HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

Département des Technologie de l'information et de la communication (TIC)
Filière Télécommunications
Orientation Sécurité de l'information

Travail de Bachelor

WiFace

WiFace - Identification et suivi de personnes utilisant le WiFi du téléphone mobile

Étudiant	**Florian Polier**
Enseignant responsable	Prof. Abraham Rubinstein
Entreprise mandante	-
	-
	-
Année académique	2019-2020

Yverdon-les-Bains, le 17 juin 2020

Département des Technologie de l'information et de la communication (TIC)
Filière Télécommunications
Orientation Sécurité de l'information
Étudiant : Florian Polier
Enseignant responsable : Prof. Abraham Rubinstein

Travail de Bachelor 2019-2020

WiFace

Nom de l'entreprise/institution

-

Résumé publiable

Dans ce travail... Ceci est le résumé publiable...

Étudiant :	Date et lieu :	Signature :
Florian Polier	.....	.....
Enseignant responsable :	Date et lieu :	Signature :
Prof. Abraham Rubinstein	.....	.....
Nom de l'entreprise/institution :	Date et lieu :	Signature :
-	.....	.....
-		

Préambule

Ce travail de Bachelor (ci-après TB) est réalisé en fin de cursus d'études, en vue de l'obtention du titre de Bachelor of Science HES-SO en Ingénierie.

En tant que travail académique, son contenu, sans préjuger de sa valeur, n'engage ni la responsabilité de l'auteur, ni celles du jury du travail de Bachelor et de l'Ecole.

Toute utilisation, même partielle, de ce TB doit être faite dans le respect du droit d'auteur.

HEIG-VD

Vincent Peiris
Chef de département TIC

Yverdon-les-Bains, le 17 juin 2020

Authentification

Le soussigné, Florian Polier, atteste par la présente avoir réalisé ce travail et n'avoir utilisé aucune autre source que celles expressément mentionnées.

Yverdon-les-bains, le 17 juin 2020

Florian Polier

Cahier des charges

Résumé du problème

Les dispositifs Wifi diffusent en permanence des trames qui permettent de trouver rapidement les réseaux à proximité. Ces trames, appelées “probe requests” sont utilisées par des “sniffers” pour “tracer” les utilisateurs dans des centres commerciaux et d’autres endroits publics.

Ces informations jouissent pourtant d’un certain anonymat. En effet, les adresses MAC des dispositifs sont révélées par ces trames mais il n’est normalement pas possible de les associer avec l’identité d’un individu. De plus, les problématiques de « privacy » ont peu à peu amené les constructeurs à implémenter des mécanismes anonymisant les utilisateurs, par exemple en rendant les adresses MAC pseudo-aléatoires.

Dans ce projet, l’étudiant ou l’étudiante devra concevoir et développer un démonstrateur pour un système capable de combiner un “sniffer” amélioré capable de récolter les “probe request” et de prendre en même temps des photos des visages se trouvant à proximité du capteur de trames (par exemple, en face d’une vitrine d’un magasin). Le système tentera de corréler des adresses MAC et des images de visages récoltées à des endroits différents afin d’associer un visage à une adresse. Finalement, une recherche par reconnaissance d’images sur les réseaux sociaux (pour le démonstrateur, la recherche sera faite sur une base de données) essaiera d’obtenir l’identité du propriétaire du téléphone mobile et toutes les données disponibles sur ce dernier. Des attaques visant la désanonymisation pourront être mises en place afin de tracer au mieux les utilisateurs.

Objectifs du travail de Bachelor

Au terme du travail de bachelor, les objectifs suivants auront été remplis :

- Le prototype développé permettra de scanner les probes requests à proximité et à en extraire les informations utiles (adresse MAC, SSID)
- Le prototype développé permettra de prendre des photos lorsqu'il détectera des visages dans son champs d'action
- Le prototype développé utilisera des mécanismes pour associer une identité, des photos, et des appareils
- Le prototype développé utilisera des mécanismes de recherche d'information automatisée afin de compléter les profils identifiés
- L'architecture du projet permettra de faire fonctionner plusieurs prototypes en parallèle, en partageant les mêmes données persistantes
- Les données récoltées pendant le fonctionnement du prototype seront persistantes
- Une étude sur la légalité et les enjeux éthiques de ce produit sera réalisée
- Une étude sur les divers mécanismes d'anonymisation de l'adresse MAC, et l'état actuel d'implémentation sera réalisée
- Une étude sur la problématique de la reconnaissance faciale sans training set sera réalisée
- À l'aide de la documentation produite et du matériel adéquat, le prototype devra être reproductible pour un lecteur externe

Si le temps le permet, ainsi que les contraintes techniques, les objectifs suivants seront visés :

- Le prototype développé permettra de détecter la randomisation des adresses MAC
- Le prototype développé utilisera des mécanismes actifs ou passifs afin d'attaquer la randomisation des adresses MAC et ainsi de permettre la désanonymisation de l'utilisateur
- Les différentes photos d'une personne pourront être regroupées sous la même identité à l'aide de technologie de reconnaissance faciale, même sans « training set » initial

Déroulement global du travail

Le travail peut être découpé en plusieurs parties, permettant une meilleure organisation générale du travail et des délivrables :

1. Préparation au travail
    - Recherches initiales sur la problématiques et l'état de l'art
    - S'informer sur les directives et le cadre imposé pour le travail de Bachelor
    - Rédaction du présent cahier des charges

- Étude de marché pour le matériel nécessaire

2. Installation de l'environnement
   - Commande de matériel
   - Installation de l'OS
   - Installation de l'environnement de développement
3. Conception de la base de données
   - Modèle entité-association
   - Modèle logique de données
   - Script de création
4. Développement du scanner réseau
   - Capture des probes requests et Extraction des données
   - (secondaire) Détecter la randomisation des adresses MAC
   - (secondaire) Attaque de la randomisation des adresses MAC
   - Insertion dans la base de données
   - Tests du module
5. Développement du module de reconnaissance faciale
   - Prise de photo à la détection de visage
   - Reconnaissance de visage
   - Association probabiliste avec une ou plusieurs adresses MAC
   - Insertion dans la base de données
   - Tests du module
6. Test du prototype final
7. Documentation
   - Rédaction du cahier des charges
   - Rédaction du journal de travail
   - Rédaction du rapport intermédiaire et final
   - Rédaction et recherche sur l'analyse légale et éthique
   - Rédaction et recherche sur la partie théorique
   - Rédaction du mode d'emploi

Délivrables et résultats attendus

Au terme du travail de bachelor, les délivrables suivants seront rendus :

1. Un rapport final qui contiendra, en plus du contenu imposé par les directives de la HEIG-VD :
   - Une analyse sur la légalité et les enjeux éthiques de produits permettant l'identification et le traçage des utilisateurs
   - Une analyse sur le matériel à acquérir pour développer le prototype
   - La description de chaque étape d'implémentation
   - Une partie théorique sur au moins un des aspects suivants (reconnaissance faciale, identification d'un appareil à l'aide de probe request, attaques sur les mécanismes de protection de l'identité)
   - Un mode d'emploi permettant l'installation et l'utilisation du prototype
2. Un prototype remplissant les exigences mentionnées plus haut, utilisable pour une démonstration

Avant le 19 juin 2020, un rapport intermédiaire sera rendu.

Table des matières

Chapitre 1

Problématique et État de l'Art

1.1 Introduction

Ce rapport documente le travail effectué dans le cadre du Travail de Bachelor (TB) de la formation de Sécurité informatique, orientation de la filière Informatique et systèmes de communication du département Technologie de l'Information et de la Communication de la HEIG-VD.

Ce projet, intitulé "WiFace" aborde la problématique suivante :

Les dispositifs Wifi diffusent en permanence des trames qui permettent de trouver rapidement les réseaux à proximité. Ces trames, appelées "probe requests" sont utilisées par des "sniffers" pour "tracer" les utilisateurs dans des centres commerciaux et d'autres endroits publics.

Ces informations jouissent pourtant d'un certain anonymat. En effet, les adresses MAC des dispositifs sont révélées par ces trames mais il n'est normalement pas possible de les associer avec l'identité d'un individu. De plus, les problématiques de « privacy » ont peu à peu amené les constructeurs à implémenter des mécanismes anonymisant les utilisateurs, par exemple en rendant les adresses MAC pseudo-aléatoires.

Dans ce projet, l'étudiant ou l'étudiante devra concevoir et développer un démonstrateur pour un système capable de combiner un "sniffer" amélioré capable de récolter les "probe request" et de prendre en même temps des photos des visages se trouvant à proximité du capteur de trames (par exemple, en face d'une vitrine d'un magasin). Le système tentera de corréler des adresses MAC et des images de visages récoltées à des endroits différents afin d'associer un visage à une adresse. Finalement, une recherche par reconnaissance d'images sur les réseaux sociaux (pour le démonstrateur, la recherche sera faite sur une base de données) essaiera d'obtenir l'identité du propriétaire du téléphone mobile et toutes les données disponibles sur ce dernier. Des attaques visant la désanonymisation pourront être mises en place afin de tracer au mieux les utilisateurs.

1.2 Démarche et finalité

Le but recherché est la création d'un **prototype** en tant que proof of concept afin de démontrer qu'il est facile pour un particulier de mettre un place une solution de surveillance avec peu de moyen.

Les conséquences d'un tel constat seront explorées et nous montrerons qu'il n'est peut-être pas souhaitable qu'une telle solution soit disponible sur le marché.

1.3 Travail à effectuer

En décomposant la problèmatique, plusieurs axes sont mis en évidence :

1. Développement d'un sniffer de trames Wifi
2. Intégration d'un système de reconnaissance faciale
3. Conception d'un algorithme de couplage Adresse MAC - Image
4. Création d'un prototype embarquant ces technologies sous la forme d'un nano-ordinateur
5. Documentation du travail et création de divers documents techniques (mode d'emploi, étude de marché, ...)

1.4 Quelques projets existants

Une bonne manière de se rendre compte de l'état de l'art est d'explorer divers projets déjà existants.

1.4.1 Probe Kit

"A must-have hobby kit for any amateur network data collection"

Probe Kit est un projet développé par Brannon Dorsey et Nick Briz. À la frontière entre technologie et oeuvre d'art, cette initiative nous propose de "collectioner" les probe requests, sous forme visuelle. Ce logiciel sniffe les trames WiFi et établit une liste de probe request par SSID, faisant grandir notre collection au fil du temps.

