Bac Pro Systèmes Numériques

Dossier de Synthèse

Alternant au sein de IPTWINS

Assistant technicien informatique

Tuteur : Éric DEWITTE

Saifeddine kilani

Du 05/09/2020 Au 31/07/2022

Sommaire

Comme le souligne un célèbre diction espagnol, « Trois, s'aidant l'un l'autre, sont suffisants pour faire le travail de six », l'entraide est un acte des plus nobles. C'est donc tout naturellement qu'avant de débuter, je tenais à dédier une page de ce rapport d'alternance à des fins de remerciement pour toutes les personnes qui m'ont aidé et permis de pouvoir exercer ma passion à plein temps et ce en étant rémunéré et en me formant.

Sans surprise, c'est à mes parents que la première partie est consacrée. Je pourrais facilement disserter et ce durant des centaines de milliards de lignes mais pour faire simple, merci infiniment pour tous ce que vous m'avez donné, c'est grâce à vous que j'en suis là et que je peux vivre cette vie des plus désirables.

Si les mots confiance et bienveillance avaient une définition physique, ce serait sans doute IPTWINS. Nombreux savent que recruter un alternant non majeur et sans aucune expérience concrète dans le domaine est un pari des plus risqués. Cependant, j'ai été accepté là où nombreux ont été refusé, c'est donc naturellement que je remercie tous mes collègues d'entreprise et mon tuteur qui accepte de partager son savoir avec moi.

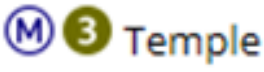
Pour bien finir, c'est avec mes ami(e)s et mes anciens professeurs que je vais finir cette parenthèse des plus importantes. Comme je dis souvent, une personne est définie entre autres par son environnement. C'est donc en partie grâce à eux si je suis ici aujourd'hui.

Enfin, ne perdons pas le nord. C'est sur ces belles phrases que je vais commencer à vous conter le contenu de mes années d'alternances.

Partie I/ IPTWINS

I/ Présentation de l'entreprise

Créé en 2002, IPTWINS est un bureau d'enregistrement de noms de domaine dédié aux titulaires de marques et à leurs représentants, ainsi qu'une société de protection des marques en ligne. IPTWINS a son siège à Paris et des filiales à Hong Kong et Singapour, ainsi que des antennes à Manille et en Espagne.



Depuis sa création, IPTWINS n'a cessé d'évoluer avec un capital social de 41 050 € et un chiffre d'affaires en 2020 de 2 103 249 €. Elle peut aussi compter sur ses 13 employés qui sont répartis en 8 juristes, 3 IT et enfin 2 directeurs.
Mû par un sens relationnel aigu, IPTWINS cultive une vision humaine du métier. IPTWINS veut avant tout fournir à ses clients des conseils pertinents et des services sur mesure tout en conservant la qualité d'échanges que l'on est en droit d'attendre de juristes spécialisés.

IPTWINS travaille, constamment pour développer des technologies et des méthodes assurant la fiabilité et la précision dans l'administration des noms de domaine et la protection des marques des clients. C'est pourquoi IPTWINS a développé différents outils spécifiques, dont Identitool, Similaritool, Domainarium et Detective .

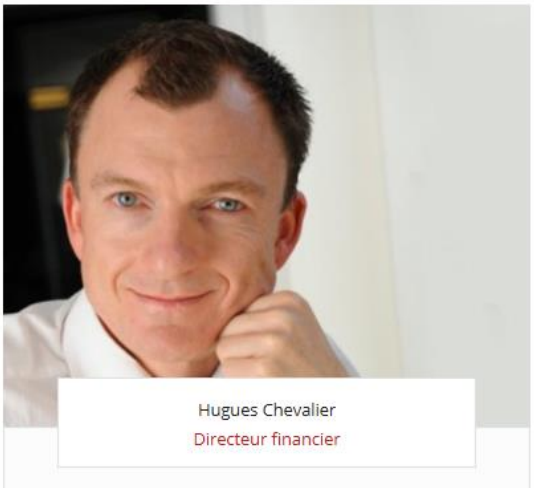
Pour plus de précision, voici la fiche d'identité de l'entreprise :

SIRET	**441 049 376 00045**
Dénomination sociale	IP TWINS
Raison sociale	SAS
Adresse	78 RUE DE TURBIGO 75003 PARIS
Téléphone	01 48 78 93 12
Effectif	13
Site (url)	https://www.iptwins.com/
Mail de contact	info@iptwins.com
Chiffre d'affaires 2020	2 103 249 €
Résultat 2020	107 845 €

