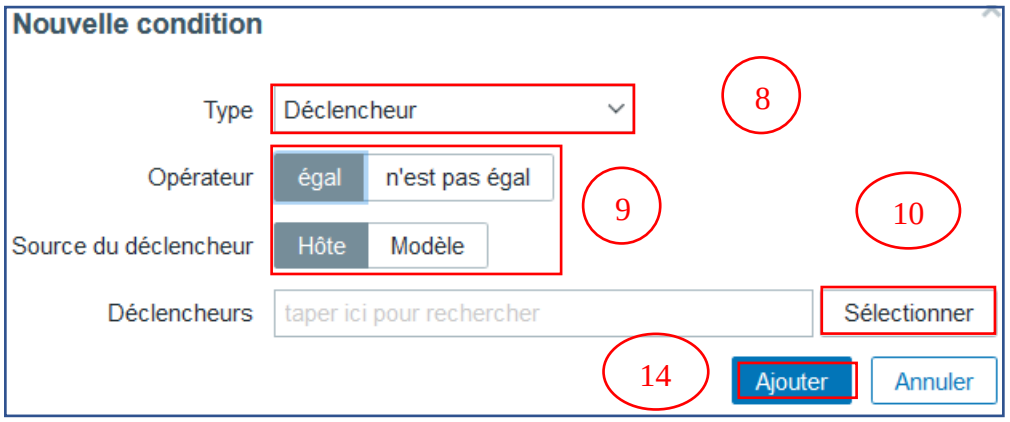


Figure IV.74 : configuration d'une nouvelle condition

Maintenant, nous allons sélectionner les déclencheurs. Tout d'abord, nous devons cliquer sur le bouton "***Sélectionner***" pour choisir un hôte. Dans notre cas, nous choisirons l'hôte "Firewall-FG". Ensuite, nous sélectionnons les déclencheurs spécifiques pour lesquels nous voulons recevoir une alerte en cliquant sur le bouton "***Sélectionner***". Voir la figure IV.75.

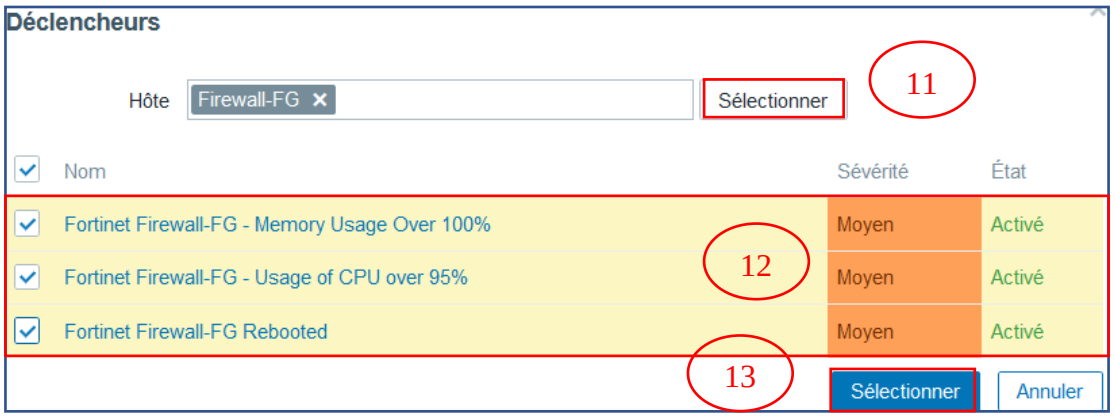


Figure IV.75 : Ajouter des déclencheurs

Après avoir ajouté les déclencheurs, nous cliquons sur le bouton "***Opération***" pour accéder à la section des opérations. Ensuite, nous cliquons sur le bouton "***Ajouter***" pour ajouter une nouvelle opération. Voire La figure IV.76.

