Administration Réseau II
Cahier des charges
Rapport technique
Rapport de sécurité
2TL2 - Groupe 3

Louis Arys Geoffrey Brogniet Martin Perdaens
Jean-Michaël Tang

30 mars 2020

Table des matières

1 Cahier des charges

1.1 Idée du projet informatique

Le but de ce projet est de permettre à l'entreprise WoodToys d'avoir son propre réseau, lui permettant de communiquer aussi bien entre leurs propres employés qu'avec leurs clients.

1.2 Contexte du projet

L'entreprise WoodyToys est un fabriquant artisanal de jouets en bois. WoodyToys dispose dans son usine : un atelier de fabrication de jouets, un hangar de stockage où les revendeurs récupèrent leurs produits, le bureau du directeur et les bureaux des comptables, les bureaux des commerciaux et le bureau de la secrétaire.

Internet étant disponible dans l'usine, les employés peuvent accéder au réseau Wifi via leur appareils portables (laptops et smartphones).

L'atelier, le hangar et le bureau sont également connectés à une infrastructure IP, accessible via les postes de travail ou via les téléphones mis à disposition des employés.

L'entreprise nous demande nos services pour concevoir une nouvelle infrastructure d'hébergement des services informatiques, car leurs serveurs, étant dépassé par les nouvelles technologies, ont besoin d'être renouvelés.

1.3 Objectifs du projet

L'objectif de ce projet est de pouvoir mettre en place une infrastructure réseaux qui réponds aux besoins de l'entreprise WoodyToys.

Il nous est donc demandé de concevoir et de configurer différents systèmes de manière autonome, tout en respectant les besoins de WoodyToys.

Pour cela, nous utiliserons des VPS afin de pouvoir construire le réseau de démonstration.

1.4 Contraintes techniques

Les contraintes techniques diffèreront selon les zones dans l'usine, ainsi que dans les réseaux tant en interne qu'en externe avec les clients. Ces contraintes vont être spécifié ci-dessous par catégorie.

1.4.1 Contraintes dans l'atelier de production

Les différents postes (compta, commerciaux, cafétaria, services informatiques et services internes) doivent pouvoir accéder aux services internes et externes. Pour ce faire, nous allons pour cela utiliser une résolution DNS et un service web pour permettre ces accès de manière sécurisé. Le traffic Web devrait être contrôlé par les employés du service informatique pour assurer la sécurité, ainsi que de ne pas laisser les employés aller sur des sites indésirables et/ou tabou par WoodyToys.

Une suggestion serait d'utiliser une gestion des identités des employés dans les services internes.

1.4.2 Contraintes dans les services Web

Chaque employés possède une adresse mail de format nom.prenom@woodytoys.be. Deux adresses mails génériques sont également disponible :

— La secrétaire reçoit les messages reçu de contact@woodytoys.be

— Les commerciaux quand à eux, disposent de l'adresse mail b2b@woodytoys.be

Tous les employés doivent, comme tout bon service web, pouvoir consulter leur couriel et envoyer des mails via un client mail classique aussi bien dans l'entreprise, qu'en déplacement ou à domicile.

1.4.3 Contraintes dans la téléphonie IP

Puisque WoodyToys rénove son réseau, il faudra concevoir un nouveau plan d'adressage dans la téléphonie IP. Pour cela, nous allons développer différents points ci-dessous.

1.4.4 Contraintes de l'accessibilité de VoIP

Le VoIP de WoodyToys doit être accessible depuis Internet pour pouvoir être contacté par les clients. La secrétaire devra gérer ces appels avec l'adresse mail contact@woodytoys.be redirigé vers cette dernière.

1.4.5 Contrainte de la communication entre employés

La communication est un élément clé pour le bon fonctionnement de tout entreprise. Pour cela, il va donc falloir permettre aux employés de pouvoir communiquer aussi bien à l'intérieur de l'entreprise que quand ils sont chez eux ou en déplacement, particulièrement pour les commerciaux qui se déplacent bien plus souvent.

