Institut Universitaire de Technologie,
Aix-Marseille Université

RAPPORT DE STAGE
Diplôme Universitaire de Technologie
Spécialité Réseaux et Télécommunications

Administration systèmes et réseaux

Maxime JULIEN

Institut de Mathématiques de Marseille

Responsable entreprise : Olivier CHABROL

Responsable académique : Tin NGUYEN

2018

Remerciements

Je tenais à remercier l'équipe du service informatique du site de Château Gombert pour son accueil et son soutien technique. J'ai pu apprendre beaucoup grâce à celle-ci, non seulement en connaissance et compétence mais aussi en termes de rigueur.

Plus particulièrement, j'aimerais remercier Olivier CHABROL pour sa disponibilité et son engagement durant mon stage.

Je remercie également Pierre BARTHELEMY et Jean-Bruno ERISMANN pour m'avoir donné la possibilité de travailler avec le service informatique du site de Luminy.

Merci à la DOSI et plus particulièrement à Pascal MOURET, de m'avoir permis de visiter leur data center et d'avoir répondu à toutes mes questions sur celui-ci.

Je remercie de même M. NGUYEN pour son encadrement pendant celui-ci ainsi que M. FEVRIER pour m'avoir mis en contact avec l'Institut de Mathématique de Marseille.

De façon générale je remercie la Direction d'avoir accepté ma candidature en tant que stagiaire dans leur institut et de m'avoir permis de ce fait de vivre une expérience très enrichissante.

Table des matières

I. Introduction

A. Présentation de l'entreprise

Dans le cadre de mon **DUT***, j'ai réalisé un stage de dix semaines en entreprise. J'ai effectué ce dernier au sein le service informatique de l'**I2M***, principalement sur le site de Château Gombert.

L'I2M est une Unité Mixte de Recherche composée du **CNRS***, de l'université d'Aix-Marseille et de l'école Centrale Marseille. Il compte environ cent trente enseignants-chercheurs, une trentaine de chercheurs CNRS, une quinzaine de personnels techniques et administratifs, une soixantaine de doctorants et une vingtaine de chercheurs post-doctorants. L'institut est issu de la fusion, au 1 janvier 2014, du LATP (Laboratoire d'Analyse, Topologie et Probabilités) et de l'IML (Institut de Mathématiques de Luminy). L'I2M est principalement localisé sur trois sites : le Centre de Mathématiques et Informatique à Château Gombert et le campus de Luminy sont les deux sites principaux, le dernier site est celui du campus St Charles.

L'I2M est partie prenante de nombreux projets de recherches et actions nationaux et internationaux, et partenaire du Laboratoire d'Excellence Archimède, de l'AMIDEX et de la Chaire Jean-Morlet. Il est également membre de la Fédération de Recherche des Unités de Mathématiques de Marseille, de la Société Mathématique de France et de la Société de Mathématiques Appliquées et Industrielles.

Le service informatique est donc lui aussi reparti sur ces trois sites et comporte cinq personnes : une est uniquement sur Château Gombert, Augustino DE SOUZA ; deux autres sont basées sur Luminy, Pierre BARTHELEMY et Jean-Bruno ERISMANN ; enfin deux alternent entre les trois sites, Guillaume CHAGNARD et Olivier CHABROL, notre tuteur de stage, responsable du service informatique. Vous trouverez ci-dessous l'organigramme général de l'I2M avec encadré en rouge, le service informatique (Figure 1).

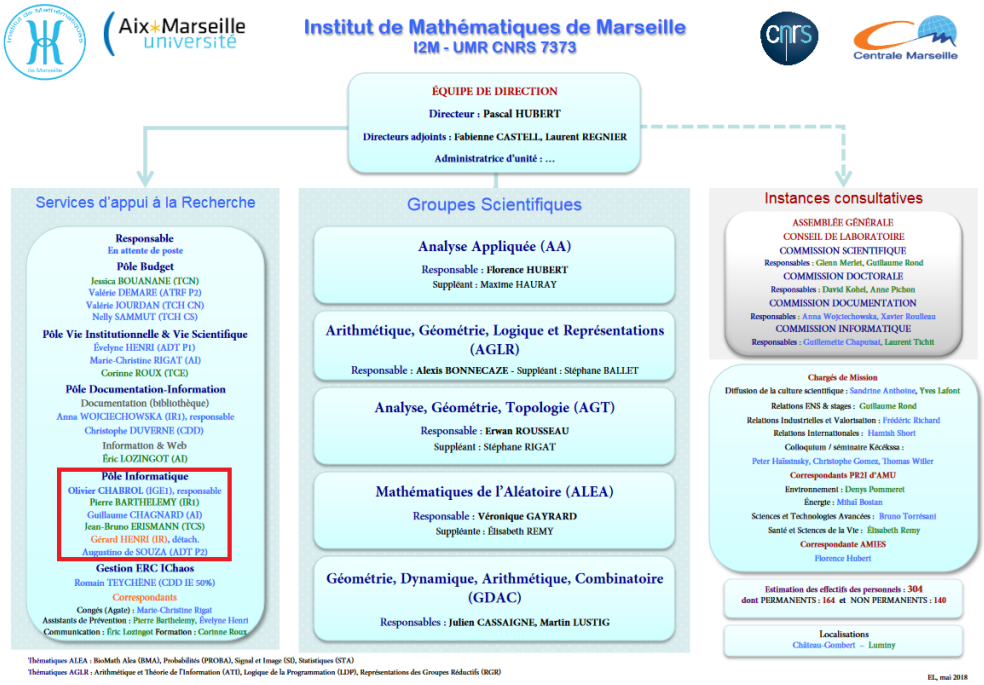

Figure 1 : Organigramme général de l'I2M

B. Présentation du cadre technique

Durant mon stage, j'ai veillé au bon fonctionnement du parc informatique à travers différentes missions, que je détaillerai plus tard dans ce rapport, reposant en grande partie sur mes connaissances d'administration systèmes et réseaux mais aussi sur la rédaction de documentation.

Pour mieux comprendre les missions qui m'ont été désignées, il faut savoir que le responsable actuel du service informatique est présent dans l'institut depuis seulement deux ans et a hérité d'un parc vieillissant et résultant de la fusion de trois anciens laboratoires possédants chacun leurs propres service informatique et politique de gestion et de sécurité. De plus, certains équipements comme des serveurs ou des commutateurs sont partagés entre le service informatique de l'I2M et la **DOSI***, ce qui complexifie la gestion de ces derniers.

Pour information, le parc informatique du site de Château Gombert est composé à l'heure actuelle de 66% d'ordinateurs sous OS X (Apple), de 32% d'ordinateurs sous Unix et de 2% d'ordinateurs sous Windows.

II. Présentation du travail réalisé

A. Descriptif des missions

La quasi-totalité de mon stage s'étant déroulée sur site de Château Gombert, la plupart des missions que je vais détailler ont été effectué là-bas. Sur ce site, il y a beaucoup de passage. Tout autant par rapport à des échanges scientifiques, à des colloques, à des conférences ou encore à des arrivées de stagiaires ou de doctorants. Cela implique des missions quotidiennes pouvant aller de l'assistance d'un utilisateur à utilisation une imprimante à la mise à jour de licences logicielles ou encore à l'installation ou réinstallation de postes. En dehors de ces missions, il m'a été confié des tâches plus importantes. Ces dernières sont :

- Le dépannage du serveur de la bibliothèque
  Une partie des utilisateurs n'y avait pas accès.

- La mise à jour des commutateurs HP du site de Château Gombert
  Les commutateurs de ce site n'avaient pas été mis à jour depuis plusieurs années, ce qui posait problème d'un point de vu sécuritaire.

- L'installation d'un scanner de vulnérabilité : **OpenVAS***
  Une fois les mises à jour faites, il fallait vérifier qu'il n'y ait pas d'autres problèmes de sécurité.

- L'installation d'un logiciel de **supervision*** du réseau : **Shinken***
  Le réseau n'était pas correctement supervisé alors j'ai mis en place Shinken.

- La rédaction d'une documentation sur le déploiement de client **LDAP*** avec **Autofs***
  Dans le contexte d'un basculement des serveurs de calculs d'un **NIS*** ancien vers le serveur LDAP, il a fallu faire des tests de configuration pour le bon fonctionnement des clients LDAP.

- Raccordement du site de Luminy sur la boucle optique
  La boucle optique sur Luminy étant finie, j'ai participé aux différentes réunions à propos du raccordement de l'I2M à celle-ci ainsi qu'au raccordement lui-même.

- Mise en service d'un nouveau serveur **DHCP*** et d'inventaire sur le site de Luminy
  L'objectif était de remplacer une ancienne machine qui servait de serveur DHCP et d'inventaire.

Mathéo KORADJIAN et moi-même avons effectué notre stage de fin d'étude dans le même service. C'est pour cela qu'il sera cité lors du détail de certaines missions.

B. Dépannage du serveur de la bibliothèque

La première mission qui nous a été confié visait à trouver la raison pour laquelle certains utilisateurs sur le réseau ne pouvaient plus accéder au serveur de la bibliothèque depuis la mise en place d'un pare-feu. Cette tâche nous a permis de découvrir le plan d'adressage du site de Château Gombert qui est un peu particulier. En effet, celui-ci est entièrement constitué d'adresses publiques sur un réseau en 147.94.64.0/23. On retrouve donc deux groupes d'adresses facilement discernables, les adresses qui sont en 147.94.64.X et celles en 147.94.65.X que l'on nommera respectivement les adresses en 64 et en 65.