II/ Présentation du personnel en images

Direction

Sylvain Hirsch
Président & fondateur



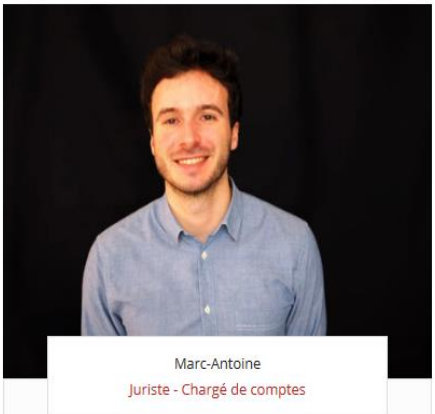
Hugues Chevalier
Directeur financier

Juristes & Chargés de comptes

Anatole
Juriste - Chargé de comptes

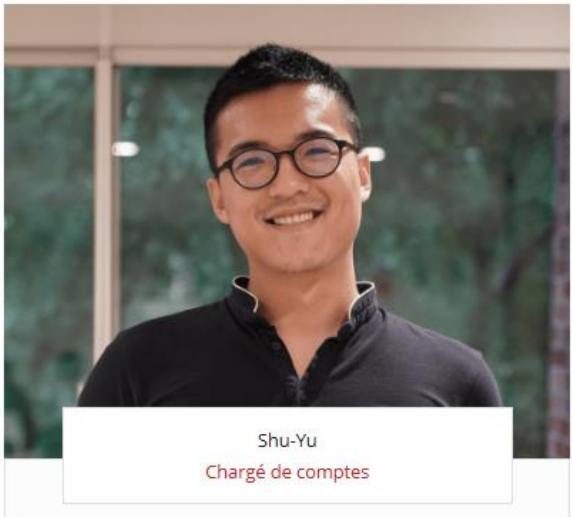
Arthur
Juriste - Chargé de compte

Emmanuel
Juriste - Consultant

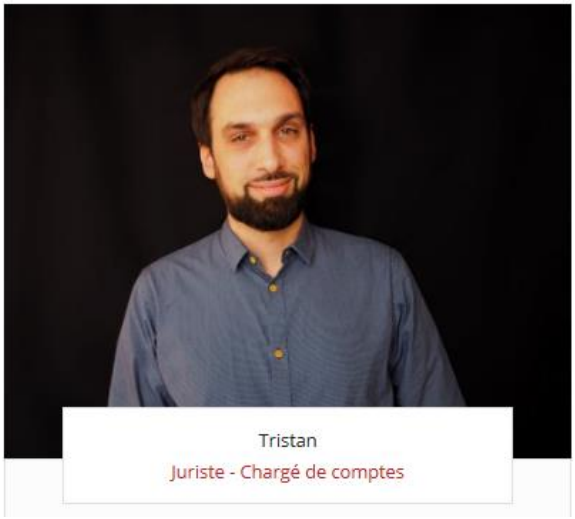


Marc-Antoine
Juriste - Chargé de comptes

Marco
Chargé de comptes

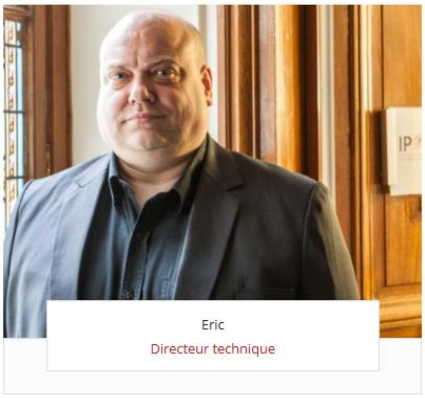


Shu-Yu
Chargé de comptes



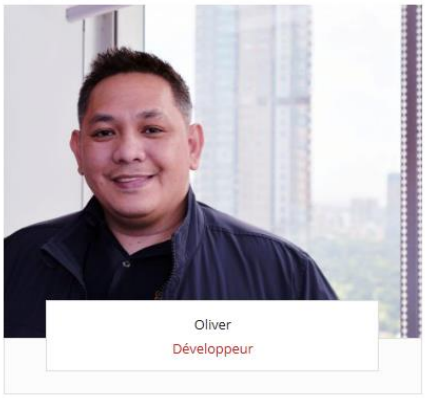
Tristan
Juriste - Chargé de comptes

IT & Administratif



Eric
Directeur technique

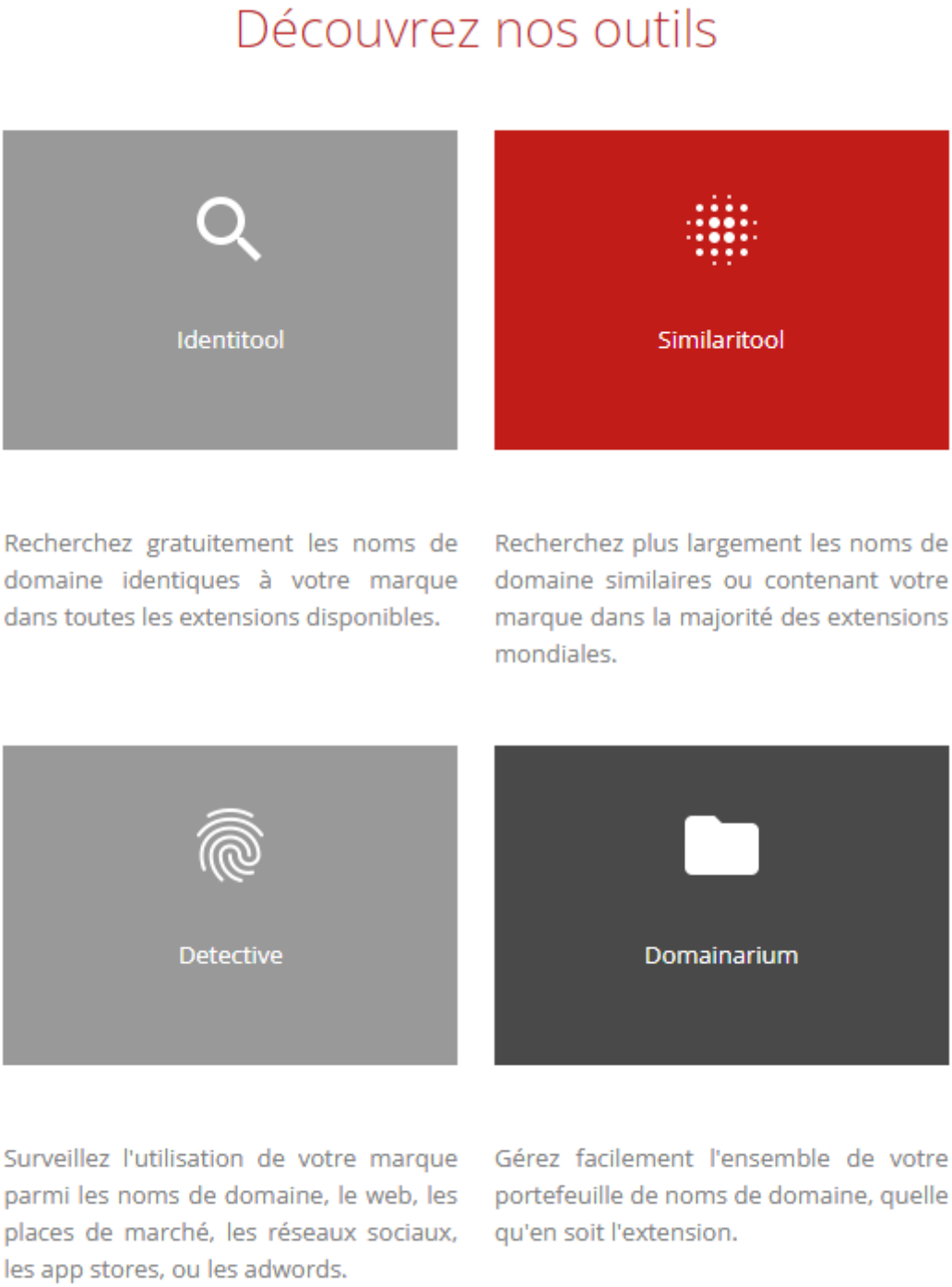
Vincent
Développeur



Oliver
Développeur

III/ Activité de l'entreprise

Les activités de IPTWINS sont multiples, nous allons donc aborder sous deux grandes parties, la gestion de noms de domaine et la protection :



Nous allons commencer avec la première activité de IPTWINS, l'enregistrement et la gestion de noms de domaine :

Pour pouvoir exercer au mieux cette tâche des plus importants pour les entreprises, IPTWINS a créé deux outils très innovants répondant aux doux noms de Identitool et Similaritool :
Identitool permet la recherche, gratuite, de noms de domaine identiques à une marque dans toutes les extensions disponibles.
Similaritool, lui, permet la recherche plus large de noms de domaine similaires ou contenant votre marque dans la majorité des extensions mondiales.

Ainsi, IPTWINS permet l'enregistrement de noms de domaine dans près de 2000 extensions. Cette méthode d'enregistrement est ciblée dans le but d'enregistrer uniquement les noms de domaine dont on a vraiment besoin.

Enfin, IPTWINS sait à quel point la gestion de ces noms n'inspire guère simplicité et c'est pour remédier à ce problème que IPTWINS a créé l'outil adéquat à cette situation, Domainarium.
Cette plateforme accessible par tous nos clients permet de gérer facilement l'ensemble des portefeuilles de noms de domaine, quelle qu'en soit l'extension et ce sur le même site. Vous en conviendrez, plus simple n'existe pas !

Nous allons maintenant passer à la deuxième partie concernant la manière d'acquérir un portefeuille complet :

Premièrement, IPTWINS a créé des algorithmes permettant d'acquérir des noms de domaine tiers en cours d'abandon, pour valoriser votre portefeuille.

De plus IPTWINS a acquis une grande expertise dans les négociations de rachat de noms de domaine, enregistrés légitimement ou de mauvaise foi. IPTWINS rachète les domaines pour votre compte sous une identité d'emprunt, évitant ainsi des prix trop élevés. Cela s'ajoute à nos recours à des tiers de confiance pour assurer la sécurité de toutes les transactions.

Enfin, nous allons aborder la manière de régler des litiges. En effet, le marché des noms de domaine offre aux cybersquatteurs toujours plus d'opportunités. Et ces derniers maitrisent de mieux en mieux la négociation et les failles juridiques. C'est pourquoi l'équipe d'IPTWINS est principalement composée de juristes spécialisés en droit de la propriété industrielle et en droit des technologies de l'information.
À ce titre, IPTWINS collecte les preuves et les données utiles pour conseiller et accompagner ses clients à toutes les étapes d'un litige en matière de nom de domaine.

Nous allons maintenant passer à l'avant dernière partie qui va présenter les services fournis par IPTWINS dans le domaine du certificats SSL et des sécurités avancées :

IPTWINS fournit des certificats de sécurité parfaitement adaptés aux besoins pour garantir la sécurité des visiteurs de votre site Internet et met en œuvre les procédures les plus rigoureuses pour éliminer les risques de détournement administratif et technique des noms de domaine de nos clients.

-Pour finir, voici la dernière rubrique concernant la Gestion DNS :

IPTWINS dispose d'un réseau de serveurs localisés sur tous les continents, ce qui nous permet de garantir un uptime de plus de 99,999 % sur les services actifs des noms de domaine de nos clients.

Pour continuer, chez IPTWINS, aucune modification sur les fichiers de zone de noms de domaine de nos clients n'est possible si elle n'est pas valide techniquement, pour empêcher toute interruption de service.

Enfin sur les noms de domaine sensibles, IPTWINS demande automatiquement l'accord des personnes autorisées pour modifier les enregistrements techniques.

Nous allons continuer et terminer cette partie avec l'Anti-Contrefaçon (Surveillance et Protection):

Avec Detective, IPTWINS identifie les enregistrements de noms de domaine identiques ou similaires à la marque de nos clients, dans toutes les extensions de noms de domaine, y compris les nouvelles extensions. Tous les systèmes d'écriture intégrés dans le système de nommage (DNS) sont couverts par nos services.

Puisqu'il peut être utile de suivre l'évolution d'un nom de domaine, IPTWINS propose gratuitement une surveillance du statut de ce nom de domaine, entièrement personnalisable : changement de titulaire, de bureau d'enregistrement, d'adresse IP ou de serveur de messagerie, mise en vente du nom ou expiration. Les clients reçoivent des alertes dès qu'un changement se produit.

Enfin, Grâce à Detective, IPTWINS identifie l'utilisation d'une marque sur internet par des tiers non autorisés, et tous les supports classiques sont couverts : presse, forums, places de marché, sites de vente en ligne, réseaux sociaux, logos, app stores, liens commerciaux géolocalisés.

Nous surveillons les réseaux sociaux les plus fréquentés, plus de 140 places de marché et tous les résultats indexés par les moteurs de recherche partout dans le monde et dans toutes les langues, selon les besoins de nos clients.

IV/ Présentation de la branche informatique de l'entreprise

IPTWINS compte sur deux types de partie dans la branche informatique, la partie réseau et les développeurs web. Tous deux fort importants pour le bon fonctionnement du travail de chacun chez IPTWINS.

Actuellement stagiaire de Éric Dewitte, directeur technique, il effectue des taches très importantes pour l'entreprise comme pour certains clients. En effet, Éric gère l'administration de tous les serveurs de l'entreprise comme la mise à jour de ces derniers. C'est lui qui s'occupe de l'installation de logiciels professionnels et de la sécurisation des postes de travail. Il s'occupe aussi de faire des pointages DNS pour certains clients. Il s'occupe de toutes les facettes qu'implique la gestion de serveurs, ce qui est fort importants pour l'entreprise. Il s'occupe aussi du matériel physique et numérique de l'entreprise. Effectivement, il gère lui-même la configuration des postes de travail. Il gère aussi le « share » système de partage de documents en local.

De l'autre côté, Vincent et Oliver, sont les développeurs, c'est donc eux qui ont créé les différentes plateformes citées dans les parties précédentes. Ils sont responsables de tous éventuels problèmes qui pourrai apparaitre au niveau de l'utilisation, c'est donc eux qui les réparent. Comme tout site avancé, une base de données est nécessaire. C'est donc eux qui s'en occupent et qui la maintiennent à jour.

V/ Prestataires, clients et partenaires

Voici quelques clients de IPTWINS :

Ensuite, voici quelques prestataires de IPTWINS :

Enfin, voici quelques partenaires de IPTWINS :

Partie II/ Moi chez IPTWINS

I/ Curriculum Vitae

Tout d'abord, voici le CV que j'ai envoyés à IPTWINS suite à l'appel téléphonique que j'ai eu avec Hugues Chevalier :

Saïfeddine Kilani
22 bis rue Michelet
93170 Bagnolet

15 ans
nationalité française
06.49.98.14.18
mohamed.kilani@laposte.net

Préparer un Bac pro en systèmes numériques

Formation :

2019-2020 : seconde générale et technologique. Lycée Saint-Benoist de l'Europe. Bagnolet

Juin 2019 : DNB série générale (mention bien). Collège Saint-Benoist de l'Europe. Bagnolet

Févier 2019 : stage d'observation au centre d'études et de recherches en informatique et communications du CNAM de Paris : découverte du métier de programmeur et des différents types de logiciels pour programmer robots, applications, images, images en mouvement.

Logiciels de programmation : Scratch, note pade++, Ardino, processing…

Bureautique : Word, Power Point, internet, Paint.

Langues :

Anglais : lu, parlé, écrit. Actuellement en préparation du Preliminary English Test de l'université de Cambridge et du Diplôme National en Linguistique.

Espagnol : lu, parlé, écrit

Arabe : lu, parlé, écrit

Activités extra-scolaires :

Basket-Ball (championnat de France en 2019 et en cours de qualification pour le championnat de 2020)

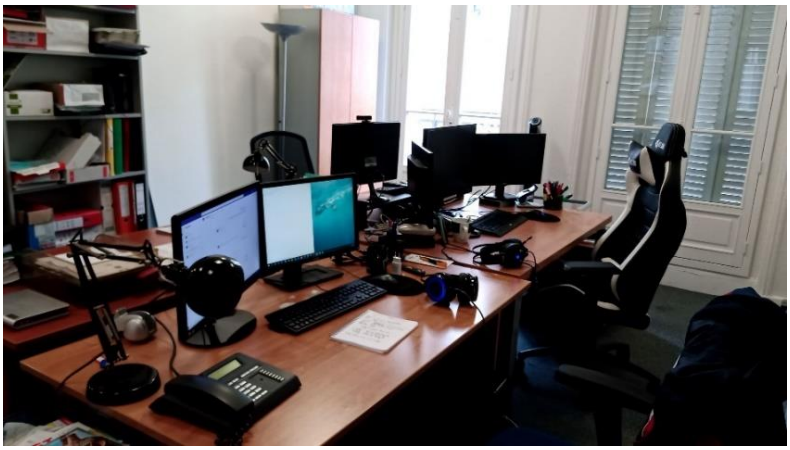
Vô-Vietnam (art martial vietnamien) depuis 7 ans

Tennis de table depuis 3 ans

II/ Mes début chez IPTWINS

Mes débuts à IPTWINS ont été marqués par la création de différents comptes pour que je puisse travailler convenablement. Mon premier mail fut celui de mon tuteur qui m'a souhaité la bienvenue et qui a commencé à me donner les premières consignes qui étaient de lire des documents. Ils évoquaient les bonnes pratiques et ce qui est déconseillé en matière de sécurité pour la messagerie, les mots de passe, le chiffrement de données, la navigation sur internet et enfin la politique de sécurité concernant la sauvegarde de donnée sur les postes de travail. Je me suis vite rendu compte que dans ce domaine, rien n'est laissé au hasard.

