III.3.2.3 La MIB

- • Présentation

Pour que SNMP fonctionne, il est nécessaire qu'un protocole d'échange soit définit. Il y a aussi une standardisation des informations que ce protocole peut transporter. C'est un protocole Internet, il doit être utilisable sur des plates-formes hétérogènes (matériel comme système d'exploitation).

C'est pour cette raison que l'on parlera de MIB (Management Information Base).En effet, la MIB est une base de données des informations de gestion maintenue par l'agent. C'est cette base à laquelle on va demander les informations.

- • Structure de la MIB

La structure de la MIB est hiérarchique : les informations sont regroupées en arbre. Chaque information a un OID (Object identifier), une suite de chiffres séparés par des points, qui l'identifie de façon unique et un nom, indiqué dans le document qui décrit la MIB.

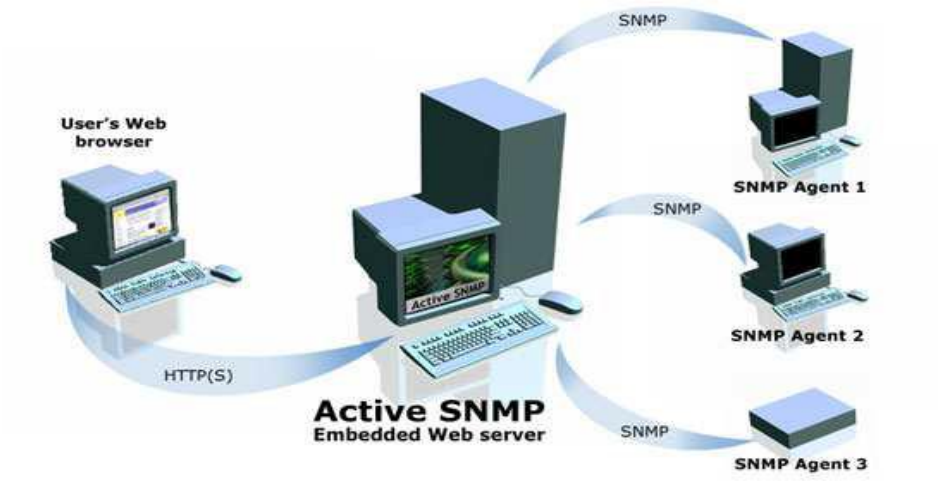

Figure III.1: Eléments de base du protocole SNMP

III.3.2.4.Les commandes SNMP

Il existe 4 types de requêtes SNMP :

- get-request : Le Manager SNMP demande une information à un agent SNMP
- get-next-request : Le Manager SNMP demande l'information suivante à l'agent SNMP

- set-request : Le Manager SNMP met à jour une information sur un agent SNMP
- trap : L'agent SNMP envoie une alerte au Manager

Les alertes sont transmises lorsqu'un événement non attendu se produit sur l'agent. Ce dernier informe le manager via une « trap ». Plusieurs types d'alertes sont alors possibles : ColdStart, WarmStart, LinkDown, LinkUp, AuthentificationFailure.

Pour chaque envoi de message, une réponse est retournée à l'exception de la commande « trap ». Les réponses sont du type suivant :

- get-response : L'information a bien été transmise.
- NoSuchObject : Aucune variable n'a été trouvée.
- NoAccess : Les droits d'accès ne sont pas bons.
- NoWritable : La variable ne peut être écrite.

III.3.2.5 Echange de message

Voici un schéma récapitulant les échanges pouvant être effectués entre un agent et le manager :

