# TABLE DES MATIERES

**REMERCIEMENTS** ..... **i**

**TABLE DES MATIERES** ..... **ii**

**LISTE DES ABREVIATIONS** ..... **vi**

**INTRODUCTION GENERALE** ..... **1**

**CHAPITRE 1 GNERALITES SUR LES RESEAUX INFORMATIQUES** ..... **2**

***1.1 Introduction*** ..... ***2***

***1.2 Architecture physique*** ..... ***2***

**1.2.1 Les catégories de réseau** ..... **2**

**1.2.2 Les composants matériels** ..... **3**

**1.2.3 La topologie physique** ..... **7**

***1.3 Architecture logique*** ..... ***8***

**1.3.1 Les modèles OSI et TCP/IP** ..... **8**

**1.3.2 Le modèle hiérarchique en trois couches** ..... **10**

**1.3.3 La topologie logique d' un réseau et la technologie Ethernet** ..... **12**

**1.3.4 Les typologies des réseaux** ..... **12**

***1.4 Le cloisonnement des réseaux*** ..... ***16***

***1.5 Conclusion*** ..... ***17***

**CHAPITRE 2 ETUDES ET CONCEPTION D'UN SERVEUR DE MONITORING** ..... **19**

***2.1 Introduction*** ..... ***19***

***2.2 Le protocole SNMP*** ..... ***19***

**2.2.1 Les composants de l'environnement SNMP** ..... **20**

**2.2.2 Le message SNMP** ..... **21**

**2.2.3 Les versions du protocole SNMP** ..... **23**

***2.3 Etude de la plateforme de monitoring*** ..... ***24***

***2.4 Les prérequis avant élaboration*** ..... ***25***

**2.4.1 Un noyau Linux** ..... **25**

**2.4.2 Une connexion à internet** ..... **25**

**2.4.3 Un paquetage Net-SNMP** ..... **25**

**2.4.4 Les bases utiles** ..... **26**

**2.4.5 Un serveur de messagerie** ..... **26**

***2.5 Les logiciels libres utilisés*** ..... ***27***

**2.5.1 Nagios et Thruk** ..... **27**

**2.5.2 Cacti** ..... **31**

**2.5.3 NagVis** ..... **32**

***2.6 Contrôle d' accès web*** ..... ***33***

**2.6.1 Configuration de l' authentification** ..... **33**

**2.6.2 Déclaration des utilisateurs** ..... **34**

***2.7 Conclusion*** ..... ***35***

**CHAPITRE 3 CONFIGURATIONS DES ELEMENTS A MONITORER** ..... **36**

***3.1 Introduction*** ..... ***36***

***3.2 L'activation de SNMP sur les équipements*** ..... ***36***

**3.2.1 Utilité de Nmap** ..... **36**

**3.2.2 Machines sous Linux** ..... **37**

**3.2.3 Machines sous Windows** ..... **37**

**3.2.4 Dispositifs d'interconnexion et périphériques** ..... **38**

**3.2.5 Test de SNMP** ..... **38**

***3.3 Les objets*** ..... ***38***

**3.3.1 Introduction** ..... **38**

**3.3.2 Les templates** ..... **39**

**3.3.3 Les hôtes** ..... **40**

**3.3.4 Les services** ..... **41**

**3.3.5 Les contacts** ..... **41**

**3.3.6 Les « timeperiods »** ..... **41**

**3.3.7 Les groupes d'hôtes** ..... **42**

**3.3.8 Les groupes de services** ..... **42**

**3.3.9 Les groupes de contacts** ..... **42**

**3.3.10 Les commandes** ..... **42**

***3.4 Création de graphes dans Cacti*** ..... ***45***

**3.4.1 Collection des données de performance vers la base de données RRD de Cacti ....45**

**3.4.2 Création de graphe ....46**

***3.5 Création de carte de supervision avec NagVis ....47***

***3.6 Conclusion....48***

**CHAPITRE 4 SIMULATION DE MONITORING....49**

***4.1 Introduction....49***

***4.2 Les machines virtuelles ....49***

**4.2.1 Description....49**

**4.2.2 Gestion réseau ....50**

**4.2.3 Création d' un site WordPress....50**

***4.3 Création de la topologie réseau....51***

**4.3.1 L' accès à Internet....51**

**4.3.2 Intérêt du firewall ....51**

***4.4 Configuration pour la supervision....53***

**4.4.1 Au niveau des hôtes....53**

**4.4.2 Au niveau du serveur Nagios ....53**

***4.5 Visualisation des interfaces web ....55***

**4.5.1 Paramétrage de la machine réelle....55**

**4.5.2 L' interface web de Nagios ....55**

**4.5.3 L' interface web de Thruk ....57**

**4.5.4 L' interface de Cacti ....58**

**4.5.5 L' interface web de NagVis ....58**

***4.6 Création d' un incident pour tester la réaction ....59***

**4.6.1 Vérification de la réaction de Thruk et de NagVis....60**

**4.6.2 Réception de la notification dans la boîte de réception Gmail ....61**

***4.7 Conclusion....62***

**CONCLUSION GENERALE ....63**

**ANNEXE A1 QUELQUES APERÇUS DE L'ELABORATION DU SYSTEME DE MONITORING…….. ....64**

**ANNEXE A2 EXTRAITS DE CONFIGURATION RESEAU DES HOTES VIRTUELS .....67**

**ANNEXE A3 QUELQUES EXTRAITS DE CODES SOURCES.....70**

**BIBLIOGRAPHIE.....72**

**FICHE DE RENSEIGNEMENTS.....75**

## LISTE DES ABREVIATIONS

| | |
|---|---|
| **ACL** | Access Control List |
| **ADSL** | Asymmetric Digital Subscriber Line |
| **ASA** | Adaptive Security Appliance |
| **ASN.1** | Abstract Syntax Number one |
| **ATM** | Asynchronous Transfer Mode |
| **BNC** | Bayonet Neill–Concelman connector |
| **CGI** | Common Gateway Interface |
| **CIDR** | Classless Inter-Domain Routing |
| **CPU** | Central Processing Unit |
| **CSMA/CD** | Carrier Sense Multiple Access with Collision Detection |
| **DARPA** | Defense Advance Research Project Agency |
| **DHCP** | Dynamic Host Configuration Protocol |
| **DMZ** | Demilitarized Zone |
| **DNS** | Domain Name Service |
| **FAI** | Fournisseur d' Accès Internet |
| **FDDI** | Fiber Distributed Data Interface |
| **FIFO** | First In First Out |
| **FQDN** | Fully Qualified Domain Name |
| **FTP** | File Transfer Protocol |
| **GLPI** | Gestion Libre de Parc Informatique |
| **GPL** | General Public License |

| | |
|---|---|
| **HTML** | Hypertext Markup Language |
| **HTTP** | Hypertext Transfer Protocol |
| **ICMP** | Internet Control Message Protocol |
| **IHM** | Interface Homme Machine |
| **ID** | Identifier |
| **IEEE** | Institute of Electrical and Electronics Engineers |
| **IETF** | Internet Engineering Task Force |
| **IIS** | Internet Information Services |
| **IMAP** | Interactive Message Access Protocol |
| **IOS** | Internetwork Operating System |
| **ISO** | International Standard Organisation |
| **IT** | Information Technology |
| **IP** | Internet Protocol |
| **LDAP** | Lightweight Directory Access Protocol |
| **LAMP** | Linux Apache MySQL PHP |
| **LAN** | Local Area Network |
| **MAC** | Media Access Control |
| **MAN** | Metropolitan Area Network |
| **MIB** | Management Information Base |
| **MTA** | Mail Transfer Agent |
| **N2Cacti** | Nagios-To-Cacti |
| **NAS** | Network Attached Storage |
| **NAT** | Network Address Translation |

**NMS** Network Management Station

**NSCA** Nagios Service Check Acceptor

**NSClient** Nagios Service Client

**NRPE** Nagios Remote Plugin Executor

**OID** Object IDentifier

**OSI** Open System Interconnnection

**P2P** Peer-To-Peer

**PHP** Hypertext Preprocessor

**PNG** Portable Network Graphics

**POP** Post Office Protocol

**PDU** Packet Data Unit

**RAID** Redundant Arrays of Inexpensive Disks

**RAM** Random Access Memory

**RFC** Request For Comments

**RHEL** RedHat Enterprise Linux

**RJ45** Registered Jack 45

**RLE** Réseau Local d' Entreprise

**RMON** Remote network Monitoring

**RRD** Round Robin Database

**RTA** Round Trip Average

**SMI** Structure of Management Information

**SMS** Short Message Service

**SMTP** Simple Mail Transfer Protocol

| | |
|---|---|
| **SNMP** | Simple Network Management Protocol |
| **SQL** | Structured Query Language |
| **SQLiteDB** | Structured Query Lite Data Base |
| **SSH** | Secure Shell |
| **TCP** | Transmission Control Protocol |
| **URL** | Uniform Resource Locator |
| **VLAN** | Virtual Local Area Network |
| **VPN** | Virtual Private Network |
| **XML** | Extensible Markup Language |
| **WAN** | Wide Area Network |
| **Wi-Fi** | Wireless Fidelity |
| **WMI** | Windows Management Instrumentation |

# INTRODUCTION GENERALE

Les réseaux informatiques sont actuellement présents partout et leurs tailles ne cessent de grandir de jour en jour. Etant rentrés dans nos mœurs, ils occupent une place prépondérante surtout dans le monde professionnel où la plupart des activités et des échanges sont devenus informatisés. Le contrôle en permanence de tous les éléments du réseau s'avère donc indispensable du fait qu'une simple défaillance pourrait engendrer de lourdes conséquences financières et organisationnelles. Or, devant la complexité des systèmes informatiques qui ont un rôle vital dans la production, il est impossible d'effectuer une surveillance manuelle.

Le monitoring, concept né il y a maintenant une trentaine d'années, est applicable dans plusieurs domaines comme l'industrie, la médecine, l'écologie, les télécommunications et les réseaux informatiques. Ce terme anglophone désigne un procédé de surveillance à distance accompagné d'un mécanisme d'alerte en temps réel. Avec l'explosion des systèmes d'information depuis les années 90, le monitoring permet à l'administrateur de pouvoir observer l'état et la performance du réseau depuis son poste de travail et de n'avoir à se déplacer qu'en dernier recours lorsqu' un dépannage d'ordre matériel est nécessaire.

C'est dans cette voie qu'est alors née l'idée de ce mémoire, intitulée : « ELABORATION D'UNE SOLUTION DE MONITORING A APPLIQUER DANS UN RESEAU INFORMATIQUE ». Le travail consiste à trouver une solution économique à long terme permettant à la fois de gagner en temps et en efficacité. Celle apportée est basée sur le programme modulaire Nagios qui est couplé à d'autres outils open-sources pour constituer un outil complet. Ceci pourra alors être appliqué sur tous les équipements du réseau afin de garantir la disponibilité permanente des services dépendant de l'informatique.

Cet ouvrage renferme quatre parties : la première représentera d'une manière générale notre contexte qui concerne les réseaux informatiques et leur importance ainsi que les notions de base à connaitre avant d'entrer dans la deuxième partie qui étudie le thème, puis explique l'élaboration du serveur de monitoring et son fonctionnement. Ensuite, la troisième partie développe les procédures à accomplir au niveau des hôtes et du serveur pour avoir les résultats exacts des contrôles. Enfin, dans la quatrième partie, on va démontrer la faisabilité de la solution proposée par une simulation dans une infrastructure réseau virtuelle.

# CHAPITRE 1
# GNERALITES SUR LES RESEAUX INFORMATIQUES

## 1.1 Introduction

Selon le contexte, la nature des nœuds et des liaisons, il existe plusieurs types de réseau. On peut citer par exemple les réseaux de transport, les réseaux de neurones, les réseaux routiers, les réseaux cellulaires et bien d'autres mais en particulier les réseaux informatiques, qui constituent le contexte du présent thème.

*Définition 1.01 :*

Un *réseau informatique* est un ensemble de moyens matériels et logiciels interconnectés afin de permettre l'échange d'informations numériques entre les utilisateurs.

Un tel réseau présente un lot d'intérêts dont :

- le partage de données (documents, images, sons, vidéos, etc…) et de ressources (processeurs, disques de stockage, périphériques, etc…) ;
- la communication entre processus et utilisateurs ;
- l'unicité de l' information ;
- le gain de temps.

Dans ce chapitre seront présentées brièvement les architectures physique et logique d'un réseau informatique ainsi que leur fonctionnement pour avoir ainsi des notions de base avant d'entrer dans le vif du sujet.

## 1.2 Architecture physique

### *1.2.1 Les catégories de réseau*

Les réseaux informatiques peuvent être classifiés selon leurs étendues géographique et leurs utilisations. Il existe plusieurs types de réseaux mais les principaux sont :

- le LAN (Local Area Network) ou Réseau Local :il appartient à un foyer ou à une organisation siégeant dans une zone géographiquement limité (immeuble, campus, etc…). Un tel réseau permet de relier des ordinateurs et des périphériques situés à proximité les uns des autres. C'est le type de réseau le plus répandu dans les entreprises. On parle de RLE ou Réseau Local d'Entreprise.

- le MAN (Metropolitain Area Network) ou réseau métropolitain: de son nom, il s' agit d' un réseau dont la taille peut couvrir toute une ville, voire plus. On obtient ce type de réseau en interconnectant plusieurs réseaux locaux.
- le WAN (Wide Area Network) ou réseau étendu : c'est un réseau à grande distance résultant de la liaison de plusieurs LAN et MAN. Internet (Interconnected Network), le réseau mondial le plus connu, est un exemple de WAN et est certainement le plus grand réseau de ce type. [1]

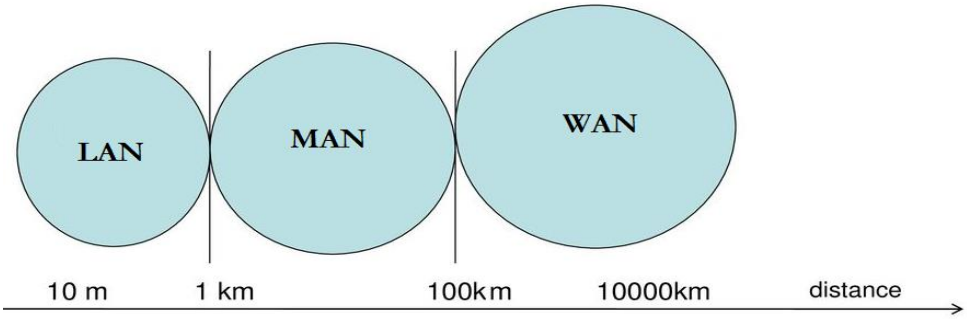


**Figure 1.01 :** *Classification des réseaux selon leurs limites géographiques*

### *1.2.2 Les composants matériels*

Généralement, le plan matériel d'un réseau informatique comprend trois types d'élément : les nœuds, les supports de transmissions et les dispositifs d'interconnexion.

#### 1.2.2.1 Les nœuds

Dans une infrastructure informatique, on désigne par nœud chaque équipement muni d'une carte réseau ou d' un pilote de réseau local qui lui permet de se connecter au réseau par une adresse physique unique appelée adresse MAC (Media Access Control) représentée sur six octets en hexadécimal (exemple : 5A:FF:B6:A2:C6:0E). Un ordinateur, un routeur spécifique, une imprimante ou une multifonction peuvent constituer des nœuds. [2]

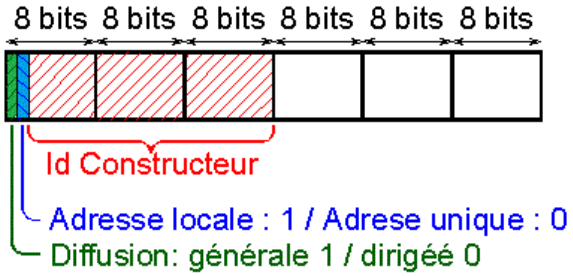


**Figure 1.02 :** *Composition d'une adresse MAC*

L’ ordinateur est une machine qui comprend plusieurs composants dirigés par le système d’ exploitation et dont les principaux sont :

- Le processeur ou CPU (Central Processing Unit) : c’ est une puce électronique considérée comme le cerveau ou l’ élément central de l’ ordinateur, lui permettant de traiter les données suivant les instructions d’ un programme. La fréquence, exprimée en Hz, indique la vitesse à laquelle il fonctionne. Le processeur ne peut pas tout gérer mais coopère avec la *mémoire vive* ou RAM (Random Access Memory).
- La RAM est le deuxième composant important dans un ordinateur. Elle mémorise temporairement pendant l’exécution d’ un programme les données à traités. Celles-ci sont sélectionnées par l’utilisateur à l’aide des périphériques d’entrée. Les résultats des opérations, c’est-à-dire les informations, sont ensuite transmises aux périphériques de sorties pour être compréhensibles.
- Le disque dur : dont la capacité peut aujourd’hui se mesurer en termes de giga-octets, est un dispositif servant à sauvegarder en permanence les programmes et les données numériques.

Le bon fonctionnement de l’ordinateur dépend alors surtout de ces trois éléments qui doivent être toujours disponibles. A part le stockage et le traitement de données, l’ordinateur permet de travailler en réseau, c’est-à-dire, d’effectuer des tâches collaboratives, par le recours à d’autres équipements.

#### 1.2.2.2 Les supports de transmission

*Définition 1.02 :*

Les *supports de transmission* sont des canaux physiques nécessaires pour relier les différentes unités de communication. Ils sont caractérisés par leurs impédances caractéristiques et leurs bandes passantes. [3]

On en distingue généralement deux types :

- les supports avec un guide physique
  - la paire torsadée : composée de huit fils renfermés dans une gaine protectrice et qui sont enrôlés deux à deux de façon à obtenir 4 paires torsadées et chaque extrémité du câble va entrer dans le connecteur RJ45. Les huit fils sont en correspondance avec les pins qui sont numérotés de 1 à 8 et ceci ne se fait pas au hasard mais respecte un code de couleur.

- le câble coaxial : il est relié à la carte réseau via le connecteur BNC en T. Le câble coaxial est composé d'un fil de cuivre rigide enveloppé d'une couche plastique. Cette dernière est entourée d'une feuille ou tresse métallique. L'ensemble du câble est recouvert d'une gaine plastique souple.
- la fibre optique : elle est introduite dans les réseaux informatiques pour pallier plusieurs points faibles des câbles de cuivre : la lumière qui y circule n'est pas sensible aux perturbations électromagnétiques et s'atténue beaucoup moins vite que le signal électrique transporté sur cuivre. On relie ainsi de façon fiable des sites distants de plusieurs centaines de mètres, voire de plusieurs dizaines de kilomètres. La fibre reste efficace dans des environnements perturbés, à des débits dix fois supérieurs aux câbles réseaux, mais son coût est comparativement élevé ;

– les supports sans guide physique : ce sont les ondes (hertziennes, radioélectriques et lumineuses) qui réalisent des connexions à distance entre les nœuds par onde radio. La norme la plus utilisée actuellement pour les réseaux sans fil est la norme IEEE 802.11, plus connue sous le nom de Wi-Fi.

Les supports filaires restent les plus utilisés en entreprise, leurs choix dépendent de la distance maximum entre les stations, du débit minimum, de la nature des informations, de la fiabilité et du coup. [4] Le Tableau 1.01 ci-dessous va résumer leurs caractéristiques générales.

| Type de support | Débit | Distance max. | Temps de propagation | Immunité aux bruits | Coût |
|---|---|---|---|---|---|
| Paire torsadée blindée(cat.5)  | 10 à 100Mbps | 100 m | 1µs /km | Bonne | Faible |
| Câble coaxial  | 100Mbps | 500m | 4µs /km | Très bonne | Moyen |
| Fibre optique  | Jusqu' à 1Gbps | 10km | 1ns /km | Excellente | Elevé |

**Tableau 1.01 :** *Les principaux supports physiques de transmission*

Un câble détaché accidentellement rend l'ordinateur injoignable et cela doit être signalé à temps au responsable pour limiter conséquences négatives.

1.2.2.3 Les dispositifs d'interconnexion

La possibilité des câbles est limitée. Lorsqu'on veut raccorder plus de deux terminaux, on doit avoir recours à des dispositifs d'interconnexion afin que les informations puissent circuler au sein du réseau. [5]

a) Le « hub » ou concentrateur

C'est un matériel possédant plusieurs interfaces RJ45 pour interconnecter les machines via des câbles droits. Avec le hub, quand un poste envoie une information, celle-ci est répercutée sur l'ensemble des postes connectés. Alors, si chaque poste commence à transmettre, la vitesse de transmission sur le réseau chute. Il est actuellement de moins en moins utilisé.

b) Le « switch » ou commutateur

Il s'agit d'un autre type de hub ayant une bande passante dédiée pour chaque interface qui est en mode full-duplex afin de limiter les collisions. Contrairement au hub, le switch ne transmet l'information qu'à la machine destinée. Il possède une table de commutation qui associe chaque adresse MAC à un port.

c) Le routeur

Le routeur est un boîtier qui a en minimum deux interfaces attribuées chacune d'adresse IP. Son rôle, comme son nom l'indique, est de router les paquets d'un réseau à un autre en assurant la liaison inter-réseaux. Il sert à interconnecter les ordinateurs repartis sur différents sites appartenant à une organisation, à un campus, à un établissement scolaire ou à une entreprise. La box fournie par le Fournisseur d'Accès Internet (FAI) est un routeur intégré d' un modem ADSL (Asymmetric Digital Subscriber Line) ou d' un point d'accès Wi-fi et a un rôle de passerelle pour relier un réseau privé à Internet.

d) Le « firewall » ou pare-feu

C'est un matériel similaire à un routeur mais il est surtout dédié à la sécurisation des réseaux. Il a un rôle de régulateur qui consiste à contrôler les communications à travers les ports. Cela permet d'affiner les trafics et donc d'éviter toute sorte d'intrusion depuis internet. Actuellement, les firewalls peuvent exister sous forme de logiciels.

### *1.2.3 La topologie physique*

*Définition 1.03 :*

La *topologie physique* d' un réseau est sa représentation spatiale définissant comment les nœuds sont interconnectés avec les supports de transmission.

Sur la Figure 1.03 sont illustrées les différents modes d'interconnexion entre les nœuds du réseau. Chacune de ces topologies possèdent ses avantages et ses inconvénients. [6]

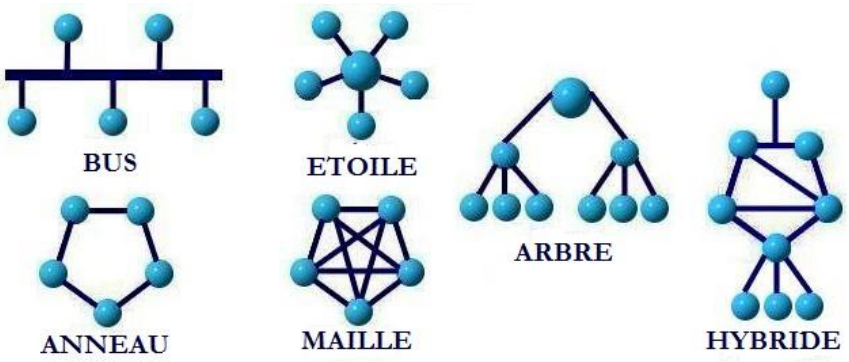


**Figure 1.03 :** *Les topologies physiques d'une infrastructure réseau*

La Figure 1.04 suivante montre une architecture réseau en général avec les composants obligatoirement présents dans un réseau informatique d' une entreprise.