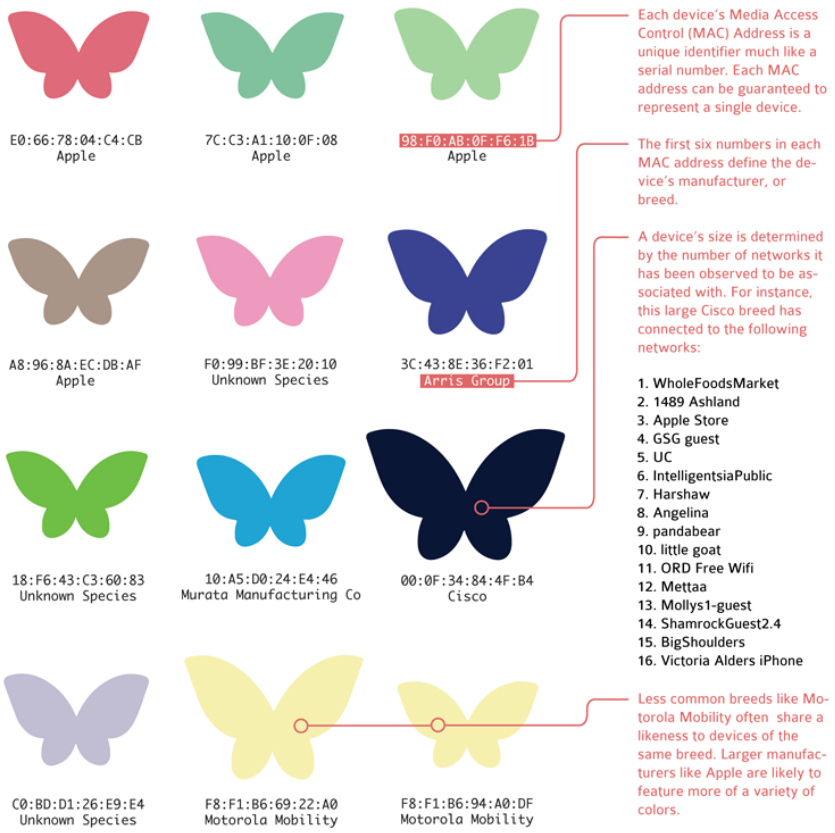

FIGURE 1.1 – Démonstration de Probe Kit

Avec cette approche ludique, cette démarche vise à sensibiliser les utilisateurs à la recolte passive de données.

1.4.2 CrowdProbe

En plus de la récolte de probe requests, il est possible d'en inférer des données. C'est ce que propose CrowdProbe [15]. En installant leur dispositif dans un musée, ils ont réussi à prédire avec une bonne précision le déplacement des utilisateurs, même lors de l'utilisation d'adresse MAC aléatoire.

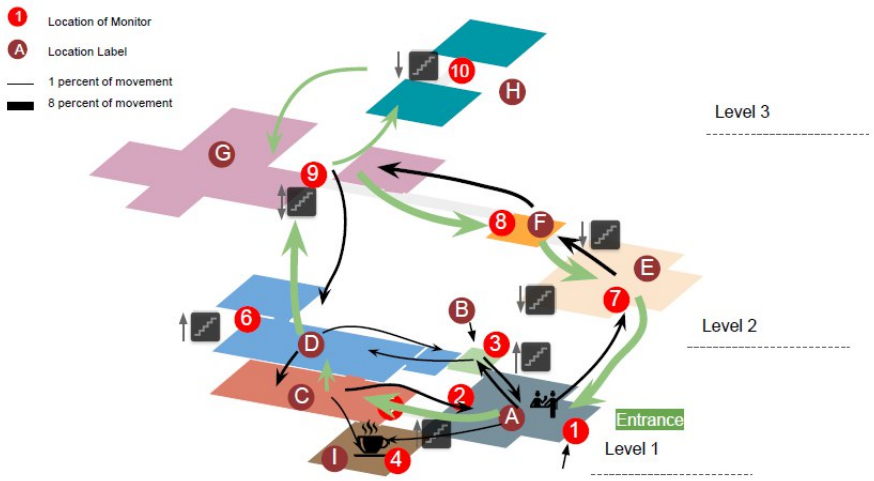

FIGURE 1.2 – Flux des visiteurs inféré à l'aide de modèles de Markov

1.4.3 Serrure déverouillable à l'aide de reconnaissance faciale

Voici un petit projet [2] permettant d'utiliser la portabilité de la Raspberry Pi afin de la coupler avec une caméra et une serrure intelligente. À l'aide d'OpenCV, le visage de l'utilisateur est reconnu et ouvre la porte si ce dernier est pré-enregistré.

J'ai inclus ce projet pour montrer qu'il est relativement aisé d'implémenter des mécanismes de reconnaissance faciale à l'aide de la Raspberry Pi

Chapitre 2

Étude sur la législation

Ce travail de Bachelor sera effectué en tant que « proof of concept » et ne sera pas utilisé dans le domaine publique, mais uniquement dans un environnement contrôlé. C'est pourquoi, il ne sera pas affecté par les différentes contraintes juridiques exprimées dans cette étude.

Toutefois, à titre informatif et pour démontrer la potentielle nocivité d'un tel système, il a été décidé d'effectuer une étude sur la légalité et la morale de l'utilisation d'outils de traçage.

Il est à noter que le cadre de cette étude ne comprend que la législation suisse. Les libertés individuelles et le lois sur la protection des données divergent beaucoup d'un pays à l'autre, c'est pourquoi mes conclusions ne sauraient être valides ailleurs.

2.1 Cas d'utilisation

Pour circonscrire ce travail dans un cadre juridique, il faut se projeter et prédire les use cases princpaux qu'une solution telle que WiFace pourrait remplir.

Pour rappel, la solution théorique de Wiface permet d'associer à une identité (Nom, prénom, adresse, informations personnelles) à un lieu, et à un device.

2.1.1 Shopping : Publicité ciblée

Il serait possible, à l'intérieur d'un grand centre commerciel, de placer plusieurs unités de tracking. Ainsi, le nom et l'adresse d'un client s'attardant régulièrement devant une enseigne permettrait de la publicité extrêmement ciblée.

Les probe-request donnant également des informations complémentaires directes (constructeur) ou indirectes (modèle du téléphone), il serait possible pour les vendeurs de matériel informatique de cibler de potentiels nouvaux acheteurs qui auraient un ancien appareil à

remplacer dans un futur proche.

2.1.2 Analyse de fréquentation d'un espace publique et prédiction de déplacement

Lors d'un travail de groupe au MIT, des étudiants ont récoltés plusieurs milions de probe requests et on fait une analyse de fréquentation de divers lieux clés du campus. Leur projet « Arealytics » les a même conduits à prédire les prochains déplacements d'un individu en fonction de son trajet courant.

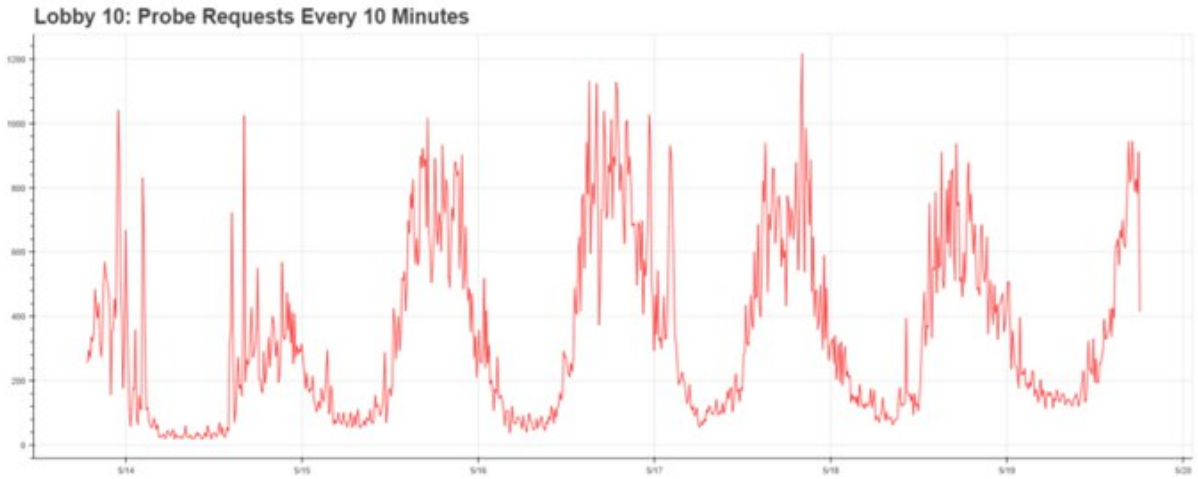

FIGURE 2.1 – Arealytics

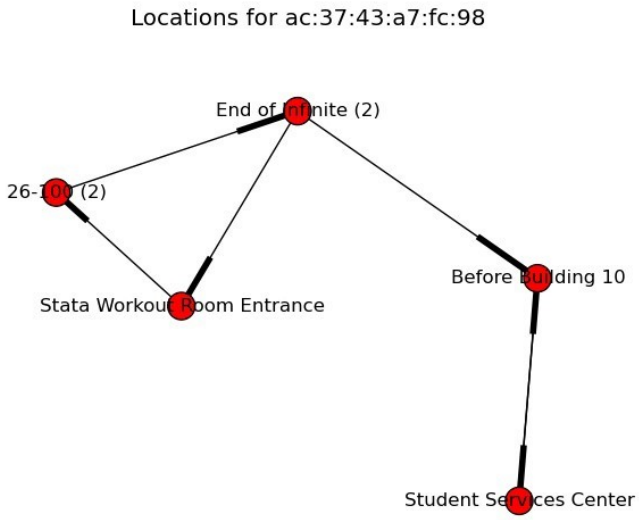

FIGURE 2.2 – Arealytics

Dans cet exemple, nous voyons que l'utilisation de simple probe request même sans prise d'image donne déjà énormément d'information brute.

2.1.3 Forensique

La capacité de pouvoir situer un individu, ou un groupe dans un espace géographique donné pourrait être utilisé comme preuve lors de procès. Par exemple, la présence d'une adresse MAC lors d'un crime, permet avec grande certitude de s'assurer de la présence de l'appareil associé. Couplé à une caméra, cette technique pourrait singulièrement améliorer les systèmes de surveillance actuels et proposer de nouveaux outils aux enquêteurs.

2.2 Définitions

Selon la Loi fédérale sur la protection des données (LPD) Section 1, Art 3 On entend par :

a données personnelles (données), toutes les informations qui se rapportent à une personne identifiée ou identifiable ;

b personne concernée, la personne physique ou morale au sujet de laquelle des données sont traitées ;

c données sensibles, les données personnelles sur :

  a les opinions ou activités religieuses, philosophiques, politiques ou syndicales,

  b la santé, la sphère intime ou l'appartenance à une race,

  c des mesures d'aide sociale,

  d des poursuites ou sanctions pénales et administratives ;

d profil de la personnalité, un assemblage de données qui permet d'apprécier les caractéristiques essentielles

e la personnalité d'une personne physique ;

f traitement, toute opération relative à des données personnelles – quels que soient les moyens et procédés utilisés – notamment la collecte, la conservation, l'exploitation, la modification, la communication, l'archivage ou la destruction de données ;

g communication, le fait de rendre des données personnelles accessibles, par exemple en autorisant leur consultation, en les transmettant ou en les diffusant ;

Ces quelques définitions vont permettre d'aborder les différentes problématiques dressée par ce travail.

2.3 Articles affectant le projet

Dans cette section, il sera fait mention de plusieurs articles de la LPD permettant de justifier – ou non – la potentielle mise en place du produit en conditions réelles de par sa licéité.

2.3.1 Section 1, Art 3, alinéa 2

« Leur traitement (des données personnelles) doit être effectué conformément aux principes de la bonne foi et de la proportionnalité. »

Certains des cas d'utilisations mentionnés ne permettent pas de mettre en œuvre le principe de proportionnalité. Par exemple, utiliser des mécanismes poussés pour découvrir le moyen de contacter un potentiel client sans intervention de sa part.