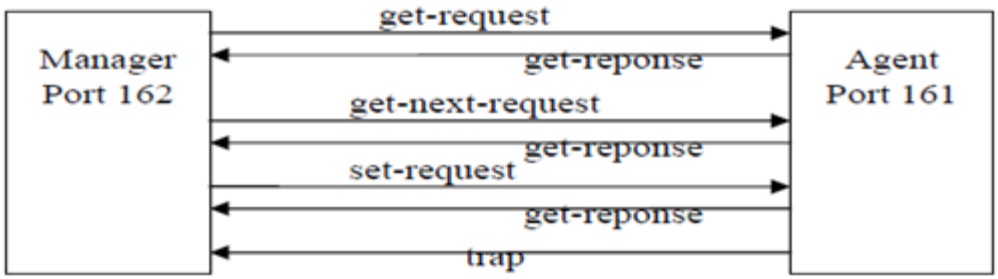

Figure III.2: Exemple d'échange SNMP

Le protocole SNMP est principalement utilisé avec UDP/IP. (Il peut aussi utiliser TCP). L'utilisation d'UDP permet un échange de message plus rapide que l'utilisation de TCP. L'inconvénient est qu'il est possible de perdre des trames lors de l'échange de messages (mode non connecté). Les ports UDP sont donc le 162 pour le manager et le 161 pour les agents.

III.3.3. SNMP en pratique

Concrètement, dans le cadre d'un réseau, SNMP est utilisé: pour administrer les équipements et pour surveiller le comportement des équipements Une requête SNMP est un datagramme UDP habituellement à destination du port 161. Les schémas de sécurité dépendent des versions de SNMP (v1, v2 ou v3). Dans les versions 1 et 2, une requête SNMP contient un nom appelé communauté, utilisé comme un mot de passe. Il y a un nom de communauté différent pour obtenir les droits en lecture et pour obtenir les droits en écriture.

Dans bien des cas, les colossales lacunes de sécurité que comportent les versions 1 et 2 de SNMP limitent l'utilisation de SNMP à la lecture des informations car la communauté circule sans chiffrement avec ces deux protocoles. Un grand nombre de logiciels libres et propriétaires utilisent SNMP pour interroger régulièrement les équipements et produire des graphes rendant compte de l'évolution des réseaux ou des systèmes informatiques (MRTG, Cacti, Nagios, Zabbix...)

III.4. Conclusion

La supervision est devenue indispensable dans tout système d'information. Elle est à la base du bon fonctionnement d'une architecture réseau et permet de réagir rapidement en cas de problèmes ou pannes. Elle se base à l'heure actuelle principalement sur le protocole SNMP qui depuis de nombreuses années a quand même du mal à évoluer. En effet, de nombreux logiciels sont encore basés sur la version 1 du protocole qui commence un peu à vieillir et qui n'est pas du tout sécurisé. En effet la version 2, apportant notamment la sécurité n'a été qu'une phase de transition vers la v3 qui est encore très peu utilisée.

Chapitre IV

OUTIL DE SUPERVISION NAGIOS

Chapitre IV

OUTIL DE SUPERVISION NAGIOS

IV.1. Introduction

La complexité et la grande quantité d'informations que Voyer sur des réseaux d'ordinateurs motiver la création d'équipements et de logiciels pour la gestion et le suivi de ces environnements informatiques. Une de ces ressources est le Nagios, outil qui vous permet de gérer plusieurs périphériques et services disponibles sur un réseau informatique. Le logiciel est conçu pour les entreprises cherchant des solutions pour gérer les réseaux locaux d'infrastructure ouverte et efficace. Il comprend des fonctions de surveillance, correction de gestion et de la faute. En outre, il a un grand nombre de plugins qui peuvent être regroupées, ce qui en fait un logiciel robuste et fiable.

IV.2. La supervision par nagios

Nagios est un logiciel qui fournit un ensemble de moyens et services pour assurer une supervision particulièrement simple, fiable, évolutive et non-propriétaire d'un parc informatique.

IV.2.1. Présentation de Nagios

Nagios est un logiciel de supervision de réseau libre sous licence GPL qui fonctionne sous Linux.

Il a pour fonction de surveiller les hôtes et services spécifiés, alertant l'administrateur des états des machines et équipements présents sur le réseau.

Bien qu'il fonctionne dans un environnement Linux, ce logiciel est capable de superviser toutes sortes de systèmes d'exploitation (Windows XP, Windows 2000, Windows 2003 Server, Linux) et également des équipements réseaux grâce au protocole SNMP.

Cette polyvalence permet d'utiliser Nagios dans toutes sortes d'entreprises, quelque soit la topologie du réseau et les systèmes d'exploitation utilisés au sein de l'entreprise.