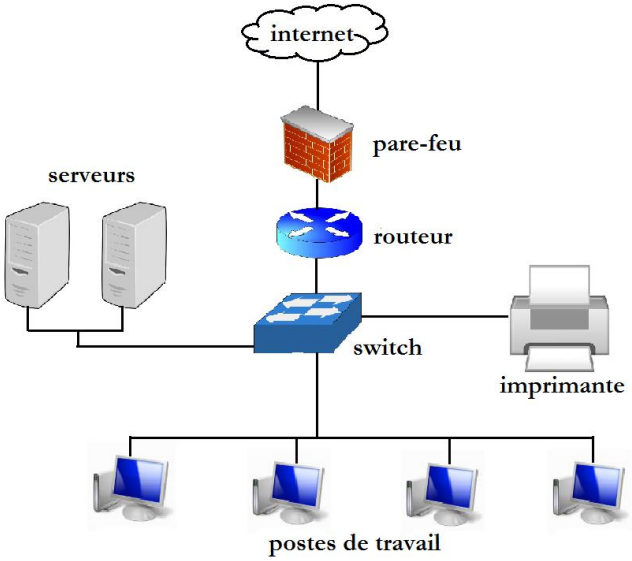


**Figure 1.04 :** *Architecture physique simplifiée d' un réseau informatique en entreprise*

## 1.3 Architecture logique

A part la liaison physique, la communication entre les nœuds repose sur une architecture logique reposant sur des protocoles.

*Définition 1.04 :*

Un *protocole* est un ensemble de règles et de procédures décrivant l'émission et la réception de données entre les éléments sur un réseau.

### *1.3.1 Les modèles OSI et TCP/IP*

1.3.1.1 Présentation

Dans le monde des réseaux informatiques et de télécommunications, il existe deux modèles principaux :

- OSI (Open System Interconnexion) : est un modèle de référence créé en 1977 par l'organisme internationale de standardisation communément connu sous ISO (International Standards Organisation) afin de permettre la communication entre les systèmes hétérogènes fournis par différents constructeurs. Il s'agit alors d'un système ouvert ayant une architecture en sept couches dont chacune détient un rôle spécifique pour établir les processus émission et réception.

| N° des | Nom des | Définition |
|---|---|---|
| Couche 7 | Application | Elle ne contient pas les applications utilisateurs, mais elle assure la communication, à l'aide de processus, entre les couches inférieures et les application utilisateurs ( transfert e fichiers, courrier électronique ). |
| Couche 6 | Présentation | Elle assure la mise en forme des données, la conversion des codes ( ASCII, EBCDIC... ), si nécessaire, pour délivrer à la couche application un message dans une syntaxe compréhensible. Elle peut aussi assurer le cryptage et la compression des données. C'est donc la première couche non impliqué dans le mécanisme de transfert d'informations. |
| Couche 5 | Session | Elle assure l'échange des données, transaction entre deux applications distantes. Elle assure surtout la synchronisation et le séquencement de l'échange par la détection et la reprise de celui-ci en cas d'erreur. Cette gestion du dialogue et de la synchronisation est assurée par jeton pour le réseau Token Ring. |
| Couche 4 | Transport | Elle assure le contrôle du transfert de bout en bout des informations entre les deux systèmes d'extrémités, afin de rendre le transport transparent pour les couches supérieures. Elle assure le découpage des messages en paquets pour le compte de la couche réseau et les reconstitue pour les couches supérieures. |
| Couche 3 | Réseau | Elle assure l'acheminement, le routage (choix du chemin à parcourir à partir des adresses ), des blocs de données entre les deux systèmes d'extrémités, à travers des relais. Et elle définit la taille de ses blocs. |
| Couche 2 | Liaison | Elle assure, le maintient de la connexion logique, le transfert des blocs de données ( les trames et les paquets ), la détection et la correction des erreurs dans ceux-ci. |
| Couche 1 | Physique | Elle assure l'établissement et le maintient de la liaison physique. Elle comprend donc les spécifications mécaniques ( connecteurs ) et les spécifications électriques ( niveau de tension ). |
| Le média ( support physique d'interconnexion ) | | |

**Figure 1.05 :** *Présentation des couches du modèle OSI*

– TCP/IP (Transmission Control Protocol/ Internet Protocol) : l'autre modèle qui fut créé dans les années 1970 par l' armée de défense DARPA (Defense Advance Research Project Agency), donc avant OSI qui devient de moins en moins utilisé dû à son complexité. Son but est l'interconnexion de réseaux sur une base planétaire avec une architecture en quatre couches. Il est donc l'origine d'internet et s'appuie sur les protocoles TCP et IP qui offrent des services de transfert de données.

Ces deux modèles permettent de classifier et d'ordonner les protocoles et les standards de communication entre les machines. Ils sont étroitement liés du fait qu' ils possèdent des couches communes et que certaines de la norme TCP/IP résultent de la combinaison de celles de l' OSI comme le montre la Figure 1.05 avec les protocoles associés.

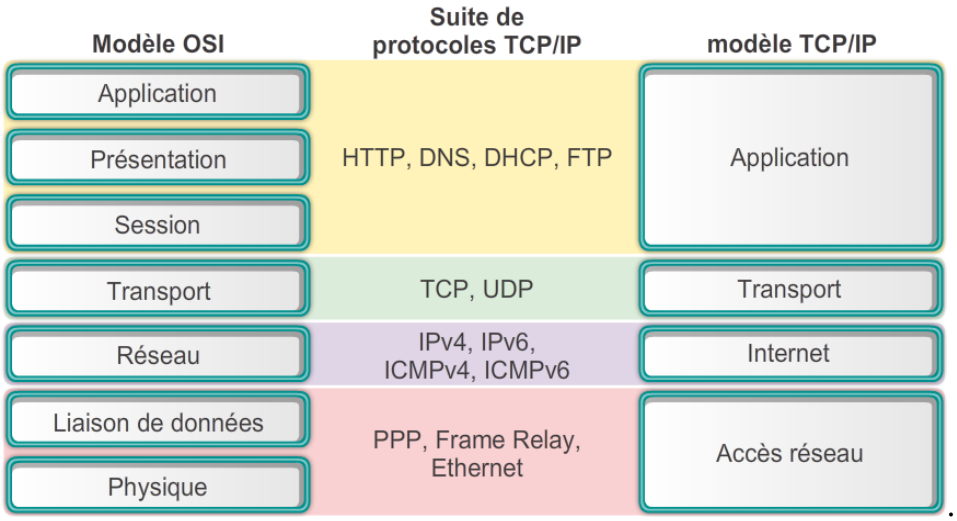


**Figure 1.06 :** *Correspondance entre modèles OSI et TCP/IP*

1.3.1.2 Adressage et classes d'adressage

IP est le protocole réseau responsable du routage, de la fragmentation et du réassemblage de paquets. L'entête de ces derniers contient une adresse IP servant à identifier une machine branchée sur internet ou sur un réseau particulier.

Une telle adresse est une suite de 32 bits regroupés en 4 octets séparés les uns des autres par des points et le tout peut être ainsi converti en décimal (exemple : 127.0.2.34) pour plus de commodité. Elle contient dans une première partie l'ID(Identifier) du réseau ou du sous-réseau et dans une deuxième partie l'ID de l'hôte. Deux machines se trouvant sur un même réseau possèdent la même adresse réseau mais pas la même adresse machine. Ce découpage en deux parties est effectué en

attribuant certains bits d'une adresse à la partie réseau et le reste à la partie machine. Il est représenté en utilisant un « masque » où sont placé à 1 les bits de la partie réseau et à 0 ceux de la partie machine. Ces adresses IP peuvent être privées ou publiques sont réparties en classes ( A, B, C, D et E) suivant les valeurs des bits les plus à gauche comme le montre le Tableau 1.02.

| Classe | Premiers bits | Plage d'adresses | Masque |
|---|---|---|---|
| A | 0 | 0.0.0.1 à 126.255.255.254 | 255.0.0.0 ou /8 |
| B | 10 | 128.0.0.1 à 191.255.255.254 | 255.255.0.0 ou /16 |
| C | 110 | 192.0.0.1 à 223.255.255.254 | 255.255.255.0 ou /24 |
| D | 1110 | 224.0.0.0 à 239.255.255.255 | Non défini |
| E | 1111 | 240.0.0.0 à 255.255.255.255 | Non défini |

**Tableau 1.02 :** *Adressage IPv4*

Au fur du temps, IPv4 rencontre une pénurie d'adresse et ne permet plus de répondre à la demande de connexion de milliards de machines informatisées dont disposeront les internautes de demain. Pour faire face à cela, deux solutions ont été recommandées :

- Le « subnetting »: segmentation logique d' un réseau important en plusieurs sous-réseaux.
- Le « supernetting », désigné aussi par CIDR (Classless Inter-Domain Routing) : permet une réduction de la table de routage et donc une économie de ressources.

Les adresses IP vont cependant manquer, IPv6 a été alors recommandé au cours du meeting de l'IETF (Internet Engineering Task Force) de juillet 1994 à Toronto. Une adresse IPv6 a une taille totale de 128 bits et est de la forme X :X :X :X :X :X :X où chaque X représente une valeur sur 16 bits. [7]

### *1.3.2 Le modèle hiérarchique en trois couches*

Du fait que les réseaux peuvent être extrêmement compliqués avec les protocoles et technologies différentes, Cisco, une société mondialement leader dans le domaine a développé un modèle hiérarchique en trois couches pour avoir une infrastructure de réseau évolutif, fiable et rentable. La section suivante discute chacune des trois couches, ayant ses propres caractéristiques et fonctionnalités, en détail :

- la couche Cœur : qui est la supérieure, également référée à un backbone . Cette couche principale est responsable du transport des grandes quantités de trafic avec un haut débit

(exemple : 10 Gigabits Ethernet). Elle relie les différents segments du réseau et assure l'interconnectivité entre des dispositifs de la couche de distribution. Son rôle inclut aussi : la vérification de la liste d' accès, le cryptage des données et la translation d' adresses. On trouve alors généralement des routeurs à ce niveau.

- la couche Distribution : sert d' interface entre la couche Cœur et la couche d' Accès. Ses fonctions principales sont de fournir le cheminement, le filtrage et l'accès WAN pour déterminer comment les paquets peuvent avoir accès au cœur. Cette couche détermine le chemin le plus rapide des requêtes de service réseau. Dans les réseaux de grande envergure, elle peut avoir différentes fonctions telles que : l'agrégation d'adresses, le filtrage le routage VLAN, la multidiffusion et la sécurité. [8]

Un VLAN (Virtual Local Area Network ) est un réseau local regroupant un ensemble de machines de façon logique et non physique. On parle alors de réseaux virtuels. En effet, dans un réseau local la communication entre les différentes machines est gérée par l'architecture physique. En effet, pour une entreprise, la création de VLANs a pour but d'isoler chaque département dans un réseau ou sous-réseau si le recours aux routeurs s'avère couteux. [9]

- La couche d' Accès : est également connu sous le nom de couche hôte-réseau. C'est la couche accessibles aux serveurs, postes des utilisateurs finaux et périphériques. Elle assure le contrôle d' accès et la livraison des paquets en établissant une liaison physique avec le média. En outre, elle se charge du filtrage d' adresses MAC et de la création de différentes domaines de collision.

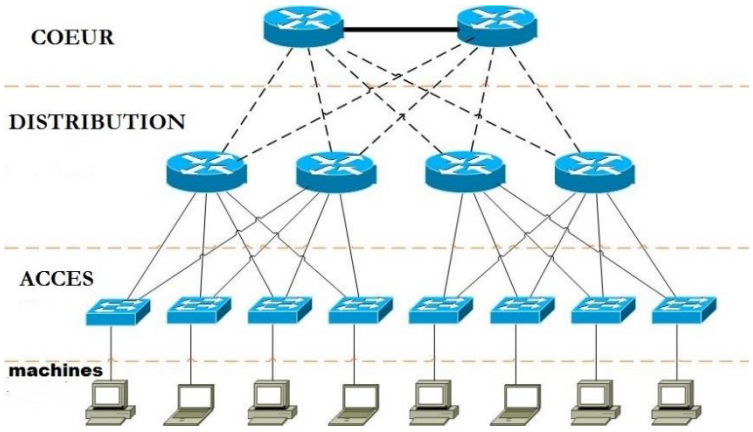


**Figure 1.06 :** *Le modèle hiérarchique en trois couches*

### *1.3.3 La topologie logique d' un réseau et la technologie Ethernet*

*Définition 1.05* :

La *topologie logique* est la structure logique d'une topologie physique, c'est à dire que la topologie logique définit comment se passe la communication dans la topologie physique. [10]

La couche de liaison de données du modèle OSI qui contrôle l'accès au support physique à l' aide des protocoles qui définissent la topologie logique du réseau souvent référée à la technologie utilisée par ce réseau.

Dans ce sens, les protocoles les plus courants sont Ethernet, Apple Talk, Token Ring (IEEE 802.5), FDDI (802.7) et ATM dont les détails sont résumés dans le Tableau 1.03.

| Technologie | Câble | Vitesse | Topologie associée |
|---|---|---|---|
| Ethernet | Paire torsadée, Coaxial, Fibre | 10 Mbps | Bus, Etoile, Arbre |
| Apple Talk | Paire torsadée | 23 Mbps | Bus, Etoile |
| Token Ring | Paire torsadée | 4 Mbps -16 Mbps | Anneau |
| FDDI | Fibre | 100 Mbps | Double anneau |
| ATM | Paire torsadée, Fibre | 155 Mbps- 2488 Mbps | Bus, Etoile, Arbre |

**Tableau 1.03 :** *Les principales technologies utilisés dans les LAN*

Dans les réseaux locaux, la technologie dominante est Ethernet qui utilise CSMA/CD (Carrier Sense Multiple Access with Collision Detection) comme méthode d' accès au support selon la norme 802.3. Il s'agit d'un protocole d'accès multiple avec surveillance de porteuse et détection de collision.

Il existe aussi plusieurs variétés d' Ethernet dont : [11]

- Les anciennes : Xerox Ethernet, 10Base5(Thick Ethernet), 10BROAD36 et 1BASE5
- Ethernet 10 Mbps : 10BASE2 (ThinNet ou CheaperNet), 10BASE-T et 10Base-F
- Fast Ethernet 100 Mbps : 100BASE-T, 100BASE-TX et 100BASE-FX
- Gigabit Ethernet 1000Mbps : 1000BASE-T, 1000BASE-X et 1000BASE-LH

### *1.3.4 Les typologies des réseaux*

Faisant partie de l'architecture logique, la typologie est le mode de communication entre les ordinateurs pour effectuer les applications : avec ou sans serveur. [12]

#### 1.3.4.1 Les serveurs

Généralement les serveurs sont des ordinateurs dédiés aux logiciels serveurs qu'ils abritent, possédant des capacités supérieures à celles des ordinateurs personnels en termes de puissance de calcul et de connexions réseau. Fonctionnant 24h/24, ils sont au centre du réseau et fournissent des services en réponse aux requêtes des clients suivant des protocoles. [13]

Voici les principaux serveurs à connaitre et leurs rôles :

- le serveur web ou serveur HTTP(Hypertext Transfer Protocol): héberge les sites web qui sont des fichiers au format HTML(Hypertext Markup Language) et répond aux requêtes des navigateurs web qui sont les clients.
- le serveur Proxy : est un serveur mandataire qui sert d'intermédiaire entre un réseau local et Internet. Il est notamment utilisé pour assurer les fonctions suivantes : stockage par cache des pages les plus souvent visitées, suivi des connexions, filtrage des connexions avec gestion des blacklists.
- le serveur de stockage en réseau : également appelé NAS(Network Attached Storage) a pour rôle de sauvegarder en continu les données générées par l'entreprise et de les centraliser en les rendant accessibles depuis plusieurs postes dans le réseau. Ce type de serveur permet la réduction du temps d'administration des postes clients en gestion d'espace disques. La technologie RAID (Redundant Arrays of Inexpensive Disks) est employée pour sécuriser les données stockées contre la défaillance d'un ou plusieurs disques durs.
- le serveur de Base de Données : permet la centralisation et l' administration des données qui sont regroupées dans des tables.
- le serveur mail: qui gère l' envoi, la réception et le stockage des messages électroniques . On a deux catégories de client pour ce serveur: le client de messagerie qui est un logiciel (Microsoft Outlook, Mozilla Thunderbird, …) et le Webmail qui est une interface web (Gmail, Yahoo, …) ;
- le serveur FTP (File Transfer Protocol) : permet le transfert de fichier. Toute personne en ayant l' autorisation peut faire le dépôt et le téléchargement des fichiers vers le serveur;
- le serveur DNS (Domain Name Service) : sert d' annuaire en faisant correspondre une adresse IP à un FQDN (Full Qualified Domain Name) composé d' un nom d' hôte et d' un nom de domaine ;

- le serveur DHCP(Dynamic Host Configuration Protocol): permet à chaque station connectée au réseau d' obtenir sa configuration réseau ( adresse IP, masque de sous-réseau, passerelle) qui est valable pendant un bail. L' adressage dynamique permet de simplifier l' administration d'un réseau et d'éviter les problèmes de conflit d' adresses IP;
- Le serveur d' impression : permet de partager une ou plusieurs imprimantes entres plusieurs utilisateurs ou ordinateurs situés sur un même réseau.
- Le serveur d' authentification : est un moyen de sécurité informatique afin de filtrer l' accès aux ressources et de garder une trace des activités réalisées durant chaque session. Tout utilisateur autorisé doit alors posséder une identification et un mot de passe. LDAP(Lightweight Directory Access Protocol) est le protocole standard permettant de gérer des annuaires, c'est-à-dire d'accéder à des bases d'informations sur les utilisateurs d'un réseau.
- Le serveur d' application :permet la distribution d'une même ressource d'application auprès de plusieurs collaborateurs
- le serveur de test : permet de vérifier le bon fonctionnement des applications web créées avant de les mettre en production.

Il existe une technique informatique, très pratiquée dans les réseaux locaux, appelée « clustering ». Cluster signifie « grappe » et désigne un groupe de serveurs vu de l'extérieur comme un seul et même serveur logique. Il répond à un double besoin : d'une part, les demandes de traitement des applications en augmentation constante auquel un seul serveur peut difficilement répondre et d'autre part, une forte demande forte de haute disponibilité d'applications. Ceci amène à la redondance des serveurs pour garantir à la fois la continuité des services mais également pour se prémunir des pannes. [14]

Cependant, le clustering est de plus en plus abandonné face à l'émergence du Cloud Computing qui consiste à exploiter la puissance de calcul ou de stockage de serveurs informatiques distants par l'intermédiaire d'un réseau, notamment Internet. [15] En effet, les avantages du Cloud portent surtout à la non limitation des ressources, la flexibilité de la connectivité et l' amélioration du rétablissement des désastres.

1.3.4.2 Les clients

Comme on l'a déjà mentionné, les clients sont des logiciels qui envoient des demandes à un serveur. Est appelé aussi client le poste sur lequel les demandes sont envoyées. [16] On peut distinguer :

- les clients légers : qui sont, au sens matériel, des machines minimalistes dépendant des serveurs pour presque toutes ses opérations. Dans le cadre d' une application web, on désigne par client léger le navigateur web(client universel), l' interface graphique étant limitée car elle n' utilise que le langage HTML;
- les clients lourds : définis comme des logiciels complexes exécutées de façon autonome sur un ordinateur. Ce dernier dispose ses propres ressources et dépend moins des serveurs. La plupart des ordinateurs personnels sont désignés en tant que tels.
- les clients riches : sont des programmes permettant l' affichage des interfaces graphiques des applications distantes. Il s'agit d'intermédiaires entre clients léger et lourd mais proposant des interfaces plus riches par l' utilisation du langage XML (Extensible Markup Language).

1.3.4.3 Les différentes architectures

Les deux notions « client » et « serveur » supposent l' existence d' une organisation lors d' un traitement des opérations au sein du réseau. [17] Dans cet axe, on distingue:

- L'architecture pair à pair (peer-to-peer ou P2P) qui est un mode de communication où chaque nœud peut jouer éventuellement le rôle de client ou de serveur ;
- les architectures à N niveaux dont :
    - l' architecture à deux niveaux (deux-tiers) qui est un environnement client-serveur proprement dit. Elle est dite centralisée puisqu' elle est basée sur un nœud central, un serveur qui peut être dédié ou non.
    - l' architecture à trois niveaux (trois-tiers) qui ajoute un autre serveur qui fournit au serveur d' application (appelé aussi middleware) les ressources nécessaires pour répondre aux clients.
    - l' architecture à N niveaux (N-tiers ou multiniveaux) qui ajoute encore des serveurs supplémentaires dont chacun est spécialisé dans une tâche précise afin d' avoir un

avantage de flexibilité et de performance. On parle alors d'architecture distribuée car les ressources sont délocalisées sur des machines réparties sur le réseau.

Ces types d'architectures sont associées à la structure des progiciels comprenant en général : une couche de présentation, une couche applicative et une couche d'accès aux données.

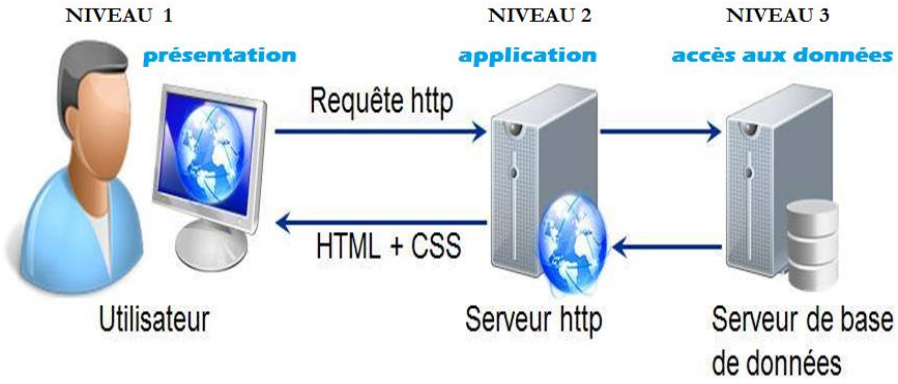


**Figure 1.08 :** *Exemple d'architecture trois-tiers*

### 1.4 Le cloisonnement des réseaux

Le système de pare-feu permet de définir des règles d'accès entre deux réseaux. Dans la pratique, les entreprises ont généralement plusieurs réseaux avec des politiques de sécurités différentes. C'est la raison pour laquelle il est nécessaire de mettre en place des dispositifs de sécurité comme les pare-feux permettant d'isoler les différents réseaux de l'entreprise: on parle ainsi de cloisonnement des réseaux .

On peut alors distinguer :

- L' intranet : un espace dans le réseau informatique interne qui permet un accès sécurisé et contrôlé à des informations et à des ressources. Il utilise les technologies ouvertes de l' Internet et donc les mêmes protocoles (TCP, IP, HTTP, DHCP, SMTP, IMAP, etc….). Dans une entreprise, il s'agit d'un internet local fournissant des sites web et des données privées seulement accessibles par les employés. Si une des machines qui assurent ces services tombe en panne, cela peut perturber les activités internes et par la suite baisser la production.
- L'extranet : une extension de l' intranet qui sert à établir l' interconnexion avec les sites distants de l' entreprise et ses partenaires externes (collaborateurs, fournisseurs, etc…). L' accès à l' extranet est fortement sécurisé grâce à la technologie VPN (Virtual Private Network) ou réseau privé virtuel. Il s' agit d' un tunnel qui est une liaison virtuelle sécurisée

permettant la communication entre deux entités distantes de manière transparente à travers internet.

