

MÉMOIRE DE MASTER

Présenté en vue de l'obtention de la
MASTÈRE PROFESSIONNEL Co-CONSTRUISTE EN INFOTRONIQUE

Spécialité : Infotronique

Déploiement de l'infrastructure IT et SI d'une entreprise et intégration des objets connectés au réseau

Par MOHAMED AZIZ KAROUI

Réalisé au sein de Zeta Engineering



Président : AKKARI Emna, Enseignante, ISTIC
Rapporteur : OUESLATI Nejla, Enseignante, ISTIC
Encadrant académique : REKIK Sana, Enseignante, ISTIC
Encadrant professionnel : CHHIMI Achref, Technicien, ZETA Engineering

Période De Stage : 01/04/2023 - 30/09/2023

Année Universitaire : 2022 - 2023

Dédicaces

Je dédie ce travail à :

À mes chers parents Tarek et Farida, qui ont soutenu mon parcours académique de bout en bout et qui ont consenti tant de sacrifices pour moi,

À tous les membres de ma famille,

À mes amis et collègues qui ont été présents tout au long de mes études,

À Madame Sana Rekik et Monsieur Achref Chhimi pour leur encadrement et leur dévouement à me guider.

Finalement, je tiens à remercier tous ceux qui m'ont soutenu.

Remerciement

Nous tenons à exprimer notre gratitude envers toutes les personnes qui nous ont soutenus tout au long de notre projet de fin d'études.

Nous voudrions tout d'abord remercier nos directeurs de projet, Mohamed Ali Adala et Nadair Adala, pour leur guidance et leur soutien tout au long de ce projet. Leurs commentaires et leur expertise ont été inestimables pour le succès de ce projet.

Nous voudrions également remercier nos amis et notre famille pour leur soutien constant tout au long de nos études et pour leur encouragement pendant ce projet de Mémoire de Mastère. Leurs encouragements et leur soutien ont été essentiels pour que nous puissions poursuivre nos études jusqu'à ce jour.

Nous voudrions également remercier nos collègues et camarades de classe pour leur collaboration, leur aide et le partage de leurs connaissances. Leurs contributions ont été précieuses pour la réalisation de ce projet.

Nous tenons également à exprimer notre gratitude envers les membres du jury qui ont pris le temps de lire et d'évaluer ce rapport de projet de Mémoire De Mastère. Leurs commentaires et leurs retours ont été extrêmement précieux pour nous.

Enfin, nous voudrions remercier Zeta Engineering pour leur contribution financière qui nous a permis de réaliser ce projet. Leur soutien financier a été crucial pour l'acquisition des ressources nécessaires à la réalisation de ce projet.

Table des matières

Introduction générale	1
1 Contexte du projet	2
1.1 Introduction	2
1.2 Présentation de l'organisme d'accueil	2
1.2.1 Organisme d'accueil	3
1.2.2 Domaines d'activités	3
1.3 Présentation du projet	3
1.3.1 Cadre de projet	3
1.3.2 Problématique	4
1.4 Méthodologie de travail et planification	5
1.4.1 Présentation de la méthodologie en cascade	5
1.4.2 Choix de la méthodologie en cascade	6
1.5 Outils et technologies utilisés	7
1.5.1 Matériels utilisés	7
1.5.2 Logiciels utilisés	8
1.6 Conclusion	10

2 Déploiement de l'infrastructure IT	11
2.1 Introduction	11
2.2 Présentation des locaux et déploiement de réseaux MAN .	11
2.2.1 Présentation des trois locaux	12
2.2.2 Définition du réseau MAN	16
2.3 L'architecture réseau	17
2.3.1 Topologie en GNS3	17
2.3.2 Configuration en GNS3	20
2.4 Implémentation	28
2.4.1 Implémentation des LANs	28
2.4.2 Implémentation du MAN et le choix de TP-Link .	29
2.4.3 Pare-feu	34
2.5 Conclusion	39
3 Déploiement du Système d'information	40
3.1 Introduction	40
3.2 Les besoins de notre système d'information	40
3.3 Les solutions choisies	41
3.3.1 OS Serveur	41
3.3.2 Virtualisation	42
3.3.3 Protocole de communication	43
3.3.4 ERP/CRM	44

3.3.5	Système de téléphonie IP	44
3.3.6	Gestion de parc et d'inventaires	45
3.3.7	Gestion des identités et contrôle d'accès	46
3.4	Déploiement des solutions choisies	47
3.4.1	La couche physique	47
3.4.2	La couche virtualisation	50
3.4.3	La couche applicative	53
3.5	Conclusion	60
4	Intégration des objets connectés	61
4.1	Introduction	61
4.2	Définition des objets connectés	61
4.3	Intégration des dispositifs de l'entreprise	63
4.3.1	Enjeux de l'intégration	63
4.3.2	Intégration des caméras de surveillance	63
4.3.3	Intégration des systèmes de pointage	64
4.3.4	Gestion intelligente de l'éclairage	65
4.3.5	Gestion intelligente du capteur de température et humidité	66
4.3.6	Solution proposée	66
4.4	Solution avec Raspberry Pi et Node-RED	66
4.4.1	Matériels utilisés pour notre prototype	67
4.4.2	Choix de Raspberry Pi et Node-RED	67

4.4.3	Installation Raspberry Pi OS et Node-RED	68
4.4.4	Configuration de Node-RED	71
4.4.5	Sécurité de Node-RED	73
4.4.6	Réalisation de projet avec Node-RED	75
4.4.7	Intégration dans le parc informatique	84
4.5	Conclusion	86
Conclusion Générale		87
Bibliographie		89
Acronyme		91

Table des figures

1.1	Méthodologie en Cascade [1]	6
2.1	Distance entre le bureau Rades et Rades Meliane (1,268 km)	13
2.2	Distance entre bureau Rades et bureau Ezzhara (2,843 km)	14
2.3	Distance totale entre les bureaux	15
2.4	Metropolitan Area Network	16
2.5	Topologie du réseau du bureau de Rades	18
2.6	Topologie du réseau du bureau de Rades Melian	19
2.7	Topologie du réseau du bureau Ezzahara	20
2.8	Console du routeur de Rades	21
2.9	Réglage de l'IP adresse	22
2.10	Test de connexion de routeur vers l'internet	23
2.11	Configurer le port f0/1	24
2.12	Ping de utilisateur4 vers l'internet	25
2.13	Configuration de routeur	26
2.14	Test de connexion des utilisateurs	27
2.15	Routeur Cisco 891F	28

2.16	Interconnexion des commutateurs pour les bureaux Rades et Ezzahara	29
2.17	Déballage du TP-Link CPE610	30
2.18	Assemblage des pièces du TP-Link CPE610	30
2.19	Initialisation du TP-Link	31
2.20	Initialisation du TP-Link en mode test	31
2.21	Panneau de contrôle du TP-Link	32
2.22	Initialisation de l'émetteur	32
2.23	Initialisation du récepteur	33
2.24	Test de connexion entre le point d'accès TP-Link et le client TP-Link	33
2.25	Installation du TP-Link CPE610 à Rades Melian	34
2.26	Installation de pfSense - Étape 1	35
2.27	Installation de pfSense - Étape 2	36
2.28	Installation de pfSense - Étape 3	36
2.29	Installation de pfSense - Étape 4	37
2.30	Installation de pfSense - Étape 5	37
2.31	Installation de pfSense - Étape 6	38
2.32	Tableau de bord de pfSense	38
3.1	Diagramme du système d'information	47
3.2	Installation de Ubuntu Server 22.04	48
3.3	Interface Putty pour la connexion à Ubuntu Server 22.04 .	49
3.4	Résultat de la commande "neofetch" sur Ubuntu Server 22.04	50

3.5	Installation des packages KVM	51
3.6	Vérification de l'état du service libvirtd	52
3.7	Accès à Virt-Manager via Xming et PuTTY	52
3.8	Mise à jour du système Linux	53
3.9	Redémarrage du service isc-dhcp	54
3.10	Configuration du fichier <code>dhcpd.conf</code>	54
3.11	Exemple de fichier de configuration <code>dhcpd.conf</code>	54
3.12	État du service ISC DHCP	55
3.13	Installation d'Asterisk sur Ubuntu	55
3.14	Configuration d'Odoo et version PostgreSQL	56
3.15	Base de données PostgreSQL dans le tableau de bord Odoo	56
3.16	Tableau de bord d'Odoo avec les modules CRM installés .	57
3.17	Tableau de bord de GLPI	57
3.18	Affichage des tables dans MariaDB	58
3.19	Tableau de bord de phpLDAPAdmin	59
3.20	Liste des machines virtuelles déployées	59
4.1	Architecture IoT basée sur CoAP	62
4.2	Logiciel SmartPSS installé	64
4.3	Machine de pointage ZKTeco et logiciel ZKTime	65
4.4	Installation du Node-RED avec le Script Officiel Bash . . .	69
4.5	En cours d'Installation du Node-RED	70
4.6	Le démarrage du Node-RED	71

4.7	Interface Node-RED	72
4.8	Installation du dashboard Node-RED	72
4.9	Installation du nœud DHT11 dans Node-RED	73
4.10	Configuration de la sécurité dans Node-RED en Settings.js	74
4.11	Création d'un code de hachage	74
4.12	Mise à jour du code de hachage dans Settings.js	75
4.13	Connexion avec le nouveau mot de passe	75
4.14	Schéma de notre réalisation	76
4.15	Ajout d'un Inject Node pour répéter l'action chaque 10 secondes	77
4.16	Configurer un nœud Fonction pour séparer l'humidité de la température	78
4.17	Le résultat du nœud Debug	79
4.18	Nœud Switch	80
4.19	Nœud Raspberry GPIO Output	81
4.20	Le projet en Node-RED	82
4.21	Circuit de mon prototype miniature	82
4.22	Le login du tableau de bord en mobile	83
4.23	Le tableau de bord en bureau	84
4.24	Installation de FusionInventory-Agent	85
4.25	Configuration de FusionInventory-Agent	85
4.26	Tableau de bord de GLPI avec le Raspberry Pi intégré . .	86

Liste des tableaux

1.1	Les nouveaux locaux de Zeta Engineering en Tunisie	4
1.2	Matériels utilisés	8
1.3	Logiciels utilisés	9
1.4	Logiciels utilisés - Suite	10
3.1	Les éléments composants le système d'information	41
4.1	Les équipements composants de notre prototype miniature	67

Introduction générale

L'intégration des objets connectés dans les systèmes d'information des entreprises est devenue une nécessité pour répondre aux exigences accrues du marché et à l'évolution incessante de la technologie [2]. Dans ce contexte, ce rapport de Mémoire de Mastère porte sur le déploiement de l'infrastructure IT, le développement d'un système d'information et l'intégration des objets connectés au système d'information de l'entreprise Zeta Engineering.

Ce rapport est structuré en quatre chapitres pour expliquer notre approche et notre travail :

- Le premier chapitre présente le projet, son contexte, les outils utilisées, ainsi que l'organisme d'accueil. Nous y exposons les activités de l'entreprise, ses objectifs et ses besoins en matière d'infrastructure.
- Le deuxième chapitre décrit le déploiement de l'infrastructure IT. Nous détaillons les défis rencontré et les solutions proposées.
- Le troisième chapitre explique le déploiement du système d'information. Nous présentons les choix technologiques que nous avons effectués, les différentes étapes de la mise en place du système.
- Le dernier chapitre se concentre sur l'intégration des objets connectés. Nous expliquons comment nous avons relié les objets non connectés au système d'information pour collecter et analyser les données. De plus, nous discutons des défis rencontrés et des solutions mises en place pour les surmonter.

En conclusion de cette introduction générale, nous avons souligné l'importance croissante de l'intégration des objets connectés dans les systèmes d'information des entreprises. Ce rapport mettra en lumière notre expérience avec l'entreprise Zeta Engineering en détaillant les étapes clés de notre projet.

Chapitre 1

Contexte du projet

1.1 Introduction

Le premier chapitre de ce rapport présente l'organisme d'accueil ainsi que notre projet.

Nous commençons par présenter l'entreprise, ses activités, et ses objectifs en matière de technologies connectées.

Ensuite, nous expliquons le contexte dans lequel s'inscrit ce projet, en évoquant les enjeux et les problématiques auxquels il répond.

Par la suite, nous exposons la méthodologie de travail que nous avons choisie et expliquons les raisons de ce choix.

Nous détaillons également les différents outils et technologies que nous utilisons au sein de notre projet.

Enfin, nous expliquons les choix techniques que nous avons effectués pour répondre aux exigences de l'entreprise et assurer une infrastructure robuste et sécurisée.

1.2 Présentation de l'organisme d'accueil

Dans cette section, nous abordons la présentation de l'organisme d'accueil Zeta Engineering ainsi que son domaine d'activité.

1.2.1 Organisme d'accueil

Notre stage de projet de Mémoire de Mastère a été réalisé chez Zeta Engineering, un bureau d'étude multifonction appartenant au groupe Agexis [3], qui intervient sur la zone européenne et africaine.

Le groupe Agexis est un bureau d'étude technique pluridisciplinaire et maître d'œuvre du bâtiment et des infrastructures. Il est implanté en Île-de-France depuis 2015.

Zeta Engineering est spécialisée dans le consulting et l'ingénierie, et se consacre principalement aux conseils en communication, marketing, IT et gestion pour ses partenaires, en Tunisie comme en France. Elle est basée en Tunisie depuis 2018.

1.2.2 Domaines d'activités

Zeta Engineering, en collaboration avec Agexis et les autres membres du groupe, est chargée de mettre en place leur stratégie marketing pour la période 2020-2030, en déployant leur communication et leur commercialisation.

Cette entreprise de bureau d'études multifonctions est située en Tunisie et travaille en étroite collaboration avec Agexis. Les services proposés par Zeta Engineering sont très diversifiés et couvrent différents secteurs.

1.3 Présentation du projet

Dans cette section, nous abordons le cadre de notre projet, ainsi que la problématique de notre projet.

1.3.1 Cadre de projet

Dans le cadre de notre projet de Mémoire de Mastère, nous avons été accueillis au sein de Zeta Engineering.

Notre mission consiste à mettre en place une infrastructure IT et SI pour relier les trois nouveau locaux (Tableau 1.3.1) de l'entreprise via un réseau MAN.

Logo	Adresse	Fonctionnalité
 ZETA ENGINEERING	Rades Meliane, Rue de la Solidarité	Réservé à l'administration et l'équipe diffusion globale
	Rades, Rue Habib Bourgiba	Centre d'appel
 ZETA PROJECTS	Ezzahra, Rue Habib Bourgiba	Abrite les ingénieurs et les développeurs informatique

TABLE 1.1 – Les nouveaux locaux de Zeta Engineering en Tunisie

De suite, nous allons connecter les objets de l'entreprise tels que les machines de pointage, le capteur de température et humidité, les lampes, les smartphones et les caméras de surveillance dans notre système d'information pour une utilisation plus efficace de ces ressources au sein de l'entreprise.

1.3.2 Problématique

La problématique de notre projet s'articule autour de trois domaines distincts, chacun présentant ses propres défis.

Le premier défi réside dans l’isolation des trois bureaux de ZETA, les empêchant de former un réseau local (LAN). Cette absence de connectivité entre les bureaux pose un obstacle majeur à la communication et au partage d’informations entre eux.

Deuxièmement, nous nous sommes confrontés à un problème lié à la gestion des informations. Il manque un système d’information. Cela engendre des difficultés pour stocker, gérer et accéder aux données collectées par les divers applications.

Enfin, l’intégration des dispositifs IoT constitue une autre problématique majeure. La plupart de ces appareils ne disposent pas d’une connectivité native pour rejoindre notre réseau ni pour interagir avec le système d’information que nous prévoyons de mettre en place.

Nous nous attacherons à résoudre ces problématiques tout au long de notre projet, en développant des solutions adaptées à chaque domaine, visant à améliorer la connectivité entre les bureaux, à établir un système d’information efficace, et à faciliter l’intégration des dispositifs IoT dans notre infrastructure.

1.4 Méthodologie de travail et planification

Dans cette section, nous abordons la présentation d’une méthodologie, ainsi que la problématique de notre projet.

1.4.1 Présentation de la méthodologie en cascade

Parmi les diverses méthodologies de gestion de projets telles que le modèle en V, les méthodologies itératives, Scrum, et bien d’autres, nous avons choisi d’adopter le modèle en cascade pour notre projet. Également connue sous le nom de méthodologie séquentielle, cette approche est souvent préférée dans les projets de Mémoire de Mastère en raison de sa simplicité et de sa facilité de compréhension.

Dans cette méthodologie, chaque étape doit être achevée avant que la suivante puisse commencer. Les étapes ne peuvent pas être rétrogradées

et doivent être exécutées dans l'ordre établi. Cela signifie que le projet doit être entièrement défini et spécifié avant que la conception ne puisse commencer, et que la conception doit être terminée avant que la mise en œuvre puisse commencer, et ainsi de suite.

La figure 1.1 illustre les étapes caractéristiques de la méthodologie en cascade. Cette approche linéaire et séquentielle guide le projet à travers une série d'étapes bien définies.

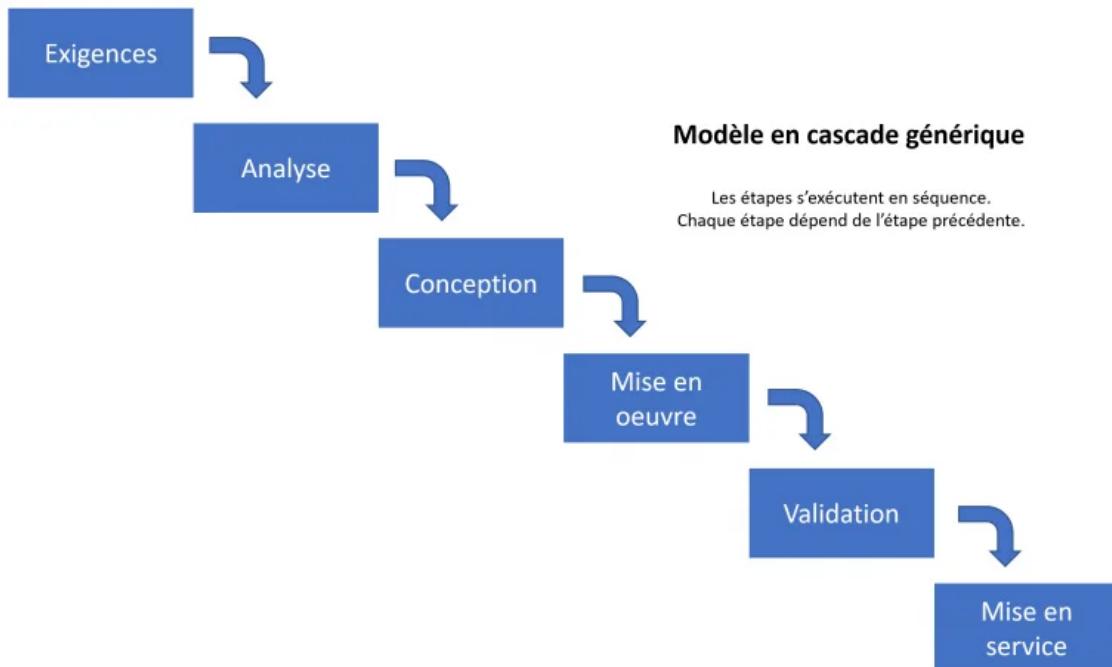


FIGURE 1.1 – Méthodologie en Cascade [1]

Le processus de la méthodologie en cascade se divise en plusieurs phases séquentielles, chacune étant une étape préalable à la suivante. Les étapes typiques incluent la définition des besoins, l'analyse, la conception, la mise en œuvre, les tests et la maintenance.

1.4.2 Choix de la méthodologie en cascade

Comme les besoins et les exigences du projet étaient relativement bien définis et stables dès le départ, le choix de la méthodologie en cascade s'est avéré approprié. Cette approche convient parfaitement lorsque les étapes du projet peuvent être planifiées à l'avance et que les changements ne sont pas fréquents.

De plus, notre équipe avait une compréhension claire des objectifs à atteindre et de la séquence des tâches à accomplir. La méthodologie en cascade nous a fourni un cadre structuré pour aborder chaque étape de manière méthodique, ce qui a permis une planification précise et une gestion efficace des ressources.

En résumé, chaque chapitre de notre rapport explique les choix matériels et logiciels adoptés, ainsi que la mise en œuvre de chaque composant du projet. Cette méthodologie nous permet de travailler de manière organisée et structurée, en veillant à ce que chaque étape soit achevée de manière adéquate avant de passer à la suivante [4].

1.5 Outils et technologies utilisés

Dans cette partie, nous allons détailler les différents outils et technologies que nous avons utilisés au sein de notre projet.

Cette présentation détaillée nous permettra de mieux comprendre les choix que nous avons effectués pour répondre aux exigences de l'entreprise et assurer une mise en œuvre réussie de notre projet.

1.5.1 Matériels utilisés

Dans cette sous-section, nous présentons le matériel utilisé pour la première partie Infrastructure IT, notamment les commutateurs, le pare-feu et les serveurs, sous forme d'un tableau 3.1.

Image de l'équipement	Nom de l'équipement	Fonctionnalité
	E-Wall Pare-Feu	<p>E-Wall est une solution de sécurité essentielle pour protéger les réseaux informatiques contre les menaces externes.</p> <p>Son utilisation permet de renforcer la sécurité, de contrôler le trafic réseau et de détecter les activités malveillantes, contribuant ainsi à la préservation de l'intégrité, de la confidentialité et de la disponibilité des données.</p>
	Commutateurs HP1910 G24	<p>C'est un modèle de commutateur d'entreprise qui offre des fonctionnalités de gestion avancées pour les réseaux LAN. Il dispose de 24 ports Gigabit Ethernet et peut être géré via une interface web ou en ligne de commande.</p> <p>Il prend en charge les protocoles de gestion de réseau SNMP, RMON et LLDP, ainsi que les VLAN, le QoS, le STP et d'autres fonctionnalités de sécurité.</p>
	TP-Link CPE610	<p>Le TP-Link CPE610 est un point d'accès extérieur sans fil conçu pour fournir une connectivité Wi-Fi à longue portée dans des environnements extérieurs. Il utilise une antenne directionnelle à haut gain pour offrir des connexions sans fil stables et fiables sur de longues distances.</p>
	Dell PowerEdge R720	<p>Le Dell PowerEdge R720 est un serveur rackable de haute performance, fiable et évolutif, offrant une grande capacité de stockage et de traitement de données pour les applications critiques de l'entreprise. Il est équipé de processeurs Intel Xeon E5-2600 v2 et peut prendre en charge jusqu'à 1,5 To de mémoire vive. Le PowerEdge R720 dispose également de fonctionnalités de haute disponibilité telles que les alimentations redondantes, les disques durs en miroir et les cartes réseau doubles.</p>

TABLE 1.2 – Matériels utilisés

1.5.2 Logiciels utilisés

Dans cette sous-section, nous présentons dans les tableaux 1.3 et 1.4 où sont répertoriés les logiciels utilisés tout au long de notre projet.

Logo	Nom de l'équipement	Fonctionnalité
	GNS3	<p>GNS3 est un logiciel open-source de simulation de réseaux informatiques.</p> <p>Il permet de créer des topologies réseau virtuelles en utilisant des images de routeurs, de commutateurs et d'autres équipements réseau.</p>
	VMware Workstation	<p>VMware Workstation est un logiciel de virtualisation puissant qui permet d'exécuter plusieurs systèmes d'exploitation sur une seule machine physique.</p> <p>Il offre la possibilité de créer et de gérer des machines virtuelles (VM) sur une plateforme hôte, ce qui permet d'exécuter plusieurs systèmes d'exploitation simultanément. VMware Workstation est disponible pour les principales plateformes telles que Windows, Linux et macOS.</p>
	VNC Viewer	<p>VNC Viewer est une application de bureau à distance qui permet de se connecter à un ordinateur distant et d'en contrôler l'écran et les applications à distance.</p> <p>Il offre la possibilité de se connecter à distance à des appareils tels que le Raspberry Pi, permettant ainsi une gestion et un contrôle pratiques de l'appareil à partir d'un autre ordinateur.</p>
	Advanced IP Scanner	<p>Advanced IP Scanner est un outil de gestion de réseau qui permet de scanner rapidement un réseau et de trouver tous les périphériques connectés, y compris les ordinateurs, les imprimantes, les routeurs, les commutateurs, les caméras IP, etc.</p> <p>Il permet également de détecter les adresses IP inactives et les ports ouverts sur les périphériques.</p>
	Fritzing	<p>Fritzing est un logiciel de conception électronique open-source qui permet de créer des schémas électroniques, des circuits imprimés et des modèles de connexion pour des projets électroniques.</p> <p>Il offre une interface conviviale et intuitive, ce qui facilite la création de schémas électroniques même pour les débutants.</p>

TABLE 1.3 – Logiciels utilisés

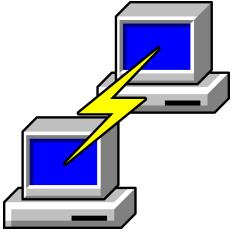
	Putty	<p>PuTTY est un logiciel client pour les protocoles de communication SSH, Telnet et Rlogin, permettant d'établir des connexions réseau sécurisées avec des ordinateurs distants.</p> <p>Il est utilisé principalement pour administrer des serveurs à distance, mais peut également être utilisé pour des transferts de fichiers et des sessions de console distantes.</p>
 Raspberry Pi OS	Raspberry Pi Imager	<p>Raspberry Pi Imager est un logiciel qui permet de facilement installer des systèmes d'exploitation sur une carte SD ou une clé USB pour Raspberry Pi.</p> <p>Il offre une interface graphique simple à utiliser pour sélectionner et télécharger les images d'OS disponibles.</p>
	Xming	<p>Xming est un serveur d'affichage X open source qui permet d'exécuter des applications graphiques Linux sur des systèmes Windows.</p> <p>Il fournit un environnement graphique pour les applications Linux, ce qui permet aux utilisateurs de Windows d'accéder et d'interagir avec des applications Linux à distance.</p>

TABLE 1.4 – Logiciels utilisés - Suite

1.6 Conclusion

En conclusion, ce premier chapitre nous a permis de présenter l'organisme d'accueil, Zeta Engineering, ainsi que le contexte du projet consistant à ouvrir trois nouveaux bureaux à Rades, Rades Meliane et Ezzahra, et les défis qui en découlent.

Nous avons ensuite abordé la méthodologie choisie pour la gestion de ce projet, à savoir la méthode en cascade, qui nous permet de travailler de manière séquentielle en terminant chaque phase avant de passer à la suivante.

Les choix en matière de matériels et logiciels pour chaque étape sont également détaillés dans ce chapitre suivants, garantissant ainsi une bonne exécution du projet.

Chapitre 2

Déploiement de l'infrastructure IT

2.1 Introduction

Le déploiement de l'infrastructure informatique est une étape cruciale pour Zeta Engineering. Après avoir choisi la méthodologie en cascade et détailler les choix matériels et logiciels, nous sommes prêts à passer à la mise en place de l'infrastructure nécessaire à la réalisation du projet.

Tout au long de ce chapitre, nous détaillons les différentes étapes du test, déploiement et expliquons les raisons de nos choix techniques afin de permettre une compréhension approfondie de notre approche.

2.2 Présentation des locaux et déploiement de réseaux MAN

Cette section, nous débutons par une présentation détaillée de nos trois locaux, chacun ayant ses caractéristiques uniques. Ensuite, nous explorons la problématique de notre réseau et la solution proposée sous la notion de réseau MAN et son importance pour notre entreprise.

2.2.1 Présentation des trois locaux

Le premier local, situé à Rades Meliane, est principalement réservé à l'administration de l'entreprise et à l'équipe chargée de la diffusion globale. Il s'agit d'un espace central pour la gestion des activités quotidiennes de l'entreprise et la coordination de ses opérations à l'échelle nationale et internationale. C'est également le local le plus important en termes de sécurité, c'est pourquoi nous y installons un Pare-Feu dédié.

Le deuxième local, situé à Rades, abrite le centre d'appel de Zeta Engineering. C'est ici que l'équipe prend en charge les appels des clients, traite leurs demandes et fournit un support essentiel pour la clientèle. Nous avons donc besoin principalement d'une connexion Internet par fibre optique fiable, c'est pourquoi nous installons un routeur Cisco à cet endroit.

Enfin, le troisième local, situé à Ezzahra, est l'espace de travail pour les ingénieurs et les développeurs informatiques de l'entreprise. C'est dans cet espace que les produits technologiques de Zeta Engineering sont conçus et développés.

Ces deux figures 2.1 et figure 2.2 ci-dessous montrent la distance entre le bureau Rades Meliane et le bureau Rades, qui est d'environ 1,268 km, ainsi que la distance entre le bureau Rades Meliane et Ezzahra, qui est d'environ 2,843 km.