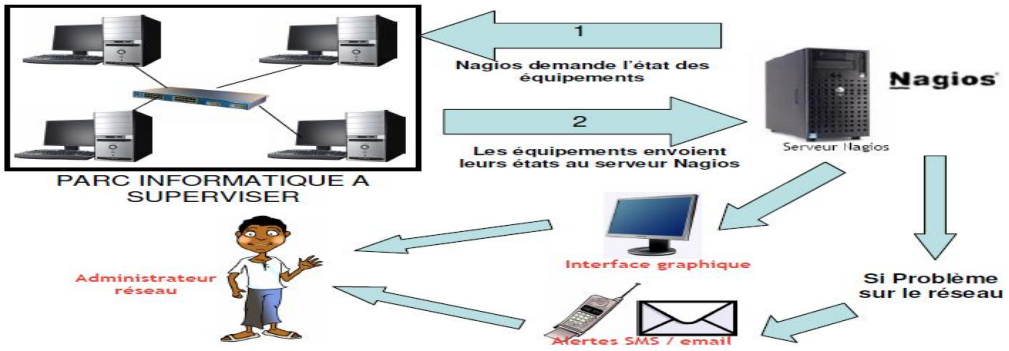

Figure IV.1 : l'interface graphique

IV.2.2. Architecture de Nagios

L'architecture de base de Nagios est simple :

- ***un ordonnanceur*** : Nagios est d'abord un moteur gérant l'ordonnancement des vérifications, ainsi que les actions à prendre sur incidents (alertes, escalades, prise d'action corrective) ;
- ***une IHM*** : la partie graphique visible à travers un simple serveur web, tel Apache est basée (pour les versions jusqu'à la 2.0) sur des CGI ;
- ***des sondes*** : les sondes de Nagios (les greffons ou *plugins*) sont de petits scripts ou programmes qui sont la base des vérifications.

Le projet Nagios fournit en standard bon nombre de greffons de base, mais la simplicité de leur mode de fonctionnement nous a permis d'en écrire un certain nombre pour nos besoins propres, que ce soit pour superviser dans notre environnement ou pour vérifier que nos clients peuvent bien se connecter chez nous. [7]

Figure IV.2 : Architecture de nagios

IV.2.3. Les plugins

Les plugins (greffons) sont des programmes exécutables ou des scripts (perl, Shell, etc..) qui peuvent être lancés depuis une ligne de commande pour tester un hôte ou un service. Le résultat de l'exécution d'un plugin est utilisé par Nagios pour déterminer le statut des hôtes ou des services sur le réseau.

Les principaux plugins utilisés par nagios sont :

- ***check_disk*** : Vérifie l'espace occupé d'un disque dur
- ***check_http*** : Vérifie le service "http" d'un hôte
- ***check_ftp*** : Vérifie le service "ftp" d'un hôte
- ***check_mysql*** : Vérifie l'état d'une base de données MYSQL
- ***check_nt*** : Vérifie différentes informations (disque dur, processeur …) sur un système d'exploitation Windows
- ***check_nrpe***: Permet de récupérer différentes informations sur les hôtes
- ***check_ping***: Vérifie la présence d'un équipement, ainsi que sa durée de réponse
- ***check_pop***: Vérifie l'état d'un service POP (serveur mail)

check_snmp : Récupère diverses informations sur un équipement grâce au protocole SNMP.

VI.2.4 Fonctionnement de nagios

Nous pouvons distinguer deux modes de fonctionnement complémentaires de Nagios : le mode actif, ou de polling et le mode passif ou de traps.

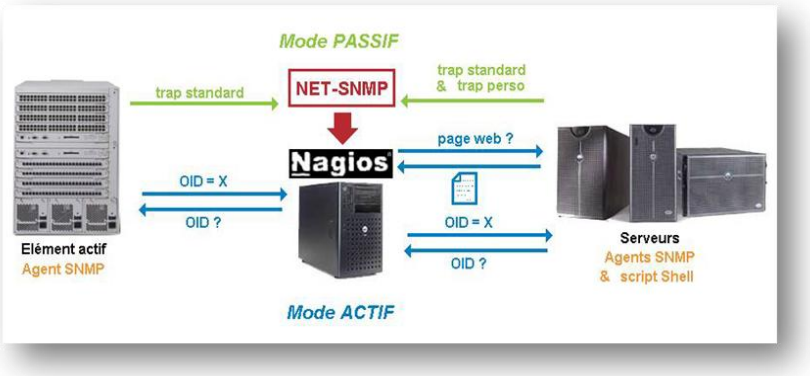

Figure IV.3 : Les deux modes de fonctionnement de Nagios

- En **mode polling**, Nagios exécute un plugin pour réaliser un test à des intervalles de temps réguliers. Il analyse ensuite la réponse et adopte un comportement en fonction de celle-ci. Ce mode de fonctionnement entraîne une génération du trafic sur le réseau.