Pour clarifier les différentes communications :

— Les ouvriers peuvent joindre les autres départements internes par le biais d'un poste de téléphonie IP dans leur atelier et dans le hangar.

— La secrétaire dispose quand à elle d'un PC où se trouve un softphone qui lui permet de contacter tout le monde.

— Le service comptable dispose d'un numéro unique. Ce numéro leur permet de joindre le premier comptable disponible, et, puisque le service comptable dispose de deux bureaux, a également un numéro spécifique par bureau. Les comptables peuvent communiquer avec n'importe qui, que ce soit à au sein de l'entreprise qu'à l'extérieur, à une exception près : ils ne peuvent pas joindre directement le directeur.

— Les commerciaux peuvent joindre tout le monde de la même manière que le service comptable, mais ne disposent que d'un seul bureau et ont des smartphones leur permettant de téléphoner tout en se déplaçant.

— La direction peuvent joindre n'importe quels postes internes ainsi que l'extérieur. En revanche, ils ne sont joignables que si la secrétaire leur permet de joindre la direction.

1.4.6 Contraintes avec la fusion des réseaux de WoodyToys

Étant donné que WoodyToys a racheté une entreprise concurrente, il faudra fusionner les deux réseaux téléphoniques. Pour cela, une boîte vocale est disponible pour les employés. Concernant le réseaux téléphoniques en lui-même, nous allons minimiser les

changements au niveau du plan d'adressage et configurer les deux serveurs de téléphonie de sorte que le nouveau plan d'adressage interne soit accessible.

1.5 Description des besoins fonctionnels

Dans cette section du rapport, nous allons développer une description des besoins, en terme informatique de la demande de WoodyToys. *Nous allons également citer les raisons pour lesquelles nous faisons certains choix plutôt que d'autres.*

1.5.1 DNS

Nous aurons besoin d'un serveur dédié, permettant de faire le lien entre les noms des différents services mis à disposition sur le réseau, et l'adresse IP de ceux-ci.

Pour cela, nous utiliserons Bind9 plutôt qu'un autre service DNS, pour la simple raison que les documentations sont plus complètes et facilement accessible, et que, en cas de problème, des solutions sont plus rapidement trouvées. Il existe évidemment d'autres services DNS comme "smart DNS PROXY" ou "PremiumDNS" qui offrent la même chose, mais ceux-ci sont payants (ce qui fera des frais en plus pour WoodyToys) et moins bien documentés ou difficilement accessible, ce qui pourrait causer des problèmes à long terme si des bugs apparaissent et ne sont pas réglé rapidement.
Pour plus d'informations, veuillez vous référer au rapport technique.

1.5.2 Serveurs Web

Nous aurons besoin de trois serveurs web différents. Un premier pour pouvoir présenter les différents produits de l'entreprise publiquement. Un second pour pouvoir effectuer la vente en ligne des produits, et un troisième pour héberger l'outil ERP Web. De plus, cela demandera une base de données pour gérer toutes les données relatives aux différents clients et revendeurs, les données personnelles des employés, ainsi que les addresses de ceux-ci. Pour les sites, il faudra penser à faire une architecture `Back-end`/`Front-end` de manière à avoir une structure dynamique et permettre une liaison entre la base de données et les serveurs.

Bien sûr, il est possible de diminuer le nombre de serveurs utilisés, mais cela reviendrait à surcharger les serveurs, ce qui n'est pas une bonne idée. Augmenter le nombre de serveurs est également possible, mais cela revient à utiliser beaucoup de ressources pour pas grand chose.

1.5.3 VoIP

Actuellement nous n'avons pas encore eu les informations nécessaires pour déterminer les besoins de ce service.