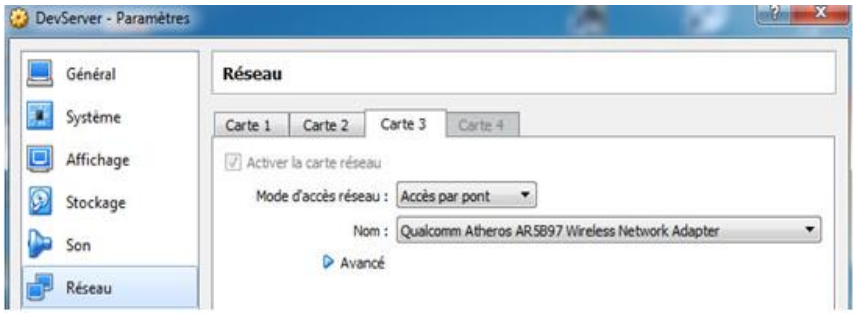
En effet, en tant que registrar, nous avions accès à beaucoup d'informations privées à propos des différentes entreprises. C'est donc logique que tout était mis en œuvre pour garantir une sécurité maximale. Ce jour-là, j'ai aussi découvert mon bureau, le lieu où j'allais passer mes deux années à apprendre l'art de l'informatique, juste à ma droite se trouve le poste de Eric, mon tuteur. Vous en conviendrez un placement stratégique car près de la cuisine.



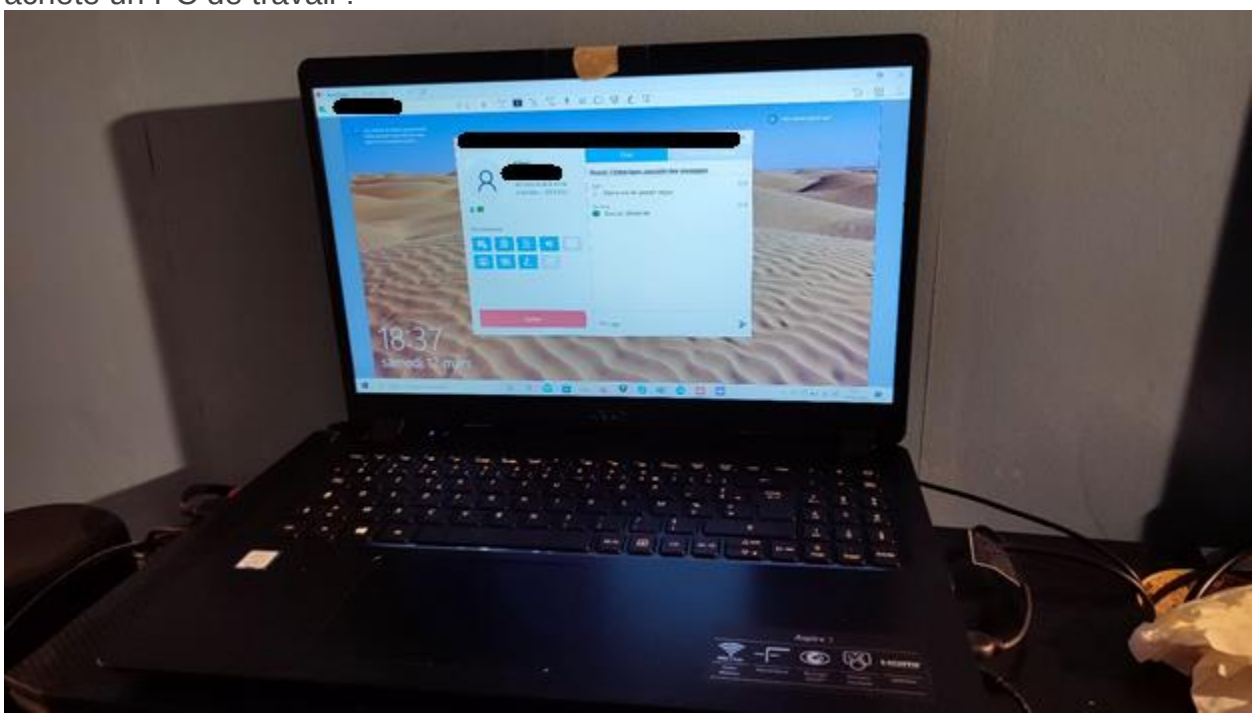
Le deuxième mail de mon tuteur contenait ma première manipulation : la création d'une VM, machine virtuelle. Pour que je puisse commencer à apprendre comment fonctionnent les logiciels de création de VM, Eric m'a donné différents liens amenant à OpenClassrooms. Vous l'aurez compris, mes débuts ont été essentiellement composés de séance de lecture et d'apprentissage. Pour faire suite à cela, mes premières taches sur mes VM furent de configurer le fichier /etc/network/interface avec les bonnes valeurs et mettre la configuration réseau en accès par pond pour que la machine puisse avoir accès internet :

```
root@debian-preprod:~# more /etc/network/interfaces
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
address 192.168.1.116
netmask 255.255.255.0
gateway 192.168.1.254
```

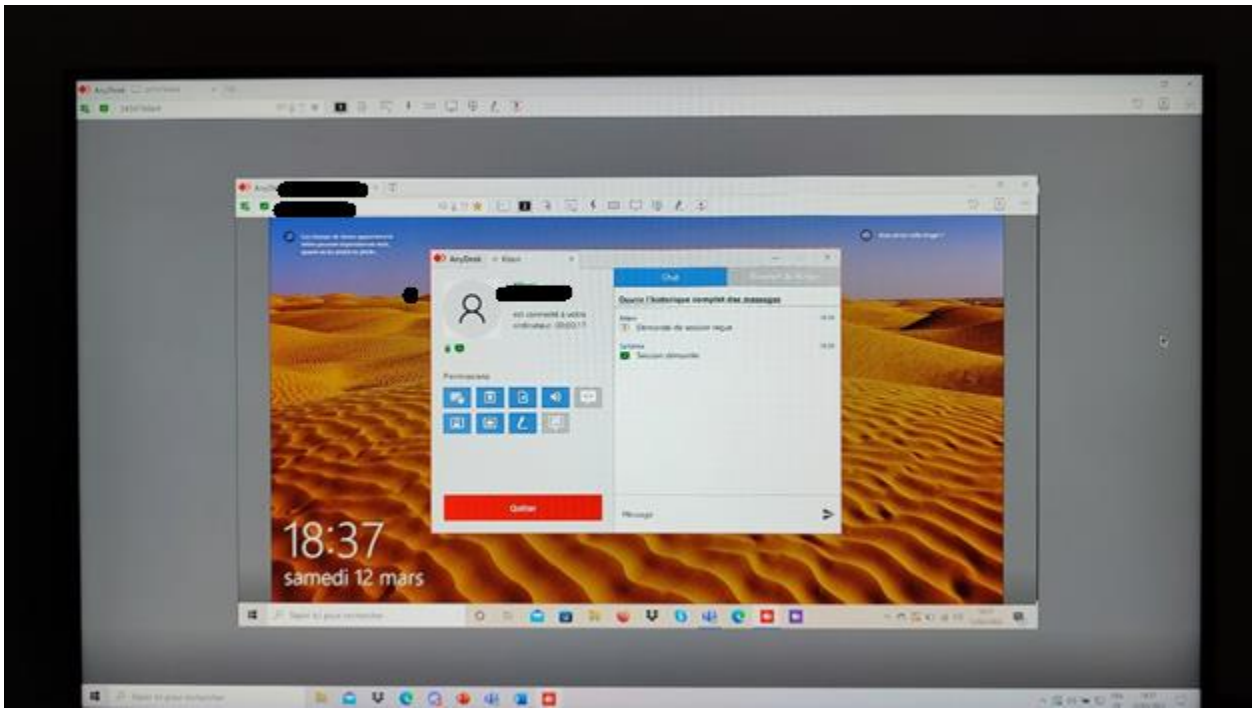
Indique que la configuration sera statique
Indique l'IP
Indique le masque de sous réseau
Indique-le « pond » à emprunter pour accéder à internet



Cette phase a aussi beaucoup été marquée par le télétravail causé par la covid-19. En effet, le personnel de IPTWINS a du travailler 3 jours sur 5. C'est ainsi que IPTWINS m'a acheté un PC de travail :



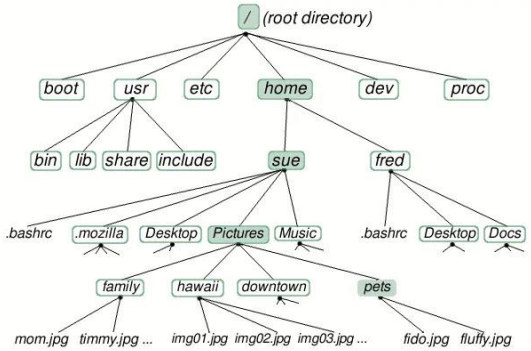
Cependant, je ne pouvais pas faire tout ce que je faisais au bureau. J'ai donc dû commencer à utiliser le logiciel Anydesk. C'est un logiciel de contrôle à distance d'un PC. Dans mon cas, je l'utilisais pour contrôler mon PC au bureau :



III/ Ma phase d'apprentissage

L'une de mes premières découvertes fut celle des différents utilisateurs et leurs droits. Comme utilisateur par défaut, il y a root. Root a tous les droits, en gros c'est l'administrateur. Le deuxième utilisateur lui est créé lors la création du serveur. Si on applique aucun changement, il n'a pas tous les droits. Il ne peut pas modifier un fichier créé par root. Il ne peut pas apporter de modification aux fichiers jugés importants. Ainsi, une solution s'offre à nous si on ne veut pas tous le temps embêter root pour faire des modifications : l'utilisation de sudo. Cette commande donner le droit root lors d'une action. Seulement pour utiliser sudo, il faut que cet utilisateur soit nommé dans le fichier */etc/sudoers.*