- En **mode passif**, Nagios reste à l'écoute de tout ce qu'on peut lui dire. Pour communiquer avec lui, il suffit d'installer le programme client send_nsca sur les serveurs à superviser et de faire tourner le démon nsca sur le serveur Nagios. Dans notre configuration, c'est le démon snmptrapd de Net-SNMP qui utilise ce programme client via le script 'traitement-trap'.

Quelque soit le mode de fonctionnement, Nagios remonte des alertes aux administrateurs définis dans ses fichiers de configuration, que soit par mail, sms. Nagios met aussi en permanence à jour sont interface web qui reflète donc en temps réel l'état du réseau et des services.

Il est possible d'utiliser des agents de supervision permettant de récupérer des informations à distances. Ils offrent la possibilité de profiter de la puissance offerte par les plugins. Il existe 2 types d'agents :

- Les agents NRPE
- Les agents NCSA

Le principe de fonctionnement des *agents* **NRPE** *(pour Nagios Remote Plugin Executor)* est simple : les plugins sont installés sur l'équipement à superviser, compilés en fonction de son architecture car c'est elle qui va les exécuter, ainsi que le démon **NRPE** faisant office de serveur. Sur la plateforme de supervision Nagios, le plugin *check_nrpe* fera alors office de client nrpe, récupérant les informations en interrogeant le démon nrpe sur l'équipement concerné.

Le plugin *check_nrpe* sur le serveur Nagios initiera une connexion vers l'agent nrpe de la machine cible et lui demandera alors l'exécution d'une vérification. L'agent nrpe lancera alors le plugin configuré en local pour obtenir l'information et retournera le code retour de l'exécution ainsi que sa sortie standard.

Les *agents ncsa (pour Nagios Service Check Acceptor)* diffèrent des agents nrpe car la vérification est plannifiée en local sur l'équipement supervisé, exécutée, puis le résultat est

envoyé au serveur Nagios. De même que pour nrpe, l'architecture ncsa demande la présence du plugin *check_ncsa* sur la plateforme Nagios.

Pour notre projet, nous avons décidé d'utiliser le type de récupération active, c'est-à-dire que Nagios prend l'initiative d'envoyer une requête pour obtenir des informations. Ceci évite donc de configurer les postes à superviser. [5]

La demande d'informations se fait grâce à l'exécution d'une commande de la part de Nagios. Une commande doit obligatoirement comporter des arguments afin de pouvoir chercher les bonnes informations sur les bonnes machines.
Ces arguments sont l'adresse IP de l'hôte sur lequel aller chercher l'information, la limite de la valeur de l'information recherchée pour laquelle l'état 'attention' sera décidé, idem pour la valeur 'critique', et enfin d'autres options qui varient selon le plugin utilisé.
Pour ne pas devoir à créer une commande par machine supervisée et par information recherchée, nous pouvons remplacer les arguments par des variables, et ainsi réutiliser la commande plusieurs fois, en remplaçant la bonne variable. Nous avons alors la possibilité de travailler avec des services. Lors de la création d'un service, il faut l'associer à un ou plusieurs hôtes puis à une commande.
Ensuite Nagios remplace automatiquement la variable de l'adresse IP dans la commande, grâce à la liste d'hôtes associée au service.
Puis on doit définir manuellement dans le service les autres variables nécessaires à la commande.
Une fois que Nagios à reçu les informations dont il avait besoin sur l'état des hôtes, celui-ci peut construire des notifications sur l'état du réseau, afin d'en informer l'administrateur.
Lorsque Nagios effectue une notification, il attribut des états aux hôtes, ainsi qu'aux services.
Un hôte peut avoir les états suivants:

- ***Up*** : en fonctionnement
- ***Down*** : éteint
- ***Inaccessible***
- ***En attente***

Les différents états d'un service sont:

- ***OK***
- ***Attention***
- ***Critique***

- ***En attente***
- ***Inconnu***

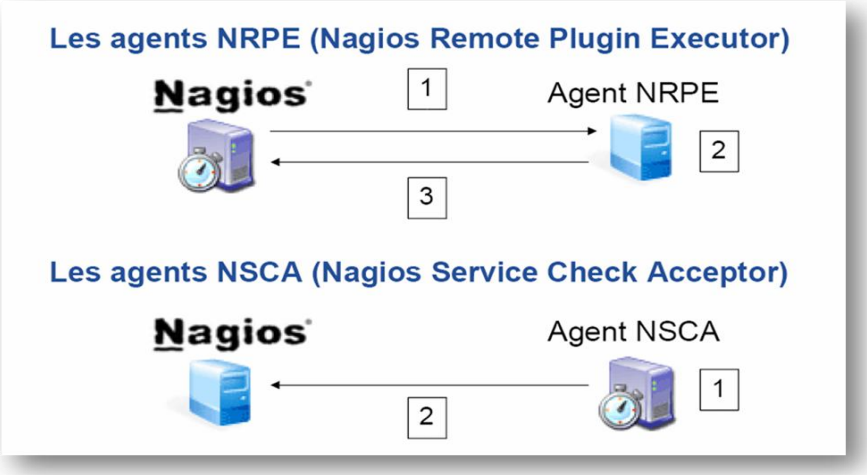

Figure IV.4 : Le fonctionnement de nagios

IV.2.4. Les fonctionnalités de nagios

Surveillance des services réseaux (SMTP, POP3, HTTP, NNTP, PING, etc.).

- Surveillance des ressources des hôtes (charge processeur, utilisation des disques, etc.).
- Système simple de plugins permettant aux utilisateurs de développer facilement leurs propres vérifications de services.
- Notifications des contacts quand un hôte ou un service a un problème et est résolu (via email, pager, ou par méthode définie par l'utilisateur).
- Possibilité de définir des gestionnaires d'évènements qui s'exécutent pour des évènements sur des hôtes ou des services, pour une résolution des problèmes
- Interface web, pour voir l'état actuel du réseau, notification et historique des Problèmes, fichiers etc.

Inconvénients

- Configuration compliquée qui oblige une très bonne connaissance de Nagios.
- Graphes pas assez clairs.
- Administration compliquée. [2]

IV.3.Supervision de serveurs Windows : NSCLient++

Nous allons décrire l'installation de NSClient, un plugin permettant de récupérer un nombre important de d'informations à surveiller sur une machine Windows.

Comme les plugins NRPE et NSCA (disponible seulement sous Linux et Mac OS X), NSClient se base sur une architecture client/serveur. La partie cliente (nommée check_nt), doit être disponible sur le serveur Nagios. La partie serveur (NSClient++) est à installer sur chacune des machines Windows à surveiller.

IV.3.1. Principe fonctionnement

- **Check_nt**

Le plugin Check_nt est un plugin récent qui permet de superviser très facilement des PC dont le système d'exploitation est Windows.

Check_nt permet de récupérer sur un système Windows les informations suivantes :

L'espace occupé sur le disque dur, le temps depuis le démarrage de l'ordinateur, la version du plugin NsClient ++, occupation du processeur, occupation de la mémoire, état d'un service.

Fonctionnement de check_nt

Lorsque Nagios veut connaître une information sur un PC, il exécute le plugin check_nt. Celui envoie une requête au PC. Sur le PC, le programme NsClient++ reçoit la requête, va chercher les informations dans les ressources du PC et renvoie le résultat au serveur Nagios.