2.3.2 Section 1, Art 5, alinéa 1

«Celui qui traite des données personnelles doit s'assurer qu'elles sont correctes. Il prend toute mesure appropriée permettant d'effacer ou de rectifier les données inexactes ou incomplètes au regard des finalités pour lesquelles elles sont collectées ou traitées. »

Au vu des algorithmes qui seraient mis en place, il sera difficile voir impossible de garantir l'exactitude des données personnelles. Par exemple, l'assignation d'une identité à une adresse MAC à l'aide d'image est un processus hasardeux qui a de grande chance d'aboutir à une mauvaise labellisation des données personnelles

2.3.3 Section 3, Art 12, alinéa 2

«Personne n'est en droit notamment de :

a traiter des données personnelles en violation des principes définis aux art. 4, 5, al. 1, et 7, al. 1 ;

b traiter des données contre la volonté expresse de la personne concernée sans motifs justificatifs ;

c communiquer à des tiers des données sensibles ou des profils de la personnalité sans motifs justificatifs» Afin de retrouver l'identité d'une personne, la solution WiFace devra communiquer des informations personnelles à des tiers (e.g recherche inversée d'image).

2.4 Conclusion

Il est apparent que la plupart des cas d'utilisation ne respectent pas la LPD. Par conséquent, l'implémentation concrète d'une solution comme WiFace semble compromise sur le marché Suisse.

Chapitre 3

Éthique et moralité des systèmes de surveillance

Au delà de l'aspect législatif abordé dans le chapitre précédent, il est essentiel pour un ingénieur de mesurer ses responsabilités morales quant aux produits qu'il aide à créer. N'étant pas convenablement formé sur ce sujet, je me permettrai de me baser sur le travail d'autrui et de faire le lien avec le projet «WiFace».

3.1 Positionnement personnel

À des fins de transparence, j'ai jugé pertinent d'inclure mon positionnement sur la question de la moralité de mon projet, notamment avant le début de mes recherches.

En tant que technophile, je suis toujours emballé à l'idée de concevoir, développer ou implémenter de nouvelles idées et faire progresser petit à petit mon savoir. Toutefois, dès que le sujet m'a été présenté, je n'ai pu m'empêcher de penser aux conséquences sociétales du développement technologique, souvent plus rapide que le développement légistlatif et moral.

Prenons l'exemple évident du système de crédit social en Chine : les outils de surveillance de masse utilisés sont technologiquement très intéressant, mais beaucoup ressentent un certain malaise quant à leur champs d'application. Dans ce travail de bachelor, je développe également un outil, sans fixer son cadre d'utilisation. Je me permets donc d'être vigilant et d'inclure ce chapitre traitant des possibles dérives d'utilisation d'un tel produit.

3.2 La reconnaissance faciale, une technologie déshumanisante

Dès le début des recherches, il est apparu clairement que la reconnaissance faciale allait être l'aspect soulevant le plus de questionements moraux dans mon travail, bien plus que l'association d'une identité et d'un terminal. Pourquoi est-ce le cas ? Les deux situations permettent pourant de caractériser une personne de manière unique.

Le Prof. Brey l'explique en qualifiant le procédé de reconnaissance faciale de «déshumanisant». En effet, pouvoir encoder le visage (ses features) d'une personne – et donc une partie de son unicité - sur quelques bytes semble dérangeant pour beaucoup. De plus, un visage est partie intégrante de l'identité d'un individu, alors qu'un terminal n'est finalement qu'une possession temporaire.

Une autre problèmatique inhérente à cette technologie est celle de l'atteinte à la vie privée et à la confidentialité. Un système permettant d'identifier, tracer, journaliser et incriminer un individu dans l'espace publique ou privé nuit gravement aux droits des individus tels que mentionnés dans l'Article 8 de la Convention européenne des droits de l'homme.

Cette dernière proclame le droit de toute personne au respect « de sa vie privée et familiale, de son domicile et de sa correspondance », concrétisant ainsi nos questionnements.

3.3 L'opposition entre sécurité et confidentialité

Malgré tout, une part de l'opinion publique approuve l'implémentation de tels systèmes de reconnaissance. Comme argument principal, nous retrouvons souvent l'augmentation de la sécurité.

L'exemple des passeports bioémtriques est parlant : certains acceptent volontiers son utilisation car les avantages (lutte contre le terrorisme, l'usurpation d'identité, augmentation de la facilité de déplacement) semblent supérieurs aux désavantages (processus déshumanisant, atteinte à la liberté) mais il s'agit d'un cadre très précis, et ces même personnes ne seraient pas forcément d'accord de trouver de pareils systèmes dans d'autres circonstances (dans l'espace publique, dans des établissements privés). Cela nous amène donc à la première dérive : le «Function creep» (dérapage fonctionnel).

3.4 Le function creep

Ce terme, emprunté de l'auteur John Woodward est le phénomème par lequel une technologie développée dans un certain but outrepasse ces derniers et élargit son champs d'utilisation. Cela peut être dû à un usage abusif ou par des changements législatifs la concernant.

L'étude susmentionnée prend l'exemple des Smart CCTV au début des années 2000 (caméras

de surveillance couplées à un système de reconnaissance faciale.) Ces dernières ont été testées par la police afin de retrouver les personnes disparues et identifier des criminels inscrit dans une base de données.

Cette technologie étant très versatile, il est facile d'imaginer d'autres use-case pour la même implémentation de ce système. Par exemple, l'utilisation abusive pour le bénéfice personnel d'un agent de police (surveillance de ces proches, utilisation illégitime des images capturées). Le système initiale semblait alors moralement acceptable mais son dérapage fonctionnel l'a rendu intolérable, pourtant ce dernier n'est pas forcément détectable par les sujets de cette technologie (le grand public).

3.5 La fiabilité et les erreurs

Comme nous l'avons expliqué, il existe déjà des implémentations servant des systèmes très critiques, pouvant mener à des arrestations et d'autres conséquences importantes pour ses sujets. Or, bien qu'elle s'améliore avec le temps, la reconnaissance facial est sujette à des erreurs. Dans les systèmes de surveillance, cela amène à une méfiance générale, et à un sentiment d'insécurité.

Il existe deux types d'erreurs ayant des conséquences différentes :

1. Le faux négatif : Une personne devant être identifiée ne l'est pas, le système ne remplit pas on objectif et donc ses désavantages outrepassent ses avantages
2. Le faux positif : Une personne ne devant pas être identifiée est mal reconnue et prise pour quelqu'un d'autre, le système incrimine une personne innocente / non-concernée, cela décridibilise le système et amène à des conséquences néfastes

La deuxième catégorie d'erreur est plus critique que la première, c'est pourquoi la majeur partie des implémentations choisisse un seuil minimal de confiance très haut même si cela mène à un nombre supérieur de faux négatifs.

Pour ne rien arranger, il a été montré dans une étude effectuée par le NIST (National Institute of Standards and Technology) que la plupart des meilleures solutions commercialisées de reconnaissance faciale possédent un biais lié à l'origine du sujet. En effet, le taux d'erreur est 10 à 100 fois supérieurs sur les sujet Afro-Américains et asiatiques par rapport aux sujets caucasiens.

Un tel biais a le potentiel d'être un facteur d'augmentation de la tension sociale et pourrait impacter la législation en criminalisant et discriminant injustement certains groupes d'individu par rapport à d'autres.

3.6 La vie privée

Le dérapage fonctionnel et la possibilité d'erreurs n'exprime pas totalement en quoi de telles technologies portent atteinte à la vie privée.

De manière empirique, nous pouvons remarque que certaines personnes pensent qu'il est incompatible de lié espace publique et vie privée, puisque le premier ne pourraient pas exister dans le deuxième, et donc que la surveillance n'est pas un problème. Pourtant, dans son essai de 1998, l'auteure Helen Nissenbaum argumente le contraire. Elle affirme que la récole et le stockage d'informations à l'aide de dispositifs automatisés amène souvent à la violation de la vie privée.

Pour appuyer ses propos, elle se base sur deux prémises : Premièrement, en sachant que la plupart des gens se sentent surpris ou mécontents quand ils apprennet que leur données personnelles ont été collectées à leur insu, même dans un espace publique, elle prouve que ces personnes s'attendaient à un certain respect de leur vie privée, même dans des lieux de vie communs.

Deuxième, elle affirme que l'automatisation de la récolte des données est bien plus problématique qu'une surveillance «manuelle» dans l'espace publique, et ce à cause de deux pratiques rendues possibles par la technologie. La première pratique rendue possible est le changement de contexte de l'information une fois enregistrée (via la vente de données recoltées par exemple). Or l'être humain donne beaucoup d'importance au contexte quand il délivre une information volontairement (on parle de ses finances à son banquier, de ses problèmes de santé à son médecin, mais pas inversemment). La journalisation des données amène alors à de potentielles atteintes à la vie privée. La deuxième pratique est « l'aggrégation des données» ou le Big Data. L'observation individuelle de certaines actions ou habitudes n'a pas vraiment de conséquences, mais lorsque ces systèmes sont capables de mettre en lien beaucoup d'information et d'en tirer des corrélation (par exemple à l'aide du machine learning) de nouvelles informations peuvent être déduites sans le consentement explicite du sujet, menant à une autre forme d'atteinte.

3.7 Conclusion

Nous avons examiné certaines implications morales d'un système de surveillance et de récolte de données, en particulier sur la technologie de la reconnaissance faciale. Parmi les problèmes principaux, nous avons identifié : l'atteinte à la vie privée, de dérapage fonctionnel, et la potentialité d'erreur.

Au moment de sa conception, WiFace ne dispose pas d'un use case établi. Il s'agit d'un outil, sans cadre prédéfini. Ce constat nous permet d'affirmer qu'il est donc fortement sujet au dérapage fonctionnel, puisque n'importe quelle personne pourrait facilement déployer cette

solution à bas coût pour son bénéfice personnel.

À la fin de ce travail, WiFace doit être amené au stade de prototype et non de projet fini. Ce statut atteste de son manque de fiabilité potentiel. Il est donc probable ques on utilisation amène à des erreurs, ce qui pose un autre problème éthique.

L'atteinte à la vie privée semble inhérente à tout système de collecte de données automatisée, WiFace ne faisant pas exception. En permettant de lier un terminal à un utilisateur, nous pouvons facilement imaginer des situations où cette information pourrait être sortie de son contexte (publicité ciblée, statistiques de fréquentation). Un autre objectif secondaire de ce travail est la recherche d'informations sur une identité, menant obligatoirement à l'agglomération de ces données.

D'après ces critères, nous pouvons affirmer que l'utilisation de ce travail ne respecte pas les standards d'éthique et de déontologie qu'un établissement comme la HEIG-VD veut s'imposer (conformémant à la charte d'éthique et de déontologie [1]). Seuls les objectifs pédagogiques et d'évaluation doivent être visés, et en aucun cas ce travail ne devra mener à la commercialisation d'un futur produit.

Chapitre 4

Étude de marché

4.1 Nano-ordinateurs

Afin de pouvoir faire fonctionner le prototype développé, il est nécessaire d'utiliser des nano-ordinateurs (un par caméra), ce qui offre le meilleur compromis entre puissance de calcul, mobilité et prix d'acquisition.

