



**Institut Universitaire de Technologie,
Aix-Marseille Université**

RAPPORT DE STAGE

**Diplôme Universitaire de Technologie
Spécialité Réseaux et Télécommunications**

Mise en place d'un serveur de supervision
NAGIOS

Drissia Chaouch

Institut FRESNEL

Responsable entreprise : Frédéric Forestier

Responsable académique : Delphine Rousseau

2017

Remerciements

Je tiens tout particulièrement à remercier Frédéric Forestier qui m'a offert une place au sein de son service informatique et m'a ainsi permis de réaliser mon stage dans de bonnes conditions étant donné les difficultés que j'ai pu rencontrer lors de la recherche de celui-ci.

Je remercie aussi Frédéric Rodriguez pour son soutien technique et son accompagnement.

Je souhaite, d'une manière générale, remercier l'ensemble des personnes présentes à l'institut Fresnel pour leur accueil toujours aussi chaleureux.

Je voudrais, d'autant plus, remercier M. Février qui m'a permis de rentrer en contact avec M. Forestier.

Table des matières

I. Introduction.....	Page 7
II. Présentation de l'Institut Fresnel.....	Page 8
i. Présentation générale.....	Page 8
ii. Organisation interne.....	Page 9
iii. Le service informatique.....	Page 10
III. Le sujet du stage.....	Page 11
i. L'énoncé du projet.....	Page 11
ii. Problématique.....	Page 11
iii. L'étude du plan réseau de l'Institut.....	Page 12
IV. Présentation du travail réalisé.....	Page 13
i. La mise en place du serveur Nagios.....	Page 13
a) Qu'est ce que Nagios ?.....	Page 13
b) Mise en place du serveur de supervision.....	Page 13
c) Configuration des notifications.....	Page 16
ii. Optimisation et automatisation du serveur de supervision.....	Page 19
a) Définition des hôtes et des services.....	Page 19
b) Extinction ordonnée de serveurs.....	Page 24
V. Bilan et conclusion.....	Page 29
VI. Bibliographie	Page 31

I. Introduction

Actuellement étudiante en deuxième année d'IUT Réseaux et Télécommunications, j'ai réalisé mon stage de fin d'études au sein de l'Institut Fresnel situé à St Jérôme, visant l'obtention de mon diplôme.

En effet, il s'est déroulé au sein du service informatique dirigé par mon tuteur M. Forestier.

Ce stage s'est parfaitement inscrit dans le cadre de mes études puisque l'on m'a confié un projet de réseau qui consistait à mettre en place une solution de supervision du type Nagios. Ainsi, j'ai eu pour missions de réunir les informations requises à la supervision réseau des équipements puis de mettre en place une solution afin d'assurer une remontée d'information rapide auprès des administrateurs.

Le stage s'est très bien déroulé avec une bonne ambiance de travail. J'ai, de plus, été capable de réaliser le projet que l'on m'avait confié avec une grande autonomie.

Nous verrons, dans un premier temps, une présentation de l'Institut Fresnel, qui a été mon lieu de travail lors de ces dix semaines de stage. Ensuite, nous nous tournerons vers le sujet de mon stage d'une manière plus explicite. Ainsi, nous enchaînerons sur le travail que j'ai pu réalisé afin d'y répondre et finalement je conclurais en réalisant un bilan de ce stage.

II. Présentation de l'Institut Fresnel

Présentation générale

L'institut Fresnel a été créé en 2000, sur le Campus de l'étoile(Marseille Nord), et résulte de la fusion de 3 laboratoires dans les domaines de la photonique, de l'électromagnétisme ainsi que du traitement du signal et des images. Afin de voir ce projet se concrétiser, les membres de l'institut ont pu compter sur le soutien de ses établissements de tutelle qui sont aujourd'hui Aix-Marseille Université, le CNRS et Centrale Marseille.

De nos jours, l'établissement gère un budget de près de 12,6 millions d'euros.

La recherche effectuée au sein de l'institut s'inscrit dans quatre thèmes : électromagnétisme et métamatériaux, nanophotonique et composants optiques, traitement de l'information et ondes aléatoires et, enfin, imagerie avancée et vivant (figure 1)