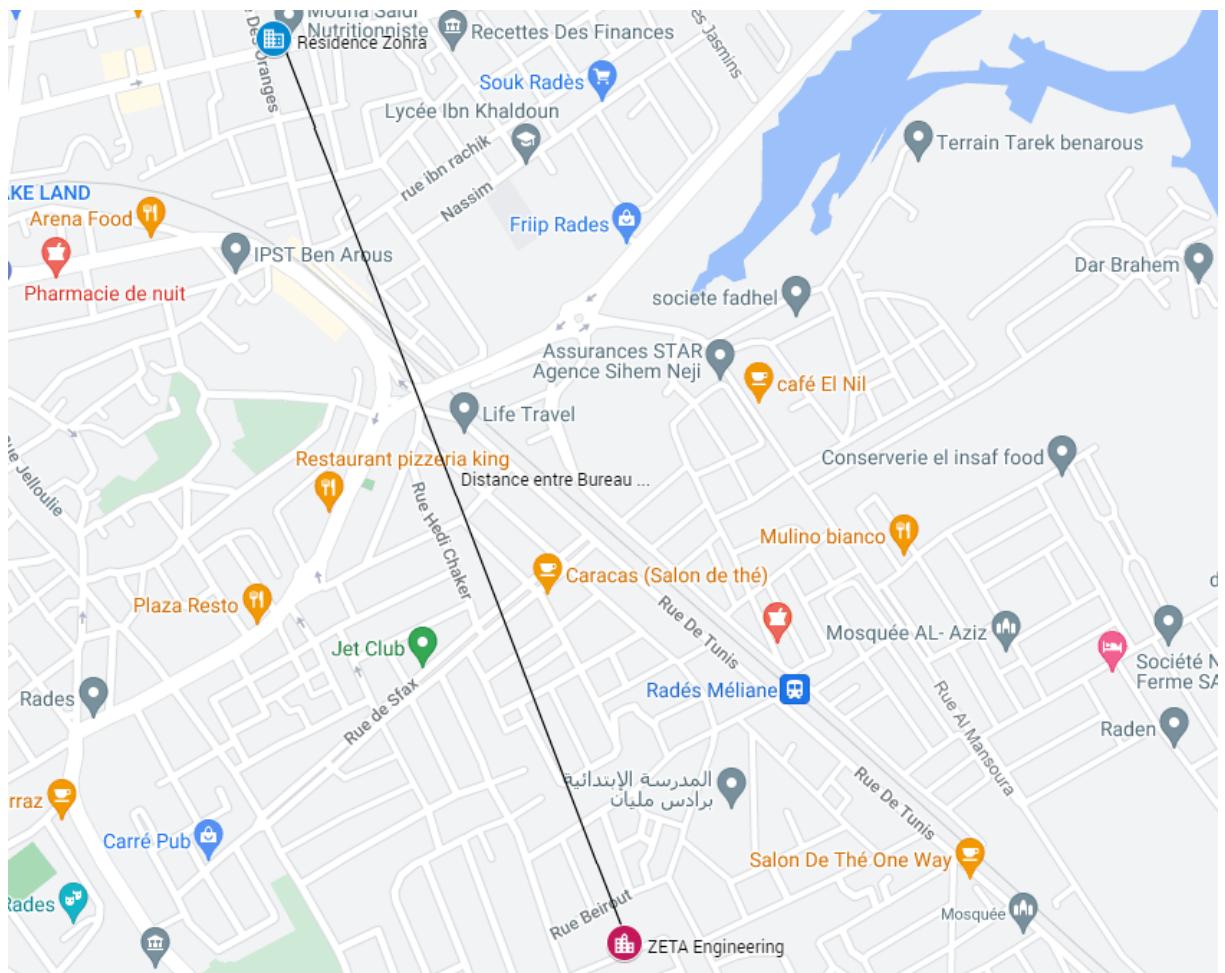


FIGURE 2.1 – Distance entre le bureau Rades et Rades Meliane (1,268 km)

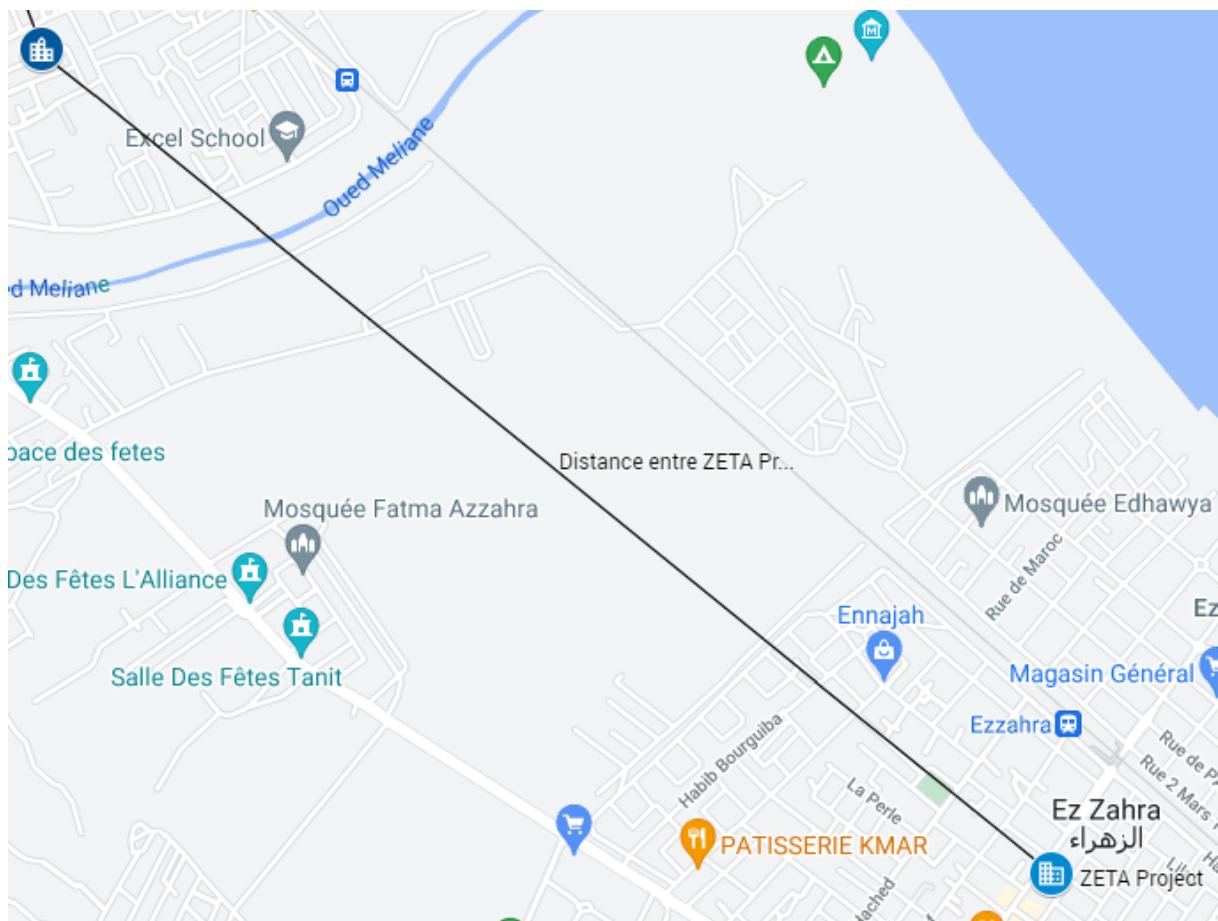


FIGURE 2.2 – Distance entre bureau Rades et bureau Ezzhara (2,843 km)

Avec ces trois locaux répartis géographiquement, comme le montre la figure 2.3, il devient essentiel de mettre en place un réseau de zone métropolitaine (MAN) pour les connecter de manière transparente.

Un réseau MAN permettra à Zeta Engineering de relier ses bureaux de Rades, Rades Meliane et Ezzahra, garantissant ainsi une communication fluide et efficace entre eux.

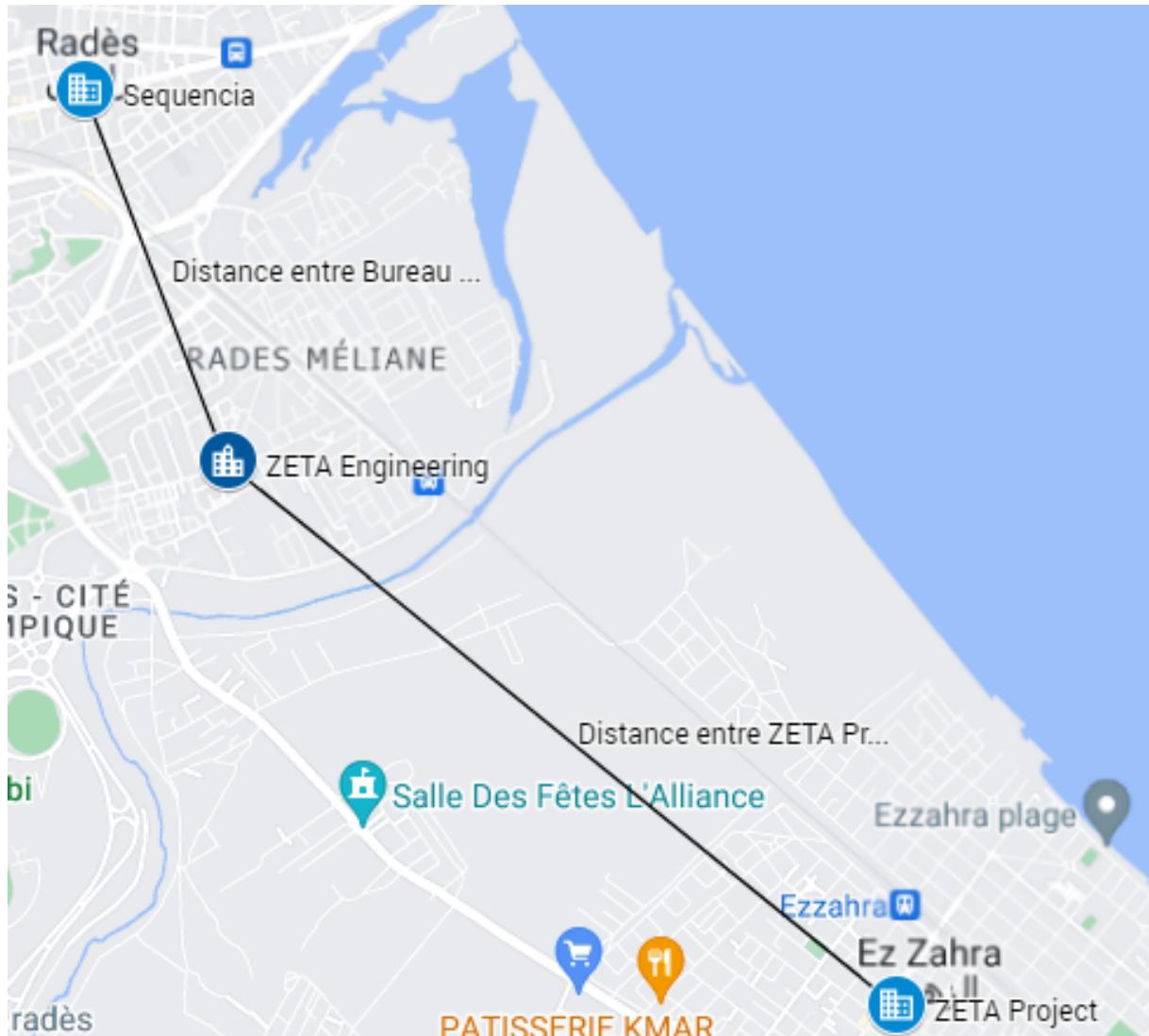


FIGURE 2.3 – Distance totale entre les bureaux

Ce réseau joue un rôle crucial dans la centralisation des données, la gestion des ressources et la coordination des activités à l'échelle de l'entreprise. De plus, il intègre également les objets connectés tels que les machines de pointage, les capteurs de température et d'humidité, les lampes, les smartphones et les caméras de surveillance, optimisant ainsi l'utilisation de ces

ressources au sein de l'entreprise.

2.2.2 Définition du réseau MAN

Un réseau MAN est un type de réseau de communication qui couvre une zone géographique intermédiaire entre un réseau LAN et un réseau WAN [5].

Metropolitan area network (MAN)

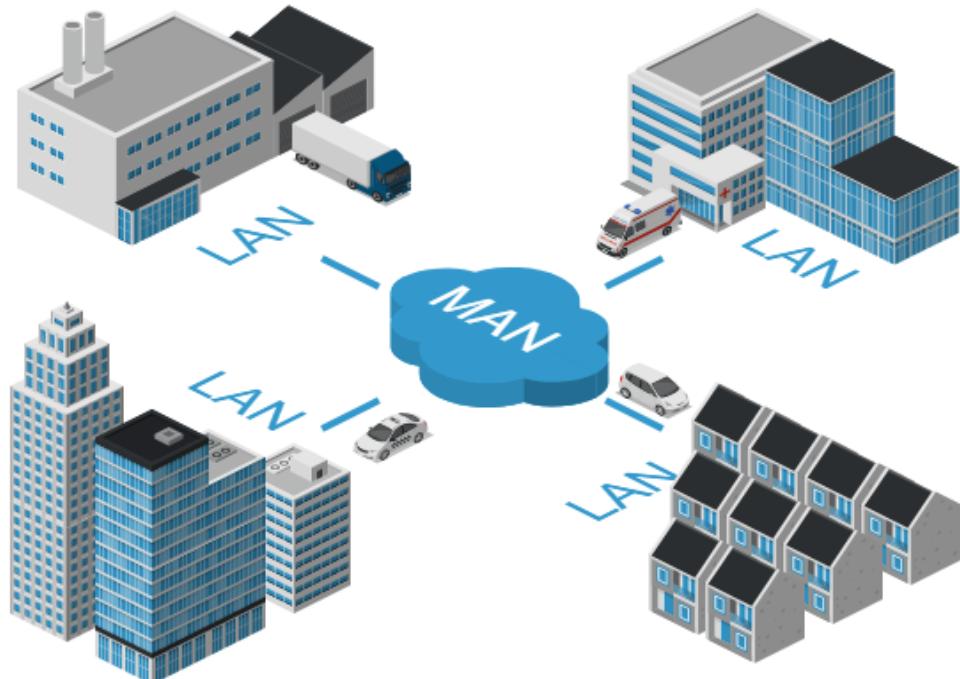


FIGURE 2.4 – Metropolitan Area Network

Contrairement à un réseau LAN qui est généralement limité à une seule installation ou à un seul site comme démontré dans la figure 2.4, un réseau MAN s'étend sur une zone géographique plus étendue, comme une ville entière ou une zone métropolitaine.

Dans le contexte de notre projet à Zeta Engineering, l'objectif est de relier les trois bureaux de l'entreprise situés à Rades, Rades Meliane et Ezzahra via un réseau MAN pour permettre une communication efficace

et une gestion centralisée des ressources.

2.3 L'architecture réseau

Dans cette section, nous plongeons au cœur de l'infrastructure qui soutient notre entreprise. Nous explorons en détail l'architecture réseau qui forme le socle de nos opérations.

De la topologie de nos bureaux aux configurations dans GNS3 et à la mise en œuvre de notre infrastructure, cette section dévoile les coulisses de notre système de communication et de connectivité.

2.3.1 Topologie en GNS3

Dans cette section, nous présentons l'architecture réseau de chaque site à l'aide de simulations réalisées dans GNS3.

Cette approche nous a également donné l'opportunité d'anticiper et de résoudre d'éventuels problèmes avant qu'ils ne se manifestent dans le réseau réel, en effectuant des ajustements aux paramètres de configuration et en ajoutant de nouveaux équipements au besoin.

Pour commencer, nous modélisons les différents équipements du réseau de Rades, comme illustré dans la figure 2.5. Le routeur C891F, par exemple, distribue l'accès Internet reçu de Topnet vers le commutateur principal HP-1910-G24. À partir de ce commutateur, nous gérons l'accès à Internet pour les utilisateurs et les connectons aux serveurs de la société.

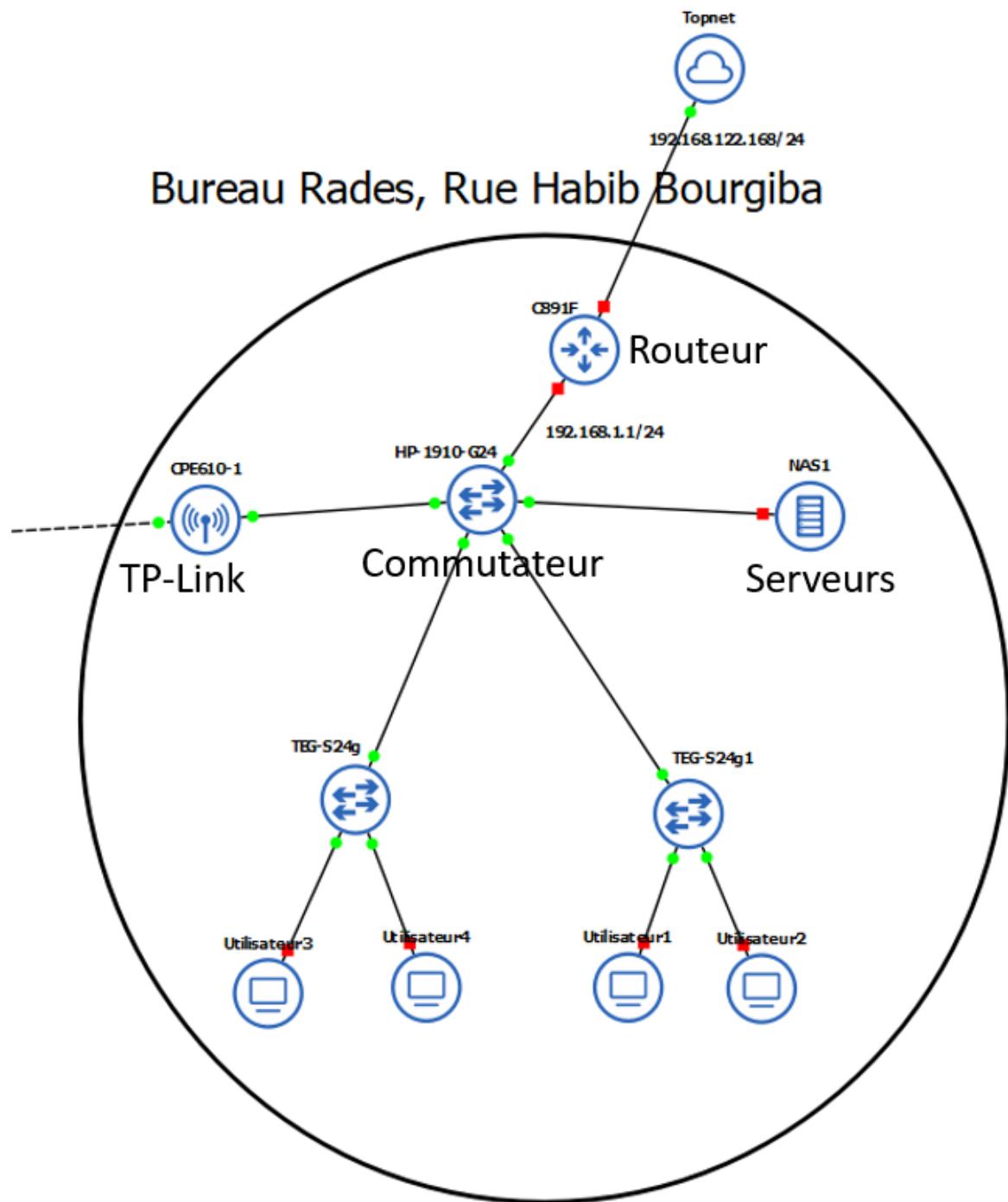


FIGURE 2.5 – Topologie du réseau du bureau de Rades

Ensuite, dans la figure 2.6 nous présentons la topologie du réseau du bureau de Rades Meliane où nous remarquons que cette structure est similaire à celle mise en place dans le bureau de Rades, à la différence que nous ajoutons pfSense pour renforcer la sécurité du réseau principal vu que c'est le local le plus important assurant la gestion des activités quotidiennes

de l'entreprise et la coordination de ses opérations à l'échelle nationale et internationale.

Bureau Rades Melian, Rue Solidarité

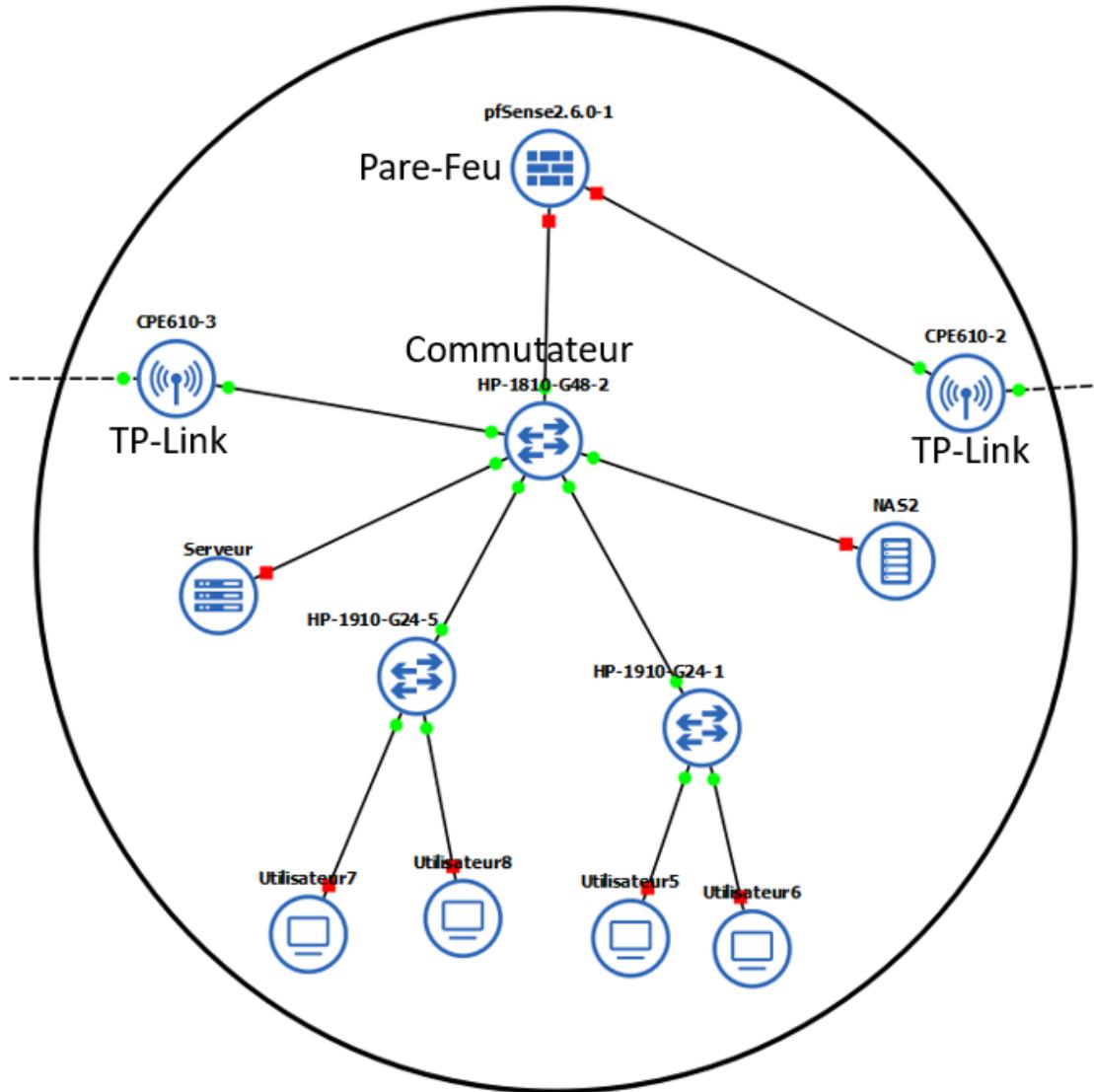


FIGURE 2.6 – Topologie du réseau du bureau de Rades Melian

Enfin dans la figure 2.7, pour le bureau Ezzahara nous ajoutons une source Internet supplémentaire à notre réseau au cas où l'accès Internet de Rades serait coupé ou interrompu.

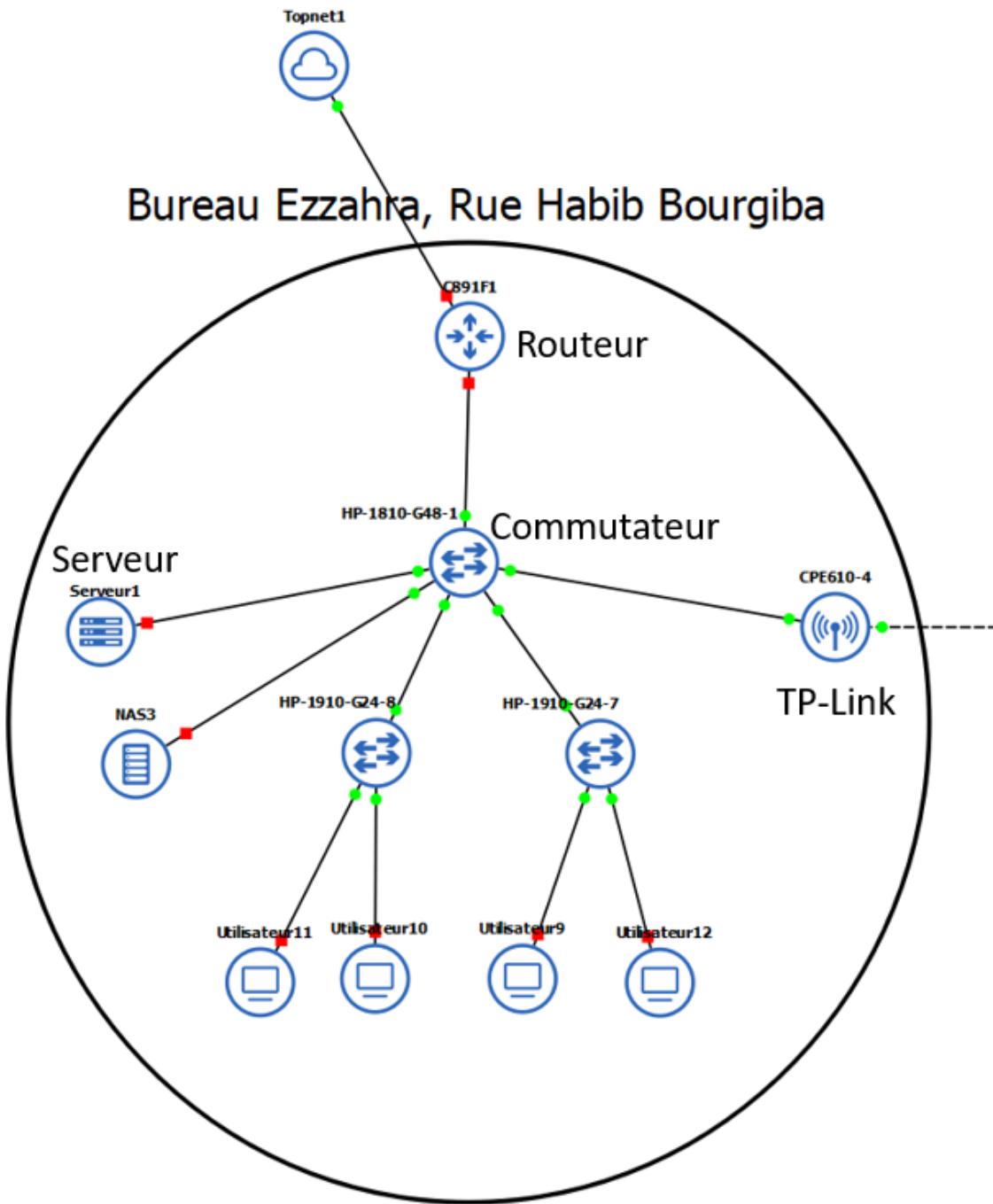


FIGURE 2.7 – Topologie du réseau du bureau Ezzahara

2.3.2 Configuration en GNS3

Dans cette section, nous présentons la configuration de notre réseau en utilisant GNS3. Tout d'abord, nous montrons la topologie du réseau dans la figure 2.8, où nous avons accès à la console du routeur.

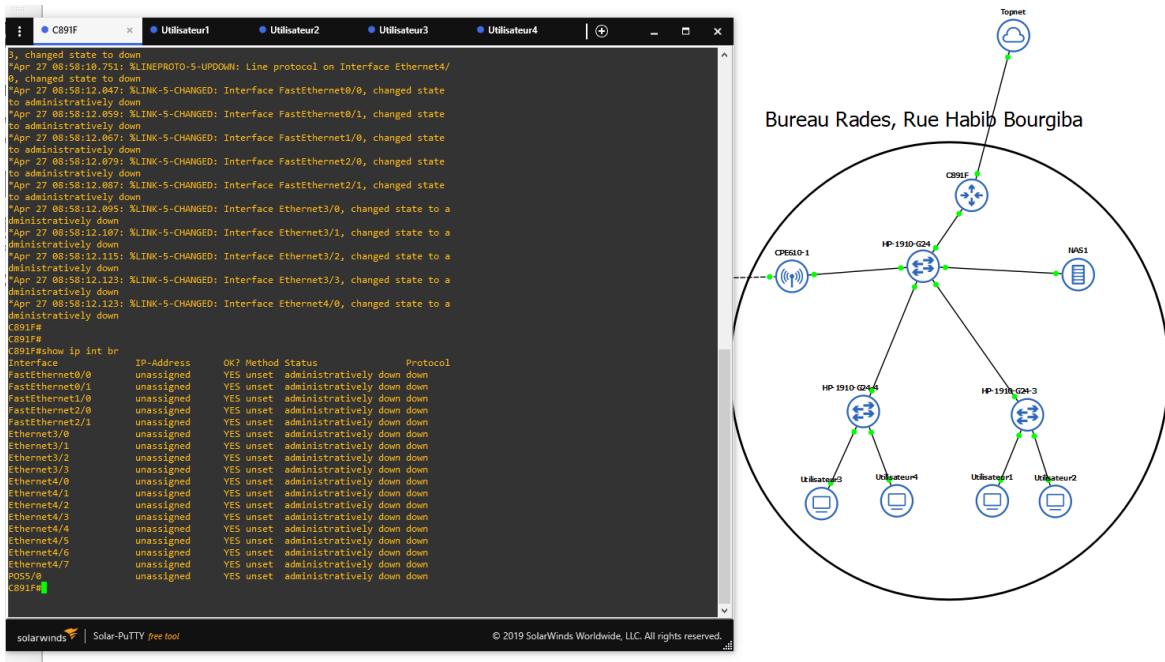


FIGURE 2.8 – Console du routeur de Rades

Ensuite, dans la figure 2.9, nous avons obtenu une adresse IP DHCP pour notre console et nous avons activé l'option "*domain-lookup*".

```

C891F#conf t
Enter configuration commands, one per line. End with CNTL/Z.
C891F(config)#int fa 0/0
C891F(config-if)#ip address dhcp
C891F(config-if)#no shut
C891F(config-if)#end
C891F#
*Apr 27 09:28:26.683: %SYS-5-CONFIG_I: Configured from console by console
C891F#
*Apr 27 09:28:27.931: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Apr 27 09:28:28.931: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
C891F#ip domain-lookup
^
% Invalid input detected at '^' marker.

C891F#
*Apr 27 09:28:39.575: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned DHCP address 192.168.122.168, mask 255.255.
255.0, hostname C891F

C891F#conf t
Enter configuration commands, one per line. End with CNTL/Z.
C891F(config)#ip domain-lookup
C891F(config)#end
C891F#
*Apr 27 09:28:57.947: %SYS-5-CONFIG_I: Configured from console by console
C891F#show ip int br
Interface          IP-Address      OK? Method Status           Protocol
FastEthernet0/0    192.168.122.168 YES  DHCP   up              up
FastEthernet0/1    unassigned      YES  unset  administratively down
FastEthernet1/0    unassigned      YES  unset  administratively down
FastEthernet2/0    unassigned      YES  unset  administratively down
FastEthernet2/1    unassigned      YES  unset  administratively down
Ethernet3/0        unassigned      YES  unset  administratively down
Ethernet3/1        unassigned      YES  unset  administratively down
Ethernet3/2        unassigned      YES  unset  administratively down
Ethernet3/3        unassigned      YES  unset  administratively down

```

SolarWinds | Solar-PuTTY free tool © 2019 SolarWinds Worldwide, LLC. All rights reserved.