1.5.4 Réseau interne

Nous aurons besoin de scinder les différentes parties de l'entreprise dans des VLANs différents, de manière à donner des permissions d'accès différentes pour chacunes d'entre-elles.

1.5.5 Mail interne

Actuellement nous n'avons pas encore eu les informations nécessaires pour déterminer les besoins de ce service. Nous aurons besoin au minimum d'un serveur SMTP et POP/IMAP pour pouvoir gérer les envois et les réceptions d'emails.

1.6 Solutions et discussion

Concernant les solutions utilisées, nous développerons les détails dans le rapport technique. La sécurité étant également importante, un autre rapport sera également rédigé dans le rapport de sécurité.

2 Rapport de sécurité

2.1 Introduction

Ce rapport est destiné à lister les risques ainsi que les solutions apportées à ces risques de l'entreprise WoodyToys tant au niveau infrastructure de celle-ci mais aussi du prototype que nous fournissons.

2.2 Risques de l'infrastructure réseau

Les risque les plus important pour WoodyToys sont la confidentialité et l'intégrité au sein de l'entreprise mais aussi à l'extérieur. Les exemples les plus clairs sont les interceptions des paquets, c'est-à-dire l'interception de message unique non identifié à cause d'une attaque de type "Man in the middle" par exemple. L'interception des paquets peut aussi mener à du "Fishing", autrement dit, le fait de se faire passer pour le site (ici WoodyToys) et de rediriger vers un site de Fishing qui pourra soit voler vos mots de passe soit agir de manière malveillante envers un client.

Pour éviter ces risques, nous pouvons envisager de sécuriser le site et d'utiliser un https qui, dans tout les cas, sera de rigueur.

Un des risques aussi de WoodyToys peut être la disponibilité du site mais aussi de ses infrastructures interne. Le site de l'entreprise doit être tout le temps opérationnel et pour cela plusieurs sécurité peuvent être mises comme par exemple ne pas tout mettre au "même endroit" ou encore bien choisir ses serveurs grâce au VPS

2.3 Risques du prototype

Le prototype peut conduire à plusieurs gros risques tant au niveau du VPS qu'au niveau des Dockers utilisés et empruntés sur le WEB.

Les risques du VPS sont les suivants :

- Les performances d'un VPS n'atteignent pas celles d'un véritable serveur. Généralement, les fournisseurs des VPS limitent les performances des serveurs afin de maximiser le nombre d'unité qu'un serveur physique peut contenir. Ce qui rejoint le risque de la disponibilité expliqué ci-dessus.

- Les VPS sont généralement des virtualisation du serveur, ce qui permet de créer un isolation entre chaque serveur. Mais le risque zéro n'existe pas et il est conseillé d'éviter les environnements partagés.

Les risques au niveau du Docker sont simplement l'exécution de code, la confiance des Docker mis à disposition ainsi que le déni de service. Afin de contrer ces risques, il faut prévenir tout cela en utilisant des sources de confiance ou alors utiliser des outils qui pourront analyser notre Docker.

Et enfin nous pourrons aussi privilégier l'utilisation d'un Fail2ban qui va analyser le nombre de tentative d'accès et bloquer l'utilisateur après 3 tentatives par exemple grâce à son adresse IP. Cela veut dire qu'avec l'adresse utilisée pour se connecter, l'utilisateur ne pourra plus essayer de se connecter pendant un moment prédéfinis.

3 Rapport Technique

Responsable de mission et bilan (13 mars 2020)

Louis Arys

Bilan : Pour le moment, tout se déroule plutôt bien, nous n'avons pas rencontré de soucis lors de la prise de contact (c'est la première fois que nous travaillons tous ensemble). Au niveau de l'implémentation du projet, nous arrivons à bien nous répartir le travail, et en cas de soucis, on peut compter sur les autres pour nous donner un coup de main. Nous nous complétons bien au niveau de nos forces et nos faiblesses. Donc pour moi le bilan est très positif !

