



Rapport d'apprentissage Master 1 - Informatique *Supervision open-source d'un système d'information*

Thomas Calatayud

2016-2017

Table des matières

Remerciements	1
Introduction.....	2
Le conseil départemental de Maine-et-Loire	3
Présentation du conseil départemental	3
L'unité Système de Production et ses missions	5
La Supervision	6
Présentation de l'outil Nagios.....	7
Etude et mise en place d'une solution libre/open-source de supervision	8
Mon activité au sein de l'unité	9
Mise à niveau technique	9
Découverte, déploiement, installation et configuration de l'outil Nagios	11
Identification des solutions libres/open-sources de supervision	13
Instantané des solutions libres/open-sources.....	13
Etablir les critères de sélection.....	14
Présentation des solutions sélectionnées.....	15
Solution 1.....	15
Solution 2.....	15
Bilan.....	15
Annexes	16
Table des illustrations	17
Sources	18

Rapport rédigé par Thomas CALATAYUD
Étudiant en Master Informatique à l'université d'Angers

Responsable de stage
M. Frédéric LARDEUX
Enseignant chercheur
LERIA, Université d'Angers
frederic.lardeux@univ-angers.fr

Tuteur en entreprise
M. Denis PITHON
Responsable de l'unité système de production
Département de Maine et Loire
d.pithon@maine-et-loire.fr

Remerciements

Introduction

En sortant de ma License Informatique, je souhaitais continuer mon parcours vers un Master Informatique. Deux choix se sont offert à moi : continuer mes études dans un parcours orienté vers la recherche ou dans un parcours professionnalisant.

Mon premier choix était particulièrement porté vers le parcours recherche. Je me suis donc inscrit en première année de Master Informatique qui est générale aux deux parcours, dans l'optique de poursuivre sur une deuxième année en Master Intelligence Décisionnelle.

Il m'a été rappelé qu'il était notamment possible d'effectuer ces deux années de Master en alternance en entreprise.

Malgrès ma passion pour les études, apprendre de nouvelles choses, la découverte et la recherche de nouveauté, l'idée d'effectuer ce Master en alternance m'a plutôt attiré, j'ai donc décidé de me lancer dans la recherche d'une entreprise, prête à m'accueillir et à m'offrir un sujet d'apprentissage qui puisse m'intéresser.

C'est à ce moment, qu'on m'a proposé une offre au Conseil Départemental de Maine-et-Loire. J'ai donc intégré le service Système de Production au sein de la Direction Logistique et Système d'Information en ayant pour projet de remettre à neuf le système de supervision. Pour cela on m'a demandé d'identifier et de proposer une solution libre et open-source et éventuellement de la mettre en place.

J'ai finalement choisis l'apprentissage pour goûter à l'informatique dans un contexte professionnel. Cela me permet d'avoir un peu plus d'expérience professionnelle dans le domaine informatique. Et grâce à cette alternance je peux continuer d'étudier à l'université tout en apprenant de nouvelles chose au Conseil Départemental. Donc j'ai l'opportunité d'étudier et de travailler en ayant un salaire dans un domaine qui me plait. Et j'ai accepter ce sujet dans le but de découvrir l'administration système, un aspect de l'informatique que je ne connaissait pas beaucoup.

Le conseil départemental de Maine-et-Loire

Présentation du conseil départemental

(Voir notes de la réunion d'accueil) Qu'est ce que c'est ? Qu'est ce qu'il fait , Organisation ? (élus, administration territoriale, directions, services ...) - Une collectivité territoriale - Missions/compétences - Organisation