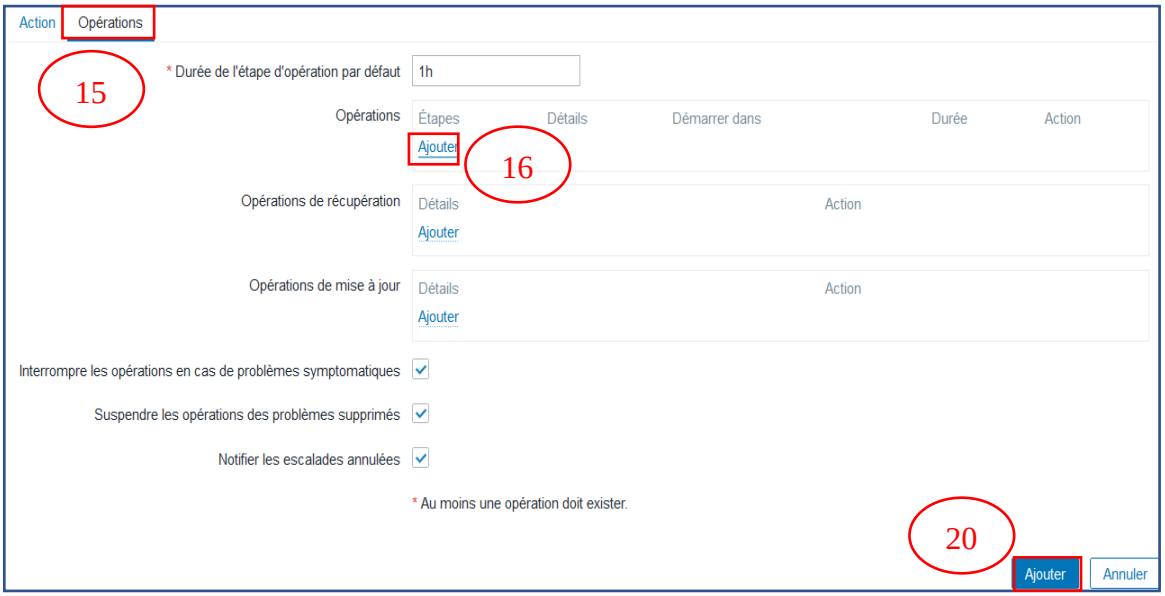


Figure IV.76 : Ajouter une opération

Après avoir cliqué sur le bouton "***Ajouter***", une nouvelle fenêtre intitulée "Détails de l'opération" s'affiche. Dans cette fenêtre, nous commençons par sélectionner le groupe auquel cette opération est destinée. Ensuite, nous sélectionnons les utilisateurs appartenant à ce groupe. Ensuite, nous sélectionnons l'application associée à cette opération, dans notre cas, il s'agit de Gmail. Enfin, nous cliquons sur le bouton "***Ajouter***". Voir la figure IV.77.

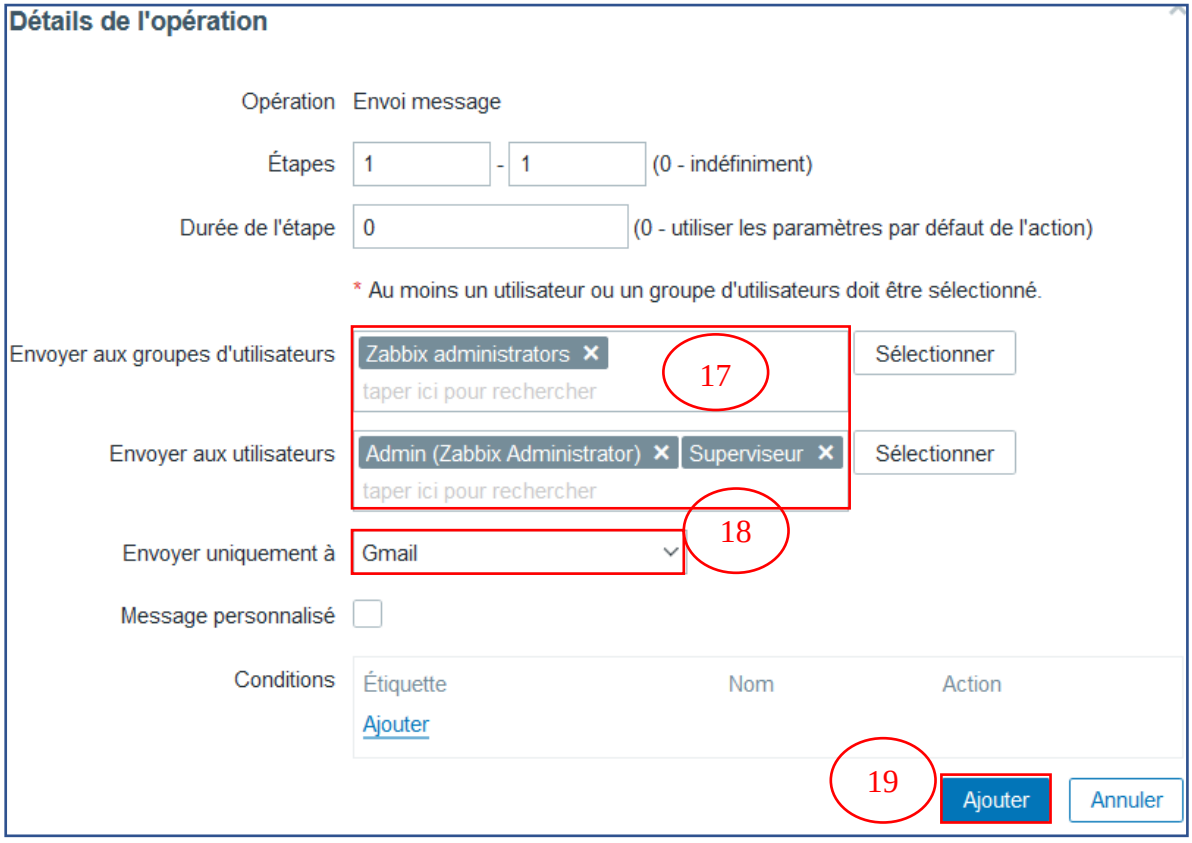


Figure IV.77 : Configuration des Détails de l'opération.

IV.1.8 Exécution des tests de surveillance avec Zabbix sur les hôtes ajoutés

Pour valider notre configuration et s'assurer du bon fonctionnement de Zabbix, nous allons effectuer une série de tests sur les hôtes que nous avons ajoutés. Ensuit nous allons consulter les résultats de ces tests à travers : le tableau de bord, la Carte et Gmail.