- La DMZ (Demilitarized Zone): Lorsque certains services du réseau interne ont besoin d'être accessibles de l'extérieur (service web, messagerie,...), il est souvent nécessaire de créer un espace intermédiaire entre l' intranet qui est le réseau à protéger et Internet, le réseau hostile. La « zone démilitarisé » pour désigner cette zone isolée hébergeant des données mises à disposition du public. Elle fait ainsi office de zone tampon entre le réseau à protéger et le réseau hostile par le placement d' un ou de deux pare-feux. Elle est aussi appelée réseau de service. [18] L' autre manière de lancer ses services en ligne est de faire appel à un hébergeur web. Les sites seront alors gérés à l' aide d' un logiciel client FTP. Si une entreprise lance sur internet un site marchand, un problème technique au niveau des serveurs peut faire baisser le chiffre d'affaire, un autre sur un site d'information peut faire passer le lecteur à la concurrence.

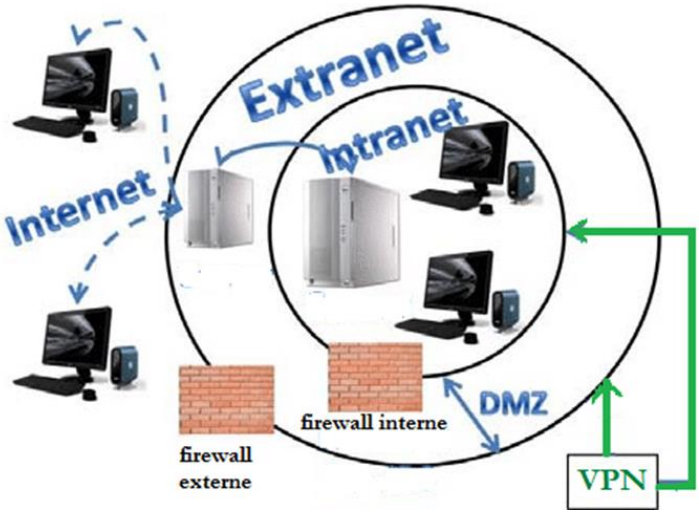


**Figure 1.09 :** Structure de sécurité hiérarchique

### 1.5 Conclusion

Ce premier chapitre nous a montré que le réseau informatique est un vaste domaine qui concerne plusieurs entités physiques et logiques ayant chacune un rôle bien défini. C'est leur interconnexion qui permet la communication, l' accès à internet et aux ressources partagées ainsi que la mise à disposition d'une plateforme de travail collaboratif. Il est alors impératif de surveiller de près tous

les équipements pour éviter l'arrêt des activités. Pour être compétitive, chaque organisation est dans l'obligation de  maintenir son réseau en bonne santé

Avant de mettre en place une plateforme de monitoring qui va répondre à ce besoin, il est indispensable de bien étudier et de comprendre l' infrastructure réseau sur laquelle on va travailler. Il faut alors bien définir la topologie du réseau et les modes d'interconnexion ainsi l'ordre d'importance de chacun des matériels. Cette étude est une étape importante pour bien définir la méthodologie de travail à adopter. Un cahier de charge doit être bien rédigé pour identifier les besoins.

Après l'étude, une critique de l'existence doit être mise en évidence afin de simplifier la mise en œuvre. En effet, il faut bien s'axer davantage sur les problèmes fréquents confrontés par les utilisateurs car le but du projet est de les repérer facilement et de pouvoir intervenir dans les plus brefs délais.

# CHAPITRE 2
# ETUDES ET CONCEPTION D'UN SERVEUR DE MONITORING

## 2.1 Introduction

Les réseaux informatiques sont tous différents par leur taille et leurs activités. Ils ont cependant pour point commun d'être le théâtre d'incidents, à un moment ou à un autre. Le monitoring est justement destiné à gérer cela. [19]

*Définition 2.01 :*

Dans ce contexte, le *monitoring* est un ensemble d'actions consistant à accueillir, à traiter et à représenter d'une manière compréhensible les informations concernant l'état d'un réseau à surveiller. Ce concept comprend deux activités principales qui sont complémentaires mais à différencier : la supervision et la métrologie.

*Définition 2.02* :

La *supervision* est une suite d'opérations qui permet d'obtenir des indicateurs d'état et des alertes en temps réel correspondant aux circonstances.

*Définition 2.03* :

La *métrologie* est définie comme le traçage de graphique décrivant les métriques de flux permettant de connaître la performance du réseau et de ses ressources.

La mise en place d'un système de monitoring vise à atteindre deux buts :

- Être réactif en étant tout de suite signalé, parfois même avant les utilisateurs, lorsqu' une anomalie ou un dysfonctionnement survient dans une partie du système informatique ;
- Être proactif en ciblant les cas alarmants et de les résoudre avant que cela ne cause de panne. [20]

On va alors élaborer une solution de monitoring. Mais avant cela, il faut maitriser les notions importantes qui tournent autour d' un protocole SNMP sur lequel reposent les activités de gestion des réseaux.

## 2.2 Le protocole SNMP

Proposé par l' IETF dans la RFC(Request For Comment)1157, SNMP est un protocole standard d'

administration de parc informatique. La supervision et la métrologie vont alors reposer sur SNMP qui permet la transmission à distance des informations concernant les dispositifs à contrôler sur le réseau. [21]

### *2.2.1 Les composants de l'environnement SNMP*

L'environnement SNMP supposent l'existence et l'interaction de plusieurs éléments-clés.

#### 2.2.1.1 Le manager SNMP

C'est un daemon à installer sur la station d'administration et qui lance des requêtes vers l'agent SNMP pour recueillir les informations de contrôle. Ces dernières sont fournies ensuite au NMS (Network Management Station) qui est une application de supervision permettant de les traiter et de les interpréter.

#### 2.2.1.2 L'agent SNMP

Il s'agit un daemon qui s'exécute sur une machine à superviser. Il est chargé de répondre aux requêtes lancées par le manager SNMP en recherchant les informations relatives à son environnement local dans la MIB (Management Information Base). Lorsqu' un événement anormal survient, il le signale au manager.

#### 2.2.1.3 La MIB et les OID

La MIB est un modèle de données associé à SNMP. C'est une bibliothèque d'objets gérables qui peuvent être des informations matérielles ou logicielles, des paramètres de configuration, des statistiques de performance ou d'autres variables qui sont directement liés au comportement de l'équipement supervisé. Il s'agit d'un fichier texte, écrit en syntaxe SMI (Structure of Management Information) et dont le nom se termine obligatoirement par « -MIB.txt » et qui peut être procuré depuis le site web du constructeur. Un OID (Object IDentifier) est une identification universelle attribué à chaque objet. Il se présente par une suite de chiffres ou de nombres séparés par des points (exemple: 1.3.0.2.6.1.7.2.9.2). Ces index numériques désignent chacun un niveau de hiérarchie dans la MIB. Ainsi, on a la translation d'un OID en un nom compréhensible. Par exemple, l'OID ifDescr (1.3.6.1.2.1.2.2.1.2) est une chaîne de caractères contenant des informations concernant une interface réseau.

Lors d'une interrogation faite à un objet, la valeur de retour peut être de type texte, entier, compteur ou tableau. Le MIB Browser aide à identifier les objets correspondant à une requête en explorant l'arborescence de la MIB.

La structure de la MIB est montrée ci-dessous sur la Figure 2.01 où chaque cadre représente nœud correspond à un OID.

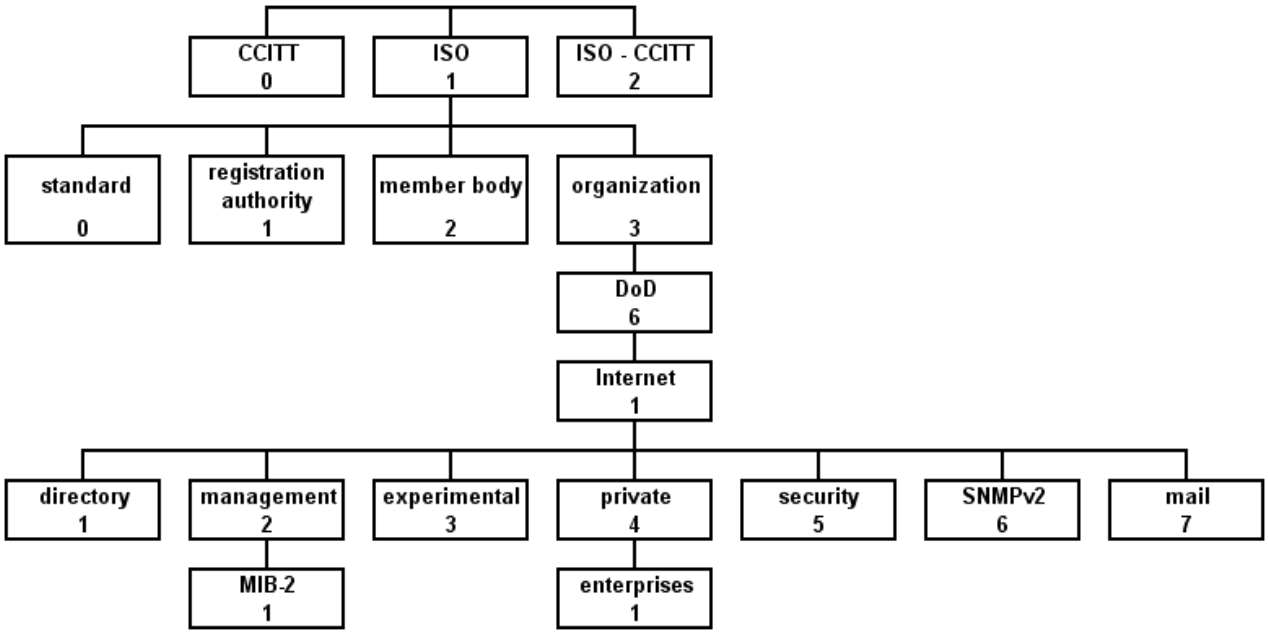


**Figure 2.01 :** *Extrait de la structure d'une table MIB*

2.2.1.3 Le nom de communauté

Afin d'éviter toutes sortes d'intrusion et de garder les informations sur le réseau confidentielles, les paquets échangés entre la station de supervision et les hôtes sont inclus chacun d'un mot de passe de type string qu'on appelle « nom de communauté SNMP ». Ceci contrôle l'accès du manager à la MIB et doit être cohérent avec le nom d'un utilisateur ou d' un groupe d' utilisateurs. Au cas où le nom de communauté défini dans le NMS ne correspond pas à celui dans l'agent, ceci ignore les requêtes. Il existe deux sortes de privilège quant à l' accès à la MIB:

- read-only (lecture seule): le manager SNMP peut seulement lire les valeurs des objets de la MIB ;
- read-write (lecture-écriture): le manager SNMP possède à la fois le droit de lire, d' ajouter et de modifier les objets de la MIB. [22]

### *2.2.2 Le message SNMP*

Comme le montre la Figure 2.02, le contenu d' un message SNMP comprend huit champs dont les deux premiers formant l'en-tête sont le numéro de version SNMP utilisé et le nom de communauté.

Ensuite, le PDU SNMP contient les informations utiles et est constitué par les six champs restants dont : le type PDU, l' ID de requête, le statut d' index, l' OID et la valeur.

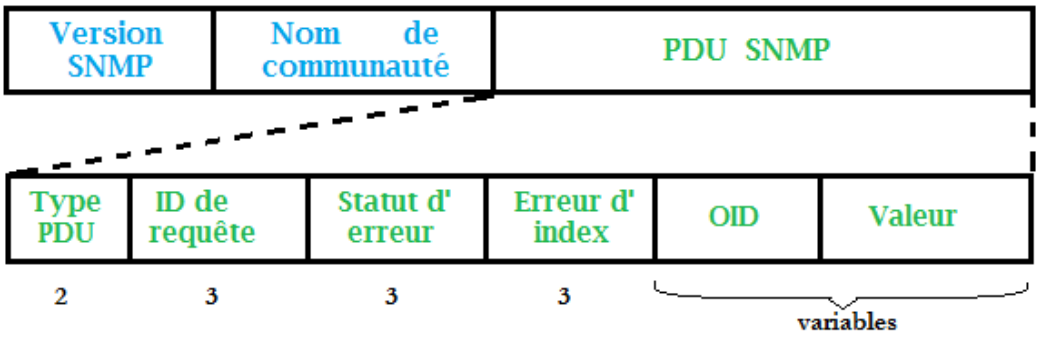


**Figure 2.02 :** *Format d'un message SNMP*

- Le type PDU est associé à une valeur numérique désignant le type de message communiqué. Il peut s'agir d'une requête, d'une réponse ou d'une alerte. Ces échanges sont illustrés sur la figure 2.03.

– get-request (type PDU = 0) : message envoyé par le superviseur à l' agent pour obtenir des informations sur l' équipement où il est installé ;

– get-next-request (type PDU=1): requête utilisée par le superviseur pour demander l' information suivante à l'agent ;

– set-request (type PDU=2): message envoyé par le superviseur à l' agent pour mettre à jour ce dernier en modifiant la valeur d' un objet dans la MIB;

– get-response (type PDU=3): signifie que l' information a bien été transmise ;

– trap (type PDU=4) : une alerte envoyée par l' agent au manager pour signaler un dysfonctionnement. Les versions du protocole SNMP sorties après SNMPv1 utilisent le terme « notifications » pour faire référence aux traps.

- L' ID de requête sert à identifier chaque requête et d' y associer les réponses correspondantes.
- Le statut d'erreur indique le type (*NoSuchObject*, *NoAccess*, ou *NoWritable)* et le nombre d' erreurs produites.
- Le champ « erreur d' index » indique la position d'une erreur en donnant la variable qui l' a causée.
- Les deux derniers champs « OID » et « Valeur » sont combinés et contiennent une liste de variables composées d'identificateurs décrits dans la MIB et de leurs valeurs.

Il existe deux types de supervision :

- La supervision active ou Polling : elle est à l'initiative du superviseur et consiste à interroger l' état de l' hôte  suivant un intervalle de temps.
- La supervision passive ou Heartbeat : elle est à l' initiative de l' hôte supervisé qui envoie une alerte au superviseur en cas de mal fonctionnement. Cela peut être la cause un arrêt non commandé, une rupture de liaison ou un échec d' authentification.

SNMP utilise généralement UDP comme protocole de transport pour délivrer les requêtes et leurs réponses. Les ports utilisés sont le 161 pour l' agent et le 162 pour le manager. [23]

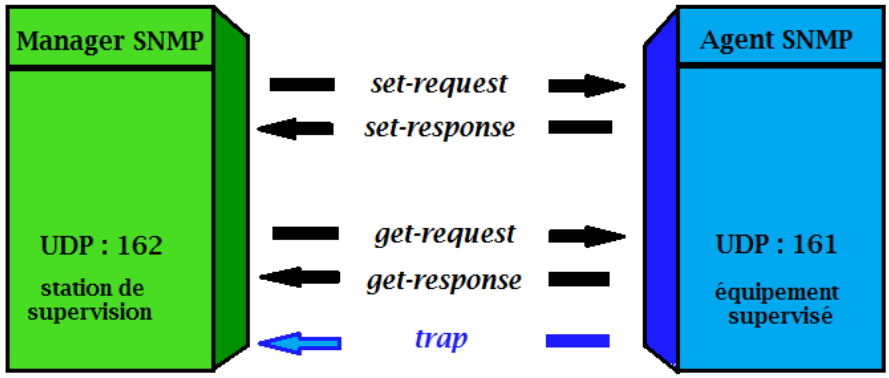


**Figure 2.03 :** *Illustration des échanges entre agent et manager SNMP*

### *2.2.3 Les versions du protocole SNMP*

Le protocole SNMP existe sous plusieurs versions dont trois sont les plus suivies, à savoir :

- SNMPv1 : c'est la première version, apparue en 1989, qui reste encore la plus utilisée aujourd'hui mais son principal inconvénient est sa faiblesse en terme de sécurité ;
- SNMPv2c : qui se différencie de SNMPv1 par l' ajout de l' opération Get-Bulk qui permet au manager de demander en bloc plusieurs variables consécutives dans la MIB de l'agent.;
- SNMPv3 :  cette version possède les mêmes caractéristiques que les deux premières mais corrige leur manque de sécurité grâce à un cryptage du PDU.

Toutefois, il faut savoir qu'il existe d'autres possibilités de faire une surveillance du réseau telles que:

- La commande Ping : à laquelle est associée l'adresse IP ou le nom de domaine d'un hôte pour savoir si ce dernier est connecté ou non sur le réseau ;

- ICMP (Internet Control Message Protocol) : un protocole utilisé par les routeurs et les commutateurs pour envoyer un message d'erreur indiquant qu'un hôte est injoignable ;
- L'utilisation des agents Syslog : qui recueillent les événements anormaux subits par un équipement (similaires aux traps SNMP) et les envoient au serveur Syslog ;
- Le standard RMON (Remote network Monitoring) : permettant de collecter des données de trafic et de les représenter sous forme tabulaire ou graphique.

### 2.3 Etude de la plateforme de monitoring

Les possibilités qu'offrent les méthodes basiques de monitoring sont très limitées (même les commandes SNMP) aux niveaux affichage, enregistrement des événements, fréquence des requêtes, etc… Cela est assez compliqué, voire impossible sans avoir recours à un logiciel spécifié. L' idée est alors d'élaborer un outil programmable qui va fonctionner d'une manière automatique. Afin d'aider au mieux les administrateurs pour leur tâche de surveillance, on est alors amené à chercher une solution destinée à accomplir les fonctionnalités suivantes :

- L'exécution des commandes : toutes les activités concernant le monitoring sont accomplies selon les directives que l' administrateur décrit à travers les configurations et les programmes écrits au préalable ;
- La supervision réseau : c'est la surveillance des équipements réseaux ( routeur, switch, hub,…) ;
- La supervision système : c'est la surveillance des ressources matérielles (processeur, espace disque,…) des postes de travail et en particulier des serveurs;
- Le traçage de graphes : ou simplement la métrologie ayant pour vocation de fournir des courbes qui illustrent analogiquement les résultats de la supervision ;
- L'alerting ou envoi de notifications : toute défaillance subite par l' infrastructure réseau doit être informée à l' administrateur via un mail ou un SMS ;
- La cartographie : ceci permet d' avoir une vue d' ensemble sur tous les éléments du réseau et de son architecture physique ;
- Le reporting : c'est la génération des rapports qui synthèses les résultats des requêtes. [24]

Il n'existe pas un progiciel qui puisse effectuer toutes ces opérations, si l'on doit mettre de côté les solutions payantes. On doit envisager plusieurs applications ayant chacune son rôle spécifique lié au monitoring et qui seront rassemblés pour former un système. On parle alors de plateforme puisqu' elle centralise tous les outils assurant les fonctions liées au monitoring.

Pour mieux saisir les notions qui gravitent autour de ce concept, on les a illustrées sur la Figure 2.04 suivante.

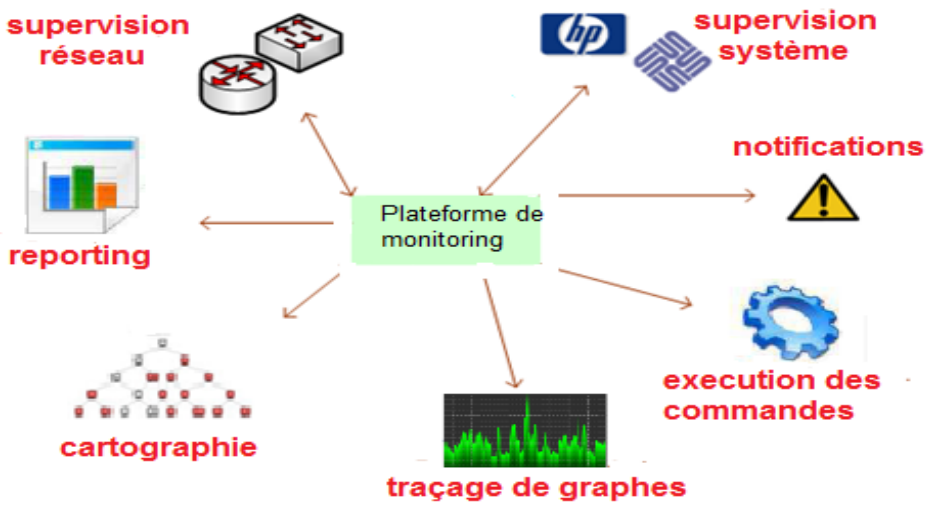


**Figure 2.04 :** *Les activités gérées par la plateforme de monitoring*

## 2.4 Les prérequis avant élaboration

Pour s'exécuter, le serveur de monitoring a besoin d'être allié à un ensemble d'éléments de base qu'il faut préalablement mettre en place.

### *2.4.1 Un noyau Linux*

Le serveur de monitoring va fonctionner sous CentOS Server, qui a été choisi pour sa qualité professionnelle. Néanmoins, pour pouvoir l' utiliser, il faut une certaine familiarisation en système d'exploitation Linux comme savoir exploiter les fichiers et utiliser l'éditeur de texte « vi ».

### *2.4.2 Une connexion à internet*

Puisque le travail se fait sur Linux, l'acquisition des paquets se fait par téléchargement sur internet et avec la commande « wget lien_de_téléchargement ». Il est également nécessaire de faire de temps en temps une mise à jour du système afin d'éviter les éventuels problèmes lors des futures installations. Avec les distributions RHEL(RedHat Enterprise Linux) comme CentOS, on utilise souvent la commande « yum ».

### *2.4.3 Un paquetage Net-SNMP*

Pour mettre en place un manager, Net-SNMP est l'application la plus adéquate étant un ensemble de programmes consoles permettant de tout faire avec SNMP. A part *snmpwalk* et *snmpget*, bien d'autres commandes SNMP sont exploitables une fois qu' on a dépaqueté et activé « netsnmpd ». Il

permet de rendre l'outil de monitoring fonctionnel. Pour ce faire, on doit créer les fichiers snmpd.conf (supervision active) et snmptrapd.conf (supervision passive) dans l' emplacement *etc/snmp/* et préciser les informations requises par le protocole SNMP.

```
#        sec.name   source   community
com2sec AllUser 127.0.0.1 public
com2sec AllUser 10.0.4.15/24 fihobiana
group MyGroup v1 mynetwork
#                sec.model  sec.name
group    AllGroup   v2c        AllUser

#SystemView which includes everything under .1.3.6.1.2.1.1.1
#                      incl/excl     subtree
view       SystemView   included       .1.3.6.1.2.1.1
view all   AllView      included       included.1
```

**Figure 2.05:** *Configuration de Net-SNMP*

### *2.4.4 Les bases utiles*

Avant l'installation des applications contribuant à la réalisation du monitoring, il est indispensable d'installer quelques bases sur lesquelles elles reposent pour fonctionner. [25] Elles requièrent :

- un serveur LAMP (Linux Apache MySQL PHP) pour permettre l' affichage d' interface web dynamique, la gestion et stockage des données;
- d'autres serveurs de bases de données : SQLiteDB et RRDtool
- diverses dépendances dont snmpd, httpd, build-essentiel, nmap, openssl, etc…

### *2.4.5 Un serveur de messagerie*

Pour permettre l'envoi des alertes par mail, il faut avoir recours à un serveur de messagerie électronique. Le processus d'une telle opération est assuré par les protocoles SMTP (Simple Mail Transfer Protocol), IMAP (Internet Message Access Protocol) ou POP3 (Post Office Protocol). Pour ce faire, Postfix a été choisi pour servir de MTA(Mail Transfer Agent) émetteur [26]

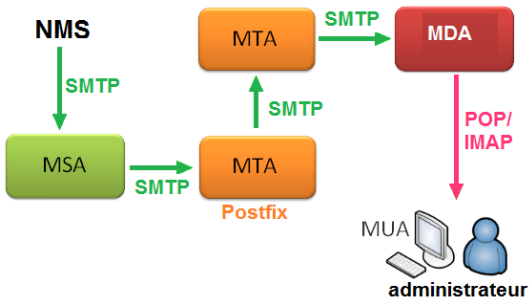


**Figure 2.06 :** *Fonctionnement du courrier électronique*

## 2.5 Les logiciels libres utilisés

### *2.5.1 Nagios et Thruk*

2.5.2.1 Présentation de Nagios

Nagios, qui est un logiciel sous licence GPL(General Public License) non propriétaire écrit en langage C, et a été choisi comme NMS de la solution. Même avec l'apparition de ses nombreux forks ayant l'air prometteurs, il reste le produit de supervision de référence. Son point fort réside surtout dans sa puissance, son architecture richement modulaire et sa grande communauté. [27] Pour notre cas, on a pris la version 3.5.1 de Nagios. On peut bénéficier des fonctionnalités suivantes :

- la supervision réseau
- la supervision système
- la supervision applicative :supervision des services réseaux (HTTP, FTP, SMTP, DNS,…)
- la notification : Le mécanisme d' envoi de mail est montré sur la Figure 2.05
- l' indication d' état : tous les éléments avec leurs états correspondants sont regroupés dans un tableau
- le reporting : consistant à obtenir des rapports sous forme de tableau de bord et de diagramme pour visualiser l' état de santé du réseau de manière générale.