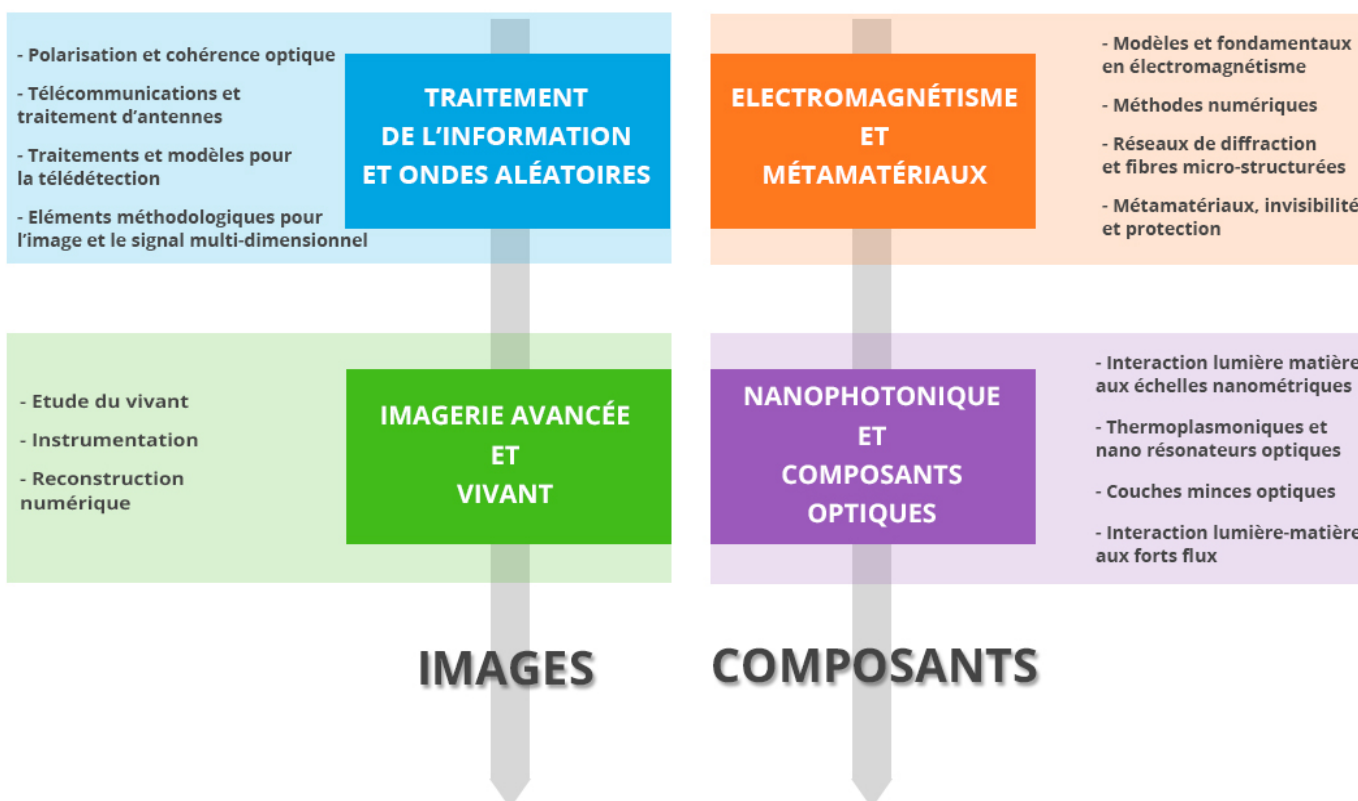


Figure 1: Thématiques de recherche

Les thèmes font ressortir deux lignes de force de la recherche menée au sein de l'Institut Fresnel autour des notions, prises au sens large, d'image (formation, traitement, applications,...) et de composants (modélisation, réalisation, caractérisation,...).

Ainsi, le projet scientifique de l'Institut Fresnel se dégage aisément autour de ces deux lignes.

II. Présentation de l'Institut Fresnel

Organisation interne

Durant ses premières années d'activités, l'institut comptait environ 74 membres dont 50 étaient permanents et 24 doctorants. Ainsi il comptait six équipes de recherche. 167 personnes travaillent à ce jour au sein de l'institut dont 52 doctorants. Ainsi le nombre d'équipes a évolué durant cette dernière période et l'institut en rassemble actuellement 13 pour 82 permanents (figure 2).

Chaque équipe travaille sur un thème de recherche. L'ILM, par exemple, a pour sujet l'interaction laser matière alors que l'équipe ATHENA va plutôt se tourner vers la modélisation linéaire et non-linéaire en électromagnétisme.

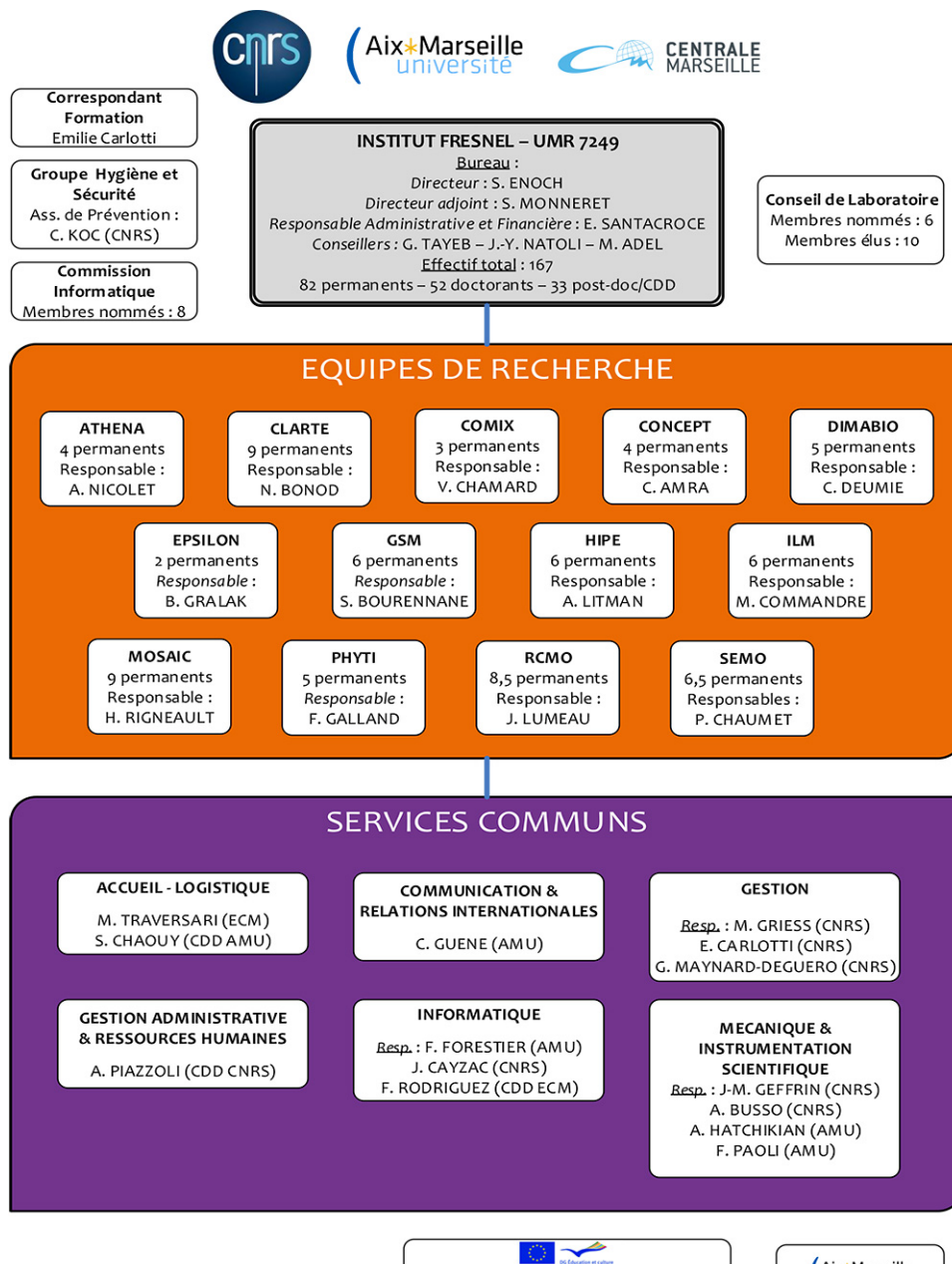


Figure 2: Organigramme de l'institut Fresnel

II. Présentation de l'Institut Fresnel

Le service informatique

Durant toute la période de mon stage, j'ai travaillé au sein du service informatique de l'institut aux côtés de Frédéric Forestier qui occupe le poste d'ingénieur et qui est à la tête de celui-ci, ainsi que Frédéric Rodriguez qui occupe un poste d'assistant informatique.