1) Tester Zabbix à travers le serveur Windows -Serveur

Pour tester Zabbix sur notre serveur Windows, nous utiliserons un logiciel appelé HeavyLoad. Ce logiciel nous permettra d'effectuer des tests de stress sur différents éléments tels que le CPU, le GPU, le stockage, etc. Cette étape nous permettra de vérifier la capacité de Zabbix à surveiller les performances et à détecter les éventuels problèmes sur notre serveur. La figure IV.78 montre l'exécution de HeavyLoad sur le serveur Windows, où le CPU, la RAM et le stockage sont mis sous stress maximum.

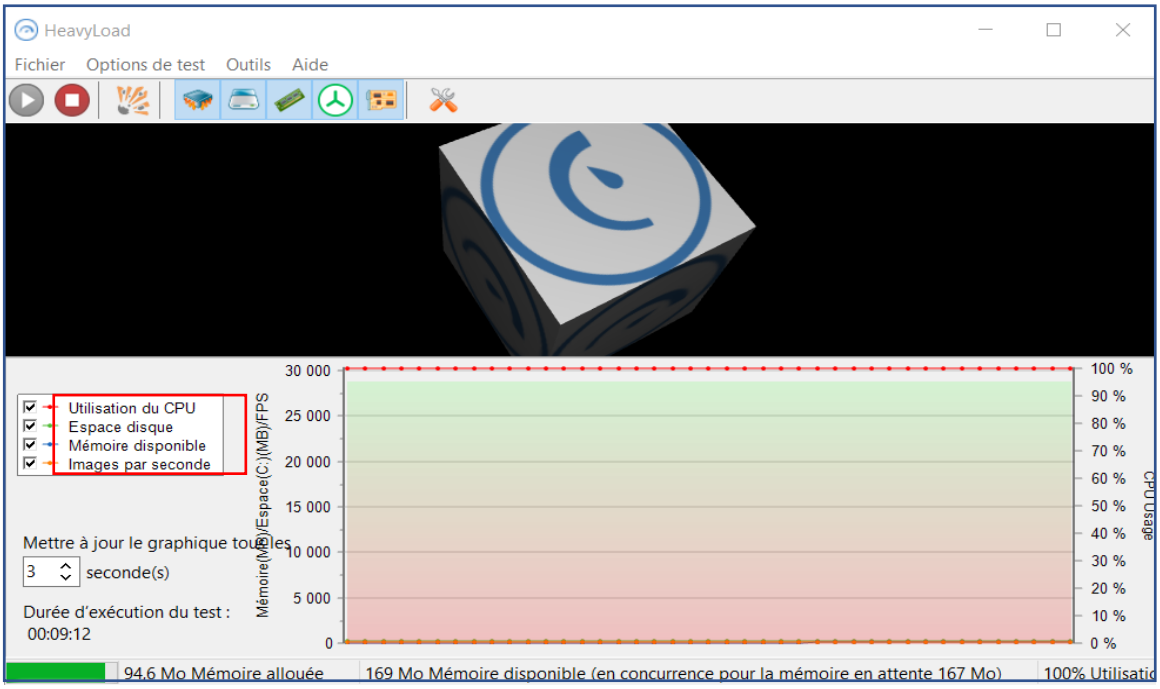


Figure IV.78 : Exécution du logiciel HeavyLoad

1.1) Résultat du test CPU, RAM, Espace Disque

Après avoir exécuté le test pendant environ 5 minutes, Zabbix a détecté plusieurs problèmes sur le serveur. Plus précisément, il a identifié trois problèmes : une utilisation élevée du CPU, un espace disque insuffisant et une utilisation élevée de la mémoire physique.

Sur le tableau de bord de Zabbix, trois alertes distinctes ont été affichées, indiquant que les seuils de 90% ont été dépassés pour ces trois ressources. De plus, Zabbix a envoyé des notifications par e-mail aux administrateurs responsables (Karim et Madina) pour les informer de ces alertes.

Ces problèmes ont également été signalés sur la carte de surveillance, offrant une visualisation claire des problèmes rencontrés.

❖ ***Les alertes affichés sur le tableau de bord***

Alerte d'utilisation CPU, mémoire physique et Espace disque affiché sur tableau de bord. Voire la figure IV.79.

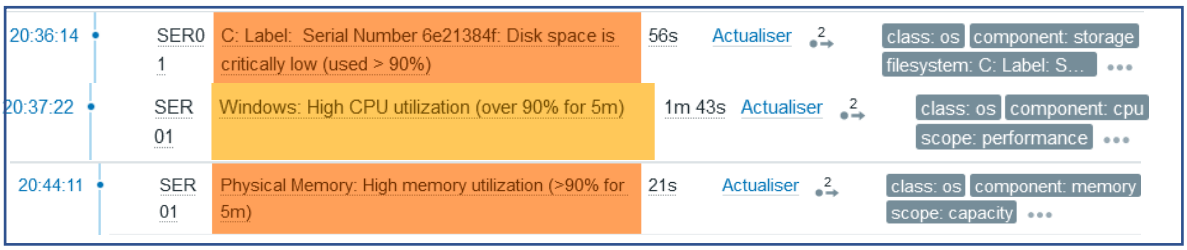


Figure IV.79 : Les alertes affiché sur tableau de bord

❖ ***Les alertes reçus par E-mail (notification)***

Alerte d'utilisation CPU, mémoire physique et Espace disque envoyées par e-mail. Voir la figure IV.80.

