## Introduction

Une étape primordiale dans le déroulement du projet se focalise sur le choix et l’étude de la solution ainsi que la phase de spécification des besoins. Dans ce chapitre, nous présentons l’étude de notre outil de supervision choisi et son mode de fonctionnement ensuite nous détaillons les besoins fonctionnels et non fonctionnels.

## 1. Spécifications des besoins

### 1.1. Besoins fonctionnels

L'objectif de ce projet est de mettre en place une solution de monitoring sur une machine virtuelle grâce à l’outil de virtualisation VMware. Cette application doit permettre de collecter des informations concernant une infrastructure informatique à plusieurs niveaux :

* La supervision des ressources des serveurs (charge du processeur, occupation des disques durs, utilisation de la mémoire paginée) sur la plupart des systèmes d’exploitation.
* La supervision des services réseau (SMTP, HTTP, NNTP, ICMP, SNMP, LDAP, etc…) en local ou sur des machines distantes.
* L’interfaçage avec le protocole SNMP.
* La vérification des services se fait en parallèle.
* La remontée des alertes est entièrement paramétrable grâce à l'utilisation de plugins

Ø Acquittement des alertes par les administrateurs.

### 1.2. Besoins non fonctionnels

Afin d’offrir une solution complète et performante à différents niveaux, notre plateforme doit couvrir les besoins non fonctionnels suivants :

* **Facilité d’utilisation** : Le système offre une interface simple facile à utiliser en donnant à l’administrateur la possibilité d’agir sur les ressources qu’il manipule.
* **Sécurité** : L’accès aux données doit être authentifié et autorisé par des moyens de sécurité.
* **Fiabilité***:* Il faut garantir la qualité du contenu et la pertinence des informations. Le produit doit fonctionner correctement.
* **Rapidité** : Le logiciel de supervision prévient dés qu’un problème survient avant même que la plupart des utilisateurs en aient conscience.
* **Extensibilité**: Le système doit être extensible et permet d'ajouter et de supporter d'autres fonctionnalités et d’intégrer tout type d’équipement réseau.
* **La performance** : une application doit être avant tout performant c'est à dire à travers ses fonctionnalités, elle répond à toutes les exigences des usagers d’une manière optimale.

## 2. Formulation des besoins sous forme de cas d’utilisation

### 2.1. Présentation du langage de modélisation UML

UML (*Unified Modeling Language*) est un [langage graphique](http://fr.wikipedia.org/wiki/Langage_graphique) de modélisation des données et des [traitements.](http://fr.wikipedia.org/wiki/Traitement_de_l%27information) Il possède de multiples avantages : C’est un support de communication performant, il décrit une application en fonction des méthodes objet avec lesquelles elle a été construite. Il cadre l’analyse et facilite la compréhension de représentations abstraites complexes et il se caractérise par sa notation graphique simple qui permet d’exprimer visuellement une solution objet .

Dans UML chaque diagramme permet d’exprimer certains points d’un même problème. La combinaison de plusieurs digrammes permettra donc d’avoir une vue complète du système informatique. Ainsi en fonction du problème à résoudre, il convient de choisir les diagrammes adéquats à utiliser.

Pour la réalisation de ce projet, il a fallu disposer d’un environnement de travail adéquat du point de vue conception, déploiement et test tout en prenant en compte les différentes technologies adoptées. Le logiciel StarUML nous a semblé être l’outil simple et idéal pour concevoir notre système vu les fonctionnalités qu’il offre.

### 2.2. Diagrammes des cas d’utilisations

Un diagramme de cas d’utilisation est un moyen simple d’exprimer des besoins. Il montre le comportement d'un composant, une classe ou un système, tel qu'un utilisateur extérieur le voit. Il correspond à un ensemble de transactions effectuées au cours d’une interaction entre l’acteur et le système.

