MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITE ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : **Ingénierie des Systèmes d'Information**

THEME :

La sécurité dans les systèmes de smart health

Etudiant(e) : « **Boulenouar saliha** »

Encadrant(e) : « MIROUD Mohamed El Mustapha »

Dédicace

Je dédie ce modeste travail :

À ceux qui ont sacrifie toute leur vie pour ma réussite, et qui n'ont guère finis de veiller a mon bien être, à mes très chers parents, et je prie DIEU de les protéger et les garder pour moi.

À mes frères et mes sœurs pour toute leur compréhension et encouragements dans la

réalisation de ce travail.

À toutes mes amies et en particulier à : YAMNA, SAMIA, HAYET et tous ceux qui me sont chers.

Merci à tous et à toutes.

Saliha

Remerciement

En tout premier lieu, je remercie le bon Dieu, tout puissant, de m'avoir donné la force pour survivre, ainsi que l'audace pour dépasser toutes les difficultés.

Jetender à exprimer toute ma reconnaissance à mon Directeur de mémoire Monsieur

Miroud Mohammed El Mustapha. *Je le remercie de m'avoirencadré, orienté et aidé. J'ai profité pendant longtemps du savoir et du savoir-faire dont j'ai pu bénéficier au cours de nombreuses discussions.J'aimerais aussi leremercier pour l'autonomie qu'il m'a accordé, et ses précieux conseils qui m'ont permis de mener à bien ce travail.*

J'adresse mes sincères remerciements à tous les intervenants et toutes les personnes qui par leurs paroles, leurs contributions dans ce travail, leurs écrits, leurs conseils et leurs critiques ont guidé mesréflexions et ont accepté de répondre à mes questions durant mes recherches. Je ne peux achever ce projet, sans exprimer mes sincères gratitudes à tous les professeurs de notre faculté, pour leur dévouement et leur assistance tout au long de ma formation.

Je profite de cette tribune pour remercier a tous nos enseignant (e) du parcours de

informatique pour leur effort fournis durant toute la période d'étude ainsi qu'a tout ceux qui ont collaborés d'une façon ou d'une autre à l'élaboration de ce travail.

Enfin je remercie les membres du jury qui ont bien voulu accepter, et ce nonobstant, leur lourdes et exaltantes responsabilités pour procéder à l'évaluation de ce modeste travail.

Liste des figures

Liste des tableaux

Liste des abréviations

Table des matières

Introduction Générale

Smart cité, smart leaving, smart health, smart building, tous ces termes apparus il n'y a pas si longtemps constituent une nouvelle façon de vivre, de se déplacer, de se soigner … pour le citoyen. Ces nouveaux modes de vies sont en général basés sur l'utilisation des technologies de l'informations et de la communication surtout avec l'avènement des réseaux sans fil à très haut débit et la démocratisation des capteurs dans tous les domaines.

La smart health peut être défini comme une utilisation particulière de la e-santé afin de permettre aux patients de se faire soigner sans pour autant se déplacer systèmatiquement. La e-santé est elle-même défini par l'utilisation des nouvelles technologies de l'information et de la communication afin de transmettre, stocker ou bien encore gérer des informations à caractère médical.

Le thème principal de ce mémoire est l'étude de l'aspect « sécurité » dans les systèmes de smart health.

Ce mémoire est constitué de quatre chapitres ainsi que d'une conclusion générale et d'une introduction générale.

Au cours du premier chapitre nous allons introduire les concepts clés de notre thème à savoir : les villes intelligentes, et leurs composantes. Ensuite nous définirons la smart health qui est la première partie du thème principale de notre recherche bibliographique, nous étudierons plus en détail ces aspects et ces bénéfices pour le patients dans le processus de soin.

Lors du second chapitre, nous parlerons d'une façon générale de l'aspect sécurité dans les systèmes d'informations, nous étudierons également cet aspect dans les systèmes d'information qui contiennent des informations à caractère sensibles et personnel (ceci constitue la deuxième partie principale de notre thème.

Nous ferons également un petit point sur différents système d'informations open source.

Dans le troisième chapitre, nous présenterons en détail l'algorithme HMAC one time passe word, que nous avons utilisé afin d'implémenter une authentification forte au sein d'un système d'information hospitalier et enfin, le quatrième chapitre abordera la partie implémentation.