Le problème était que les utilisateurs ayant une adresse en 65 ne pouvait pas accéder au serveur qui lui avait une adresse en 64 et ce, uniquement depuis la mise en place d'un pare-feu. Il était donc supposé dans le service informatique qu'une erreur dans la configuration du pare-feu était à l'origine du problème. J'avais personnellement comme avis que, le plan d'adressage étant un peu particulier, le problème venait de la configuration des paramètres réseaux du serveur.

Afin de vérifier la première hypothèse, j'ai récupéré une adresse en 65 sur mon poste et j'ai essayé d'accéder tout d'abord au dit serveur puis à un autre serveur ayant une adresse en 64. Le résultat fut celui que j'attendais : le premier serveur était bien inaccessible mais ce n'était pas le cas du second. Cela m'a donc conforté dans mon idée, j'ai donc vérifié la configuration du serveur et l'erreur venait bien de celle-ci car le masque de sous-réseau n'était pas correct. Une fois la configuration rectifiée, l'ensemble des utilisateurs du réseau pouvaient accéder au serveur.

C. Mise à jour des commutateurs HP du site de Château Gombert

Une fois le problème avec le serveur réglé, nous avons été chargés de chercher les mises à jour des différents commutateurs HP du site. Pour cela, nous avons commencé par vérifier que l'inventaire des équipements réseaux était correct car notre tuteur nous avait indiqué que cet inventaire était ancien et de ce fait certainement faussé. En effet, lors de la visite des différents sous-répartiteurs, j'ai pu constater que certains équipements avaient été remplacé ou même n'existaient plus. J'ai alors mis à jour l'inventaire puis j'ai recherché les mises à jour des commutateurs HP.

Avant de procéder à la mise à jour de tous les équipements réseaux, j'ai tout d'abord mis à jour deux commutateurs situés dans le SR6 (Figure 2). J'ai choisi ce sous-répartiteur pour plusieurs raisons. Tout d'abord, ce dernier ne dessert aucun autre sous-répartiteur. Ensuite, il contient deux modèles de commutateurs HP qui sont présents dans le reste de l'établissement. Enfin, au moment du choix, seulement quatre utilisateurs été présents à l'étage qu'il dessert et aucun d'entre eux nécessitait un haut débit de connexion et ils étaient tous d'accord pour utiliser le wifi le temps de l'opération de maintenance.

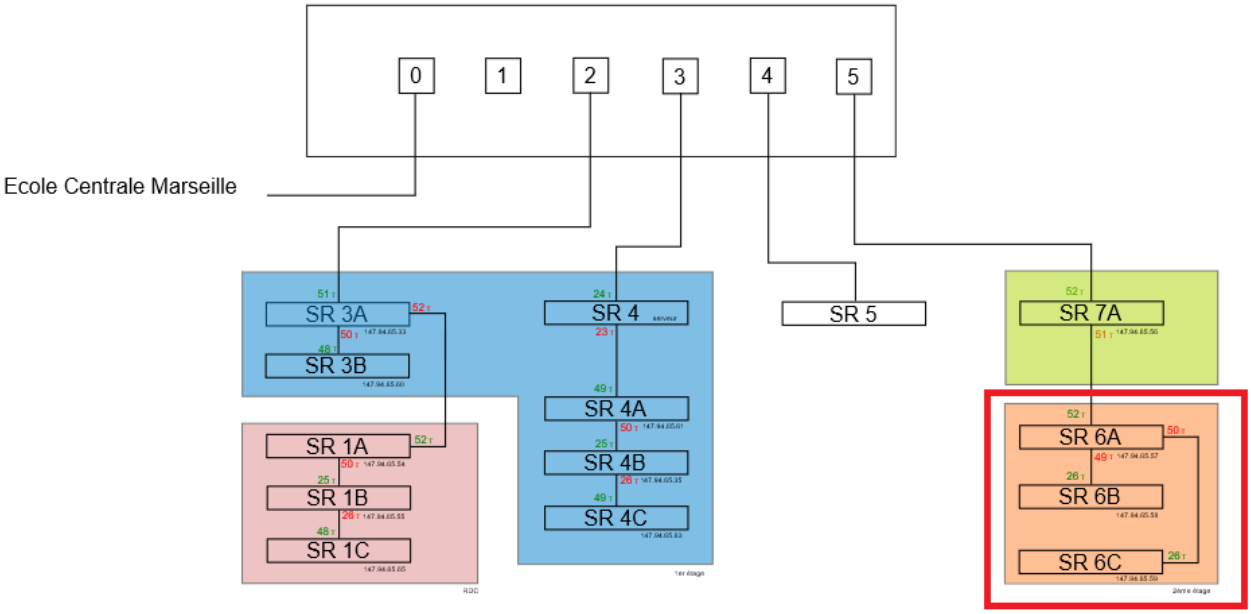

Figure 2 : Plan schématique du réseau du site de Château Gombert

Le premier commutateur fut mis à jour sans difficulté mais le second n'était plus accessible à distance ou même en connexion directe. En effet, les mots de passes de connexion à ce dernier n'étaient plus reconnus. Après des recherches dans la documentation HP du modèle de commutateur, j'ai opté pour l'utilisation du bouton « Clear » qui permet d'effacer uniquement les mots de passe de la configuration de l'équipement. J'ai donc ensuite reconfiguré les mots de passes sur les accès au commutateur.

Lors de cette opération de maintenance, j'ai découvert d'autres problèmes comme la configuration d'un mauvais port pour les accès en SSH ou des fautes dans les bannières de connexion. J'ai donc décrit la démarche suivie concernant les mots de passe ainsi que les commandes utilisées sur une page de documentation à laquelle j'ai ajouté les commandes pour activer les connexions SSH sur un port ainsi qu'une version rectifiée de la bannière de connexion aux commutateurs.

Une fois la mise à jour du SR6 effectuée avec succès, la décision de la date de mise à jour des autres sous-répartiteurs fut prise dans la journée et les utilisateurs furent prévenus de cette dernière. Afin de préparer la maintenance, j'ai décrit un plan d'action en expliquant mes choix à Guillaume CHAGNARD, qui veillait à ce que ce plan soit cohérent et simple d'exécution. Grâce à la préparation et à la documentation qui fut rédigée, la maintenance se déroula parfaitement.

D. Installation d'OpenVAS

Après avoir mis à jour les commutateurs et réglé quelques erreurs de configuration comme les ports par défaut pour les connexions SSH, nous nous sommes demandé s'il n'y avait pas d'autres erreurs du même type provoquant des failles de sécurité. Ainsi, nous avons pensé à installer un scanner de vulnérabilité : OpenVAS.

Ce logiciel a été choisi car c'est un logiciel libre, facile d'utilisation et quand il trouve une vulnérabilité, il décrit en détail la faille et une procédure pour fixer cette dernière.

L'écran d'accueil d'OpenVAS (Figure 3) présente les différents scans que l'on a réalisé au préalable ainsi que la plus haute sévérité parmi les failles trouvées.

Figure 3 : Ecran d'accueil d'OpenVAS

En cliquant tout simplement sur l'un des résultats de scan, on obtient le détail des différentes vulnérabilités (Figure 4). Ce détail est structuré de la manière suivante : la première colonne donne un nom à la faille trouvée ; on trouve ensuite son niveau de sévérité basé sur une échelle d'ordre croissant allant de 0 à 10, 10 relevant d'une faille de sécurité extrême ; la troisième colonne indique la Qualité de Détection, ici appelée « QoD », ce pourcentage représente la certitude qui est accordé à la vulnérabilité trouvée, dans l'exemple ci-dessous, il y a beaucoup de ligne qui ont une QoD de 99% ce qui indique une vulnérabilité face aux accès à distance (comme par exemple les injections SQL) ; les deux dernières colonnes informatives donne l'adresse IP de l'hôte vulnérable ainsi que le port et le protocole sur lesquels se trouve la faille.