3.1 Introduction

Vous trouverez ci-après les différentes spécifications techniques des différentes technologies que nous utiliserons dans ce projet. Elles sont suceptibles d'évoluer ou d'être étoffées selon l'évolution de celui-ci.

3.2 Méthodologie

Nous utilisons plusieurs outils :

— `Trello` : permet de nous répartir le travail, de mettre les ressources importantes et les liens de manière claire dans un tableau de ressources, ainsi qu'une version synthétisée des consignes

— `Github` : permet de centraliser le code, d'éviter les conflits lorsque nous travaillons ensemble sur le code. Nous l'utilisons aussi pour faire le wiki de notre groupe.

— `Overleaf` : pour éditer des rapports à plusieurs en LaTeX

3.3 DNS

Nous avons décidé d'utiliser un container docker dans lequel tournera un serveur DNS Bind9. La raison de ce choix est la suivante : Bind9 est le serveur DNS le plus répandu sous Linux. Il y aura donc moyen d'accéder facilement à la documentation, et en cas de soucis, nous pourrons plus facilement trouver des solutions à nos problèmes. De plus, le fonctionnement de ce serveur est relativement simple à prendre en main.

Niveau d'avancement : A l'heure actuelle, nous avons implémenté un serveur Bind9 très basique et nous l'avons testé sur un réseau docker local avec 2 autres containers Ubuntu.

Le DNS a été configuré pour lier les hôtes et un script de déployement automatique a été mise en place pour faciliter l'utilisateur au niveau des manipulations de commandes.

3.4 Database

Nous avons décidé d'utiliser une database de type mysql car nous avons tous un minimum de connaissance dans les bases de données relationnelles. De plus, une base de données relationnelle correspondra exactement aux besoins d'une entreprise comme WoodyToys, nécessitant de pouvoir gérer des clients, des stocks, ...

Pour le choix de la base de données en elle-même, nous avons arrêté notre choix sur **MariaDB**. Ce choix découle de deux choses : d'une, mysql est une base de données facile à prendre en main et intuitive. De deux, MariaDB est un fork libre de mysql, et est actuellement quasiment pareil à celui-ci. Cela nous permet d'utiliser des outils *Open-Source*, ce qui nous tient à coeur.

Niveau d'avancement : Nous n'avons pas encore implémenté la database.

3.5 Websites

Nous avons décidé de partir sur une architecture très classique. Pour le **Front-end**, nous sommes partis sur du pur *HTML/JS/CSS*. Cela permettra de très rapidement obtenir quelques pages afin que le client puisse utiliser les fonctionnalités du Back-End.

Pour le **Back-end**, nous nous sommes décidés d'implémenter un serveur Nodejs, pouvant être agrémenté si besoin est d'*EJS* pour faire du templating. Nous avons fait ce choix car nous avions de l'expérience dans l'implémentation de ce type de serveur. NodeJS est également efficace pour le routing et le déployement d'API.

Niveau d'avancement : Nous avons actuellement implémenté un serveur très basique renvoyant un hello-world au navigateur. Ce serveur a, bien entendu, été containerisé et testé.

3.6 Host

Nous avons choisi d'implémenter un container très basique, contenant un noyau Ubuntu, et permettant, actuellement, d'effectuer des ping sur d'autres machines, afin de pouvoir tester le serveur DNS. Ce container sera étendu au besoin pour nous permettre de tester nos différents services. À terme, ce container servira aussi à émuler les machines se trouvant chez WoodyToys.

Niveau d'avancement : terminé pour les besoins actuels du projet.

3.7 VoIP

Au terme de cette semaine de début de projet, nous ne pouvons nous exprimer sur le VoIP pour la raison simple que le cours abordant le sujet n'a pas encore eu lieu. Nous pouvons juste dire que nous le placerons sur un VPS à part car nous voulons séparer les différents services les uns des autres.