FIGURE 2.9 – Réglage de l’IP adresse

Dans la figure 2.10, nous effectuons un ping réussi vers "8.8.8.8", ce qui indique que nous avons accès à l’Internet, et également vers "www.google.com" grâce à l’option "domain-lookup" que nous avons utilisée.

The screenshot shows a SolarWinds Putty terminal window with four tabs at the top: C891F (selected), Utilisateur1, Utilisateur2, Utilisateur3, and Utilisateur4. The C891F tab displays the following configuration and test results:

```
Enter configuration commands, one per line. End with CNTL/Z.
C891F(config)#ip domain-lookup
C891F(config)#end
C891F#
*Apr 27 09:28:57.947: %SYS-5-CONFIG_I: Configured from console by console
C891F#show ip int br
Interface          IP-Address      OK? Method Status           Protocol
FastEthernet0/0    192.168.122.168 YES DHCP   up              up
FastEthernet0/1    unassigned      YES unset  administratively down down
FastEthernet1/0    unassigned      YES unset  administratively down down
FastEthernet2/0    unassigned      YES unset  administratively down down
FastEthernet2/1    unassigned      YES unset  administratively down down
Ethernet3/0        unassigned      YES unset  administratively down down
Ethernet3/1        unassigned      YES unset  administratively down down
Ethernet3/2        unassigned      YES unset  administratively down down
Ethernet3/3        unassigned      YES unset  administratively down down
Ethernet4/0        unassigned      YES unset  administratively down down
Ethernet4/1        unassigned      YES unset  administratively down down
Ethernet4/2        unassigned      YES unset  administratively down down
Ethernet4/3        unassigned      YES unset  administratively down down
Ethernet4/4        unassigned      YES unset  administratively down down
Ethernet4/5        unassigned      YES unset  administratively down down
Ethernet4/6        unassigned      YES unset  administratively down down
Ethernet4/7        unassigned      YES unset  administratively down down
POS5/0             unassigned      YES unset  administratively down down
C891F#ping 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/40/44 ms
C891F#ping www.google.com
Translating "www.google.com"...domain server (192.168.122.1) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 142.251.209.36, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/36/44 ms
C891F#
```

SolarWinds Putty | Solar-PuTTY free tool © 2019 SolarWinds Worldwide, LLC. All rights reserved.

FIGURE 2.10 – Test de connexion de routeur vers l'internet

Dans la figure 2.11, nous ajoutons un réseau interne (LAN) "192.168.1.1/24" sur le port f0/1.

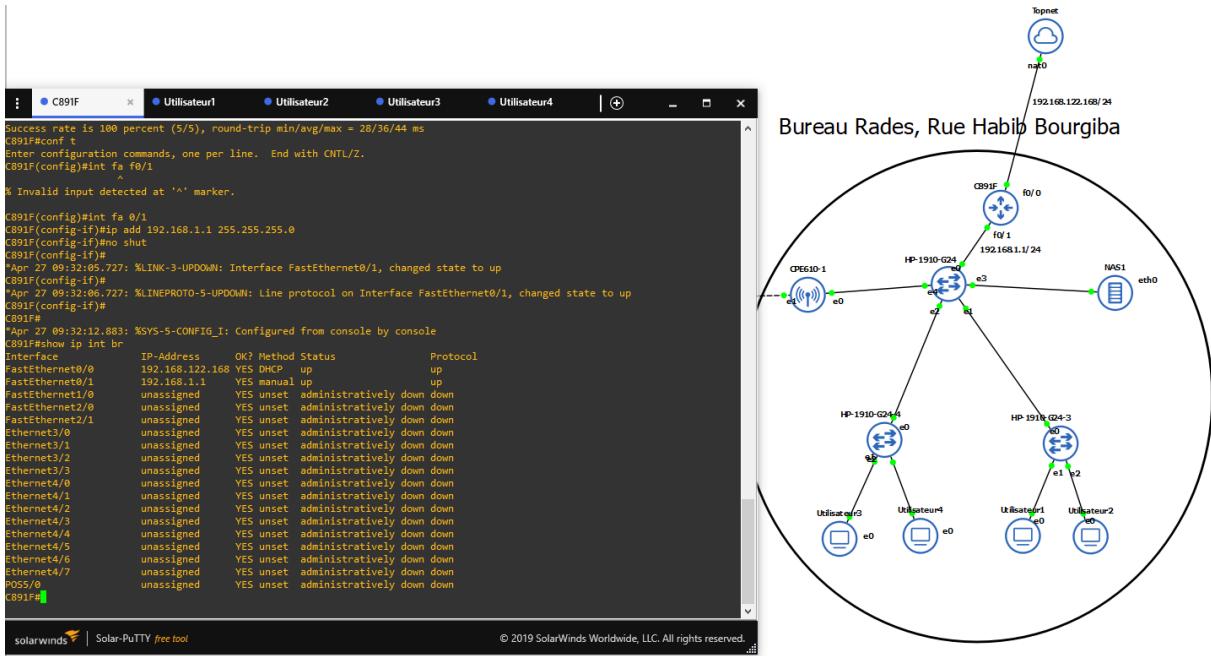


FIGURE 2.11 – Configurer le port f0/1

Cependant, comme le montre la figure 2.12, nous avons remarqué que l'utilisateur 4 ne pouvait pas toujours accéder à Internet.

Cela est dû au fait que nous devons configurer notre routeur C891F pour autoriser le trafic du port LAN f0/1 à accéder au WAN port f0/0 (le port de l'internet), et également pour permettre le transfert de tous les protocoles, tels que HTTP et SMTP.

The screenshot shows a Solar-PuTTY terminal window with four tabs at the top: C891F, Utilisateur1, Utilisateur2, Utilisateur3, and Utilisateur4. The Utilisateur4 tab is active. The terminal window displays the following text:

```
Welcome to Virtual PC Simulator, version 0.8.2
Dedicated to Daling.
Build time: Aug 23 2021 11:15:00
Copyright (c) 2007-2015, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file

Utilisateur4> ip 192.168.1.21/24 192.168.1.1
Checking for duplicate address...
Utilisateur4 : 192.168.1.21 255.255.255.0 gateway 192.168.1.1

Utilisateur4> ping 8.8.8.8

8.8.8.8 icmp_seq=1 timeout
8.8.8.8 icmp_seq=2 timeout
8.8.8.8 icmp_seq=3 timeout
8.8.8.8 icmp_seq=4 timeout
8.8.8.8 icmp_seq=5 timeout

Utilisateur4> [redacted]
```

At the bottom left, it says "solarwinds" with a logo and "Solar-PuTTY free tool". At the bottom right, it says "© 2019 SolarWinds Worldwide, LLC. All rights reserved."

FIGURE 2.12 – Ping de utilisateur4 vers l'internet

Dans la figure 2.13, nous configurons notre routeur afin d'autoriser la communication entre notre port WAN f0/0 et notre port LAN f0/1.

The screenshot shows a Solar-PuTTY terminal window titled 'C891F'. The window displays the configuration of a Cisco router. The configuration includes interface status, IP address assignments, and NAT configuration for interface fa0/1. The Solar-PuTTY logo and copyright information are visible at the bottom.

```
C891F(config-if)#
*Apr 27 09:32:06.727: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
C891F(config-if)#
C891F#
*Apr 27 09:32:12.883: %SYS-5-CONFIG_I: Configured from console by console
C891F#show ip int br
Interface          IP-Address      OK? Method Status          Protocol
FastEthernet0/0    192.168.122.168 YES DHCP   up           up
FastEthernet0/1    192.168.1.1   YES manual up           up
FastEthernet1/0    unassigned     YES unset  administratively down down
FastEthernet2/0    unassigned     YES unset  administratively down down
FastEthernet2/1    unassigned     YES unset  administratively down down
Ethernet3/0       unassigned     YES unset  administratively down down
Ethernet3/1       unassigned     YES unset  administratively down down
Ethernet3/2       unassigned     YES unset  administratively down down
Ethernet3/3       unassigned     YES unset  administratively down down
Ethernet4/0       unassigned     YES unset  administratively down down
Ethernet4/1       unassigned     YES unset  administratively down down
Ethernet4/2       unassigned     YES unset  administratively down down
Ethernet4/3       unassigned     YES unset  administratively down down
Ethernet4/4       unassigned     YES unset  administratively down down
Ethernet4/5       unassigned     YES unset  administratively down down
Ethernet4/6       unassigned     YES unset  administratively down down
Ethernet4/7       unassigned     YES unset  administratively down down
POS5/0            unassigned     YES unset  administratively down down
C891F#
C891F#
C891F#conf t
Enter configuration commands, one per line. End with CNTL/Z.
C891F(config)#int fa0/1
C891F(config-if)#ip nat inside
C891F(config-if)#int fa 0/0
C891F(config-if)#ip nat outside
C891F(config-if)#exit
C891F(config)#ip nat inside source list 1 interface fa0/0 overload
C891F(config)#ip access-list standard 1
C891F(config-std-nacl)#permit any
C891F(config-std-nacl)#[
```

FIGURE 2.13 – Configuration de routeur

Dans la figure 2.14, nous constatons que toutes les machines avaient accès à internet et pouvaient communiquer entre elles dans le réseau local.

```

Utilisateur4> ip 192.168.1.21/24 192.168.1.1
Checking for duplicate address...
Utilisateur4 : 192.168.1.21 255.255.255.0 gateway 192.168.1.1

Utilisateur4> ping 8.8.8.8

8.8.8.8 icmp_seq=1 ttl=126 time=41.847 ms
8.8.8.8 icmp_seq=2 ttl=126 time=42.280 ms
8.8.8.8 icmp_seq=3 ttl=126 time=38.360 ms
8.8.8.8 icmp_seq=4 ttl=126 time=40.817 ms
8.8.8.8 icmp_seq=5 ttl=126 time=32.885 ms

Utilisateur4> ping 192.168.1.11

84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=0.839 ms
84 bytes from 192.168.1.11 icmp_seq=2 ttl=64 time=0.826 ms
84 bytes from 192.168.1.11 icmp_seq=3 ttl=64 time=0.851 ms
84 bytes from 192.168.1.11 icmp_seq=4 ttl=64 time=0.850 ms
84 bytes from 192.168.1.11 icmp_seq=5 ttl=64 time=0.829 ms

Utilisateur4> ping 192.168.1.20

84 bytes from 192.168.1.20 icmp_seq=1 ttl=64 time=0.286 ms
84 bytes from 192.168.1.20 icmp_seq=2 ttl=64 time=0.317 ms
84 bytes from 192.168.1.20 icmp_seq=3 ttl=64 time=0.324 ms
84 bytes from 192.168.1.20 icmp_seq=4 ttl=64 time=0.319 ms
84 bytes from 192.168.1.20 icmp_seq=5 ttl=64 time=0.320 ms

Utilisateur4> [REDACTED]

solarwinds | SolarPuTTY free tool | © 2019 SolarWinds Worldwide, LLC. All rights reserved.

: Utilisateur2 | Utilisateur3 | Utilisateur4 | + - x

Welcome to Virtual PC Simulator, version 0.8.2
Dedicated to Daling.
Build time: Aug 23 2021 11:15:00
Copyright (c) 2007-2015, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file

Utilisateur2> ip 192.168.1.11/24 192.168.1.1
Checking for duplicate address...
Utilisateur2 : 192.168.1.11 255.255.255.0 gateway 192.168.1.1

Utilisateur2> ping 192.168.1.21

84 bytes from 192.168.1.21 icmp_seq=1 ttl=64 time=1.358 ms
84 bytes from 192.168.1.21 icmp_seq=2 ttl=64 time=0.805 ms
84 bytes from 192.168.1.21 icmp_seq=3 ttl=64 time=0.748 ms
84 bytes from 192.168.1.21 icmp_seq=4 ttl=64 time=0.865 ms
84 bytes from 192.168.1.21 icmp_seq=5 ttl=64 time=0.728 ms

Utilisateur2> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=41.785 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=34.817 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=126 time=35.266 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=36.690 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=126 time=35.307 ms

Utilisateur2> [REDACTED]

solarwinds | SolarPuTTY free tool | © 2019 SolarWinds Worldwide, LLC. All rights reserved.

```

FIGURE 2.14 – Test de connexion des utilisateurs

En résumé, nous avons configuré notre réseau en utilisant GNS3 en étapes, en veillant à ce que toutes les machines aient accès à internet et puissent communiquer entre elles dans le réseau local.

2.4 Implémentation

Dans la section d'implémentation, nous commençons par détailler la mise en place du réseau local (LAN) pour chaque bureau. Ensuite, nous abordons l'implémentation du réseau MAN, qui permet la connexion entre les trois LAN. Enfin, nous concluons en expliquant l'ajout du pare-feu pour l'un de nos bureaux.

2.4.1 Implémentation des LANs

Étant donné que l'infrastructure de fibre optique de Topnet/Telecom n'est pas disponible à proximité du bureau central de Rades Melian, nous trouvons une solution en achetant une connexion internet par fibre optique dans deux autres bureaux, Rades (20 Mbps) et Ezzahara (100 Mbps), et en les reliant via un réseau MAN (Metropolitan Area Network). Les deux connexions sont connectées à un routeur Cisco 891F configuré par l'équipe de Topnet, qui distribue le réseau vers un commutateur Cisco v19-10 24G.

Le routeur Cisco 891F est représenté dans la figure 2.15.

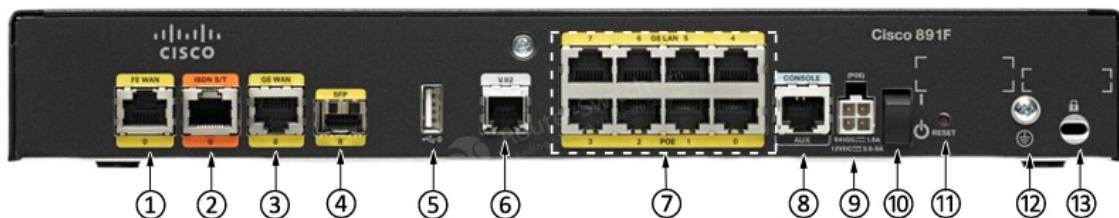


FIGURE 2.15 – Routeur Cisco 891F

À partir du commutateur principale, nous pouvons accéder à d'autres

commutateurs qui permettent la connectivité LAN des PC et des objets connectés dans le bureau. La figure 2.16 montre l'interconnexion des commutateurs principal et les routeurs pour les bureaux de Rades et Ezzahara.

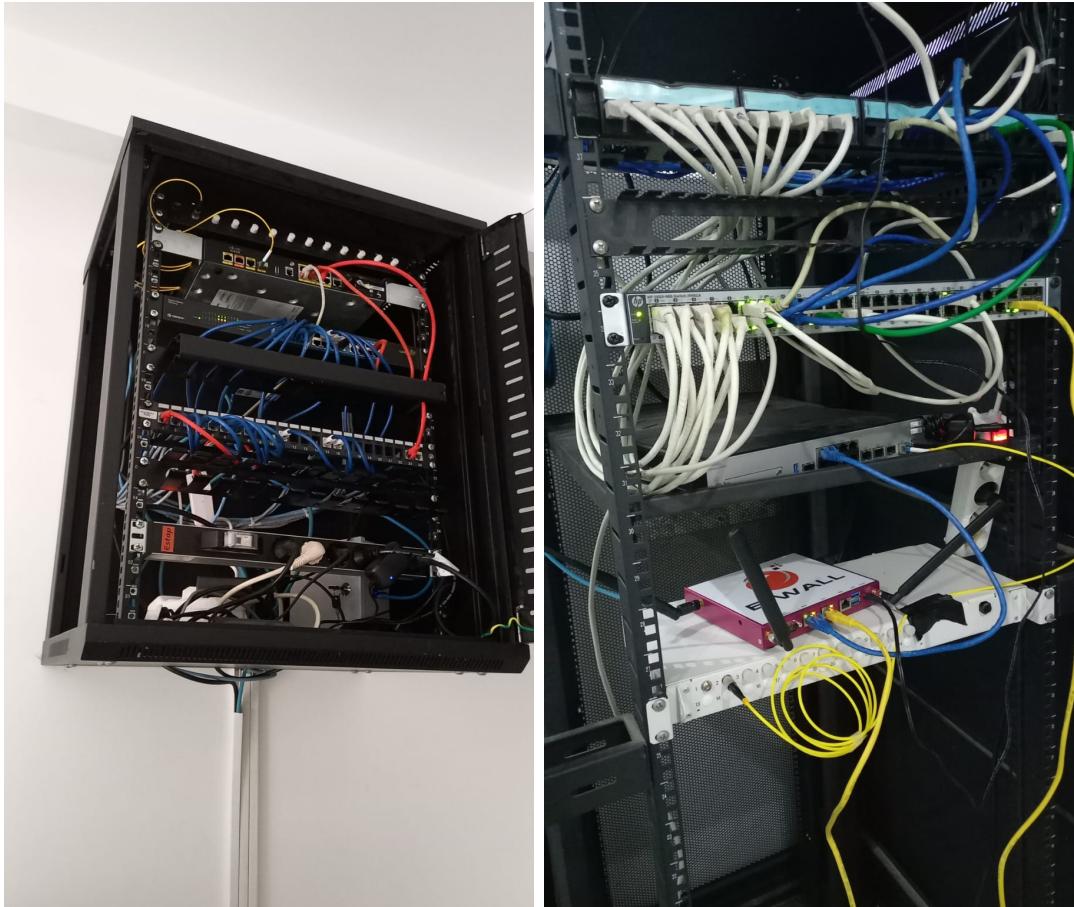


FIGURE 2.16 – Interconnexion des commutateurs pour les bureaux Rades et Ezzahara

2.4.2 Implémentation du MAN et le choix de TP-Link

Pour établir notre réseau étendu (MAN), nous choisissons d'utiliser le TP-Link CPE610. Ce point d'accès sans fil est spécialement conçu pour offrir une connectivité longue distance en utilisant la fréquence de 5 GHz.

Nous suivons les étapes d'assemblage et d'installation illustrées dans les figures 2.17 et 2.18 ci-dessous pour installer le TP-Link CPE610.



FIGURE 2.17 – Déballage du TP-Link CPE610

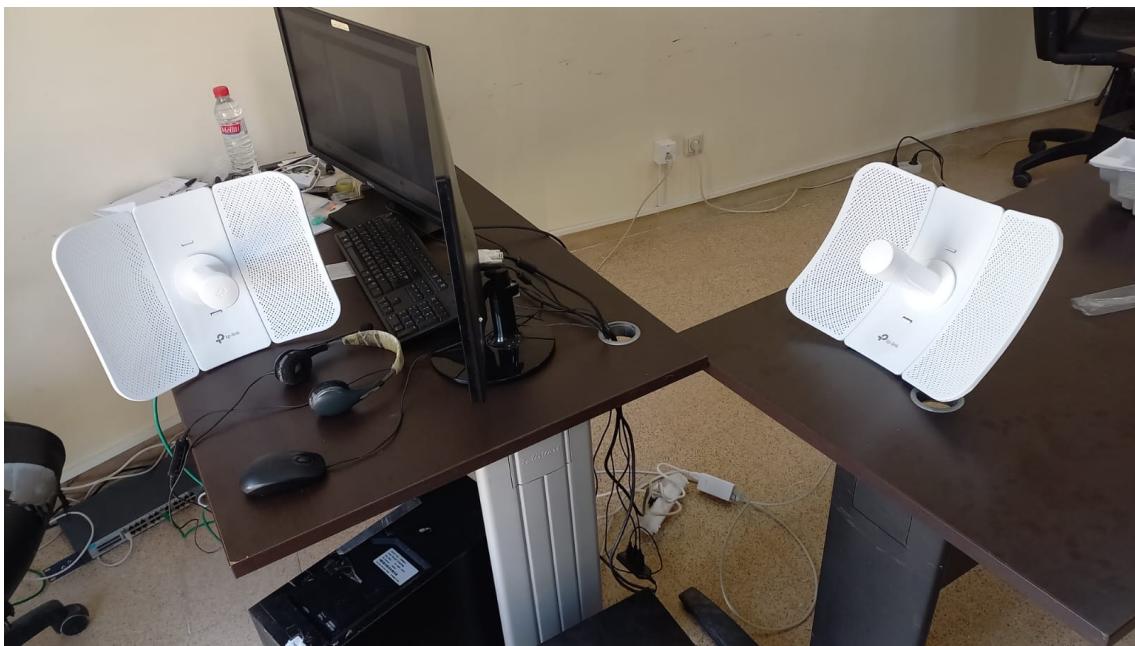


FIGURE 2.18 – Assemblage des pièces du TP-Link CPE610

Après avoir soigneusement assemblé les différentes composantes du TP-Link, nous sommes en mesure d'accéder à son panneau d'administration via une connexion LAN. À partir de là, nous entamons le processus d'initialisation et de configuration.

La figure 2.19 illustre l'étape d'initialisation du TP-Link, où nous configurons les paramètres essentiels pour son bon fonctionnement.

Login



User Name: !

Password:

Region: Please Select a Region

Language: English

TERMS OF USE

This TP-Link wireless device must be installed by a certified professional. Properly installed shielded Ethernet cable and earth grounding must be used in compliance with this product's warranty. Installers must abide by local rules and regulations in terms of legal frequency channels, output power, and Dynamic Frequency Selection (DFS) requirements. The End User accepts responsibility for maintaining the product in accordance with these rules and regulations. For further information, please visit www.tp-link.com.

I agree to these terms of use

Login

Clear

FIGURE 2.19 – Initialisation du TP-Link

La configuration en mode test est représentée dans la figure 2.20, nous permettant de vérifier les performances du dispositif avant une utilisation complète.

Login



User Name: admin

Password:

Region: Test_Mode

Language: English

TERMS OF USE

This TP-Link wireless device must be installed by a certified professional. Properly installed shielded Ethernet cable and earth grounding must be used in compliance with this product's warranty. Installers must abide by local rules and regulations in terms of legal frequency channels, output power, and Dynamic Frequency Selection (DFS) requirements. The End User accepts responsibility for maintaining the product in accordance with these rules and regulations. For further information, please visit www.tp-link.com.

I agree to these terms of use

Login

Clear

FIGURE 2.20 – Initialisation du TP-Link en mode test

Le panneau de contrôle complet du TP-Link est dépeint dans la figure 2.21, où nous pouvons régler et surveiller divers paramètres pour garantir un réseau stable.

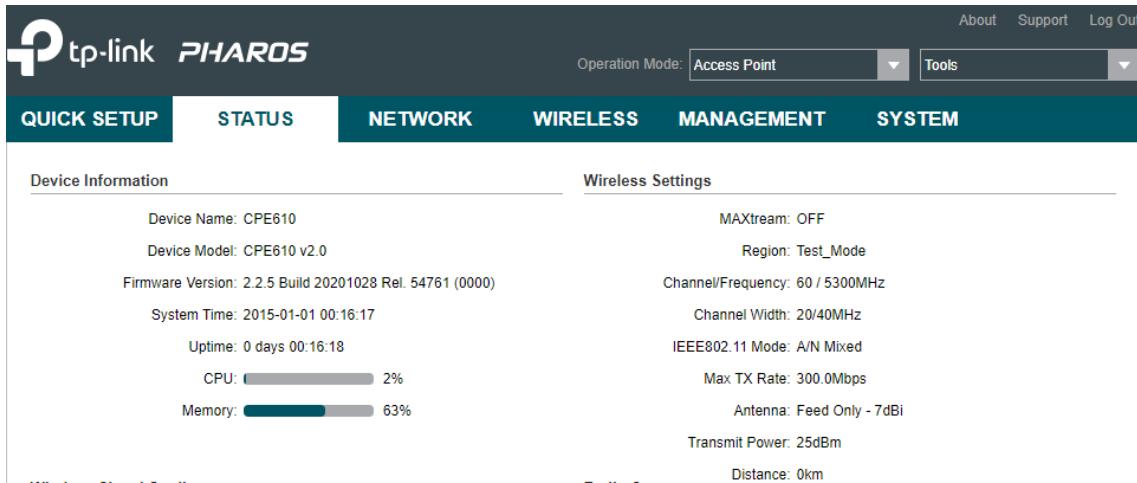


FIGURE 2.21 – Panneau de contrôle du TP-Link

Les figures 2.22 et 2.23 montrent respectivement les étapes d’initialisation de l’émetteur et du récepteur du TP-Link, éléments clés pour établir une connexion fiable.

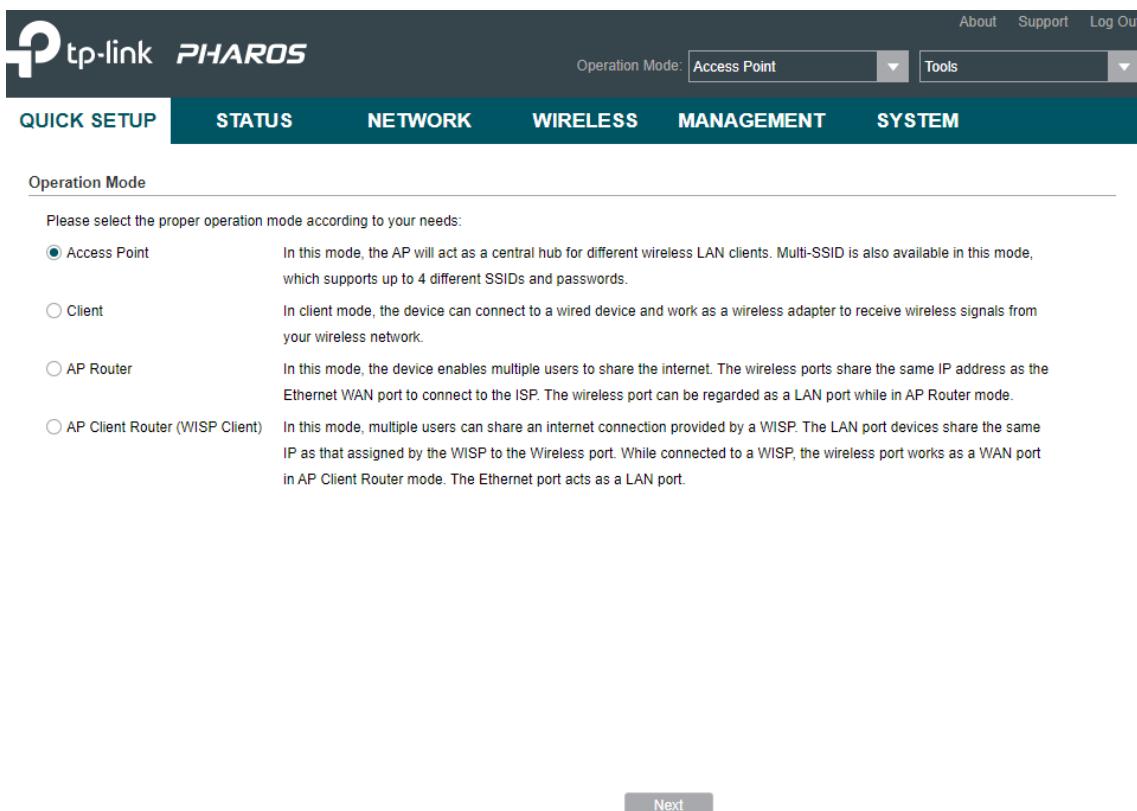


FIGURE 2.22 – Initialisation de l’émetteur

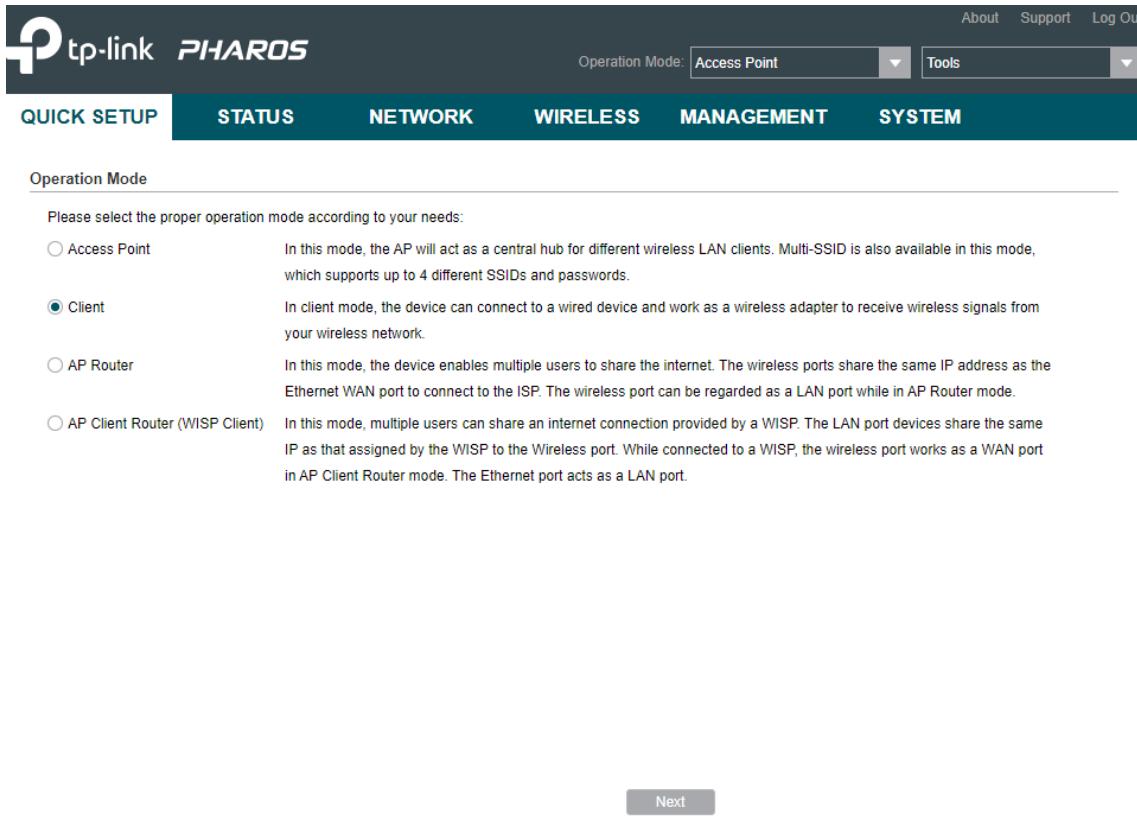


FIGURE 2.23 – Initialisation du récepteur

L’étape finale de vérification de la connexion entre le point d'accès TP-Link et le client TP-Link est détaillée dans la figure et 2.24.