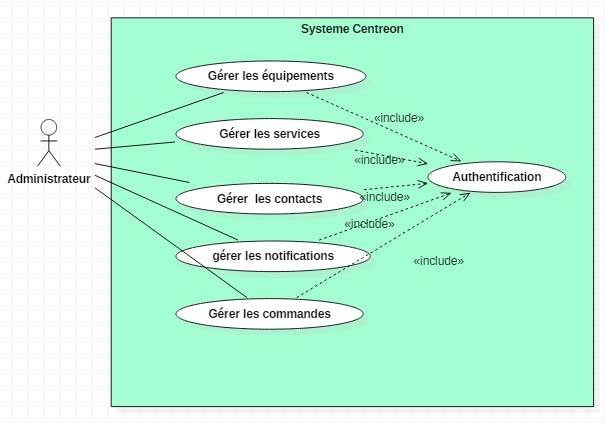
#### 2.2.1. Identification des acteurs

Globalement, nous avons distingué un seul acteur: l'administrateur, celui qui aura la fonction de la supervision réseau .en fait son rôle consiste à superviser le réseau en récupérant des informations sur les équipements et réparer les pannes détectées. Cet acteur utilise le système à travers un ensemble d’interfaces bien définies. Il doit s’authentifier pour qu’il puisse utiliser le système**.**

#### 2.2.2. Description des cas d’utilisation

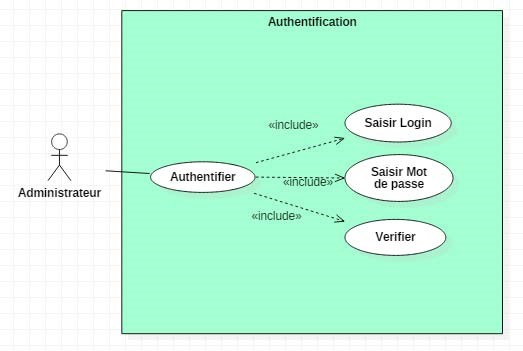
Afin de décrire les exigences fonctionnelles de notre système, voici une description du cas d’utilisation globale ainsi que d’autres qui sont les principaux.

#### a. Diagramme de cas d’utilisation générale



**Figure 9 : Diagramme de cas d’utilisation générale du système**

#### a. Diagramme de cas d’utilisation « s’authentifier »



**Figure 10 : Diagramme de cas d’utilisation « Authentification »**

#### Ø Titre : Authentification

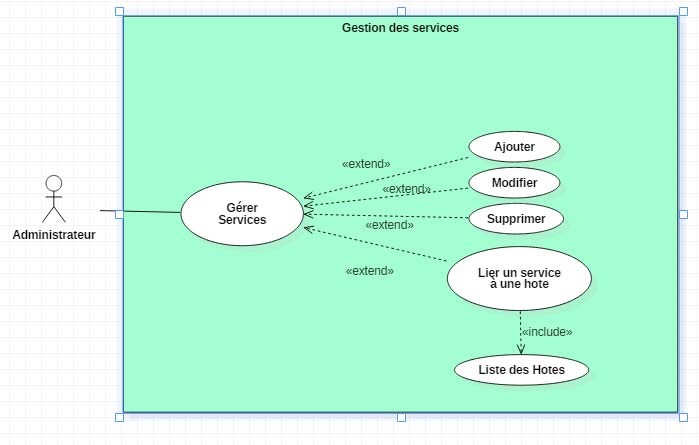
* **Objectif:** se connecter au système.
* **Acteurs: A**dministrateur.
* **Pré-Condition**: L'enchaînement démarre lorsque l’administrateur demande d’accéder au système. ***Scénario nominal***
* L’administrateur demande à s’authentifier.
* Le système affiche la page d’authentification.
* L’administrateur fournit son login et son mot de passe.

##### Enchaînement d'exception

Un message d’erreur sera affiché si le login et/ou le mot de passe sont incorrects. ***Post-condition***

L’administrateur peut accéder aux différentes fonctionnalités du système.

#### b. Diagramme de cas d’utilisation « Gestion des services »



**Figure 11 : Diagramme cas d’utilisation Gestion des services**

* **Titre :** Gestion des services
* **Objectif:** ajouter, modifier, supprimer un service ou lier un service à une hôte parmi la liste des hôtes.
* **Acteurs: A**dministrateur.
* **Pré-Condition**: L'administrateur est déjà identifié.

##### Scénario nominal

* L’administrateur affiche la partie de gestion des services.
* L’administrateur choisi une opération à faire.
* L’administrateur effectue l’opération choisie précédemment.Ø Le système enregistre les modifications effectuées.