3.8 Mails

Au terme de cette première semaine e projet, nous ne pouvons pas encore nous exprimer sur notre architecture Mail de manière concrète. La seule chose que nous pouvons dire pour le moment c'est que notre serveur mail se trouvera dans la DMZ

3.9 Firewall

Au terme de cette semaine, nous ne pouvons pas encore nous prononcer sur le firewall vu que la sécurité n'est pas encore de mise.

3.10 Problèmes rencontrés

3.10.1 Pour le DNS

Il a fallu comprendre comment marchait Bind9 ainsi que la manière de le containeriser proprement. Mais pour le moment tous les problèmes rencontrés ont été résolus.

3.10.2 Pour le serveur NodeJS

Pas de problèmes rencontrés jusqu'à maintenant

3.10.3 Pour les outils d'organisation

Il a fallu un petit temps pour les prendre en mains pour ceux qui ne les avaient pas encore utilisés. Cela a été vrai surtout pour Overleaf(LaTeX) qui demande un petit temps pour le prendre en main.

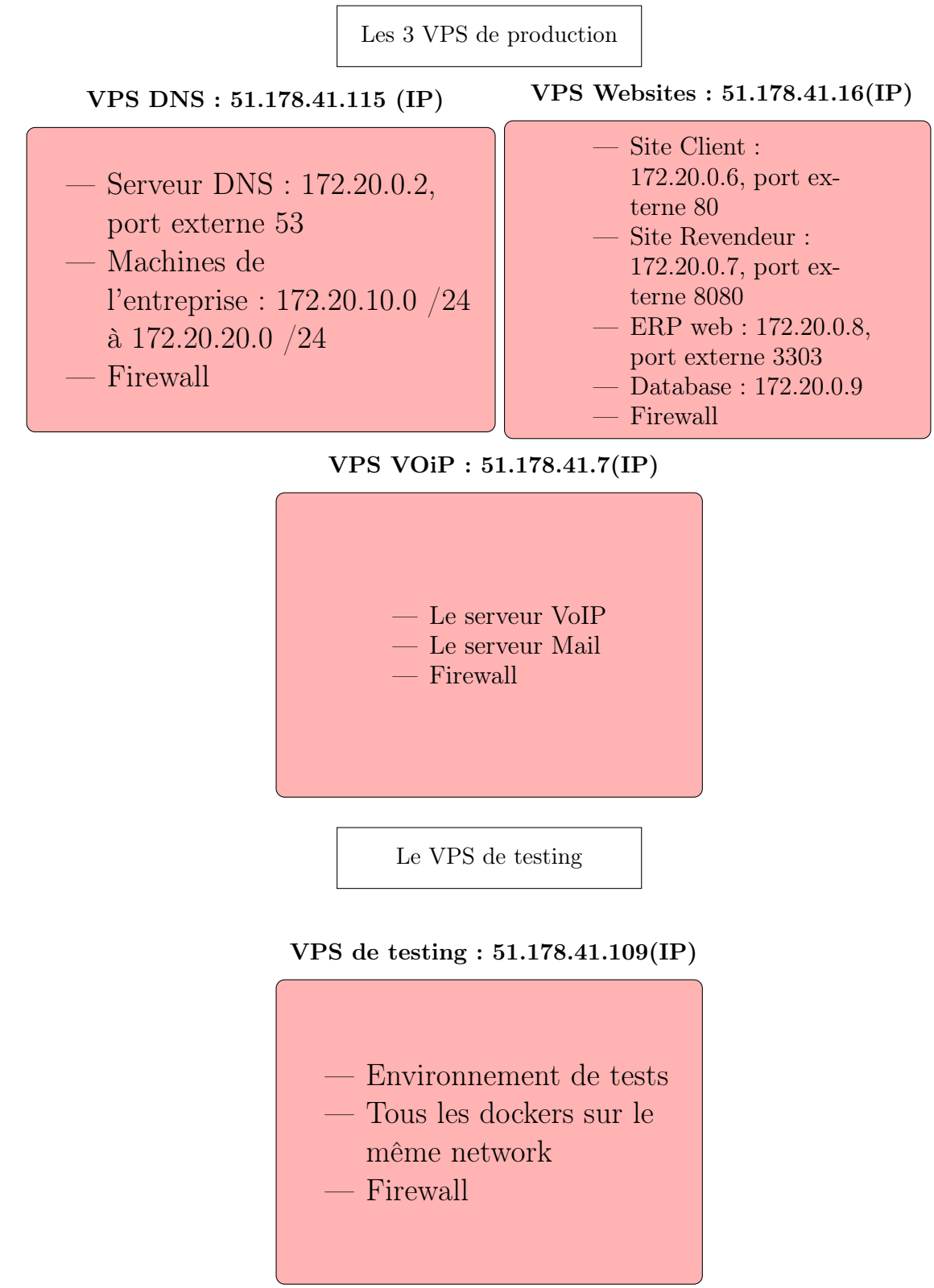
4 Changements effectués

Dans le cahier des charges, nous avons rajouté des informations dans la section DNS, dans la section Serveur Web et dans la section Solution et discussion. En ce qui concerne les commentaires reçu du groupe 2TL2-2, nous avons jugé qu'il n'était pas nécessaire de mettre des informations technique, et qu'il suffisait de se référer au rapport technique.

Dans le rapport technique, nous avons rajouté des détails de l'avancement dans le DNS et refait le schéma de WoodyToys comme demandé ainqi que la tabl d'adressage qu'il manquait dans la première version du rapport technique. Une table d'adressage a été ajouté à la fin, après les deux schémas.

Dans le rapport de sécurité, aucun changement n'a été effectué.