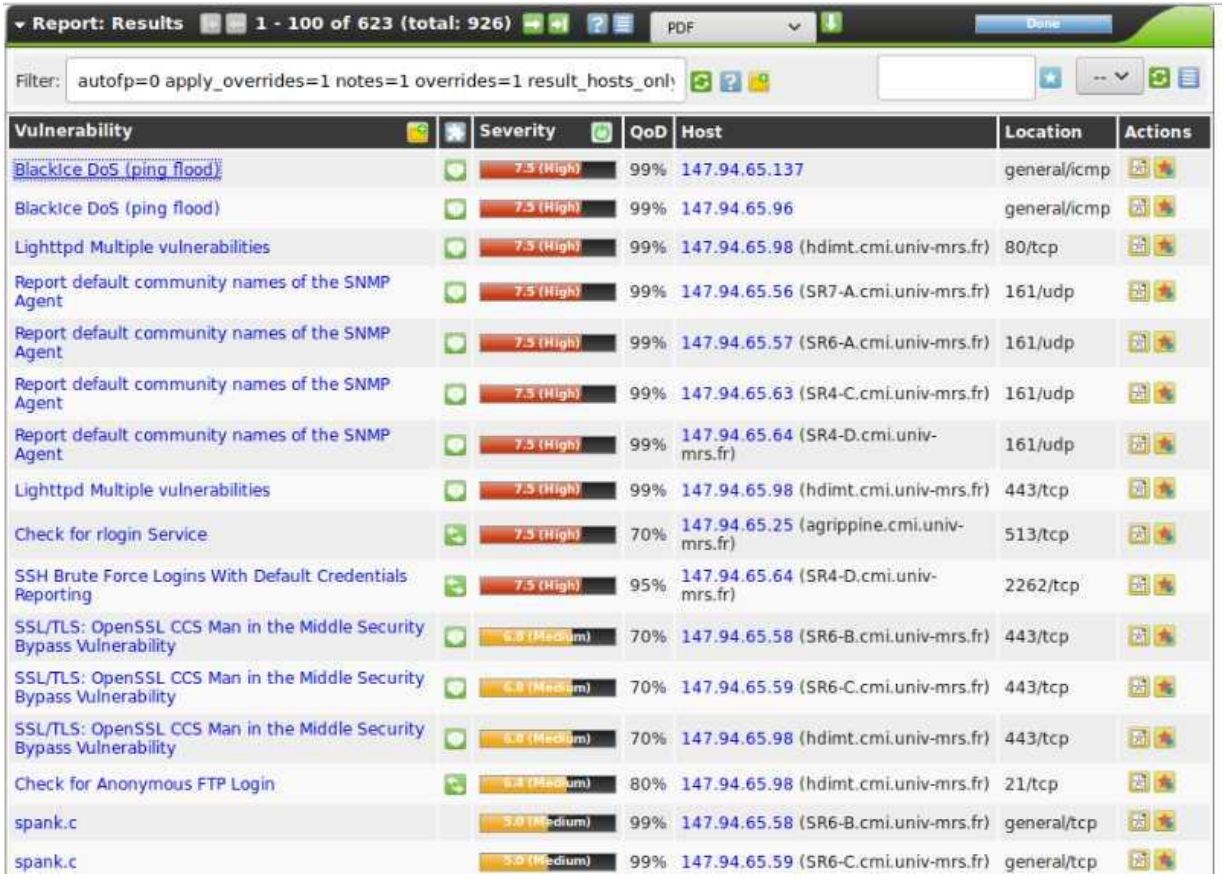

Figure 4 : Liste des vulnérabilités trouvées

Une fois sur la liste des vulnérabilités trouvées, on peut décider de voir le détail d'une de celles-ci (Figure 5). Sur cette fiche, on retrouve la même ligne que sur la liste puis il nous est donné des informations complémentaires comme : un résumé de la vulnérabilité, comment est-ce qu'elle a été détectée, l'impact qu'elle pourrait avoir si elle était exploitée ainsi qu'une solution pour la fixer.

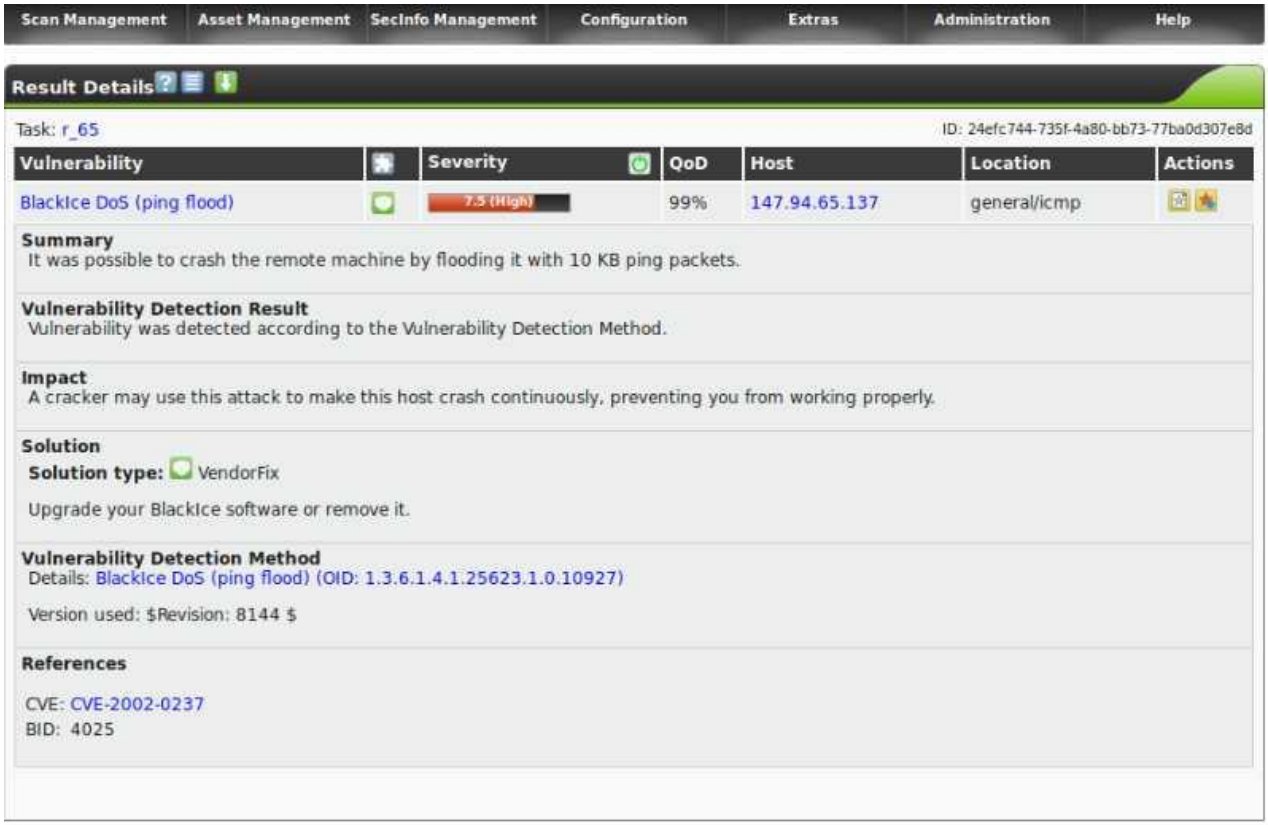

Figure 5 : Détail d'une vulnérabilité

E. Installation d'un logiciel de supervision : Shinken

Pendant que Mathéo KORADJIAN installait OpenVAS, j'ai commencé à me pencher sur la supervision du réseau.

1. Qu'est-ce que la supervision ?

En informatique, la supervision est la surveillance du bon fonctionnement d'un équipement ou encore d'un service. Par exemple, si un équipement est pourvu de sonde de température, il est possible avec un logiciel adapté de surveiller sa température et donc de pouvoir agir avant que celle-ci impacte le fonctionnement de l'équipement.

Dans le cas présent, un logiciel de supervision permettrait au service informatique de surveiller le fonctionnement de multiples équipements et services déployés sur les trois sites afin d'agir avant que les utilisateurs sur réseaux fassent remonter la panne.

2. Pourquoi Shinken ?

En discutant de la supervision du réseau avec le service informatique, j'ai appris que deux solutions avaient été déployées mais qu'aucune n'était utilisée car les services surveillés étaient pour la plupart obsolètes. Ces deux solutions étaient : Netdisco et Shinken. Après plusieurs recherches, je suis arrivé à la conclusion que Netdisco et Shinken étaient tous deux assez simple d'installation et de configuration ; je me suis donc basé sur une caractéristique beaucoup moins technique mais tout autant importante : leur interface graphique. En effet, je pense qu'il est très important pour un tel logiciel d'être très « visuel » car, pour être informer le plus rapidement sur ce qu'il se passe, il faut que ce dernier fasse remonter le problème sans qu'aucune action de la part d'un administrateur ne soit nécessaire. C'est pour cela que j'ai décidé de redéployer Shinken car il a une interface web dotée d'un tableau de bord (Figure 6) ainsi qu'une autre page détaillant les services surveiller sur un hôte (Figure 7).