Monitor											
		Throughput		Stations		Interfaces		ARP Table		Routes	
MAC Address	Device Name	Associated SSID	Signal / Noise(dBm) Chain0/Chain1	CCQ (%)	Negotiated Rate (Mbps)	Data TX / RX (kbps)	Distance (km)	IP Address		Connection Time	
54-AF-97-BD-95-2A	CPE610	CPE610_EM	-21/-19,-87/-87	100	300.0	3/2	0.00		0 days 00:02:41		

FIGURE 2.24 – Test de connexion entre le point d'accès TP-Link et le client TP-Link

Enfin, la figure 2.25 met en évidence l’installation réussie du TP-Link CPE610 au toit du Rades Melian, marquant une étape cruciale de notre déploiement.



FIGURE 2.25 – Installation du TP-Link CPE610 à Rades Melian

En suivant ces étapes, nous avons réussi à tester nos TP-Link et nous allons maintenant procéder à leur installation dans les différents bureaux.

2.4.3 Pare-feu

Dans cette section, après avoir configuré notre routeur et notre réseau MAN, nous introduisons également un pare-feu dans notre bureau principal à Rades Meliane, ainsi que dans le bureau des développeurs à Ezzahara à l'avenir.

Choix pfSense

Nous choisissons pfSense pour plusieurs raisons essentielles. Tout d'abord, c'est une solution open source, ce qui signifie qu'elle est gratuite et offre une grande flexibilité. De plus, pfSense est reconnu pour sa stabilité et sa fiabilité, en faisant un choix sûr pour sécuriser notre infrastructure. Il bénéficie d'une communauté active et d'une documentation complète, ce qui facilite son utilisation et sa maintenance. En outre, pfSense offre une gamme étendue de fonctionnalités de sécurité, y compris la gestion des règles de pare-feu, la détection d'intrusion, la protection contre les logiciels

malveillants, etc. En somme, pfSense est la solution idéale pour renforcer la sécurité de notre réseau tout en offrant souplesse et support actif.

Mise en œuvre du pfSense

Pour déployer pfSense en tant que pare-feu dans notre infrastructure du bureau Rades Melian, qui est la branche principale, nous avons suivie les étapes d'installation suivantes.

Nous utilisons la boîte E-Wall Firewall comme hôte pour notre déploiement de pfSense.

Nous procédons à l'installation de pfSense sur la boîte E-Wall Firewall en suivant les étapes décrites ci-dessous dans les figures de 2.26 jusqu'à 2.31.

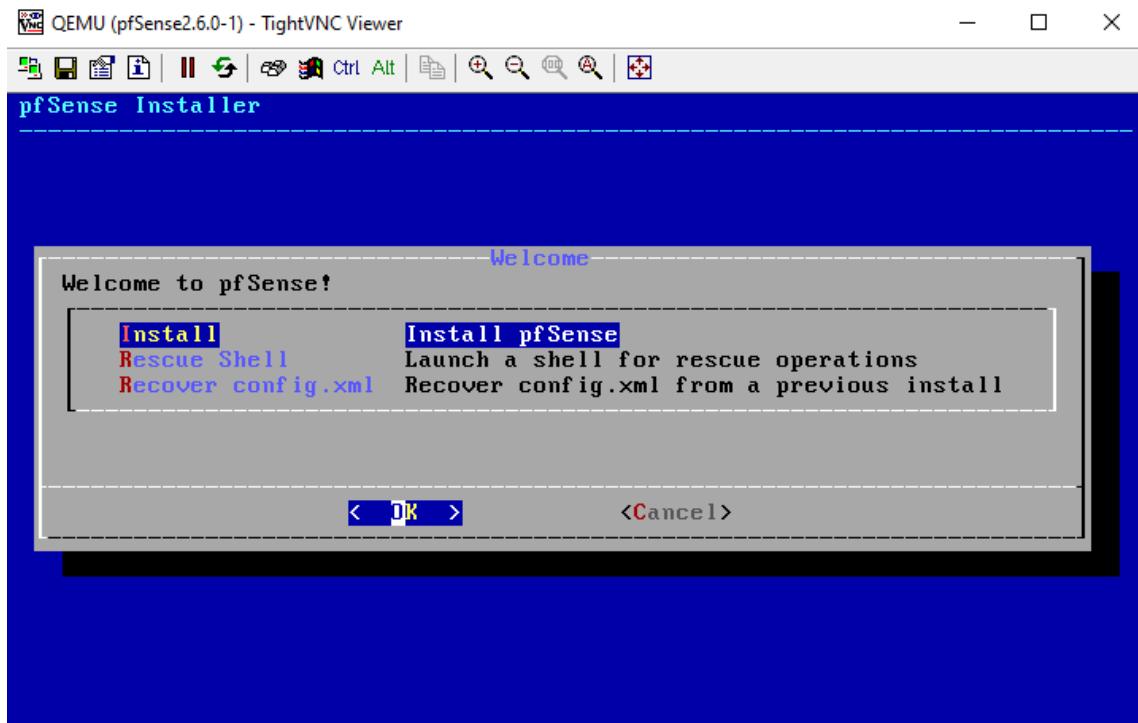


FIGURE 2.26 – Installation de pfSense - Étape 1

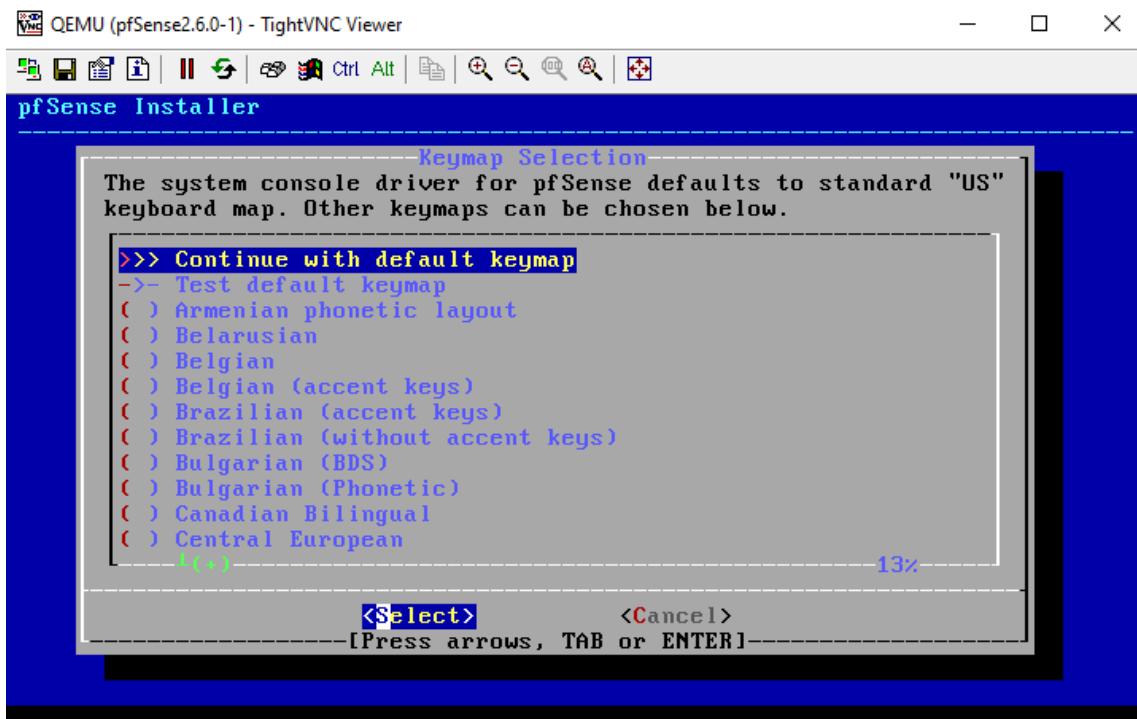


FIGURE 2.27 – Installation de pfSense - Étape 2

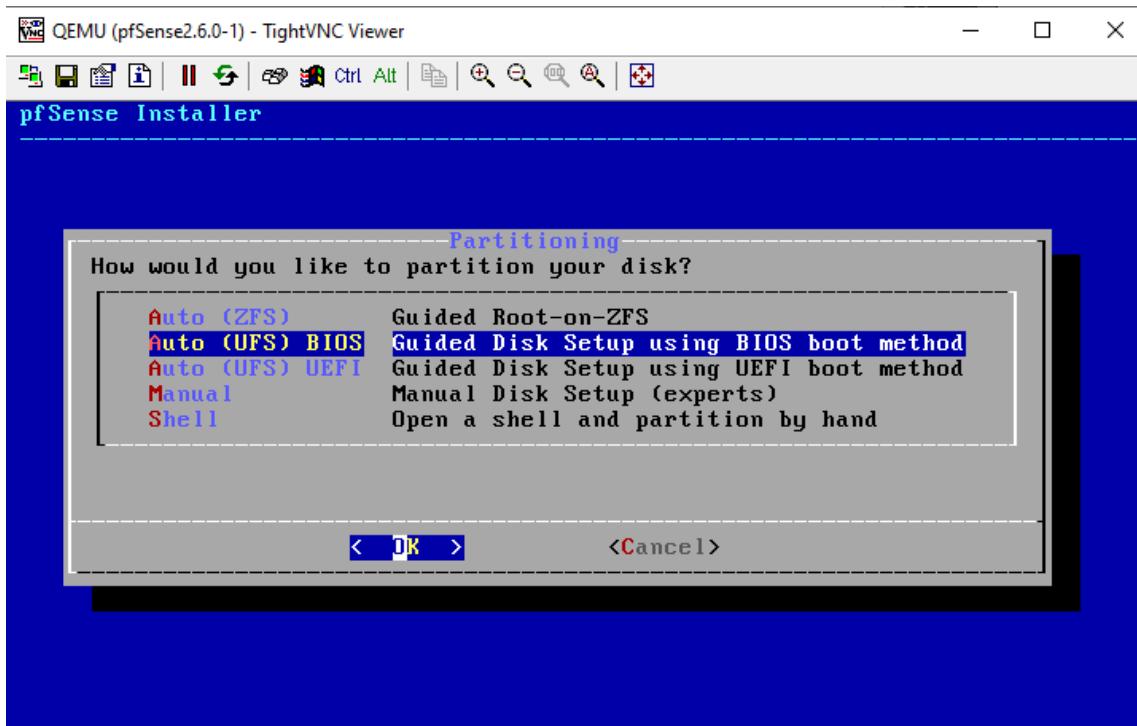


FIGURE 2.28 – Installation de pfSense - Étape 3

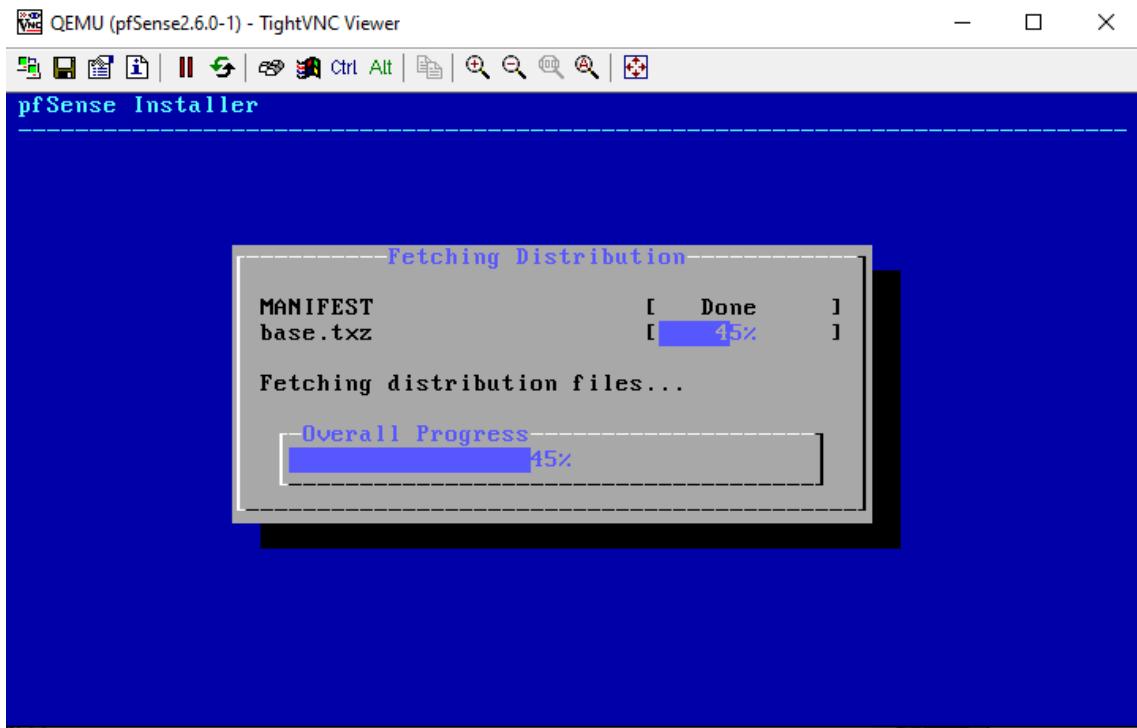


FIGURE 2.29 – Installation de pfSense - Étape 4

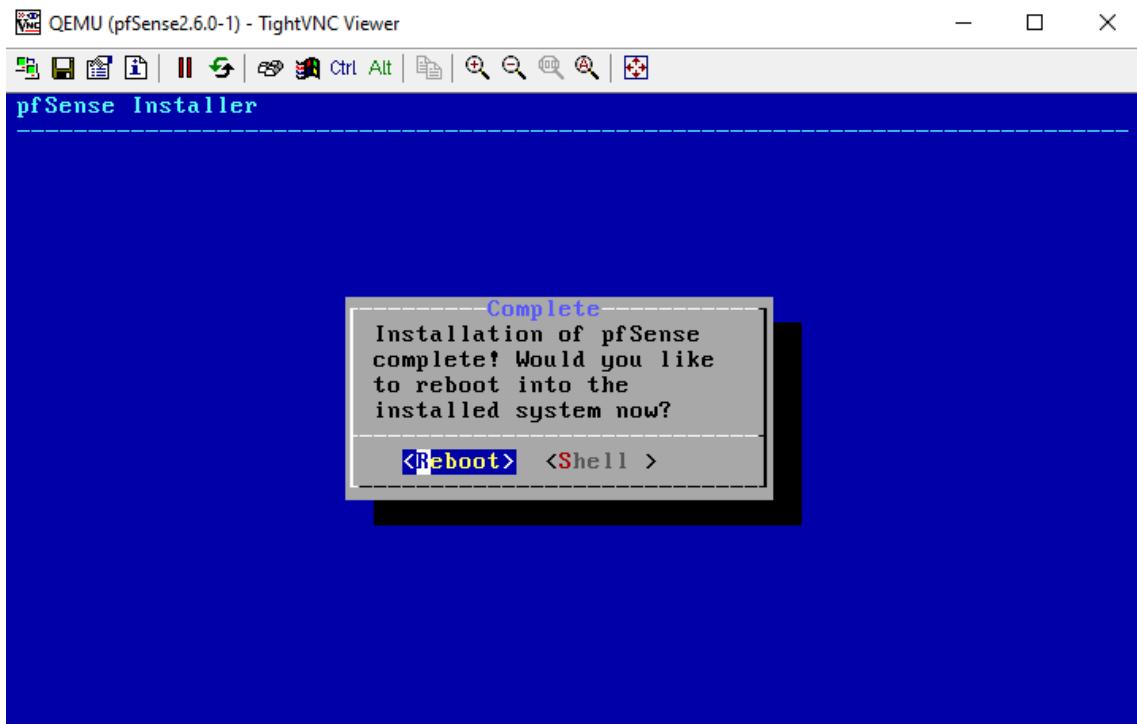


FIGURE 2.30 – Installation de pfSense - Étape 5

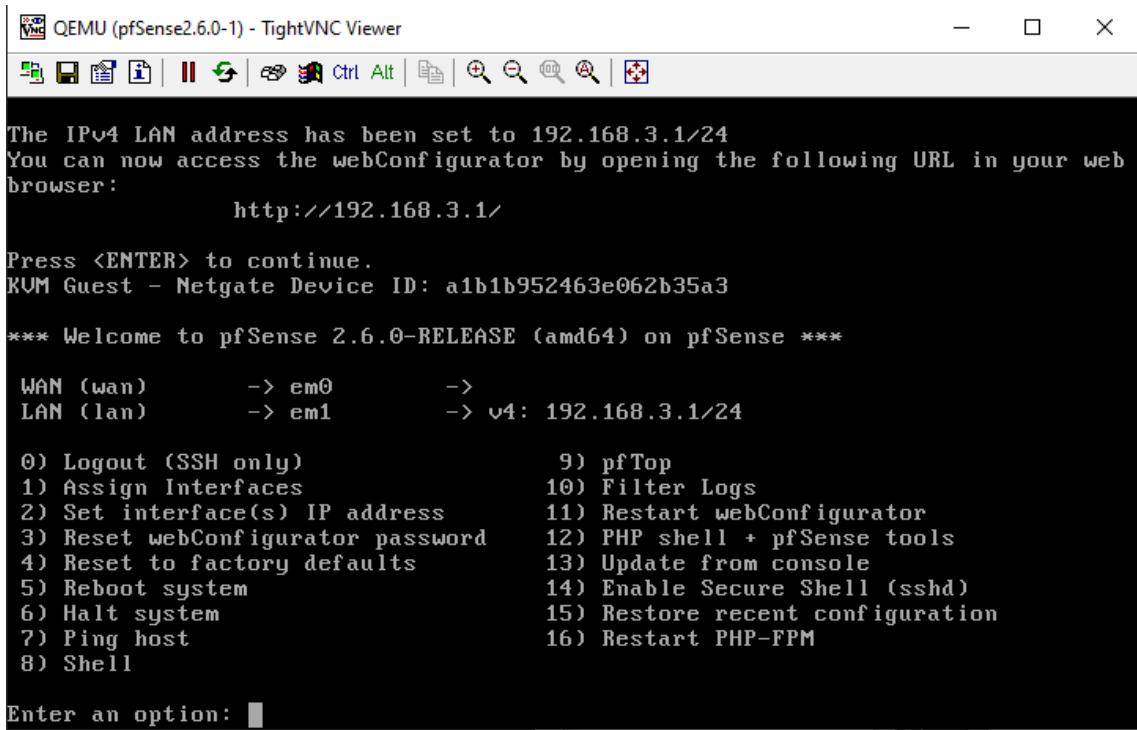


FIGURE 2.31 – Installation de pfSense - Étape 6

FIGURE 2.32 – Tableau de bord de pfSense

Une fois l'installation terminée, nous accédons au tableau de bord de pfSense dans la figure 2.32 pour configurer les interfaces réseau, les règles

de pare-feu et d'autres paramètres de sécurité nécessaires pour protéger notre infrastructure.

pfSense offre une gamme d'options de configuration avancée, notamment la gestion des interfaces, les règles de pare-feu, les services VPN, la surveillance du trafic, etc.

En utilisant ces fonctionnalités, nous pouvons personnaliser notre pare-feu pour répondre à nos exigences spécifiques de sécurité et de réseau.

2.5 Conclusion

En résumé, la création de l'infrastructure informatique pour notre projet a exigé une analyse approfondie des besoins de l'entreprise.

Nous avons soigneusement examiné les différentes architectures réseau, englobant le réseau local, le réseau MAN, et les configurations spécifiques pour les trois bureaux.

Ce chapitre nous a permis de détailler les diverses étapes du déploiement de l'infrastructure IT pour le projet.

Dans le prochain chapitre, nous aborderons le déploiement du système d'information pour compléter notre approche globale visant à répondre aux besoins de l'entreprise avant d'intégrer les objets au sein du ce système d'information.

Chapitre 3

Déploiement du Système d'information

3.1 Introduction

Le déploiement du système d'information est une étape cruciale pour toute entreprise moderne. Afin de répondre aux besoins spécifiques de Zeta Engineering, nous avons identifié les éléments essentiels à intégrer dans notre système d'information, tels que les protocoles de communication et les serveurs.

Dans ce chapitre, nous détaillons les différentes étapes du déploiement de ce système d'information, en commençant par l'installation et la configuration des serveurs nécessaires à leur fonctionnement. Nous expliquons également les choix techniques que nous avons effectués pour répondre aux exigences de l'entreprise et assurer une infrastructure robuste et sécurisée [6].

3.2 Les besoins de notre système d'information

Dans cette section, nous montrons les besoins essentielles de notre entreprise.

Besoin	Description
OS Serveur	Système d'exploitation spécialement conçu pour les serveurs, offrant des fonctionnalités de stabilité, de performance et de compatibilité avec divers logiciels et technologies.
Virtualisation	Permettre l'exécution de systèmes virtuels sur différents systèmes d'exploitation en simulant du matériel standardisé.
Protocole de communication	Ensemble de règles et de conventions permettant à différents composants du réseau de communiquer entre eux de manière cohérente.
ERP/CRM	Plateforme intégrée de gestion d'entreprise qui offre des fonctionnalités pour la gestion des ressources, des ventes, des finances, des ressources humaines et bien plus encore.
Système de téléphonie IP	Gérer les appels téléphoniques, y compris la gestion des lignes et d'autres fonctionnalités avancées. Il prend en charge différents protocoles de communication, permettant ainsi la mise en place de communications vocales à travers le réseau IP.
Gestion de parc et d'inventaires	Suivre et gérer les actifs matériels et logiciels d'une organisation, y compris les utilisateurs, les ordinateurs, les périphériques et les licences logicielles.
Gestion des identités et contrôle d'accès	Gérer les utilisateurs, les ordinateurs et les groupes au sein d'un réseau. Fournir des fonctionnalités d'authentification, de contrôle d'accès et de gestion des politiques de sécurité.

TABLE 3.1 – Les éléments composants le système d'information

3.3 Les solutions choisies

Dans cette section, nous présentons les solutions essentielles que nous avons choisies pour intégrer le système d'information de notre entreprise.

3.3.1 OS Serveur

Nous choisissons Ubuntu Server 22.04 comme système d'exploitation pour nos serveurs en raison de ses performances exceptionnelles, de sa sta-

bilité prouvée et de sa compatibilité avec de nombreux logiciels et technologies. Ubuntu Server est une distribution Linux conçue spécialement pour les serveurs, ce qui en fait un choix optimal pour notre environnement.

La version 22.04 d'Ubuntu Server est la dernière itération de cette distribution basée sur le noyau Linux. Elle offre plusieurs améliorations et fonctionnalités, notamment :

- Des performances accrues : Ubuntu Server 22.04 est optimisé pour offrir des performances élevées, permettant de gérer efficacement même les charges de travail les plus exigeantes.
- Une sécurité renforcée : Cette distribution intègre des fonctionnalités de sécurité avancées, telles que la prise en charge du chiffrement des données, des mises à jour automatiques et la gestion des certificats pour garantir la confidentialité et l'intégrité des données.
- Support à long terme : Ubuntu Server 22.04 bénéficie d'un support à long terme (LTS) de cinq ans, ce qui signifie que des mises à jour de sécurité et des correctifs seront disponibles sur une période prolongée, assurant ainsi la stabilité et la fiabilité du système.
- Écosystème riche : Ubuntu Server est soutenu par une communauté active d'utilisateurs et de développeurs, offrant un soutien abondant, une documentation complète et une large gamme de logiciels et d'outils pour répondre aux besoins spécifiques de nos serveurs.

3.3.2 Virtualisation

Nous choisissons KVM (Kernel-based Virtual Machine) pour la virtualisation de nos serveurs en raison de ses avantages significatifs. KVM est une technologie de virtualisation open source qui permet l'exécution de plusieurs machines virtuelles sur un seul serveur physique. Les avantages de cette solution sont les suivants :

- Efficacité et performances : KVM est intégré au noyau Linux, ce qui lui confère une grande efficacité et des performances élevées. Il tire pleinement parti des ressources matérielles du serveur, assurant ainsi une expérience de virtualisation fluide.
- Isolation et sécurité : KVM garantit une isolation totale entre les machines virtuelles, assurant la sécurité des données et des applica-

tions. Chaque machine virtuelle fonctionne de manière indépendante, réduisant ainsi les risques de failles de sécurité.

- Compatibilité : KVM est compatible avec divers systèmes d'exploitation invités, ce qui permet d'exécuter différents environnements logiciels au sein de machines virtuelles distinctes.
- Évolutivité : La virtualisation avec KVM est hautement évolutive. Nous pouvons ajouter ou supprimer des machines virtuelles selon les besoins de votre entreprise sans perturber les opérations en cours.

3.3.3 Protocole de communication

Pour gérer les adresses IP de notre réseau, nous optons pour ISC DHCP (Internet Systems Consortium - Dynamic Host Configuration Protocol). ISC DHCP est un protocole de communication qui joue un rôle essentiel dans l'attribution automatique des adresses IP aux dispositifs connectés à notre réseau. Voici les raisons de ce choix :

- Fiabilité : ISC DHCP est largement reconnu pour sa fiabilité. Il assure une distribution efficace des adresses IP tout en minimisant les conflits d'adresses et les interruptions de réseau.
- Facilité de gestion : La configuration et la gestion d'ISC DHCP sont conviviales, ce qui simplifie la maintenance de notre réseau. Nous pouvons aisément définir des plages d'adresses, des options de configuration personnalisées et surveiller l'état du service DHCP.
- Prise en charge des protocoles : ISC DHCP prend en charge plusieurs protocoles de communication, notamment IPv4 et IPv6. Cela garantit la compatibilité avec divers dispositifs et systèmes d'exploitation.
- Évolutivité : Notre choix d'ISC DHCP est évolutif, ce qui signifie que nous pouvons facilement adapter notre réseau aux besoins croissants de notre entreprise en ajoutant de nouveaux dispositifs sans soucis de gestion des adresses IP.

Ce protocole de communication joue un rôle crucial dans l'attribution et la gestion des adresses IP au sein de notre réseau, assurant ainsi une connectivité fiable pour tous nos dispositifs.

3.3.4 ERP/CRM

Pour les plates-formes ERP/CRM, nous sélectionnons Odoo. Odoo est une solution intégrée de gestion d'entreprise qui offre une gamme complète de fonctionnalités pour la gestion des ressources, des ventes, des finances, des ressources humaines et bien plus encore. Notre choix d'Odoo repose sur plusieurs avantages clés :

- Intégration complète : Odoo propose une intégration complète de toutes les fonctions essentielles de l'entreprise. Cela signifie que nous pouvons gérer divers aspects de notre entreprise, tels que la comptabilité, les ventes, les stocks et la gestion des projets, à partir d'une seule plate-forme centralisée.
- Personnalisation : Odoo est hautement personnalisable, ce qui nous permet d'adapter la plate-forme à nos besoins spécifiques. Nous pouvons personnaliser les flux de travail, les rapports et les tableaux de bord pour répondre à nos exigences métier uniques.
- Facilité d'utilisation : Odoo est réputé pour sa convivialité. Notre équipe pourra rapidement prendre en main la plate-forme, ce qui réduira la courbe d'apprentissage et favorisera une utilisation optimale.
- Communauté active : Odoo bénéficie d'une communauté mondiale active d'utilisateurs et de développeurs. Cela signifie que nous avons accès à une vaste base de connaissances, à des modules complémentaires personnalisés et à un support communautaire en cas de besoin.

Avec Odoo comme plate-forme ERP/CRM, nous disposons d'un outil puissant pour optimiser nos opérations commerciales, améliorer notre gestion des clients et devenir plus efficaces dans l'ensemble de notre entreprise.

3.3.5 Système de téléphonie IP

Pour notre système de téléphonie IP, notre choix se porte sur Asterisk, une solution de gestion des appels téléphoniques au sein de notre entreprise. Cette décision repose sur plusieurs avantages clés :

- Flexibilité : Asterisk est reconnu pour sa grande souplesse. Il prend en charge divers protocoles de communication, tels que SIP et RTP, nous permettant ainsi d'établir des communications vocales via notre

réseau IP.

- Fonctionnalités avancées : Asterisk propose une vaste gamme de fonctionnalités avancées pour la gestion des appels téléphoniques. Cela englobe la gestion des lignes, la messagerie vocale, les conférences téléphoniques, et bien plus encore.
- Personnalisation : Asterisk est hautement personnalisable. Nous avons la possibilité d'ajuster le système en fonction de nos besoins spécifiques, créant ainsi une solution de téléphonie sur mesure pour notre entreprise.
- Économies de coûts : L'utilisation d'Asterisk se traduit par des économies significatives sur nos frais de communication. La téléphonie IP est généralement plus rentable que les systèmes de téléphonie traditionnels.

En choisissant Asterisk comme solution de téléphonie IP, nous sommes en mesure de mettre en place une infrastructure de communication vocale moderne et adaptable pour répondre aux besoins de notre entreprise.

3.3.6 Gestion de parc et d'inventaires

Pour la gestion de parc et d'inventaires, nous choisissons GLPI avec l'intégration de FusionInventory. Cette solution nous permet de suivre et de gérer efficacement les actifs matériels et logiciels au sein de notre organisation. Notre choix de GLPI et FusionInventory repose sur plusieurs avantages clés :

- Suivi complet des actifs : GLPI nous offre la possibilité de suivre de manière exhaustive tous les actifs de notre parc, y compris les ordinateurs, les périphériques, les logiciels, et bien plus encore. Cela garantit une gestion précise de nos ressources.
- Gestion des utilisateurs : GLPI nous permet de gérer les utilisateurs au sein de notre réseau de manière efficace. Nous pouvons attribuer des actifs aux utilisateurs et suivre leur utilisation de manière transparente.
- Automatisation des tâches : FusionInventory, intégré à GLPI, automatise de nombreuses tâches de gestion, telles que l'inventaire des actifs et la gestion des mises à jour. Cela nous fait gagner du temps et améliore la précision des données.