Voici les critères permettant de choisir le modèle à utiliser :

- Installation libre d'une distribution Linux compatible avec les outils à utiliser (librairies, langages, drivers)
- Compatibilité hardware avec les cartes réseaux 802.11 et les caméras choisies au point suivant
- Puissance processeur et RAM suffisante pour exécuter les tâches les plus lourdes du projet (Machine Learning, reconnaissance faciale)
- Prix adéquat et compatible avec le budget associé au travail de Bachelor
- Documentation fournie et/ou communauté active

Voici un tableau comparatif des solutions proposées

	Installation libre	**Compatibilit caméra**	**Compatibilité antenne 802.11**	**Puissance processeur**
NanoPC-T4	Partielle	Module caméra existant	Oui	Dual-Core Cortex-A72(up to 2.0GHz) + Quad-Core Cortex-A53(up to 1.5GHz)
Raspberry Pi 4	Oui	Module caméra existant	Oui	Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
ODROID-XU4	Partielle	USB-CAM	Oui	Samsung Exynos5 Octa ARM Cortex™-A15 Quad 2Ghz and Cortex™-A7 Quad 1.3GHz CPUs

	RAM	Prixsans extension	Qualité de la documentation
NanoPC-T4	Dual-Channel 4GB LPDDR3-1866	Env. 150 dollars	Un wiki
Raspberry Pi 4	1GB, 2GB or 4GB LPDDR4-3200 SDRAM	Env. 65 dollars (Modèle 4GB)	Communauté très active Beaucoup de documentation Beaucoup de tutoriel indépendant
ODROID-XU4	2Gbyte LPDDR3 RAM at 933MHz	Env. 80 dollars	Quelques user guides Un wiki

Au vu de la qualité de la documentation, du prix raisonnable du modèle possédant 4GB et de tous les autres critères remplis, la meilleure alternative pour le projet semble être la **Raspberry Pi 4**.

4.2 Caméra pour la reconnaissance faciale

Afin de pouvoir enregistrer des images, pour les traiter et ainsi reconnaître des visages, une caméra devra être associée à la raspberry pi. Ce use-case étant fréquent, un module caméra officiel est proposé à la vente. Voici quelques spécifications :

Prix	Environ 25 dollars
Résolution	8 Megapixels
Modes vidéo	1080p30, 720p60 and 640 × 480p60/90
Driver Linux	V4L2 driver

Au vu des nombreux projets open-source utilisant ce module pour faire de la reconnaissance faciale, et du fait qu'il soit un produit agréé, ce dernier sera choisi pour le projet. Il existe également une version V1 de ce module, proposant des performences moindres, pour un prix sensiblement identique.

4.3 Antenne 802.11

Afin de pouvoir sniffer les probe requests, il faudra une carte WiFi capable de se mettre en mode « Monitor » (pour recevoir les paquets qui ne sont pas directement adressé à l'adresse de la Raspberry). Comme il sera nécessaire que la raspberry soit également connectée à internet pour envoyer des données, une antenne supplémentaire sera utilisée. L'école possède déjà des antennes AWUS036H compatibles, qui permettront d'effectuer le sniffing.

Chapitre 5

Conception

5.1 La base de donnée

Comme des probes requests, des images, et des données personnelles vont être enregistrées et que du traitement logique leur sera associé, il est important de concevoir une base de donnée claire et bien structurée.

5.1.1 Modèle entité-association

Plusieurs versions du modèle EA ont été imaginée avant d'arriver à la version finale. Afin de mieux cerner les enjeux de la structure de la base de donnée, ces dernières vont être commentées. Mais d'abord, listons les différentes entités

Vendor : Il s'agit du fabricant qui possède un OUI. Étant unique, cet identifiant servira comme clé de la table.

MACAddress : Représente une adresse MAC. Un booléean lui est associé pour savoir si elle semble aléatoire. Une adresse MAC étant unique (ou presque) elle est utilisée comme clé de la table

Probe : Représente une probe request. L'heure à laquelle elle a été capturée ainsi que le SSID visé font partis de l'entité.

Place : Représente l'endroit où la capture a eu lieu.

Picture : Représente une photo prise par le module caméra

Identity : Représente l'identité d'une personne. On y trouvera les données personnelle récupérée via différents vecteurs.

5.1.1.1 Première version

Dans cette première ébauche, toutes les entités sont reliées de manière plutôt simpliste. Regardons toutefois quelques-unes de ces relations. Une entité « vendor » peut n'avoir délivré aucune adresse MAC puisqu'ils seront tous inséré dans la base de donnée à sa création.

Dans ce modèle, une « Identity » est reliée à aucune ou plusieurs MACAddress et photo. (L'individu peut posséder plusieurs appareils et plusieurs photo ont pu être prise)

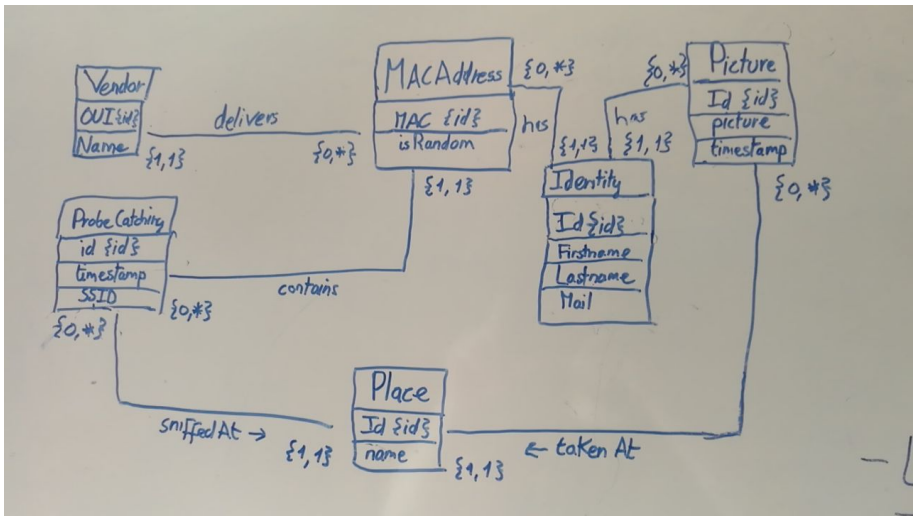

FIGURE 5.1 – Première version entité-association

Problème : Dans cette conception, il n y a aucune notion de probabilité. Quand une identité a été reliée à une adresse MAC par exemple, il n'est pas possible de modéliser le doute. Or, dans notre projet, il n y aura que très peu d'occurence où la certitude est présente. Il faut donc modéliser cette propriété de probabilité.

5.1.1.2 Deuxième version – Modélisation des probabilités

Des relations intermédiaires ont été ajoutées. L'association entre une adresse MAC ou une photo avec une identité est maintenant « pondérée » par une probabilité.

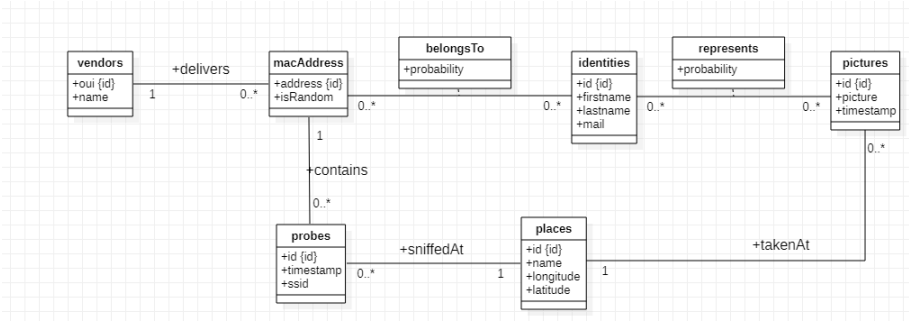

FIGURE 5.2 – Deuxième version entité-association

Nouveau problème : Dans ce schéma, il est possible d'obtenir une identité sans image. Or, d'après les spécifications, ce n'est pas possible puisque c'est grâce à la recherche inversée qu'une identité est établie.

5.1.1.3 Troisième version – Identité à partir d'une photographie

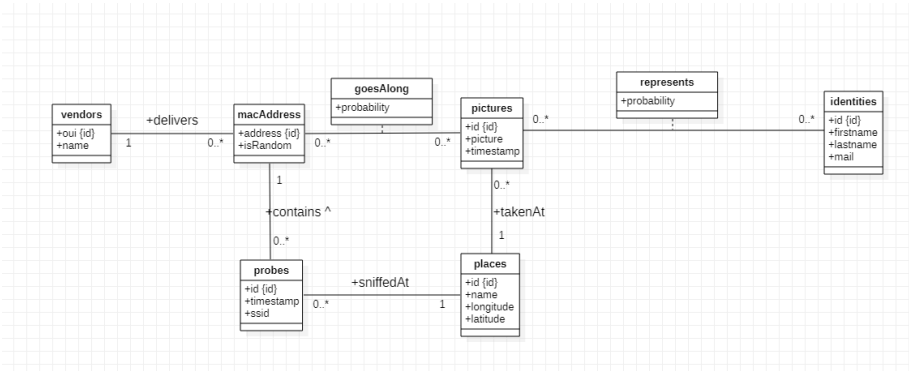

FIGURE 5.3 – Troisième version entité-association

Le problème susmentionné a été résolu en détachant l'identité de l'adresse MAC. Seul la photo y est attachée, et c'est maintenant le lien entre une adresse et une image qui est pondéré.

5.2 API rest

Pour des raisons d'évolutivité, de sécurité et de workflow, il a été décidé de développer une API pour interroger la base de donnée depuis les clients. Ainsi, la responsabilité d'effectuer

la grande partie de la logique métier pourra être distribuée au serveur.

5.3 L'API WiFace

Afin de développer un projet plus évolutif, sécurisé, et scalable, il a été décidé d'implémenter une API rest permettant les opérations CRUD et d'autres traitement sur les entités présentée dans la section 5.1.

La liste des routes est donnée en annexe de ce document.

5.4 Le client Raspberry

Les différents clients Raspberry auront la responsabilité de collecter les informations, et de les envoyer au serveur principal via l'API.

Pour se faire, elles exécuteront en continu deux processus :

1. scanProbe : Récupère les données 802.11 (extraction des MAC Address, du constructeur, analyse de l'aléatoire)
2. recognizeFace : Analyse des frames de la caméra, si un visage est détecté localement par OpenCV, envoi de l'image à l'API

Ce même client peut être dupliqué autant de fois que nécessaire sans adaptation.

5.5 Dashboard : Visualisation des informations

Afin qu'un opérateur puisse observer les données récoltées, les données reçues par l'API sont disponibles sous forme de front-end.