CHAPITRE I :

La Smart-Health : Généralités

1.1 Introduction

L'informatique médicale est devenue au fil des années un membre à part entière du processus de soin dans de nombreux pays. L'émergence des nouvelles technologies de communications et d'informations, à également permis le développement d'un nouveau concept de villes dites « intelligentes » qui offrent aux citoyens de nouvelles façon d'appréhender leur besoins quotidiens, que se soit leurs façons de gérer leurs déchés, de trouver une place de stationnement, d'éviter les embouteillages ou bien encore de se soigner. Ce nouveau mode de promulgation de soins peut être nommé « smart health ».

Nous allons au cours de ce chapitre définir quelque peu le concept de smart cité ou « ville intelligente » nous définirons par la suite le concept de « smart health » son but et ses domaines d'utilisations.

1.2 Les villes intelligentes

1.2.1 Définition

Il n'existe pas de définition univoque et consensuelle, à proprement parler, du concept de « ville intelligente ».

De façon générale, le concept de ville intelligente appliqué à la planification et aux politiques urbaines réfère à la façon dont les nouvelles technologies de l'information et de la communication sont utilisées en matière de gestion publique pour améliorer la situation d'une ville dans différentes sphères et régler diverses problématiques urbaines. Une ville intelligente est celle qui a su intégrer les TIC à grande échelle dans différents secteurs d'activité afin d'améliorer la vie quotidienne des usagers et des citoyens. De plus, les TIC permettent d'engager un changement de comportement chez les citoyens, mais aussi au sein de l'administration et dans les entreprises vers une croissance plus durable.

Au fil du temps, la croissance urbaine a créé de nouveaux enjeux pour les pouvoirs publics. De nombreuses villes à travers le monde ont dépassé la capacité de leurs infrastructures, de leurs réseaux de transport ainsi que de leurs services publics rendus. Cette urbanisation a un effet sur les infrastructures physiques, sur la capacité fiscale des villes et sur l'environnement et les ressources naturelles. Mais, ces nouvelles technologies favorisent la

mise en place d'innovations et de nouvelles solutions technologiques qui peuvent en outre améliorer les infrastructures physiques de la ville dont le transport, l'approvisionnement en eau, la gestion de l'énergie et la gestion des eaux usées ainsi que les services d'urgence et de santé. Ce concept, qui renvoie notamment à la réduction des impacts environnementaux du développement urbain, est aussi une façon de repenser nos modes de fonctionnement au sein de la ville et ses processus de gestion. [1]

1.2.3 Les composantes de la ville intelligente

Une ville intelligente peut avoir six dimensions :

- **Smart People**
  - Compétences des personnes, accès à l'éducation et à la formation grâce aux TIC (les technologies de l'information et de la communication).
- **Smart Gouvernance**
  - Intégration des organisations publiques, privées et civiles.
  - Prise de décision participative.
  - Transparence, ouverture des données, e-gouvernance, et e-services.
- **Smart économie**
  - Délivrance de nouveaux services, production de nouveaux produits et développement de nouveaux modèles d'affaires.
  - Interconnexion entre monde local et monde globalisé.
- **Smart Mobilité**
  - Mise en place de systèmes durables de transport, intégrés et communicant.
  - Création de systèmes de transport sûrs et interconnectés englobant tramways, bus, trains, métros, voitures, vélos et piétons en multi modalité.
  - Utilisation pertinente des informations en temps réel.
- **Smart Environnent**
  - Développement d'un urbanisme durable et vert au sein des villes.
  - Services urbains: éclairage des rues, gestion des déchets, système de drainage, réseaux d'eau et gestion du territoire.

- Gestion équilibrée des ressources naturelles et patrimoniales, diminution de la pollution.
- Développement d'une gestion énergétique intelligente.
- Energies renouvelables et réseaux d'énergie.

- **Smart Living**
  - Modes de vie, consommation, logements, cohésion sociale, accès à la culture.
  - Qualité de vie en ville: culture, santé, logement, et tourisme. [2]

Parmi les six dimensions que nous avons citées, la dernière inclue une augmentation de la qualité de soins prodigués au citoyen dans le domaine de la santé, grâce à de nouvelles méthodes de traitement utilisant les TIC : ces nouvelles pratiques sont basées sur la e-santé qui peut elle-même être définie comme « l'utilisation des TIC, pour le stockage, ou la transmission de données à caractère médical, à des fin de traitement du patient ou de recherche. ». Cette utilisation particulière de la e-santé peut être défini comme étant la « smart health ».