Nagios peut aussi fournir des graphes si on fait appel au module « pnp4nagios » mais on a dédié cette tâche à un autre outil plus spécifique qui les réalisent mieux. Il s' agit de Cacti dont on parlera dans les prochaines lignes.

2.5.2.2 Configuration de l' alerte

Il est probable que l'administrateur ne passe la plupart de son temps à être assis devant sa station pour visualiser les états et les graphes de performance du réseau. Or, il doit toujours être au courant des incidents qui peuvent survenir. C'est pour cette raison que la fonction de remontée d'alerte (non visuelle) est essentielle. Les mails d' alerte peuvent être consultés momentanément par l'utilisation d' un smartphone sur lequel est installée une application qui génère du son lorsqu' une notification est reçue.

Pour avoir les notifications Nagios dans la boîte de réception, on configure Postfix en tant que relais SMTP et on active l' authentification en éditant le fichier */etc/postfix/main.cf* .

Dans notre cas, on va utiliser Gmail.

```
relayhost=[smtp.gmail.com]:587
smtp_use_tls=yes
smtp_sasl_auth_enable=yes
smtp_sasl_password_maps=hash:/etc/postfix/sasl_passwd
smtp_tls_CAfile=/etc/ssl/certs/ca-bundle.crt
smtp_sasl_security_options=noanonymous
smtp_sasl_tls_security_options=noanonymous
```

**Figure 2.07:** *Déclaration de Gmail en tant que serveur SMTP*

Pour préciser l' adresse mail à utiliser pour envoyer les mails, on crée le fichier *sasl/passwd.cf* en précisant l' identifiant du compte gmail et le mot de passe correspondant.

```
[smtp.gmail.com]:587 fihobiana21@gmail.com:[redacted]
```

Ensuite, on exécute ce fichier ainsi que postmap pour créer une sorte de base de donnée puis on redirige le certificat vers */etc/postfix/ca-bundle.crt*.

```
chmod 400 /etc/postfix/sasl_passwd
postmap /etc/postfix/sasl_passwd
cat /etc/ssl/certs/monitoring-server.crt | sudo tee -a /etc/postfix/ca-
bundle.crt
```

2.5.2.3 Fonctionnement

L'architecture globale de Nagios se décompose en trois parties alliées :

- un moteur, le cœur de Nagios : il ordonnance les vérifications, analyse les informations reçues et fait déclencher alertes. En résumé, il s'agit d'un planificateur de tâches ;
- des plugins ou greffons : ce sont des programmes externes ou des scripts qui servent à contacter les hôtes et en récupérer les informations souhaitées. Un grand nombre de plugins standards est déjà livré lors de l'installation de Nagios mais l'on peut encore en créer.
- une IHM (Interface Homme Machine) : c'est une interface web de visualisation basée sur les CGI (Common Gateway Interface) pour afficher les résultats d' une manière compréhensible. Elle est accessible via un navigateur web grâce à Apache.

Le principe de fonctionnement de Nagios repose sur l'utilisation des plugins qui sont installés et compilés sur le système qui le supporte. Chaque équipement supervisé a ses plugins correspondants. Les agents SNMP reçoivent les requêtes envoyées par Nagios, recherchent localement les

informations demandées dans la MIB et les transmettent au manager pour ensuite être stockées dans la base de données MySQL via le script « ndo2db ». Le moteur analyse les résultats obtenus et met à jour l'interface web grâce à un autre script appelé « ndomod ».

En cas d'un événement critique, il va générer des alertes depuis Postfix vers le serveur de messagerie du responsable. [28]

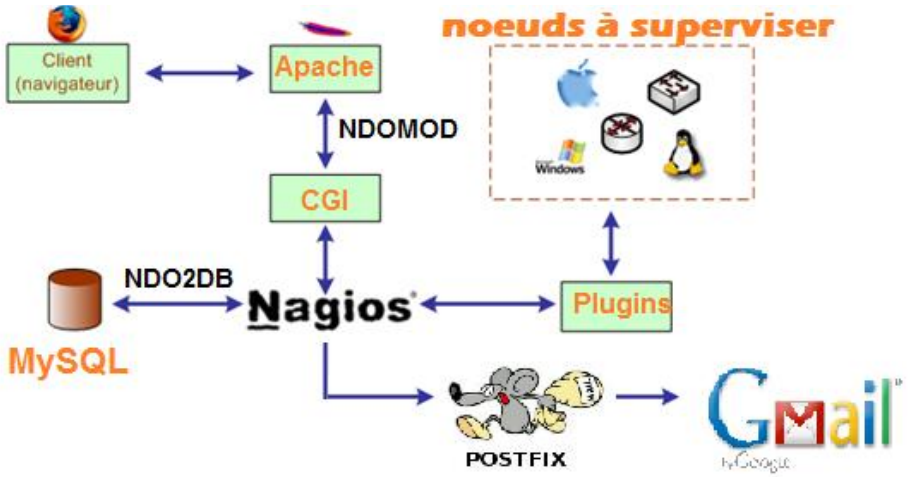


**Figure 2.08 :** *Schéma du fonctionnement général de Nagios*

2.5.2.4 Etablissement du lien entre Nagios et Thruk

Pour pouvoir représenter les résultats de Nagios de manière intuitive, on va connecter ce dernier en tant que backend au module Thruk , dans le fichier */etc/xinetd.d/livestatus.cfg.* On y précise le port utilisé par le module « MK Livestatus » utilisé par Thruk pour communiquer avec le moteur de Nagios pour accéder à ses données de les afficher.

```
service livestatus
{
            type=UNLISTED
            port=6557
            socket_type=stream
            protocol=tcp
            wait=no
            flags=NODELAY
            user=nagiosadmin
            server=/usr/local/nagios/bin/unixcat
            server_arges=/usr/local/nagios/var/rw/live
            only_from=10.0.4.15
            disable=no
}
```

**Figure 2.09:** *Déclaration de Nagios en tant que backend de Thruk*

La création de tableau de bord à afficher sur l' interface de Thruk se fait par l' édition d' un patch par lequel on a choisi l' apparence « Pie Chart » pour le contenu des widgets. (Voir Annexe A3.1) Il va alors s'exécuter en tant que procès Fast CGI et requiert l' ouverture d' une socket pour que les informations de Nagios puissent être affichées. [29]

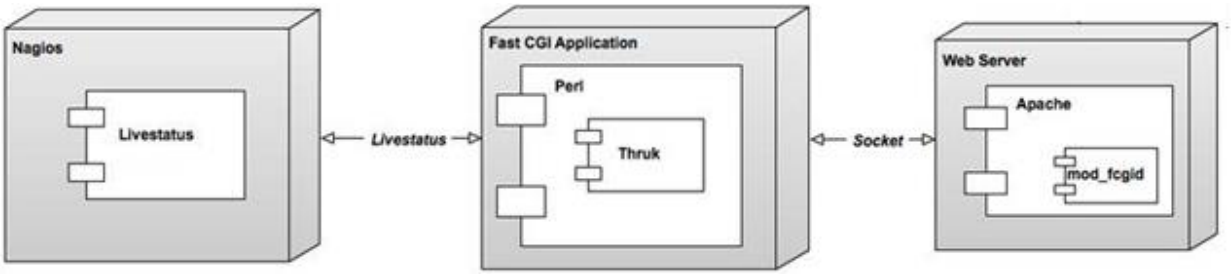


**Figure 2.10:** *Liaison entre Nagios et Thruk*

2.5.2.4 Les agents NRPE et NCSA

Pour superviser les serveurs et les postes de travail qui sont au plus grand nombre sur le réseau, Nagios utilise des agents de supervision particuliers pour récupérer les informations à distances. Le surplus de ces agents est qu'ils permettent de sécuriser les trafics SNMP grâce au protocole SSL (Secure Socket Layer). Celui-ci est un complément de TCP/IP et applique un système de chiffrement (cryptage symétrique ou asymétrique) sur n'importe quel protocole utilisant ce modèle. Son utilisation vise à atteindre trois objectifs dont : la confidentialité, l'intégrité et l'authentification.

Pour les machines fonctionnant sous Linux, il existe 2 types d'agent :

- Les agents NRPE (Nagios Remote Plugin Executor): permettant de faire une supervision active. Leur principe de fonctionnement est simple : les plugins étant installés sur l' équipement à superviser, le daemon NRPE va faire office de serveur tandis que sur l' hôte de Nagios, le plugin « check_nrpe » jouera le rôle de client NRPE. Ce dernier initie une connexion vers l'agent NRPE de la machine cible et lui demande alors l'exécution d'une vérification qui est le lancement d'un plugin correspondant configuré en local pour obtenir l'information. Un code de retour va être retourné à Nagios. Pour les machines tournant sous Windows, l' agent est NSClient++ et « check_nt » le plugin correspondant.
- Les agents NSCA (Nagios Service Check Acceptor) : qui diffèrent des agents NRPE car la vérification est planifiée en local sur l'équipement supervisé (supervision passive), puis le

résultat est envoyé au serveur Nagios. De même que pour NRPE, l'architecture NCSA demande la présence du plugin « check_ncsa » sur Nagios. [30]

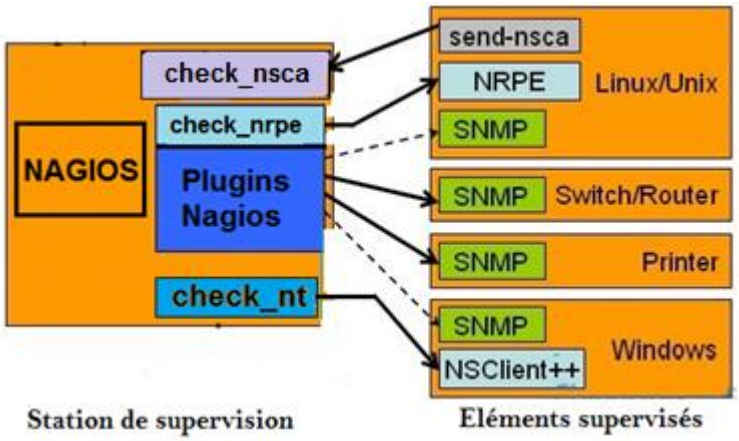


**Figure 2.11 :** *Les plugins et leurs agents correspondants*

### *2.5.2 Cacti*

2.5.3.1 Présentation

Cacti, un logiciel open-source de métrologie permet de surveiller l' état de son réseau informatique à partir de graphiques quotidiens, hebdomadaires, mensuels et annuels. Cette solution n'est donc pas destinée à alerter en temps réel sur les dysfonctionnements d'un système mais de donner une vision de l'évolution d'indicateurs matériels et logiciels (trafic réseau, occupation des disques, taux d' erreur, temps de réponse, etc...) sur les dispositifs utilisant SNMP. On a installé la version 1.1.0 de Cacti pour compléter la solution de monitoring.

2.5.3.2 Fonctionnement

Cacti fonctionne de façon similaire à Nagios à l' aide du protocole SNMP et lance les requêtes sous forme de plugins. Mais il s'appuie sur la technologie de bases de données RRDTool pour stocker les données acquises et utilise aussi MySQL pour gérer les données de configuration. RRD est l'acronyme de Round Robin Database, qui peut se traduire par «base de données cyclique». Ce mécanisme permet de stocker des données dans des fichiers de taille invariante, définie à la création, par un mécanisme de pile FIFO (First In First Out). Les graphes sont obtenus grâce au daemon appelé « poller.php». [31]

2.5.2.3 Etablissement du lien entre Nagios et Cacti

Le programme « N2Cacti» est une passerelle entre l'outils de supervision Nagios et le frontend RRDTool de Cacti.

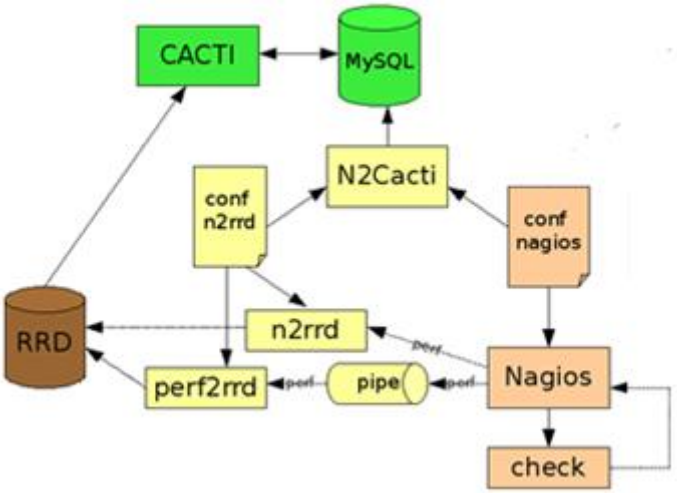


**Figure 2.12:** *Fonctionnement de Cacti avec l' intégration de Nagios*

En effet, les données de contrôles retournées par les plugins de Nagios doivent être converties en données de performance pour pouvoir être transportées dans la base de données RRD de Cacti par le programme « perf2rrd ». Cela se fait par le mécanisme « pipe ». L' autorisation de ces opérations passe par l' édition du fichier *n2rrd.conf* (Annexe A1.1) et par la configuration de */etc/nagios.cfg* dont on reparlera une fois qu' on aura maitrisé les notions d' objets dans la troisième partie de l' ouvrage. Nagios réalise toutes les opérations d'ordonnancement, il laisse à Cacti le soin de générer les graphiques liés à l'activité de métrologie. [32]

### *2.5.3 NagVis*

2.5.3.1 Présentation

NagVis est un programme développé en PHP, permettant de réaliser une cartographie en mode graphique. Il peut ainsi faire une représentation en miniature d' un réseau informatique dans son interface web.

2.5.3.2 Etablissement du lien entre Nagios et NagVis

A part la création de carte, NagVis offre la possibilité de visualiser l'état correspondant à chaque élément supervisé par Nagios. Ce dernier est alors déclaré comme backend dans le script

*nagvis.ini.php-sample* (Annexe A3.2) et ses données sont transmises à NagVis via le module « Livestatus ». Ainsi, l'administrateur peut facilement avoir un aperçu de l'ensemble du parc informatique avec les données de Nagios via l'interface web de NagVis. Celui-ci utilise MySQL et SQLiteDB comme bases de données et PHP pour la représentation dynamique des états fournis par Nagios. [33]

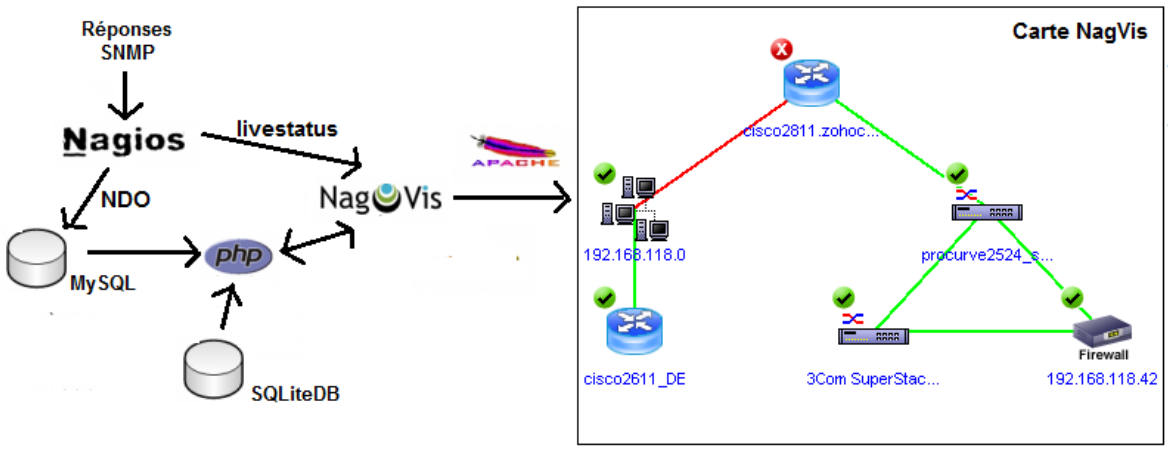


**Figure 2.13 :** *Lien entre Nagios et NagVis avec exemple de cartographie*

### 2.6 Contrôle d' accès web

L' accès aux interfaces graphiques pour visualiser les résultats des opérations de monitoring suit le principe du serveur web Apache.

*Définition 2.04 :*

Un *utilisateur authentifié* est quelqu'un qui s'est enregistré auprès d'un serveur http avec un nom d'utilisateur et un mot de passe, et à qui ce serveur a donné accès aux CGI.

#### *2.6.1 Configuration de l' authentification*

La première étape est de s'assurer que le fichier d'Apache access.conf contient une directive 'AuthOverride AuthConfig' concernant les répertoires */sbin* et */share* de Nagios. Ce changement n'est appliqué qu'au redémarrage du serveur web.

```
<Directory /usr/local/nagios/sbin>
AllowOverride AuthConfig
order allow,deny
allow from all
Options ExecCGI </Directory>
```

**Figure 2.14 :** *Autorisation de l'accès des données de Nagios à s' afficher dans l' interface web*

La seconde étape consiste à créer un fichier .htaccess aux racines de ces mêmes répertoires.

```
AuthName "Nagios Access"
AuthType Basic
AuthUserFile /usr/local/nagios/etc/htpasswd.users
require valid-user
```

**Figure 2.15 :** *Création d'un fichier d'accès pour les utilisateurs*

### *2.6.2 Déclaration des utilisateurs*

Maintenant que le serveur web est configuré pour réclamer un accès authentifié aux CGI, on doit déclarer les utilisateurs autorisés à y accéder. La commande « htpasswd » crée un fichier *htpasswd.users* dans le répertoire */usr/local/nagios/etc* et une entrée pour le nom d'utilisateur avec un mot de passe choisi. Pour notre cas, on ne va créer qu' un seul compte. [35]

```
-c /usr/local/nagios/etc/htpasswd.users fihobiana
```

L'activation des cookies apporte un avantage intéressant dans le cadre de la gestion des usagers : c'est la synchronisation des authentifications. Cet affinement de droit d'accès permet à un même usager connecté à Nagios de ne plus être obligé de se relogger pour passer aux interfaces de Thruk, de Cacti et de NagVis. Les fichiers concernés se trouvent dans le répertoire */etc/httpd/conf.d/* [34]

```
<Directory /usr/local/cacti>
    AuthType Basic
    AuthName "Restricted"
    AuthFormAuthoritative On
    AuthFormMySQLSocket /var/lib/mysql/mysql.sock
    AuthoFormMySQLUsername nagios
    AuthFormMySQLPassword root
    AuthFormMySQLDB nagios
    AuthFormMySQLTableSID sessions,users,groupright
    AuthFormMySQLFieldUID sessions.user_id
    AuthFormMySQLTableSIDCondition "'sessions'.'session_id'=$session_id AND
'sessions'.'user_id'=$user_id AND 'users'.'group_id'=$group_id AND 'groupright'
.'group_id'=$group_id AND 'groupright'.'tab_6'="1"
    AuthFormPageLogin /login.php
    AuthFormSessionCookies On
    Require valid-user
    SetEnvIfCookie "user_name=([^;]+)" REMOTE_USER=$1
```

**Figure 2.16 :** *Activation des cookies pour Cacti*

Toutes les ressources matérielles et logicielles nécessaires pour monter le serveur de monitoring sont rassemblés sur une même machine donc on peut dire que notre serveur a une architecture

centralisée. Les diverses dépendances installées sur CentOS sont utilisées par les logiciels open-source choisis pour pouvoir fonctionner.

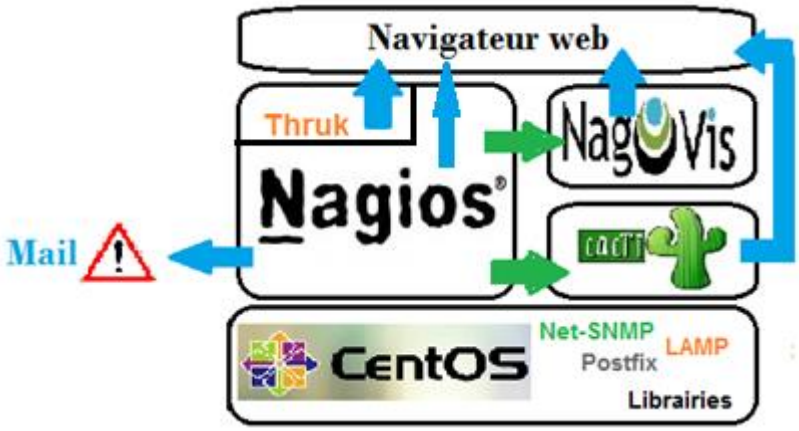


**Figure 2.17 :** *Composants de la solution de monitoring*

## 2.7 Conclusion

Le protocole SNMP est le plus puissant de tous les protocoles d'administration réseau et système. Il met en jeu plusieurs composantes ayant chacune un rôle spécifique et qui contribue à la réalisation de la supervision.

La mise en cohérence des différents outils choisis se fait par l'écriture de scripts, le passage à diverses configurations et la simplification de l'accès aux interfaces web. Nagios s'appuie sur le protocole SNMP pour collecter les données qui seront ensuite traitées et peuvent être affichées sur les pages de Thruk, Cacti et NagVis. Ainsi, ils n'opèrent pas indépendamment les uns des autres mais se voient comme un seul outil.

A travers ce deuxième chapitre, on a fait une approche du concept de monitoring qui regroupe plusieurs fonctionnalités coopérant logiquement entre elles. Celles-ci ont pour but d'apporter une évolution dans le cadre de la gestion d' un parc informatique.

# CHAPITRE 3
# CONFIGURATIONS DES ELEMENTS A MONITORER

## 3.1 Introduction

Une fois que le serveur de monitoring est monté, il faut maintenant prendre en compte l'environnement que l'on souhaite contrôler. L'exécution de cette tâche suit une logique à bien respecter pour avoir les résultats exacts.