Les pages présentes sont les suivantes :

Statistiques : Diverses informations générales, fil d'événement, pourcentages

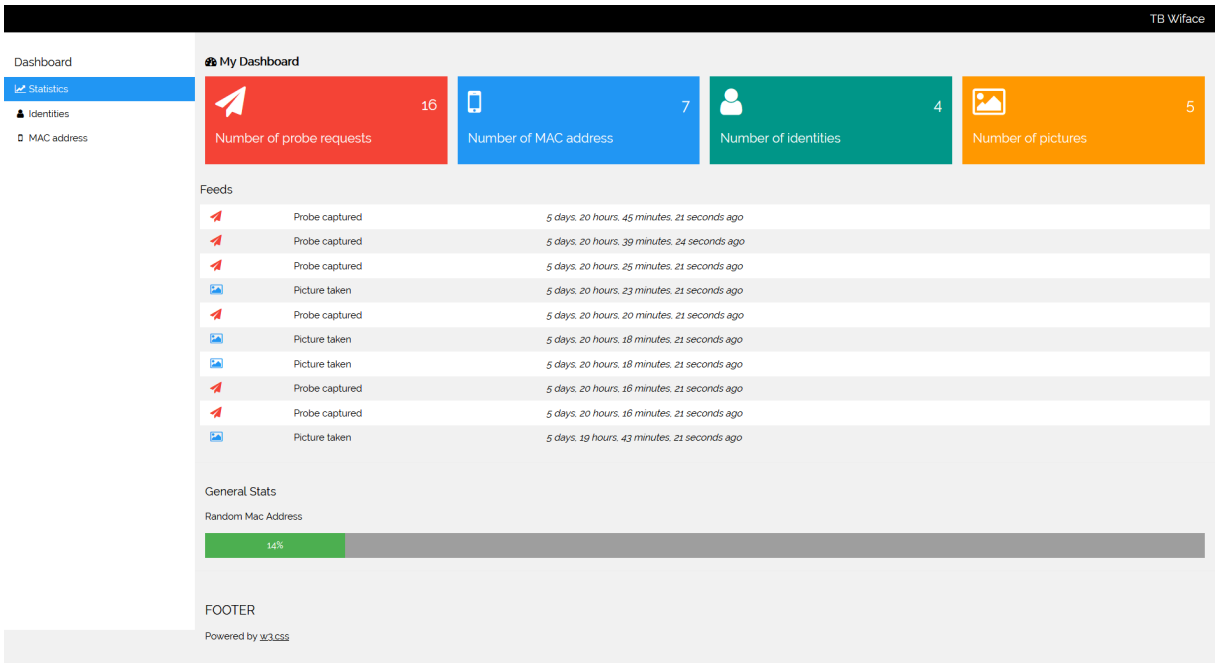

FIGURE 5.4 – Page de statistiques générales

Liste des identités : Aggrégation des identités connues (Prénom, nom, adresse mail, UUID)

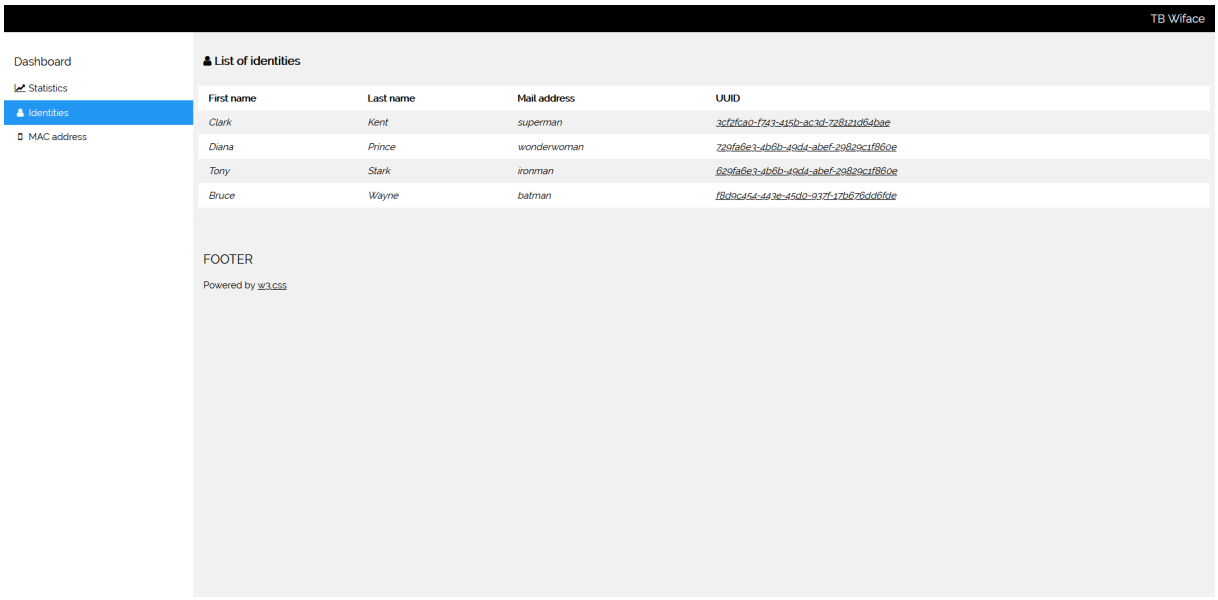

FIGURE 5.5 – Page listant les identités

Détails sur une identité : Toutes les informations concernant une identité spécifique. Liste des adresse MAC probablement associées.

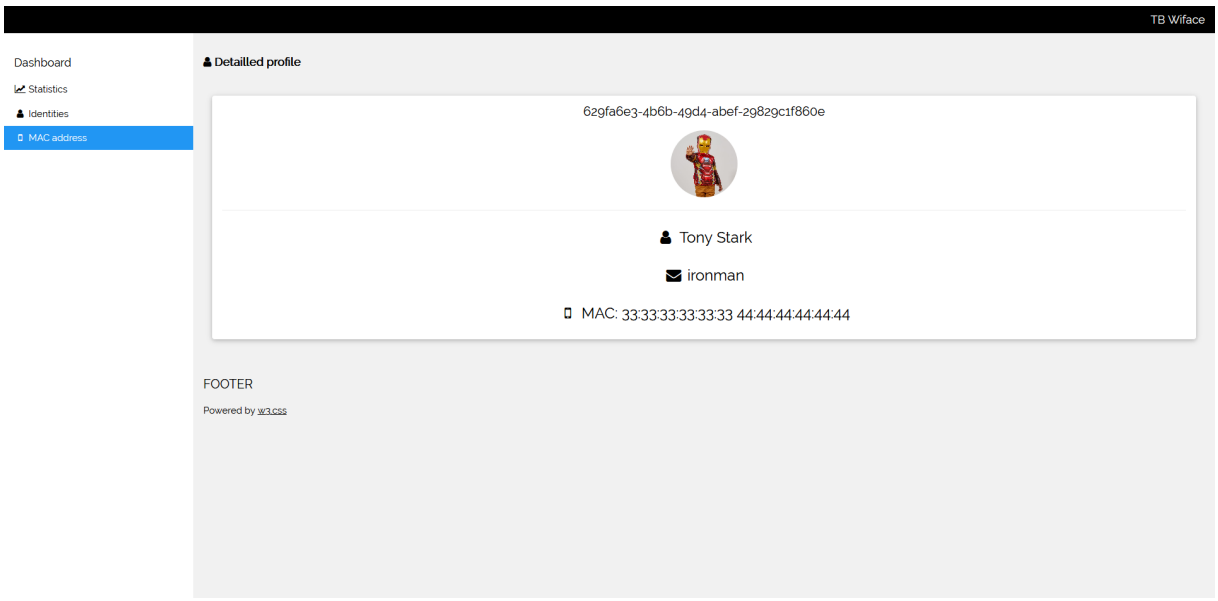

FIGURE 5.6 – Page de détails sur une identité

Liste des adresses MAC : Aggrégation des adresses MAC connues (Adresse, constructeur, Randomness, Nombre d'occurrences dans les probes request)

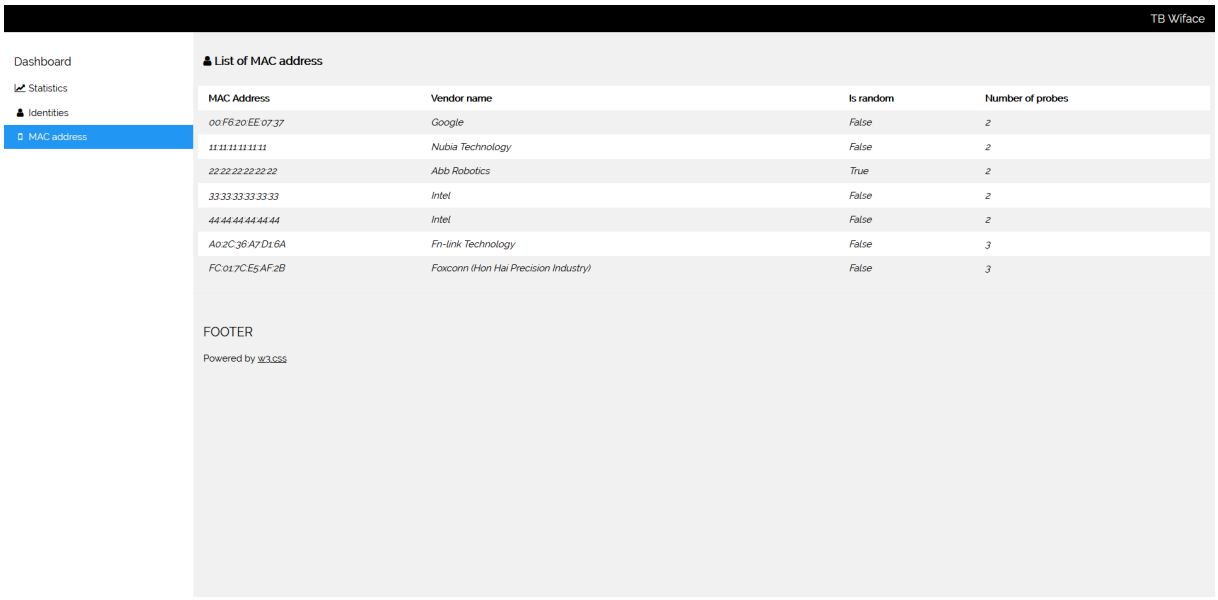

FIGURE 5.7 – Page listant les adresses MAC

5.6 Algorithme PP2I

L'algorithme PP2I (Probe requests and Pictures to Identity) a la responsabilité d'associer une ou plusieurs adresses MAC à une identité.

Afin d'en analyser le fonctionnement, un lexique doit être établi :

- Identité : Personne unique reconnue par l'API Rekognition d'Amazon.
- Fenêtre : Interval temporelle centré autour d'un événement.

Le processus va se dérouler en quatre étapes.

1. Initialisation et incrémentation
2. Décrémentation majeure dûe à l'absence de l'adresse MAC
3. Décrémentation mineure dûe à l'absence de la photo
4. Normalisation des scores

5.6.1 Initialisation

Dans cette phase de l'algorithme, toutes les identités sont parcourues. Pour chaque identité, toutes les photos correspondantes sont mises en relation avec probes requests. Si un couple

(photo, probe request) se trouve dans la même fenêtre, alors on initialise le couple et on l'ajoute au dictionnaire.

Algorithm 1: Initialisation et création de couples

Input: List of pictures timestamped and labeled with corresponding identity, list of probe request timestamped

Result: Dictionnary containing (adress, identity) tuple as key and probability as value

```
dict_couple ← ∅;
foreach Identity in AllIdentities do
    foreach Picture in Identity do
        beginning ← Picture.timestamp − half_window_duration;
        end ← Picture.timestamp + half_window_duration;
        place ← Picture.place;
        foreach Probe taken at place between beginning and end in allProbes do
            dict_couple.add(key = (address, identity), value = init_score)
        end
    end
end
return dict_couple
```

5.6.2 Décrémentation majeure dûe à l'absence de l'adresse MAC

Dans cette phase, nous regardons pour un couple donné, si il y a des instances où l'on trouve l'identité (une photo) sans l'adresse MAC (une probe request). Comme c'est un cas peu probable si le couple est correct (une probe request est plus facilement récupérable qu'une

photo), on descend beaucoup le score du couple si cela arrive.