##### Enchaînement d'exception

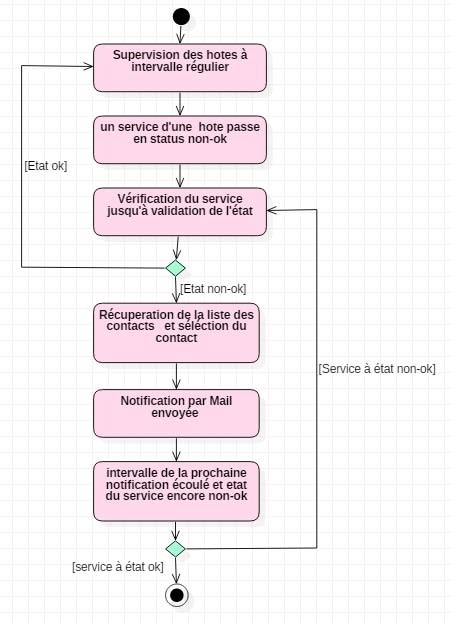
Un message d’erreur sera affiché si les informations introduites lors de l’opération sont incorrects ou si un champ obligatoire est vide.

Le Diagramme d'activités est un autre diagramme important dans UML pour décrire les aspects dynamiques du système. Il est essentiellement un organigramme pour représenter le flux d’une opération vers une autre opération.

La description d'un cas d'utilisation par un diagramme d'activités correspond à sa traduction algorithmique.

###### 2.2.3. Diagramme d’activité « Notification »

Ce diagramme décrit les différentes activités que prend le système lorsqu’il détecte un service ou équipement non fonctionnel. A ce stade le système commence par vérifier l’état du service correspondant à l’hôte jusqu'à la validation de l’état non-ok. Ensuite, il récupère la liste des contacts afin d’en choisir un et le notifier par un mail.si l’intervalle de temps de la prochaine notification est écoulé et que l’état du service est encore non-ok, le système recommence la vérification des services.

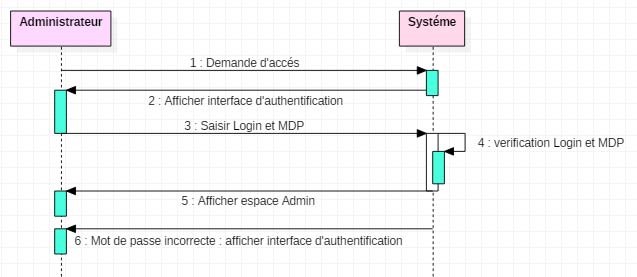


**Figure 12: Diagramme de cas d'activité Notification**

###### 2.2.4. Diagramme de séquence

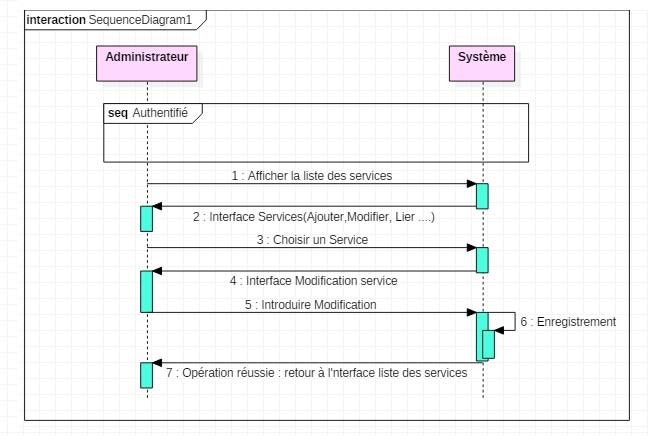
Un diagramme de séquence est un diagramme d'interaction qui montre comment les processus fonctionnent entre eux et dans quel ordre. Il est une construction d'un graphique de séquence de message. Un diagramme de séquence montre les interactions d'objets disposés selon un ordre chronologique. Il représente les objets et les classes impliquées dans le scénario. Les diagrammes de séquence sont généralement associés aux diagrammes de cas d'utilisation dans la vue logique du système.