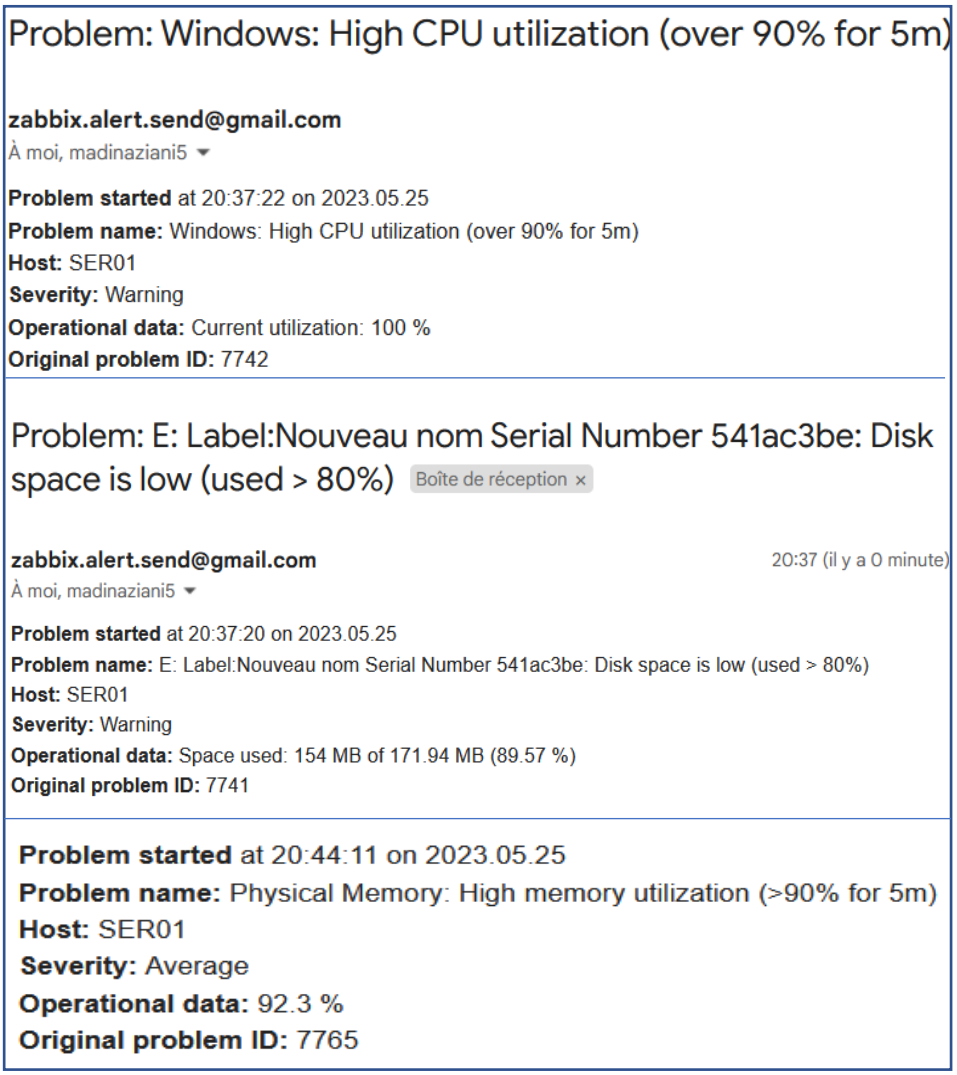


Figure IV.80 : Les notifications envoyées par e-mail

❖ ***Alertes affichés sur carte de surveillance***

Alerte d'utilisation CPU, mémoire physique et Espace disque affiché sur carte de surveillance. Voire la figure IV.81.