Toujours dans l'optique de me former à la manipulation de l'os LINUX, je multipliais la création de VM Debian et à chaque fois, à l'aide des cours de OpenClassrooms, j'apprenais de nouvelles facettes de cet OS. J'ai rapidement découvert que LINUX est construit de manière pyramidale. En effet les données étaient enregistrées dans différents répertoires :



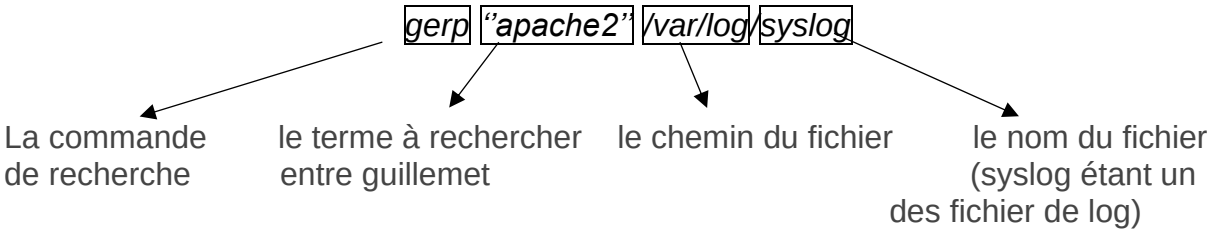
J'ai aussi découvert le monde des commandes. En effet, selon les préinstaller qu'on choisit, il se peut qu'on se retrouve sans interface graphique. Habitué à toujours avoir une interface, cette découverte fut surprenante. Pour pouvoir installer ou naviguer dans le serveur, il n'y a pas de souris. Tout se passe en ligne de commande. Pour aller dans un répertoire, on utilise la commande « *cd* » suivi du chemin.

Comme commande très importante, on retrouve les commandes de mise à jour. Ce sont les commandes *apt-get update* et *apt-get upgrade.* Comme autre commande importante, on retrouve celle qui permet d'installer un programme ou un service. Pour ce faire, il faut avoir un accès à internet, et sur une interface de commande, taper *apt-get install* suivi du nom du programme. Par exemple, si on veut installer un serveur web (apache) qui va nous permettre de créer des pages web, on peut taper cette commande : *apt-get install apache2*.

Pour modifier un fichier, on utilise nano suivi du chemin et le nom du fichier en question. Pour voir le contenu d'un répertoire, on fait *ls* suivi du chemin. Tout ça pour dire que je passais beaucoup de temps à découvrir de nouvelles commandes et répertoires. Comme exemple, il y a le répertoire */var/log/* qui contient des fichiers de log en tous genre (log = historique de tous ce qui se passe dans le serveur).

Ce répertoire est donc très important : si on n'arrive pas à installer un paquet ou tout autre problème, les logs nous aideront à comprendre pourquoi cela n'a pas fonctionné. Seulement un problème apparaît. Si vous êtes perspicace vous l'aurez deviné : comme l'historique de tout ce qui se passe dans le serveur est dans ce répertoire, les fichiers sont extrêmement longs.

Autant dire que pour trouver ce qu'on cherche dans ces fichiers de millions de mots serait comparable à chercher une aiguille dans une botte de foin. Mais à tout problème, une solution. Il y a une commande qui permet de retrouver une ou plusieurs mots dans un fichier. Cette commande répond au nom de *grep*. Pour l'utiliser il faut indiquer le ou les mots que l'on veut retrouver et le chemin du fichier. Dans le cas où on n'arrive pas à installer son serveur apache et qu'on veut savoir pourquoi via les logs : on peut effectuer la commande :



IV/ Mes premières taches concrètes sur les serveurs et en physique

Après plusieurs semaines d'apprentissage vint enfin le début des taches concrètes que ce soit sur les serveurs numériques ou sur les postes locaux. IPTWINS détient beaucoup de serveurs dont les serveurs appelés screenshot et puppeteer qui effectuent différentes actions telles que la surveillance de place de marché. Les puppeter sont au nombre de 22 et les screenshot sont 32. Pour s'y connecter, il faut aller sur le logiciel Secure CTL. Celui-ci va vous demander le host Name ou l'ip, le port, l'utilisateur et enfin le mot de passe.

L'une des premières taches fut de mettre à jours ces serveurs. Etant donné que ce ne sont pas des serveurs critiques, Éric me permit de commercer mes premières taches concrètes dessus. Je n'avais pas besoin de taper les commandes à la main car mon tuteur avait créé un script de mise à jour qui s'appelle *groupdate.sh* et qui se situe dans le répertoire de *root*. Pour l'activer, il faut faire /root/goupdate.sh. Ce fichier contient les commandes (l'explication de ces commandes est dans une partie future)

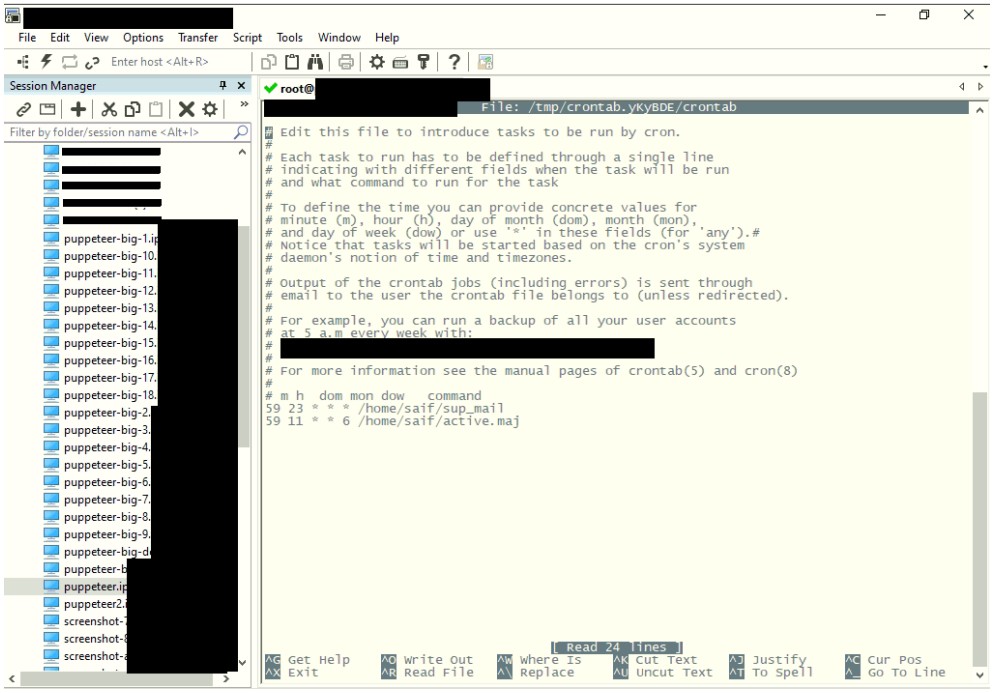
```
apt-get update
apt-get upgrade -y

more /var/log/dpkg.log | grep "status installed" > rep3
```

Une fois le résultat affiché, je n'ai plus qu'à prendre les lignes qui indiquent ce qui a été mis à jour et le mettre dans un fichier Excel qui sert de registre de ce qui est mis à jour par les serveurs.

La deuxième grande tache que j'ai effectuée sur les serveurs a été la création d'un script de suppression de log jugé inutile par mon tuteur et qui prennent beaucoup de place. J'ai découvert les commandes echo > /root/mail qui permettent de supprimer les log situer dans le répertoire /root/mail . J'ai aussi découvert une nouvelle fonctionnalité des

serveurs LINUX, le *crontab*. Il faut se dire que c'est beau de créer des scripts mais si on doit les taper à la main ça ne sert presque à rien car on ne gagne pas de temps. C'est là que *crontab* vient à la rescousse. *Crontab* est un fichier modifiable grâce à la commande *crontab -e* et qui permet l'activation de manière automatique un script :



La ligne qui nous concerne est la ligne `59 23 * * *` `/home/saif/sup_mail`,

Cette série de nombres et d'astérisques définit tout à quelle heure le script va s'activer (ici le script va s'activer tous les jours à 23h59)

Cette partie de la ligne indique la localisation du script (le chemin)

Nous allons maintenant aborder les taches concrètes en physique. Tout d'abord, il y a une multitude d'actions que j'ai menée dans la baie serveurs. C'est une salle sécurisée par une serrure nécessitant un badge se situant dans les locaux. C'est ici qu'on trouve les serveurs, des switches et routeurs. Il y a aussi la box fibre et les postes pour le personnel qui travaille constamment en distanciel. J'ai été amené à ajouter les postes dans cette salle, à en replacer. A redémarrer manuellement certains serveurs en cas de nécessité. J'y ai aussi fait du câble management. Voici ci-dessous différentes images pour illustrer mes propos

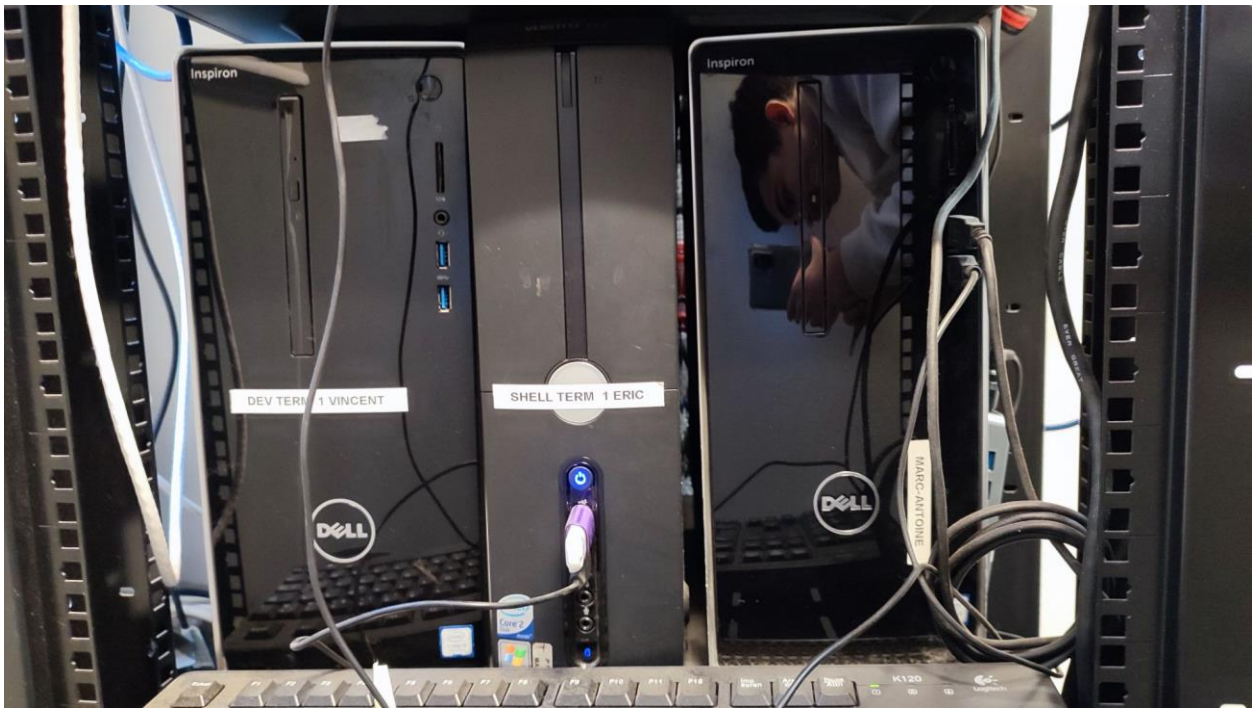
Voici une photo de la baie serveurs :