Nagios comporte déjà une configuration initiale et opérationnelle du localhost qui est le système sur lequel il est installé. D'ailleurs, ce fait vérifie que le serveur est fonctionnel. Mais la solution a comme vocation de monitorer tous les hôtes du réseau et leurs services.

Avant d'aller plus loin, il faut assurer que le serveur soit connecté au réseau et que les requêtes « ping » vers tous les éléments à surveiller retournent des réponses positives.

Ce chapitre montre les notions à maitriser et la démarche à suivre pour chaque élément à monitorer. Il se concentre davantage à Nagios mais cela est un passage pour configurer plus simplement Cacti et NagVis puisque ces outils sont déjà mis en cohérence.

## 3.2 L'activation de SNMP sur les équipements

### *3.2.1 Utilité de Nmap*

L' utilisation de l' outil d' autodécouverte Nmap permet de reconnaitre facilement tous les équipements possédant une adresse IP connectés au réseau. Cela consiste à créer un « job » et de saisir une plage d'adresses IP puis Nmap réalise un scan et affiche la liste des hôtes détectés. Cela permet d'éviter les gaspillages de temps en effectuant des inventaires sur terrain. [36]

La prochaine étape consiste ensuite à les configurer un par un. Il faut en premier lieu mettre en place l' agent SNMP sur chaque hôtes afin d' établir la communication avec le manager SNMP et permettre à celui-ci de lancer les requêtes. Ensuite, l'activation SNMP consiste à indiquer l' adresse IP du serveur de monitoring, le type de supervision, la version ainsi que le nom de communauté utilisé par Net-SNMP.

En pratique réelle, cette activation peut être rendue plus simple par la possibilité de se connecter aux hôtes et de les configurer à distance. Ainsi, lorsqu' un problème d' ordre non matériel surgit , on peut les dépanner sans avoir à se déplacer. Tout cela permet de bénéficier d' un gain de temps considérable.

### *3.2.2 Machines sous Linux*

Telnet est une référence pour l'accessibilité entre systèmes linux mais il est de moins en moins utilisé aujourd'hui au profit de SSH qui est plus sécurisé. Si le paquet « openssh-server » est installé sur une machine équipée de linux, elle peut être accéder à distance depuis le serveur de monitoring par la commande « ssh login@IP » avec le login créé dans */etc/ssh/ssh_config* et l' adresse IP locale de cette machine. [37]

La connexion est établie après saisi du mot de passe. On installe enuite l' agent NRPE qui requiert une connexion à internet puis on édite le fichier de configuration correspondant. La syntaxe des commandes dépend du type de la spécification des versions Linux. Lorsque la machine possède une interface graphique, l' éditeur de texte « nano » peut être utilisé à la place de « vi ».

### *3.2.3 Machines sous Windows*

Pour la prise en main à distance d'un poste Windows, on peut avoir recours à l' outil « rdesktop » qui est à installer sur le serveur de monitoring. On active le « Bureau à Distance » dans les propriétés système de Windows. [38]

L' activation du protocole SNMP dans les machines sous Windows est plus simple car cela peut se faire en mode graphique. Puis, il faut disposer du programme NSCA pour ajouter NSClient++ parmi les services. On peut alors ensuite indiquer les informations nécessaires afin d'autoriser la supervision sur la machine Windows.


**Figure 3.01 :** *Ajout du protocole SNMP pour Windows 7 et 8*

### *3.2.4 Dispositifs d'interconnexion et périphériques*

Il n'existe pas de méthode générale concernant l' implémentation de l' agent SNMP sur les dispositifs d' interconnexion (hub, switch, routeur, firewall, etc…) et les périphériques (imprimantes, multifonctions, onduleurs, postes téléphoniques, etc…), cela dépend de leurs natures et des spécifications des constructeurs. Sur certains (les routeurs Mikrotik par exemple), SNMP est déjà activé et il ne reste plus qu' à y indiquer les informations de supervision.

### *3.2.5 Test de SNMP*

Lorsque l'agent SNMP est activé sur un équipement, on peut alors le tester depuis le serveur s'il répond aux requêtes de supervision. Pour cela, on regarde les résultats que retournent les commandes SNMP. [39] Elles ont une syntaxe de la forme :

```
[nom de la commande SNMP] -v[version du protocole SNMP] -c [nom de
communauté] [adresse IP du hôte à superviser] [OID de l' objet à requêter ou
translation de l' OID défini dans la MIB]
```

```
[root@monitoring-server ~]# snmpwalk -v2c -c fihobiana 10.0.4.3 system
SNMPv2-MIB::sysDescr.0 = STRING: Hardware: Intel64 Family 6 Model 37 Stepping 2
AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 9600 Multiprocessor Free
)
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.311.1.1.3.1.1
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (647109) 1:47:51.09
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING: PC-Fifi
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 76
```

**Figure 3.02:** *Exemple de commande SNMP avec les résultats retournés*

On peut trouver tous les OID qui peuvent être supervisés en saisissant la commande suivante :

```
snmpwalk(ou snmpget) -v [version du protocole SNMP] -c [nom de communauté]
[adresse IP du hôte à superviser] ALL
```

## 3.3 Les objets

### *3.3.1 Introduction*

Lorsque la configuration de l' agent SNMP est effectuée sur l' hôte, on ajoute ce dernier dans Nagios. Pour ce faire, on doit créer des objets.

Les objets sont tous les éléments impliqués dans la logique de notification et de contrôle. Ils se trouvent par défaut dans le répertoire */usr/local/etc/nagios/objects/* de Nagios. Chacun d' eux

possède son propre fichier texte qui est d' extension «.cfg». Définir un objet consiste à donner à un élément à superviser ses propriétés qui sont des attributs affectés de valeurs.

### *3.3.2 Les templates*

Le système de templates consiste à définir des modèles d'objets dont les propriétés seront héritées par des objets réels. La méthode d'héritage permet de rendre le travail de maintenance au niveau du serveur de supervision plus facile en évitant de réécrire les attributs communs à plusieurs objets. Son utilisation est recommandée, voire nécessaire lorsqu' il s' agit de surveiller une infrastructure étendue. Trois attributs sont à tenir en compte dans la notion de templates:

- *name* : c'est la variable à laquelle on affecte une chaîne de caractère qui fait référence au nom du template ou modèle à créer;
- *register* : c'est un attribut qui est affectée d'une valeur « 1 » au cas où l' on veut enregistrer le template créé dans Nagios pour l'appliquer sur les prochains éléments à ajouter, sinon on lui affecte la valeur « 0 »;
- *use* : c'est un attribut qui permet d' indiquer le nom du template dont on veut hériter les propriétés. On le met à la dernière ligne lors de la définition des objets réels tandis que *name* et *register* sont inclus dans la configuration des templates. [40]

Les objets réels sont assignés d' attributs locaux avec des valeurs propres. Celles-ci sont prioritaires lorsque le template et l' objet contiennent les mêmes attributs. Pour chaque type d' objet, on va mentionner les variables obligatoirement à inclure dans sa configuration textuelle dans Nagios. Rappelons que chaque définition d' objet est de la forme :

```
define template{
nom          nom_template
attribut1    valeur1
attribut2    valeur2
...
register     0 #ou 1
}
define nom_objet{
attribut3   valeur3
attribut4   valeur4
...
use          nom_template
}
```

**Figure 3.03:** *Méthode de définition de template*

Chaque type d' objet peut avoir ses propres templates.

### *3.3.3 Les hôtes*

Les hôtes sont des objets centraux ayant pour critères d'être :

- des dispositifs physiques sur le réseau (serveurs, postes de travail, routeurs, commutateurs, imprimantes, etc…) ;
- identifiables (adresses IP et MAC, nom DNS) ;

Suivants sont les attributs à définir obligatoirement pour chaque hôte dans *hosts.cfg*:

e) hostname : son nom net-bios ;
f) alias : une courte description permettant l'identifier plus facilement ;
g) address : son adresse IP ou son FQDN(Fully Qualified Domain Name ou nom de domaine complet) si le service DNS est actif mais ce dernier peut poser des problèmes suite à une indisponibilité des serveurs de noms de domaine ;
h) hostgroup :le groupe auquel il appartient ;
i) parents : adresses IP des hôtes intermédiaires qui se trouvent entre le serveur de monitoring et l' hôte à définir qui est alors considéré comme hôte enfant ;
j) children : adresses IP des hôtes qui se trouvent derrière l' hôte en question qui devient un hôte parent.

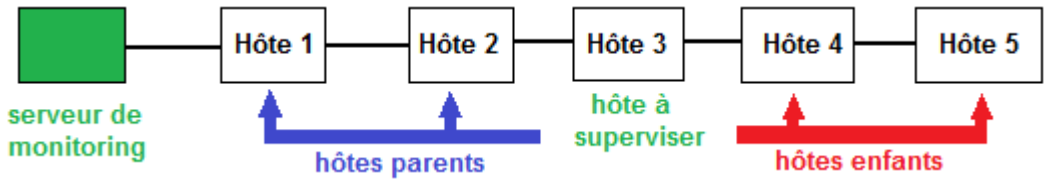


**Figure 3.04:** *Schéma expliquant la relation parent/enfant entre les hôtes*

L' exécution d' une commande «*traceroute* » pointant sur l' adresse IP d' un hôte pourrait déterminer les adresses IP de ses parents. Cette dépendance entre les hôtes est à définir dans le fichier */usr/local/etc/nagios/objects/dependencies.cfg* pour être prises en compte par Nagios. Ainsi, on peut éviter l'encombrement des notifications dans la boite de réception. Par exemple, si un switch (parent) tombe en panne, les machines (enfants) qui y sont connectées deviennent par la suite injoignables et de ce fait, seule le disfonctionnement du switch est signalé par mail à l' administrateur.

### *3.3.4 Les services*

Les services sont les plus objets les plus importants car réellement, ce sont les éléments à superviser. Ils sont associés aux hôtes et peuvent être :

- des caractéristiques d' hôte : la charge CPU, l' occupation de mémoire, les interfaces, uptime/downtime, nombre d' utilisateurs connectés, temps de réponse, etc…
- des services réseaux fournis par les hôtes de type serveurs: HTTP, FTP, DNS, DHCP, etc…

Les principaux attributs à inclure dans la définition de chaque service dans *services.cfg*:

- hostname :le nom de l' hôte auquel il appartient ;
- service_description : un nom ou une brève description d' un l' élément à superviser sur l' hôte. Chaque service doit avoir son propre nom ou description ;
- check_command : le plugin que Nagios exécutera pour déterminer son état (état du service) ;
- notification_option : définit à quel ou à quels états Nagios doit envoyer les notifications ;
- contacts : noms des personnes qui reçoivent les notifications de Nagios ;
- action_url : pour un service dont la performance est illustrée par Cacti, on indique le chemin */etc/nagios/share/cacti_nagios?ip=[adresse_ip_de_l'hôte]*.

### *3.3.5 Les contacts*

Dans le contexte de supervision, un contact est une personne impliquée aux procès de notification. Pour chaque définition de contact dans *contacts.cfg*, les attributs suivants sont obligatoires:

- contact_name : le nom de la personne qui reçoit les notifications par mail de Nagios ;
- host_notification_command : le nom du script qui ordonne à Nagios d' envoyer les alertes ;
- email : l' adresse mail vers laquelle les notifications .

### *3.3.6 Les « timeperiods »*

Au cas où l' on souhaite que le réseau soit supervisé en permanence, c'est-à-dire 7jours/7 et 24heures/24, on affecte aux attributs « check_period » et « notification_period » la valeur " 24X7 " lors de la définition des hôtes.

Si l' on veut définir des périodes de supervision afin de fixer à quels moments le serveur de monitoring est lancé et pendant quelles périodes les notifications sont envoyées aux responsables, on édite *timeperiods.cfg*. Ainsi, on ajoute l' attribut « timeperiod » dans la définition d' hôte et qui

est affecté de la même valeur que « timeperiod_name ». Pour mieux expliquer, on va montrer ci-dessous un exemple.

```
define timeperiod{
       timeperiod_name         periodes_de_surveillance
       monday                  08:00-12:00,14:00-22:00
       tuesday                 09:00-13:00,15:00-22:00
       saturday                10:00-22:00
       }
```

### *3.3.7 Les groupes d'hôtes*

Ce type d'objet, définit dans *hostgroups.cfg* permet de rassembler des hôtes selon un rôle, une application ou autres éléments communs. Il est obligatoire qu'un hôte appartienne à un groupe d' hôtes pour que Nagios le prenne en compte.

### *3.3.8 Les groupes de services*

C'est la partie de configuration la moins utilisée car sa gestion est assez lourde mais peut être utile sur des configurations plus complexes ou pour regrouper des services comme les clusters applicatifs.

### *3.3.9 Les groupes de contacts*

Lorsque le monitoring est sous la responsabilité de plusieurs personnes, cet objet doit être configuré dans *contactgroups.cfg* pour la gestion des alertes.

Pour inclure un objet (hôte, service ou contact défini au préalable) dans son groupe, on insère son nom dans la variable « *members* » dans le fichier de configuration de son groupe correspondant. [41]

### *3.3.10 Les commandes*

3.3.10.1 Les plugins

Les plugins constituent une notion importante dans le domaine de supervision. Ce sont des programmes exécutables ou des scripts qui peuvent être lancés indépendamment par ligne de commande sans utilisation de logiciel. Mais ils sont intégrés dans Nagios pour que celui-ci puisse récupérer le statut des hôtes ou des services en exécutant ces plugins régulièrement. Ainsi, ils sont localisés dans le répertoire */usr/local/nagios/libexec/*. Les plugins installés par défaut lors de l' installation de Nagios ne permettent pas de tout superviser mais on peut en créer manuellement sous n'importe quel langage de programmation, puisque Nagios est extensible.

Un plugin doit contenir :

- des codes de retour pour correspondre aux résultats des tests et qui prennent des valeurs distinctes de 0 à 3;
- des textes de retour qui indique des informations de performance destinées à l' utilisateur

Ce Tableau 3.01 montre les états possibles que peut avoir un service lors du lancement des plugins.

| Valeur numérique | Statut | Description |
|---|---|---|
| 0 | OK | Le plugin a été capable de vérifier le service et celui-ci semble fonctionner correctement. |
| 1 | WARNING | Le plugin a été capable de vérifier le service mais celui-ci semble au dessus de certains seuils d'alerte ou il ne fonctionne pas correctement. |
| 2 | CRITICAL | Le plugin a détecté que le service ne fonctionnait pas ou qu'il était au-dessus d'un seuil critique. |
| 3 | UNKNOWN | Impossible de déterminer l' état |

**Tableau 3.01 :** *Les différents états que peut prendre un objet*

Nagios donne un état appelé « PENDING » pour un service ou un hôte fraîchement configuré mais qui n'a pas encore été contrôlé par l'ordonnanceur.

3.3.10.2 Les commandes et les macros

Pour utiliser un plugin pour pouvoir superviser un service avec Nagios, on crée une commande en éditant le fichier */nagios/objects/commands.cfg* où l' on affecte le nom du plugin à l' attribut « command_name » et c'est ce même nom qu' on indique à l' attribut « check_command » lors de la déclaration de service dans */nagios/objects/services.cfg*. Le deuxième attribut à ajouter lors de la création de commande est « command_line ». On y indique le plugin et les macros associés. Les macros sont des arguments qui servent à fixer les seuils permettant à Nagios de classer l' état d' un service à partir du résultat obtenu. Prenons un exemple : on va affecter à l' attribut « command_line » la valeur suivante : "/usr/local/nagios/libexec/check_ping -w 300.0,30% -c 900.0,30%" (ou bien "check_ping!300.0,30%!900.0,90% " sur « check_command » lors de la définition de service au cas où la commande n'est pas privée). Alors le service aura un état :

- OK si le temps de réponse moyen (RTA) est inférieur à 300 millisecondes ;
- WARNING si le RTA est entre 300 et 900 millisecondes ;
- CRITICAL si le RTA est plus élevé 900 millisecondes ou que le nombre de paquets perdus est supérieur ou égal à 90% ;
- UNKNOWN si aucune réponse ne peut être délivrée ;

En effet, Nagios se sert des réponses aux requêtes ping pour définir l' état des hôtes : UP(ou allumé) si OK, « Down » (ou éteint) si CRITICAL, « Unreachable » (ou injoignable) si UNKNOWN , sinon PENDING. Si un hôte subit un arrêt, tous ses services sont par suite en état CRITICAL.

3.3.10.3 Paramétrage de l' envoi d' alerte

Le réglage de l' envoi d' alerte se fait lors de la définition d' un service. On peut affecter à l' attribut *notification_option* les valeurs : **r** (recovery ou OK) , **w** (WARNING), **c** (CRITICAL) et **u** (UNKNOWN) sinon **n** (none) pour ne pas recevoir de notification.

3.3.10.4 Les escalades de vérification

Quand on se trouve dans une période d'activité d'un service, plusieurs paramètres rentrent en jeu pour fixer la durée et la fréquence des vérifications (checks) du service en question. Quand un objet est OK, il est vérifié toutes les *check_interval* minutes. S'il passe en état WARNING, CRITICAL ou UNKNOWN, il est alors vérifié *max_check_attempts* fois à un intervalle de *retry_interval* minutes. Si l'état de l'objet n'est pas revenu à OK au bout des *max_check_attempts* essais, l'intervalle de vérification redevient en *check_interval* minutes.

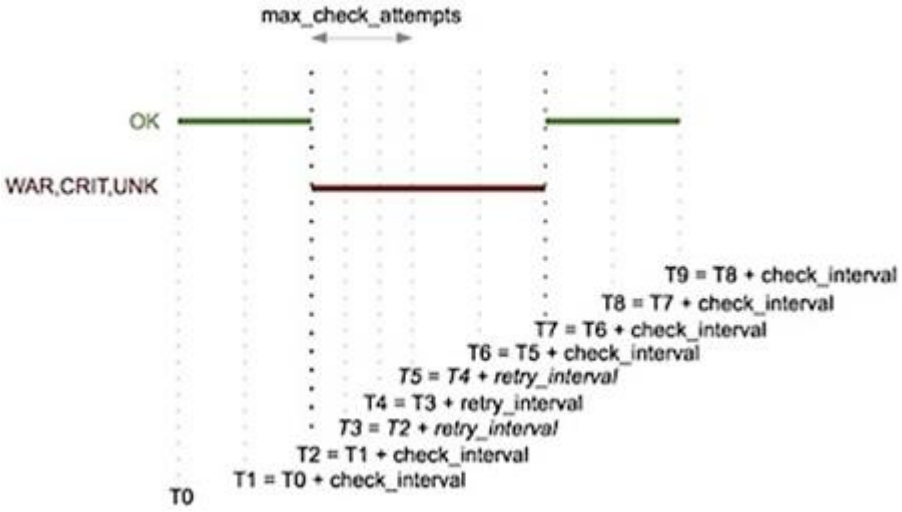


**Figure 3.05:** *Les différents intervalles*

Les durées sont données en minutes et la valeur minimum est de 1,5 donc on peut dire que les résultats sont donnés en quasi temps réel.

3.3.10.5 Les escalades de notifications

Il existe des variables permettant de fixer comment les notifications sont remontés aux administrateurs.

La première variable est *first_notification_delay*. Elle permet de définir le temps (en secondes) que Nagios doit attendre avant d'envoyer une notification quand un objet passe d'un état OK à un état WARNING, CRITICAL ou UNKNOWN. Une valeur 0 permet d'envoyer la notification dès ce changement d'état.

La variable *notification_interval* permet, en cas de problème sur un objet, de fixer l'intervalle de temps (en minutes) entre deux notifications. Pour que Nagios n'envoie qu'une seule fois une alerte, il faut fixer cette variable à 0.

Les définitions d'escalades pour un service sont optionnelles, Nagios les fixe par défaut. [42]

## 3.4 Création de graphes dans Cacti

### *3.4.1 Collection des données de performance vers la base de données RRD de Cacti*

Initialement, les plugins de Nagios renvoient une seule ligne de texte indiquant l'état d'une quelconque donnée mesurable. Par exemple, « check_ping » peut renvoyer une ligne de texte comme suit:

```
PING ok - Packet loss = 0%, RTA = 0.80 ms
```

Dans le fichier */etc/nagios/nagios.cfg*, on doit ordonner à Nagios de fournir des données de performance.

```
process_performance_data=1
service_perfdata_file=/var/log/nagios/perfdata.pipe
service_perfdata_file_template=DATATYPE::SERVICEPERFDATA\tTIMET::$TIMET$\tHOSTNA
ME::$HOSTNAME$\tSERVICEDESC::$SERVICEDESC$\t$SERVICESTATEID$\t$SERVICEOUTPUT$|$s
SERVICEPERFDATA$|
service_perfdata_command=process-sevice-perfdata
```

**Figure 3.06 :** *Les lignes d' instruction permettant d'avoir les données de performances*

La valeur 1 signifie que Nagios traitera les données liées aux performances des contrôles de service. Ces données sont écrites dans le fichier « perfdata.pipe » via la commande « process-service-perfdata». Le script « perf2rrd » convertit ces données en fichiers .rrd et les transportent dans la base de données RRD de Cacti.

Puis, on doit définir la commande « process-service-perfdata» dans le fichier *commands.cfg* pour être exécutée par Nagios. [43]

```
define command {
command_name    process-service-perfdata
command_line    /bin/mv /var/log/nagios/service-perfdata /var/log/nagios.$TIMET$
}
```

**Figure 3.07:** *Définition de la commande « process-service-perfdata »*

Les données de performance sont définies par Nagios par toutes les information présentes après le « | » (ou « pipe ») résultant de l' exécution des plugins. Le texte de retour montré précédemment devient alors :

```
PING ok - Packet loss = 0%, RTA = 0.80 ms | percent_packet_loss=0, rta=0.80
```

### *3.4.2 Création de graphe*

Après avoir exécuté le script « nagios2cacti.pl », on doit faire la configuration de « n2rrd.conf » (Nagios to Round Robin Database) pour indiquer :

- les chemins d' accès à la racine de Cacti, à celle de Nagios et vers la méthode de transfert nommée. Pour cette dernière, on utilise le mécanisme de pipe.
- les informations sur la base de données utilisée par les perfdb (nom,utilisateur, mot de passe, nom du serveur)
- le nom des éléments dont on souhaite visualiser les performances graphiquement avec Cacti

Puisqu' on a réalisé une intégration de Nagios dans Cacti, il y a une différence sur la méthode de création de graphes par rapport avec celle à suivre si Cacti fonctionne d'une manière indépendante. Cela consiste à créer une source de données d' un service et à décrire l' aspect de son graphe. Cependant, la définition des templates, au même sens que dans Nagios , est obligatoire et se fait dans l' interface web de Cacti. On doit alors créer trois types de template:

- « Data Template » ou modèle de sources de données permettant de réutiliser les paramètres de données dont le plus important est le: « Data Input Method » ou méthode d' acquisition des données par Cacti**.** Pour notre cas**,** cela est assuré par le script « nagios2cacti.pl ». Avec ce dernier, pour créer une source de données pour un service dont on souhaite illustrer les performances, on insère le nom de l' hôte à qui il appartient et sa description dans le champ du paramètre « Input String ». Chaque service ajouté possède alors un fichier RRD devant être associé au moins à un RRA (Round Robin Archive) qui correspond à un cycle de conservation de données (jour, semaine, mois ou année). Le paramètre « Step » permet de fixer l' intervalle de temps imposé à Cacti pour faire les mises à jours des affichages grâce au script « poller.php ».
- « Graph Template » ou modèle de graphisme qui contient les caractéristiques communes des graphes que l' on veut avoir : format, couleurs des courbes, hauteur et largeur (en pixel) du plan, … Ce type de template est créé uniquement une fois.
- « Host Template » ou modèle d' hôte permet d' associer facelement un modèle de données à un graphe.