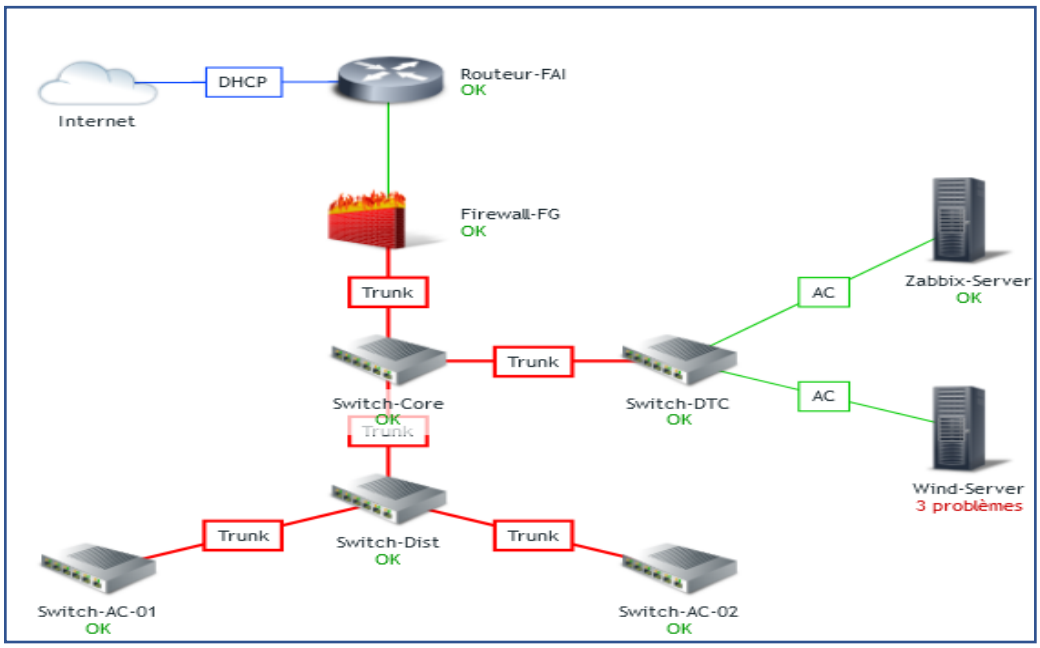


Figure IV.81 : L'affichage des problèmes sur la carte de surveillance.

2) Tester Zabbix à travers le Firewall FortiGate

Dans le cas du pare-feu FortiGate, nous allons effectuer un test de redémarrage afin de vérifier si Zabbix peut différencier un redémarrage du pare-feu d'une déconnexion du câble ou d'une autre erreur. Pour redémarrer le pare-feu, nous exécutons une commande spécifique, comme indiqué dans la figure IV.82.

```
FG # execute reboot
This operation will reboot the system !
Do you want to continue? (y/n)y


System is rebooting...
```

Figure IV.82 : redémarrage du pare-feu

2.1) Résultat du test de redémarrage du pare-feu

Après avoir redémarré le pare-feu, Zabbix a rapidement détecté le redémarrage et affiché l'alerte correspondante sur le tableau de bord. De plus, Zabbix a envoyé une notification par e-mail pour informer les administrateurs du redémarrage du pare-feu. Cette alerte a également été signalée sur la carte de surveillance, où le redémarrage est clairement indiqué. Les captures

d'écran des figures IV.83, IV.84 et IV.85 illustrent respectivement l'alerte sur le tableau de bord, la notification envoyée par e-mail et l'affichage du redémarrage sur la carte de surveillance.

❖ ***Alerte affichée sur le tableau de bord***

Figure IV.83 : Alerte affiché sur tableau de bord

❖ ***Les alertes reçus par E-mail (notification)***

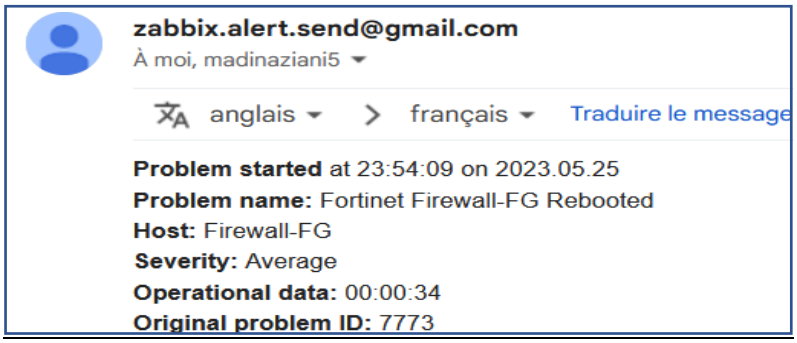


Figure IV.84 : Alerte envoyées par e-mail

❖ ***Alertes affichés sur carte de surveillance***

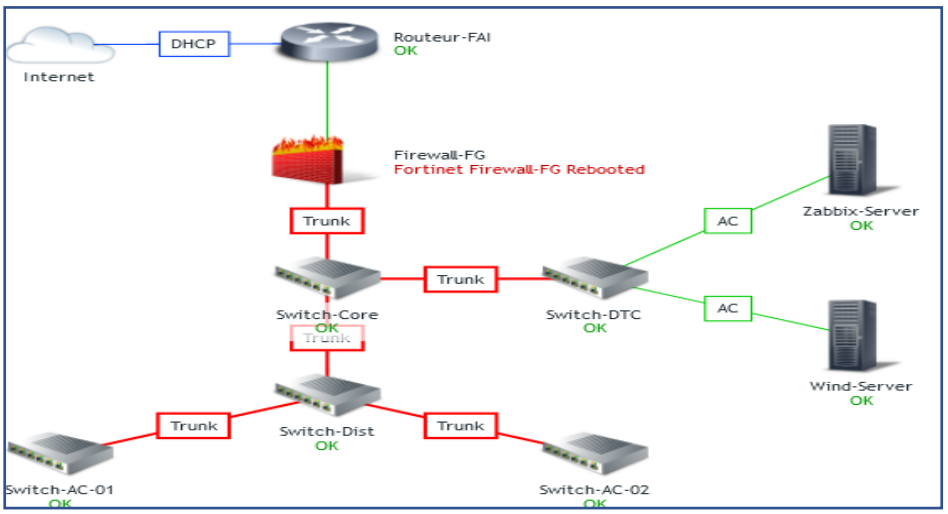


Figure IV.85 : Alerte affiché sur la carte de surveillance

3) Test d'un équipement Cisco sur Zabbix

Dans le cadre du test des équipements Cisco, nous allons effectuer des manipulations spécifiques sur un routeur (R-FAI) et un commutateur (Sw-AC01). Pour le routeur, nous allons modifier le mode du port eth0/0 en le passant en mode Half-Duplex. En ce qui concerne le commutateur, nous allons commencer par étendre l'interface eth0/1, puis nous allons couper la

liaison entre ce commutateur et le commutateur de distribution (Dist). Les captures d'écran des Figures IV.86, IV.87 et IV.88 illustrent respectivement les manipulations effectuées sur le routeur et le commutateur.