Voici le système de sécurité pour accéder à la baie serveurs, vous l'aurez deviné, on ne peut y accéder que grâce à ce badge :

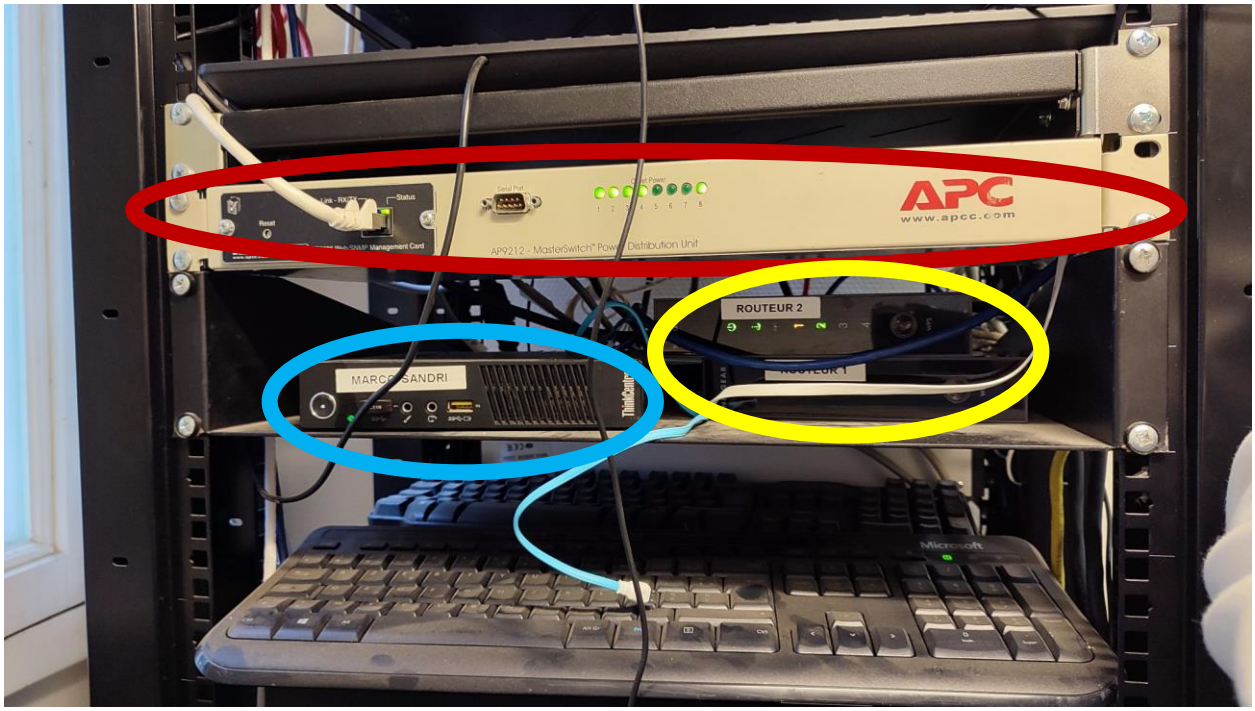
Ci-dessous une image du system de refroidissement de la baie :

Vous avez ici une photo des onduleurs de la baie serveurs. Ce sont des boitiers qui permettent en cas de panne de courant d'assurer le relais électrique pendant quelques minutes.

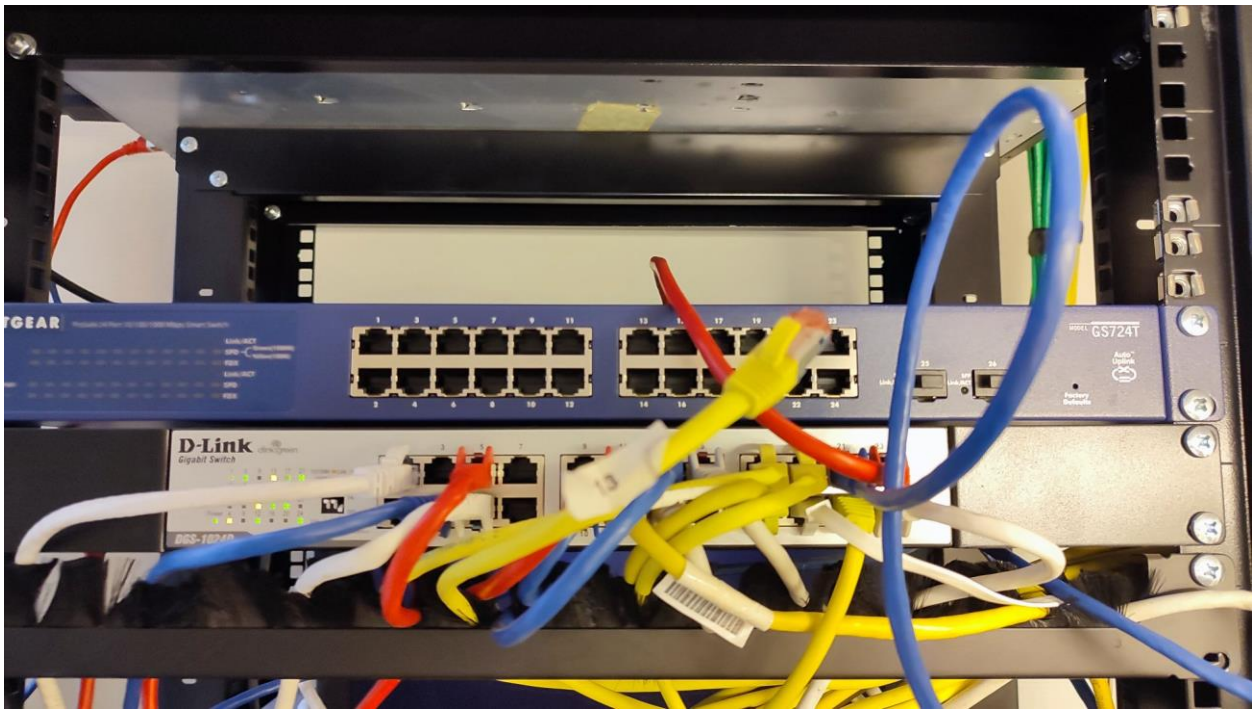
Vous pouvez ici voir 3 ordinateurs mis dans baie pour les isoler du réseau, c'est une mesure de sécurité appliquée pour certains ordinateurs qui remplissent certaines conditions : poste sensible, poste d'administration …



Vous pouvez percevoir ici : un switch électrique entouré en rouge, deux routeurs entourés jusqu'à en jaune et enfin un autre ordinateur entouré en bleu.



Ici vous avez deux switch réseau :



Pour finir, vous avez ici deux serveurs dell qui nous servent pour différents services : stockage de données, administration Active Directory …

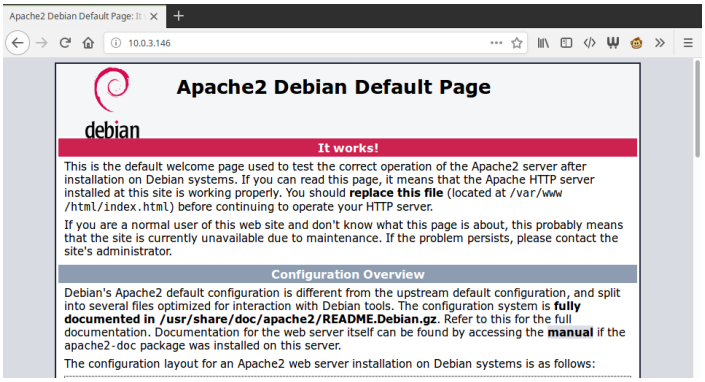
J'ai aussi été amené à réinstaller Windows sur des postes où il ne fonctionnait plus correctement. C'est aussi moi qui ai mené les premières configurations sur les nouveaux postes. Entre autres, il y a celle de mettre une ip et tout ce qui va avec.

J'ai aussi changé plusieurs fois des batteries d'onduleurs.

Batterie d'onduleur Onduleur

V/ Mes débuts dans le développement

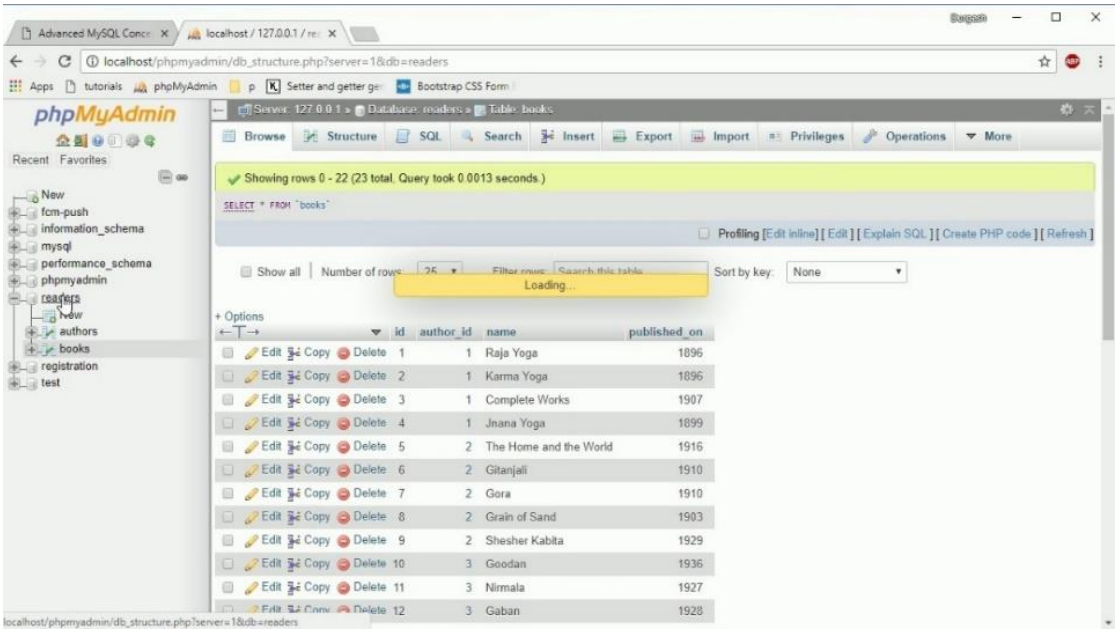
Après un petit moment, Eric me proposa de commencer à me former au développement via OpenClassrooms et m'a donné comme défit de créer une page web qui devrait communiquer avec une base de données. L'idée m'a plu, c'est donc comme cela que j'ai commencé à créer une VM Debian. Une fois toutes les configurations faites, j'ai testé plusieurs manières de créer un site avec base de données. Après de multiples ratés, j'ai décidé de tout simplement installer xampp. XAMPP contient plusieurs programmes entièrement gratuits et faciles à installer tels qu'Apache (pour créer le site), MySQL (la base données), PHP (un langage qui permet de rendre son site interactif) et Perl. Le paquetage open source XAMPP a été mis au point pour être incroyablement facile à installer et à utiliser. Une fois tous ces logiciels installés, il n'y a plus qu'à taper son adresse ip dans la barre de recherche, si tout marche bien et que le serveur apache est en marche, une page comme celle-ci devrait apparaitre :