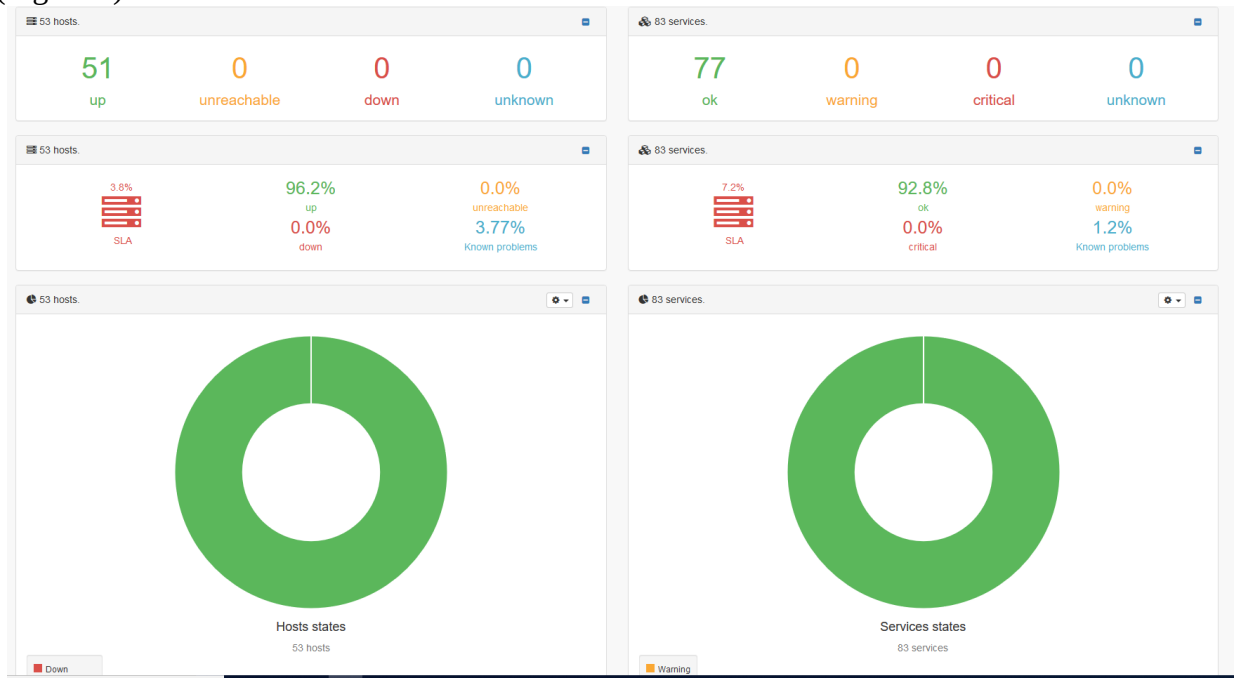

Figure 6 : Tableau de bord de Shinken

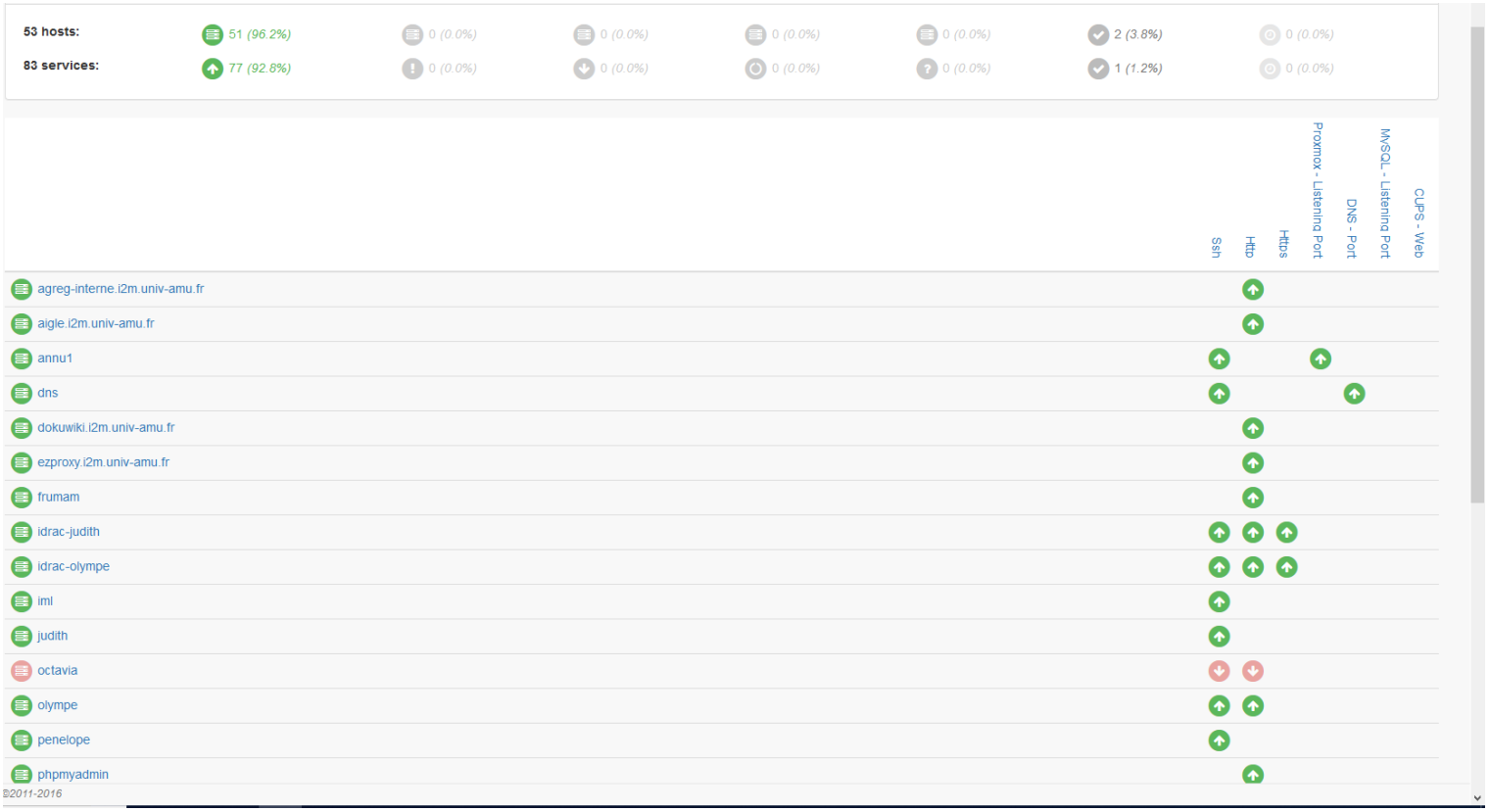

Figure 7 : Liste des hôtes et de leurs services surveillés

Comme le montre les images, Shinken permet de se rendre compte instantanément qu'un hôte ou qu'un de ses services est en panne, ou du moins inaccessible. De plus, on peut rajouter à cela le déclenchement de multiples types de notifications comme des alertes sonores, des notifications par SMS, par courriel, par bipeur, par **fenêtre intrusive*** ou bien d'autres encore. Je détaillerai plus tard comment est-ce-que j'ai mis en place le système de notifications par courriel.

Shinken a un autre avantage qui est lié à sa création. En effet, à l'origine, Shinken était un module de **Nagios*** mais lorsque l'équipe de développement du module demanda à devenir une branche du logiciel, les développeurs de ce dernier refusèrent. C'est ainsi que Shinken devenu un logiciel à part entière et totalement libre de source mais aussi entièrement compatible avec les modules de Nagios et bien plus flexible et simple d'utilisation que ce dernier. Cette compatibilité avec ces modules permet de profiter d'énormément d'extensions qui permettent par exemple la diversité des moyens de notifications que j'ai énoncée plus haut.

3. Installation de Shinken

Shinken a été installé sur une machine ayant pour système d'exploitation **Ubuntu*** 16.04 de la façon suivante.

Tout d'abord, il faut installer les paquets python-pip et python-cherrypy pour gérer et installer facilement des paquets en relation avec **Python*** :

```
$ apt-get install python-pip python-cherrypy
```

On crée ensuite un utilisateur shinken qui sera utilisé par le logiciel :

```
$ adduser shinken
```

On installe Shinken, on l'ajoute à la liste des démons à lancé au démarrage de la machine et on le démarre :

```
$ pip install shinken
$ update-rc.d shinken defaults
$ /etc/init.d/shinken start
```

On initialise ensuite Shinken :

```
$ shinken --init
```

On commence ensuite à installer des modules :

- Simple-log pour avoir des journaux plus clairs

```
$ shinken install simple-log
```

- WebUI 2 pour avoir l'interface montrée plus haut aux figures 3 et 4

```
$ pip install bottle==0.12.8 pymongo>=3.0.3 requests arrow passlib
$ pip install -r https://raw.githubusercontent.com/shinken-
monitoring/mod-webui/develop/requirements.txt
$ shinken install webui2
```

- Nagios-plugins pour activer la compatibilité avec Nagios et ses modules

```
$ apt-get install nagios-plugins
```

- Monitoring-plugins qui permet de superviser de services et d'équipement facilement à l'aide de modèles déjà fait

```
$ wget https://www.monitoring-plugins.org/download/monitoring-
plugins-2.1.2.tar.gz
$ tar xvf monitoring-plugins-2.1.2.tar.gz
$ cd monitoring-plugins-2.1.2/
$ ./configure
$ make
$ make install
```

Afin de ne pas laisser un accès libre à l'interface web de supervision, il faut activer un module d'authentification sur WebUI 2. Pour cela, on modifie le fichier '/etc/shinken/modules/webui2.cfg' et on ajoute à la ligne 90 :

```
modules auth-cfg-password
```

Il ne reste plus qu'à activer les modules simple-log et WebUI 2 ainsi qu'à télécharger quelques modèles pour faciliter la définition des services à surveiller ensuite. Pour cela, on ajoute dans le fichier '/etc/shinken/brokers/broker-master.cfg' la ligne :

```
modules     simple-log,webui2
```
Puis on installe les modèles :

```
$ shinken install http
$ shinken install ssh
$ shinken install postgresql
```

En suivant ces étapes, le serveur de supervision est maintenant en place. Il ne reste plus qu'à le configurer.