Les premiers graphes ne s' affichent qu' après cinq minutes. En effet, c'est l' intervalle de temps minimum supporté par Cacti. [44]

On peut ajouter tous les services de Nagios dans Cacti, mais l' utilité de celui-ci est surtout appréciable dans la surveillance des services comme la consommation en bande passante, la latence, le taux d' erreur, etc… permettant d' évaluer la performance du réseau à l' aide des graphes continus.

### 3.5 Création de carte de supervision avec NagVis

Pour créer sa propre cartographie, les images à afficher doivent être de format PNG. Il faut en même temps savoir que l' architecture du répertoire de NagVis se compose de la façon suivante :

k) *nagvis/etc/maps/* : localisation de la configuration des cartes
l) *nagvis/etc/nagvis.ini.php* : fichier de configuration générale de NagVis
m) *nagvis/nagvis/images/* :localisation des formes, les icônes et les images de fond pour les cartes
n) *nagvis/nagvis/gadgets/* : localisation des gadgets
o) *nagvis/nagvis/templates/* :localisation des templates pour l' interface web

Dans l' interface de NagVis, l' upload et la manipulation des images se font à l' aide des options « Manage Background » et « Manage Map ». Une fois que la carte est créée, on peut ensuite l' associer avec la supervision. Pour ce faire, on va dans la section *Cartes >Add Icon >Host*. A chaque sélection d'un hôte, on obtient son icône reflétant son état mais qui est encore à déplacer à sa proximité. Les icônes changent dynamiquement suivant les états retournés par Nagios. Tout cela est commandé par le plugin « nagvis.ini.php » qui est à éditer pour préciser les icônes à utiliser et leur emplacement (*nagvis/nagvis/images*). On peut également faire la même chose avec les services mais cela peut encombrer la carte. [45]

Chaque ajout ou modification des fichiers doit être suivi d' un redémarrage de service par ligne de commande.

### 3.6 Conclusion

Ce chapitre explique comment adapter la solution de monitoring élaborée à un réseau informatique pour la faire fonctionner convenablement. Cela se fait par un passage à une série de configurations.

La tâche est répartie sur les hôtes distants et le serveur qui héberge la solution. Elle doit être bien entretenue pour que le projet fonctionne. La complexité de la mise en œuvre dépend de l'étendue du réseau et du nombre de services à monitorer.

Les supports de transmission ne sont pas pris en compte. Cependant, si un hôte devient injoignable, cela peut être indiqué par un changement de couleur sur le tableau technique ou par notification selon la commande définie pour cet hôte.

# CHAPITRE 4
# SIMULATION DE MONITORING

## 4.1 Introduction

Avant de mettre en place un projet sur un réseau réel, il est obligatoire de passer à une phase de simulation. Cela consiste à reproduire l'architecture du réseau à monitorer et de faire une application à l'aide des logiciels. Pour notre cas, on va pratiquer avec VirtualBox et GNS3.

Lorsque les résultats affichés sur les pages web du serveur correspondent aux expectations selon les manipulations, cela signifie que la solution marche et on pourra déployer le système de monitoring sur le un réseau réel. Les configurations sur les hôtes réels seront les mêmes.

## 4.2 Les machines virtuelles

VirtualBox est un hyperviseur permettant de créer et de gérer des machines virtuelles.

*Définition 4.01 :*

La *virtualisation* est un mécanisme informatique qui consiste à faire fonctionner sur une même machine réelle, plusieurs hôtes fonctionnant sous des systèmes d'exploitation identiques ou différents. [46]

### *4.2.1 Description*

Le serveur de monitoring élaboré sous CentOS Server a été la première instance de machine virtuelle créée dans VirtualBox. D'autres machines supplémentaires seront créées et elles auront chacune son propre rôle dans l'architecture qu'on va concevoir ultérieurement avec GNS3. Pour les installer, il faut d'abord disposer de leurs fichiers de disque optique virtuel.

En effet, on va concevoir un réseau informatique similaire à celui d'une entreprise et qui est muni d'une liaison à internet. Ceci est réparti en trois zones : le LAN, la DMZ et la partie WAN (Figure 4.02). On va créer trois postes de travail obtenu à partir d' un double clonage d' une machine virtuelle fonctionnant sous Windows 7. Ils seront différenciés par leurs paramétrages réseaux. L' un fera office d' un poste d' un utilisateur externe connecté à internet. Les deux autres seront placés dans le réseau privé (LAN) avec deux serveurs : celui qui opère pour le monitoring et un deuxième qui fonctionne sous Debian. Les machines et les serveurs seront séparés logiquement par la création de deux VLANs à l' aide d' un commutateur Cisco IOUvL2 monté par VirtualBox. Dans la zone DMZ, on va placer une machine Windows Server 2008 R2 où on va héberger localement un site

web accessible par le public via internet. La réalisation de cette section s'est faite par l' outil IIS (Internet Information Services).


**Figure 4.01 :** *Aperçu des hôtes virtuels créés avec VirtualBox*

### *4.2.2 Gestion réseau*

Pour chaque machine, dans la configuration réseau dans VirtualBox : la « Carte 1 » est configurée en mode « Réseau Privé Hôte » et la « Carte 2 » en mode « Pilote Générique ». [47] Puisque les machines sont réparties dans quatre réseaux différents, il a fallu créer quatre cartes virtuelles « VirtualBox Host-Only Network ». (Annexe A2.02)

### *4.2.3 Création d' un site WordPress*

Pour mettre en place un centre de communication dans le LAN, on va créer et héberger un site WordPress sur la machine Ubuntu à l' aide d'un serveur LAMP.


**Figure 4.02 :** *Aperçu du site WordPress créé*

## 4.3 Création de la topologie réseau

GNS3 (Graphical Network Simulator) est un logiciel qui simule graphiquement des réseaux, allant des plus simples aux plus complexes, dans un environnement virtuel mais avec une vision assez proche de la réalité. Sa manipulation suit les mêmes procédures à suivre pendant l' implémentation. Il est à la fois un simulateur et un émulateur grâce à l'incorporation d'autres outils dans son architecture.

On va commencer par la création d'un nouveau projet puis la mise en place se fait sur l'espace de travail. Une fois que tous les équipements y sont importés, on les relie par câbles puis on les démarre un par un pour les configurer afin de rendre la topologie fonctionnelle.

### *4.3.1 L' accès à Internet*

Puisque l' accès à Internet est nécessaire pour la réception des mails d' alerte ainsi que pour tester l' accessibilité au site hébergé sous Windows Server, on va créer deux cartes réseaux virtuelles de bouclage Microsoft dans la machine réelle. On va ensuite modifier les propriétés de la carte réseau de cette dernière pour activer le partage Wi-Fi. Puis, on importe sur l' espace de travail un nuage dont deux interfaces sont connectées aux deux cartes réseaux virtuelles fraichement créées. [49] L'incorporation de VirtualBox dans l'architecture de GNS3 permet d'ajouter les hôtes virtuels. GNS3 utilise des images IOS réelles et des émulateurs comme Dynamips et Qemu pour simuler les autres équipements.

### *4.3.2 Intérêt du firewall*

Pour notre topologie, on a installé un pare-feu Cisco ASA **(**Adaptive Security Appliance) de série 5500 pour constituer une ligne de défense contre les attaques provenant de l' extérieur en isolant la DMZ du réseau privé. On a activé trois de ses interfaces dont à chacune est attribuée un nom typique :

- *outside*: qui est attribuée un niveau de sécurité 0. Elle qui est reliée au nuage faisant référence à internet à l' aide d' un commutateur. On a joint une des machines sous Windows 7 à l' aide d'un routeur Cisco 3660 à ce nuage. Ceci est pour tester l' accès au serveur web de la DMZ depuis l'extérieur. Le monitoring ne sera pas appliqué aux équipements de la partie WAN.
- *DMZ* : qui est reliée directement à la machine Windows Server. Son niveau de sécurité est par défaut 0 mais on l' a fixé à 50. Par défaut, les trafics passant d' un niveau de sécurité

inférieur vers un autre plus élevé sont refusés mais cela peut être ignoré par une liste de contrôle d' accès (ACL) appliquée à l' interface de niveau de sécurité inférieur.

– *inside* : connectée à un routeur Cisco 3660 qui est par la suite lié au switch CiscoIOUvL2 pour assurer le routage inter-VLAN à l' aide d' une liaison « trunk ». Le serveur de monitoring et la machine Ubuntu hébergeant le site Wordpress appartiennent au VLAN 1, les deux machines sous Windows 7 au VLAN 2. L' attribution de nom « inside » permet d' assigner un niveau de sécurité maximale égale à 100.

Le pare-feu jouant aussi le rôle de passerelle entre le réseau de l'entreprise et internet, à son niveau sont configurées les séries de NAT/PAT pour la translation d' adresses et la redirection de ports. Cela commence par la création d' objets. Pour permettre l' accès « outbound », on configure le NAT dynamique afin que les clients en interne passent depuis leurs interfaces respectives vers l' interface « outside » du pare-feu ASA. Quant à l' accès « inbound » vers la zone DMZ, on doit assigner une deuxième adresse, qui est publique au serveur web et on configure un NAT statique. Pour cette tâche, le pare-feu ASA translatera le port 80 du serveur web (192.168.2.2) afin de ressembler à l' adresse publique au port 80 sur l' interface « outside ». [50]

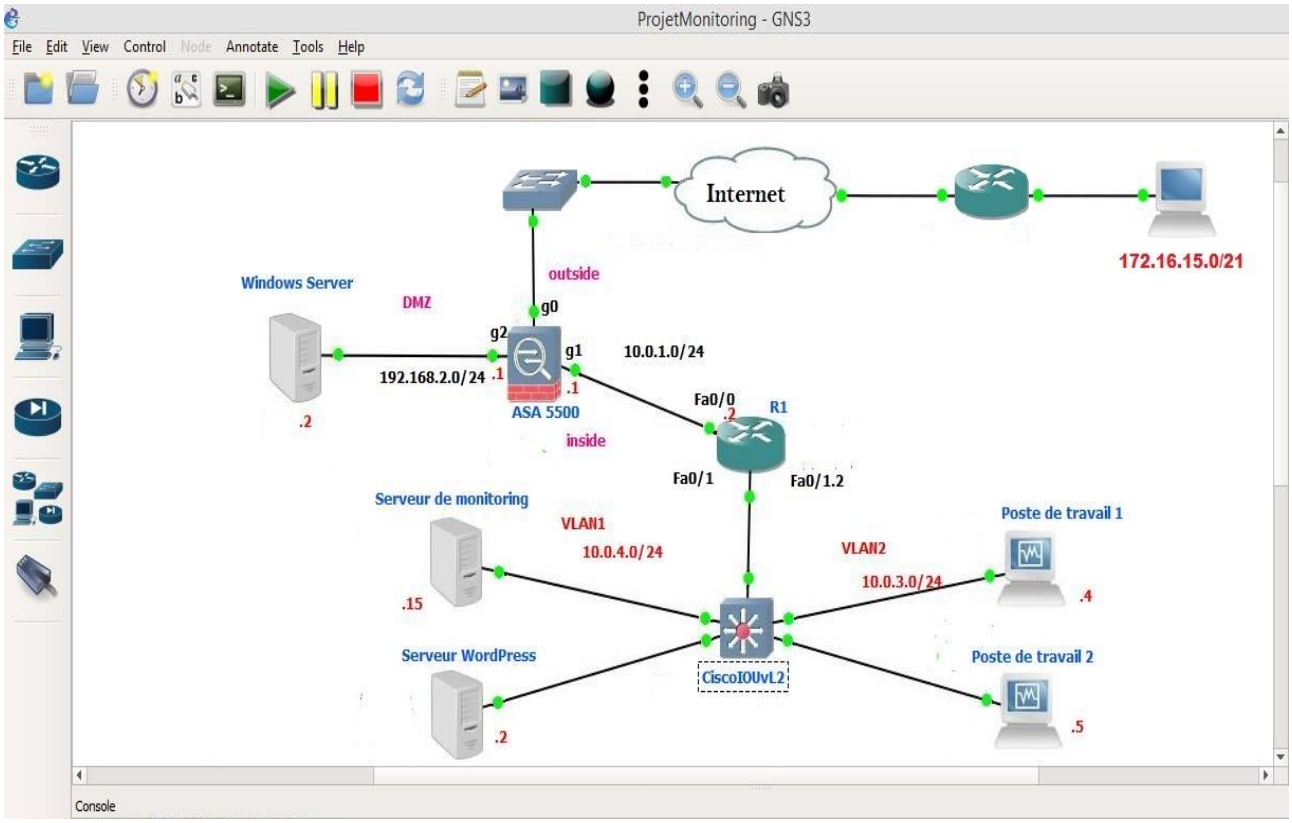


**Figure 4.03 :** *Aperçu de la topologie réalisée*

## 4.4 Configuration pour la supervision

Dans le projet GNS3 précédemment créé, les hôtes à superviser sont au nombre de huit : ce sont la machine de la DMZ, le pare-feu ASA et les six équipements dans le LAN.

### *4.4.1 Au niveau des hôtes*

La méthode de configuration SNMP au niveau des hôtes enfants, qui sont les machines incluant postes de travail et serveurs, est déjà étalée dans le chapitre précédent. Pour le partage du paquet NSCA entre les machines Windows, on a dû activer VirtualBox Guest. Dans l' architecture, les équipements Cisco sont les hôtes parents. [51] Pour chacun d' eux, on doit revenir sur le terminal pour autoriser la supervision active (polling) par Nagios et saisir la même syntaxe :

```
snmp-server host <nameif> <IP> poll [community <KEY>] [version <1|2c|3>]
```

Chaque dispositif d'interconnexion est identifié par l'adresse IP de l'interface du côté du serveur de monitoring (`<nameif>`). Ainsi, pour le firewall :

```
firewallASA5500(config)#snmp-server host 10.0.1.1 10.0.4.15 poll fihobiana
version 2c
```

Pour le router :

```
R1(config)#snmp-server host 10.0.2.1 10.0.4.15 poll fihobiana version 2c
```

Pour le cas du commutateur, l' interface VLAN1 est considérée par défaut. [52]

```
SW(config)#snmp-server host 10.0.4.254 10.0.4.15 poll fihobiana version 2c
```

### *4.4.2 Au niveau du serveur Nagios*

Lorsque l' agent SNMP est activé au niveau des hôtes dans leurs systèmes, on passe à la définition des objets dans Nagios. Comme tous les hôtes doivent appartenir à un groupe alors que les trois matériels Cisco se diffèrent par leurs natures, on doit alors créer un groupe pour chacun d' eux.

Voici quelques détails des services spécifiques supervisés qui nécessitent la création de nouvelles commandes:

– L’état des ports : (qui peut être différent de celui des hôtes) est référencié par l’ objet « ifOperStatus » dans la table MIB du matériel en question. Comme les dispositifs d’ interconnexion sont dotés de plusieurs ports, on doit associer par l’ argument « -n » les noms de ceux qui sont actifs au plugin *check_snmp_int* qui a été téléchargé et étudié avant de l’ utiliser avec Nagios. On peut connaitre la liste des interfaces par lancement direct de ce plugin par ligne de commande en ajoutant la syntaxe « -n zzzz -v ». Cela est nécessaire puisqu’ on doit utiliser des noms définis dans la MIB. L'argument « -r 1 » ordonne de retourner un état OK si la valeur de « ifOperStatus » est 1 dans la MIB (1 indique un état HAUT signifiant que le port est opérationnel), sinon CRITICAL. [53]

```
# verification des interfaces reseaux
define command{
command_name     check_snmp_int
command_line      /usr/local/nagios/libexec/check_snmp_int    -h  10.0.1.1 -c fi
hobiana -n "gigabitethernet0/0" -r 1
```

**Figure 4.04:** *Définition de commande pour l’ état de l’ interface « outside » du firewall*

– WordPress : Pour veiller à ce qu’ un site web soit toujours disponible, le service principal à surveiller est HTTP. Il faut assurer qu’ il retourne des réponses aux requêtes, et ce, dans une période de temps raisonnable. Mais cela dépend aussi du serveur de base de données, des ressources de la machine physique pour répondre correctement aux requêtes, du nombre d’ utilisateurs qui se connectent et éventuellement du serveur de nom de domaine puisque les utilisateurs ne sont pas toujours censés connaître l’ adresse IP du serveur web. On va alors utiliser le plugin standard de Nagios «check_http » et les seuils de temps de réponses sont donnés en secondes. [54]

```
#supervision du site wordpress
define command{
command_name    chech_wordpress_site
command_line    /usr/local/nagios/libexec/check_http -H 10.0.4.2 -p 80   -w 20 -
c 60
}
```

**Figure 4.05:** *Définition de commande pour vérifier la disponibilité du site WordPress*

– La mesure des trafics entrants et sortants : En pratique, au cas où la vitesse de la connexion internet baisse, on peut identifier la machine de l’utilisateur responsable. Sinon, il s’agirait d’ un programme malveillant qui s’arroge une vaste partie de la bande passante pouvant être

une des causes de congestion du réseau. Pour ce faire, on va écrire un plugin en shell appelé « check_traffic » à inclure dans « check_nt ». Dans notre projet, on va le tester sur la machine Windows d'adresse IP 10.0.3.4 du VLAN2. En effet, il qui sera associé au port de connexion 12489 et au compteur de performances des systèmes Windows, appelé « COUNTER ». Celui-ci doit opérer sur un objet, désigné par le paramètre « -l », composé par le nom de la carte réseau virtuelle de la machine et le type de flux à mesurer. On précise ensuite les seuils des états à l'aide des paramètres « -w » et « -c ». [55]

```
#supervision_trafic_entrant
define command{
command_name    check_traffic_in
command_line    /usr/var/nagios/libexec/check_nt  -H 10.0.3.4 -p 12489 -v COUNTE
R -o KB -l "\VirtualBox Host-Only Ethernet Adapter #2 \Bytes Received/sec" -w 50
-c 80
}
#supervision_trafic_sortant
define command{
command_name   check_taffic_out
command_line    /usr/var/nagios/libexec/check_nt  -H 10.0.3.4 -p 12489 -v COUNTE
R -o KB -l "\VirtualBox Host-Only Ethernet Adapter #2 \Bytes Sent/sec" -w 50 -c
80
}
```

**Figure 4.06:** *Définition des commandes permettant de mesurer les trafics entrants et sortants*

## 4.5 Visualisation des interfaces web

### *4.5.1 Paramétrage de la machine réelle*

Lorsque tout est mis en place, on fait l' initialisation de tous les services pour que les configurations soient prises en compte puis on démarre le projet dans GNS3. On a dû configurer l'adresse IP de la carte réseau virtuelle correspondant au même réseau que le serveur de monitoring puisque celui-ci n'est pas muni d' interface graphique. On va alors se servir de la machine réelle. Pour cela, on modifie les propriétés du Protocole Internet version 4 (TCP/IPv4) dans les paramètres de la carte réseau créée par VirtualBox (VirtualBox Host-Only Network).

### *4.5.2 L' interface web de Nagios*

Après le paramétrage de la machine réelle, on va ouvrir son navigateur Web Mozilla Firefox pour voir les résultats. On saisit alors l' adresse « http://10.0.4.15/nagios » dans la barre URL. La bannière d' authentification propose alors de taper le login et le mot de passe. Sur la partie principale, on trouve des tableaux affichant les informations de supervision(Figure 4.06). En haut, on a deux tableaux statistiques qui donnent le nombre d' hôtes et de services pour chaque état.

En bas, on a le tableau technique à plusieurs colonnes montrant la liste des hôtes. Pour chacun d' eux, tous les services ajoutés avec leurs statuts sont indiqués dans les deux colonnes à côté. Le reste du tableau est complété par des détails dont la dernière heure de vérification, la duration totale, le nombre de tentatives et les informations sur les statuts retournés. La mise à jour est faite toutes les 90 secondes.

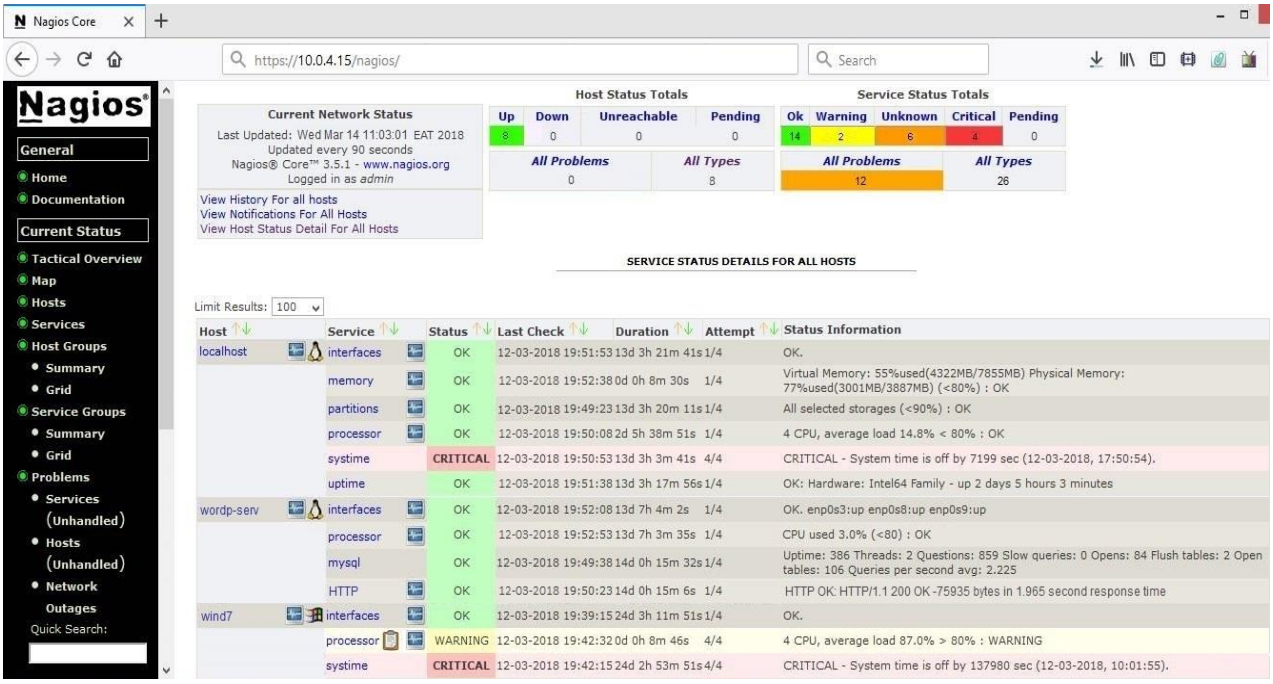


**Figure 4.07:** *L' interface web de Nagios*

Il faut glisser la barre de défilement vers le bas pour voir le reste des éléments supervisés. La Figure 4.07 donnée ci-dessous présente les informations sur les services des équipements Cisco ajoutés. (Annexe A4.04)

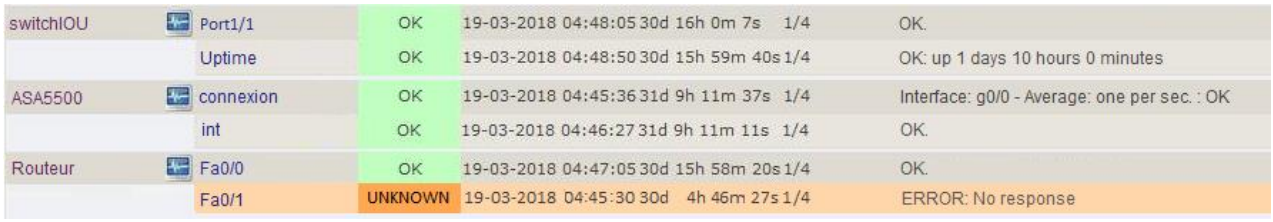


**Figure 4.08:** *Détails des résultats de contrôle des équipements Cisco*

### *4.5.3 L' interface web de Thruk*

On va maintenant consulter la page de Thruk en remplaçant « nagios » par « thruk » dans la barre d' adresse. Dans la section « Panorama View », trois widgets sont affichés :

- « Site Status » montre la liste des backends, leurs versions et le temps écoulés depuis la connexion à Thruk
- « Hosts » et « Services » présentent respectivement les sections des états des hôtes et des services dans deux diagrammes circulaires. Chacune des couleurs correspond à un état tel que :

– Vert pour UP et OK ;
– Rouge pour DOWN et CRITICAL;
– Jaune pour WARNING ;
– Orange pour UNREACHABLE et UNKNOWN.
– Violet Pour PENDING

Quatre couleurs sont toutes présentes dans le diagramme des services. La présence des indications d'état WARNING et CRITICAL signifient que les seuils définis dans les commandes sont atteints. Cela est surtout lié à l'insuffisance des ressources allouées aux hôtes du fait qu'on doit partager celle de la machine réelle car il s'agit d'une simulation.