Algorithm 2: Décrémentation majeure dûe à l'absence de l'adresse MAC

Input: *dict_couple*, list of pictures timestamped and labeled with corresponding identity, list of probe request timestamped
Result: Dictionnary containing (adress, identity) tuple as key and probability as value

```
mac_addresses ← ∅;
foreach Couple in dict_couple do
    foreach Picture representing Couple[Identity] do
        beginning ← Picture.timestamp − half_window_duration;
        end ← Picture.timestamp + half_window_duration;
        place ← Picture.place;
        foreach Probe taken at place between beginning and end in allProbes do
            mac_addresses.add(Probe.mac)
        end
    end
    if Couple[mac] not in mac_addresses then
        dict_couple[(adress, identity)] -= BigMalus;
    end
end
return dict_couple
```

5.6.3 Décrémentation mineure dûe à l'absence de la photo

Dans cette phase, nous regardons pour un couple donné, si il y a des instances où l'on trouve l'adresse MAC (une probe request) sans l'identité (une photo). Comme c'est un cas fréquent,

on ne baissera qu'un peu le couple donné.

Algorithm 3: Décrémentation mineure dûe à l'absence de la photo

Input: *dict_couple*, list of pictures timestamped and labeled with corresponding identity, list of probe request timestamped
Result: Dictionnary containing (adress, identity) tuple as key and probability as value

```
identities ← ∅;
foreach Couple in dict_couple do
    foreach Probe containing Couple[address] do
        beginning ← Probe.timestamp − half_window_duration;
        end ← Probe.timestamp + half_window_duration;
        place ← Probe.place;
        foreach Picture taken at place between beginning and end in allPictures
         do
            identities.add(Picture.identity)
        end
    end
    if Couple[identity] not in identities then
        dict_couple[(adress, identity)] -= BSmallMalus;
    end
end
return dict_couple
```

5.6.4 Normalisation [26]

Afin de faciliter le traitement, les scores seront normalisés dans une intervalle [0 ;1]. Pour ce faire, toutes les adresses candidates à une identités sont sélectionnées et mises à l'échelle entre elles à l'aide de la formule : suivante

Listing 5.6.1: Formule de normalisation

```
X_std = (X - X.min) / (X.max - X.min)
X_scaled = X_std * (max_range - min_range) + min_range
```

Chapitre 6

Reconnaissance faciale

Une partie critique du projet consiste à pouvoir identifier des personnes en fonction des images capturées par la caméra. La reconnaissance faciale étant un sujet d'étude très complexe, il sera nécessaire d'utiliser des solutions clés en main afin de pouvoir l'incorporer dans l'ensemble du projet. Il peut toutefois être intéressant de comprendre les bases et les implications de cette technologie.

6.0.1 Une petite touche de théorie

La reconnaissance faciale utilise la technologie du Machine Learning, et plus précisement du Deep Learning. Le principe consite à l'extraction de données intéressantes d'une image (les caractérstiques) afin de lier une « empreinte biométrique » à un visage. Par exemple en mesurant l'écartement des yeux, la longeur du nez, la profondeur des orbites, on obtient des données uniques qui permettront d'identifier la même personne sur une image différente.

Les cas d'utilisation de la reconnaissance faciale sont nombreux. En voici une liste non-exhaustive :

- Identifier l'utilisateur d'un téléphone (Apple FaceID)
- Aide à la recherche des personnes disparues
- Aide aux personnes malvoyantes (détection et notification du sourire de ces interlocteurs)
- Labélisation des personnes sur les réseaux sociaux
- Lutte contre le terrorisme

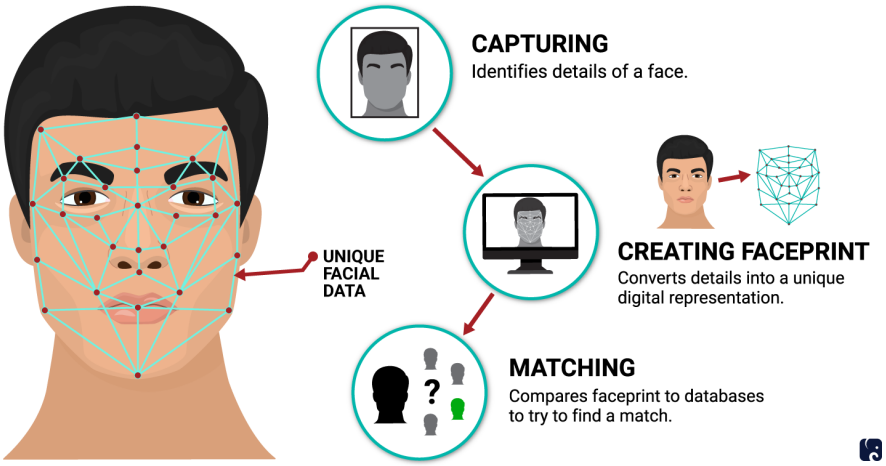

FIGURE 6.1 – Processus de reconnaissance faciale

6.1 Choix de la solution

À l'aide de quelques valeurs clés (précision, rappel, nombre de faux-positifs, ...) nous allons comparer plusieurs solutions existantes et "clés en main" de reconnaissance faciale.

	Kairos	Amazon	Google	Microsoft	IBM	Affectiva	OpenCV
Face Detection	✓	✓	✓	✓	✓	✓	✓
Face Recognition (Image)	✓	✓	✗	✓	✗	✗	✗
Face Recognition (Video)	✓	✓	✗	✗	✗	✗	✓
Emotional Depth (%)	✓	✗	✗	✓	✗	✓	✗
Emotions Present (Y/N)	✓	✓	✓	✓	✗	✓	✗
Age & Gender	✓	✓	✗	✓	✓	✓	✗
Multi-face Tracking	✓	✓	✓	✓	✓	✓	✓
SDK	✓	✗	✗	✗	✗	✓	✓
API	✓	✓	✓	✓	✓	✓	✗
Ethnicity	✓	✗	✗	✗	✗	✗	✗

FIGURE 6.2 – Tableau comparatif de solution de reconnaissance faciale

Ce qui est nécessaire, c'est la « Face recognition » sur image (plus compatible avec la notion d'IoT qu'un flux vidéo complet.) Les trois solutions le proposant sont :

- Kairos
- Amazon AWS : Rejognition
- Microsoft Azure : Face

Trois facteurs sont alors important dans le cadre du projet : La précision, le prix et la facilité d'utilisation.

6.1.1 Performences

Concernant la précision, on peut le mesure à l'aide des concepts de True Positive (Un visage est correctement identifié), True Negative (Un visage est correctement exclu), False Positive (Un visage est identifié en tant qu'un autre) et False Negative (Un visage n'a pas été identifié comme connu)

Une source a utilisé un dataset de visage de l'université de Essex afin de mesurer ces valeurs dans différents contextes pour les trois solutions. Voici les résultats principaux.

- Precision (TP/(TP+FP)
- Recall (TP/(TP+FN))

Provider	**TP**	**FP**	**TN**	**FN**	**Precision**	**Recall**
AWS rekognition	149	0	150	1	100%	99.3%
Microsoft cognitive services	131	0	150	19	100%	87.3%
Kairos	108	0	150	42	100%	72%

TABLE 6.1 – Seuil de confiance AWS rekognition : 95% Microsoft : 80% Kairos : 95%

Provider	**TP**	**FP**	**TN**	**FN**	**Precision**	**Recall**
AWS rekognition	150	2	148	0	98.7%	100%
Microsoft cognitive services	137	0	150	13	100%	91.3%
Kairos	148	0	150	2	100%	98.7%

TABLE 6.2 – Seuil de confiance AWS rekognition 70% Microsoft : 50% Kairos : 70%

Provider	**TP**	**FP**	**TN**	**FN**	**Precision**	**Recall**
AWS rekognition	150	2	148	0	98.7%	100%
Microsoft cognitive services	137	10	140	13	93.2%	91.3%
Kairos	150	19	131	0	88.7%	100%

TABLE 6.3 – Seuil de confiance AWS rekognition : 50% Microsoft : 30% Kairos : 50%

Parmi ces résultats, nous pouvons voir que rekognition se démarque par son absence presque totale de FN et un taux faible de faux positifs, même avec un seuil de confiance bas (ce qui est un avantage dans un environnement moyennement précis tel que la raspberry).

Instance	Transactions Per Second (TPS) *	Features	Price
Free	20 transactions per minute	Face Detection Face Verification Face Identification Face Grouping Similar Face Search	30,000 transactions free per month
Standard	10 TPS	Face Detection Face Verification Face Identification Face Grouping Similar Face Search Face Storage	0-1M transactions - $1 per 1,000 transactions 1M-5M transactions - $0.80 per 1,000 transactions 5M-100M transactions - $0.60 per 1,000 transactions 100M+ transactions - $0.40 per 1,000 transactions $0.01 per 1,000 faces per month

6.1.2 Tarification

Concernant la tarfication, Amazon et Microsoft se démarque de Kairos grâce à leur prix moins élevés, et à des solutions gratuites.

6.1.2.1 Kairos

Kairos pratique des prix mensuels fixes ainsi qu'un supplément par transaction. Par exemple, l'offre « Student Cloud » coûte 19$par mois + 0.02$ par transaction, avec tout de fois 2 semaines d'essai gratuites.

6.1.2.2 Microsoft Azure Face

Microsoft propose deux solutions : « Free - Web/Container » et « Standard - Web/Container ». Il est à noté que la tier gratuit ne comprend pas le « Face Storage » et que l'entraînement est compté dans les transactions (1 transaction / 1000 images utilisées pendant l'entrainement)

6.1.2.3 Amazon AWS rekognition

Rekognizion est compris dans l'offre gratuite d'AWS durant les 12 premiers mois d'utilisation et ce pour 5'000 transaction ainsi que 1'000 métadonnées par mois.

6.1.3 Facilité d'utilisation

Concernant la facilité d'utilisation, une documentation de bonne qualité semble disponible pour Microsoft et Amazon. Toutefois, il semble très important pour Microsoft d'avoir des

Type de coût	Tarification	Prix pour 1 000 images
Premier million d'images traitées par mois	0,001 USD par image	1,00 USD
9 millions d'images traitées suivants par mois	0,0008 USD par image	0,80 USD
90 millions d'images traitées suivants par mois	0,0006 USD par image	0,60 USD
Plus de 100 millions d'images traitées par mois	0,0004 USD par image	0,40 USD

photos d'entraînements avant le processus de reconnaissance, ce qui ne répond pas au cahier des charges.

6.1.4 Choix de la solution

Grâce aux différents facteurs analysés, c'est la solution d'Amazon (Rekognition) qui a été retenue pour ce projet. À l'aide d'un tiers gratuit (majoritairement suffisant pour la période de développement), de très bonne performences et d'un use case compatible avec le cahier des charges, nous pouvons affirmer qu'il est pertinent de faire ce choix.

6.2 Amazon Rekognition : présentation

Comme expliqué précédemment, Amazon rekognition est un service de reconnaissance d'image. Bien que nous utilisons seulement les fonctionnalités concernant les visages, il est aussi possible par exemple d'identifier du texte, des objets ou encore des images inappropriées (contenu explicite et suggestif).

6.2.1 Concepts et opérations

Pour pouvoir utiliser Rekognition, il faut connaître les opérations principales de l'API et en comprendre les concepts principaux.

Features (caractéristique) : En machine learning, les caractéristiques sont les variables d'entrées permettant de simplifier le problème et de sur lesquels nous allons nous baser pour faire des classifications. Par exemple, pour la détection de visage, certaines des caractéristiques sont : écartement des yeux, couleur des yeux, la longueur du nez, la forme des joues, la profondeur des orbites, ou encore la largeur de la mâchoire. La combinaison de ces caractéristiques permettent souvent d'identifier de manière unique un individu.