Usage

Pour aller chercher les informations sur un PC grâce à check_nt, Nagios exécute une commande ayant la syntaxe suivante :

```
check_nt -H host -v variable [-p port] [-w warning] [-c critical][-l params]
```

Avec :

-H : Adresse IP de l'hôte à superviser

-v : ce qu'il faut superviser (ex : CPULOAD

-p : Port sur lequel il faut envoyer la requête

-w : Seuil pour lequel le résultat est considéré comme une alerte

-c : Seuil pour lequel le résultat est considéré comme critique

-l : Paramètres supplémentaires (nécessaire ou non en fonction du paramètre "v")

Pour notre projet, nous utiliserons ce plugin pour superviser tous les postes Windows sauf pour contrôler l'espace des dossiers des profils des utilisateurs. En effet, ce plugin ne permet pas d'effectuer cette vérification. Nous utiliserons un autre plugin pour cela.

- **Check_nrpe**

Le plugin Check_nrpe est un plugin qui permet de superviser des PC dont le système d'exploitation est Windows ou Linux.

Check_nrpe utilise une connexion SSL (Secure Socket Layout) pour aller chercher les informations sur les postes. Ceci permet de crypter les trames d'échanges.

Fonctionnement de check_nrpe

Lorsque Nagios veut connaître une information sur un PC, il exécute le plugin check_nrpe.

Celui envoie une requête au PC. Sur le PC, le programme NsClient++ (ou nrpe si linux) reçoit la requête, va chercher les informations dans les ressources du PC et renvoie le résultat au serveur Nagios.

Usage :

Pour aller chercher les informations sur un PC grâce à check_nrpe, Nagios exécute une commande ayant la syntaxe suivante :

```
check_nrpe -H <adresse de l'hôte à superviser> -c <nom de la commande à exécuter sur le serveur>
```

Puis sur les postes à superviser, dans le fichier de configuration (NSC.ini pour Windows, nrpe.conf pour Linux), on doit définir la commande à exécuter pour chaque nom de commande.

Exemple pour Windows :

```
Command [check_cpu]=inject checkCPU warn=80 crit=90 5 10 15
```

Exemple pour Linux:

```
Command[check_cpu]=/usr/local/nagios/libexec/check_load -w 15,10,5 -c 30,25,20
```

Ces deux commandes vérifient la charge du processeur.

On remarque alors que la mise en place de nrpe dans une grande entreprise est très complexe car il faut configurer toutes les commandes sur chaque hôte à superviser (contrairement à check_nt qui ne nécessite pas de configuration). En revanche, nrpe offre une meilleure sécurité puisque les échanges client – serveur sont sécurisés (grâce à SSL).

- **Check_snmp**

Le plugin Check_snmp est un plugin qui permet de superviser tous les équipements. En revanche, il est très instable pour superviser les PC.

Nous utiliserons check_snmp pour superviser le routeur.)

Fonctionnement de check_snmp

La MIB (Management Information Base) est une base de données sur le routeur qui stocke toutes les informations de celui-ci (statistiques, débit, état des interfaces…).

Lorsque Nagios veut connaître une information sur le routeur, il exécute le plugin check_snmp. Celui envoie une requête au routeur. Le routeur reçoit la requête, va chercher les informations dans sa MIB et renvoie le résultat au serveur Nagios.

Usage :

Pour aller chercher les informations sur le routeur grâce à check_snmp, Nagios exécute une commande ayant la syntaxe suivante :

check_snmp -H <adresse de l'hote à superviser> –o <adresse de l'information a récupérer dans la MIB> -C<communauté SNMP>

Check_ping

Le plugin Check_ping est un plugin qui permet de vérifier qu'un hôte est bien joignable.

Usage :

Pour vérifier qu'un hôte est joignable, Nagios exécute une commande ayant la syntaxe suivante :

check_ping -H <adresse de l'hôte> -w <temps maxi de réponse>,<Pourcentage de réussite des pings> -c<temps maxi de réponse>,<Pourcentage de réussite des pings>

Avec:

-w : Seuil pour lequel le résultat est considéré comme une alerte

-c : Seuil pour lequel le résultat est considéré comme critique [6]

IV.4 Conclusion

Nous avons présentés dans ce chapitre les notions de base de la supervision par Nagios qui est indispensable dans tout système d'information. Elle est à la base du bon fonctionnement d'une architecture réseau et permet de réagir rapidement en cas de problèmes ou pannes. Dans le prochain chapitre nous présentons les étapes de configuration et administration de notre projet« Nagios ».