Leurs missions peuvent être de natures différentes :

- L'administration réseau et système
- La gestion des serveurs
- La sécurité
- Le support à l'utilisation des ressources informatique
- Une assistance informatique aux personnels de l'institut
- L'Installation physique des équipements

Le réseau informatique de l'Institut Fresnel, avec ses serveurs de mail et de calcul ainsi que tous les équipements de climatisation et de protection électrique est regroupé en un seul lieu. *

*Tirée du rapport d'activité 2006-2010 de l'institut.

III. Le sujet du stage

L'énoncé du projet

Le projet principal sur lequel j'ai travaillé durant ces deux mois de stage consistait à mettre en place un serveur de supervision de type Nagios afin que mes tuteurs puissent superviser depuis leur poste toutes les machines présentes sur le réseau de l'institut. De plus, en cas d'alerte ou de modification inattendue, ils devaient être notifiés par mail.

Cependant, ils possédaient déjà d'un serveur de supervision mis en place par un précédent stagiaire. Mais il n'était pas fonctionnel et certains services ne marchaient pas en plus des notifications obsolètes. Puisque je n'avais jamais travaillé sur un tel serveur, celui-ci m'a servi afin de pouvoir me familiariser avec l'outil Nagios. J'ai ensuite étudié le plan réseau de Fresnel afin de réunir les informations nécessaires au plan de supervision.

C'est ensuite que j'ai mis en place un nouveau serveur sur une nouvelle machine virtuelle, avec une organisation plus fonctionnelle, une version plus récente et des notifications qui marchent.

Cependant, le projet a évolué notamment avec la supervision d'onduleurs et de serveurs. En effet, il a fallu dans un second temps que je travaille sur l'extinction de serveurs de manière automatisée en cas de coupure de courant inattendue. De plus, il fallait ajouter une notification par sms en plus de la notification mail au préalable configurée.

Problématique

Tout bon administrateur réseau doit veiller au bon fonctionnement de ses serveurs ainsi que de ses machines. Ainsi, une vérification quotidienne de leurs états, des logs systèmes devient nécessaire. Cependant, l'administrateur, ayant d'autres missions, ne peut dédier son temps à ce type de tâches et peut ainsi apprendre un dysfonctionnement plus tard voir le lendemain par exemple ce qui peut poser un réel problème.

Ainsi, la mise en place d'un serveur de supervision lui faciliterait la tâche, garantissant une remontée d'information rapide.

De plus, il faudra s'assurer que le protocole SNMP (simple network management protocol) soit configuré sur les machines notamment les switchs qui sont présents en majorité.