1.3 La Smart health

1.3.1 Définition

La smart-health décrit une formulation particulière de la e-santé, qui utilise de nouvelles méthodes de traitement des patients, ceux-ci ne sont plus fixés dans les endroits traditionnels comme les hôpitaux ou les cliniques. De nos jours, la smart-health fait partie intégrante de la médecine moderne (dans de nombreux pays tout du moins). [3]

1.3.2 Pourquoi utiliser la smart-health ?

La smart-health a de nombreux avantages sur deux plans : le plan stratégique et le plan opérationnel.

Sur le plan stratégique, la smart-health pourrait :

- Rendre les logements plus effectifs tout en permettant aux contribuables d'économiser.
- Augmenter la qualité des logements.

- Fournir aux médecins traitant la possibilité de contrôles en temps-réel des données du patient et à distance.
- Développement de nouveaux marchés à travers des applications innovatrices.
- Transparence du traitement et des services.

Sur le plan opérationnel, la smart-health a pour objectif de :

- Simplifier les procédures de facturation et les tâches administratives.
- Etablir une compréhension transparente des informations lors d'une thérapie individuelle entre le médecin et son patient. [3]

1.3.3 Domaines d'utilisation de la smart-health

On peut distinguer plusieurs domaines d'application de la smart health plus ou moins calqués sur la médecine traditionnelle :

- **La téléconsultation et tété-expertise** : qui a pour but de mettre en relation deux ou plusieurs acteurs de la santé tels que les médecins, les radiologues, les infirmiers ou toute personne participant au processus de soins.
- **La téléassistance** : qui permet d'assister à distance et donc de pouvoir donner un diagnostic ou des conseils de santé à un patient qui ne bénéficie pas de la possibilité de se déplacer.
- **La télésurveillance** : qui consiste à surveiller une fonction vitale à domicile ou en déplacement, par exemple lors du transfert d'un patient en ambulance.
- **La télé chirurgie et le télé diagnostique :** qui consiste en l'action de diagnostiquer ou de pratiquer une opération à distance.
- **Le dossier médical personnel partageable (DMP) :** qui consiste en le stockage ou la transmission de données médicales de patients dans un réseau informatique (internet ou intranet) afin de donner la possibilité aux médecins traitant d'accéder directement aux données du patient, par exemple ces antécédents médicaux. (Le DMP fera l'objet d'une description plus détaillée un peu plus loin dans ce document).
- **Le cyber formation** : ou le e-learning qui consiste à enseigner ou à offrir une formation continue à distance. Cette idée n'est pas nouvelle puisque déjà

utilisée dans d'autres domaines mais elle s'applique uniquement ici à la formation médicale.

- **L'e-management** : quia pour but d'offrir au patient ainsi qu'aux médecins un accès permanent aux dossiers médicaux [3], et qui peut être inclus dans le DMP.

Conclusion

Tout ce que peut nous rapporter la smart health comme bénéfices ne doit pas nous détourner d'un aspect important relatifs à la grande quantité d'informations collectées, stockées ou traitées dans ce genre de système : cet aspect est la sécurité .Nous allons dans le chapitre suivant nous concentrer dessus et parler d'abord de la sécurité de l'information de façon générale, par la suite, nous évoquerons le type de données que peut contenir un système de smart health et leur caractère spécial, enfin nous citerons quelque système d'informations médicaux open source.

CHAPITRE II :

La Sécurités dans les Systèmes de Smart Health

2.1 Introduction

Les systèmes de smart health comportent de nombreuses informations sensibles et personnelles. Ces informations doivent être protégées, contre toute indiscrétion, ou menaces. Comment définir cependant ce qu'est une information à caractère sensible ? Nous allons au cours de ce chapitre, d'abord, nous intéresser à la sécurité de l'information de façon générale, par la suite nous évoquerons brièvement quelques techniques d'authentification utilisées afin de garantir la confidentialité des données dans les systèmes d'informations. Ensuite, nous définirons ce qu'est une donnée à caractère personnelle et évoquerons également quelques techniques trouvées dans la littérature, utilisées afin de contrôler l'accès dans des systèmes contenant des données à caractère sensible. Enfin nous citerons quelques systèmes d'information hospitaliers open source.