4. Configuration de Shinken

Avec Shinken, il est possible de définir tous ses hôtes et services dans un seul fichier de configuration mais cela est très déconseiller. La méthode la plus simple et claire est de définir un hôte par fichier de configuration. Il est aussi fortement conseiller de déclarer les services à surveiller sur cet hôte dans le même fichier de configuration (Figure 8). La déclaration des hôtes se fait dans le répertoire '/etc/shinken/hosts/'.

```
define host{
       use                     generic-host,ssh,http
       hostgroups              linux
       contact_groups          admins
       host_name               catalogue
       address                 147.94.64.106
       }

define service{
       use                      generic-service
       host_name                catalogue
       check_command            check_http! -p 8080
       check_interval           1
       service_description      Intranet
}
```
Figure 8 : Exemple de fichier de définition d'un hôte et de ses services

L'exemple de fichier de configuration ci-dessus, permet d'apprendre beaucoup sur le fonctionnement de Shinken, c'est pourquoi je vais détailler la façon dont est construit ce fichier.

Ici, le fichier est séparé en deux parties. Une partie définie l'hôte et l'autre le service à surveiller. Dans la première partie, on observe tout d'abord une ligne 'use' qui indique que l'hôte utilise plusieurs modèles. Ici, ces modèles sont generic-host, ssh et http. Le simple appel de ces modèles sur cette ligne provoque, intrinsèquement, la déclaration de variables propres aux modèles utilisés qui permettent ensuite de définir les services de l'hôte plus facilement. Par exemple, si cette ligne commençant par 'use' ne contenait pas 'html', il aurait alors fallu rajouter les trois lignes ci-dessous, à la fin de la déclaration de l'hôte, pour permettre à la définition du service de fonctionner correctement.

```
_CHECK_HTTP_AUTH        admin:m+28-qSJ
_CHECK_HTTP_DOMAIN_NAME $HOSTADDRESS$
_CHECK_HTTP_URI         /
```

La seconde ligne est ensuite la déclaration d'appartenance à un groupe. Elle a avant tout un intérêt organisationnel. La troisième ligne définit le groupe de contact à notifier en cas de panne ou de résolution de panne. La ligne suivante donne le nom qui est associé à cet hôte sur l'interface web. La dernière ligne renseigne simplement l'adresse IP de l'hôte.

Dans la seconde partie, on observe certaines lignes déjà présente dans la partie précédente. La première ligne appelle toujours un modèle. La seconde ligne indique cette fois-ci sur quel hôte ce service doit être surveiller. La troisième ligne indique la commande à utiliser pour vérifier le fonctionnement du service. Ici, la commande check_http provient du paquet http qui a été installé précédemment. Par défaut, elle s'exécute sur le port 80, port habituel des serveurs web, mais ici, l'utilisation de '-p 8080' indique qu'elle doit s'exécuter sur le port 8080 à la place. La ligne suivante indique simplement un intervalle de temps en minutes à laquelle réexécuter la commande pour surveiller que le service ne tombe pas en panne. La dernière ligne quant à elle définit le nom qui sera afficher sur l'interface web au-dessus de la vérification du service.

Comme indiqué plus haut, on peut définir des groupes dans Shinken. Ces groupes ont principalement un intérêt organisationnel, ils sont donc utilisés pour regrouper les hôtes possédant des caractéristiques similaires. Par exemple, le groupe « Switchs » (Figure 9) est un groupe regroupant des commutateurs. La déclaration des groupes d'hôtes se fait dans le répertoire '/etc/shinken/hostgroups'.

```
define hostgroup{
        hostgroup_name      switchs
        alias               Switchs
}
```

Figure 9 : Exemple de fichier de configuration d'un groupe d'hôtes

En plus des groupes et des hôtes, il est nécessaire de définir au minimum un contact. C'est dans la définition de ce contact (Figure 10). La déclaration des contacts se fait dans le répertoire '/etc/shinken/contacts'.

```
define contact{
        use                 generic-contact
        contact_name        i2m-support
        password            [redacted]
        email               i2m-informatique@univ-amu.fr
        notificationways    email
        is_admin            1
        expert              1
}
```

Figure 10 : Exemple de fichier de configuration d'un contact

Dans cet exemple, un contact 'i2m-support' est déclaré. Il a besoin d'un mot de passe car on a activé l'authentification sur l'interface web ; d'une adresse électronique car, comme l'indique 'notificationways', il sera notifié par courriel ; enfin, il est déclaré comme administrateur et comme expert.

5. Différents problèmes lors du déploiement de Shinken

La majorité du déploiement de Shinken s'est déroulée sans le moindre problème mais certains points m'ont bloqué à certains moments.

a) Installation d'un module avec un mauvais utilisateur

Après l'installation d'un module Nagios nommé « check_nwc_health » qui permet de surveiller des équipements réseau, Shinken n'arrivait pas à effectuer les vérifications nécessaires. Le message d'erreur indiquait que le module avait été installé pour un autre utilisateur.

En recherchant la cause du problème dans le script d'installation du module, je me suis rendu compte que le plugin ayant été créé pour Nagios, il s'installer avec comme utilisateur nagios dans le répertoire '/var/lib/nagios'. Le comportement que je souhaitais étant qu'il s'installe pour l'utilisateur shinken dans le répertoire '/var/lib/shinken'.

J'ai donc désinstallé puis réinstallé le module et lors de sa configuration j'ai indiqué les paramètres que j'ai indiqué plus précédemment :

```
$ ./configure --prefix=/var/lib/shinken --with-nagios-user=shinken
 --with-nagios-group=shinken
```

Une fois cette configuration entrée, les équipements réseaux étaient enfin correctement surveillés.

b) Notification par mail impossible

Comme indiqué, Shinken permet de notifier des problèmes détectés par de multiples moyens. Dans le cadre de cette installation, nous avons décidé de simplement activer la notification par courriel comme montré avec la figure 10.

Tout d'abord, j'ai essayé l'envoie d'un mail manuellement pour vérifier que la base fonctionnait bien. Cette étape fut nécessaire car l'envoie manuel était impossible premièrement. J'ai dû changer de serveur **SMTP*** et passer de celui de Gmail à celui de l'université d'Aix-Marseille pour régler ce premier problème.

Ensuite, j'ai configuré l'envoie automatique mais aucun message n'était envoyé. En regardant dans les journaux du système, j'ai découvert que Shinken essayait d'envoyer le message à partir de l'utilisateur 'root' sur lequel il n'a pas de droit. J'ai donc dû modifier le script d'envoi de courriel pour indiquer quel utilisateur devait envoyer le mail. Ce dernier faisait appel à une fonction 'get_user()' qui retournait, par le jeu de plusieurs variables, l'utilisateur 'root'. J'ai donc commenté cette fonction puis j'en ai créé une autre du même nom qui cette fois-ci retourne comme utilisateur 'shinken'.

c) Définition de nouvelles vérifications de services

Sur un hôte en particulier, il était nécessaire de surveiller deux pages web (donc deux ports différents) or ce n'est pas possible avec la configuration de base du modèle http. Mais comme indiqué plus haut, Shinken est extrêmement flexible et je me suis servi de cette flexibilité pour surveiller ces deux services en ajoutant des commandes de vérifications aux commandes du modèle http sans passer par ce modèle.

Pour que ce soit plus clair, il faut regarder la définition de cet hôte un peu particulier (Figure 11).

```
define host{
        use                     generic-host,ssh
        hostgroups              linux
        contact_groups          admins
        host_name               v-srvl-10
        address                 147.94.64.169
        _CHECK_HTTP_AUTH        [redacted]
        _CHECK_HTTP_DOMAIN_NAME $HOSTADDRESS$
        _CHECK_HTTP_URI         /
}

define service {
                use                             generic-service
                host_name                       v-srvl-10
                check_command                   check_http_5000
                check_interval                  1
                service_description             NetDisco - Web
}

define service {
                use                             generic-service
                host_name                       v-srvl-10
                check_command                   check_http_dhcp
                check_interval                  1
                service_description             interface DHCP
}
```

Figure 11 : Fichier de configuration de l'hôte v-srvl-10

Comme on peut le voir dans la partie de déclaration de l'hôte, sur la ligne 'use', http n'est pas mentionné alors que les deux services à surveiller sont des pages web. Si j'ai choisi de ne pas utiliser le modèle http proposé, c'est parce que ce modèle fixe le port sur 80 par défaut or ici, il y a deux ports différents à surveiller. En revanche, j'ai déclaré des variables locales de la même forme que celles du modèle http. Ainsi, il ne me reste plus qu'a créé deux commandes spécifiques aux deux services que je souhaite surveiller, ici ces deux commandes sont 'check_http_5000' et 'check_http_dhcp'.