- Les commandes effectuées sur le routeur. Voire la figure IV.86

```
R-FAI(config)#int eth0/0
R-FAI(config-if)#duplex half
R-FAI(config-if)#end
```

Figure IV.86 : Modification de l'interface du routeur.

- Les commandes effectuées sur le commutateur. Voire la figure IV.87

```
SW-AC01(config)#int eth0/1
SW-AC01(config-if)#shut
SW-AC01(config-if)#end
```

Figure IV.87 : Etendre l'interface eth0/1 du commutateur.

- Couper la liaison entre le commutateur Dist et le commutateur AC-01. Voire la figure IV.88

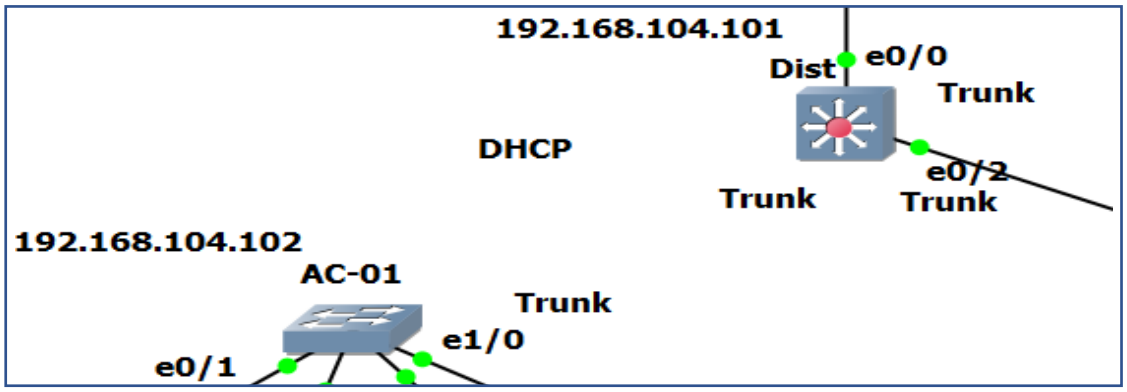


Figure IV.88 : coupure de la liaison entre les commutateurs Dist et AC-01

3.1) Les résultats de test sur les équipements Cisco

Après avoir effectué les manipulations sur le routeur (R-FAI) et le commutateur (Sw-AC01), Zabbix a détecté les problèmes correspondants et a affiché les alertes sur le tableau de bord, envoyé des notifications par e-mail et affiché les problèmes sur la carte de surveillance.

Pour le routeur (R-FAI), Zabbix a détecté la modification du mode du port eth0/0 en mode Half-Duplex. Cette alerte a été affichée sur le tableau de bord et a été notifiée aux administrateurs par e-mail. Quant au commutateur (Sw-AC01), Zabbix a détecté deux problèmes distincts. Tout d'abord, l'extension de l'interface eth0/1 a été détectée, ce qui a généré une alerte affichée sur le tableau de bord et notifiée par e-mail. Ensuite, la coupure de la liaison entre le commutateur et le commutateur de distribution (Dist) a également été détectée par

Zabbix, générant une autre alerte sur le tableau de bord et une notification par e-mail aux administrateurs.

Les captures d'écran des Figures IV.89, IV.90 et IV.91 montrent respectivement les alertes sur le tableau de bord, les notifications envoyées par e-mail et l'affichage des problèmes sur la carte de surveillance.