2.2 La sécurité des systèmes d'information

2.2.1 Définition

La sécurité des systèmes d'informations est l'ensemble des techniques et outils mis en œuvre pour minimiser la vulnérabilité (failles) et protéger les ressources du système d'information contre des menaces accidentelles ou intentionnelles.

Les différences entre accidents et malveillances (intentionnelles) :

- Sécurité = " Safety" C'est la protection contre des événements accidentels imprévisibles. Ex : Pompiers.
- Sécurité = "Security" (Sureté) C'est la protection par rapport à des événements intentionnels. Ex : Police.

2.2.2 L'Objectifs de la sécurité des systèmes d'information

- La sécurité informatique vise généralement cinq principaux objectifs :

  L'intégrité **:** permet d'empêcher la modification non-autorisée de données.

- La confidentialité **:** permet d'empêcher la divulgation non-autorisée de données

- La disponibilité : permet de maintenir le bon fonctionnement du système d'information, c'est-à-dire de garantir l'accès à un service ou à des ressource à n'import quel moment.
- La non répudiation : permet de garantir qu'une transaction ne peut être niée.
- L'authentification : permet d'empêcher l'utilisation non-autorisée de ressources.

2.3 L'authentification

2.3.1 Définition

L'authentification est la vérification d'informations relatives à une personne ou à un processus informatique. L'authentification complète le processus d'identification dans le sens où l'authentification permet de prouver une identité déclarée. Dans un serveur, un processus de contrôle valide l'identité et après authentification, donne l'accès aux données, applications, bases de données, fichiers ou sites Internet. Dans le cas contraire, l'accès est refusé.

L'authentification peut se faire de multiples manières, et notamment par la vérification de :

« Ce que je sais », un mot de passe par exemple,

« Ce que je sais faire », une signature manuscrite sur écran tactile/digital.

« Ce que je suis », une caractéristique physique comme une empreinte digitale,

« Ce que je possède », une carte à puce par exemple.

Le choix de telle ou telle technique dépend en grande partie de l'usage que l'on souhaite en faire : authentification de l'expéditeur d'un email, authentification d'un utilisateur qui se connecte à distance, authentification d'un administrateur au système, authentification des parties lors d'une transaction de B2B (Business to Business), etc.

La combinaison de plusieurs de ces méthodes (aussi appelées facteurs d'authentification) permet de renforcer le processus d'authentification, on parle alors d'authentification forte.

2.3.2 Authentification forte

L'authentification forte est en sécurité des systèmes d'information une procédure d'identification qui requiert la concaténation d'au moins deux facteurs d'authentification. En effet, l'utilisation d'un simple nom d'utilisateur et d'un mot de passe n'est pas considérée comme sûre, l'ajout d'un troisième ou d'un quatrième facteur va considérablement baisser les risques d'usurpation d'identité.

Parmi les techniques utilisées comme facteur d'authentification forte, nous pouvons citer la biométrie sous ses différentes formes.

Les mots de passe à usage unique sont aussi utilisés dans les procédures d'authentification forte, les certificats électroniques ainsi que les cartes à puce constituent également des alternatives fiables. Nous verrons dans la suite de ce document plus en détails les mots de passe à usage uniques et comment on peut les utilisés pour une authentification forte.

2.3.3 Identifier et authentifier

Les personnes qui accèdent à une ressource non publique doivent être identifiées ; leur identité doit être authentifiée, leurs droits d'accès doivent être vérifiés au regard des habilitations qui leur ont été attribuées à ces trois actions correspond un domaine des techniques de sécurité, les méthodes d'authentification, de signature, de vérification de l'intégrité des données et d'attribution de droits (une habilitation donnée à un utilisateur et consignée dans une base de données adéquate et une liste de droits d'accès et de pouvoirs formulés de telle sorte qu'un système informatique puisse les vérifier automatiquement).

2.4 Méthodes courantes d'authentification

2.4.1 Mots de passe

Les mots de passe pris dans leur ensemble sont le moyen d'authentification le plus répandu à ce jour. On distingue deux catégories : les mots de passe statiques et les mots de passe dynamiques.