Pour créer ces deux commandes, il suffit de les ajouter dans le fichier de configuration des commandes du modèle http qui se trouve dans le répertoire '/etc/shinken/packs/http/' (Figure 12). C'est commandes sont basiquement les mêmes que celles utilisées par le modèle http mais il y a une notion d'authentification et de ports différents en plus.

```
define command {
   command_name   check_http_5000
   command_line   $NAGIOSPLUGINSDIR$/check_http -H
 $_HOSTCHECK_HTTP_DOMAIN_NAME$ -u $_HOSTCHECK_HTTP_URI$ -p 5000 --
authorization=$_HOSTCHECK_HTTP_AUTH$
}
define command {
   command_name   check_http_dhcp
   command_line   $NAGIOSPLUGINSDIR$/check_http -H
$_HOSTCHECK_HTTP_DOMAIN_NAME$ -u $_HOSTCHECK_HTTP_URI$ -p 80 --
authorization=$_HOSTCHECK_HTTP_AUTH$
}
```

Figure 12 : Extrait du fichier de configuration des commandes du modèle http

F. Installation d'un client LDAP avec Autofs

1. Qu'est qu'un annuaire et comment est-il utilisé à l'I2M ?

Un annuaire informatique est similaire à un annuaire dans la vie de tous les jours. Il permet de structurer des informations comme on le souhaite. Si on prend l'exemple d'un annuaire téléphonique, il permet à partir d'un numéro de téléphone d'avoir un nom et une adresse. En informatique, un annuaire suit le même fonctionnement. Dans le cas de l'I2M, comme dans le cas de beaucoup d'entreprises, cet annuaire est utilisé pour authentifier les utilisateurs sur des applications et des machines mais aussi pour monter dynamiquement le répertoire des utilisateurs sur les machines. C'est-à-dire que si un utilisateur est inscrit dans la base LDAP, il pourra se connecter sans avoir à déplacer ses données sur n'importe quelle machine connectée au serveur LDAP et configurée pour cette utilisation.

Ma mission était donc de configurer des machines, ici des serveurs de calculs, en tant que clients du serveur LDAP pour que tout utilisateur de l'annuaire puisse s'y connecter.

2. Installation d'un client

Avant de configurer les serveurs de calculs, j'ai commencé par tester de configurer une machine locale. L'installation complète du client sera détaillée en annexe, je vais seulement parler des principaux fichiers de configurations ainsi que des différents paquets à installer.

Il est important de connaître le fonctionnement d'Autofs pour comprendre comment les répertoires des utilisateurs vont être monté sur la machine. Autofs utilise un système de cartes de montage, ces cartes sont des entrées dans la base LDAP qui définissent toutes les informations nécessaires à identifier l'utilisateur mais aussi son répertoire. Par défaut sur les distribution Linux, tous les utilisateurs ont leur répertoire dans le répertoire '/home'. Or, les machines sur lesquelles sera installé le client LDAP sont des systèmes Linux. Ainsi donc, si on souhaite garder des utilisateurs locaux sur la machine, comme par exemple un compte administrateur, il faut déplacer ces derniers dans un autre répertoire pour qu'ils ne soient pas écrasés par le montage dynamique effectué par Autofs.

La première étape consiste donc à déplacer les utilisateurs que l'on souhaite garder localement. Il faut ensuite installer tous les paquets nécessaires au bon fonctionne du client LDAP avec Autofs :

```
$ apt-get install ldap-utils autofs-ldap ldap-auth-client nscd libnss-ldapd libpam-ldapd libpam-mount
```

Les principaux fichiers de configuration sont :

- '/etc/ldap/ldap.conf' dans lequel on entre toutes les informations concernant le serveur LDAP auquel on souhaite se lier, comme la base de l'annuaire, son URI (Uniform Resource Identifier), sa version, etc. mais aussi le certificat de connexion si nécessaire.

- '/etc/default/autofs' dans lequel on entre principalement les informations concernant le type de cartes utilisé.

- '/etc/auto.master' dans lequel on renseigne le répertoire sur lequel on va monter les répertoires des utilisateurs, ici ce serra '/home/', et l'emplacement des cartes de montage dans la base LDAP.

- '/etc/nsswitch.conf', qui est un fichier de configuration primordial et qu'il faut donc manipuler avec beaucoup de précaution.

Cette mission a été la plus longue de toute celle que j'ai eu car il s'est passé beaucoup de temps avant de je trouve une documentation à jour qui a ensuite réglé la totalité de mes problèmes de configuration et elle fut entrecoupée de multiples tâches plus petites comme l'installation de postes clients, le changement de pièces d'ordinateurs ou encore la recherche d'un souci avec le service de création automatique de tickets de **GLPI***.

Une fois la configuration fonctionnelle et testée à plusieurs reprises sur une machine test, j'ai pu réinstaller entièrement le premier serveur de calcul qui était sous **CentOS*** et qui dépendait encore d'un ancien NIS. Il a été réinstallé sous Ubuntu 18.04 et est maintenant un client du serveur LDAP avec Autofs. Etant un serveur de calcul, j'ai aussi installé Matlab et tous ses modules dessus. Pour être sûr que l'installation est complétement fonctionnelle, le service informatique a décidé d'attendre au moins un mois avant de faire la réinstallation des deux autres serveurs de calcul.

G. Raccordement du site de Luminy à la boucle optique

Sur le campus universitaire de Luminy, des travaux ont eu lieu afin de former une boucle de fibre optique desservant la plupart des bâtiments du campus. Une fois cette boucle terminée, le service informatique de Luminy fit la demande auprès de la DOSI de se raccorder sur cette nouvelle liaison.

La DOSI accepta et proposa des réunions pour discuter de l'opération à effectuer. J'ai donc été appelé sur Luminy pour participer à ces réunions car je n'avais jamais manipulé de liaison en fibre optique et car c'était une bonne occasion pour discuter de l'aspect technique de la boucle mais aussi du nouveau cœur de réseau du campus.

En effet, le cœur du réseau du campus a été déplacé car le bâtiment dans lequel il était situé, le TPR1, est en rénovation. Anciennement au premier étage du TPR1, il se trouve actuellement au sous-sol de ce même bâtiment. L'accès à cette salle est bien entendu très limité et ce, par deux facteurs. Premièrement, étant situé dans un bâtiment en travaux, l'accès à la zone est restreint. Il faut s'équiper d'une tenue de chantier ainsi que d'un badge magnétique pour avoir accès à cette zone.
Deuxièmement, cette salle étant très sensible d'un point de vue à la fois physique, car il faut limiter au maximum l'entrée de poussière pour ne pas déclencher l'alarme à incendie (Figure 13), et sécuritaire. Cette salle est donc complétement étanche et la porte est équipé d'un lecteur d'empreintes digitales.

Figure 13 : Système d'extinction automatique d'incendie par gaz inerte

Sur cette photographie, on observe le boitier d'alarme, un panneau d'alerte 'évacuation immédiate' et sept bombonnes de gaz inerte. Des panneaux d'alerte similaires sont répartis dans les autres couloir de la salle afin que toute personne présente soit informé du danger. En cas d'incendie, le panneau s'allume, une alarme retentit et après trente secondes, les sept bombonnes de gaz se vide jusqu'à ce qu'il n'y ait plus assez d'oxygène dans l'air de la pièce pour que le feu continue à brûler. Le départ de feu est détecté par des aspirations d'air situé sur les tuyaux rouges visibles au-dessus des bombonnes de gaz inerte.

Le reste de la salle est composé de quatre couloirs qui donnent accès à des équipement réseaux et des serveurs (Figure 14).

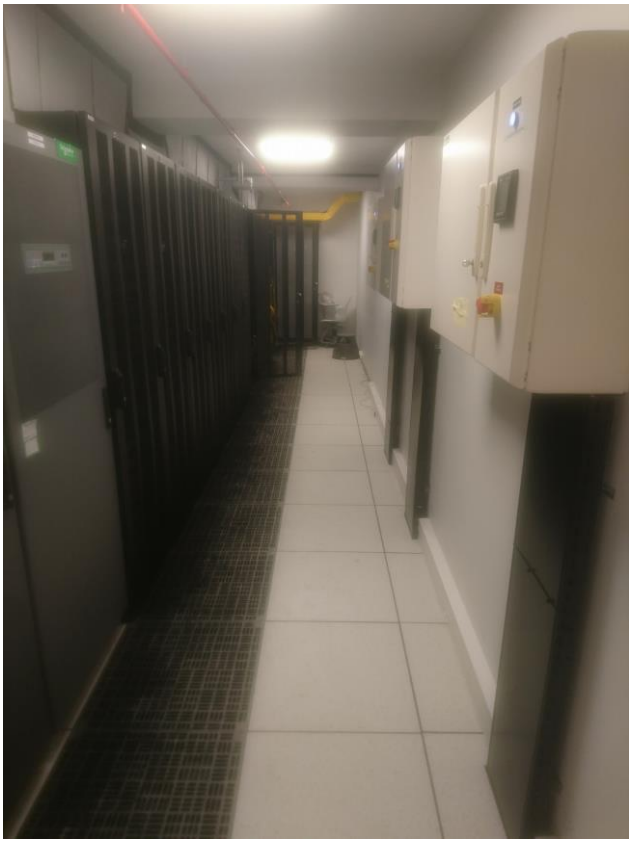

Figure 14 : Un couloir du data center

Lors des réunions de discussions, nous avons eu plusieurs choix de raccordement. Soit on déplaçait simplement notre liaison actuelle sur la nouvelle boucle optique, soit on doublait cette nouvelle liaison en configurant de l'agrégation de liens, solution qui permet de gagner en débit et redondance. Nous avons choisi de simplement déplaçait notre ancienne liaison fibre pour des raisons de matériel. En effet, l'équipement réseau qui sert de routeur fédérateur à l'I2M de Luminy possède uniquement un port fibre optique, il était donc impossible de choisir la seconde solution. Une fois les discussions terminées, nous avons fixé une date pour effectuer ce transfert. Le jour de l'opération, Mathéo KORADJIAN et moi-même nous sommes repartis entre l'I2M de Luminy et le data center afin de procéder au déplacement de la liaison (Figure 15).