Pour vérifier si xampp est bien installé, taper son adresse ip puis */dashboard* et une page similaire devrait apparaitre :

Une fois toutes ces étapes effectuées, j'ai pu passer à l'étape suivante, rentrer dans le vif du sujet, la création des différentes pages. Il y a un fichier s'appelant */opt/lampp/htdocs/* et qui permet de créer des pages qui vont pouvoir s'afficher via internet. C'est donc ici que j'ai commencé à découvrir la base de la base, le html. Le html permet de créer une page statique. En effet, avec les différentes balises, on peut écrire du texte de la couleur que l'on veut et le placer ou l'on veut avec la police que l'on veut. On peut mettre des couleurs

sur la page et bien d'autre choses. A cet html va venir s'ajouter du css. Ce langage a un seul but, embellir la page html. Cet outil va permettre de rendre la page plus accueillante. Une de cette fonctionnalité est de permettre de mettre des images facilement. Enfin le dernier langage qui vient se mêler aux deux autres sont le php. Ce langage est une pièce maitresse. En effet, un site bien présenté mais qui n'est pas interactif n'est pas intéressant. C'est donc ici que le php allié au phpmyadmin et au mysql va donner de la vitalité au site. Phpmyadmin est un programme permettant d'enregistrer des données tels que les identifiants et les mots de passe et bien d'autre choses. Et dans un autre temps de les restituer, ce qui va permette au visiteur de se connecter et bien d'autres fonctionnalités. Voici à quoi ressemble l'interface de phpmyadmin :



Après arrive la partie la plus dure pour moi et heureusement que Vincent, le développeur de IPTWINS, était là pour m'expliquer et m'aider. Il faut s'imaginer qu'il y a deux manières d'entrer une information dans un tableau phpmyadmin. Soit on le fait depuis l'interface graphique ci-dessus. C'est assez simple mais ça signifie que pour toute modification, ajout, suppression, il faut passer par la page de phpmyadmin et donc il faut avoir les accès. Vous imaginez bien que ce n'est pas faisable. Le visiteur doit pouvoir s'inscrire depuis le site. Et c'est là qu'intervieni la deuxième solution. Via le langage SQL, on peut écrire plusieurs lignes de code qui vont permettre au visiteur de s'inscrire depuis un formulaire sur le site et les informations qu'il aura saisi vont venir s'enregistrer dans la base de donner de phpmyadmin. Le seul défaut de cette technique est qu'il n'est pas évident pour un débutant de faire tous ces commandes :

```
<?php

      $bdd = new
PDO('mysql:host=localhost;dbname=Incidents_interne;charset=utf8', 'root', '');

  $requete = $bdd->prepare('INSERT INTO incidents(event_user, event_objet,
event_com, event_dt_debut) VALUES(?, ?, ?, ?)');

  $requete->execute(array($_POST['event_user'], $_POST['event_objet'],
$_POST['event_com'], $_POST['event_dt_debut']));
?>

La ligne vient d'être ajouter, pour aller voir le tableau, cliquer <a
href=connexion.php/> ICI </a>
```

Pour expliquer de manière simple, on ouvre les balises qui signifie que l'on va s'exprimer en langage php : *<?php […] ?>* . On a ensuite les lignes

```
        $bdd = new
PDO('mysql:host=localhost;dbname=Incidents_interne;charset=utf
8', 'root', '');
```

Celles-ci définissent la connexion à la base de données, on définit où est-ce qu'est situé phpmyadmin. Ici c'est en *localhost*, cela veut dire que phpmyadmin est sur ce pc. Après on définit sur quel tableau on veut ajouter quelque hose. Et enfin, on met l'identifiant et le mdp, ici l'identifiant est root et il n'y a pas de mot de passe. Bien sûr, les valeurs indiquées ici ne sont valables que dans mon cas. Il y a aussi des variables :

```
    $requete->execute(array($_POST['event_user'],
$_POST['event_objet'], $_POST['event_com'],
$_POST['event_dt_debut']));
```

Les variables seront définies par les réponses de l'utilisateur à un questionnaire posé avant cette étape. Voici à quoi ressemble le script qui va créer le formulaire :

```
<html>
    <head>
        <title>Inscription</title>
    </head>
    <body>
        <form action="" method="">
            nom : <input type="text" value=""><br>
            prénom : <input type="text" value=""><br>
            age : <input type="number" value=""> ans<br>
            telephone : <input type="number" value=""><br>
            adresse : <textarea></texta<br>
            login : <br>
            mot de passe : <br>
        </form>
    </body>
</html>
```

VI/ Obtention d'un parc serveur & Mon script d'automatisation de MAJ

Peu de temps après le début les cours, j'ai obtenu ''mon'' Park serveur. En effet, Eric a décidé de me « donner » les machines Screenshot et Puppeteer pour que je m'en occupe au niveau des mises à jour. Ainsi, à chaque début de semaine en entreprise, je passais une journée à faire la mise à jour des serveurs. Après plusieurs semaines en entreprise, je m'étais fait une remarque. Si je passe deux journées par mois à faire des mises à jour, en deux ans, je perds un mois, deux semaines et quatre jours à faire des mises à jour. C'est à partir de ce constat que m'est venue l'idée de faire un script de mises à jour automatiques qui répondent aux mêmes exigences que celui qui s'effectue manuellement à savoir : mettre à jour et afficher ce qui a été mis à jour.

A partir de là, plusieurs solutions se présentèrent à moi : mettre le script goupdate.sh dans un crontab et puis chercher manuellement ce qui a été mis jour. Cette solution n'était pas intéressante car on devait se connecter sur chaque serveur pour prendre ce qui a été mis à jour. Ensuite, je me suis demandé comment faire pour que je reçoive un compte rendu de ce qui a été mis à jour. C'est alors que Eric me parla de postfix. Un service qui sert à envoyer des mails. A partir de là, tout s'est accéléré. Eric m'a demandé de créer une VM pour pouvoir tester la Alpha de mon script et les installations nécessaires. J'ai d'abord installé et configuré postfix : *apt-get install postfix sasl2-bin*. Après quelques dizaines de tuto pour envoyer un mail depuis un serveur, j'avais réussi à envoyer un mail depuis ma VM vers mon mail professionnel grâce à la commande :

```
root@puppeteer:/home/saif# sendmail saifeddine.kilani@iptwins.com
From:iptwins@iptwins.com
Subject:test
test
```

Et voilà ce que j'ai reçu :

Comme le script fonctionnait, j'ai eu l'autorisation de le mettre sur un serveur. Le script a connu beaucoup de modifications pour être le plus efficace possible. Mais ce qui a posé le plus grand problème, c'était la configuration postfix. En effet, on a eu la mauvaise surprise de voir que les serveurs étaient en liste noire du prestataire de mail car un des scripts présents sur les serveurs créait des milliers de mail. C'est donc ici que

commença un combat acharné pour trouver comment réussir à faire marcher mon script. Et après avoir mis entre autres tous les serveurs dans la white liste et après avoir changé les valeurs dans les fichiers de configuration, on arrivait enfin à voir le bout du tunnel grâce à cette ligne :

```
mail -s "puppeteer" saifeddine.kilani@iptwins.com < /home/saif/rep4
```

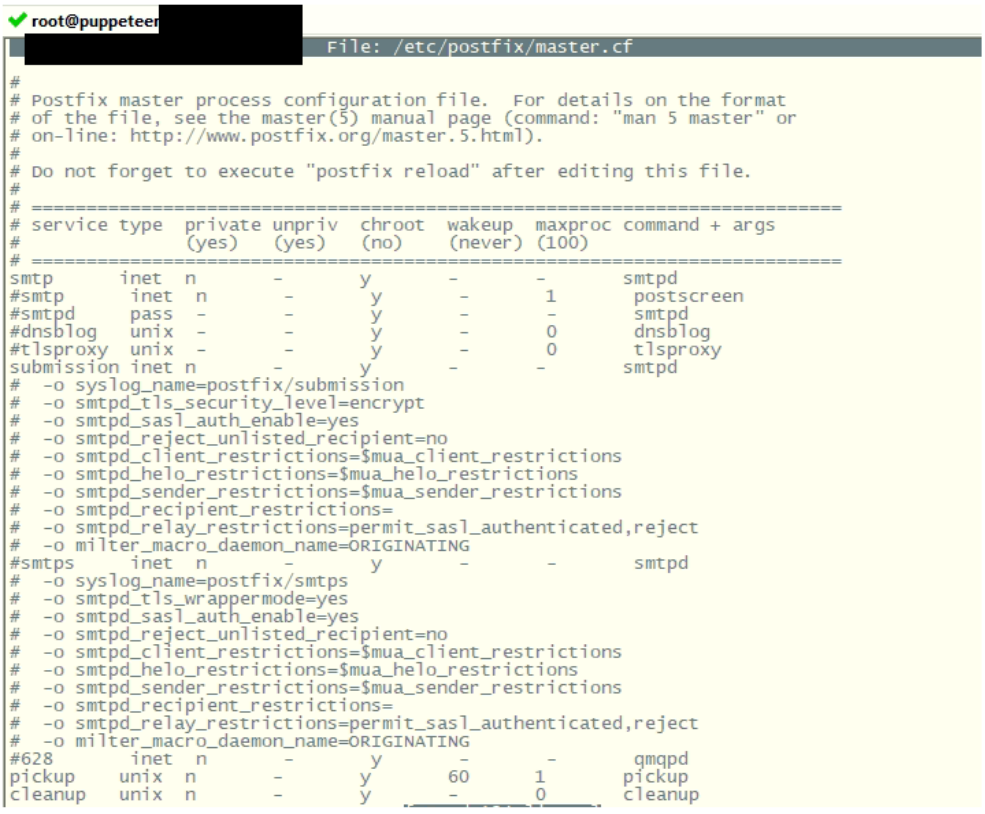
La commande d'envoi de Mail

l'objet du Mail

Le destinataire du mail

le corps du mail (ici c'est un fichier qui va servir de corp)