- Personnalisation : GLPI est hautement personnalisable pour répondre à nos besoins spécifiques. Nous pouvons adapter les champs et les processus pour une gestion des actifs sur mesure.

Avec GLPI et FusionInventory, nous avons la capacité de gérer de manière efficace l'ensemble de nos actifs, de suivre les changements, et d'optimiser l'utilisation de nos ressources.

3.3.7 Gestion des identités et contrôle d'accès

Pour la gestion des identités et le contrôle d'accès, nous avons choisi OpenLDAP en raison de son caractère open source, de sa flexibilité et de sa robustesse. OpenLDAP est une solution de répertoire qui nous permet de gérer les identités des utilisateurs, les groupes et les autorisations d'accès au sein de notre infrastructure informatique. Voici plusieurs raisons convaincantes qui motivent notre choix d'OpenLDAP :

- Gratuit et Open Source : OpenLDAP est une solution open source, ce qui signifie qu'elle est disponible gratuitement. Cela correspond parfaitement à notre souci de maîtriser les coûts tout en bénéficiant de fonctionnalités avancées.
- Personnalisable : OpenLDAP offre une grande souplesse en matière de personnalisation, nous permettant de configurer des schémas et des attributs spécifiques en fonction de nos besoins. Nous pouvons ainsi adapter la structure du répertoire à notre organisation.
- Interopérabilité : OpenLDAP prend en charge les protocoles LDAP standards, facilitant ainsi son intégration avec d'autres applications et services. Nous pouvons l'utiliser pour centraliser l'authentification et l'autorisation sur diverses plateformes.
- Fonctionnalités de sécurité : OpenLDAP propose des fonctionnalités de sécurité robustes, notamment une authentification renforcée, le chiffrement des données et la gestion des politiques d'accès. Cela garantit la protection de nos données sensibles.
- Engagement de la communauté : OpenLDAP reçoit le soutien d'une communauté dynamique d'utilisateurs et de développeurs, ce qui garantit un appui constant, un accès à des ressources utiles et des mises à jour fréquentes.

Avec OpenLDAP, nous mettons en place un système solide et sécurisé de gestion des identités, répondant à nos besoins en matière de contrôle d'accès et d'authentification au sein de notre infrastructure informatique.

3.4 Déploiement des solutions choisies

Nos solutions choisies se répartissent en trois couches distinctes, comme illustré dans la figure 3.1. La couche inférieure représente la couche physique, où nous installons le système d'exploitation serveur. Au-dessus, nous avons la couche de virtualisation, où nous créons des environnements virtuels pour nos applications. Enfin, la couche supérieure est la couche applicative, où nous déployons nos machines virtuelles et exécutons nos applications.

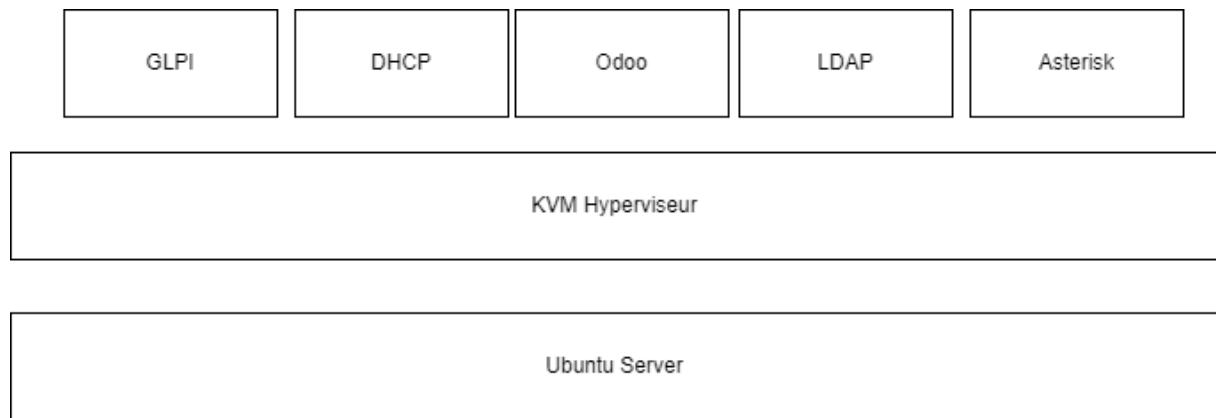


FIGURE 3.1 – Diagramme du système d'information

3.4.1 La couche physique

La configuration de la couche physique de notre infrastructure s'est déroulée en suivant scrupuleusement le guide d'installation d'Ubuntu Server 22.04 [7].



FIGURE 3.2 – Installation de Ubuntu Server 22.04

La figure 3.2 illustre notre procédure d’installation d’Ubuntu Server.



```
zetaadmin@zetasrv: ~
login as: zetaadmin
zetaadmin@192.168.3.234's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-60-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Thu Jun 15 10:21:39 AM UTC 2023

System load: 0.09814453125   Processes:           188
Usage of /:   49.4% of 97.87GB  Users logged in:      1
Memory usage: 45%
Swap usage:   1%
Temperature: 35.0 C          IPv4 address for eno1: 192.168.3.234
                             IPv4 address for virbr0: 192.168.122.1

Expanded Security Maintenance for Applications is not enabled.

8 updates can be applied immediately.
8 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

7 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

*** System restart required ***
Last login: Wed Jun 14 07:55:33 2023 from 192.168.3.227
zetaadmin@zetasrv:~$
```

FIGURE 3.3 – Interface Putty pour la connexion à Ubuntu Server 22.04

Dans la figure 3.3, nous exposons en détail l'interface de Putty, l'outil que nous employons pour effectuer la connexion à notre serveur Ubuntu 22.04. Cette interface est l'interface de contrôle permettant la gestion à distance de notre serveur.

```

5 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

7 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

*** System restart required ***
Last login: Sat Aug 26 15:47:43 2023 from 192.168.3.227
zetaadmin@r720zetasrv:~$ neofetch
    .-/+ooooooooo+-.
     `:+ssssssssssssssssssssssss+`:
     -+ssssssssssssssssssssssyyssss+-+
     .osssssssssssssssssssssdMMMNyssssso.
     /sssssssssshhdmNNmmyNMMMMhssssss/
     +ssssssssshmydMMMMMMNddddyssssssss+
     /ssssssssshNMMyhhyyyhmNMMNhssssssss/
     .ssssssssdMMMNhssssssssssssssssss.
     +sssshhhyNMMNyssssssssssssyNMMMyssssss+
     ossyNMMMNyMMhsssssssssssssshhmmhssssssso
     ossyNMMMNyMMhsssssssssssssshhmmhssssssso
     +sssshhhyNMMNyssssssssssssyNMMMyssssss+
     .ssssssssdMMMNhssssssssssssssss.
     /ssssssssshNMMMyhhyyyhdNMMMNhssssssss/
     +sssssssssdmydMMMMMMMddddyssssssss+
     /sssssssssshhdmNNNNmmyNMMMMhssssss/
     .osssssssssssssssssssssdMMMNyssssso.
     -+ssssssssssssssssssssyyssss+-+
     `:+ssssssssssssssssssssss+-:
     .-/+ooooooooo+-.

                                     zetaadmin@r720zetasrv
                                     -----
OS: Ubuntu 22.04.2 LTS x86_64
Host: PowerEdge R720
Kernel: 6.2.0-26-generic
Uptime: 20 days, 3 hours, 49 mins
Packages: 959 (dpkg), 4 (snap)
Shell: bash 5.1.16
Resolution: 1024x768
Terminal: /dev/pts/0
CPU: Intel Xeon E5-2609 0 (4) @ 2.400GHz
GPU: 0c:00.0 Matrox Electronics Systems Ltd. G200eR2
Memory: 3906MiB / 32005MiB

```

FIGURE 3.4 – Résultat de la commande "neofetch" sur Ubuntu Server 22.04

La figure 3.4 affiche le résultat détaillé de l'exécution de la commande "neofetch" sur notre serveur Ubuntu 22.04. Cette commande nous a permis d'obtenir des informations système essentielles pour assurer le bon fonctionnement de notre infrastructure.

3.4.2 La couche virtualisation

Avant d'installer KVM, nous effectuons des vérifications pour garantir la compatibilité matérielle.

Tout d'abord, nous exécutons la commande `egrep -c '(vmx|svm)' /proc/cpuinfo` pour vérifier la présence des instructions de virtualisation matérielles (VT-x pour Intel ou AMD-V pour AMD) sur le processeur. Un résultat supérieur à zéro indique que le processeur prend en charge la virtualisation matérielle.

Ensuite, nous utilisons la commande `kvm-ok` pour vérifier si les modules de virtualisation étaient correctement chargés dans le système et si la virtualisation était prise en charge.

Une fois ces vérifications réussies, nous procédons à l'installation des

packages nécessaires en exécutant la commande suivante :

```
"sudo apt install -y qemu-kvm virt-manager libvirt-daemon-system virtinst libvirt-clients bridge-utils"
```

- qemu-kvm est l'hyperviseur qui permet l'exécution de machines virtuelles, fournissant un environnement pour les systèmes d'exploitation invités.
- virt-manager est une interface graphique conviviale qui facilite la gestion des machines virtuelles, offrant des fonctionnalités de création, de configuration et de surveillance.
- libvirt-daemon-system inclut les démons de libvirt, essentiels pour la gestion des connexions et des machines virtuelles.
- virtinst est un outil en ligne de commande qui permet la création et la gestion des machines virtuelles de manière efficace.
- libvirt-clients englobe des outils en ligne de commande qui permettent d'interagir avec le démon libvirt pour le contrôle des VM.
- bridge-utils est un package essentiel pour gérer les ponts réseau, permettant ainsi la configuration de la connectivité réseau des machines virtuelles.

Ci-dessous, nous pouvons voir que les figures 3.5 et 3.6 illustrant les étapes d'installation de KVM et la vérification de l'état du service libvirtd

```
zetaadmin@zetasrv:~$ egrep -c '(vmx|svm)' /proc/cpuinfo
16
zetaadmin@zetasrv:~$ kvm-ok
INFO: /dev/kvm exists
KVM acceleration can be used
zetaadmin@zetasrv:~$ sudo apt install -y qemu-kvm virt-manager libvirt-daemon-system virtinst libvirt-clients bridge-utils
```

FIGURE 3.5 – Installation des packages KVM

```

[sudo] password for zetaadmin:
● libvирtd.service - Virtualization daemon
   Loaded: loaded (/lib/systemd/system/libvирtd.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2023-06-12 14:35:23 UTC; 2 days ago
TriggeredBy: • libvирtd-ro.socket
    • libvирtd-admin.socket
    • libvирtd.socket
      Docs: man:libvирtd(8)
           https://libvirt.org
 Main PID: 38246 (libvирtd)
   Tasks: 24 (limit: 32768)
  Memory: 26.2M
    CPU: 4min 24.335s
   CGroup: /system.slice/libvирtd.service
           └─38246 /usr/sbin/libvирtd
               ├─38388 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro --dhcp-script
               ├─38389 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro --dhcp-script

Jun 13 15:52:28 zetasrv libvирtd[38246]: cannot parse process status data
Jun 13 15:52:28 zetasrv libvирtd[38246]: cannot parse process status data
Jun 13 15:52:28 zetasrv libvирtd[38246]: Failed to open file '/sys/fs/cgroup/machine.slice/machine-qemu\x2d12\x2d0doo
Jun 13 15:52:28 zetasrv libvирtd[38246]: Unable to read from '/sys/fs/cgroup/machine.slice/machine-qemu\x2d12\x2d0doo
Jun 13 15:52:28 zetasrv libvирtd[38246]: Failed to open file '/sys/fs/cgroup/machine.slice/machine-qemu\x2d12\x2d0doo
Jun 13 15:52:28 zetasrv libvирtd[38246]: Unable to read from '/sys/fs/cgroup/machine.slice/machine-qemu\x2d12\x2d0doo
Jun 13 15:52:28 zetasrv libvирtd[38246]: cannot parse process status data
Jun 13 15:52:28 zetasrv libvирtd[38246]: cannot parse process status data
Jun 13 21:27:03 zetasrv libvирtd[38246]: End of file while reading data: Input/output error
Jun 14 10:01:35 zetasrv libvирtd[38246]: End of file while reading data: Input/output error
lines 1-27/27 (END)

```

FIGURE 3.6 – Vérification de l'état du service libvирtd

Enfin, grâce à l'utilisation de Xming et de PuTTY, nous pouvons accéder au gestionnaire graphique Virt-Manager à distance depuis notre utilisateur Windows, comme illustré dans la figure 3.7. Cela nous permet de configurer nos machines virtuelles de manière conviviale.

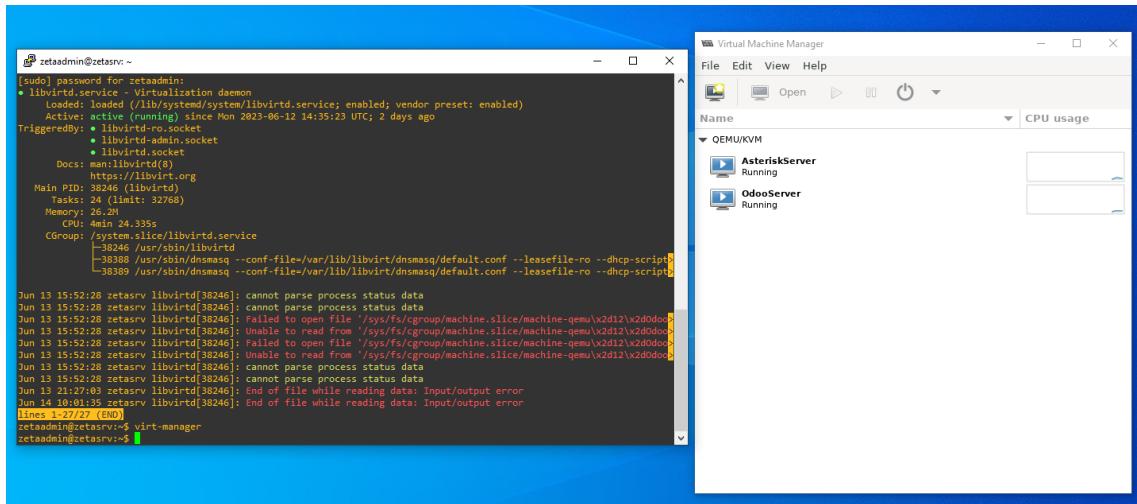


FIGURE 3.7 – Accès à Virt-Manager via Xming et PuTTY

3.4.3 La couche applicative

a. DHCP

Pour mettre en place et configurer un serveur DHCP sur une machine virtuelle Linux, nous avons suivi les étapes suivantes [8].

Tout d'abord, nous effectuons une mise à jour du système Linux en exécutant la commande "**apt-get update**" comme illustré dans la figure 3.8.

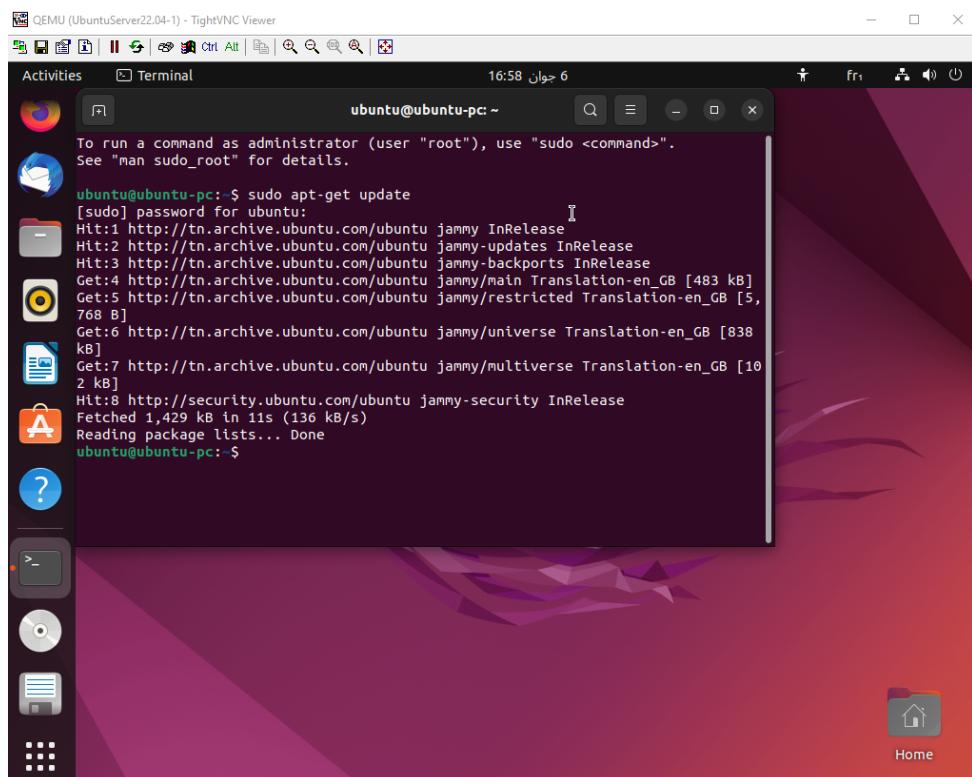


FIGURE 3.8 – Mise à jour du système Linux

Ensuite, nous redémarrons le service isc-dhcp pour appliquer les modifications, comme le montre la figure 3.9.

```

ubuntu@ubuntu-pc:~$ sudo systemctl restart isc-dhcp-server.service
ubuntu@ubuntu-pc:~$ cd /etc/dhcp
ubuntu@ubuntu-pc:/etc/dhcp$ ls -lah
total 44K
drwxr-xr-x  5 root root  4.0K 17:00 6   .   جوان
drwxr-xr-x 129 root root 12K 17:00 6   ..  جوان
drwxr-x---  2 root dhcpd 4.0K 17:00 6   جوان ddns-keys
-rw-r--r--  1 root root 1.4K 2021 9   أوت debug
-rw-r--r--  1 root root 1.7K 2021 9   أوت dhclient.conf
drwxr-xr-x  2 root root 4.0K 04:59 23  فيفي dhclient-enter-hooks.d
drwxr-xr-x  2 root root 4.0K 16:28 6   جوان dhclient-exit-hooks.d
-rw-r--r--  1 root root 3.3K 22:54 31  جانفي dhcpd6.conf
-rw-r--r--  1 root root 3.6K 22:54 31  جانفي dhcpd.conf

```

FIGURE 3.9 – Redémarrage du service isc-dhcp

Nous procémons ensuite à la configuration du fichier dhcpd.conf à l'aide de l'éditeur de texte gedit, comme indiqué dans la figure 3.10.

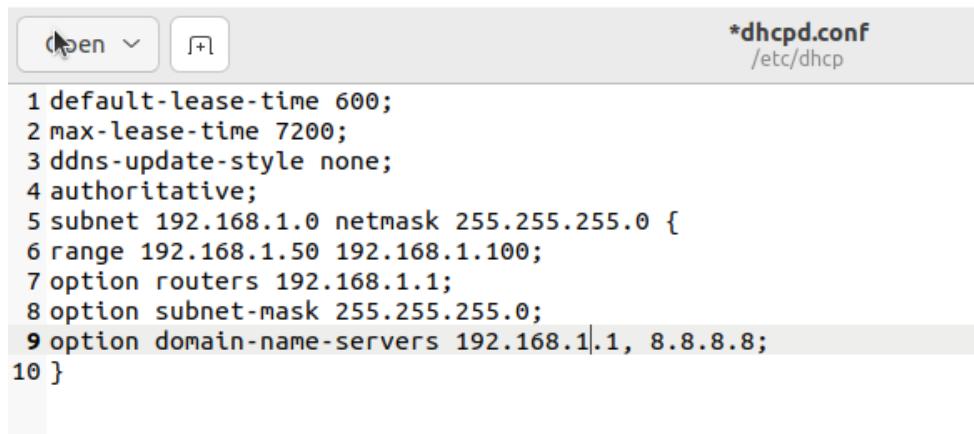
```

ubuntu@ubuntu-pc:/etc/dhcp$ mv -v dhcpd.conf dhcpd.conf.backup
mv: cannot move 'dhcpd.conf' to 'dhcpd.conf.backup': Permission denied
ubuntu@ubuntu-pc:/etc/dhcp$ sudo mv -v dhcpd.conf dhcpd.conf.backup
renamed 'dhcpd.conf' -> 'dhcpd.conf.backup'
ubuntu@ubuntu-pc:/etc/dhcp$ sudo gedit dhcpd.conf

```

FIGURE 3.10 – Configuration du fichier dhcpd.conf

La figure 3.11 illustre un exemple de fichier de configuration dhcpd.conf.



```

*dhcpd.conf
/etc/dhcp

1 default-lease-time 600;
2 max-lease-time 7200;
3 ddns-update-style none;
4 authoritative;
5 subnet 192.168.1.0 netmask 255.255.255.0 {
6 range 192.168.1.50 192.168.1.100;
7 option routers 192.168.1.1;
8 option subnet-mask 255.255.255.0;
9 option domain-name-servers 192.168.1.1, 8.8.8.8;
10 }

```

FIGURE 3.11 – Exemple de fichier de configuration dhcpd.conf

Enfin, nous vérifions que le serveur DHCP ISC était en cours d'exécution en utilisant la commande "systemctl status isc-dhcp-server.service", comme montré dans la figure 3.12.

```

ubuntu@ubuntu-pc:/etc/dhcp$ sudo systemctl restart isc-dhcp-server.service
ubuntu@ubuntu-pc:/etc/dhcp$ sudo systemctl status isc-dhcp-server.service
● isc-dhcp-server.service - ISC DHCP IPv4 server
  Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor
  Active: active (running) since Tue 2023-06-06 17:14:20 CET; 2s ago
    Docs: man:dhcpd(8)
   Main PID: 13200 (dhcpd)
      Tasks: 4 (limit: 4617)
     Memory: 4.5M
        CPU: 16ms
       CGroup: /system.slice/isc-dhcp-server.service
               └─13200 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/…

17:14:20 ٠٦ جولن ubuntu-pc sh[13200]: PID file: /run/dhcp-server/dhcpd.pid
17:14:20 ٠٦ جولن ubuntu-pc dhcpd[13200]: Wrote 0 leases to leases file.
17:14:20 ٠٦ جولن ubuntu-pc sh[13200]: Wrote 0 leases to leases file.
17:14:20 ٠٦ جولن ubuntu-pc dhcpd[13200]: Listening on LPF/ens3/0c:17:2b:dc:00:0>
17:14:20 ٠٦ جولن ubuntu-pc sh[13200]: Listening on LPF/ens3/0c:17:2b:dc:00:00/1>
17:14:20 ٠٦ جولن ubuntu-pc dhcpd[13200]: Sending on   LPF/ens3/0c:17:2b:dc:00:0>
17:14:20 ٠٦ جولن ubuntu-pc sh[13200]: Sending on   LPF/ens3/0c:17:2b:dc:00:00/1>
17:14:20 ٠٦ جولن ubuntu-pc dhcpd[13200]: Sending on   Socket/fallback/fallback->
17:14:20 ٠٦ جولن ubuntu-pc sh[13200]: Sending on   Socket/fallback/fallback-net
17:14:20 ٠٦ جولن ubuntu-pc dhcpd[13200]: Server starting service.

Lines 1-21/21 (END)

```

FIGURE 3.12 – État du service ISC DHCP

Ainsi, nous réussissons à mettre en place et configurer un serveur DHCP sur notre machine virtuelle Linux. Cela nous permettra de fournir automatiquement des adresses IP et d'autres informations de configuration réseau aux clients DHCP de notre réseau.

b. VoIP/Téléphonie

Après avoir suivi les étapes d'installation d'Asterisk [9], comme illustré dans la figure 3.13, notre système Asterisk fonctionne correctement.

```

ubuntu@AsteriskServer:~$ systemctl status asterisk
● asterisk.service - LSB: Asterisk PBX
  Loaded: loaded (/etc/init.d/asterisk; generated)
  Active: active (running) since Tue 2023-06-13 15:24:16 CET; 20h ago
    Docs: man:systemd-sysv-generator(8)
   Tasks: 37 (limit: 4618)
  Memory: 49.4M
  CGroup: /system.slice/asterisk.service
          └─752 /usr/sbin/asterisk -U asterisk -G asterisk

15:24:15 ١٣ جولن ubuntu-VirtualBox systemd[1]: Starting LSB: Asterisk PBX...
15:24:16 ١٣ جولن ubuntu-VirtualBox asterisk[690]: * Starting Asterisk PBX: ast>
15:24:16 ١٣ جولن ubuntu-VirtualBox asterisk[690]: ...done.
15:24:16 ١٣ جولن ubuntu-VirtualBox systemd[1]: Started LSB: Asterisk PBX.

lines 1-13/13 (END)

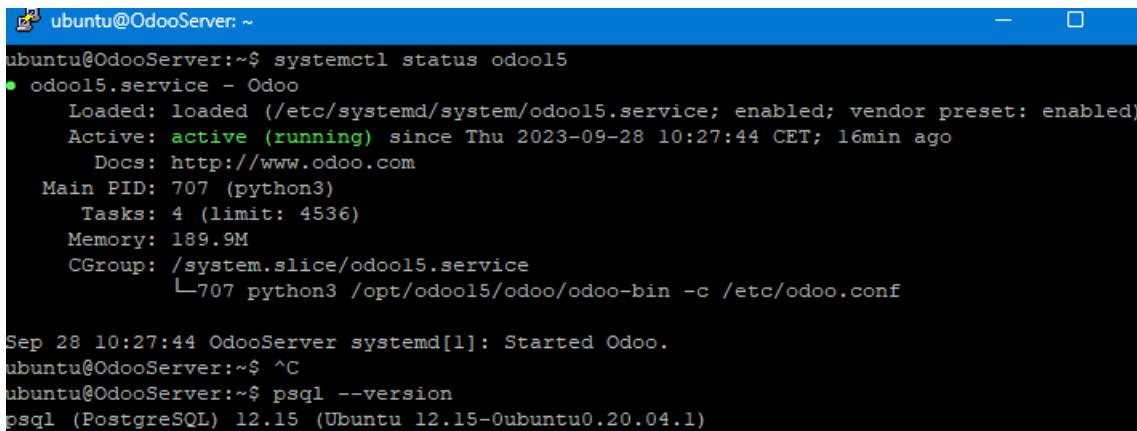
```

FIGURE 3.13 – Installation d'Asterisk sur Ubuntu

c. ERP/CRM

L'installation d'Odoo [10] sur notre machine virtuelle Ubuntu.

Nous poursuivons avec la configuration et la vérification d'Odoo en utilisant PostgreSQL. La figure 3.14 affiche le résultat de la commande "systemctl status odoo15" et "psql --version," confirmant ainsi la compatibilité d'Odoo avec PostgreSQL.



```
ubuntu@OdooServer:~$ systemctl status odoo15
● odoo15.service - Odoo
    Loaded: loaded (/etc/systemd/system/odoo15.service; enabled; vendor preset: enabled)
    Active: active (running) since Thu 2023-09-28 10:27:44 CET; 16min ago
      Docs: http://www.odoo.com
      Main PID: 707 (python3)
         Tasks: 4 (limit: 4536)
        Memory: 189.9M
       CGroup: /system.slice/odoo15.service
               └─707 python3 /opt/odoo15/odoo/odoo-bin -c /etc/odoo.conf

Sep 28 10:27:44 OdooServer systemd[1]: Started Odoo.
ubuntu@OdooServer:~$ ^C
ubuntu@OdooServer:~$ psql --version
psql (PostgreSQL) 12.15 (Ubuntu 12.15-0ubuntu0.20.04.1)
```

FIGURE 3.14 – Configuration d'Odoo et version PostgreSQL

De plus, pour assurer l'intégration réussie avec PostgreSQL, nous présentons la figure 3.15 qui montre la base de données PostgreSQL dans le tableau de bord d'Odoo.

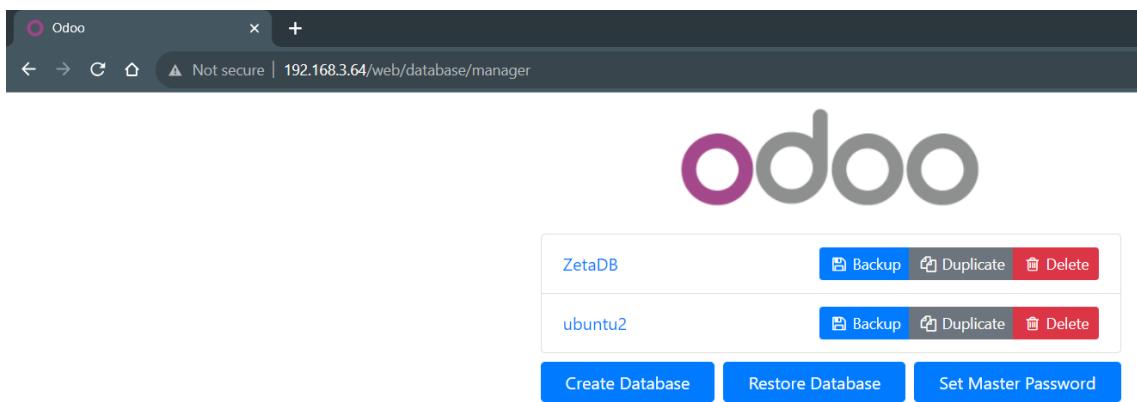


FIGURE 3.15 – Base de données PostgreSQL dans le tableau de bord Odoo

Enfin, pour confirmer le bon fonctionnement d'Odoo et sa compatibilité avec PostgreSQL, nous affichons le tableau de bord d'Odoo avec les modules CRM installés dans la figure 3.16.

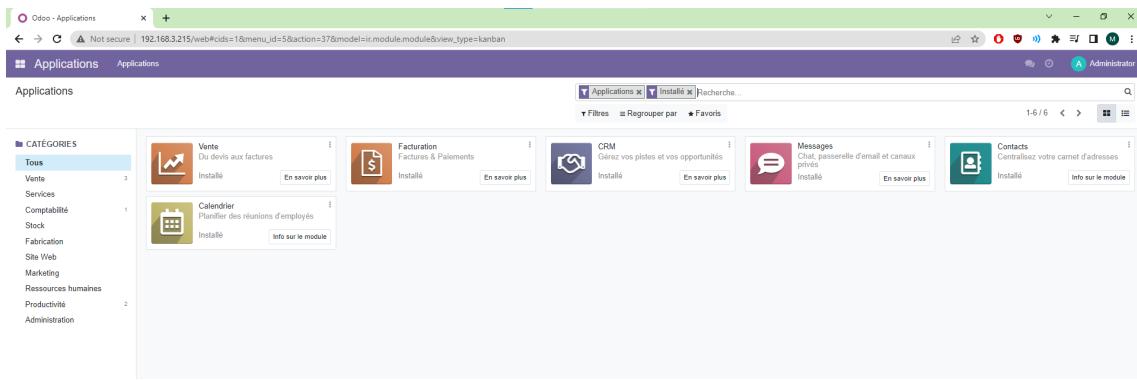


FIGURE 3.16 – Tableau de bord d’Odoo avec les modules CRM installés

d. GLPI

L’installation de GLPI [11] est suivie d’un aperçu du tableau de bord de GLPI, où toutes les informations sur les équipements peuvent être consultées et gérées efficacement, comme montré dans la figure 3.17.