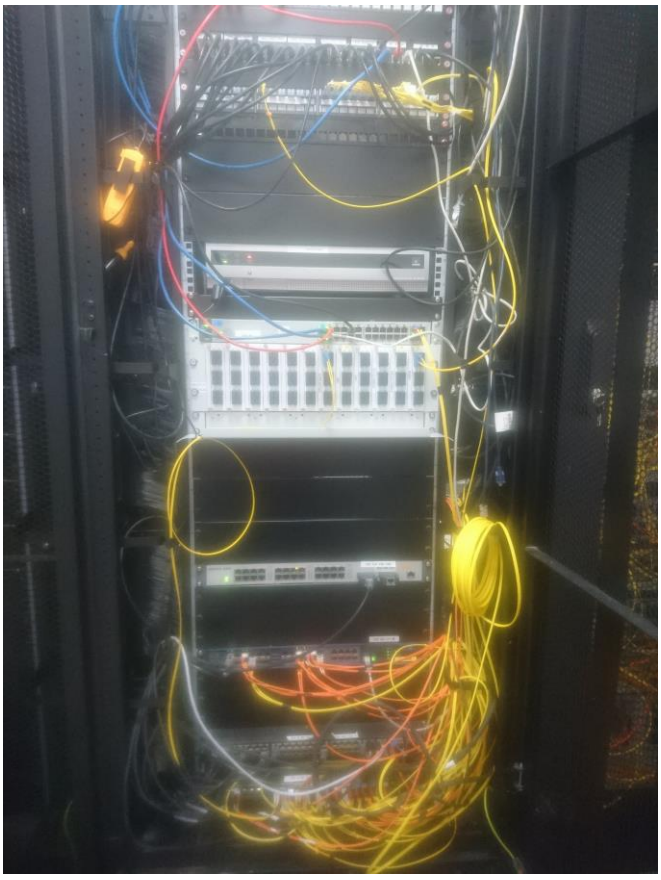

Figure 15 : Image du raccordement

H. Migration du serveur DHCP et d'inventaire de Luminy

Après le raccordement du site à la nouvelle boucle optique, j'ai reçu une autre mission sur ce site. Il m'a été demandé d'installer une nouvelle machine qui servira de serveur DCHP et d'inventaire au site. J'ai donc installé et configuré ces deux serveurs sur une machine sous Ubuntu 16.04.

1. Installation et configuration du serveur DHCP

Pour le serveur DHCP, j'ai récupéré l'ancien fichier de configuration et j'ai purgé les entrées qui avaient plus de huit ans et j'ai commenté celles dont l'adresse MAC n'était donné dans le serveur d'inventaire.

L'installation de ce service est très simple, il suffit d'installer le paquet (ici 'isc-dhcp-server'), de le configurer grâce au fichier de configuration '/etc/dhpc/dhcpd.conf' puis de lancer le service avec la commande :

```
$ service isc-dhcp-server restart
```

Il ne faut pas oublier de désactiver l'autre serveur DHCP pour être sûr que dorénavant, tous les appareils qui utilise le DHCP discutent avec ce nouveau serveur.

2. Installation et remplissage du serveur d'inventaire

Le serveur d'inventaire utilisé sur le site de Luminy est en réalité une base de données **MySQL***. De ce fait, il y a une table servant d'inventaire dans la base de données qui répertorie tous les équipements du site de Luminy. Cette base est accessible par **phpMyAdmin***, ce qui permet de la gérer facilement et à distance.

L'installation de ce serveur d'inventaire est elle aussi très facile. Il suffit d'installer un serveur web sur la machine, ici apache2, et de lui ajouter un interpréteur **PHP***. Ensuite d'installer MySQL puis phpMyAdmin.

Pour le remplissage de la base, il suffit d'importer l'ancienne table sur le nouveau serveur. Cet import est extrêmement simple à réaliser lorsque l'on est connecté aux deux bases simultanément.

Afin de sécuriser un minimum l'accès à cette base de données, j'ai rédigé un guide des étapes à suivre pour forcer l'accès à l'interface web en **HTTPS***. Je n'ai pas pu appliqué ces étapes moi-même car il faut qu'un **URL*** soit assigné à l'IP de la machine dans le **DNS*** de la DOSI.

III. Conclusion

Durant mon stage, j'ai principalement travaillé sur l'administration des systèmes mais j'ai aussi eu des missions relevant de l'administration des réseaux.

La prédominance dans mon stage de tâches orientées vers l'administration des systèmes informatiques m'a permis, non seulement de consolider mes connaissances sur le sujet mais aussi d'acquérir de nouveaux savoirs dans ce domaine.

Le contact avec les utilisateurs m'a aussi permis d'apprendre à être à l'écoute et à la disposition des personnes, si elles ont des questions ou des craintes, mais aussi à donner des directives claires et simple, dans le cadre d'une prise en charge à distance par exemple.

Mes différentes missions m'ont permis d'élargir plus encore mon spectre de compétences et de connaissances dans le domaine de l'Administration informatique en général. Grâce à ce stage, j'ai non seulement découvert le monde de l'entreprise dans un service informatique mais aussi toutes les responsabilités qu'impliquent le fait de travailler dans un tel service.

Cette expérience de dix semaines a confirmé ma volonté de travailler dans l'Administration des Systèmes et des Réseaux. Qui plus est, mon cursus en IUT et cette dernière me permettent d'avoir une base solide dans ce domaine, ce qui par conséquent, me permet de devenir un alternant en Licence Professionnelle à la fois motivé et qualifié.

IV. Glossaire

DUT, Diplôme Universitaire de Technologie.
I2M, Institut de Mathématiques de Marseille.
CNRS, Centre National de Recherche Scientifique, est un organisme public de recherche.
DOSI, Direction Opérationnelle des Systèmes d'information, a pour mission de mettre en œuvre la politique de l'université d'Aix-Marseille en matière de systèmes informatiques.
OpenVAS, un scanner de vulnérabilités qui permet de scanner des réseaux et ordonne les failles en fonction de leur gravité tout en proposant une solution.
Supervision, surveillance de manière continue de la disponibilité des services en ligne.
Shinken, une application permettant la surveillance système et réseau.
LDAP, Lightweight Directory Access Protocol, un protocole d'interrogation et de modification d'annuaire.
Autofs, module de LDAP permettant via des **démons*** de monter automatiquement des répertoires quand ils sont utilisés et de les démontés après une période d'inactivité.
Démon (informatique), processus qui s'exécute en arrière-plan.
NIS, Network Information System, protocole client-serveur développé par Sun permettant la centralisation d'informations sur un réseau Unix.
DHCP, Dynamique Host Configuration Protocol, protocole réseau qui assure la configuration automatique des paramètres réseaux d'un hôte.
SSH, Secure Shell, à la fois un programme informatique et un protocole de communication sécurisé.
Fenêtre intrusive, aussi appelée fenêtre pop-up, s'affiche, sans avoir été sollicitée par l'utilisateur, devant la fenêtre de navigation principale.
Ubuntu, distribution de Linux.
Python, langage de scripting.
SMTP, Simple Mail Transfer Protocol, protocole permettant d'envoyer des courriels sur le réseau.
GLPI, Gestion Libre de Parc Informatique, logiciel de gestion de par cet de services d'assistances.
CentOS, distribution de Linux principalement destinée aux serveurs.
MySQL, l'un des systèmes de gestion de bases de données les plus utilisés au monde.
phpMyAdmin, application web de gestion de base de données MySQL.
PHP, PHP : Hypertext Preprocessor, est un langage de programmation libre principalement utilisé pour créer des pages dynamiques.
HTTPS, HyperText Transfert Protocol Secure, version sécurisée du protocole HTTP qui permet de naviguer sur internet.
URL, Uniform Resource Locator, chaîne de caractère qui permet d'identifier les pages et sites web.
DNS, Domain Name System, service informatique qui traduit les noms de domaines Internet en adresses IP.

V. Bibliographie

- **Documentation des commutateurs HP**

ftp://ftp.hp.com/pub/networking/software/59908821-0904.pdf

- **Documentation sur l'installation et la configuration de Shinken**

https://shinken.readthedocs.io/en/latest/
http://support.i2m.univ-amu.fr/doku.php?id=reserve:reseau:stage2016

- **Documentation sur l'installation et la configuration d'un client LDAP avec Autofs**

https://doc.ubuntu-fr.org/ldap_client
https://help.ubuntu.com/community/AutofsLDAP
https://doc.ubuntu-fr.org/autofs

- **Documentation pour la migration du serveur sur Luminy**

https://doc.ubuntu-fr.org/isc-dhcp-server
https://doc.ubuntu-fr.org/phpmyadmin
https://www.guillaume-leduc.fr/4-securiser-son-serveur-phpmyadmin.html

Institut Universitaire de Technologie,
Aix-Marseille Université

ANNEXES
Diplôme Universitaire de Technologie
Spécialité Réseaux et Télécommunications

Administration systèmes et réseaux

Maxime JULIEN

Institut de Mathématiques de Marseille

Responsable entreprise : Olivier CHABROL

Responsable académique : Tin NGUYEN

2018

Journal de bord :

Journal hebdomadaire présentant les missions réalisées au sein de l'I2M.