III. Le sujet du stage

Étude de plan réseau de l'institut

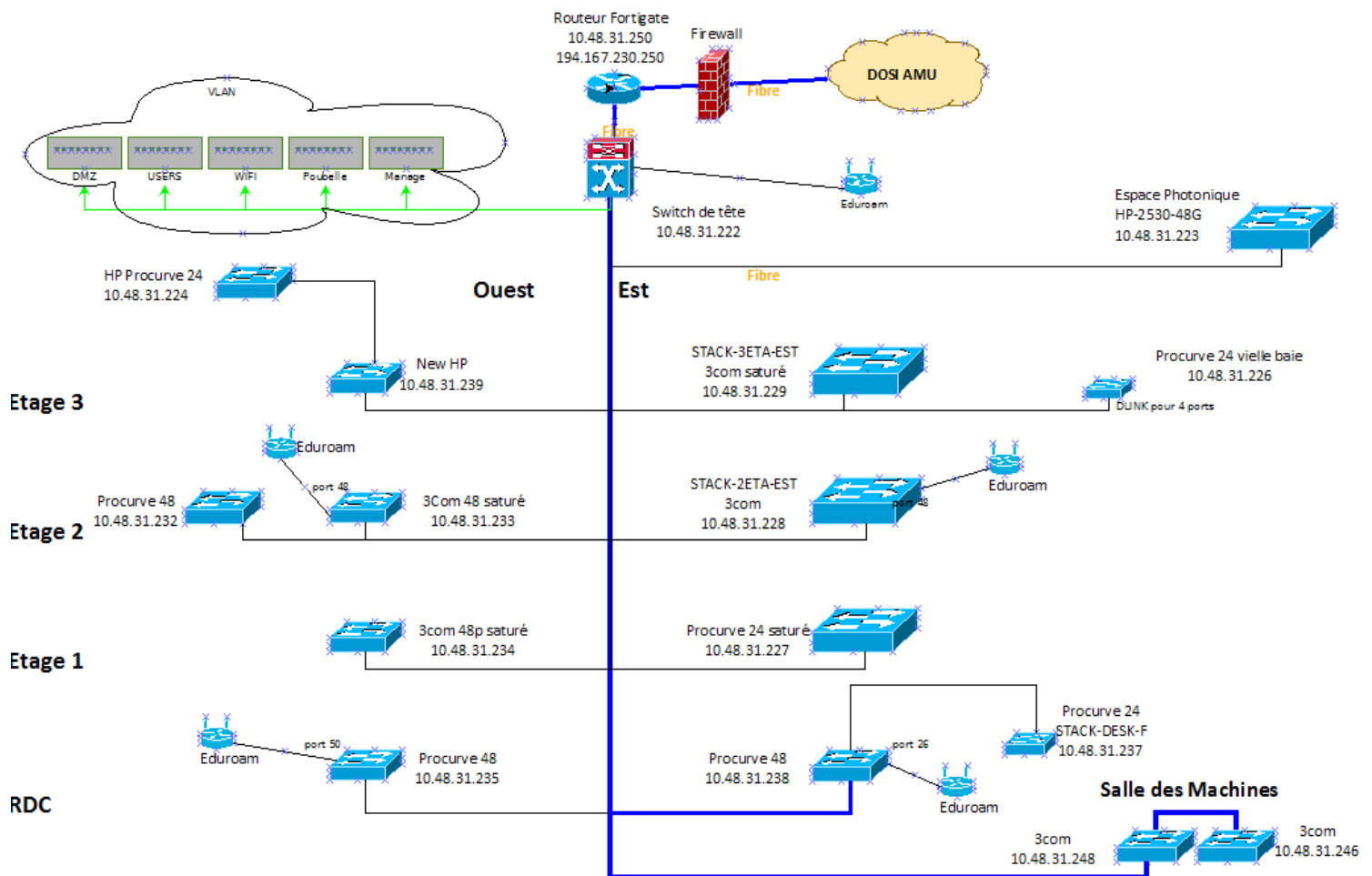


Figure 3 : Plan réseau de l'institut Fresnel

Comme dit précédemment, j'ai effectué une étude du plan réseau de l'institut afin de savoir quelles machines je devrais superviser. Ainsi, j'ai récupéré chaque adresse ip ainsi que leur position afin que je puisse, plus tard, les organiser de manière fonctionnelle à l'affichage de l'interface web.

IV. Présentation du travail réalisé

La mise en place du serveur Nagios

a) Qu'est ce que nagios ?

Nagios, appelé Netsaint jusqu'en Mai 2002, est un outil de supervision système et réseau. Il permet de surveiller l'état des services, et d'alerter l'administrateur si un problème persiste. Il est lié à une interface web accessible en réseau local, afin de pouvoir observer l'état de tous les hôtes (machines, serveurs).

b) Mise en place du serveur de supervision

Afin de pouvoir débiter l'installation du serveur, il fallait que j'installe une machine virtuelle. J'ai pu créer une machine virtuelle de par le proxmox, déjà configuré au sein du réseau de l'institut, qui est une solution de virtualisation libre. J'ai choisi une machine virtuelle Ubuntu 16.10 puisqu'elle est compatible avec le serveur et son utilisation reste simple. J'ai ensuite utilisé le logiciel mobaXterm, qui permet d'utiliser sous windows les commandes Unix/Linux, sur lequel je me suis connectée en SSH (secure shell, permet des connexions à distance sécurisées).

Afin de débiter l'installation de nagios j'ai récupéré le lien de téléchargement que voici :

```
#wget https://sourceforge.net/projects/nagios/files/nagios-4.x/nagios-4.3.1/nagios-4.3.1.tar.gz/download
```

Le format étant .tar.gz j'ai dû extraire le fichier au moyen de la commande :

```
#tar xzf nagios-4.3.1.tar.gz
```

Avant de continuer l'installation, j'ai pu constater qu'il fallait installer certains pré-requis pour le bon fonctionnement du serveur, ce que j'ai donc fait :

```
#apt-get update
```

```
#apt-get install -y autoconf gcc libc6 make wget unzip apache2 php libapache2-mod-php7.0 libgd2-xpm-dev
```

En effet, un paquet tel que Apache 2 sera nécessaire à l'affichage de l'interface web par exemple.

Ensuite, afin de finaliser l'installation, il faut lancer la compilation avec ces commandes :

```
# ./configure --with-httpd-conf=/etc/apache2/sites-enabled
```

```
#make all
```

```
#make install Cette étape installe les fichiers binaires, les CGI et les fichiers HTML
```

IV. Présentation du travail réalisé

b) Mise en place du serveur de supervision

Il faut ensuite créer l'utilisateur ainsi que le groupe Nagios :

```
#useradd nagios
```

```
#usermod -a -G nagios www-data
```

La structure des fichiers de configuration s'organise de la façon suivante :

- Les fichiers de configuration général

```
# /usr/local/nagios/etc
```

On y retrouve nagios.cfg par exemple.

- Localisation des plugins

```
# /usr/local/nagios/libexec
```

- Fichiers de configuration nagios

```
#/usr/local/nagios/etc/objects
```

Voici la liste des fichiers que l'on peut y trouver :

```
commands.cfg contacts.cfg localhost.cfg printer.cfg server.cfg switch.cfg  
templates.cfg thermometre.cfg timeperiods.cfg ups.cfg windows.cfg
```

Je citerais par la suite les fichiers que j'ai utilisé et dans quel cas.

Le serveur étant configuré sur l'adresse 10.48.0.73, il faut taper <http://10.48.0.73/nagios> afin de se rendre sur l'interface Web, ce qui permet d'avoir une vue d'ensemble sur la supervision des équipements.

- Configuration des utilisateurs

Une fois l'installation réussie, j'ai pu me consacrer à la configuration des utilisateurs nagios. En effet, on peut définir un ou plusieurs administrateurs en y ajoutant notamment leurs adresses mail afin qu'ils puissent recevoir les alertes.

IV. Présentation du travail réalisé

b) Mise en place du serveur de supervision

Afin d'y parvenir, il faut éditer le fichier **contacts.cfg** qui possède déjà une configuration d'origine. Ainsi, je l'ai modifié afin qu'il corresponde à nos besoins. La configuration d'un utilisateur s'effectue de la manière suivante :

```
define contact{
    contact_name          fred
    use                   generic-contact
    alias                 Frédéric Forestier
    host_notifications_enabled 1
    service_notifications_enabled 1
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,u,r
    service_notification_commands notify-service-by-email
    host_notification_commands notify-host-by-email
    email                 f.forestier@fresnel.fr
    can_submit_commands 1
    contactgroups         admins
}
```

Il s'agit ici de la définition de l'utilisateur Frédéric Forestier qui est à la tête du service informatique. On peut voir en rouge les éléments nécessaires à la notification par mail dont on étudiera la configuration plus tard.

Dans ce même fichier on retrouve aussi le groupe administrateur. On peut en créer plusieurs mais je n'en ai créé qu'un seul qui regroupe les deux utilisateurs.

Ex :

```
define contactgroup{
    contactgroup_name    admins
    alias                Nagios Administrators
    fred, fred2 }
members
```

IV. Présentation du travail réalisé

c) Configuration des notifications

Les notifications sont cruciales au bon fonctionnement du serveur. En effet, elles sont à l'origine d'une remontée d'informations rapide et efficace auprès des administrateurs.

J'ai tout d'abord installé Postfix, puisqu'il va me servir à mettre en place le serveur mail. En effet, Postfix est un serveur de messagerie qui permet d'acheminer les courriels électroniques.