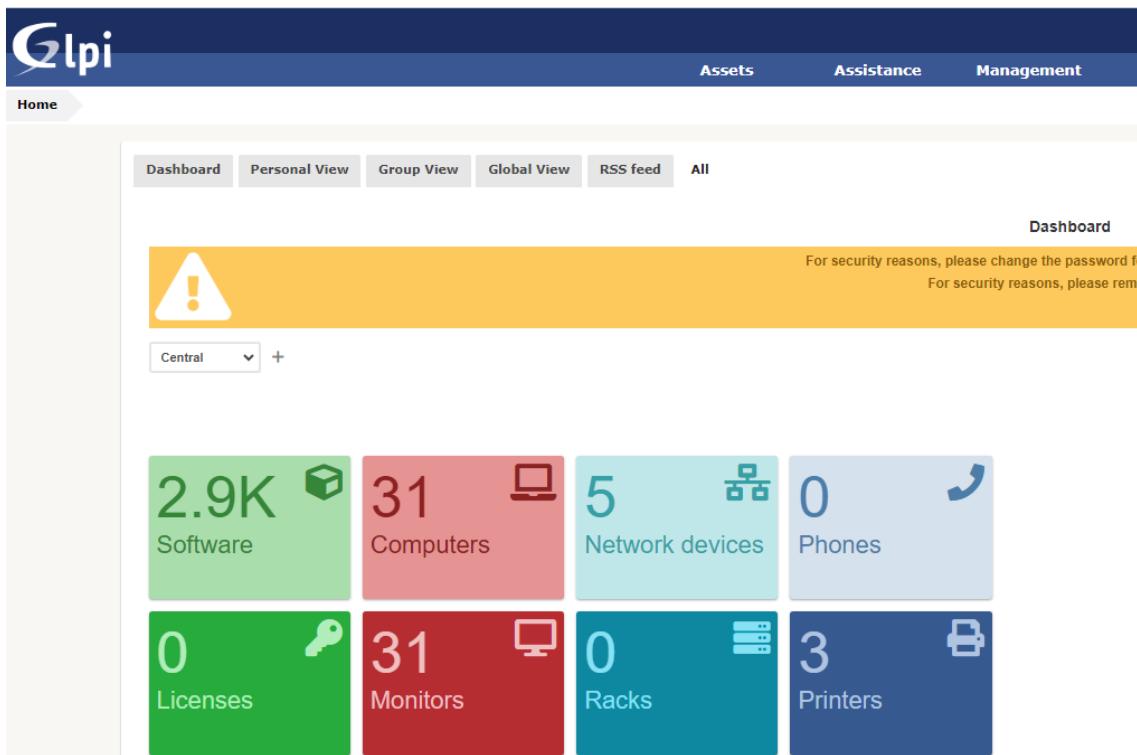


FIGURE 3.17 – Tableau de bord de GLPI

Pour intégrer harmonieusement GLPI dans notre infrastructure, nous ajoutons une base de données MariaDB. MariaDB est un choix judicieux en raison de sa compatibilité et de ses performances élevées. La figure

3.18 témoigne du bon fonctionnement de notre système avec l'affichage de "SHOW TABLES ;". Cette étape garantit que GLPI est prêt à être utilisé dans notre environnement pour gérer les informations sur les équipements de manière efficace.

```
Sep 28 09:53:01 ubuntu-VirtualBox systemd[1]: Started The Apache HTTP Server.  
ubuntu@ubuntu-VirtualBox:~$ mysql -u ocsuser -p  
Enter password:  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 101  
Server version: 10.3.38-MariaDB-Ubuntu0.20.04.1 Ubuntu 20.04  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
MariaDB [(none)]> USE ocsdb;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
MariaDB [ocsdb]> SHOW TABLES;  
+-----+  
| Tables_in_ocsdb |  
+-----+  
| accesslog |  
| accountinfo |  
| accountinfo_config |  
| archive |  
| assets_categories |  
| auth_attempt |  
| batteries |  
| bios |  
| blacklist_macaddresses |  
| blacklist_serials |  
| blacklist_subnet |  
| config |  
| conntrack |  
| controllers |  
| cpus |  
| cve_search |  
| cve_search_computer |  
| cve_search_correspondance |
```

FIGURE 3.18 – Affichage des tables dans MariaDB

e. OpenLDAP

Après avoir suivi les étapes pour installer OpenLDAP [12], nous présentons le tableau de bord de phpLDAPAdmin, une interface graphique pour la gestion de l'annuaire LDAP 3.19.



FIGURE 3.19 – Tableau de bord de phpLDAPAdmin

f. Les machines virtuelles de nos applications

```

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

72 updates can be applied immediately.
5 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

7 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Wed Sep 13 13:05:39 2023 from 192.168.2.120
zetaadmin@r720zetasrv:~$ virsh list --all
  Id   Name           State
  --
 -  AsteriskServer  shut off
 -  DHCPServer      shut off
 -  LDAPServer       shut off
 -  OCSGLPIServer   shut off
 -  OdooServer       shut off

zetaadmin@r720zetasrv:~$ 

```

FIGURE 3.20 – Liste des machines virtuelles déployées

La figure 3.20 présente la liste complète de nos VMs actuellement déployées, chacune d'entre elles étant dédiée à l'une de nos applications. Ce déploiement nous permet de garantir l'isolation des ressources et de maintenir la disponibilité de nos applications de manière efficace.

3.5 Conclusion

Le système d’information de Zeta Engineering est déployé de manière structurée, en utilisant des solutions adaptées : Ubuntu Server 22.04 comme OS serveur, KVM pour la virtualisation, DHCP pour la gestion des IP, Odoo pour la gestion d’entreprise, Asterisk pour les appels, GLPI et FusionInventory pour l’inventaire, et OpenLDAP pour l’identité.

Ce déploiement crée une base solide et sécurisée. L’équipe informatique peut maintenant optimiser le système pour les besoins en constante évolution de l’entreprise.

En conclusion, ce déploiement garantit le bon fonctionnement efficace et fiable de données de Zeta Engineering. Nous passerons maintenant au dernier chapitre, où nous aborderons l’intégration des objets connectés au sein de notre réseau et système d’information.

Chapitre 4

Intégration des objets connectés

4.1 Introduction

L'intégration des objets est un élément clé pour une infrastructure informatique moderne et efficace. Dans ce dernier chapitre, nous allons explorer les différentes étapes de l'intégration des objets connectés dans notre infrastructure existante.

Dans ce chapitre, nous examinons les différentes technologies et protocoles utilisés pour l'intégration des objets connectés, ainsi que les défis qu'elle présente.

Enfin, nous décrivons également les différentes étapes du processus d'intégration.

4.2 Définition des objets connectés

Les objets connectés, également connus sous le nom d'internet des objets (IoT), font référence à des appareils physiques qui sont connectés à Internet et peuvent échanger des données avec d'autres appareils ou systèmes [13].

Les objets connectés sont souvent équipés de capteurs, de logiciels et d'autres technologies permettant de collecter et de communiquer des données en temps réel. Ces données peuvent être utilisées pour améliorer

l'efficacité, la productivité et la sécurité dans un large éventail de domaines, tels que la santé, les transports, l'agriculture, l'industrie manufacturière, les villes intelligentes, les maisons intelligentes, etc.

La figure 4.1 est un exemple de topologie IoT extraite de l'article "Transfert de données IoT basé sur CoAP d'un Raspberry Pi vers le Cloud" [14]. Elle présente la topologie d'un système conçu par Thomas Scott et Amna Eleyan.

Ce système se compose de quatre éléments principaux : le capteur, le Raspberry Pi, une passerelle basée sur le protocole CoAP (Constrained Application Protocol) et la plateforme cloud. Le capteur collecte les données et les transmet au Raspberry Pi. Le Raspberry Pi est ensuite responsable de la manipulation des données dans un format adapté pour la transmission via le protocole CoAP. L'implémentation du protocole CoAP communique avec la plateforme cloud, laquelle stocke les données, permettant ainsi l'accès aux utilisateurs.

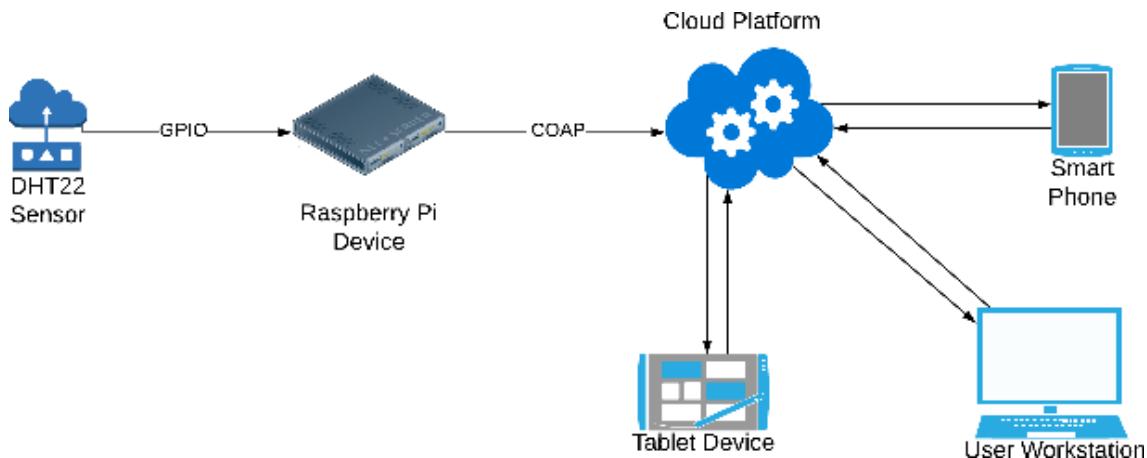


FIGURE 4.1 – Architecture IoT basée sur CoAP

L'intégration des objets connectés (IoT) dans les foyers a connu une croissance significative ces dernières années. Selon Statista, en 2020, le nombre d'appareils IoT connectés dans le monde a atteint 26,66 milliards, et ce chiffre devrait augmenter à 75,44 milliards d'ici 2025 [15]. Les dispositifs IoT incluent une variété d'objets connectés tels que les caméras de surveillance, les machines de pointage, et les lampes intelligentes (smart blubs).

Les smart blubs sont l'une des applications les plus courantes de l'IoT dans les foyers. Selon un rapport de Grand View Research, le marché des

lampes intelligentes devrait atteindre une valeur de 38,68 milliards de dollars d'ici 2027 [16], avec un taux de croissance annuel composé de 20,8%. Bien que la plupart des lampes ne soient pas des smart blubs, elles peuvent être connectées à notre réseau grâce à des cartes IoT comme Raspberry Pi ou Arduino.

4.3 Intégration des dispositifs de l'entreprise

Dans cette section, nous explorons l'intégration des dispositifs au sein de notre entreprise. Nous discutons des défis que nous rencontrons dans cette démarche et des solutions que nous envisageons pour les surmonter.

4.3.1 Enjeux de l'intégration

L'intégration des dispositifs d'entreprise constitue un défi essentiel à relever pour améliorer notre infrastructure. Nous devons connecter divers dispositifs qui étaient auparavant isolés au réseau de l'entreprise. Les principaux dispositifs que nous traitons sont :

- Caméras de surveillance
- Système de pointage
- Lampes d'éclairage
- Capteur de température et d'humidité

Chacun de ces dispositifs présente des exigences spécifiques en termes d'intégration et de connectivité. Leur incorporation dans notre réseau doit être soigneusement planifiée pour garantir une opération harmonieuse.

4.3.2 Intégration des caméras de surveillance

Notre première étape consiste à connecter nos caméras de surveillance au réseau de l'entreprise. Pour ce faire, nous configurons chaque caméra pour lui attribuer une adresse IP unique, lui permettant ainsi d'être reconnue sur le réseau.

Parallèlement, nous allons utiliser un NVR (Network Video Recorder)

pour stocker en continu les enregistrements vidéo et gérer les flux issus des caméras. De plus, nous installons le logiciel SmartPSS sur les ordinateurs des directeurs pour faciliter l'accès aux flux vidéo des caméras.



FIGURE 4.2 – Logiciel SmartPSS installé

SmartPSS 4.2 est une solution logicielle développée par Dahua, une entreprise leader dans le domaine des systèmes de vidéosurveillance. Il offre une gestion centralisée des caméras, ainsi que des fonctionnalités d'enregistrement et de lecture des vidéos enregistrées.

4.3.3 Intégration des systèmes de pointage

L'intégration des systèmes de pointage est une autre composante cruciale de notre initiative. Nous utilisons le logiciel ZKTime 4.3, développé par ZKTeco, pour connecter et contrôler ces systèmes au sein de notre réseau d'entreprise.

Avec ce logiciel nous assurons une communication fluide entre les machines de pointage et le réseau, garantissant ainsi un accès facile et sécurisé aux données de pointage des employés.

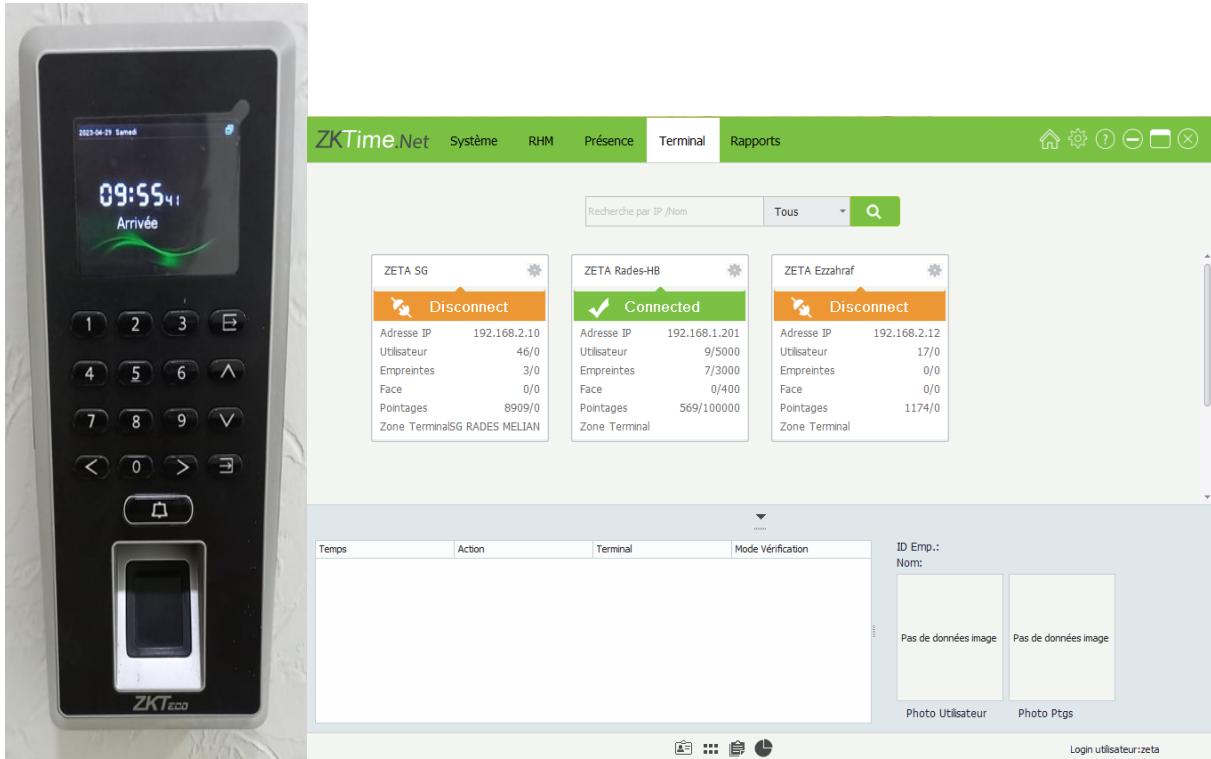


FIGURE 4.3 – Machine de pointage ZKTeco et logiciel ZKTime

4.3.4 Gestion intelligente de l'éclairage

Pour répondre à la demande croissante d'un SmartOffice, ce qui nécessite la mise en place d'un système de contrôle à distance des lampes, nous utilisons une carte électronique, telle que Raspberry Pi ou Arduino, pour connecter les lampes à notre réseau d'entreprise.

Cette configuration nous permet une gestion centralisée de l'éclairage, contribuant ainsi à optimiser l'utilisation de l'énergie et à réaliser des économies considérables sur les coûts énergétiques. En fin de compte, l'intégration de dispositifs connectés, tels que des lampes intelligentes, dans notre infrastructure d'entreprise, peut considérablement améliorer l'efficacité opérationnelle et la qualité de vie de nos employés.

4.3.5 Gestion intelligente du capteur de température et humidité

De manière similaire à notre approche pour l'éclairage, nous cherchons à mettre en place une gestion intelligente de notre capteur de température et humidité modèle DHT11. Cela nous permettra de surveiller et de contrôler les conditions thermiques de nos locaux de manière efficace.

4.3.6 Solution proposée

Pour relever ces défis d'intégration et connecter nos dispositifs non-connectés, tels que les lampes et le capteur de température et humidité, à notre réseau, nous proposons d'utiliser une approche basée sur Raspberry Pi en combinaison avec Node-RED.

Cette solution offre une méthode cohérente et efficace pour intégrer ces dispositifs dans notre infrastructure existante, tout en fournissant une interface de contrôle conviviale pour la gestion et le suivi de ces dispositifs au sein de notre entreprise.

4.4 Solution avec Raspberry Pi et Node-RED

Dans cette section, nous explorons en détail la solution que nous proposons pour l'intégration de dispositifs au sein de notre entreprise en utilisant Raspberry Pi et Node-RED.

Cette approche offre une méthode cohérente et efficace pour connecter nos dispositifs qui ne sont pas nativement connectés, notamment les lampes et le capteur de température, à notre réseau tout en fournissant une interface de contrôle conviviale pour la gestion et le suivi de ces dispositifs au sein de notre entreprise.

4.4.1 Matériels utilisés pour notre prototype

Pour créer notre prototype, nous commençons par présenter les matériaux utilisés, comme indiqué dans le tableau 4.1. Ce prototype sera ultérieurement mis en œuvre sur le bureau principal, puis sur les deux autres bureaux respectivement.

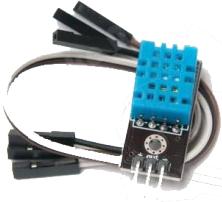
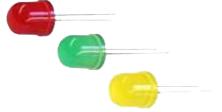
Image de l'équipement	Description de l'équipement
	Raspberry Pi B+ V1.2 : Un ordinateur monocomposant (single-board computer) de la taille d'une carte de crédit, développé par la fondation Raspberry Pi. Il est équipé d'un processeur ARM Cortex-A53 quad-core à 1,4 GHz, de 1 Go de RAM, de plusieurs ports USB, d'un port Ethernet, de connecteurs audio et vidéo, et d'un slot pour carte microSD pour le stockage. Il peut être utilisé pour une grande variété de projets, notamment des serveurs, des media centers, des stations de développement, des robots, des systèmes embarqués, etc.
	Capteur de température et d'humidité DHT11 : Un composant électronique capable de mesurer la température et l'humidité de son environnement. Il fournit des données essentielles pour notre système de contrôle environnemental.
	Diode LED : Une diode électroluminescente utilisée pour simuler le fonctionnement de nos lampes. Elle permet de tester notre système de contrôle d'éclairage.

TABLE 4.1 – Les équipements composants de notre prototype miniature

4.4.2 Choix de Raspberry Pi et Node-RED

Face à ce défi, notre choix se porte sur l'utilisation de Raspberry Pi pour connecter les objets à notre réseau. Raspberry Pi est une carte informatique très performante qui peut être utilisée pour de nombreux projets IoT [17].

Nous décidons d'utiliser Node-RED, un outil de programmation visuel pour créer une application permettant de contrôler les lampes à distance.

Node-RED est une plateforme open source qui facilite la création de flux de données entre différents objets connectés [18].

4.4.3 Installation Raspberry Pi OS et Node-RED

a. Installation de l'OS Raspberry Pi

Pour débuter, nous installons l'OS Debian spécifique au Raspberry Pi sur une carte mémoire, que nous insérons ensuite dans notre carte électronique Raspberry Pi pour démarrer le système.

Après avoir téléchargé le Raspberry Pi Imager depuis le site officiel, nous utilisons cet outil pour graver l'image sur la carte mémoire. Cette opération permet ensuite d'insérer la carte mémoire dans le slot prévu sur notre carte électronique Raspberry Pi, nous donnant ainsi la possibilité de démarrer le Raspberry Pi et d'accéder à l'interface utilisateur du système d'exploitation.

Une fois le Raspberry Pi OS démarré, nous pouvons procéder aux configurations supplémentaires nécessaires pour intégrer les objets connectés à notre infrastructure. Ces configurations incluent potentiellement la connexion au réseau, l'installation de logiciels supplémentaires, et la personnalisation des paramètres pour répondre à nos besoins spécifiques.

En résumé, l'installation du Raspberry Pi OS sur une carte mémoire et son insertion dans notre carte électronique Raspberry Pi marquent la première étape pour mettre en place notre infrastructure et intégrer les objets connectés. Cette phase est cruciale pour assurer la compatibilité et la stabilité de notre système.

b. Installation de Node-RED

Après avoir préparé le Raspberry Pi, nous procédons à l'installation de Node-RED sur la carte à l'aide du script bash officiel disponible sur le site de Node-RED, comme illustré dans la figure 4.4.

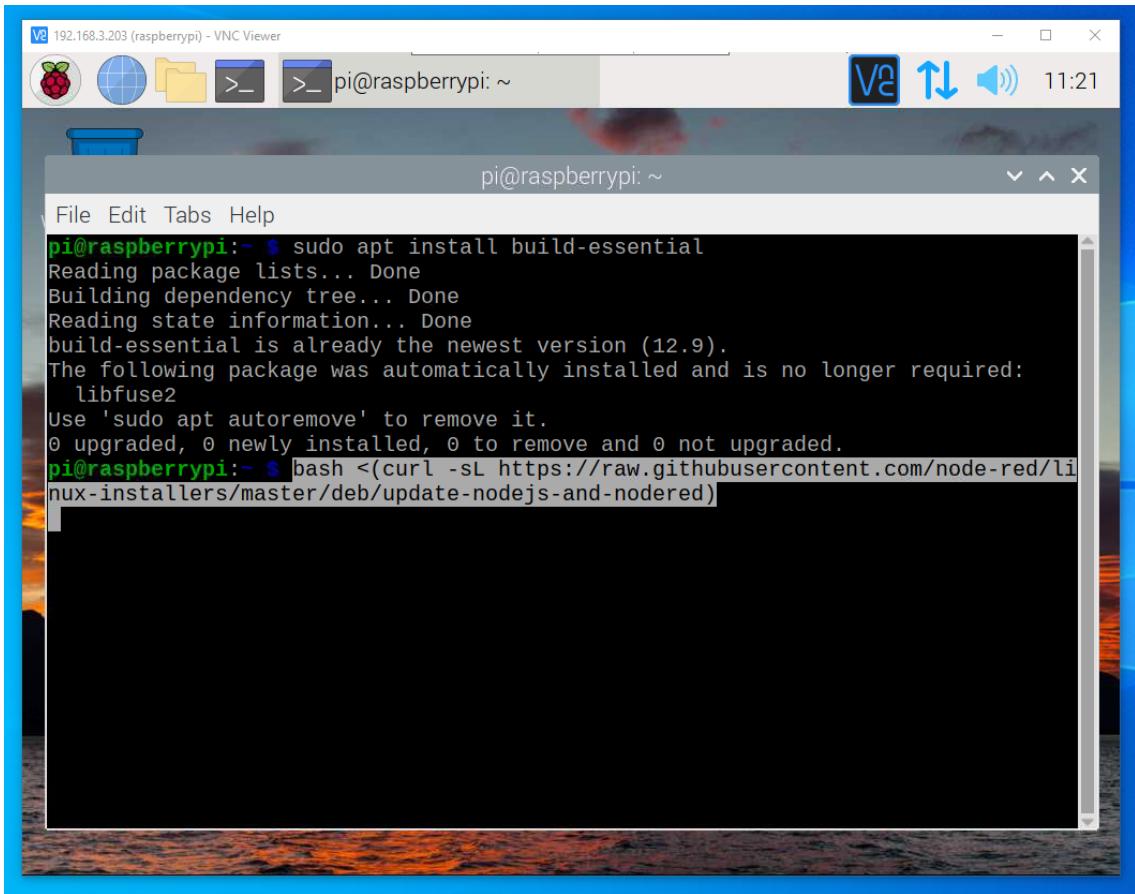


FIGURE 4.4 – Installation du Node-RED avec le Script Officiel Bash

Ce script effectue une série d'étapes, telles que l'arrêt de Node-RED, la suppression des anciennes versions de Node-RED et de Node.js, l'installation de Node.js pour Armv6, la purification du cache npm, l'installation de Node-RED Core, le déplacement des noeuds globaux vers un emplacement local, la reconstruction des noeuds existants, l'installation de noeuds supplémentaires spécifiques à Raspberry Pi, l'ajout de commandes raccourcies et la mise à jour du script systemd comme la montre la figure 4.5.

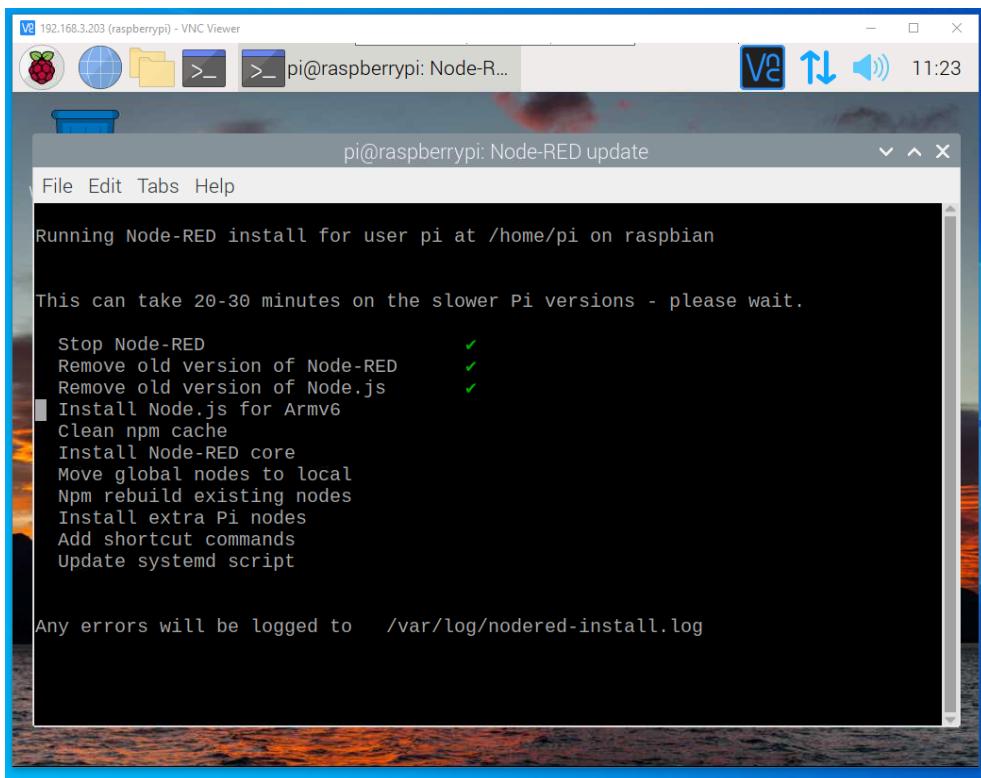


FIGURE 4.5 – En cours d'Installation du Node-RED

Une fois l'installation terminée, nous pouvons démarrer Node-RED en utilisant la commande "node-red-start". Le résultat final du démarrage est illustré dans la figure 4.6.

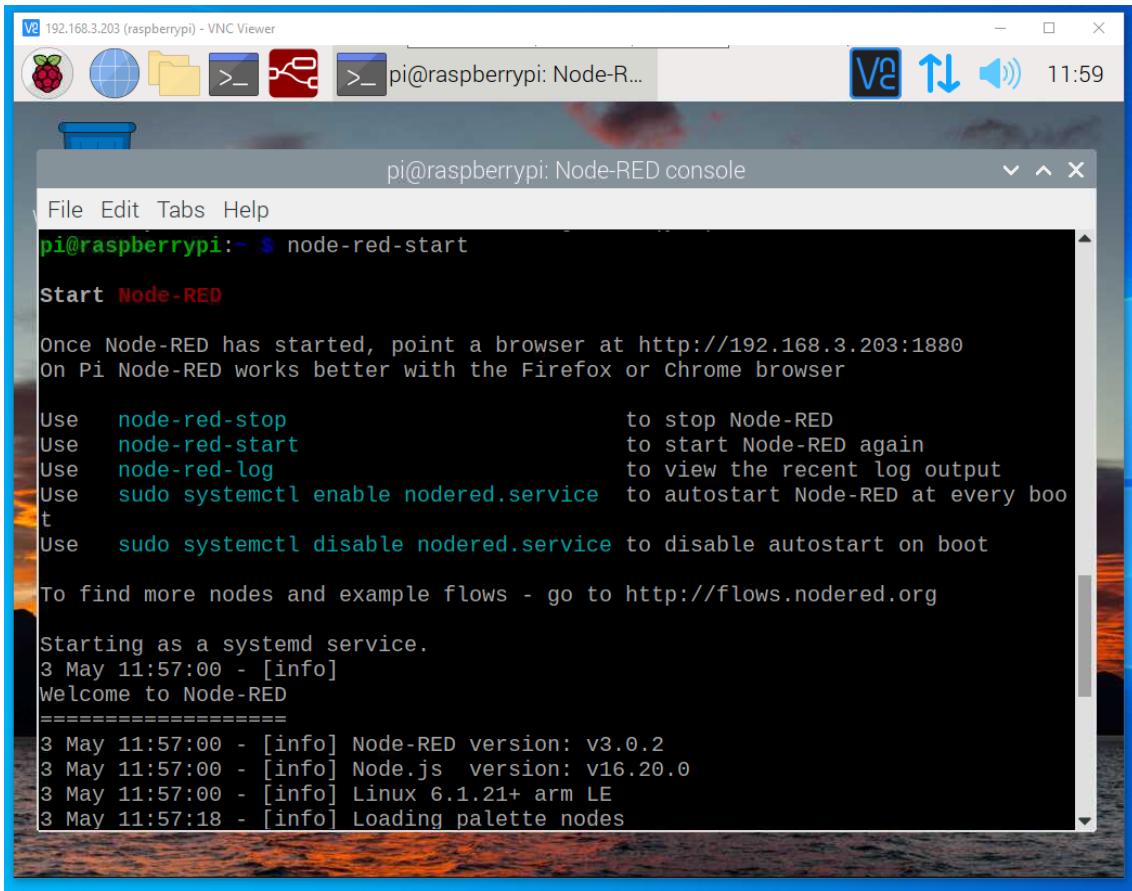


FIGURE 4.6 – Le démarrage du Node-RED

Node-RED est maintenant installé et opérationnel sur notre Raspberry Pi, prêt à être utilisé pour créer des flux et interagir avec les objets connectés de notre infrastructure.