Collection : Les collections sont des « conteneurs » AWS server-side qui rassemblent les informations (métadata) des visages ajoutés. Une collection est identifiée par un ARN (Amazon

Resource Name) unique et peut être créée grâce à l'opération « CreateCollection »

IndexFace : Opération d'ajout d'un ou plusieurs visages (métadatas) dans une collection. En entrée, il faut principalement une image (blob), le nom de la collection dans laquelle insérer les visages, le nombre maximum de visages à indexer (les visages sont indexés du plus petit au plus grand. Ainsi, sur une image contenant 5 personnes, si le paramètre vaut 2, seulement les 2 plus grand visages seront ajoutés à la collection)

SearchFacesByImage : Opération prenant une image en entrée, récupère le visage le plus visible et cherche dans une collection donnée si ce visage est reconnu. Si c'est le cas, on peut alors inférer que l'image représente la même personne que les métadonnées correspondantes.

Voici un workflow détaillé d'Amazon Rekognition

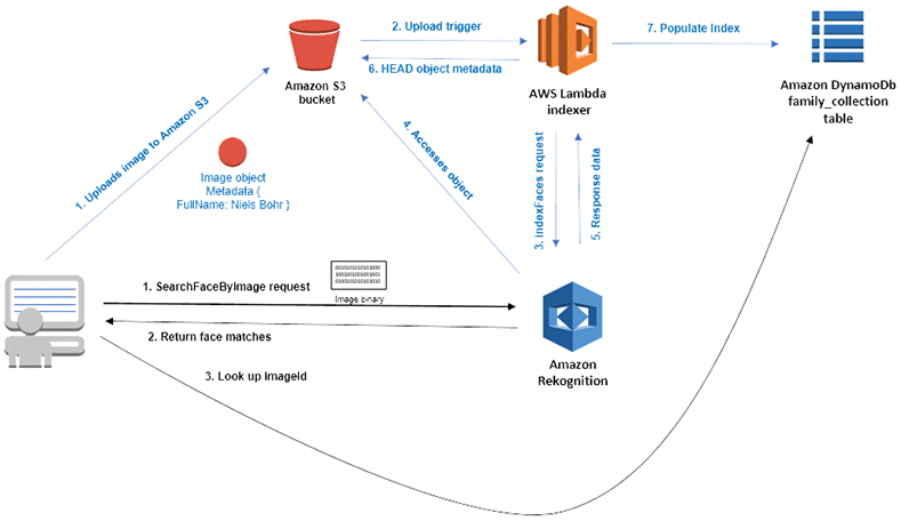

FIGURE 6.3 – Workflow d'Amazon Rekognition

6.2.2 Prérequis

Afin de pouvoir travailler avec Amazon Rekognizion, il est nécessaire d'avoir une clé d'authentification à leur API. Pour cela, il faut créer un compte et s'inscrire au service. Une fois la clé obtenue, nous allons utiliser leur SDK pour Python : boto3 1 et leur CLI : awscli grâce auquel nous pouvons effectuer la configuration de base. Avec la comande 'awsconfigure' nous pouvons saisir la clé qui sera alors utilisée pendant nos appels d'API

Listing 6.2.1: Configuration du client amazon

```
aws configure
AWS Access Key ID [None]:  YOUR_ACCESS_KEY_ID
AWS Secret Access Key [None]: YOUR_SECRET_ACCESS_KEY
Default region name [None]: us-east-1
Default output format [None]: ENTER
```

Chapitre 7

Adresses MAC et probes requests

Afin de pouvoir détecter et tracer les différents périphériques réseau présents pendant les scans, il est impératif de pouvoir les identifier de manière unique et irrévocable.

Pour cela, nous allons utiliser l'adresse MAC. Cet identifiant unique peut être récupéré de plusieurs manières, mais ce sont les **probe requests** qui vont nous offrir la plus grande souplesse.

Explorons ces deux concepts.

7.1 Adresse MAC, un identifiant pas si unique

7.1.1 Définition [27] :

"Une adresse MAC (Media Access Control1), parfois nommée adresse physique, est un identifiant physique stocké dans une carte réseau ou une interface réseau similaire. À moins qu'elle n'ait été modifiée par l'utilisateur, elle est unique au monde.

MAC constitue la partie inférieure de la couche de liaison (couche 2 du modèle OSI). Elle insère et traite ces adresses au sein des trames transmises. Elle est parfois appelée adresse ethernet, UAA (Universally Administered Address), BIA (Burned-In Address), MAC-48 ou EUI-48."

7.1.2 Format

Les 48 bits d'une adresse sont formatés comme suit :

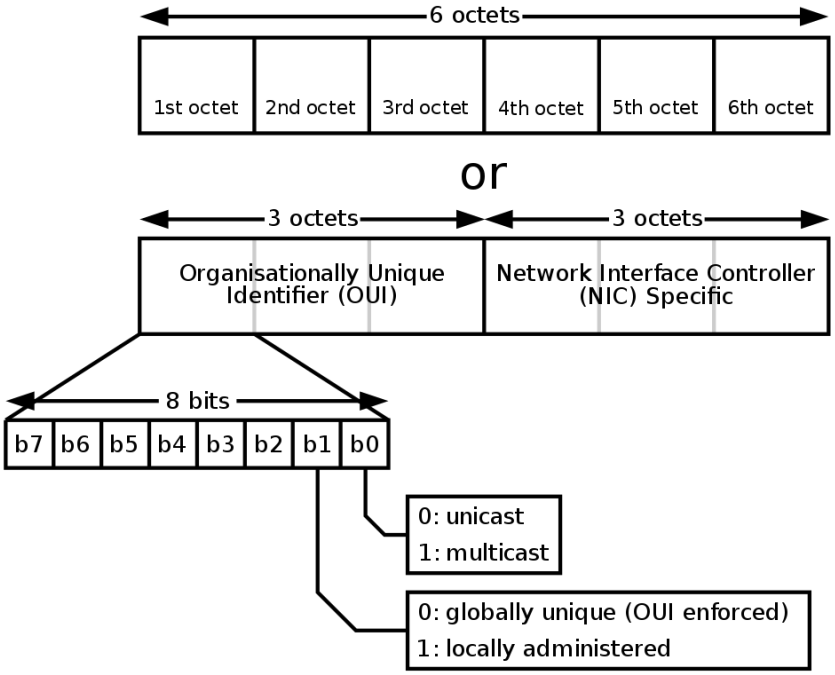

FIGURE 7.1 – Structure de l'adresse MAC

Les 3 octets de poids faible sont variables et change pour chaque carte mère, c'est ce qui rend chaque adresse du même constructeur unique, il s'agit du **NIC**.

Les 3 octets de poids fort sont presque fixes pour chaque constructeur, il s'agit du **OUI**. Toutefois, les deux bits de poids faible du premier octet peuvent varier (b1 et b0 sur l'illustration).

b0 indique si l'adresse est individuelle, auquel cas le bit sera à 0 (pour une machine unique, unicast) ou de groupe (multicast ou broadcast), en passant le bit à 1

b1 indique 0 si l'adresse est universelle (conforme au format de l'IEEE) ou locale, 1 pour une adresse administrée localement (ce bit sera crucial concernant l'analyse de la randomisation)

7.1.3 Problèmes de vie privée et randomisation

La propriété d'unicité des adresses MAC soulève évidemment des problématiques liées à la vie privée. la vie privée. Par exemple, selon les dires d'Edward Snowden, la NSA se servait des adresses MAC afin de monitorer les déplacements d'individus. [16]

Le sujet de ce travail de bachelor soulève les même problématiques. Par conséquent, certaines marques ont mis en place la **MAC Address randomization**.

7.1.4 Randomisation des adresses MAC [6]

Depuis 2014, de plus en plus de constructeurs ont mis en place des mesures afin de protéger la véritable adresse MAC de l'appareil.

- iOS à partir d'iOS 8 ;
- Windows depuis Windows 10 ;
- Android depuis Android 6.0 (un patch gère également Android 5.0 pour certains appareils) ;
- certains drivers Linux depuis le kernel 3.18.

L'objectif étant de varier suffisamment régulièrement l'adresse pour que empêcher le traçage. Le processus de randomisation n'étant pas strictement standardisé, l'implémentation peut varier pour chaque constructeur.

Certaines de ces implémentations sont bonnes (iOS randomise les 6 octets de l'adresse MAC2 sauf b1 et b0) alors que d'autres le sont moins (Android possède des OUI fixes pendant la randomisation, ce qui permet déjà la divulgation d'informations sur l'appareil.)

7.2 Les probes requests

Les adresses MAC, aléatoires ou non, sont transmises dans chaque trame MAC, ainsi que bien d'autres informations. Voici leur structure :

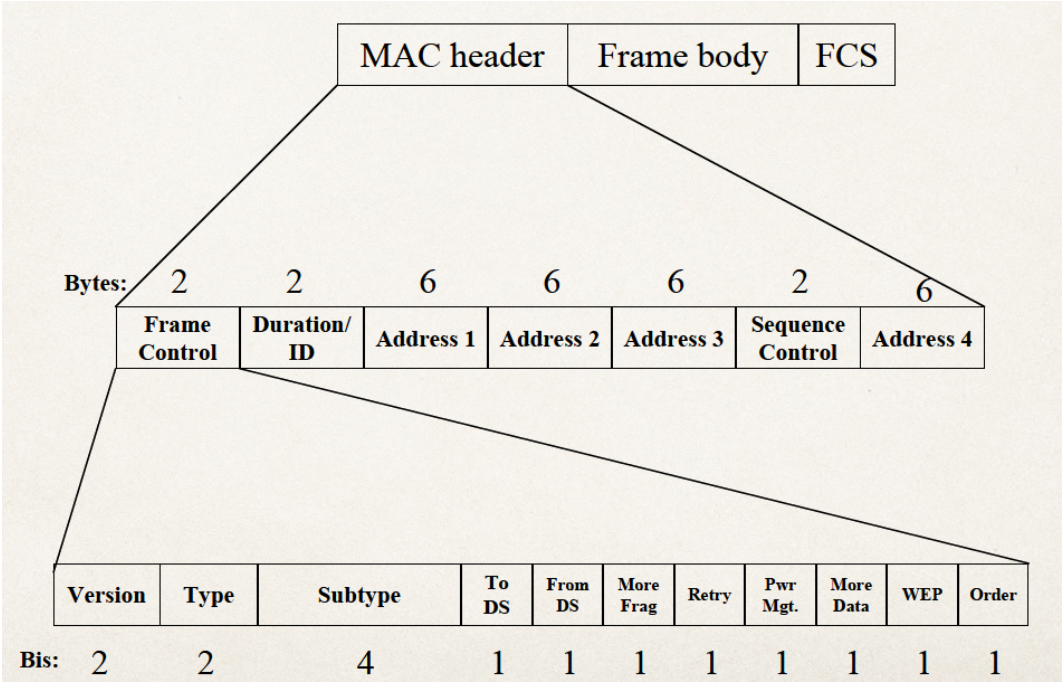

FIGURE 7.2 – Structure d'une trame MAC

On retrouve notamment les adresses MAC de source et de destination dans les champs Address[1|2|3]. Malheureusement, pour qu'un appareil diffuse des trames de **données**, il faut qu'il soit associé avec un point d'accès. Il existe cependant les trames de **management**. Le champs Subtype (cf figure 7.2) définit justement la nature de ces frames.

Voici les trames de management principales :

- Beacon
- Probe request & response
- Authentification
- Association request & response
- Re-association request & response
- Disassociation
- De-Authentification

Les trames **Beacon** sont utilisées par les point d'accès pour signaler leur existence.