Schéma du prototype

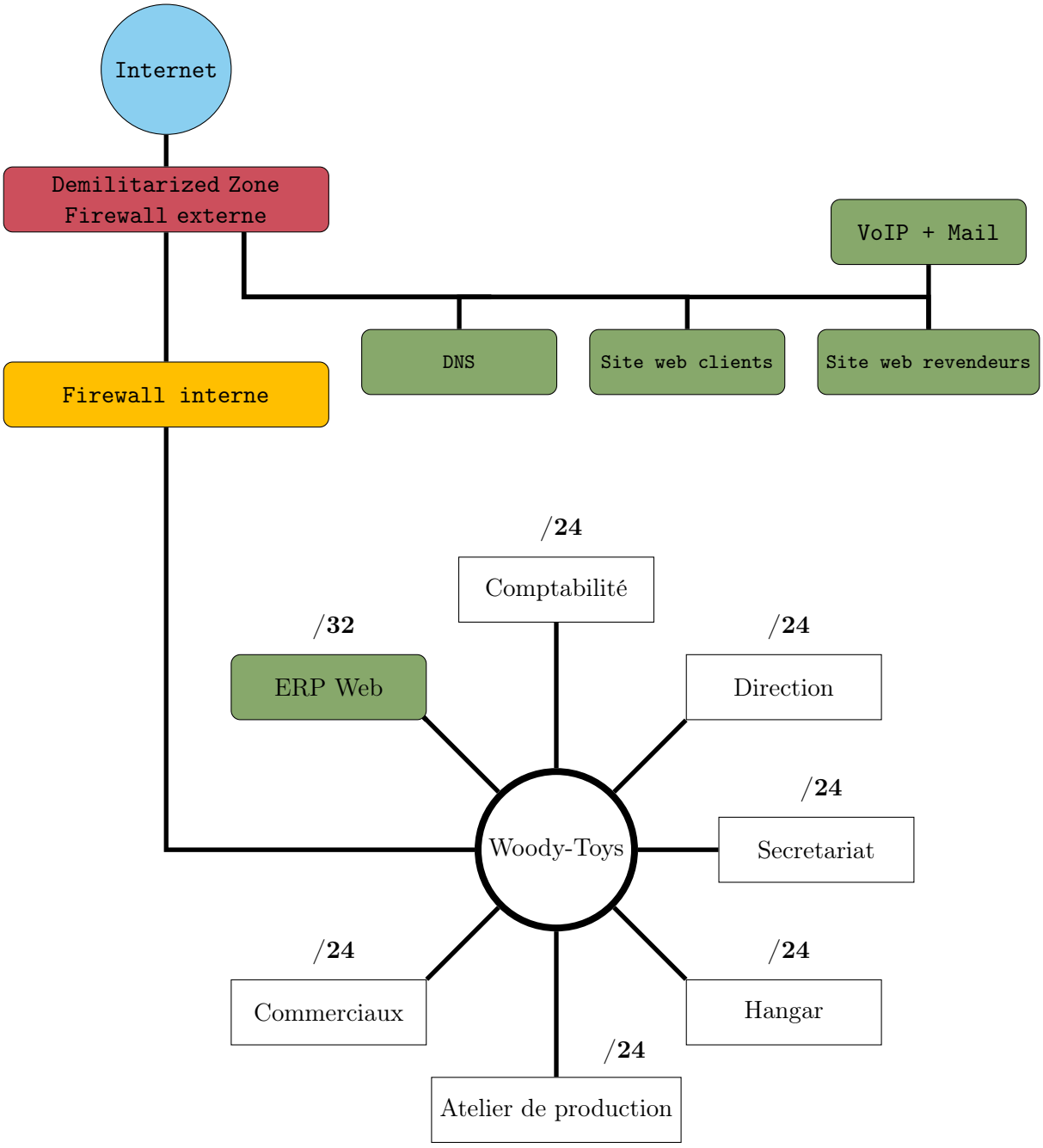


Le schéma ci-dessus représente la manière dont seront répartis les différents services sur nos 4 VPS. Notre choix a été de séparer au maximum les différents services que

nous devons implémenter : le DNS dans le premier, les sites webs et la database dans le deuxième, et la VoIP dans le troisième. Cela a pour but de simplifier notre architecture au maximum, et d'avoir un schéma simple à comprendre. De plus, cela permettra d'éviter la perte de plusieurs services d'un coup en cas d'attaques.

Le quatrième VPS vient du fait que nous soyons un groupe de 4, ce qui nous donnes l'avantage de pouvoir l'utiliser comme environnement de test, pour pouvoir déployer plus facilement les parties nouvellement implémentées sur les VPS de production.

Schéma de Woody-Toys