Afin récupérer le paquet, il faut simplement taper : **#apt-get install postfix**.

Ensuite, j'ai édité le fichier de configuration *main.cf* afin de le modifier pour l'adapter à mon environnement de travail.

```
##### Extrait du fichier
# Config general
config_directory = /etc/postfix
mydomain = fresnel.fr
myhostname = supervision
mydestination = localhost.localdomain, localhost, fresnel.fr
myorigin = fresnel.fr
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
relayhost = smtp.fresnel.fr
mynetworks = 127.0.0.1/32, 10.48.0.0 /24, 194.167.230.0/24
mailbox_size_limit = 0
recipient_delimiter = +
inet_protocols = ipv4
```

Du côté de Nagios, il faut éditer le fichier *commands.cfg* afin qu'il puisse envoyer des mails aux administrateurs.

IV. Présentation du travail réalisé

c) Configuration des notifications

Il faut savoir que lorsque j'avais fini ma configuration dans un premier temps, les mails ne marchaient toujours pas. Ainsi, après quelques recherches, je me suis rendu compte qu'il manquait le répertoire /usr avant le /bin/mail qui était déjà défini (voir l'extrait ci-dessous).

```
# 'notify-host-by-email' command definition
define command{
    command_name    notify-host-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type:
$NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState: $HOSTSTATE$\nAddress:
$HOSTADDRESS$\nInfo: $HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME$\n" |
/usr/bin/mail -s "*** $NOTIFICATIONTYPE$ Host Alert: $HOSTNAME$ is $HOSTSTATE$
***" $CONTACTEMAIL$
}

# 'notify-service-by-email' command definition
define command{
    command_name    notify-service-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type:
$NOTIFICATIONTYPE$\n\nService: $SERVICEDESC$\nHost: $HOSTALIAS$\nAddress:
$HOSTADDRESS$\nState: $SERVICESTATE$\n\nDate/Time:
$LONGDATETIME$\n\nAdditional Info:\n\n$SERVICEOUTPUT$\n" | /usr/bin/mail -s "***
$NOTIFICATIONTYPE$ Service Alert: $HOSTALIAS$/$SERVICEDESC$ is
$SERVICESTATE$ ***" $CONTACTEMAIL$
}
```

La commande est définie deux fois, une pour les hôtes et la seconde pour les services. La notification a finalement été opérationnelle suite à ce petit changement.

Chaque définition d'hôtes et de services comporte une option `< contact_groups >` qui spécifie les groupes de contact qui reçoivent des notifications pour cet hôte ou service particulier.

IV. Présentation du travail réalisé

c) Configuration des notifications

Voici un mail envoyé par Nagios pour nous prévenir qu'un problème est présent sur l'un des équipements du réseau. Il s'agit d'un mail que j'ai reçu (figure 4).

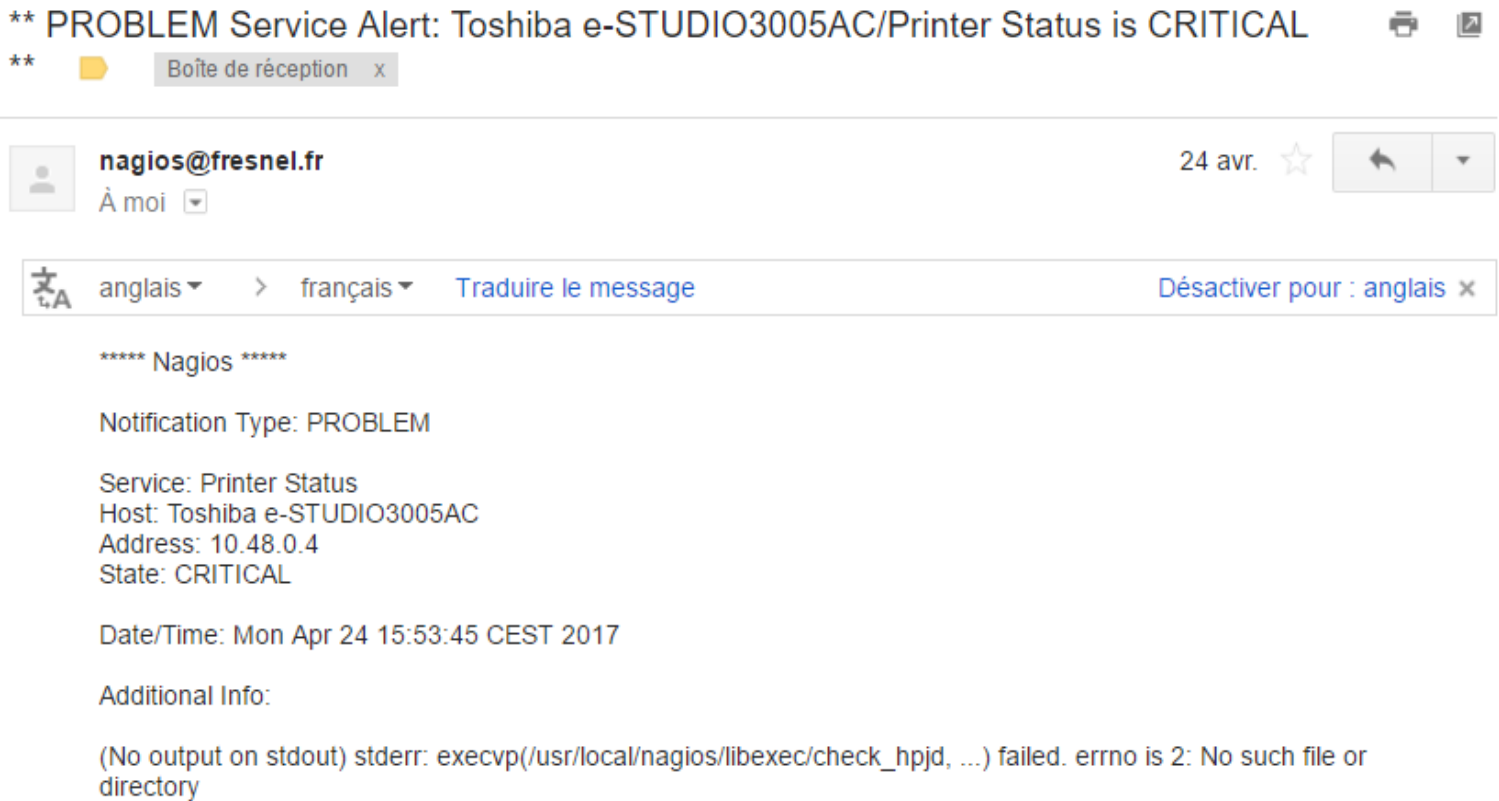


Figure 4 : Mail d'alerte nagios

Le problème ici vient d'un plugin associé à un service. En effet, celui-ci n'est pas reconnu et ne récupère pas l'information voulue, ainsi cela génère une erreur.