4.4.4 Configuration de Node-RED

Pour accéder à Node-RED via le réseau, nous utilisons l'adresse IP de notre Raspberry Pi et le port 1880/ui.

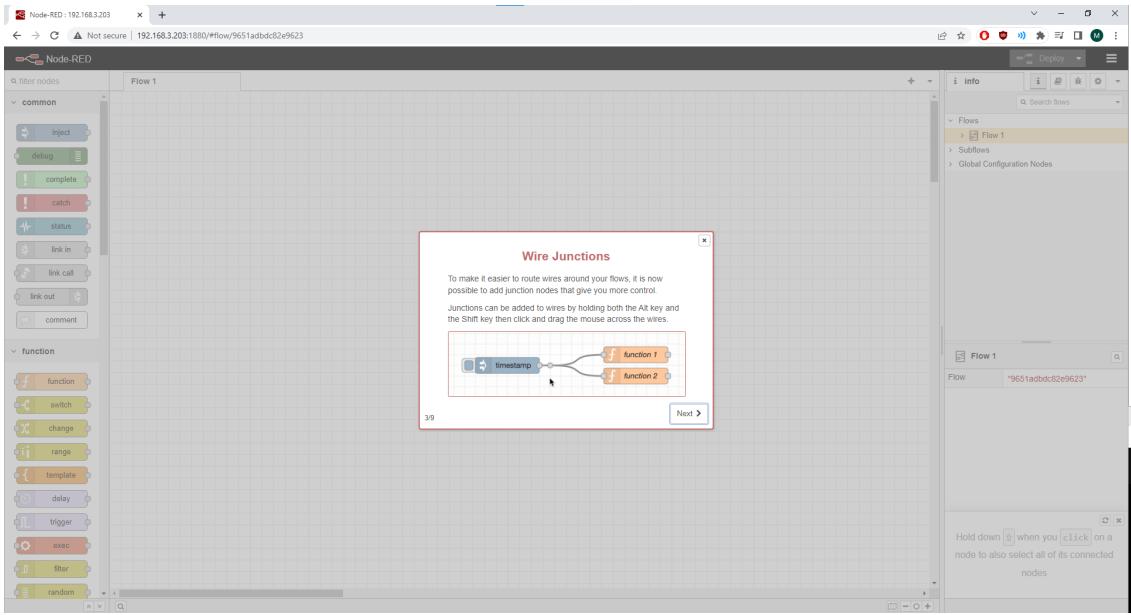


FIGURE 4.7 – Interface Node-RED

Dans la figure 4.7, nous observons l’interface de Node-RED, où nous développons notre projet en utilisant des nœuds et des connexions pour créer des flux de données.

Pour notre projet, nous devons installer quelques nœuds supplémentaires. Par exemple, le "dashboard" pour créer une interface graphique conviviale permettant d’interagir avec le projet, et le nœud DHT11 pour détecter notre capteur de température dans Node-RED.

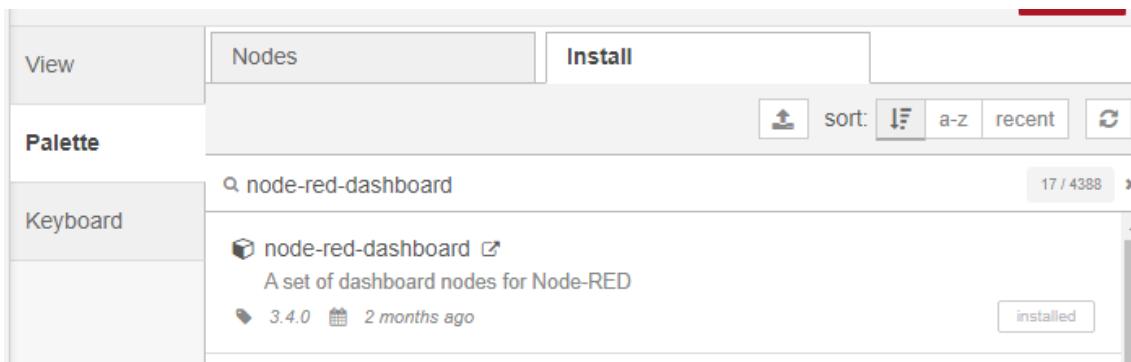


FIGURE 4.8 – Installation du dashboard Node-RED

La figure 4.8 montre le processus d’installation du nœud "dashboard" dans Node-RED, qui permet de créer des tableaux de bord interactifs pour contrôler et visualiser les données de notre projet.

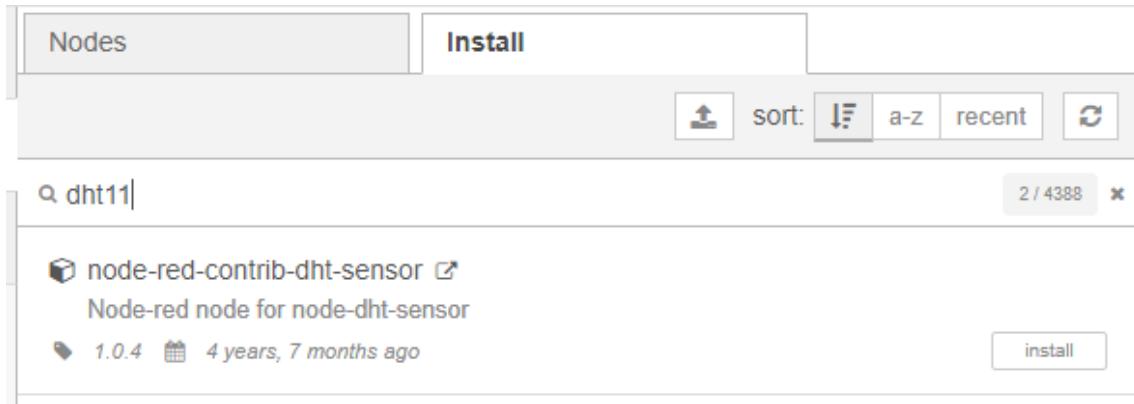


FIGURE 4.9 – Installation du nœud DHT11 dans Node-RED

Dans la figure 4.9, nous installons le nœud DHT11 dans Node-RED, qui nous permet de capturer les données de notre capteur de température et d’humidité.

Ces différentes figures illustrent la préparation de Node-RED et l’installation des nœuds nécessaires pour notre projet. Elles servent de base à la réalisation de notre système de surveillance et de contrôle.

4.4.5 Sécurité de Node-RED

Pour sécuriser Node-RED, nous utilisons une méthode fournie par le site officiel de Node-RED, qui nous permet de créer des mots de passe hachés et de les ajouter à la configuration de notre projet. Les figures suivantes illustrent ce processus.

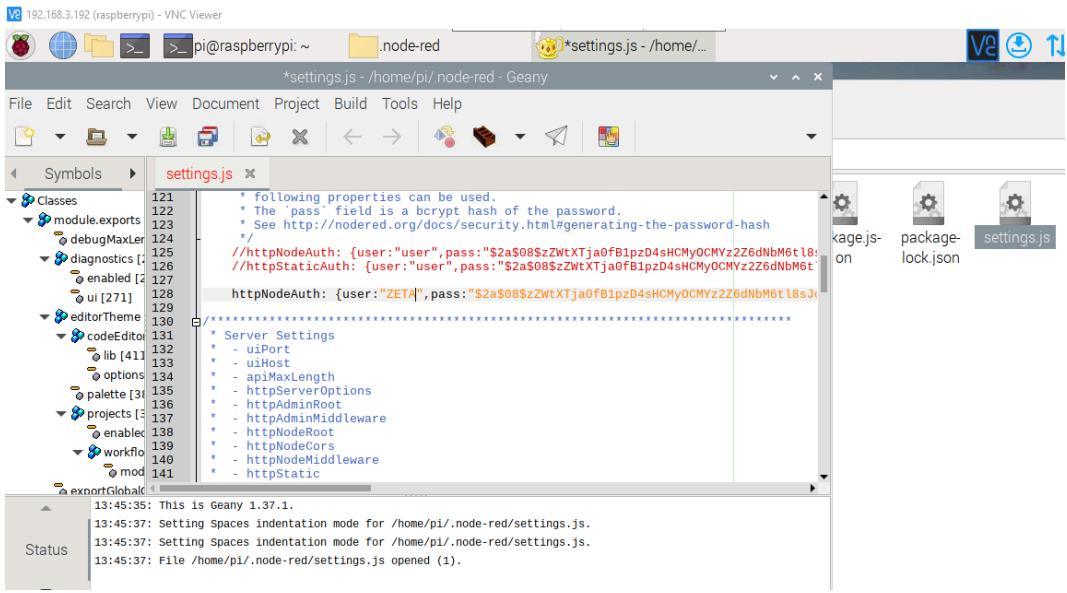


FIGURE 4.10 – Configuration de la sécurité dans Node-RED en Settings.js

Dans la figure 4.10, nous modifions le fichier de configuration "Settings.js" de Node-RED pour inclure des paramètres de sécurité. Cela permet de définir des utilisateurs et des mots de passe pour restreindre l'accès à notre projet.

Nous utilisons la commande "node-red admin hash-pw" pour générer un code de hachage sécurisé, comme le montre la figure 4.11.

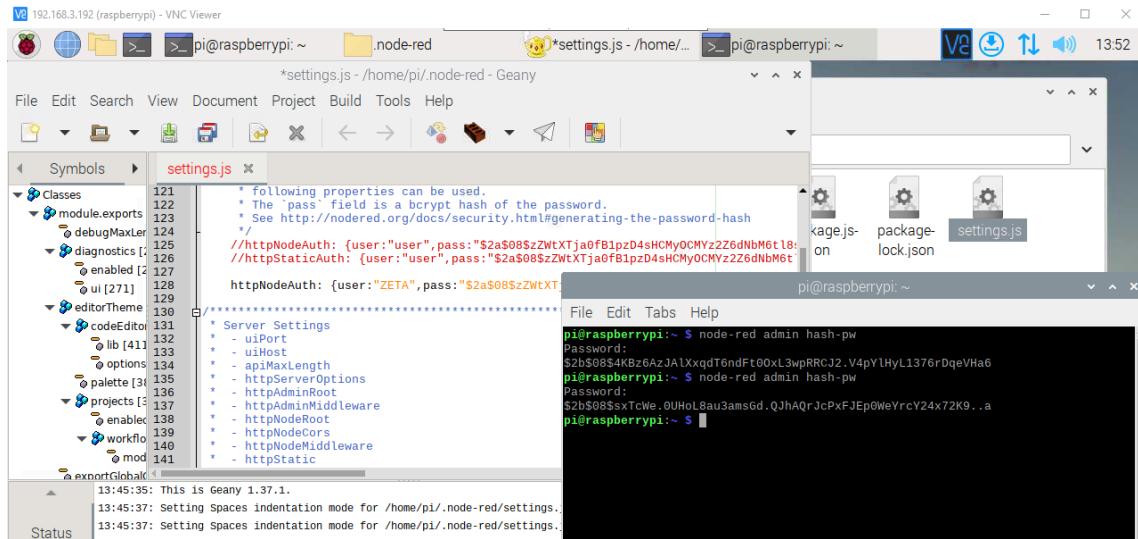
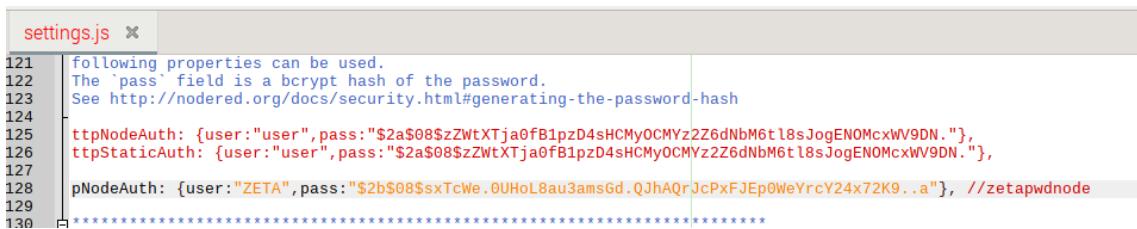


FIGURE 4.11 – Crédit d'un code de hachage

Le code de hachage généré est ensuite ajouté au fichier "Settings.js" pour mettre à jour les informations de sécurité, comme indiqué dans la

figure 4.12.



```
settings.js x
121  following properties can be used.
122  The 'pass' field is a bcrypt hash of the password.
123  See http://nodered.org/docs/security.html#generating-the-password-hash
124
125  ttpNodeAuth: {user:"user",pass:"$2a$08$zWtXTja0fB1pzD4sHCMY0CMYz2Z6dNbM6tl8sJogENOMcxWV9DN."},
126  ttpStaticAuth: {user:"user",pass:"$2a$08$zWtXTja0fB1pzD4sHCMY0CMYz2Z6dNbM6tl8sJogENOMcxWV9DN."},
127
128  pNodeAuth: {user:"ZETA",pass:"$2b$08$sxTcWe.OUHoL8au3amsGd.QjhAqrJcPxFJEp0WeYrcY24x72K9..a"}, //zetapwdnode
129
130 *****
```

FIGURE 4.12 – Mise à jour du code de hachage dans Settings.js

Une fois la sécurité configurée, nous pouvons nous connecter à Node-RED en utilisant le nouveau mot de passe, comme le montre la figure 4.13.

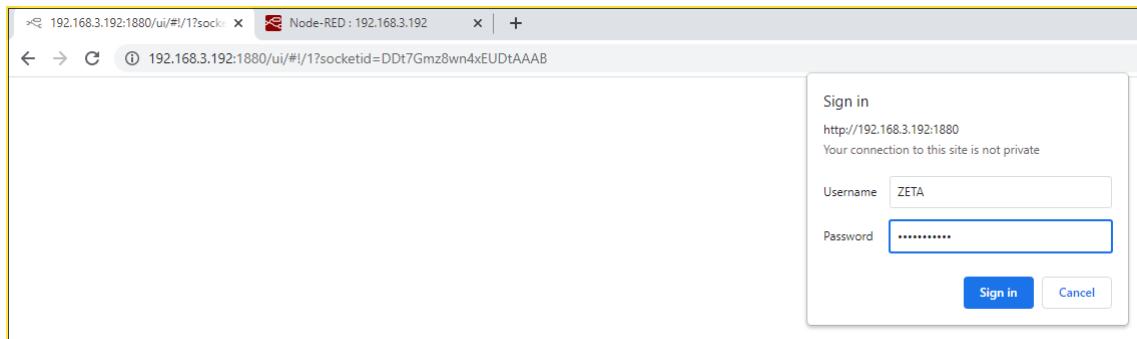


FIGURE 4.13 – Connexion avec le nouveau mot de passe

Ces différentes figures décrivent le processus de sécurisation de Node-RED en utilisant des mots de passe hachés. Cela garantit que seuls les utilisateurs autorisés peuvent accéder à notre projet et protège ainsi notre système contre les accès non autorisés.

4.4.6 Réalisation de projet avec Node-RED

Dans notre topologie, nous utilisons 2 LED, un capteur DHT11 et une breadboard, que nous connectons aux GPIO de Raspberry Pi, comme le montre la figure 4.14, réalisée à l'aide de Fritzing.

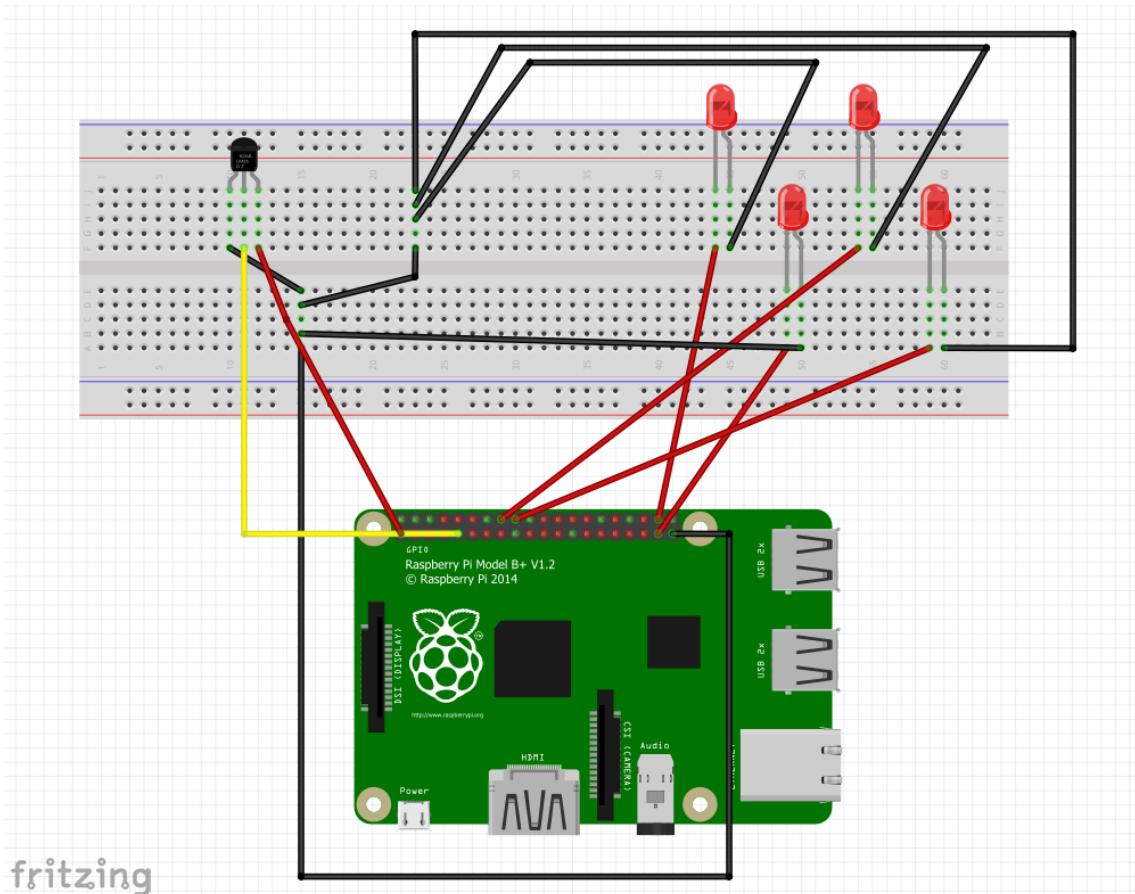


FIGURE 4.14 – Schéma de notre réalisation

Ensuite, nous configurons les nodes DHT11 dans Node-RED pour séparer les données reçues entre Température et Humidité et les configurons pour actualiser les données toutes les 10 secondes.



FIGURE 4.15 – Ajout d'un Inject Node pour répéter l'action chaque 10 secondes

Dans la figure 4.15, nous ajoutons un nœud Inject pour répéter l'action

toutes les 10 secondes, permettant ainsi la mise à jour régulière des données.

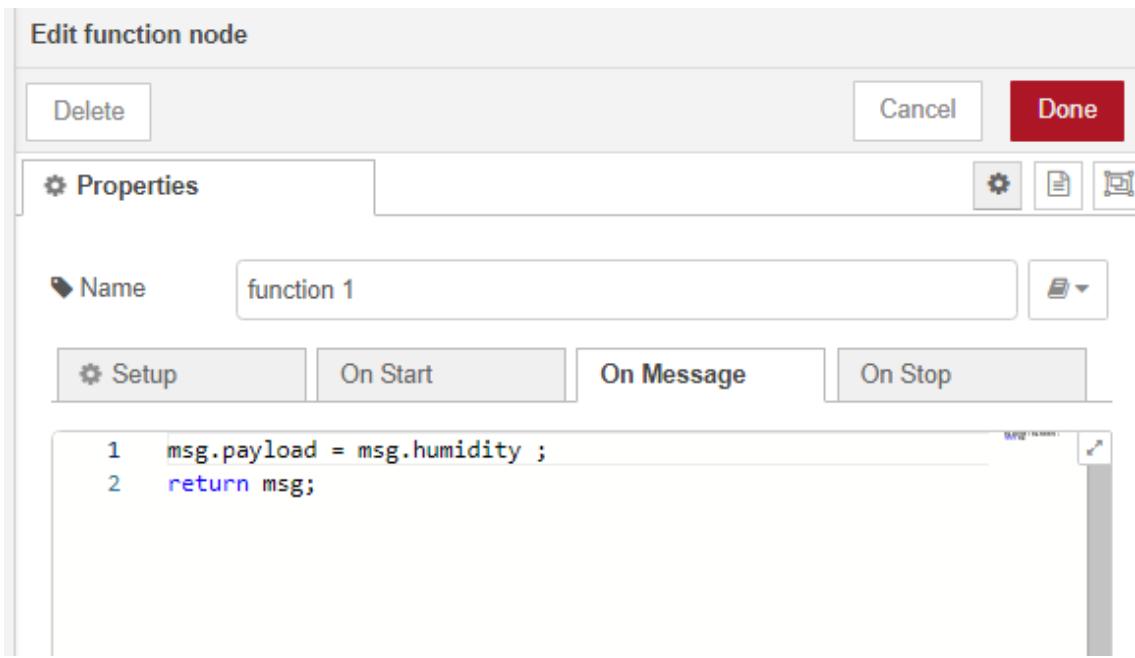


FIGURE 4.16 – Configurer un nœud Fonction pour séparer l'humidité de la température

Dans la figure 4.16, nous configurons un nœud Fonction pour séparer l'humidité de la température, ce qui permet d'obtenir des valeurs distinctes pour chaque paramètre.

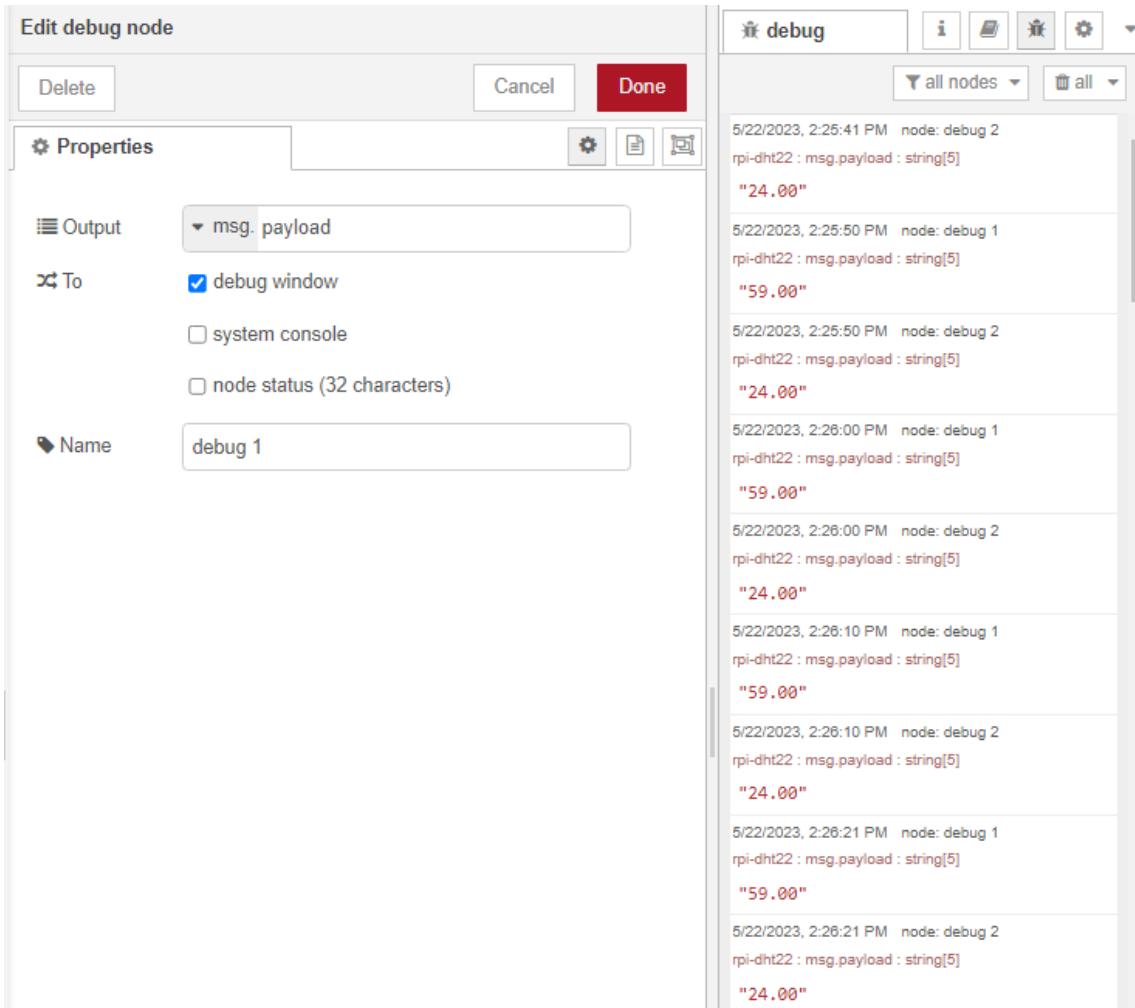


FIGURE 4.17 – Le résultat du nœud Debug

La figure 4.17 montre le résultat obtenu à l'aide d'un nœud Debug, où les données séparées d'humidité et de température sont affichées.

De plus, nous créons des nœuds interrupteurs (Switches) pour contrôler les LEDs.

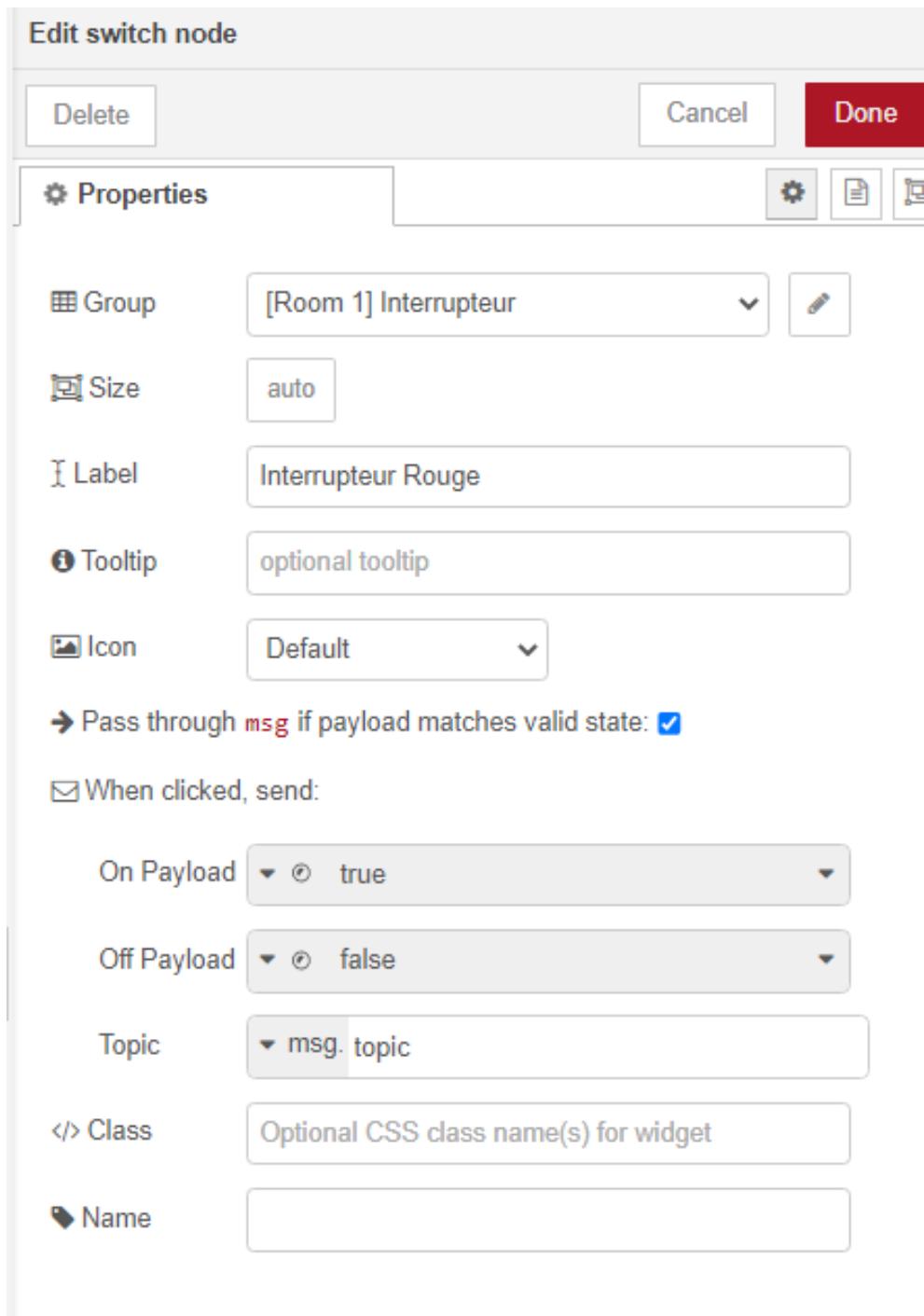


FIGURE 4.18 – Nœud Switch

Dans la figure 4.18, nous avons ajouté un nœud Switch permettant de contrôler l'état des LEDs lorsqu'il est cliqué.