ORGANIGRAMME

du Département
de Maine-et-Loire



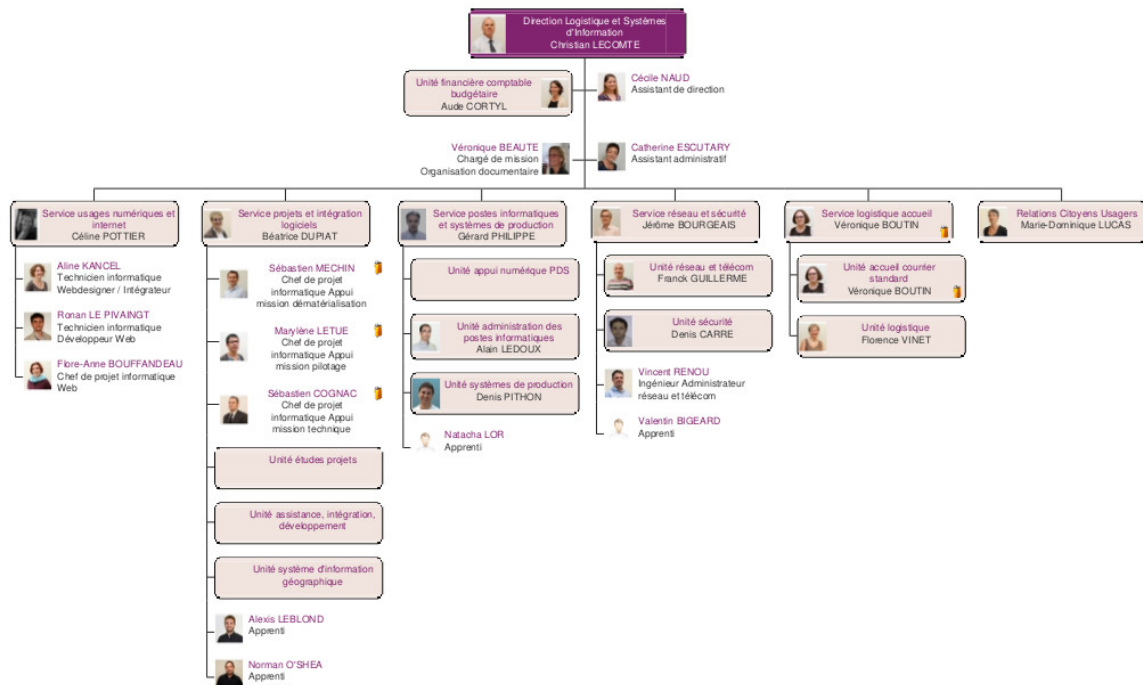
maine-et-loire.fr

Mise à jour - Mai 2017

DÉPARTEMENT DE MAINE-ET-LOIRE
anjou

- La DLSI : qu'est ce qu'on fait ?
- Les différents services
- L'unité système de production