IV. Présentation du travail réalisé

Optimisation et automatisation du serveur de supervision

a) Définition des hôtes et des services

Après avoir étudié le plan réseau de l'institut et récupérer les informations nécessaires à leur supervision, je les ai défini dans des fichiers de configuration.

Il faut encore une fois se rendre dans /objects.

Cependant, le fichier de configuration choisi va dépendre de l'élément que l'on veut ajouter ou modifier.

Pour les switches et routeurs, il faut éditer switch.cfg.

Pour les serveurs, il faut éditer server.cfg.

Pour les imprimantes, il s'agit de printer.cfg.

Pour les sondes de température, thermometre.cfg.

Et pour finir les onduleurs, ups.cfg.

On prend l'exemple des sondes de température.

Lorsque l'on édite le fichier de configuration, on trouvera d'abord la définition des hôtes. On y définit notamment le nom de l'équipement ainsi que son adresse ip.

```
define host{
    use                generic-switch
    host_name          Sonde
    alias              Sonde température
    address            10.48.31.10
    notification_interval 1440
    notification_period 24x7
    notification_options d,u,r
    hostgroups         Sondes
    check_command      check-host-alive
    check_interval     5
    retry_interval     1
    max_check_attempts 5
    check_period       24x7
    contact_groups     admins
    notifications_enabled 1 }
```

IV. Présentation du travail réalisé

a) Définition des hôtes et des services

Dans le même fichier on retrouve le groupe.

```
define hostgroup{
    hostgroup_name Sondes ; The name of the hostgroup
    alias           Sonde T ; Long name of the group
}
```

Enfin, on y définit les services qui vont permettre de récupérer une information spécifique. Ici, on veut récupérer le niveau de température de la sonde. En effet, elle mesure la température de la salle des machines et l'administrateur veut être alerté si la température dépasse les 26 degrés.

```
define service {
    host_name           Sonde
    service_description Temperature
    check_command       check_ste-hwg!public!215
    use                 generic-service,nagiosgraph
    normal_check_interval 5 ; Check the service eve$
    retry_check_interval 1 ; Re-check the service $
    max_check_attempts 5
    check_period        24x7
    notification_interval 1440
    notification_period 24x7
    notification_options w,c,r
    notifications_enabled 1
    contact_groups      admins
    event_handler_enabled 1
    event_handler        send_sms
}
```

IV. Présentation du travail réalisé

a) Définition des hôtes et des services

Lorsque l'on définit un nouveau service, il faut toujours veiller à ce que la commande existe bien dans `commands.cfg`, puisque ce fichier regroupe toutes les commandes que va utiliser le serveur nagios.

Pour le service défini précédemment, la commande correspondante est :

```
#Check-hwg command definition
define command {
    command_name    check_ste-hwg
    command_line    $USER1$/ check_ste-hwg -C $ARG1$ -H
                   $HOSTADDRESS$ -S $ARG2$    }

```

\$USER1\$ correspond au chemin `/usr/local/nagios/libexec`

\$HOSTADDRESS\$ récupère l'adresse ip de l'hôte

\$ARG1\$ récupère le premier argument défini dans le service qui est ici `public`

\$ARG2\$ récupère le deuxième argument défini dans le service qui est ici `215`

Le script `check_ste-hwg` était présent dans `/usr/local/nagios/libexec`. Je n'ai pas réalisé le script des plugins puisque Nagios en propose en libre téléchargement dans sa librairie.

Finalement, après avoir défini chaque aspect de la configuration, je vérifie mes changements et je relance le service pour les appliquer au moyen de ces commandes :

```
# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```


```
# service nagios restart
```

Nous pouvons à présent observer de par l'interface web l'état de l'hôte ainsi que de ses services (figure 5).

IV. Présentation du travail réalisé

a) Définition des hôtes et des services

Limit Results: 100

Host	Service	Status	Last Check	Duration
Sonde	HTTP	OK	06-09-2017 11:12:08	0d 16h 53m 39s
	PING	OK	06-09-2017 11:11:55	0d 16h 53m 53s
	Temperature 	OK	06-09-2017 11:12:38	0d 16h 53m 11s

Attempt	Status Information
1/5	HTTP OK: HTTP/1.1 200 Ok - 3161 octets en 0,362 secondes de temps de réponse
1/5	PING OK - Paquets perdus = 0%, RTA = 4.55 ms
1/5	Sensor: Sensor 215, State: normal, Value: 22.8

Figure 5 : Vu de l'hôte et de ses services depuis l'interface web

Ainsi, on relève bien la température dans la partie « Status Information » qui est de 22,8 degrés. Son statut est donc OK.

Les statuts peuvent être différents selon la nature de l'alerte. Il peut être « unknown », c'est à dire que le serveur ne va pas reconnaître une configuration par exemple. Le statut peut devenir en mode « warning » s'il détecte un problème mais que le nombre de relance maximum n'a pas encore été atteint. Enfin, il peut être « critical » si le problème persiste ou si un certain seuil défini a été franchi.

**Pour aller un peu plus loin*

Comme on peut l'observer à droite du service « Température » sur la figure 5, il y a un icône de graphes. Effectivement, l'ajout d'un graphe permettrait d'avoir une vue d'ensemble sur les différents changements encourus pour une période. Ainsi, je me suis documentée à ce propos et je me suis tournée dans un premier temps vers la solution PNP4Nagios qui est un programme permettant de visualiser des graphes de performances.

Cependant, je n'ai pas réussi à aboutir avec cette solution, mes graphiques ne s'affichant toujours pas. Je me suis donc dirigée vers une autre solution NagiosGraph.

L'installation s'est bien déroulée et, du côté de Nagios, il fallait simplement ajouter la commande, toujours dans `commands.cfg` ainsi qu'une définition dans le fichier `templates.cfg`.

IV. Présentation du travail réalisé

a) Définition des hôtes et des services

- *Commands.cfg*

```
define command {  
command_name process-service-perfdata-for-nagiosgraph  
command_line /usr/local/nagiosgraph/bin/insert.pl    }
```

- *Templates.cfg*

```
define service {  
name nagiosgraph  
action_url /nagiosgraph/cgi-bin/show.cgi?host=$HOSTNAME&service=$SERVICEDESC$  
register 0    }
```

Ainsi pour ajouter la vue du graphe au service « Température », il m'a fallu ajouter la directive `nagiosgraph` à la suite de `generic-service` comme ci-dessous.