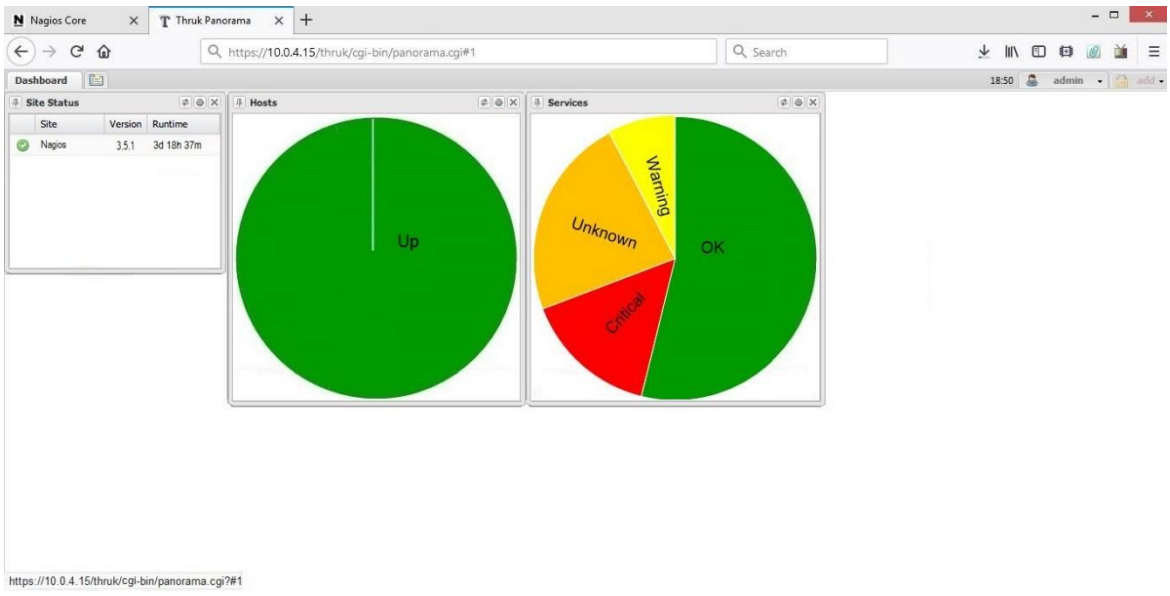


**Figure 4.09:** *L'interface « panorama view » de Thruk*

### *4.5.4 L' interface de Cacti*

On a ajouté quelques graphes dans Cacti mais on va expliquer les graphes des flux de données entrants et sortants. Sur chacun d' eux, la section en vert montre le trafic sortant et la ligne colorée en bleu montre le trafic entrant. Cacti met à jour ces graphes suivant un intervalle de temps de 5 minutes. En se connectant à internet avec cette machine, on peut avoir ce type de graphe avec Cacti. [56]

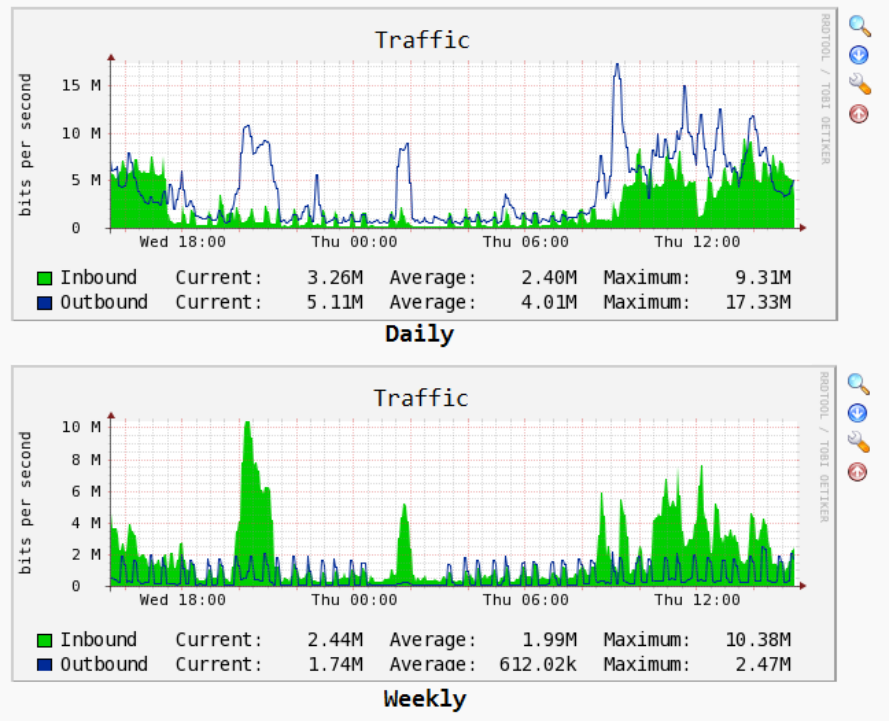


**Figure 4.10:** *Visualisation des trafics sortants et entrants*

### *4.5.5 L' interface web de NagVis*

Sur la page de NagVis, on a importé une image de fond qui représente la partie monitorée dans l'architecture réseau construite dans GNS3. Les noms des hôtes et les adressages ont été masqués (couleur des textes changée en blanc dans l' option « Edit text » de GNS3) afin de mettre en évidence la cartographie de la supervision. On a recadré l' image obtenue par capture d' écran pour ne représenter que les éléments à superviser. On a importé les quatre icônes indiquer les états des hôtes dans la mémoire du serveur (repertoire : */nagvis/nagvis/images/*) par activation de VirtualBox Guest. On édite ensuite le fichier *nagvis.ini.php* pour que ces icônes soient pris en compte par NagVis.

Les hôtes étant tous en fonction et la supervision bien menée par Nagios, on obtient effectivement huit icônes correspondantes à l'état UP. Il suffit ensuite de les déplacer et de les redimensionner pour rendre la carte plus présentable.

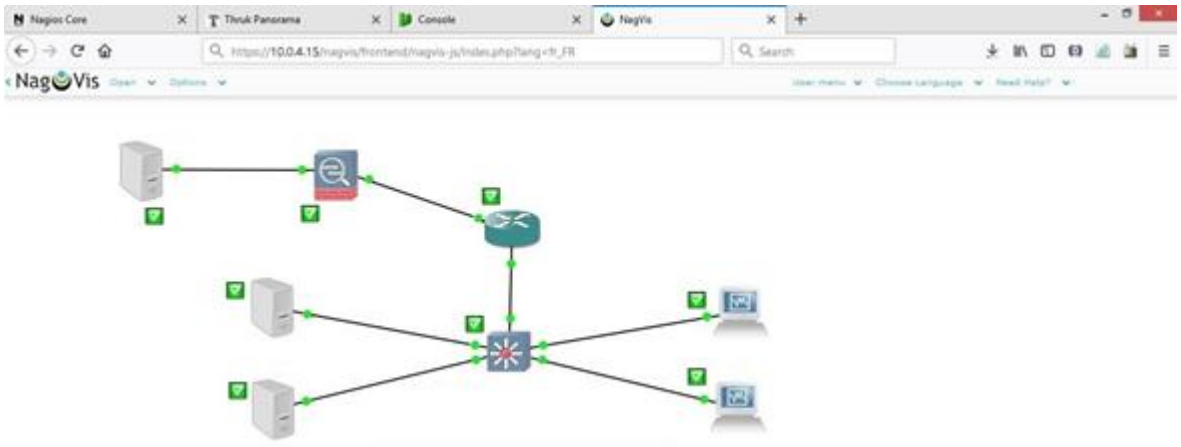


**Figure 4.11 :** *La cartographie de la supervision créée avec NagVis*

### 4.6 Création d' un incident pour tester la réaction

Afin de s' assurer que le monitoring fonctionne parfaitement, on va créer un incident et vérifier si cela va donner des changements au niveau des interfaces web. On va rompre la liaison avec Windows Server du DMZ pour qu' elle devienne injoignable. Cela revient au même si on l'éteint via VirtualBox.

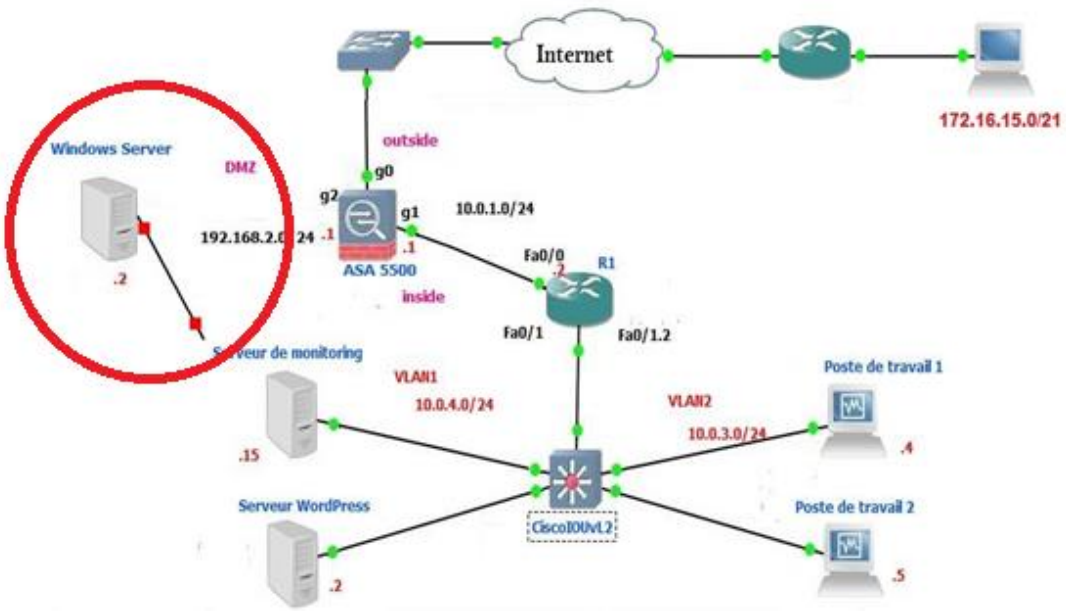


**Figure 4.12:** *Rupture forcée du lien avec la machine Windows Server*

### *4.6.1 Vérification de la réaction de Thruk et de NagVis*

Maintenant, on va revenir sur l' interface de Thruk pour vérifier que Nagios a bien réagi vis-à-vis de cette rupture de liaison. On remarque alors des changements au niveau des diagrammes par rapport à ce qui s'affichait sur la Figure 4.09. Dans le premier, c'est-à-dire, celui qui correspond aux hôtes, une partie proportionnelle à un hôte a basculé en rouge, c'est celle de Windows Server dont tous les services deviennent par suite en état CRITICAL.


**Figure 4.13:** *Visualisation des modifications au niveau des diagrammes circulaires*

Sur la carte de la page de NagVis, l'icône associé à la machine Windows Server éteinte a été changé. Comme il est synchronisé avec Nagios, la durée de remise à jour de l' interface web de NagVis est de 90 secondes.

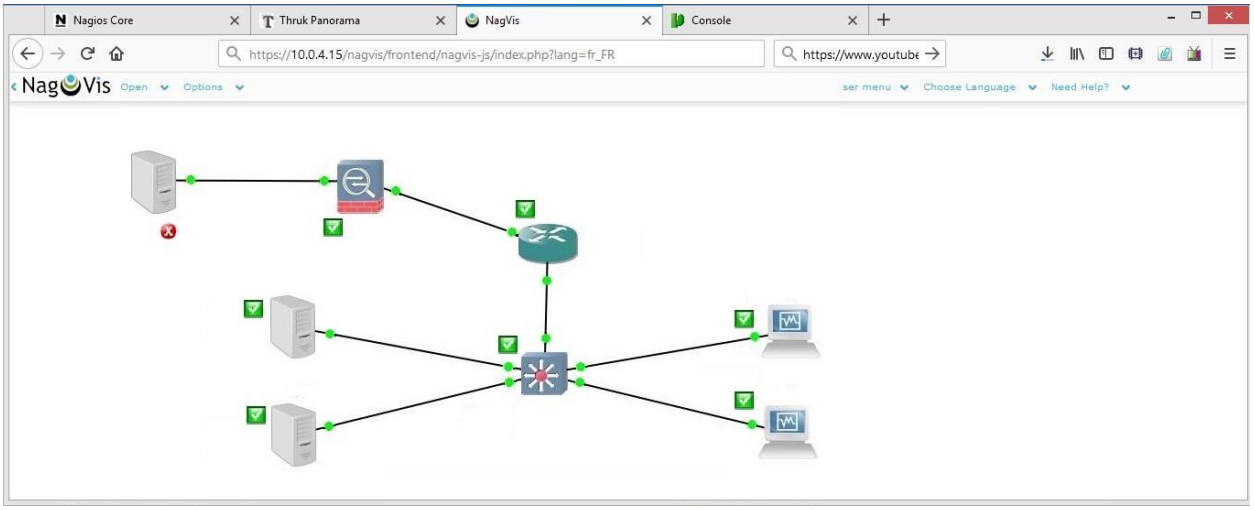


**Figure 4.14:** *Modification au niveau de la cartographie dans NagVis*

### *4.6.2 Réception de la notification dans la boîte de réception Gmail*

Lors des configurations dans Nagios, on n' a planifié de notification que pour le plugin *check_ping* si celui-ci retourne une valeur correspondant à l' état CRITICAL, c' est à dire que la machine Windows Server est injoignable ou éteinte.

```
define service{
hostname                        win-serv
service_description             statut_machine_DMZ
check_command                   check_ping
notification_option             c
contacts                        fihobiana
}
```

**Figure 4.15:** *Définition du service « check_ping » pour Windows Server*

```
define contact{
contact_name                    fihobiana
host_notification_command       host-notify-by-mail
email                           fihobiana21@gmail.com
}
```

**Figure 4.16:** *Définition du contact pour indiquer l' adresse destinataire*

En se connectant dans le compte Gmail correspondante à l' adresse ajoutée lors de la définition de contact (Figure 4.16), on a un nouveau courrier dans la boîte de réception. Lorsqu' on l' ouvre, on peut lire le message de Nagios(Figure 4.17). On rappelle que c'est la même adresse ajoutée lors de la configuration de Postfix, donc elle est à la fois émettrice et réceptrice.

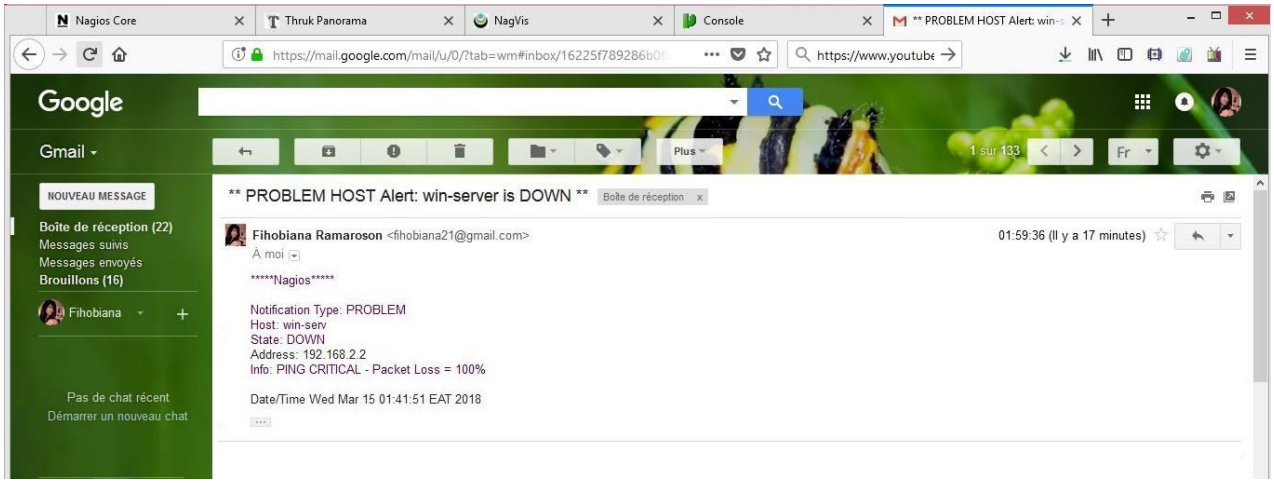

**Figure 4.17:** *Affichage du message reçu dans la boîte Gmail*

## 4.7 Conclusion

Faute de ressources matérielles, l' architecture réseau conçue est assez basique. La machine sur laquelle on effectue la simulation doit avoir de puissantes ressources pour pouvoir démarrer plusieurs hôtes virtuels. On peut encore l'exploiter en transformant le LAN en véritable intranet et ajouter plus de services dans la DMZ.

En réalité, pour bien surveiller un réseau, il faut prendre en compte tous les hôtes et leurs services en appliquant la supervision active qui représente la plus grande partie du monitoring sans négliger Cacti qui est plus spécialisé dans l' évaluation de la performance du réseau.

Quant à NagVis, celui-ci aide à identifier rapidement un hôte accidentellement éteint, ce qui constitue son plus par rapport à Thruk. L' autodécouverte avec Nmap n'est pas nécessaire que lorsqu' il s' agit de monitorer une infrastructure assez étendue où il est difficile de faire l' inventaire sur terrain de tous les hôtes.

# CONCLUSION GENERALE

Un défi majeur pour l' administrateur réseau et système est de garantir la disponibilité permanente des hôtes et de leurs services. Cela est rendu moins ardu grâce à un outil de monitoring qui regroupe un ensemble de fonctionnalités ayant comme vocation de suivre de près l'état de l'infrastructure des systèmes d'information. Ainsi, on peut réagir de manière proactive face aux signaux d'avertissement afin d'éviter les pannes. Et puisqu' il n'est pas possible de visualiser l'écran à temps plein, un mail est envoyé en cas de défaillance d' un élément du réseau.

Le serveur de monitoring est élaboré grâce à l'assemblage de plusieurs logiciels open-sources personnalisés dont le principal est Nagios. Celui-ci repose sur le protocole SNMP pour effectuer la supervision. Thruk, Cacti et NagVis importent les données de Nagios pour réaliser le reporting, la métrologie et la cartographie. Cette solution permet de visualiser facilement tous les éléments d'un réseau informatique à travers quatre interfaces web conviviales affichant des informations globales et détaillées. La mise en place d' un système de monitoring doit cependant respecter la politique de sécurité existante et veiller à ne pas perturber les processus déjà établis.

L'inconvénient de cette solution, étant à base de logiciels libres, c'est qu'elle requiert une longue période d'études et un simple oubli ou bien la moindre erreur dans les séries de configurations peut l' empêcher de bien fonctionner. Elle impose alors une mise en œuvre méticuleuse pour fournir les résultats exacts. Néanmoins, même avec les solutions payantes, on ne pourrait bénéficier d'une surveillance nettement en temps réel. Pour s'en rapprocher, la machine sur laquelle on installe la plateforme doit être munie d'un puissant processeur pour minimiser le temps de calcul et d'une grande capacité de stockage.

En perspective du projet, une optimisation du mécanisme d'alerte est envisagée par l'utilisation de la notification par SMS qui est sans doute plus pratique que les mails exigeant une connectivité en permanence à Internet. Il faudrait également chercher une solution pour les problèmes d'adaptation de la supervision au service DHCP. Nagios étant extensible, on peut aussi multiplier les fonctionnalités de la solution élaborée par le recours à d'autres modules. Les interfaces des logiciels choisis sont personnalisables, on peut donc les rendre plus interactives par le développement de nouveaux plugins. Mais étudier le monitoring dans les réseaux satellitaires de télécommunication parait encore plus intéressant.