Direction Logistique et Systèmes d'Information

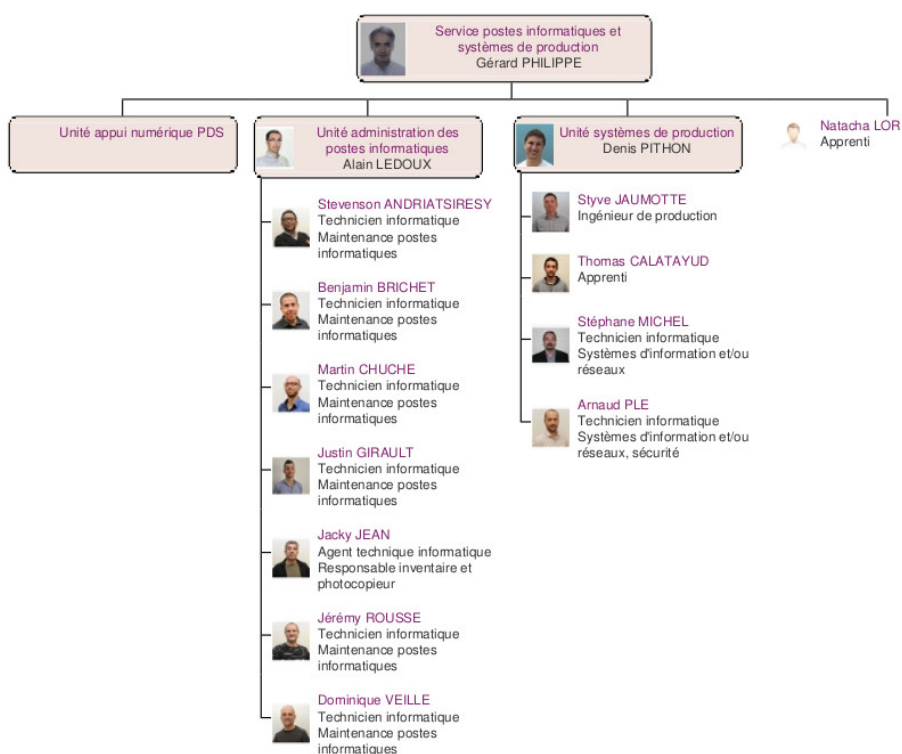


L'unité Système de Production et ses missions

L'unité système de production travaille au sein de la DLSI du Conseil Départemental de Maine-et-Loire sur les problématiques liées au stockage, à la sauvegarde, à la virtualisation et à la supervision des matériels et applications cotés serveurs, ainsi qu'à l'administration des systèmes Linux et Windows.

Quoi ? missions ? chiffres ? → Stockage → Sauvegarde/restauration → Virtualisation → Administration système et bases de données → Gestion des profils/boite mail → SUPERVISION

Service postes informatiques et systèmes de production



La Supervision

La supervision est une fonction permettant d'indiquer, contrôler, commander l'état d'un système ou d'un réseau. Les outils de supervision remontent des informations techniques et fonctionnelles du système d'information. Le tout dans un but de détection et de traitement le plus automatique possible.

L'informatique est intégrée et est devenue un outil indispensable d'une entreprise, quel que soit son secteur d'activité, le système d'information est placé désormais au centre de l'activité de différentes entités métiers et doit fonctionner plainement et en permanence pour garantir l'efficacité de l'entreprise. A tous les niveaux, les réseaux, les terminaux utilisateurs, les serveurs d'applications et les données constituent autant de maillons sensibles dont la disponibilité et la qualité de service conditionnent le bon fonctionnement de l'entreprise.

Il existe deux enjeux majeurs pour les directions informatiques. Le premier est de garantir la disponibilité et les niveaux de service du système en cas de panne ou de dégradation des performances. Le second est de tenter de prévenir en cas de problème et, le cas échéant, garantir une remontée d'information rapide et une durée d'intervention minimale. Ces enjeux sont donc assurés par la supervision.

Ainsi, la supervision inclut plusieurs activités :

- Surveiller
- Visualiser
- Analyser
- Piloter
- Agir
- Alerter

Elle permet de superviser l'ensemble du Système d'Information d'une entreprise :

- Le réseau et ses équipements
- Les serveurs
- Les périphériques
- Les applications
- Le workflow
- ...

Présentation de l'outil Nagios

Nagios, qui s'appelait précédemment NetSaint, est un outil de supervision libre sous licence GPL. Développé en 1996, Nagios, s'architecture autour d'un moteur écrit en C. Il permet d'auditer en permanence des machines, des services sur ces machines, de recevoir des alertes en cas de problème et de disposer d'un tableau de bord de l'état du système à un moment donnée. C'est un programme modulaire qui se décompose en trois parties :

- Le moteur de l'application qui vient ordonnacer les tâches de supervision.
- L'interface web, qui permet d'avoir une vue d'ensemble du Système d'Information et des éventuelles anomalies.
- Les sondes (ou plugins), une centaine de mini programmes/scripts que l'on peut compléter, voir même créer, en fonction des besoins de chacun pour superviser chaque service ou ressource disponible sur l'ensemble des éléments du réseaux du Système d'Information.

Cet outil offre de nombreuses possibilités :

- Superviser des services réseaux (SMTP, HTTP, ICMP, ...)
- Superviser les ressources des serveurs (charge du processeur, occupation des disques durs, utilisation de la mémoire, ...) sur la majorité des systèmes d'exploitation.
- Superviser les équipements réseau (CPU, ventilateurs, ...)
- Superviser les Bases de données
- Superviser l'environnement (température, luminosité, humidité, climatisation, ...)
- Interface via le protocole SNMP
- Supervision à distance via SSH, tunnel SSL ou agent NRPE.
- Remonter des alertes par mails, sms via un système de notification.
- Gestions d'utilisateurs (accès limité à certains utilisateurs)
- Les plugins sont écrits dans des langages de programmation les plus adaptés à leur tâche : scripts shell (bash, ksh, ...), C++, perl, Python, Ruby, PHP, C#, ... et il est possible de créer les siens.