Semaine 1 :
- Découverte des locaux
- Mise à jour des commutateurs HP
- Installation d'OpenVAS
- Installation de Shinken et début de sa configuration

Semaine 2 :
- Configuration de Shinken

Semaine 3 à 6 :
- Installation de postes utilisateurs sous Mac
- Mise à jour de licences Matlab
- Installation du client LDAP sur une machine locale
- Réinstallation du premier serveur de calcul

Semaine 7 et 8 :
- Rédaction du plan des prises réseaux des trois étages
- Récupération de pièces sur d'anciens postes utilisateur
- Réparation de postes utilisateur
- Discussion au sujet du raccordement à la boucle optique sur Luminy

Semaine 9 et 10 :
- Rédaction du rapport de stage
- Préparation à l'oral
- Raccordement à la boucle optique
- Migration du serveur DHCP et d'inventaire sur Luminy.

Guide d'installation du client LDAP avec Autofs (type Copy and Paste)

Avant de suivre l'installation suivante, s'il y a des comptes locaux sur la machine cliente, il faut les déplacer si on souhaite les utiliser plus tard.

Pour cela, on utilise la commande : `usermod -d NEWHOME -m USER`

Il ne faut pas créé le nouveau home, il est créé automatiquement.

- Installation des paquets suivant : `ldap-utils autofs-ldap ldap-auth-client nscd libnss-ldapd libpam-ldapd libpam-mount`
- Modification du fichier `/etc/ldap/ldap.conf`

```
#
# LDAP Defaults
#

# See ldap.conf(5) for details
# This file should be world readable but not world writable.

BASE    dc=i2m,dc=univ-amu,dc=fr
URI     ldap://ldap.i2m.univ-amu.fr

ldap_version 3

scope sub
bind_policy soft
pam_filter objectclass=posixAccount
pam_login_attribute uid
pam_password md5

nss_base_passwd ou=accounts,dc=i2m,dc=univ-amu,dc=fr
nss_base_shadow ou=accounts,dc=i2m,dc=univ-amu,dc=fr
nss_base_group  ou=Groups,dc=i2m,dc=univ-amu,dc=fr

ssl start_tls
tls_reqcert allow

#SIZELIMIT      12
#TIMELIMIT      15
#DEREF          never

# TLS certificates (needed for GnuTLS)
TLS_CACERT      /etc/ssl/openldap/ca-certs
tls_cacertdir /etc/ssl/certs

nss_initgroups_ignoreusers avahi,backup,bin,bind,colord,daemon,fetchmail,games,gnats,irc,klog,lib
uuid,list,lp,mail,man,messagebus,news,nslcd,proxy,root,smmsp,smmta,sshd,statd,sync,sys,syslog,usb
mux,uucp,www-data,x2gouser
```

- Modification du fichier `/etc/default/autofs`

```
#
# Init syatem options
#
# If the kernel supports using the autofs miscellanous device
# and you wish to use it you must set this configuration option
# to "yes" otherwise it will not be used.
#

master_map_name="/etc/auto.master"

timeout=300

browse_mode="no"

logging="verbose"

LDAP_URI="ldap://ldap.i2m.univ-amu.fr"

MAP_OBJECT_CLASS="automountMap"
ENTRY_OBJECT_CLASS="automount"
MAP_ATTRIBUTE="ou"
ENTRY_ATTRIBUTE="cn"
VALUE_ATTRIBUTE="automountInformation"

USE_MISC_DEVICE="yes"
#
# Use OPTIONS to add automount(8) command line options that
# will be used when the daemon is started.
#
#OPTIONS=""
#
```

- Modification du fichier `/etc/auto.master`

```
#
# Sample auto.master file
# This is a 'master' automounter map and it has the following format:
# mount-point [map-type[,format]:]map [options]
# For details of the format look at auto.master(5).
#
#/misc  /etc/auto.misc
#
# NOTE: mounts done from a hosts map will be mounted with the
#       "nosuid" and "nodev" options unless the "suid" and "dev"
#       options are explicitly given.
#
/net    -hosts
#
# Include /etc/auto.master.d/*.autofs
# The included files must conform to the format of this file.
#
#+dir:/etc/auto.master.d
#
# Include central master map if it can be found using
# nsswitch sources.
#
# Note that if there are entries for /net or /misc (as
# above) in the included master map any keys that are the
# same will not be seen as the first read key seen takes
# precedence.
#
#+auto.master
/home   ldap://ldap.i2m.univ-amu.fr/ou=auto.home,dc=i2m,dc=univ-amu,dc=fr
```

- Modification du fichier `/etc/autofs_ldap_auth.conf`

```
<?xml version="1.0" ?>
<!--
This files contains a single entry with multiple attributes tied to it.
See autofs_ldap_auth.conf(5) for more information.
-->

<autofs_ldap_sasl_conf
        usetls="yes"
        tlsrequired="no"
        authrequired="no"
/>
```

- Modification du fichier `/etc/nsswitch.conf`

```
 /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the `glibc-doc-reference' and `info' packages installed, try:
# `info libc "Name Service Switch"' for information about this file.

# pre_auth-client-config # passwd:         compat ldap
passwd: files ldap
# pre_auth-client-config # group:          compat ldap
group: files ldap
# pre_auth-client-config # shadow:         compat ldap
shadow: files ldap
gshadow:        files

hosts:          files mdns4_minimal [NOTFOUND=return] dns
networks:       files

protocols:      db files
services:       db files
ethers:         db files
rpc:            db files

# pre_auth-client-config # netgroup:       nis
netgroup: nis

automount:      files ldap
```

- On exécute la commande `auth-client-config -t nss -p lac_ldap`
- Modification du fichier `/etc/nslcd.conf`

```
# /etc/nslcd.conf
# nslcd configuration file. See nslcd.conf(5)
# for details.

# The user and group nslcd should run as.
uid nslcd
gid nslcd

# The location at which the LDAP server(s) should be reachable.
uri ldap://ldap.i2m.univ-amu.fr

# The search base that will be used for all queries.
base dc=i2m,dc=univ-amu,dc=fr

# The LDAP protocol version to use.
ldap_version 3

# The DN to bind with for normal lookups.
#binddn cn=annonymous,dc=example,dc=net
#bindpw secret

# The DN used for password modifications by root.
#rootpwmoddn cn=admin,dc=example,dc=com

# SSL options
ssl start_tls
tls_reqcert allow
tls_cacertfile /etc/ssl/certs/ca-certificates.crt

# The search scope.
#scope sub
```

- On redémarre les services nslcd et nscd : `service nslcd restart` & `service nscd restart`
- On lance la commande `pam-auth-update`
- On modifie le fichier `/etc/pam.d/common-session`

```
#
# /etc/pam.d/common-session - session-related modules common to all services
#
# This file is included from other service-specific PAM config files,
# and should contain a list of modules that define tasks to be performed
# at the start and end of sessions of *any* kind (both interactive and
# non-interactive).
#
# As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
# To take advantage of this, it is recommended that you configure any
# local modules either before or after the default block, and use
# pam-auth-update to manage selection of other modules.  See
# pam-auth-update(8) for details.

# here are the per-package modules (the "Primary" block)
session [default=1]                     pam_permit.so
# here's the fallback if no module succeeds
session requisite                       pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
session required                        pam_permit.so
# The pam_umask module will set the umask according to the system default in
# /etc/login.defs and user settings, solving the problem of different
# umask settings with different shells, display managers, remote sessions etc.
# See "man pam_umask".
session optional                        pam_umask.so
# and here are more per-package modules (the "Additional" block)
session required        pam_unix.so
session optional        pam_mount.so
session [success=ok default=ignore]     pam_ldap.so minimum_uid=1000
session optional        pam_systemd.so
# end of pam-auth-update config
```

- On récupère les certificats de protis et on les place dans le répertoire `/etc/ssl/certs/`
- On crée le répertoire : `mkdir -p /etc/ssl/openldap`
- Copie du certificat de protis `/etc/ssl/openldap/ca-certs` sur la machine cliente (au même emplacement)
- On redémarre autofs : `service autofs restart`

Guide des étapes à suivre pour forcer l'accès à phpMyAdmin en https

L'interface web de phpmyadmin est accessible en https MAIS aussi en http car il faut que le serveur soit déclaré dans le DNS pour pouvoir bloquer les accès.

Ainsi donc, une fois l'obtention d'un url dans le DNS, les étapes qu'il reste à faire sont :

1. `a2dissite phpmyadmin.conf` pour désactiver le site
2. `service apache2 reload`
3. Modification de `/etc/phpmyadmin/apache.conf` : il faut commenter (en rajoutant un '#' au début de la ligne)
   ```
   Alias /phpmyadmin /usr/share/phpmyadmin
   ```
4. `service apache2 reload`
5. Modification de `/etc/apache2/sites-available/phpmyadmin.conf` : il faut remplacer les DEUX `ex.univ-mrs.fr` par l'url obtenu dans le DNS
6. `a2ensite phpmyadmin.conf` pour réactiver le site
7. `service apache2 reload`

Une fois ces étapes réalisées, l'accès devrait être possible uniquement en passant par l'url obtenu ET en HTTPS.