#### a. Diagramme de séquence cas d’utilisation « Authentification »



**Figure 13 : Diagramme de séquence « Authentification »**

#### b. Diagramme de séquence modification d’un service



**Figure 14 : Diagramme de séquence « Gestion d'un service »**

## 2. Principe de fonctionnement de l’outil à réaliser

D’abord on a commencé par l’installation du logiciel de virtualisation qui est dans notre cas VMware, sur lequel on a ensuite mis en place une machine Virtuelle possédant une image iso Ubuntu 12.04 LTS comme système d’exploitation. Sur ce dernier on a installé :

* L’outil de supervision Nagios avec ses différents plugins qui seront nécessaire à son fonctionnement. Afin de faciliter l’exploitation de ce Nagios.
* Centreon qui est un logiciel de supervision joue le rôle d’une interface graphique pour Nagios et il va faciliter l’utilisation des fonctionnalités de Nagios.
* Configuration des différents équipements à superviser ainsi que les services qu’on a besoin.
* Configuration de Cenreon à l’envoie des alertes lors d’un problème au niveau d’un équipement. On a voulu utilisé les 2 méthodes disponible sur Centreon qui sont l’envoie des alertes par mails et par SMS.

è

## 4. Centreon

### 4.1. Présentation

Centreon est une application open source et gratuite qui permet de gérer Nagios et ses fichiers de configuration via une interface Web. L'avantage de Centreon est qu'elle dispose d'une interface plus intuitive et conviviale [17].

Centreon reprend donc les avantages du moteur de Nagios et permet ainsi d'être entièrement compatible avec des solutions existantes. Son interface reprend un découpage classique :

* Home : Page d'accueil avec Le "Tactical Overview" de Nagios permettant un coup d'oeil rapide aux problèmes survenus et accès aux statistiques des performances du moteur et de ses composants.
* Monitoring : Possède plusieurs vues, mais reprend la grande idée de l'arbre des groupes d'équipements. Reprend également la vue Nagios.
* Views : Permet d'accéder à tous les graphiques avec un menu arborescent. Accès à une cartographie du réseau en applet Java.
* Reporting : Un dashboard ressemblant à celui de Zabbix en ajoutant une frise chronologique de la disponibilité de l'équipement.
* Configuration : Pour tout configurer de A à Z.
* Administration : Configuration des accès utilisateurs.

#### 4.1.1. Avantages

Le choix de Centreon s’est basé sur les avantages suivants :

* La robustesse et la renommée de Nagios
* Une interface beaucoup plus sympathique, permettant de tout configurer, de garder un œil sur tout le réseau en permanence
* Les utilisateurs de Nagios ne seront pas perdus pour autant, l'interface reprenant avantageusement certaines vues Nagios
* Une solution complète permettant le reporting, la gestion de panne et d'alarmes, gestion utilisateurs, ainsi que la cartographie du réseau.
* Une entreprise qui pousse le développement
* Peut être décoléré du serveur Nagios et tourner tout seul sur un autre serveur

#### 4.1.2. Inconvénients

Centreon possède des inconvénients parmi lesquelles :

* L'interface peut paraître complexe car il existe beaucoup d'options .
* Un développement qui n'est pas encore en phase avec celui de Nagios : Parfois des problèmes de compatibilité.
* Un peu plus lourd que du Nagios pur.

### 4.2. Principe de fonctionnement

Le principe de fonctionnement de Centreon est simple. L’administrateur configure les options de supervisions, hôtes, services, plugins, etc…grâce à son interface graphique.

Ensuite toutes ces configurations sont stockées dans une base de données, mais elles ne sont pas immédiatement appliquer au moteur Nagios qu’aprés avoir relancer Centreon.

Dans certains cas l’administrateur fait une faute de frappe lors de la configuration d’un fichier, ce qui entraine le non fonctionnement de nagios et par la suite une perte de temps de la part l’administrateur en essayant de retrouver l’erreur. Centreon évite ce problème car il contrôle les données entrées par l’administrateur avant de les valider [9].

Il est aussi possible avec Centreon de créer ses propres commandes selon les besoins afin de configurer les services voulus.

## Conclusion

Le but de ce document était de présenter en détails l’outil de supervision choisi ainsi que ses compléments. Certains ont été choisis pour leur nécessité comme les greffons NRPE et NSClient, et d’autres participaient surtout à l’amélioration de la manipulation et l’utilisation de Nagios, et surtout facilité de sa configuration.