Et voici les fichiers de configurations :



File: /etc/postfix/main.cf

```
compatibility_level = 2

# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package for
# information on enabling SSL in the smtp client.

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destin$
myhostname = puppeteer.iptwins.guru
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = $myhostname, puppeteer.iptwins.guru, localhost.iptwins.guru, , localhost
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
inet_protocols = all

smtp_generic_maps = hash:/etc/postfix/generic
relayhost = smtp.mandrillapp.com:587
smtp_sasl_auth_enable = yes
smtp_sasl_mechanism_filter = plain, login
smtp_sasl_security_options = noanonymous
smtp_sasl_password_maps = hash:/etc/postfix/sasl/sasl_passwd
smtp_tls_security_level = encrypt
smtp_tls_CAfile = /etc/ssl/certs/ca-certificates.crt
sender_canonical_maps = hash:/etc/postfix/canonical
```

Et voilà à quoi ressemble le script final :

```
#!/bin/sh
apt-get update > rep1
sleep 50
apt-get upgrade -y > rep2
sleep 50
more /var/log/dpkg.log | grep "status installed" > rep3
sleep 50
cat /etc/hostname /home/saif/rep1 /home/saif/rep2 /home/saif/rep3 >> /home/saif/rep4
sleep 50
mail -s "puppeteer" saifeddine.kilani@iptwins.com < /home/saif/rep4
sleep 50
rm /home/saif/rep4
```

Pour expliquer, `apt-get update > rep1` signifie que on va rechercher les mises à jour disponibles et qu'on met la réponse dans un fichier nommé rep1.

`sleep 50` signifie qu'on attend 50 secondes avant de passer à l'etape suivante.

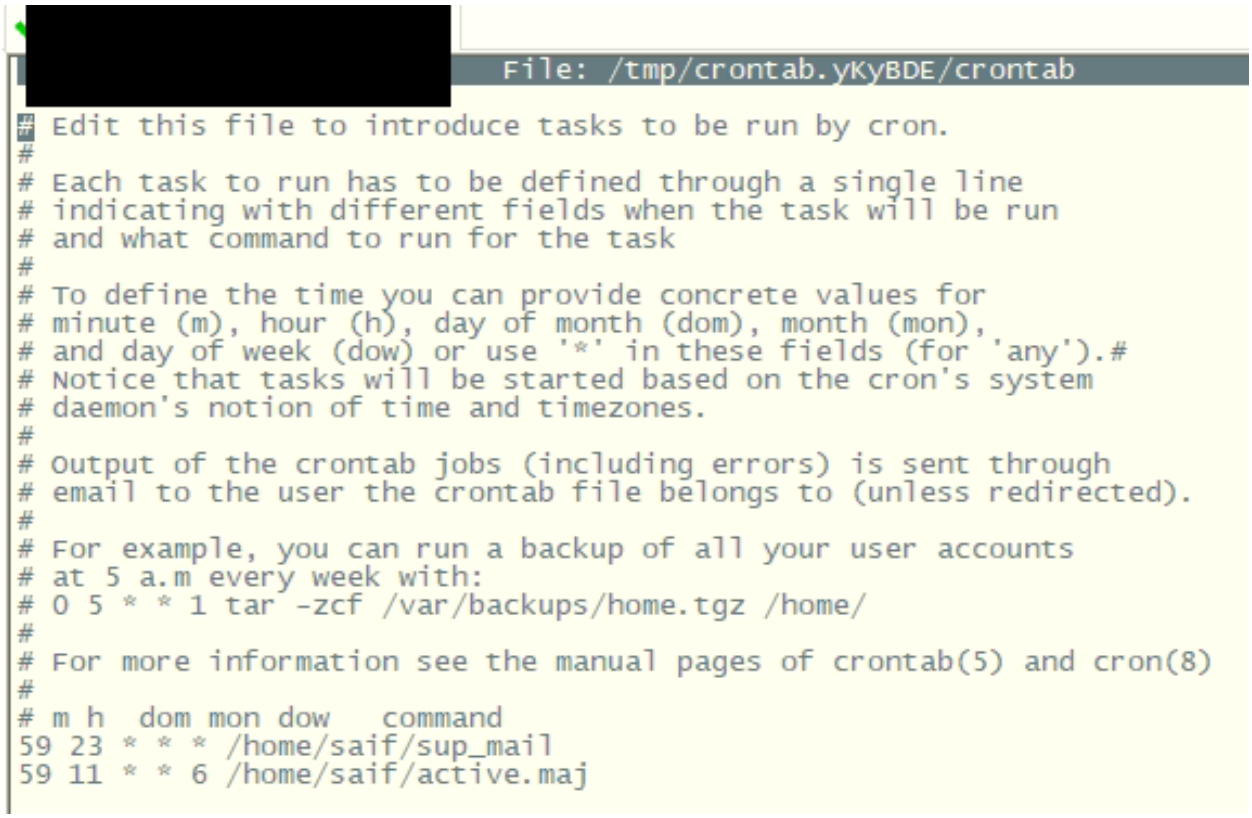
La ligne `apt-get upgrade -y > rep2` installe les mises à jour identifiées avec `apt-get update > rep1` et qu'on met la réponse dans un fichier nommé rep2.

`more /var/log/dpkg.log | grep "status installed" > rep3` Cette ligne va chercher dans les logs les lignes où il y a les termes *status installe* et va tout mettre dans un fichier rep3.

`cat /etc/hostname /home/saif/rep1 /home/saif/rep2 /home/saif/rep3 >> /home/saif/rep4`
Cette ligne très importante prend les fichiers rep1/2 et 3 et les fusionnent pour créer un seul fichier rep4.

Enfin la commande `mail -s "puppeteer" saifeddine.kilani@iptwins.com < /home/saif/rep4` envoie le fichier rep4 à saifeddine.kilani@iptwins.com.

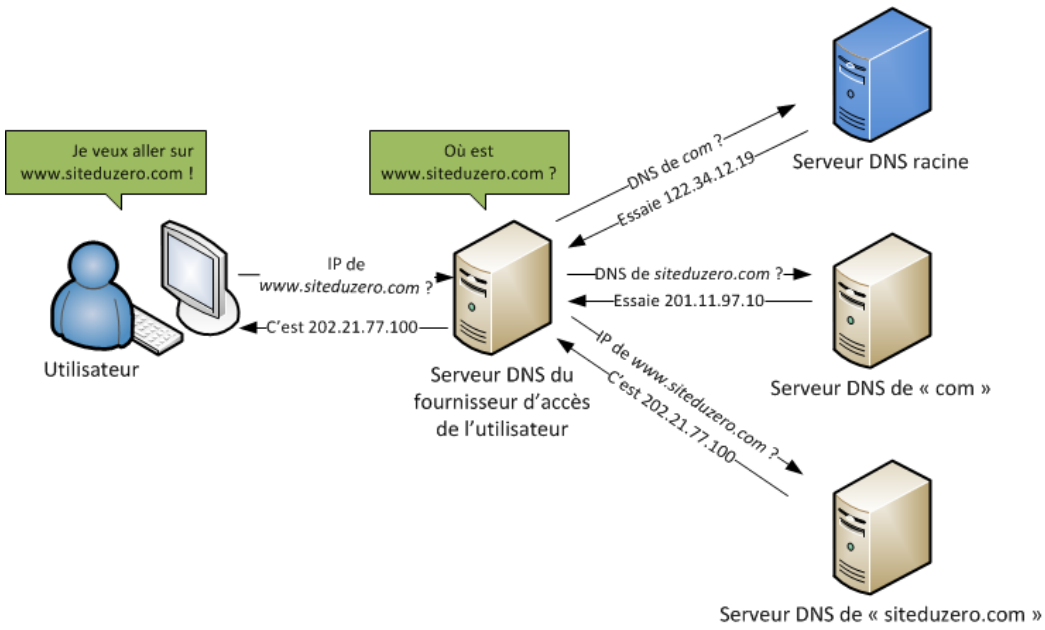
Pour finir voici le fichier *crontab* :



Tout montre que nous avions réussi à créer le script parfait, mais comme bien des personnes le savent, résoudre un problème en apporte bien d'autres. Ici les mails s'envoient mais la machine ne se mettaient pas à jour. En effet, je recevais des mails de paquets mis à jour il y a un mois. Au début, je me disais que juste cette fois les serveurs n'avaient rien à mettre à jour mais j'ai gardé cette réflexion en tête et heureusement car le script ne fonctionnait qu'à moitié. Je me suis donc lancé dans des recherches avec Éric et c'est que récemment que nous avons trouvé. Il fallait juste ajouter cette ligne à la fin de la ligne du crontab : `>/dev/null 2>&1`. Cette ligne permet d'ignorer la réponse du script. Et voila, après tant de temps, on a réussi à mettre en place ce script.

VII/ Etude de cas portant sur la mise en place d'un service DNS

Dans le cadre d'un entrainement en cas de panne majeure, mon tuteur m'a appris comment mettre rapidement en place un service DNS sur Debian9 ou 10 comprenant un serveur principal dit maître et un second dit exclave. Mais avant de rentrer dans les détails techniques, une explication de ce qu'est le DNS s'impose :
Le DNS , Domain Name System, est un service informatique qui traduit les noms de domaine Internet en adresse IP ou autres enregistrements. Pour faire simple, le DNS est l'ombre d'internet permettant ainsi d'aller sur un site sans avoir à apprendre son IP par cœur.