Les mots de passe statiques sont des mots de passe qui restent identiques pour plusieurs connexions sur un même compte. Cette technique d'authentification est la plus utilisée dans les entreprises mais aussi la moins robuste. En fait, les entreprises devraient

restreindre l'usage des mots de passe statiques à une authentification locale d'un utilisateur car les attaques qui permettent de capturer un mot de passe qui circule sur un réseau sont nombreuses et faciles à mettre en pratique. Pour pallier les faiblesses de l'usage des mots de passe statiques, sont apparues des solutions d'authentification combinant deux facteurs (« ce que je possède » et « ce que je sais ») afin d'obtenir une authentification forte. Les mots de passe sont obtenus par des Générateurs de mots de passe activés à l'aide d'un code d'identification personnel ou PIN (Personal Identification Number). La mise en place d'un tel mécanisme d'authentification forte rend la capture du mot de passe en cours d'aucune utilité puisque, dès que le mot de passe dynamique a été utilisé, celui-ci devient caduc. Parmi ces mots de passe à usage unique – One Time Password (OTP) en Anglais – on trouve notamment le programme SKEY dont la sécurité repose sur une fonction à sens unique et qui permet degénérer un mot de passe différent pour chaque nouvelle connexion. En version logicielle, ces générateurs de mots de passe dynamiques utilisent certains composants du PC, comme le CPU ou l'Horloge interne (on parle alors de méthode d'authentification en mode synchrone dépendant du temps). Que le mot de passe à usage unique soit obtenu à partir d'un générateur matériel ou logiciel, l'utilisateur est authentifié de manière forte grâce à la vérification du mot de passe dynamique par un serveur appelé serveur d'authentification.

Par ailleurs, l'authentification peut aussi reposer sur un protocole d'authentification réseau, par exemple le protocole Kerberos[Mil 87], [Ste 88], qui permet de sécuriser les mots de passe statiques lorsqu'ils sont transmis sur le réseau. Ce protocole, créé au le Massachusetts Institute of Technology (MIT), utilise la cryptographie à clés publiques.

2.4.2 La biométrie

Pris au sens large, la biométrie recouvre l'ensemble des procédés tendant à identifier un individu à partir de la « mesure » de l'une ou de plusieurs de ses caractéristiques physiques, physiologiques ou comportementales. [8]

Le choix des caractéristiques physiques est important. Il faut qu'elles soient toutes à la fois :

- Discriminantes, pour différencier les personnes sans équivoque.
- Invariables, pour assurer leur permanence.
- Universelles, pour être appliquées à tout le monde.

- Faciles à exploiter et acceptables culturellement par les utilisateurs.
- Difficilement falsifiables.

La biométrie est basée sur l'analyse des données liées à l'individu et peut être classée en trois grandes catégories:

- L'analyse morphologique (empreintes digitales, forme de la main, traits du visage,...)
- Les traces biologiques (odeur, salive, ADN,...)
- L'analyse comportementale (dynamique du tracé de la signature, frappe sur un clavier d'ordinateur,...)

Sur ces catégories ont émergé différentes techniques et procédés biométriques parmi lesquels nous pouvons citer :

L'ADN, la rétine, l'iris, l'empreinte digitale (et l'empreinte palmaire), la reconnaissance faciale, la géométrie du contour de la main, la voix et l'écriture manuscrite. [8].

La Qualité d'une méthode d'authentification ne se mesure pas uniquement à ses avantages et inconvénients généraux ou à sa robustesse théorique face aux attaques mais avant tout à sa capacité à répondre aux besoins de sécurité de l'organisme, tout en prenant en considération le secteur d'activité dans lequel elle est mise en place (son environnement) i.e. en tenant compte des risques mais aussi des contraintes techniques, organisationnelles, temporelles de même que financières et légales. En outre, le budget alloué à l'informatique en général et à la sécurité informatique en particulier, de même que la taille de l'organisme et les compétences internes sont des facteurs très influents sur le choix de telle ou telle solution d'authentification.

2.4.3 Les fonctions de hachage

Une fonction de hachage est aussi appelée fonction de hachage à sens unique ou "one-way hash function" en anglais. Ce type de fonction est très utilisé en cryptographie, principalement dans le but de réduire la taille des données à traiter par la fonction de

chiffrement. En effet, la caractéristique principale d'une fonction de hachage est de produire un haché des données, c'est-à-dire un condensé de ces données. Ce condensé est de taille fixe, dont la valeur diffère suivant la fonction utilisée. Cette taille des condensés à une importance capitale dans la sécurité des données traitées.