**ANNEXE A1**

**QUELQUES APERÇUS DE L'ELABORATION DU SYSTEME DE MONITORING**

```
CACTI_DIR    var/www/html/cacti
NAGIOS_CONF_DIR   /usr/local/nagios/nagios.cfg
ROTATION h
PID_FILE  perf2rrd
SERVICE_PERFDATA_PIPE  /var/log/nagios.perfdata.pipe
PERFDB_NAME nagioscacti
PERFDB_USER nagiosadmin
PERFDB_PASSWORD  cianalord
PERFDB_HOST   10.0.4.15
TEMPLATE_SEPARATOR_FIELD    @
```

**Figure A1.01 :** *Ecriture du contenu de n2rrd.conf*

```
define global {
   allowed_for_config=nagiosadmin
   allowed_user=nagiosadmin
   alias=projet_monitoring
   iconset=std_small
   backend_id=live_1
   label_show=1
   label_border=transparent
   render_mode=directed
   rankdir=TB
   width=500
   height=300
}
```

**Figure A1.02 :** *Création de carte dans NagVis*

```
define service {
name                        generic-service
active_checks_enableld      1
check_period                24x7
parallelize_check           1
retain_status_information   0
process_perf_data           1
retry_check_interval        1
max_check_attempts          1
notification_period         24x7
register                    0
}
```

**Figure A1.03:** *Définition d' un template de service*

```
define service{
use      generic-service
hostname      switchIOU
service_description    Port1/1
check_command    check_cisco!-C fihobiana -i f1/1
normal_check_interval   5
retry_check_interval   1
}
define service{
use     generic-service
host_name   switchIOU
service_description Uptime
check_command   check_snmp!-C fihobiana -o sysUpTime.1
}
define service {
use generic-service
hostname  ASA5500
service description  connexion
check_command  check_asa_connexion!g0/0! -w 50  -c 80
}
define service {
use generic-service
hostname ASA5500
service description  int
check_command  check_snmp_int
}
define service {
use   generic-service
hostname  Routeur
service_description   Fa0/0
check_command  check_snmp!-C fihobiana -o ifOperStatus.1 -r 1 -m RFC1213-MIB
}
define service{
use generic-service
hostname Routeur
service_description Fa0/1
check_command check_snmp!-C fihobiana -o ifOperStatus.2 -r 1 -m RFC1213-MIB
}
define service{
use   generic-service
hostname   wordp-serv
service_description  disponibilite_du_site
chech_command     chech_wordpress_site
}
```

**Figure A1.04:** *Extraits du contenu du fichier « services.cfg »*


**Figure A1.05:** *Modification du niveau de sécurité de Gmail*

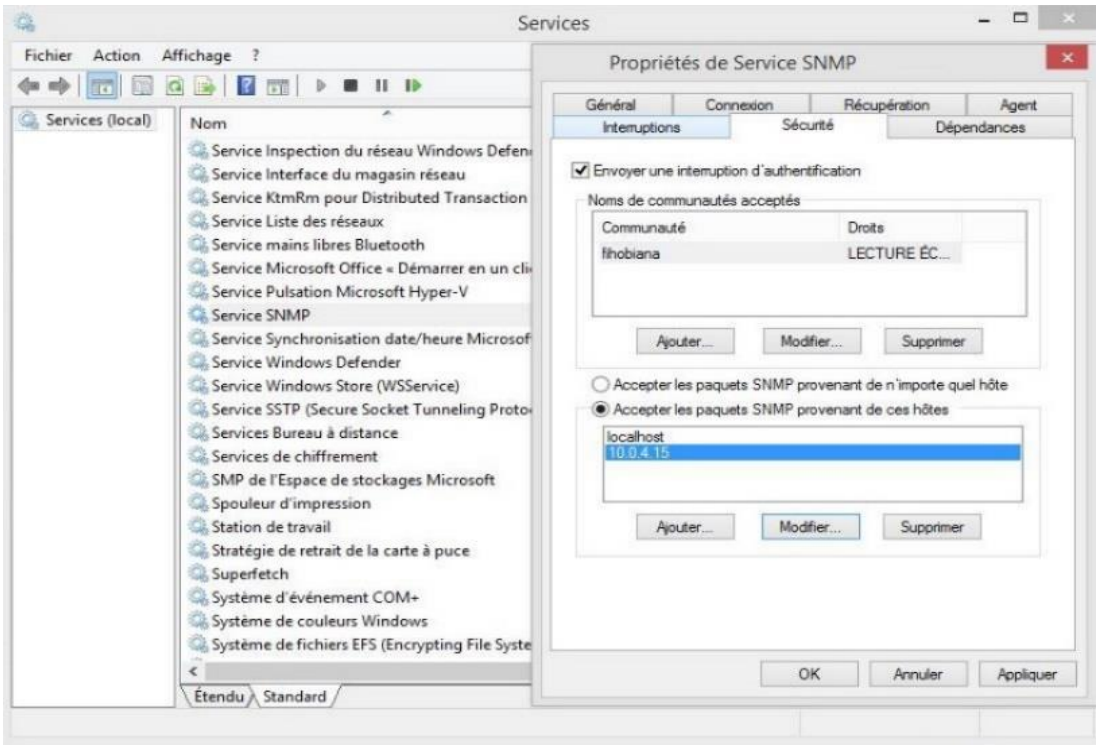


**Figure A1.06 :** *Activation de l' agent SNMP sur les machines Windows*

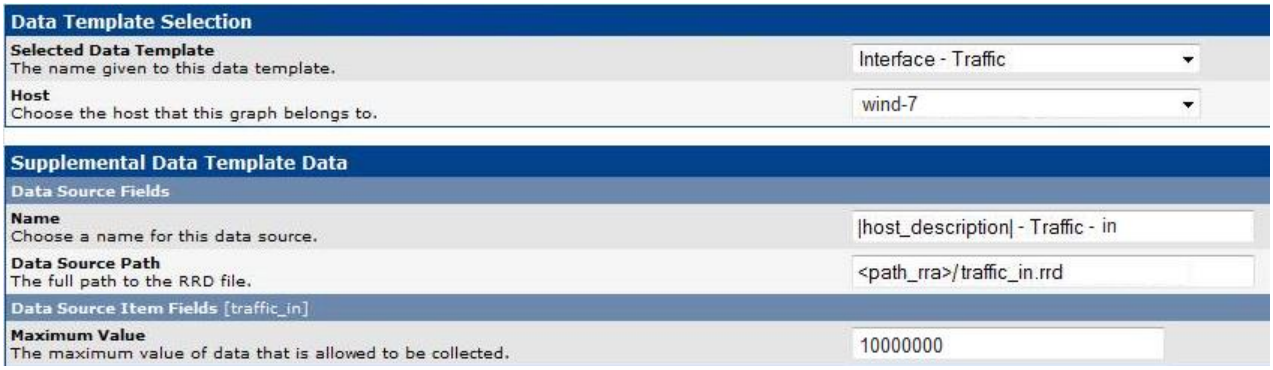


**Figure A1.07 :** *Création d' un « Data Template » dans Cacti*

**ANNEXE A2**

**EXTRAITS DE CONFIGURATION RESEAU DES HOTES VIRTUELS**

```
SW>en
SW#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.

SW(config)#int f1/0
SW(config-if)#no switchport  !makes the interface Layer 3 capable
SW(config-if)#ip address 10.0.1.2 255.255.255.0    !
SW(config-if)#exit
SW(config)#ip routing 0.0.0.0 0.0.0.0 10.0.1.1
SW(config)#int vlan 1
SW(config-if)#ip add 10.0.4.254 255.255.255.0
SW(config-if)#no shut
%LINK-5-CHANGED: Interface Vlan1, changed state to up
SW(config-if)#exit
SW(config)#int vlan 2
SW(config-if)#ip add 10.0.3.254 255.255.255.0
%LINK-5-CHANGED: Interface Vlan2, changed state to up
SW(config-if)#no shut
SW(config-if)#exit
SW(config)#int f1/1
SW(config-if)#switchport access vlan 1
SW(config)#int f1/2
SW(config-if)#switchport access vlan 1
SW(config)#int f1/3
SW(config-if)#switchport access vlan 2
SW(config)#int f1/4
SW(config-if)#switchport access vlan 2
Switch(config-if)#switchport mode trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/1, changed state to up
SW(config-if)#exit
```

**Figure A2.01 :** *Configuration du switch Cisco dans VirtualBox*

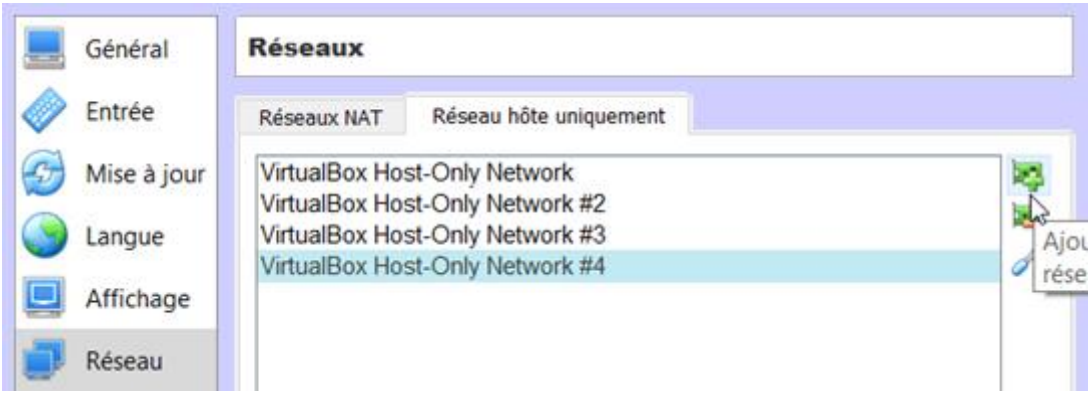


**Figure A2.02 :** *Création de cartes réseaux virtualles dans VirtualBox*

```
2194 bytes copied in 0.450 secs
[OK]
firewallASA5500(config)#int g0/0
firewallASA5500(config-if)#no sh
firewallASA5500(config-if)#ip address 169.254.102.160 255.255.255.0
firewallASA5500(config-if)#nameif outside
INFO: Security-level for "outside" set to 0 by defaults.

firewallASA5500(config)#int g0/1
firewallASA5500(config-if)#ip address 10.0.1.1 255.255.255.0
firewallASA5500(config-if)#nameif inside
INFO: Security-level for "inside" set to 100 by defaults.
firewallASA5500(config-if)#no shutdown


firewallASA5500(config)#int g0/2
firewallASA5500(config-if)#ip address 192.168.2.254 255.255.255.0
firewallASA5500(config-if)#nameif DMZ
INFO: Security-level for "DMZ" set to 0 by defaults.
firewallASA5500(config-if)#security-level 50
firewallASA5500(config-if)#no shutdown

firewallASA5500(config-if)#exit

firewallASA5500(config)#object network inside-subnet
firewallASA5500(config-network-object)#subnet 10.0.1.0 255.255.255.0
firewallASA5500(config-network-object)#nat (inside,outside) dynamic interface
firewallASA5500(config-network-object)#nat (dmz, inside) static interface
firewallASA5500(config-network-object)#exit
firewallASA5500(config)#object network dmz-subnet
firewallASA5500(config-network-object)# subnet 192.168.2.0 255.255.255.0
firewallASA5500(config-network-object)# nat (dmz,outside) dynamic interface
firewallASA5500(config-network-object)#exit

firewallASA5500(config)#object network webserver-external-ip
firewallASA5500(config-network-object)#host 198.51.100.101

firewallASA5500(config)#object network webserver
firewallASA5500(config-network-object)#host 192.168.2.2
firewallASA5500(config-network-object)#nat (dmz,outside) static webserver-external-ip service tcp www www
firewallASA5500(config)#router eigrp 20
```

**Figure A2.03 :** *Extrait de la configuration réseau du firewall*

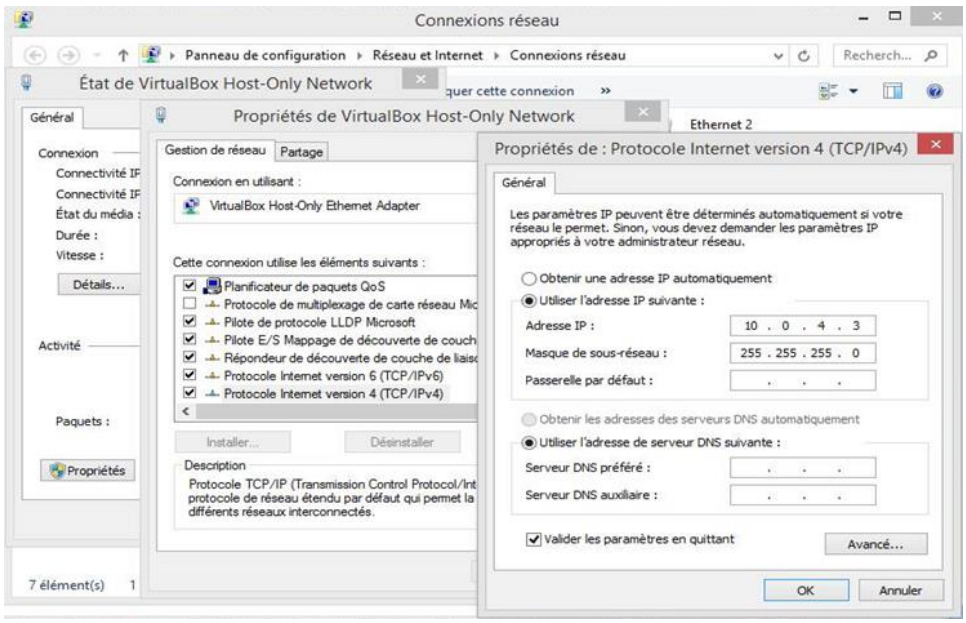


**Figure A2.04:** *Configuration réseau de la machine réelle*


**Figure A2.05 :** *Création de répertoire virtuel pour le site à héberger localement*

## ANNEXE A3

## QUELQUES EXTRAITS DE CODES SOURCES

### A3.1 thruk.patch

```
{
  "general": {
    "host": "",
    "service": "",
    "incl_hst": "on"
  },
  "layout": {
    "x": "1500",
    "y": "1500"
},
  "appearance": {
    "type": "pie",
    "piewidth": 1450,
    "piecolor_ok": "#34C924",
    "piecolor_warning": "#CDCD0A",
    "piecolor_critical": "#EB0000",
    "piecolor_unknown": "#FF7F00",
    "piecolor_up": "#34C924",
    "piecolor_down": "#EB0000",
    "piecolor_unreachable": "#FF7F00",
    "piecolor_pending": "#FF00FF"
```

### A3.2 nagvis.ini.php-sample

```
[path]
base= "/usr/local/nagvis/share"
htmlbase="/nagvis"
htmlcgi="usr/local/nagios/cgi-bin"
[defaults]
contextmenu=0
icons="std_small"
hosturl=[htmlcgi]/status.cgi?host=monitoring-
servermapurl=[htmlbase]/index.php?mod=Map&act=view&show=monitoring-map
[worker]
Interval=10
requestmaxparams=0
updateobjectstates=30
[backend_live_1]
backendtype="mklivestatus"
backend="live_1:nagios"
socket="unix:/usr/local/nagios/var/rw/live"
dbhost="localhost"
dbport=3306
dbname="nagios"
dbuser="root"
```

### A3.3 win_traff.sh

```
#!/bin/bash
while getopts "H:p:i:o:d" Input;
do
       case ${Input} in
       H)      host=${OPTARG};;
       p)      port=${OPTARG};;
       i)      interface=${OPTARG};;
       o)      output=${OPTARG};;
       d)      detect=1;;
       *)      echo "Wrong option given."
               exit 1
               ;;
       esac
done
if [[ -n ${port} ]]
then insertport=${port}
else insertport=12489
fi
if [[ ${detect} -eq 1 ]]; then
  if [[ -n ${password} ]]; then
    ${pluginlocation}/check_nt -H ${host} -p ${insertport} -s ${password} -v
INSTANCES -l "Network Interface" | sed "s/OK&//"; exit 0
  else
    ${pluginlocation}/check_nt -H ${host} -p ${insertport} -v INSTANCES -l
"Network Interface" | sed "s/OK&//"; exit 0
  fi

bytes_in=$(${pluginlocation}/check_nt -H ${host} -p ${insertport} -v COUNTER
-l "\\Network Interface(${interface})\\Bytes Received/sec")
bytes_out=$(${pluginlocation}/check_nt -H ${host} -p ${insertport} -v
COUNTER -l "\\Network Interface(${interface})\\Bytes Sent/sec")
fi
if [ -n "${output}" ]
then
        if [ "${output}" = "KB" ]
        then return_bytes_in=$(expr ${bytes_in} / 1024)
        return_bytes_out=$(expr ${bytes_out} / 1024)
        value="KBytes"
        elif [ "${output}" = "MB" ]
        then return_bytes_in=$(expr ${bytes_in} / 1024 / 1024)
        return_bytes_out=$(expr ${bytes_out} / 1024 / 1024)
        value="MBytes"
        fi
else
return_bytes_in=${bytes_in}
return_bytes_out=${bytes_out}
value="Bytes"
fi

echo "Network OK - ${return_bytes_in} ${value} received/sec,
${return_bytes_out} ${value} sent/sec|bytes_in=${bytes_in}B;;;;
bytes_out=${bytes_out}B;;;;"
exit 0
```

## BIBLIOGRAPHIE


[1] J. Dordoigne, « *Réseaux informatiques : Notions fondamentales* », Ecole nationale d'Informatique de Québec, 7ème édition, 2014.

[2] http://www.ordinateur.cc/réseaux/Réseaux-locaux/71827.html, Janvier 2018.

[3] http://www.hotosting.com/cresite/support-transmission.html, Janvier 2018.

[4] https://repo.zenksecurity.com/supports/caracteristiques_des_supports_de_transmission.pdf, Janvier 2018.

[5] http://www.mi.parisdescartes.fr/~mea/cours/L3/4_L3_interconnexion.pdf, Janvier 2018.

[6] http://www.coursnet.com/2016/12/les-topologies-physiques.html, Janvier 2018.

[7] R. Legrand, L. Schalkwijk, « *Les réseaux avec Cisco: Connaissances approfondies sur les réseaux»*, 3ème édition ENI, 2015.

[8] https://www.google.mg/url?La-gamme-Cisco-trois-couches-du-modele-hierarchique.htm, Janvier 2018.

[9] http://www.nolot.eu/Download/Cours/reseaux/m2pro/CRC-0809/crc-cours2-vlan.pdf, Janvier 2018.

[10] https://openclassrooms.com/courses/les-reseaux-de-zero/les-topologies-1, Janvier 2018.

[11] https://www.techniques-ingenieur.fr/base-documentaire/archives-th12/, Janvier 2018.

[12] http://pedagogie.ac-limoges.fr/eco-gest/IMG/pdf/communcation_reseau.pdf, Janvier 2018.

[13] D. Harkey, R. Horfali, J. Edwards « *Client/Serveur, Guide de survie* », Enseignement supérieur en Réseau et Système, 3ème édition, 2016.

[14] http://www-igm.univ-mlv.fr /vayssade/ clustering.htm, Janvier 2018.

[15] https://fr.wikipedia.org/wiki/Cloud_computing, Janvier 2018.

[16] http://perso.citi.insa-lyon.fr/sfrenot/cours/SID/cours/SID12-CS.pdf, Janvier 2018.

[17] https://fr.slideshare.net/HeithemAbbes1/architectures-ntiers, Janvier 2018.

[18] https://www.researchgate.net/Security-Access-Architecture-InternetExtranetIntranet, Janvier 2018.

[19] https://fr.slideshare.net/1/ -mise-en-place-dune-solution-de-monitoring.htm, Octobre 2017.

[20] https://www.riverbed.com/document/fpo/media-cms/Riverbed-Network-Monitoring-For-Dummies.pdf.pdf, Janvier 2018.

[21] http://www.open-source-guide.com/ /Supervision-et-la-metrologie.htm, Février 2018.

[22] http://www.irisa.fr/prive/bcousin/Cours/14-SNMP.fm.pdf, Février 2018.

[23] http://www.frameip.com/snmp/, Octobre 2017.

[24] F. Pignet, « *Réseaux Informatiques : Supervision et Administration* », Collection Expert IT, Edition ENI, janvier 2016.

[25] http://munier.perso.univ-pau.fr/temp/ /Cours-Supervision-Reseau.pdf, Octobre 2017.

[26] https://www.alsacreations.com/tuto/lire/614-Serveur-mail-Postfix.html, Décembre 2017.

[27] http://www.esaracco.fr/documentation/nagios/nagios.pdf, Novembre 2017.

[28] A. Madjoub, « *Nagios, la clé de la supervision informatique* », Collection Epsilon, Nouvelle Edition, Janvier 2016.

[29] http://www.hpckp.org/images/conference/2014/hpckp14_omd-intro.pdf, Décembre 2017.

[30] https://docs.nsclient.org/howto/nrpe/, Février 2018.

[31] https://www.scribd.com/document/261805377/CACTI-formation-cacti-monitoring-infrastructure-pdf, Février 2018.

[32] http://nicolargo.developpez.com/tutoriels/nagios/serveur-supervision-libre-installation-cacti/, Février 2018.

[33] https://wiki.monitoring-fr.org/nagios/addons/nagvis/star, Février 2018.

[34] http://nagios.manubulon.com/traduction/docs14fr/cgiauth.html, Février 2018.

[35] https://doc.monitoring-fr.org/securityandperformancetuning-cgisecurity.html, Février 2018.

[36] https://blog.nicolargo.com/nmap-le-scanneur-de-reseau.html, Février 2018.

[37] http://irt.enseeiht.fr/anas/cours/tp-ssh.pdf, Février 2018.

[38] https://memo-linux.com/prise-en-main-a-distance-dun-poste-windows-a-partir-dun-gnulinux/, Février 2018.

[39] http://dept-info.labri.fr/~magoni/miarea/TD-SNMP.pdf, Février 2018.

[40] http://nagios.manubulon.com/traduction/docs14fr/templaterecursion.html, Février 2018.

[41] https://doc.monitoring-fr.org/3_0/html/configuringnagios-objectdefinitions.html, Février 2018.

[42] https://assets.nagios.com/Understanding-Nagios -Notification-Escalations.pdf, Mars 2018.

[43] http://nagios.manubulon.com/traduction/wiki_preview/perfdata.html, Mars 2018.

[44] http://cainamur.be/images/pdf/20050627_introduction_cacti.pdf, Mars 2018.

[45] https://wiki.monitoring-fr.org/nagios/addons/nagvis/nagvis-manuel-utilisation, Mars 2018.

[46] www.lemagit.fr/definition/Virtualisation, Février 2018.

[47] https://www.smnet.fr/gns3/gns3-res1.html, Janvier 2018.

[48] A. William, « *Wordpress for Beginners »*, A Visual Step-by-Step Guide to Mastering Wordpress, Novembre 2017.

[49] https://gns3.com/news/article/connect-gns3-to-internet, Janvier 2018.

[50] https://www.cisco.com/c/fr_ca/support/docs/security/asa-5500-x-series-next-generation-firewalls/115904-asa-config-dmz-00.pdf, Janvier 2018.

[51] https://www.cisco.com/c/en/us/support/docs/ip/simple-network-management-protocol-snmp/7282-12.html, Février 2018.

[52] https://www.cisco.com/web/learning/le21/le34/downloads/689/academy/2013/sessions/BRK-135T_CCNA_Switching.pdf, Janvier 2018.

[53] https://wiki.monitoring-fr.org/_media/nagios-doc-2x-fr.pdf, Mars 2018.

[54] https://www.linickx.com/how-to-monitor-wordpress-with-nagios, Mars 2018.

[55] https://doc.ubuntu-fr.org/tutoriel/script_shell, Février 2018.

[56] https://exchange.nagios.org/directory/Plugins/Network-Connections/-Stats-and-Bandwidth/Monitoring-Interface-Bandwidth-Utilization-Using-Cacti-Data/details, Mars 2018

**FICHE DE RENSEIGNEMENTS**

**Nom :** RAMAROSON

**Prénom :** Fihobiana

**Adresse du logement de l'auteur :** Lot II B 108 Bis E
Ambatomainty Antananarivo 101 – Madagascar
**Tel:** +261 34 31 366 12
**E-mail**: fihobiana21@gmail.com

**Titre du mémoire :**

**« ELABORATION D' UNE SOLUTION DE MONITORING A APPLIQUER DANS UN RESEAU INFORMATIQUE»**

**Nombre de pages :** 75

**Nombre de tableaux :** 4

**Nombre de figures :** 50

**Directeur du mémoire :** RATSIMBAZAFY Andriamanga

**Grade :** Maître de Conférences

**Tel:** +261 75 46 638 84

## RESUME

Le monitoring figure parmi les activités indispensables dans le cadre de l'administration réseau et système. La solution créée à partir du couplage de trois logiciels open-source dont Nagios, Cacti et NagVis est basée sur le protocole SNMP. Elle peut être proposée à l'équipe du système d'information afin de l'aider à être informée de l'état du réseau à tout moment. Elle est adaptable à n'importe quel environnement informatique puisqu'elle est capable d'opérer sur un grand nombre d'équipements quels que soient la topologie et le type de matériel existants. Le monitoring conduit également à l'amélioration de l'infrastructure IT à partir de la localisation des problèmes fréquents qui y surviennent.



**Mots-clés:** Réseau, Système, Monitoring, SNMP, Objet.



## ABSTRACT

Monitoring is one of the essential activities in network and system administration. The solution created from the coupling of three open-source software including Nagios, Cacti and NagVis is based on the SNMP protocol. It can be offered to the information system team to help them be informed of the state of the network at any time. It is adaptable to any computer environment since it is capable of operating on a large number of devices whatever the topology and the type of existing equipment. Monitoring also leads to improvement of the IT infrastructure by locating the frequent problems that occur there.



**Key-words:** Network, System, Monitoring, SNMP, Object.