❖ ***Alerte affichée sur le tableau de bord***

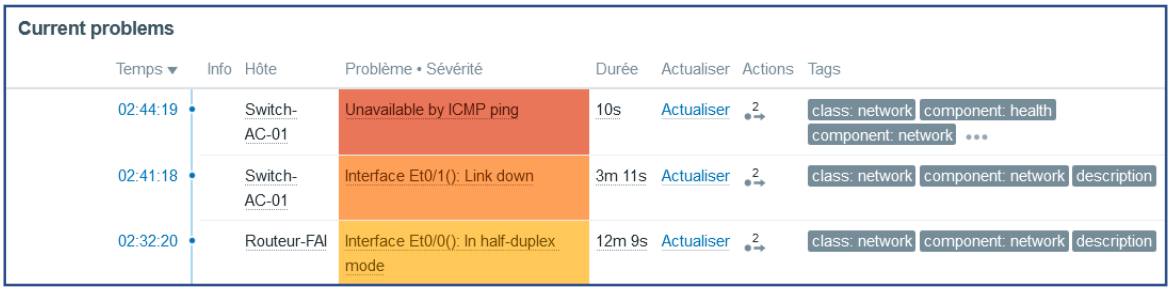


Figure IV.89 : Alertes des équipements Cisco affiché sur tableau de bord

❖ ***Alertes affichés sur carte de surveillance***

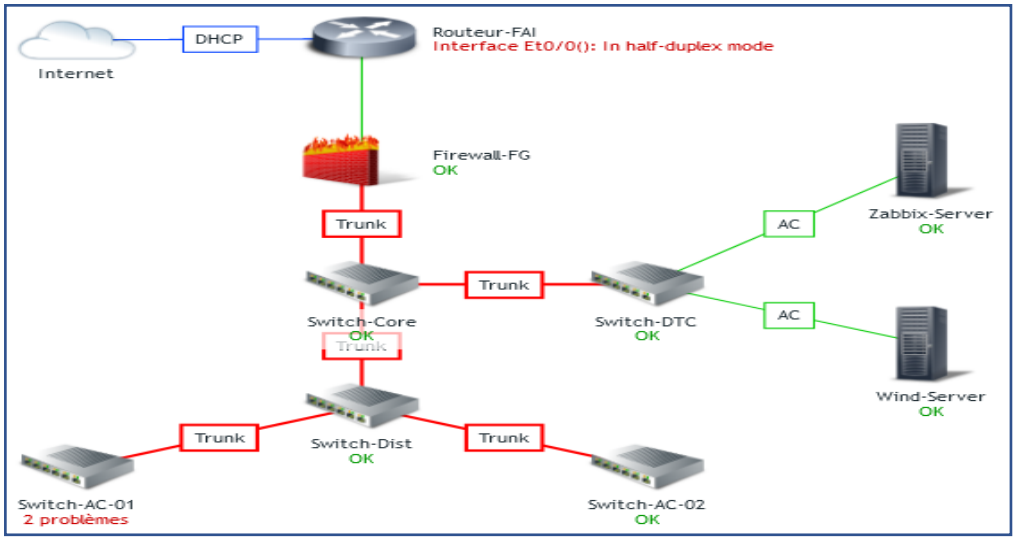


Figure IV.90 : Alertes des équipements Cisco sur la carte de surveillance

❖ ***Les alertes reçus par E-mail (notification)***

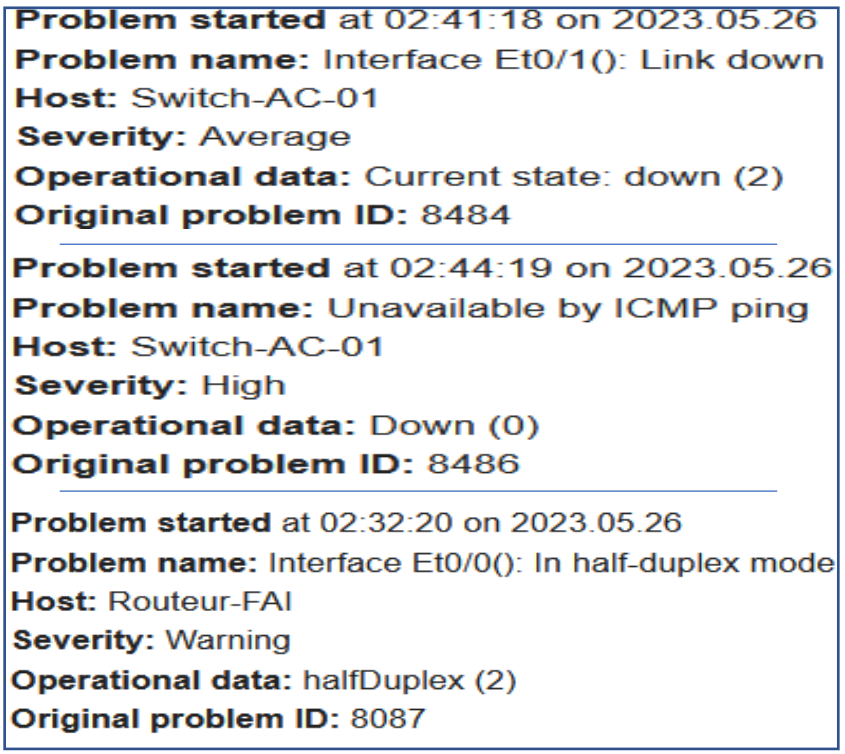


Figure IV.91 : Alertes des équipements Cisco envoyées par e-mail

Conclusion

Dans l'ensemble, les objectifs fixés en début de ce chapitre ont été atteints en démontrant qu'Ansible peut automatiser la configuration, l'installation des applications et la gestion du réseau à l'aide des playbooks injectés via SSH. Par conséquent, on peut affirmer qu'Ansible est une excellente solution d'automatisation de l'infrastructure.

De plus, grâce aux alertes affichées sur le tableau de bord, aux notifications envoyées par e-mail et à l'affichage des problèmes sur la carte de surveillance, les administrateurs Karim et Madina ont pu être informés en temps réel des incidents et prendre les mesures nécessaires pour les résoudre.

Ces fonctionnalités de Zabbix ont amélioré la réactivité et la gestion des problèmes, garantissant ainsi une meilleure disponibilité et performance du réseau. En conclusion, Zabbix se positionne comme une solution fiable et puissante pour la surveillance et la gestion des systèmes.