Dans la réalité, les fonctions de hachage à sens unique sont inspirées des fonctions de compression. Ces fonctions qui sont à sens unique, produisent une valeur de longueur n, à partir d'une entrée de longueur m supérieure à n. L'entrée de la fonction de compression est un bloc de texte et l'empreinte du bloc précédent (voir la figure 2). La sortie est l'empreinte de tous les blocs jusqu'à ce point. C'est-à-dire que l'empreinte du bloc Aii est donnée par:

hi = f (Mi,hi-i).

L'empreinte en cours et le bloc de texte suivant servent de paramètres à l'application suivante de la fonction de compression. L'empreinte du dernier bloc devient l'empreinte de tout le message.

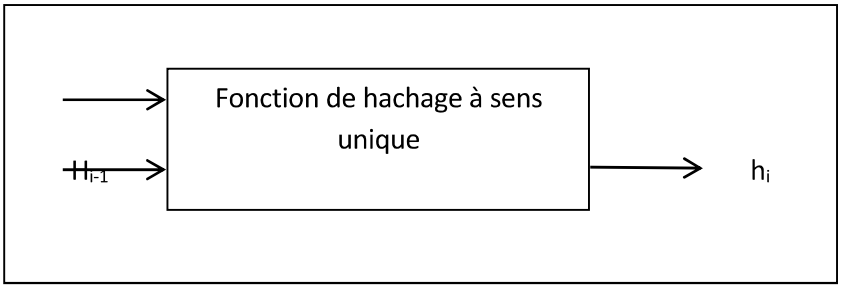

Figure 1 : Fonction de hachage à sens unique

2.4.4 Destruction d'information - Conservation de propriétés

Prenons l'exemple des empreintes digitales : dans la perception que nous en avons à l'heure actuelle, une empreinte digitale est unique et représente un individu d'une façon si certaine que nous pouvons la qualifier de sûre. Pourtant la connaissance de cette empreinte ne permet pas à elle-seule de remonter à l'individu, ni de reconstituer cet individu. Il faut que la correspondance ait été préalablement établie dans une base de données pour que l'identification puisse avoir lieu par comparaison.

C'est exactement ce genre de propriétés que présente une fonction de hachage. En effet, le haché est caractéristique d'un texte ou de données uniques. Différentes données donneront toutes des condensés différents. De plus, tout comme l'empreinte digitale, le

condensé ne contient pas assez d'informations en lui-même pour permettre la reconstitution du texte original : c'est pour cela que l'on parle d'ailleurs de fonction à sens unique (l'opération de hachage est destructrice dans le sens où elle conduit à une perte d'information). Mais il faut bien comprendre que le but d'un condensé n'est pas de véhiculer ou de transporter de l'information. Il est juste représentatif d'une donnée particulière et bien définie. D'autant que les algorithmes de hachage les plus courants sont publics et ne représentent pas en eux-mêmes un secret.

2.4.5 Le but du hachage

Le but d'un condensé est simple : représenter des données de façon certaine tout en réduisant la taille utile qui sera réellement chiffrée. Prenons l'exemple de la cryptographie asymétrique; tout le monde admet qu'elle est très sûre, fiable et durable. Néanmoins, sa complexité (calcul sur des nombres premiers de plusieurs centaines de chiffres par exemple) peut éventuellement entrainer une certaine lourdeur d'emploi (charge CPU, etc...). Les concepteurs des systèmes d'information évitent donc de l'utiliser pour de grandes masses de données qu'ils jugent peu sensibles.

Par contre imaginons que nous souhaitions envoyer un fichier par mail, mais que ce fichier soit de taille importante. Nous souhaitons de plus rassurer le destinataire sur la provenance de ce fichier et sur son contenu. Plutôt que de chiffrer notre fichier directement avec notre clé privée, nous allons hacher notre fichier et chiffrer le condensé obtenu avec notre clé privée. Nous enverrons ensuite notre fichier original ainsi que le condensé chiffré (la signature) à notre destinataire.

Celui-ci va, lors de la réception, hacher d'une part le fichier reçu et d'autre part déchiffrer le condensé reçu (au moyen de notre clé publique).