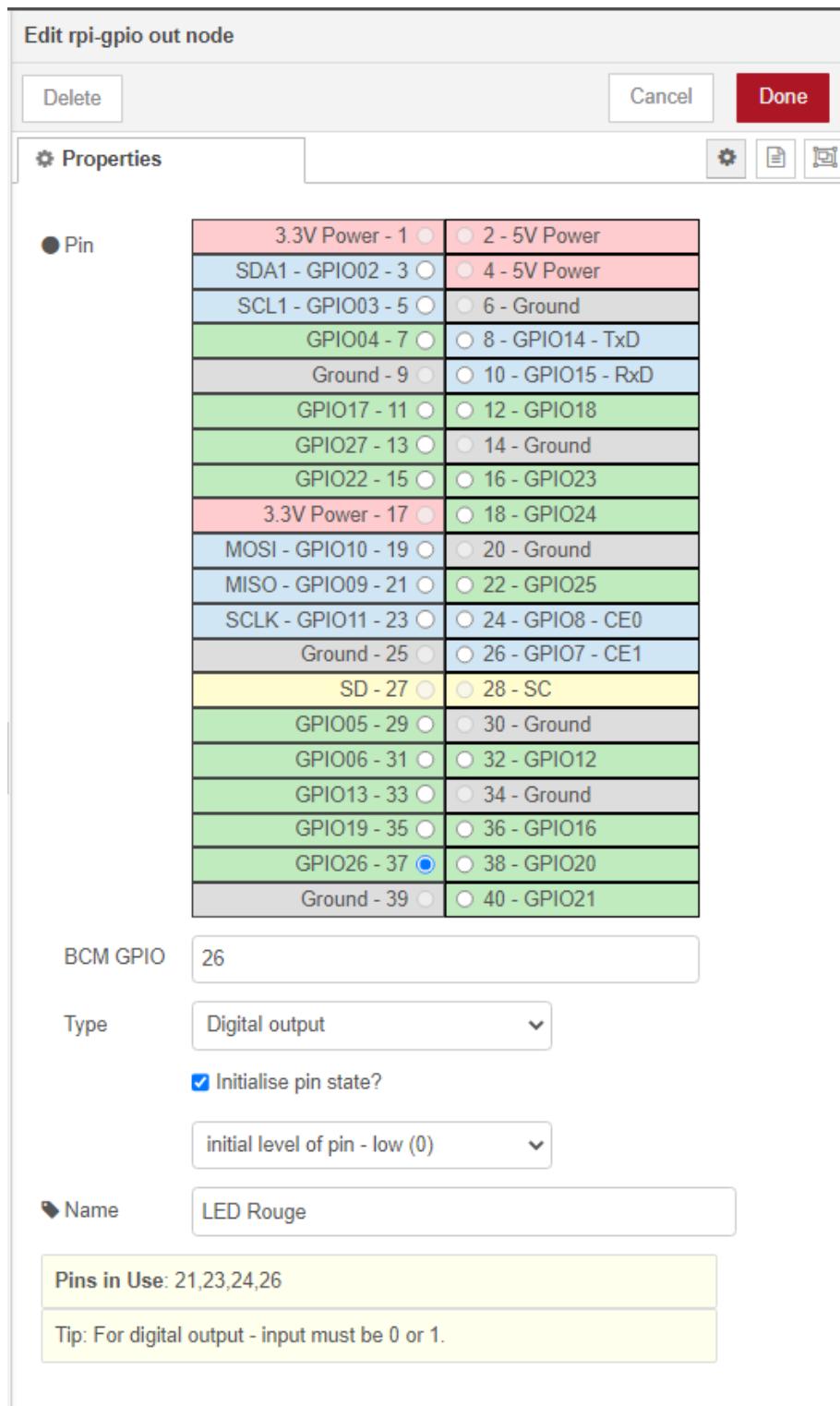


FIGURE 4.19 – Nœud Raspberry GPIO Output

La figure 4.19 présente le nœud Raspberry GPIO Output utilisé pour contrôler les GPIO du Raspberry Pi et allumer/éteindre les LEDs.

Finalement, voici les nœuds et l'exécution finale de mon projet Node-

RED.

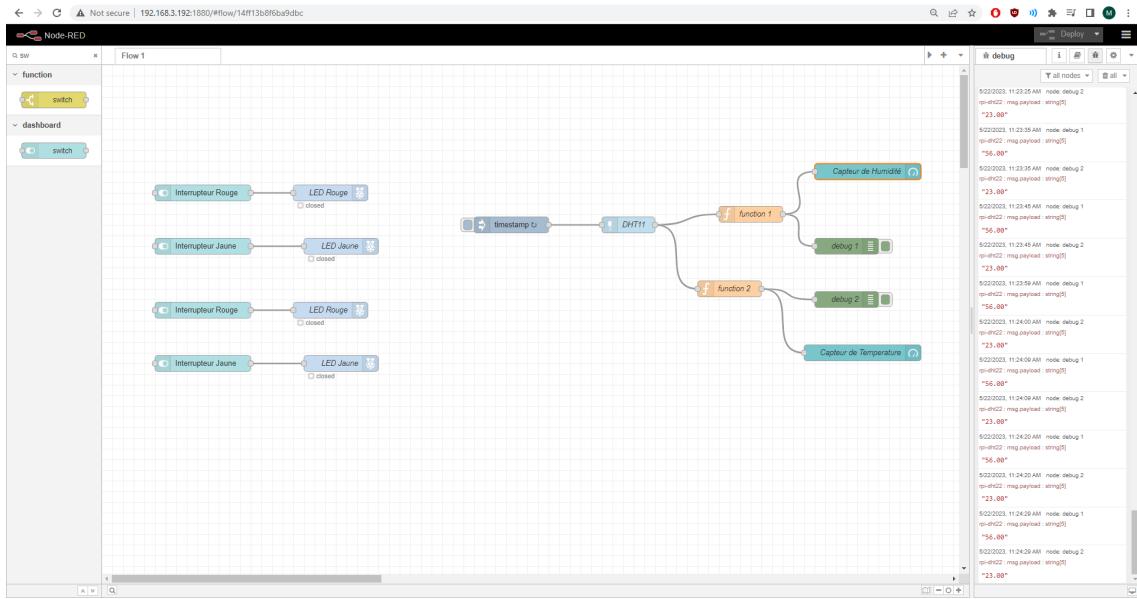


FIGURE 4.20 – Le projet en Node-RED

La figure 4.20 présente le projet Node-RED complet avec tous les nœuds configurés et connectés.

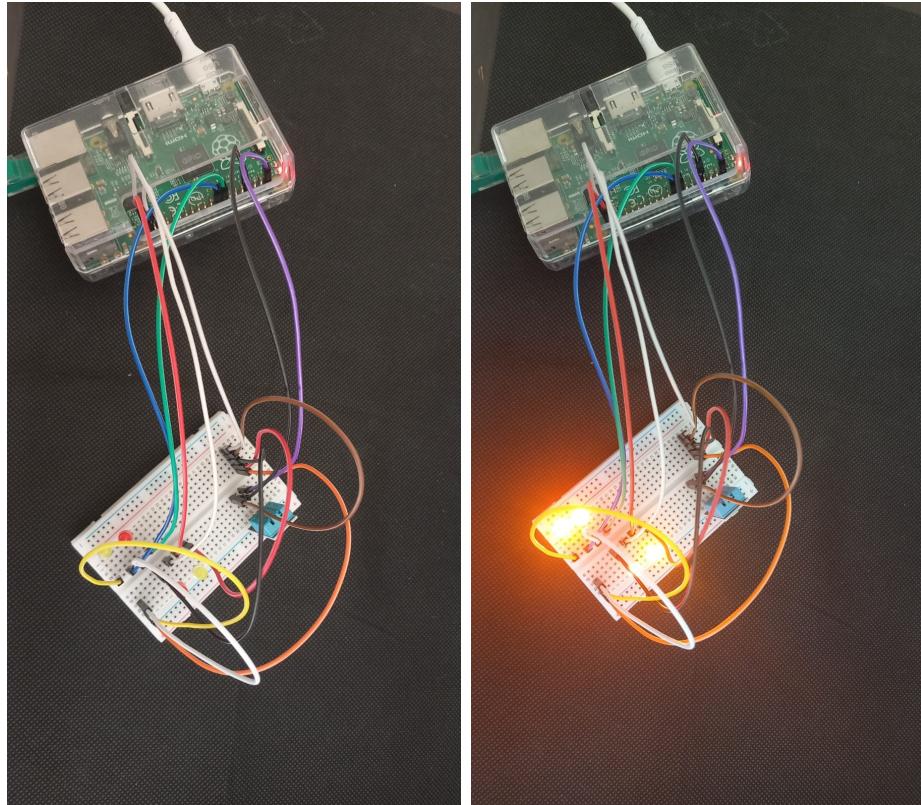


FIGURE 4.21 – Circuit de mon prototype miniature

Dans la figure 4.21, nous pouvons voir le circuit prototype miniature de notre projet, avec les composants connectés sur la breadboard et le Raspberry Pi.

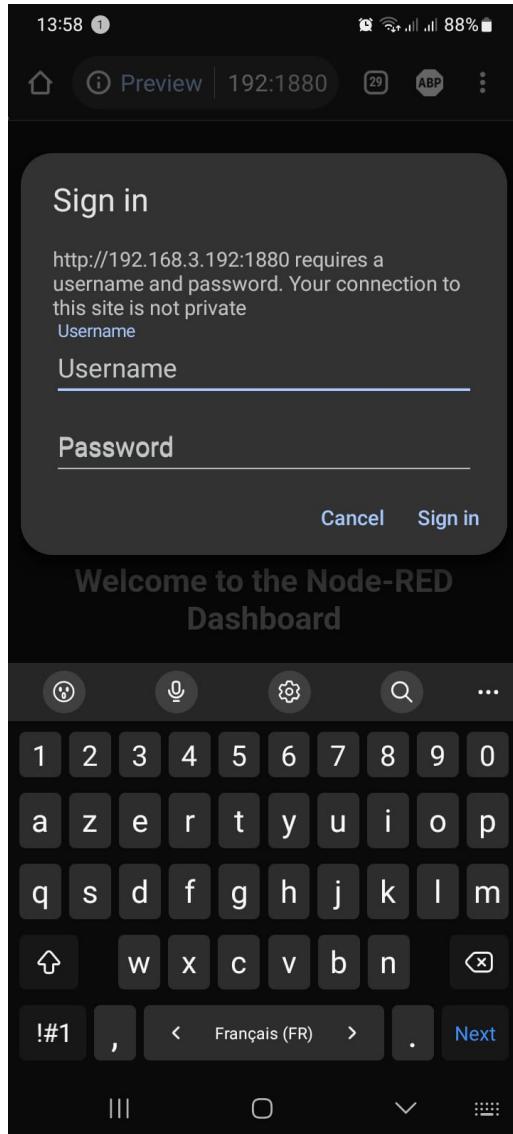


FIGURE 4.22 – Le login du tableau de bord en mobile

La figure 4.22 illustre l'interface de connexion (Dashboard Login) de notre projet, conçue pour une utilisation sur mobile. Cette figure démontre que notre projet dispose d'un accès réseau via l'adresse "192.168.3.192 :1880". Ici, "192.168.3.192" correspond à l'adresse IP de notre Raspberry Pi, et "1880" indique le port sur lequel Node-RED est en cours d'exécution.

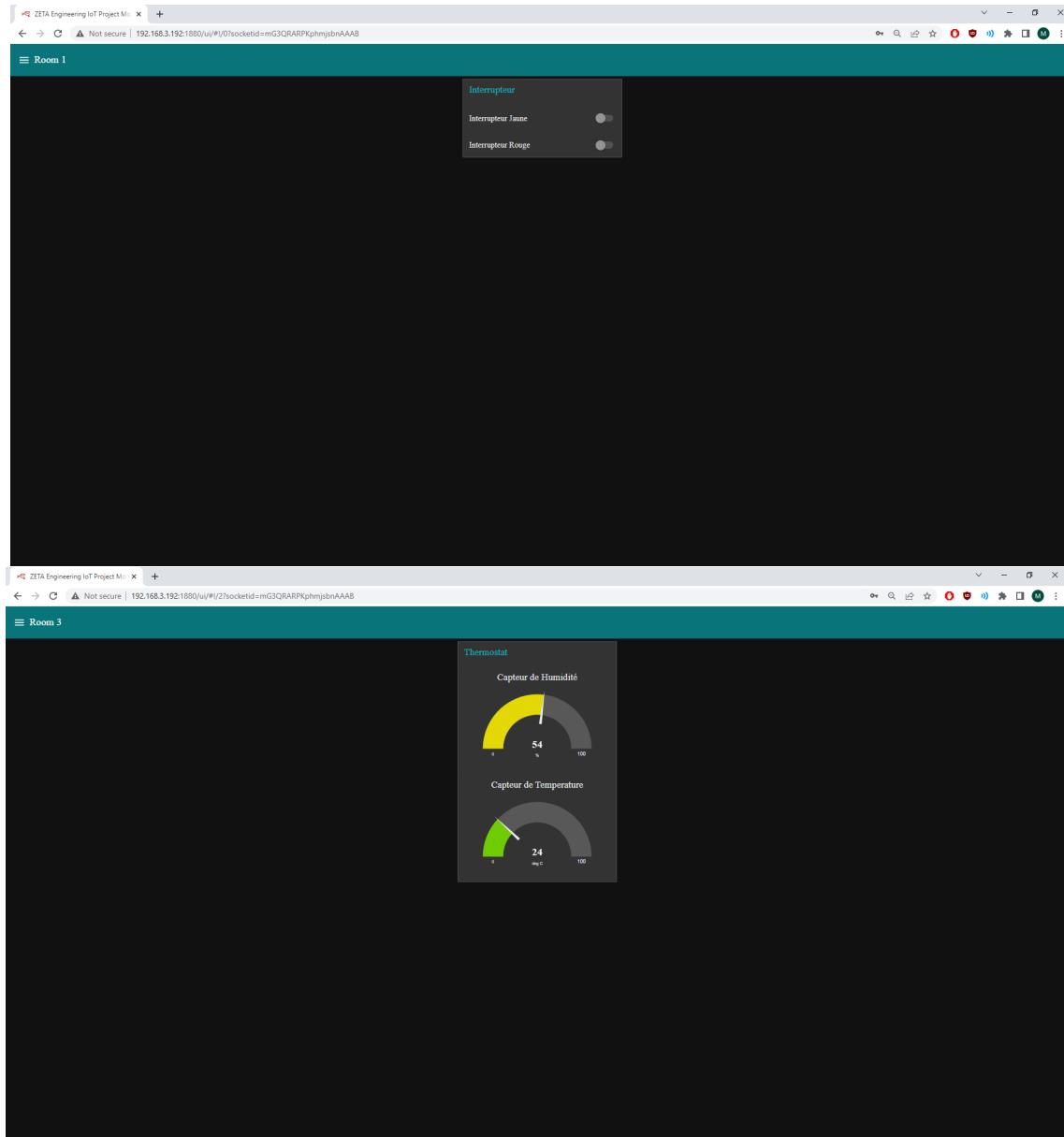


FIGURE 4.23 – Le tableau de bord en bureau

La figure 4.23 montre le tableau de bord (Dashboard) de notre projet en bureau.

4.4.7 Intégration dans le parc informatique

Une fois notre application prête, nous intégrons celle-ci dans le parc informatique que nous avons créé dans le troisième chapitre à l'aide de GLPI.

Pour cela, nous installons le FusionInventory-Agent sur notre Rasp-

berry Pi, comme illustré ci-dessous en figure 4.24.

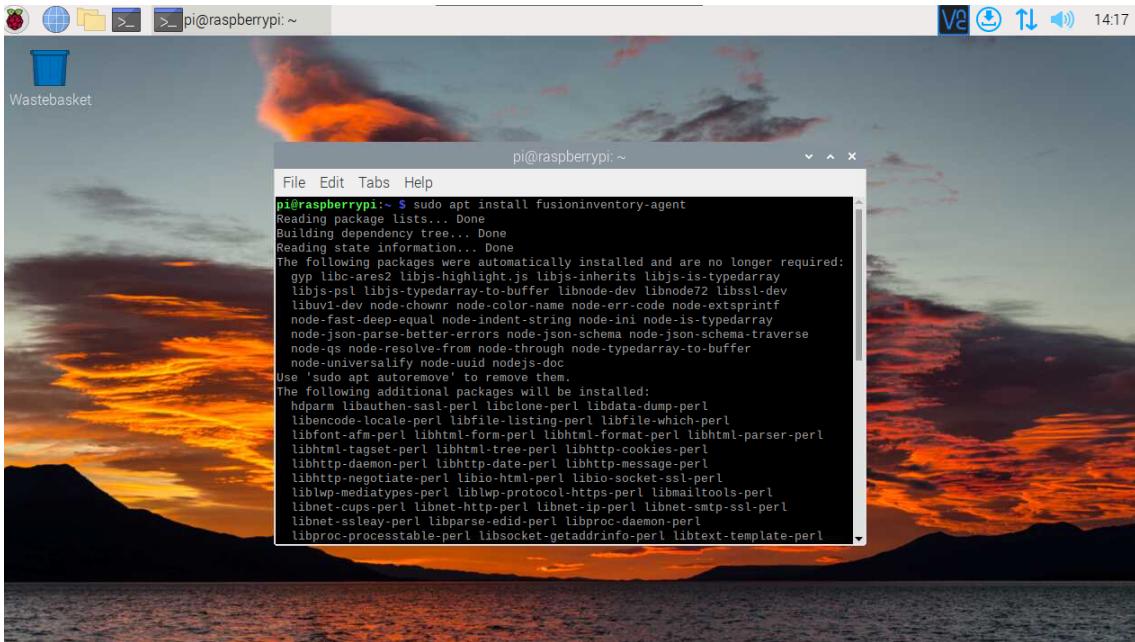


FIGURE 4.24 – Installation de FusionInventory-Agent

Ensuite, nous configurons le FusionInventory-Agent pour qu'il envoie les données de notre périphérique au serveur GLPI à l'adresse "192.168.3.66 / glpi / fusioninventory ", comme présenté dans la figure 4.25.

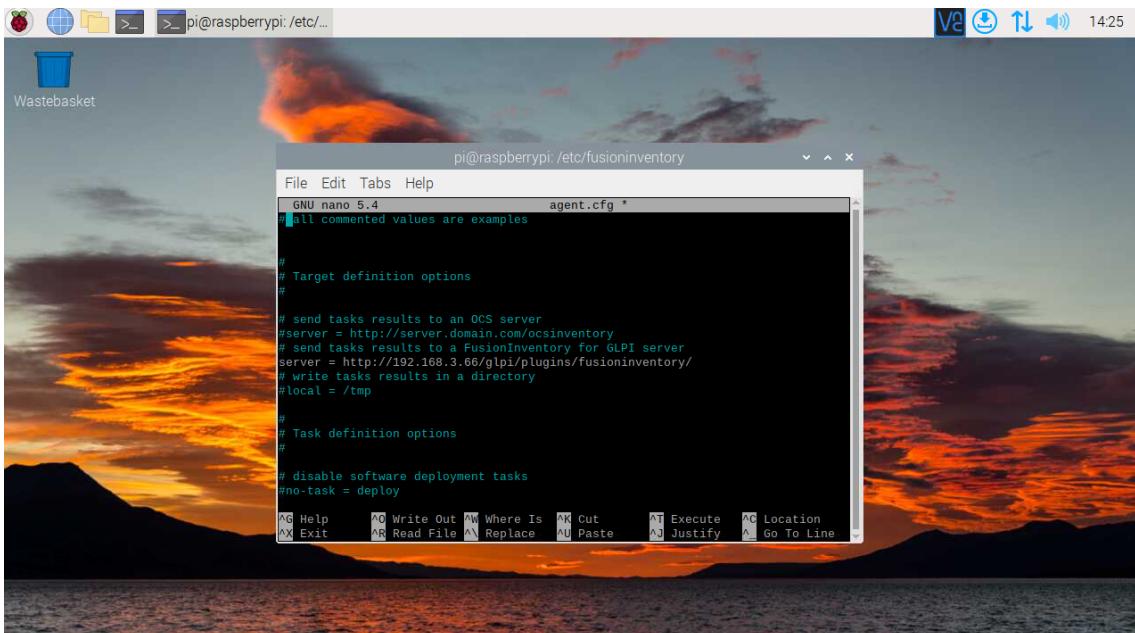


FIGURE 4.25 – Configuration de FusionInventory-Agent

Enfin, nous visualisons notre Raspberry Pi intégré dans le tableau de

bord de GLPI 4.26.

The screenshot shows the GLPI 4.26 interface. The top navigation bar includes Home, Assets (selected), Assistance, Management, Tools, Administration, and Setup. A search bar is at the top right. Below the navigation is a breadcrumb trail: Home > Assets > Computers. A sidebar on the left lists various asset categories with counts: Computer (1), Impact analysis, Operating systems (1), Components (4), Volumes (2), Software (1494), Connections, Network ports (2), Management, Contract, Documents, Virtualization, Antiviruses, Knowledge base, Tickets, Problems, Changes, and External links. The main content area is titled "raspberrypi". It contains two tabs: "Computer" and "FusionInventory". The "Computer" tab displays detailed information: Name (raspberrypi), Status (available), Location (empty), Type (BCM2835), Technician in charge of the hardware (empty), Manufacturer (empty), Group in charge of the hardware (empty), Model (BCM2835), Alternate username number (empty), Serial number (0000000453a24a3), Alternate username (pi), Inventory number (empty), User (empty), Group (empty), Network (empty), UUID (empty), and Update Source (empty). The "Comments" field is also empty. The "FusionInventory" tab shows: Agent (raspberrypi-2023-07-06-14-28-02), Useragent (FusionInventory-Agent_v2.8.2), Status (cannot contact the agent), FusionInventory tag (IoT PFE), Public contact address (192.168.3.77), Last contact (2023-07-06 14:28), Last inventory (2023-07-06 14:28), and Last boot (2023-07-06 14:12). At the bottom, it says "Created on 2023-07-06 14:28" and "Last update on 2023-07-06 14:28".

FIGURE 4.26 – Tableau de bord de GLPI avec le Raspberry Pi intégré

4.5 Conclusion

En conclusion de ce chapitre dédié à l'intégration des objets connectés, nous avons exploré en détail les différentes étapes nécessaires pour incorporer ces dispositifs dans l'infrastructure que nous avons créer au chapitre précédent. Les objets connectés, de par leur capacité à collecter, communiquer et analyser des données en temps réel, offrent un potentiel considérable pour améliorer l'efficacité et la qualité de vie de l'entreprise.

L'intégration des objets connectés représente un pas significatif vers une infrastructure informatique moderne et intelligente, et ce chapitre nous a fourni les connaissances et les outils nécessaires pour relever ce défi avec succès.

Conclusion Générale

Ce projet de Mémoire de Mastère a exploré en profondeur l'univers de l'informatique en entreprise. Nous avons exploré trois aspects fondamentaux : le déploiement de l'infrastructure informatique, la mise en place du système d'information et l'intégration des objets connectés.

Lors du déploiement de l'infrastructure informatique, nous avons conçu un réseau MAN pour relier les trois locaux de l'entreprise, utilisant des technologies avancées comme le TP-Link CPE610 pour garantir une connectivité stable.

Le déploiement du système d'information a été rigoureux, avec des choix technologiques judicieux comme Ubuntu Server 22.04, KVM et GLPI pour créer un système sur mesure répondant aux besoins spécifiques de l'entreprise.

Pour l'intégration des objets connectés, divers dispositifs, tels que des capteurs, des lampes et des machines de pointage, ont été connectés au réseau à l'aide de la carte Raspberry Pi et de logiciels spécialisés.

En conclusion, cette Mémoire de Mastère témoigne que l'intégration des objets connectés dépasse largement le cadre d'une simple tendance technologique. Elle représente une transformation profonde dans la manière dont les entreprises évoluent et interagissent avec leur environnement. Nous sommes fermement convaincus que les entreprises qui sauront capitaliser sur ces avancées technologiques seront mieux préparées pour faire face aux défis de demain et maintenir leur compétitivité dans un monde en constante évolution.

Bibliographie

- [1] B. G. de Projet. (s.d.) Le modèle en cascade. Consulté le 10 mai 2023. [Online]. Available : <https://blog-gestion-de-projet.com/modele-en-cascade/>
- [2] T. Antoine-Santoni, B. Poggi, E. Vittori, H. Van Hieu, D. Araujo, and A. Aiello, “Vers un système d’information pervasif pour un smart village,” in *Evolution des SI : vers des SI Pervasifs ?, 2019*.
- [3] Agaxis site web. C’est le site web du Agaxis. [Online]. Available : <https://www.agaxis.com>
- [4] C. Fagarasan, O. Popa, A. Pisla, and C. Cristea, “Agile, waterfall and iterative approach in information technology projects,” in *IOP Conference Series : Materials Science and Engineering*, vol. 1169, no. 1. IOP Publishing, 2021, p. 012025.
- [5] D. Sze, “A metropolitan area network,” *IEEE Journal on Selected Areas in Communications*, vol. 3, no. 6, pp. 815–824, 1985.
- [6] Open source guide. Est le site ou on a utilisé pour vérifier les meilleures solutions à implementer. [Online]. Available : <https://open-source-guide.com/Solutions>
- [7] S. Hameed, “How to download and install ubuntu 22.04 [step by step],” *Linux Genie*, 2023. [Online]. Available : <https://linuxgenie.net/how-to-download-and-install-ubuntu-22-04/>
- [8] How to install dhcp server on ubuntu. LinuxHint. [Online]. Available : https://linuxhint.com/install_dhcp_server_ubuntu/
- [9] S. Singh and S. Patel. (2023) How to install asterisk on ubuntu 22.04. Published on VegaStack. [Online]. Available : <https://vegastack.com/tutorials/how-to-install-asterisk-on-ubuntu-22-04/>
- [10] D. Daga and S. Patel. (2023) Step-by-step tutorial to install odoo 16 erp on ubuntu 22.04. Published on 4th July 2023. [Online]. Available : <https://vegastack.com/tutorials/step-by-step-tutorial-to-install-odoo-16-erp-on-ubuntu-22-04/>

- [11] J. Kiarie. (2022) How to install glpi [it asset management] tool on rhel systems. Last Updated on May 14, 2022. [Online]. Available : <https://www.tecmint.com/install-glpi-asset-management-rhel/>
- [12] J. Patil, “How to install openldap server and configure the openldap client ?” *Clairvoyant Blog*, October 2020, accessed on September 12, 2023. [Online]. Available : <https://blog.clairvoyantssoft.com/installation-of-openldap-server-and-configure-the-openldap-client-8be45698f8c8>
- [13] A. Gazis, “What is iot ? the internet of things explained,” *Academia Letters*, p. 2, 2021.
- [14] T. L. Scott and A. Eleyan, “Coap based iot data transfer from a raspberry pi to cloud,” *2019 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–6, 2019. [Online]. Available : <https://api.semanticscholar.org/CorpusID:208212600>
- [15] S. R. Department. (2016) Internet des objets : nombre d'appareils connectés dans le monde 2015-2025. * Données prévisionnelles de 2017 à 2025. [Online]. Available : <https://fr.statista.com/statistiques/584481/internet-des-objets-nombre-d-appareils-connectes-dans-le-monde-2020/>
- [16] S. Thomas, “Plus de 100 statistiques importantes sur l'automatisation en 2023,” *MarketSplash*, 2023. [Online]. Available : <https://marketsplash.com/fr/statistiques-dautomatisation/>
- [17] M. Richardson and S. Wallace, *Getting started with raspberry PI*. " O'Reilly Media, Inc.", 2012.
- [18] M. Lekić and G. Gardašević, “Iot sensor integration to node-red platform,” in *2018 17th International Symposium Infoteh-Jahorina (Info-teh)*. IEEE, 2018, pp. 1–5.

Liste des acronymes

IP : *Internet Protocol*

LAN : *Local Area Network*

WAN : *Wide Area Network*

MAN : *Metropolitan Area Network*

IT : *Information Technology*

SI : *Système d'Information*

QoS : *Quality of Service*

VPN : *Virtual Private Network*

SNMP : *Simple Network Management Protocol*

RMON : *Remote Monitoring*

LLDP : *Link Layer Discovery Protocol*

VLAN : *Virtual Local Area Network*

STP : *Spanning Tree Protocol*

Mbps : *Megabits per second*

IoT : *Internet of Things*

OS : *Operating System*

NVR : *Network Video Recorder*

Résumé

Ce rapport présente le travail réalisé dans le cadre du projet de Mémoire De Mastère au sein de la société ZETA Engineering. L'objectif était de concevoir et déployer une infrastructure IT et SI, ainsi que d'intégrer des objets connectés.

Nous avons suivi une méthodologie rigoureuse, analysé l'infrastructure existante et élaboré un plan détaillé pour la mise en place d'une infrastructure solide. L'implémentation a impliqué la configuration du matériel et des logiciels nécessaires, ainsi que l'intégration avec les systèmes existants.

Nous avons également intégré des objets connectés pour améliorer l'efficacité opérationnelle et collecter des données en temps réel. Cela a nécessité la sélection et la configuration d'appareils IoT appropriés.

En résumé, ce projet visait à renforcer les capacités IT et SI de Zeta Engineering en fournissant une infrastructure fiable et sécurisée, ainsi qu'en intégrant des objets connectés pour améliorer la productivité et éclairer la prise de décision.

Abstract

This report outlines the implementation of an IT and SI infrastructure plan and the integration of connected devices within ZETA Engineering company. The objective was to design and deploy a comprehensive IT and SI infrastructure to support the company's operations.

Throughout our internship, we followed a systematic approach, conducted analysis, and developed a detailed plan for a robust infrastructure. The implementation involved setting up necessary hardware and software, as well as seamless integration with existing systems.

Furthermore, we integrated connected devices to enhance operational efficiency and gather real-time data. This involved selecting and configuring appropriate IoT devices and establishing necessary connectivity protocols.

In summary, this project aimed to enhance Zeta Engineering's IT and SI capabilities by providing a reliable and secure infrastructure, as well as integrating connected devices to improve productivity and decision-making.