Les trames d'**(De-)Authentification** sont utilisées par une station et un point d'accès afin d'établir leur identité (pas de chiffrement à ce stade).

Les trames d'**(Dis|Re)Association** sont utilisées pour lier un AP et une station après l'authentification.

Les trames de **Probing** servent à un client (request) à demander à un AP ses informations nécessaires à la connexion (e.g le canal). L'AP concerné envoie alors une probe response avec les informations.

Une probe request peut être dirigée (un SSID spécifique est spécifié, on l'appelle alors "directed probe request") ou alors aucun SSID n'est spécifié, dans ce cas on l'appelle "null probe request"

Pour notre solution de scanning, nous cherchons une trame qui n'a pas d'interaction direct avec un access point préexistant et qui est envoyée par le client. Parmi les trames de management mentionnées, seul les probe requests respectent ces deux contraintes.

Un autre avantage des probe request est que ces dernières sont envoyées très régulièrement (plusieurs fois par minute) afin de permettre à l'appareil de trouver rapidement les WiFi disponibles. Elles sont même envoyées en rafale ("burst") afin de couvrir un maximum de canaux 802.11.

Voici un exemple de Probe request dirigée capturée avec Wireshark :

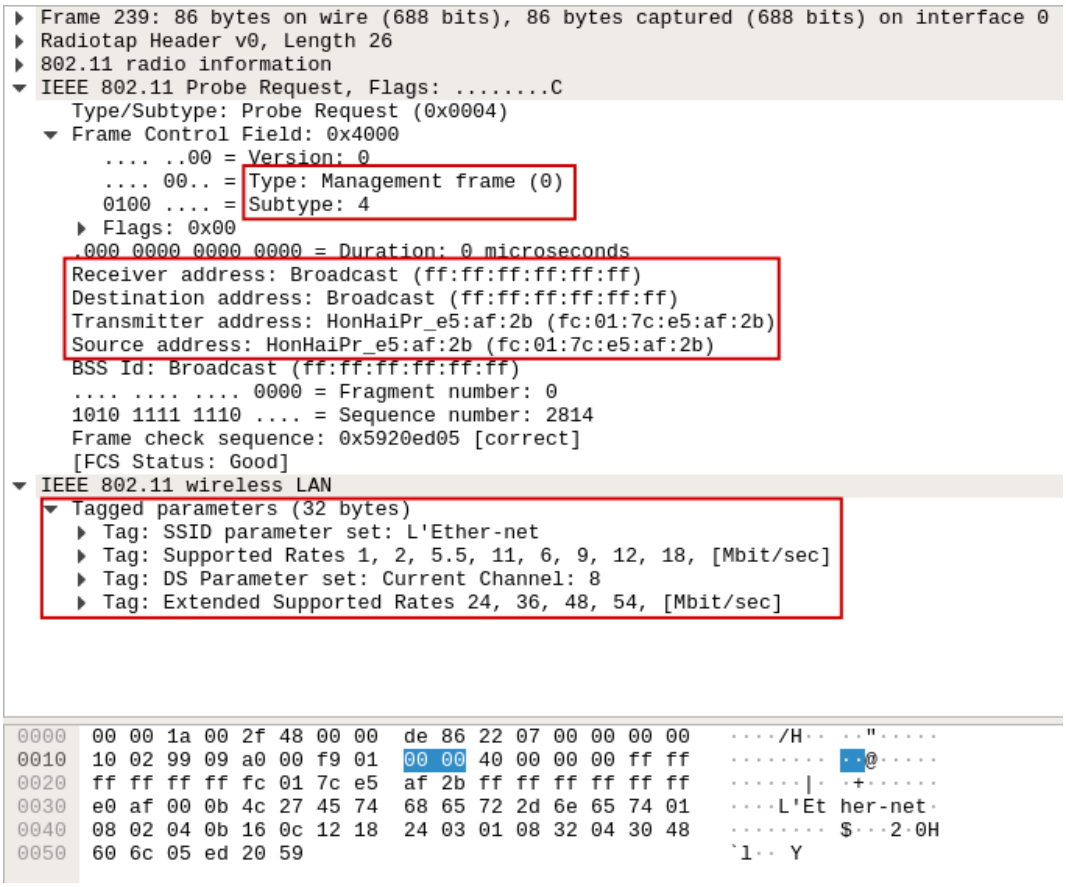

FIGURE 7.3 – Probe request dirigée

On y trouve le sous-type de la trame de management qui identifie les Probe requests (4), l'adresse de destination (broadcast) , l'adresse source (un téléphone Honor 8) et différentes informations dont le **SSID** recherché : "L'Ether-net"

Les même informations sont disponibles pour une null probe request, mais le SSID est remplacé par un wildcard :

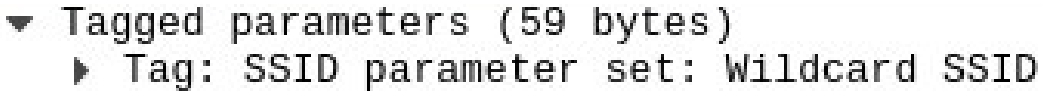

FIGURE 7.4 – null-probe request

Il est intéressant de noter que des informations peuvent parfois être déduites des probes requests dirigées. La plupart du temps, les requêtes sont dirigées car l'appareil s'est déjà

connecté au réseau concerné. Si le nom du réseau est connu et assez unique, il devient alors possible de déduire où l'appareil se situait par le passé. Par exemple, sur la figure 7.4, le SSID est l'ancien nom de mon réseau, je peux donc en déduire qu'il s'agit d'un de mes appareils, et qu'il n'est pas récent.

Cette technique est aussi utilisée afin de découvrir des réseaux cachés (il s'agit de réseaux qui ne répondent pas aux null probe requests et qui n'envoient pas de beacons).

Chapitre 8

Implémentation

8.1 La base de données

8.2 L'API WiFace

8.2.1 Choix de la stack

Du côté serveur, le langage utilisé est le Python (3.7). Pour faciliter le développement d'une API rest, le micro- framework **Flask** a été choisi. Au vu de la documentation et de mon expérience personnelle, ce choix est pertinent dans le cadre de ce projet.

Avantages de Flask :

- Simple et léger
- Convient bien au développement d'application de petite ou moyenne envergure
- Très flexible
- Prise en main rapide
- Compatible avec l'ORM sqlalchemy

L'ORM qui a été choisi pour fonctionner avec Flask est **SQLAlchemy**. Il existe en effet un module python flask- sqlalchemy qui rend l'intégration simple. Les données sont sérialisées à l'aide de **marshmallow**. La gestion de l'authentification se fait à l'aide de token JWT et du module correspondant **flask-jwt** et **pyjwt** La spécification est écrite à l'aide de swagger, qui propose également une documentation automatique.

8.3 Le client Raspberry

8.4 Algorithme PP2I

Chapitre 9

Tests du prototype

Chapitre 10

Conclusion

10.1 Difficultés rencontrées

10.2 Améliorations futures

10.3 Retour personnel

10.4 Remerciements

Bibliographie

[1] Charte d'éthique et de déontologie des hautes écoles universitaire et spécialisée de genève. 2020-04-29.

[2] adneovrebo. Face-recognition-door-opener-raspberry-pi. `https://github.com/adneovrebo/Face-recognition-door-opener-raspberry-pi`, 2019.

[3] Oleg Agapov. Jwt authorization in flask. `https://codeburst.io/jwt-authorization-in-flask-c63c1acf4eeb`, 2017. 2020-04-29.

[4] Philip Brey. Ethical aspects of facial recognition systems in public places. *Journal of Information, Communication and Ethics in Society*, 2(2) :97–109, 2004.

[5] Wiki contributors. Nanopc-t4. `http://wiki.friendlyarm.com/wiki/index.php/NanoPC-T4`, 2020. 2020-04-29.

[6] Matte Célestin. Mac address randomization : Tour d'horizon. `https://connect.ed-diamond.com/MISC/MISC-096/MAC-Address-Randomization-tour-d-horizon`, 2019. 2020-06-17.

[7] dataturks. Comparing the top face recognition apis. `https://dataturks.com/blog/face-verification-api-comparison.php`, 2018. 2020-04-29.

[8] Gouvernement de la Confédération suisse. Loi fédérale sur la protection des données, 2018. 2020-04-29.

[9] Brannon Dorsey. The perils of probe requests. `https://medium.com/@brannondorsey/wi-fi-is-broken-3f6054210fa5`, 2017. 2020-04-29.

[10] Doug Farrell. Python rest apis with flask, connexion, and sqlalchemy. `https://realpython.com/flask-connexion-rest-api/`, 2019. 2020-04-29.

[11] Fondation Raspberry Pi. *Camera Module*, 2020. 2020-04-29.

[12] Fondation Raspberry Pi. *Setting up your Raspberry Pi*, 2020. 2020-04-29.

[13] Patrick Grother, Mei Ngan, and Kayee Hanaoka. Face recognition vendor test part 3 :. Technical report, December 2019.

[14] hash3liZer. Wifi karma : A brief guide on probe response frames. `https://www.shellvoide.com/wifi/wifi-karma-a-brief-guid-on-probe-response-frames/`, 2019. 2020-04-29.

[15] Hande Hong, Girisha Silva, and Mun Chan. Crowdprobe : Non-invasive crowd monitoring with wi-fi probe. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2 :1–23, 09 2018.

[16] Swati Khandelwal. Spying agencies tracking your location by capturing mac address of your devices. `https://thehackernews.com/2014/01/spying-agencies-tracking-your-location_31.html`, 2019. 2020-06-17.

[17] Oisín Kyne. Mac address de-anonymisation, 5 2017. 2020-04-29.

[18] Jeremy Martin, Travis Mayberry, Collin Donahue, Lucas Foppe, Lamont Brown, Chadwick Riggins, Erik Rye, and Dane Brown. A study of mac address randomization in mobile devices and when it fails. *Proceedings on Privacy Enhancing Technologies*, 2017, 03 2017.

[19] Microsoft. Cognitive services pricing - face api. `https://azure.microsoft.com/en-us/pricing/details/cognitive-services/face-api/`, 2020. 2020-04-29.

[20] Moin Nadeem. How we tracked and analyzed over 200,000 people's footsteps at mit. `https://www.freecodecamp.org/news/tracking-analyzing-over-200-000-peoples-every-step-at-mit-e736a507ddbf/`, 2017. 2020-04-29.

[21] Gangadharan Natarajan. How to install opencv on raspberry pi 3b+. `https://nerdynat.com/programming/2019/how-to-install-opencv-on-raspberry-pi-3b/`, 2019. 2020-04-29.

[22] Cade Metz Natasha Singer. Netherlands 'will pay the price' for blocking turkish visit – erdoğan, 2017.

[23] WiFi Nigel. Randomized mac addresses in 802.11 probe frames. `https://wifinigel.blogspot.com/2018/04/to-address-perceived-privacy-issues.html`, 2018. 2020-04-29.

[24] nlm. macaddr.py. `https://gist.github.com/nlm/9ec20c78c4881cf23ed132ae59570340`, 2017.

[25] odroid. *USER MANUAL ODROID-XU4*. 2020-04-29.

[26] sklearn. *sklearn.preprocessing.MinMaxScaler¶*. 2020-06-16.

[27] Wikipédia. Adresse mac — wikipédia, l'encyclopédie libre, 2020. [En ligne ; Page disponible le 3-mars-2020].

Table des figures

Liste des tableaux