S'il n'y a pas égalité entre les 2 résultats, cela signifiera soit :

- Soit que la signature n'est plus la nôtre, donc que quelqu'un a intercepté le fichier (pour le modifier ou le remplacer, etc...)
- Soit que le fichier n'est plus le même que l'original (mais la signature n'a pas été remplacée); dans ce cas, le hachage ne peut plus donner le même condensé ce qui conduit au rejet lors du test de comparaison.

- Dans les deux cas, ni l'intégrité ni l'authentification du fichier n'ont été vérifiées. Il ne faut donc pas faire confiance au fichier.

- Nous voyons comment dans ce cas simple, l'utilisation d'une fonction de hachage permet de s'assurer de l'intégrité des données et indirectement de les authentifier.

- De façon générale, les fonctions de hachage sont utilisées avec d'autres méthodes cryptographiques comme la signature électronique de documents.

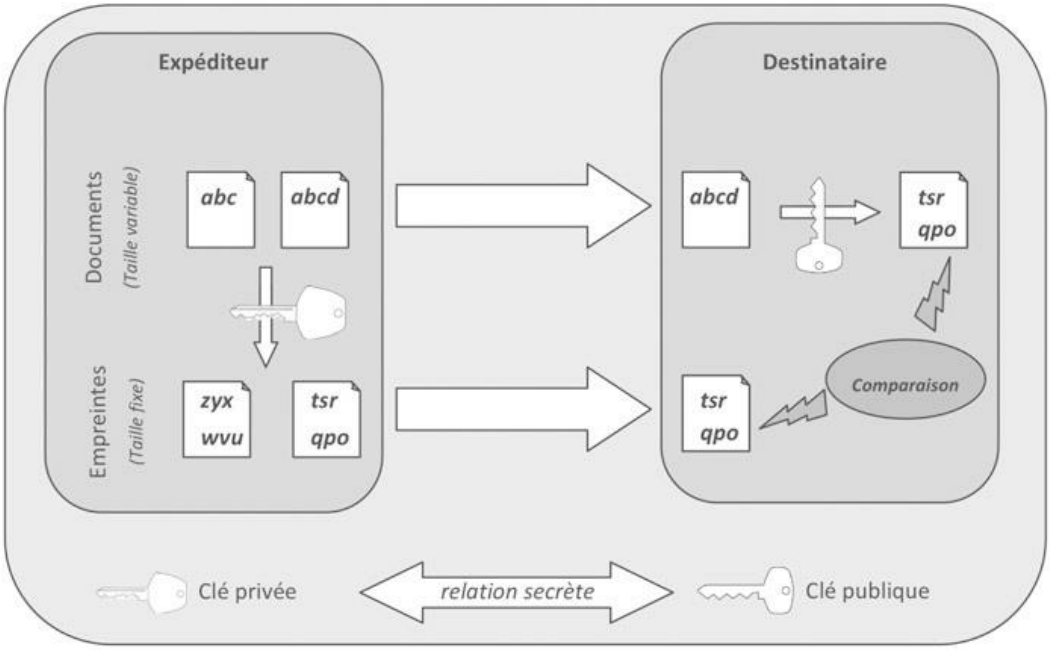

Figure 2 : Signature numérique

Sur la figure 2, on observe que l'expéditeur crée une empreinte de taille fixe du message qui, lui, est de taille variable, par une technique de hachage. Les fonctions de hachage sont dites à sens unique si le calcul de leur inverse est considéré comme irréalisable.

2.4.6 Fonctions de hachage usuelles

MD4 et MD5 (Message Digest) furent développées par Ron Rivest. MD5 produisent des hachés de 128 bits en travaillant les données originales par blocs de 512 bits. [9].

Le SHA-1 (Secure Hash Algorithm 1), comme MD5, est basé sur MD4 dont une description détaillée se trouve dans [10]. Il fonctionne également à partir de blocs de 512 bits

de données et produit par contre des condensés de 160 bits en sortie. Il nécessite donc plus de ressources que MD5.

SHA-2 (Secure Hash Algorithm 2) est destiné à remplacer SHA-1. Les différences principales résident dans les tailles de hachés possibles : 256, 384 ou 512 bits. [Nis 12] Spécifie les algorithmes qualifiés comme étant sûrs, ils sont qualifiés de la sorte parce qu'à partir d'un condensé, il est impossible en l'état actuel de la technologie de :

1. De trouver un message qui corresponde à ce condensé.

2. De trouver deux messages différents qui produisent le même haché, le moindre changement dans le message original va avec une très haute probabilité produire un condensé différent, ceci va induire à un échec lors de la vérification, si la fonction de hachage est utilisée dans une méthode de signature numérique ou de code d'authentification de message (MAC). [Nis 12].