Etude et mise en place d'une solution libre/open-source de supervision

Actuellement, la supervision de l'ensemble du système d'information est opérée par Nagios. Cette solution, en place depuis près de 10 ans, contrôle un peu plus de 2700 points de fonctionnement du SI (espaces disques, sites webs, bases de données, consommations CPU, RAM ...).



Quelques éléments d'information concernant le système d'information :

Virtualisation sur oVirt (Linux/KVM)

- ~ 365 VMs (55% Linux, 45% Windows) réparties sur 42 serveurs physiques
- la moitié de ces VMs servent les applications métiers des 2500 agents
- Stockage NAS (NFS et CIFS) répliqué sur deux salles
- 14 To consommés pour les VMs
- 15 To consommés pour la bureautique
- Supervision avec Nagios

Il m'est demandé dans le cadre de mon apprentissage de :

1. Identifier et comparer les solutions libres/open-sources de supervision
2. Préconiser la solution la plus adaptée aux besoins de l'unité
3. Mettre en place la solution de supervision retenue

Mon activité au sein de l'unité

Mise à niveau technique

Dans un premier temps, à mon arrivé, il m'a été conseillé de commencer par me mettre à niveau, pour gagner en compétences techniques et monter en puissance sur le système. On m'a donc proposé une série de petits exercices à difficulté progressive. Ils ont pour but de me faire progresser sur l'environnement Linux côté serveur et les outils qui lui sont habituellement associés et me familiariser avec l'administration système pour gagner en autonomie.

Avant tout, il a fallu que j'installe et je configure entièrement mon poste de travail sous Linux.

J'ai ensuite découvert l'outil Ovirt que notre unité utilise pour l'installation et la gestion de machines virtuelles, pour installer et configurer une machine virtuelle Windows.

Monter un disque virtuel

On m'a ensuite demandé de construire et monter un disque virtuel de 500 Po sur mon poste. Il a donc fallu que je trouve un moyen de créer un disque réellement utilisable de 500 Po. Il s'est avéré que désormais, le système d'exploitation empêche de manipuler des volumes aussi gros. J'ai donc pu monter un disque d'une taille seulement de 15 To, ce qui reste un disque conséquent.

```
tcalatayud@tcalatayud-CD49:~$ df -lh
...
/dev/loop0      15T  6,3M   15T   1% /media/tcalatayud/e9567653-9578-4332-b449-
37eb63cab7b ①
...
```

- ① J'obtiens donc un disque d'une taille de 15 To sur lequel je peux écrire et lire des fichiers. Cependant, il est bien entendu évident qu'avant de pouvoir le remplir complètement je risque d'avoir quelques problèmes étant limité par la taille du disque dur physique.

Script d'alertes mail, inotify

Il m'a ensuite été proposé, d'écrire un script permettant de transmettre un fichier par mail lorsqu'il apparaît dans un répertoire donnée, puis le supprimer. Il s'agit donc de surveiller un répertoire et lorsqu'un fichier est écrit et est disponible à l'intérieur de ce dossier pour ensuite le supprimer.

Dans un premier temps j'ai écrit un premier petit script en shell bash. Ce script transfère par mail un fichier donnée en paramètre, si ce fichier est dans le dossier surveillé et le supprime une fois qu'il a été envoyé.

J'ai ensuite écrit une deuxième version, améliorée, utilisant le système inotify.

```
#!/bin/sh

inotifywait -mqr -e close_write "/home/tcalatayud/sendMailInotify/dossier" | while
read FILE ①

do
echo $FILE
BEGIN=$(echo $FILE | cut -d' ' -f 1)
END=$(echo $FILE | cut -d' ' -f 3)
echo $BEGIN
echo $END

PATHFILE=$BEGIN$END

echo $PATHFILE

echo "Message" | mailx -a $PATHFILE -S from="t.calatayud@maine-et-loire.fr" -S
smtp=smtp://smtp.cg49.fr -s "Object du message" t.calatayud@maine-et-loire.fr
rm $PATHFILE
done
```

- ① cette deuxième version utilise donc la commande inotifywait qui permet de surveiller un répertoire.