D'autres notions sont utilisés dans le DNS comme les extinctions qui reflètent le type de site internet et son objet (org = organisation, com = commercial, net = network, fr = France, gouv = gouvernement …)

Sur les deux serveurs une fois les deux machines créées on va installer BIND9 :

```
apt-get install bind9 dnsutils
```

Ensuite, toujours sur les deux, nous allons indiquer le chemin de nos différents fichiers de configurations dans /etc/bind/named.conf :

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";
include "/etc/bind/named.conf.log";
include "/etc/bind/tsig.conf";
```

Sur les deux serveurs, ensuite nous allons enchainer avec les configurations de bases de BIND9, dans le fichier /etc/bind/named.conf.options, on y retrouvera les différentes options à adapter selon le besoin :

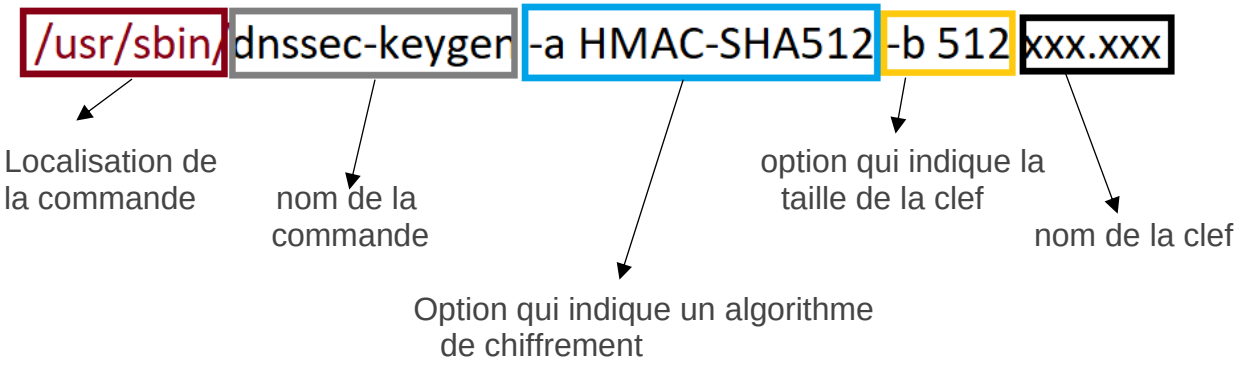
```
        directory "/var/cache/bind";
        dnssec-enable yes;
        dnssec-validation yes;
        dnssec-lookaside auto;
        empty-zones-enable no;
        auth-nxdomain no;    # conform to RFC1035
        listen-on-v6 { any; };
        # adresse ip publique du serveur :
        listen-on {; };
        version "SECRET";
        edns-udp-size 4096;
#        notify-source 188.165.162.39;
#        notify-source-v6 2001:41d0:1:d6e::110;
#        allow-transfer { key key-ns2; key key-ns3; key key-ns4; };
#        allow-notify {;};
        allow-notify {;};
        allow-transfer {"none";};
        recursion no;
        allow-query-cache {none;};
        additional-from-cache no;
        additional-from-auth no ;
        zone-statistics yes;
rate-limit {
responses-per-second 3;
slip 2;
# adresses IP des serveurs DNS authoritaires à exclure des limitations de requêtes :
exempt-clients {;};
};
};
```

Pour finaliser, un aspect fort important et non négligeable, la sécurité. En effet, le flux DNS de ces serveurs de doit pas être traçable et être crypté, pour cela nous allons voir comment sécuriser les flux DNS avec TSIG :

Sur le serveur principal, après avoir créé un répertoire keys et lui avoir donner les droits adéquats :

```
mkdir keys
cd /keys
chown -R bind:bind keys
```

Nous allons créer les clefs avec la commande dnssec-keygen :



Cette commande créera deux fichiers contenant l'un une clef publique et l'autre la clef privée. La clef publique sera donnée aux autres serveurs pour s'identifier et la clef privée restera sur le serveur initial.

Dans le fichier named.conf, ajouter les lignes suivantes :

```
// * TSIG *
key test-test {
    algorithm hmac-sha512;
    secret   "La clef se situe dans le fichier qui fini par .private"
} ;
```

Pour finir sur le maître, ajouter la ligne suivante dans /etc/bind/named.conf.options :

```
allow-transfer {key xxx-xxx.;};
```

Maintenant passons au secondaire, commençons par modifier le fichier named.conf et ajouter :

```
Key dns-dns. {
        algorithm hmac-sha512;
        secret  " La clef se situe dans le fichier qui fini par . private ";
};
Ip primaire {
        keys {
       xxx-xxx.;
};
```

C'est bon, votre service BIND9 est installé avec succès. Je vais maintenant vous expliquer comment l'utiliser. On va ici prendre un cas d'école : on veut ajouter un enregistrement DNS exemple.com

On va commencer par aller dans le fichier /etc/bind/named.conf.local et on va ajouter ces lignes :

```
> nano /etc/bind/named.conf.local

//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

//_____________________________________________
// ZONE DE RESOLUTION DE DOMAINE exemple.com    |
//_____________________________________________|
zone "exemple.com" {
  type master;
  file "/etc/bind/db.exemple.com";
```

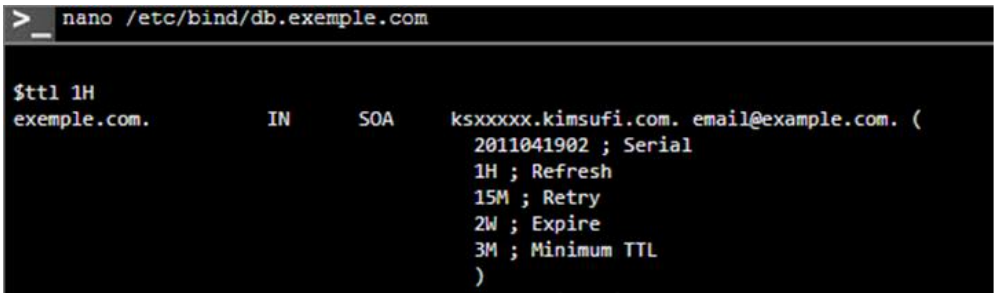
On déclare l'enregistrement
On déclare le type du serveur, maître ou esclave
On indique l'emplacement du fichier où seront les informations

Ensuite nous allons créer ce fichier db.exemple.com à l'emplacement indiqué ci-dessus pour y indiquer des informations telles que les IP :

```
> nano /etc/bind/db.exemple.com

$ttl 1H
exemple.com.          IN      SOA     ksxxxxx.kimsufi.com. email@example.com. (
                                        2011041902 ; Serial
                                        1H ; Refresh
                                        15M ; Retry
                                        2W ; Expire
                                        3M ; Minimum TTL
                                        )
exemple.com.   IN      NS              ksxxxxx.kimsufi.com.
exemple.com.   IN      NS              ns.kimsufi.com.
exemple.com.   IN      A               111.222.111.222
```

Pour analyser ce fichier on va le couper en deux parties



Dans cette partie on va retrouver les informations relatives à l'administration et à la validité des informations
On y retrouve notamment :

exemple.com.		C'est votre nom de domaine, attention : ne pas oublier le point.
IN		Signifie internet, c'est à dire que la zone après le IN Est destinée à internet
SOA		Star Of Authority indiquant le serveur de nom faisant autorité c'est à dire votre DNS principal.
ksxxxx.kimsufi.com.		DNS principal de votre domaine
email@example.com		Adresse email (valide de préférence) il faut remplacer le @ par un point et on termine par un point également.
1H	$TTL	TTL (Time to Live) pour cette zone. Temps pendant lesquels les informations de la zone peuvent être considérées comme valides et être mises en cache
2011041902	Serial	N° de série à incrémenter à chaque modification de ce fichier. Par convention, on écrit : année-mois-jour-numéro_à_2_chiffres.
1H	Refresh	A l'expiration du délai Refresh, le serveur esclave va entrer en communication avec le maitre, s'il ne le trouve pas, il fera une nouvelle tentative au bout du délai Retry, si au bout du délai Expire il considérera que le serveur n'est plus disponible.
15M	Retry	Nombre de secondes avant d'effectuer une nouvelle demande au serveur maître en cas de non-réponse.
2W	Expire	Temps (en secondes) d'expiration du serveur principal en cas de non-réponse.
3M	Minimum TTL	Durée de vie minimum du cache en secondes

Passons maintenant à la deuxième partie :

```
exemple.com.   IN      NS              ksxxxxx.kimsufi.com.
exemple.com.   IN      NS              ns.kimsufi.com.
exemple.com.   IN      A               111.222.111.222
```

Cette partie définit où sera redirigé la requête, ici ce sera vers l'IP 111.222.111.222 mais en réalité, il existe des dizaines de types de types d'enregistrement :

A : il s'agit des enregistrements d'adresses faisant correspondre un nom d'hôte à une adresse IPv4 de 32bits. En IPv6, on utilise des enregistrements AAAA codés sur 128bits.

CNAME : il s'agit d'enregistrements canoniques créant un alias d'un domaine vers un autre. L'alias hérite de tous les sous-domaines de l'original.

MX : définit les serveurs de messagerie pour le domaine.

PTR : associe une adresse IP à un enregistrement de nom de domaine (on parle de reverse puisqu'il s'agit du contraire de l'enregistrement A).

NS : définit les serveurs DNS du domaine (primaire et secondaire).

SOA : fournit les informations générales de la zone : serveur principal, contact, délai d'expiration, n° de série de la zone.

SRV : généralise la notion d'enregistrement MX en proposant des fonctions avancées : taux de répartition de charge (décrit dans la RFC2782).

NAPTR : donne accès aux règles de réécriture de l'information permettant de lier le nom de domaine et une ressource (RFC3403).

TXT : permet à l'administrateur d'insérer un texte quelconque pour un enregistrement DNS.

Conclusion

Ainsi s'achève cette synthèse de mes deux années d'alternance. Vous avez donc eu un aperçu assez détaillé de ce que j'ai jugé intéressant à présenter. Pour moi comme pour mon entreprise, cette première année fut une expérience nouvelle et pour moi une expérience très instructive. En effet, quand j'ai quitté la filière générale pour me consacrer à mon domaine de prédilection. Comme toute nouvelle expérience, celle-ci s'est précédée de moments d'hésitations et de remise en question. Je n'avais jusqu'à l'aucune idée de ce pouvait être le monde du travail. Et à ma grande surprise, c'est une expérience certes unique mais pas désagréable contrairement à ce qu'on pourrait penser. J'ai, dans ma jeunesse, souvent entendue que si on choisissait un travail que l'on affectionne, on ne passerait pas une journée de sa vie à travailler. Aujourd'hui, je peux affirmer que cette phrase est juste. C'est sûr que dans mon cas, j'ai eu de la chance de trouver cette entreprise dont j'avais besoin et où je m'épanouis. C'est donc naturellement que je remercie IPTWINS pour tout ce qu'elle me permet de faire et j'espère que mon travail ravit mes collègues.

C'est avec ces belles paroles que se termine ce rapport de mes 2 années chez IPTWINS. Naturellement, c'est une page qui se tourne et une autre qui s'ouvre.

Sur ce, j'espère que mon rapport vous aura ravi et qu'il a su démontrer au mieux le travail effectué ces années. Ne soyez pas triste de voir ce rapport se terminer ici, dites-vous que le début d'un long voyage répondant au nom de : « la vie ».