```
define service {  
[...]  
use generic-service,nagiosgraph  
[...]  
}
```

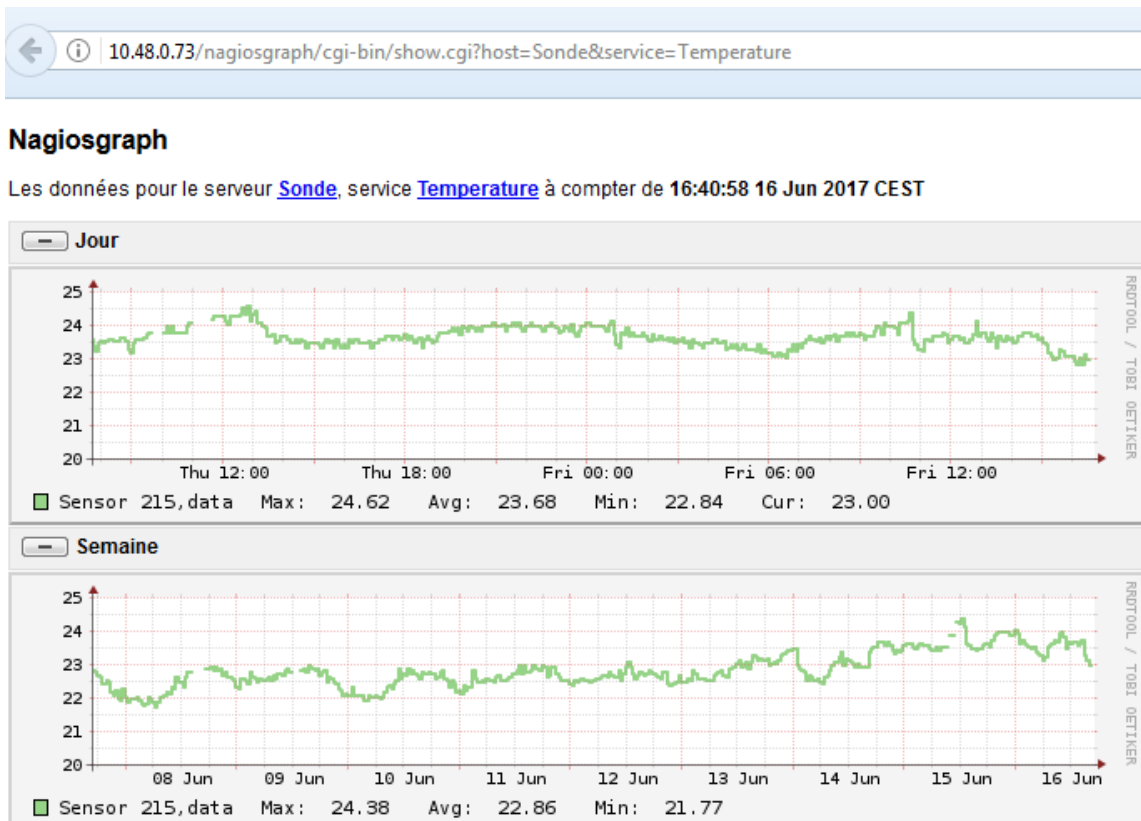


Figure 6 : Graphiques donnant les niveaux de température atteints par la sonde

IV. Présentation du travail réalisé

b) Extinction ordonnée de serveurs

J'ai par la suite été amenée à superviser des serveurs. Les étapes de la configuration restent les mêmes, cependant, afin de récupérer des données du serveurs, il est nécessaire d'installer NRPE (Nagios Remote PluginExecutor). En effet, celui-ci permet d'exécuter des plugins sur une machine distante. Nrpe doit être installé sur les deux serveurs, celui de nagios et celui que l'on souhaite superviser. Mon tuteur s'est chargé de l'installation sur les serveurs. J'ai donc juste installé le paquet NRPE et défini les serveurs dans les fichiers de configuration nagios. Ainsi, cela a rendu possible la supervision du serveur mail par exemple (figure 7).

Host	Service	Status	Last Check
Serveur Mail	CPU Load	OK	06-16-2017 16:36:17
	Check SMTP	OK	06-16-2017 16:34:47
	Disk space	OK	06-16-2017 16:35:47
	Mail queue	OK	06-16-2017 16:35:17
	Total Processes	WARNING	06-16-2017 16:33:47
	Zombie Processes	OK	06-16-2017 16:37:17
Duration	Attempt	Status Information	
37d 23h 40m 49s	1/5	OK - load average: 0.48, 0.49, 0.76	
37d 4h 53m 22s	1/5	SMTP OK - 0.001 sec. response time	
37d 23h 41m 16s	1/5	DISK OK - free space: / 4269126 MB (83% inode=99%):	
10d 23h 52m 1s	1/5	OK: postfix mailq (54) is below threshold (100/200)	
0d 7h 22m 50s	5/5	PROCS WARNING: 575 processes	
37d 23h 36m 59s	1/5	PROCS OK: 0 processes with STATE = Z	

Figure 7 : Supervision du serveur mail

Ayant exploré les possibilités qu'offre ce serveur de supervision, il est en effet possible de déclencher un événement pour un changement d'état. Ce sont les Event Handlers, une option de gestionnaire d'événements proposé par nagios, qui va permettre ce processus.

Ce que j'ai souhaité réalisé est le fait d'ordonner l'extinction d'un serveur en cas de panne de courant détectée par l'onduleur qui l'alimente.

Ayant déjà mis en place une solution de supervision pour les onduleurs, je devais trouver le moyen de faire passer l'information au serveur sur lequel je souhaite exécuter le script.

Une étude plus approfondie des Event handlers m'a permise de me familiariser avec ce processus et de mieux en comprendre la configuration.

IV. Présentation du travail réalisé

b) Extinction ordonnée de serveurs

Nagios offre un script de base, qu'il faut ensuite modifier selon nos besoins. J'ai récupéré le script qu'il faut placer dans `/usr/local/nagios/libexec/eventhandlers`. En voici un extrait qui correspond à la partie que je devais moi modifier.