Notre architecture se base sur une architecture incluant deux firewalls et une DMZ(Demilitarised Zone). Le but de cette architecture est de séparer les ressources accessible de l'extérieur de celles ne devant être seulement disponible en interne. Cela permet de mettre deux couches de sécurité à notre architecture : un premier firewall permettant de se protéger des attaques de l'extérieur, mais laissant passer toutes les demandes/requêtes venant de l'internet. Un second, effectuant un filtrage stricte des requêtes allant vers l'intérieur, et empêchant ainsi n'importe qui de rentrer dans le

réseau interne de l'entreprise. C'est aussi pourquoi nous avons mis le serveur `ERP Web` à l'intérieur, car il n'est pas nécessaire qu'il soit accessible de l'extérieur.

Plan d'adressage

Services	**Adresse réseaux**	**Masque**
Atelier/hangar	*172.16.1.0 /24*	*255.255.255.0*
Commerciaux	*172.16.2.0 /24*	*255.255.255.0*
Direction	*172.16.3.0 /24*	*255.255.255.0*
Compta	*172.16.4.0 /24*	*255.255.255.0*
DMZ	*172.16.5.0 /24*	*255.255.255.0*
Service Interne	*172.16.6.0 /24*	*255.255.255.0*

TABLE 1 – **Table d'adressage**