Et enfin, j'ai écrit une dernière version en Python qui s'exécute en tant que daemon, c'est à dire que le script s'exécute en tâche de fond par le système, sans le contrôle de l'utilisateur.

J'y ai inclus la gestion de logs pour qu'on puisse avoir un rapport, si nécessaire, des actions que le script a effectué et pour permettre d'avoir un aperçu du bon fonctionnement et de la bonne exécution du programme.

J'ai notamment utilisé un fichier de configuration .ini qui permet de définir à l'utilisateur et de rassembler des variables dans un même endroit pour pouvoir les utiliser ensuite dans le programme.

```
[config_mail]
fromaddr = t.calatayud@maine-et-loire.fr ①
toaddr = t.calatayud@maine-et-loire.fr ②
server = smtp.cg49.fr ③
port = 25

[config_inotify]
watchFolder = /home/tcalatayud/sendMailPython/dossier

[config_daemon]
pidfile = /home/tcalatayud/sendMailPython/daemon.pid
logfile = /var/log/MyLog/MyScriptDaemon.log
```

Découverte, déploiement, installation et configuration de l'outil Nagios

Dans la continuité de cette mise à niveau, j'ai commencé à regarder l'outil nagios, à voir comment il fonctionne, comment l'installer, comment le configurer, comment l'utiliser.



J'ai donc décidé, pour prendre en main cet outil complexe et puissant, d'installer et de configurer ma propre version de Nagios.

Il m'a donc été nécessaire d'abord de configurer un serveur pour pouvoir le déployer. J'ai donc installé et configuré une nouvelle machine virtuelle sous CentOS. Puis je me suis lancé dans l'installation du Nagios sur lequel j'ai configuré quelques sondes pour comprendre leurs fonctionnement.

- Script de synchronisation
- SAEIR, installation des baies de stockage et de serveurs. Déplacement dans les salles. Et installation du nagios.

Identification des solutions libres/open-sources de supervision

- Problématique :
- Solution vieillissante
- Problème de maintenance
- mise à jour ardue (BSD, versions, ...)

Vu le manque de réactivité du développeur principal de Nagios et sa volonté de ne plus diffuser tous les modules sous licence libre, de nombreux développeurs actifs sur le projet ont fait diverger Nagios. Ainsi, de nombreux outils similaire à Nagios ont été créés. S'ouvre donc l'éventualité de voir et découvrir ces nouvelles solutions.

Instantané des solutions libres/open-sources.

Premier critère : Libre/Open-source

- Inventaire exhaustif
- Donner les différents types de solutions
- Les protocoles utilisés
- Les types de configuration
- Remontée des alertes
- ...

Pour trouver une nouvelle solution de supervision adaptée au besoin de l'unité, il est nécessaire de connaître les enjeux de la supervision, de connaître les solutions existantes sur le marché et ce qu'elles proposent.

Inventaire le plus exhaustif possible, recherche et exploration la plus complète possible et la plus "naïve" possible, avec un regard neuf sur la supervision. En expliquant pourquoi je fais ça. Pourquoi je n'utilise pas de cahier des charges.

→ apparitions des premiers critères évidents (snmp, plugins, alertes, ...), éliminations des solutions qui sont vraiment pas adaptés, premier filtrage.

Etablir les critères de sélection

- Identifier le domaine à superviser (sur le nagios)
- Lister les sondes
- Croiser les éléments ressorti lors de l'inventaire
- Dresser la liste des critères

Comment je les ai établi (recherche plus poussée sur les solutions les plus intéressantes, études des sondes sur le nagios actuel), pourquoi ils sont nécessaires, à quoi ils servent, catégorisation, listes avec explication et appréciations, ...

Présentation des solutions sélectionnées

Présentation des 2 solutions : carte d'identité des solutions

Solution 1

Solution 2

Bilan

Synthèse et ouverture Dire ce que j'ai fais durant cette première année, le déroulement. (Missions annexes) Mon ressenti ce que j'ai apporté, ce qu'on m'a apporté

Annexes

Table des illustrations

Sources