Conclusion Générale

Conclusion générale

En conclusion, notre mémoire met en évidence l'importance de l'automatisation et de la supervision dans la gestion des réseaux informatiques. Nous avons examiné le réseau de l'entreprise d'accueil lors de notre stage et identifié les problèmes spécifiques rencontrés dans la gestion, la configuration et la surveillance de leur réseau.

Pour remédier à ces problèmes, nous avons mis en place une solution complète en utilisant Ansible pour automatiser les processus réseau et Zabbix pour superviser son bon fonctionnement. Cette solution repose sur un serveur d'automatisation et un serveur de supervision. Nous avons décrit les différentes étapes de déploiement de notre solution et évalué son efficacité à l'aide de simulations.

Grâce à l'adoption de notre solution, nous avons constaté une amélioration de l'optimisation et de la configuration du réseau, ce qui a réduit les risques d'erreurs humaines et minimisé les temps d'arrêt non planifiés, renforçant ainsi la fiabilité des systèmes. De plus, la supervision constante a permis une détection rapide des pannes et une réaction immédiate, minimisant ainsi l'impact sur les activités et maintenant la satisfaction des clients. Cette solution a apporté des avantages significatifs en améliorant globalement les performances du réseau et en assurant une continuité opérationnelle plus efficace.

Enfin, notre travail souligne l'importance cruciale de l'automatisation et de la supervision dans la gestion des réseaux informatiques, en particulier dans un environnement où les systèmes sont devenus essentiels et complexes. Nous espérons que notre solution pourra servir de référence pour d'autres projets similaires et encourager les entreprises à explorer les avantages offerts par l'automatisation et la supervision pour améliorer leurs opérations informatiques.

Webographie

[2] Cisco, «What is network automation?,» 2023. [En ligne]. Available: https://www.cisco.com/c/en/us/solutions/automation/network-automation.html. [Accès le 14 Avril 2023].

[3] Terry Slattery, «How to build a network automation architecture in 5 phases,» [En ligne]. Available: https://www.techtarget.com/searchnetworking/tip/How-to-build-a-network-automation-architecture-in-5-phases. [Accès le 14 Avril 2023].

[4] «Git --fast-version-control,» [En ligne]. Available: https://git-scm.com/. [Accès le 14 Avril 2023].

[6] Red Hat, «Comment fonctionne l'automatisation des réseaux ?,» Red Hat, [En ligne]. Available: https://www.redhat.com/fr/topics/automation/what-is-network-automation. [Accès le 14 Avril 2023].

[8] Marc. Pare, «Pourquoi choisir Ansible pour l'automatisation,» 26 Juin 2018. [En ligne]. Available: https://www.cloudops.com/fr/blog/pourquoi-choisir-ansible-pour-lautomatisation/. [Accès le 5 Avril 2023].

[12] ManageEngine, «Qu'est-ce qu'un SNMP ?,» [En ligne]. Available: https://www.manageengine.com/fr/network-monitoring/what-is-snmp.html. [Accès le 15 Avril 2023].

[14] «Différence entre SNMP v1 et v2,» 27 décembre 2017. [En ligne]. Available: https://waytolearnx.com/2017/12/difference-entre-snmp-v1-et-v2.html. [Accès le 20 Avril 2023].

[17] Clever. Technologies, «Les protocoles de supervision réseaux,» [En ligne]. Available: https://supervision-clever.fr/monitoring-protocoles-reseaux/. [Accès le 15 Avril 2023].

Bibliographie

[1] Edelman Jason, Scott S.Lowe et Matt Oswalt, Network Programmability And Automation_ Skills For The Next-Generation Network Engineer, 1 éd., Gravenstein Highway North, Sebastopol, CA 95472.: O'Reilly Media, Inc, 2018. [Accès le 14 Avril 2023].

[5] DIF SAIDA &. BAKHTI FADILA,Study on Python for Network Automation, Ziane Achour university of Djelfa, 2018/2019.

[7] OUIZI Sonia & HAMMOU Ines, *Projet de Fin d'Etudes Automatisation Avec Ansible Cas : Entreprise RTC Sonatrach Bejaia*, Bejaia: Université Abderrahmane Mira, 2021/2022.

[9] Briche Thierry et Voland Matthieu , *Les outils d'administration et de supervision réseau L'exemple de Nagios,* Décembre 2004.

[10] Belkadi Mourad et Bouchata Zakaria, *La mise en place d'un système de supervision réseau (Cas Pratique Univ BLIDA), Blida :* Université SAAD DAHLAB, 2015/2016.

[11] Zakaria Abdelmoiz DAHI, *Support de cours pour Master en Sciences et Technologies de l'information et de la Communication, « Interconnexion et Gestion de Réseaux », Université Constantine 2 – Abdelhamid Mehri.*

[13] O. B. HONVOH, *Conception d'une plate-forme pour la supervision des réseaux informatiques,* 2008-2009.

[15] Abir Trabelsi, *Mastère Professionnel en Nouvelles Technologies des Télécommunications et Réseaux (N2TR) : Mise en place d'un outil de supervision système et réseau open source, Université Virtuelle de Tunis* ,2014/2015.

[16] M. Souleymane MARENA, *Mémoire de fin d'études Sécurité et Supervision de réseaux,* Ziguinchor : Université Assane SECK, 2020/2021.