```
CRITICAL)
  case "$2" in
    SOFT)
      case "$3" in
        3)
          echo -n "Arrêt du serveur (3rd soft critical state)..."
          # Call the init script to stop the server
          /usr/local/nagios/libexec/check_nrpe -H 10.48.31.37 -p 5666 -c stop_server

          ;;
        HARD)
          echo -n "Arrêt du serveur (3rd soft critical state)..."
          # Call the init script to stop the server
          /usr/local/nagios/libexec/check_nrpe -H 10.48.31.37 -p 5666 -c stop_server

          ;;
      esac
    esac
  ;;
esac
exit 0
```

*ip du serveur à stopper en cas d'alerte critique

*script d'arrêt du serveur

Le script va s'exécuter si l'état du service reste critique au bout de la troisième relance. Sachant que j'ai défini une intervalle d'une minute entre chaque vérification, il faut compter 3 minutes avant que le script ne s'exécute.

La commande `check_nrpe` va ici permettre d'exécuter à distance le script d'arrêt présent sur le serveur à éteindre.

IV. Présentation du travail réalisé

b) Extinction ordonnée de serveurs

J'ai écrit le script `stop_server` de manière à ce qu'il annonce une extinction du serveur dans la minute qui suit afin que les utilisateurs puissent être avertis. De plus, il envoie un sms pour prévenir les administrateurs nagios (figure 8).

- Script `Stop_server`

```
#!/bin/bash
sudo shutdown -h +1 "le serveur va s'arrêter dans une minute... " ; sudo
wget https://smsapi.free-mobile.fr/sendmsg?user=10020428&pass=6W5qtXhjoPkr0V\&msg=Coupure%20de%20courant:%20Le%20serveur%20Opto48%20va%20s%20arreter%20dans%20une%20minute! -P
/usr/lib/nagios/plugins/sms/
```

Message
ven. 19 mai à 10:25

Coupure de courant: Les serveurs vont sarreter dans une minute!

Figure 8 : Une alerte SMS que j'ai reçu durant mes tests

Ce script a été ensuite associé au service de l'onduleur qui va récupérer l'information concernant la coupure de courant . Pour cela, il faut ajouter les lignes :

```
define service{
[...]
```

<code>enable_event_handlers</code>	<code>1</code>
<code>event_handler</code>	<i>*nom de la commande*</i>

```
[...] }
```

IV. Présentation du travail réalisé

b) Extinction ordonnée de serveurs

Il ne faut pas oublier de définir la commande dans `commands.cfg`.

```
define command{
    command_name    *nom de la commande*
    command_line    /usr/local/nagios/libexec/eventhandlers/*nom du script*
                   $SERVICESTATE$ $SERVICESTATETYPE$ $SERVICEATTEMPT$
}
```

De cette manière, j'ai pu mettre en place un automatisme qui permet de régler un problème sans l'intervention de l'administrateur ce qui peut-être très utile s'il n'est pas présent sur son lieu de travail lorsque l'incident a lieu.

Ordonner l'extinction de serveurs dans une telle situation permettra d'autant plus de garantir une protection des données.

V. Bilan et Conclusion

Durant ces deux mois de stage, j'ai pu découvrir un autre aspect de l'administration réseau, un aspect différent de celui que j'ai pu étudier à l'IUT. En effet, n'ayant jamais fait de la supervision réseau, ce projet semblait tout nouveau pour moi. Lors de ces deux dernières années d'études, j'ai pu acquérir les connaissances nécessaires à la configuration de routeurs ainsi que de switchs de par de nombreux protocoles par exemple. Mais la supervision réseau n'a pas été un sujet sur lequel nous avons travaillé. Ainsi, j'y ai trouvé l'occasion d'acquérir de nouvelles compétences dans ce domaine.

Je me suis, dans un premier temps, beaucoup documentée afin de me familiariser avec ce type d'outil. Ce n'est pas quelque chose qui m'a semblé compliqué étant donné que je possédais tout de même des connaissances en réseau. De plus, toutes les configurations se faisaient de par une machine virtuelle linux, ce que je maîtrisais déjà grâce à l'enseignement que j'ai reçu. Cependant, ce qui m'a semblé nouveau pour moi était de travailler avec du linux afin de récupérer des informations sur des éléments présents sur le réseau. J'ai ainsi découvert toutes ces possibilités qui m'étaient inconnues auparavant.

J'ai finalement pu créer une vision globale de l'état des serveurs et équipements réseaux présents au sein de l'institut (Voir annexe 1). Il y en d'autres encore que mon tuteur souhaite ajouter par la suite. Je lui ai donc fait une petite documentation réunissant les éléments à savoir afin de pouvoir configurer le serveur nagios.

D'un point de vue personnel, je me suis trouvée une autonomie que je ne pensais pas autant avoir. De plus, j'ai aimé être confrontée à des problèmes afin de trouver les solutions nécessaires à leurs résolutions.

Finalement, ce stage a pu me permettre d'acquérir de nouvelles compétences mais aussi d'en apprendre plus sur moi-même et mes capacités. Cela m'a conforté dans les choix que je souhaite entreprendre pour mon avenir professionnel, me diriger vers les métiers d'innovation et de développement.

VI. Bibliographie

Lien principal de la documentation nagios

- <https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/toc.html>

Liens de documentation qui ont complété mes recherches

- http://www.hw-group.com/software/Nagios/download/Nagios_HWg-STE_en.pdf
- <https://doc.ubuntu-fr.org/postfix>
- <https://wiki.monitoring-fr.org/infra/postfix>
- <https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/notifications.html>
- <http://www-igm.univ-mlv.fr/~dr/XPOSE2004/nchaveron/Supervision.html#marche>
- <http://www.linuxfunda.com/2013/04/02/steps-to-configure-nagiosgraph-with-nagios-core/>
- <https://sourceforge.net/projects/nagios/files/nagios-4.x/>

Informations concernant l'institut Fresnel

- <http://www.fresnel.fr>