3. La sécurité des systèmes d'information médicaux

Dans le contexte d'un système d'information hospitalier, il est primordial de fournir tous les moyens permettant de maîtriser les risques et les menaces pouvant toucher de près ou de loin aux informations liées à la santé des patients. [5] en effet, les données médicales contenues dans les dossiers des patients sont très sensibles et sujettes au secret médical. Cependant, avant de nous intéresser à leur sécurité, nous allons d'abord définir quelques notions : la notion de « secret médical », celle de « donnée à caractère personnel », ainsi que quelques règles relatives à leur traitement.

3.1. Le secret médical

3.1.1. Définition

La reconnaissance pour le patient « d'un droit au secret » est une notion très ancienne, bien antérieure à la médecine moderne, puisque Hippocrate en faisait déjà état dans son serment : « Quoi que je voie ou entende dans la société pendant, ou même hors de l'exercice de ma profession, je tairai ce qui n'a jamais besoin d'être divulgué, regardant la discrétion comme un devoir en pareil cas ».

Le secret médical tel qu'il est défini dans le code de déontologie médicale Algérien (dans son article 37) est une obligation à laquelle est soumis le corps médical dans l'exercice de ses fonctions ; il concerne tout ce que le médecin a vu, compris ou lui a été confié dans l'exercice de ses fonctions. Voir tout ce qui a pu être interprété lors de l'exercice médical. Pour cela, on peut dire que ça concerne les déclarations du malade, les diagnostics, les thérapeutiques, les fiches ou dossiers médicaux, mais aussi les conversations surprises au domicile lors d'une visite médicale de contrôle. Le secret médical est également un droit du malade.

Le code de déontologie médical Algérien dans son article 36, stipule que le secret médical « s'impose à tout médecin et chirurgien dentiste sauf lorsque la loi en dispose autrement. ».

L'article 40 du Code de déontologie médical Algérien stipule également que « Quand le médecin, le chirurgien dentiste se sert de ses dossiers médicaux pour des publications scientifiques, il doit veiller à ce que l'identification du malade ne soit pas possible. ».

3.1.2. Les données à caractère personnel

3.1.2.1. Définition

Toute information relative à une personne physique identifiée ou identifiable directement ou indirectement, par référence à un numéro d'identification ou à un ou plusieurs éléments, propres à son identité physique, physiologique, génétique, psychique, culturelle, sociale ou économique.

Dans l'article n°2 de la loi informatique et liberté française, on trouve la définition suivante : Constitue une donnée à caractère personnel toute information relative à une personne physique identifiée ou qui peut être identifiée, directement ou indirectement, par référence à un numéro d'identification ou à un ou plusieurs éléments qui lui sont propres. Pour déterminer si une personne est identifiable, il convient de considérer l'ensemble des moyens en vue de permettre son identification dont dispose ou auxquels peut avoir accès le responsable du traitement ou toute autre personne.

Nous avons pris la loi « informatique et libertés » comme référence parce que nous n'avons pas trouvé d'équivalent en Algérie.

Selon la CNIL (La Commission Nationale Française de l'Informatique et des Libertés), les données sont considérées « à caractère personnel » dès lors qu'elles concernent des personnes physiques identifiées directement ou indirectement.

Une personne est identifiée lorsque par exemple son nom apparaît dans un fichier.

Une personne est identifiable lorsqu'un fichier comporte des informations permettant indirectement son identification (ex. : adresse IP, nom, n° d'immatriculation, n° de téléphone, photographie, éléments biométriques tels le numéro d'Identification Nationale Français Étudiant (INE), ensemble d'informations permettant de discriminer une personne au sein d'une population (certains fichiers statistiques) tels que, par exemple, le lieu de résidence, la profession, le sexe et l'âge,....)

Des données que nous pourrions considérer comme anonymes peuvent constituer des données à caractère personnel si elles permettent d'identifier indirectement ou par recoupement d'informations une personne précise. Il peut en effet s'agir d'informations qui ne sont pas associées au nom d'une personne mais qui permettent aisément de l'identifier et de connaître ses habitudes ou ses goûts.

En ce sens, constituent également des données à caractère personnel toutes les informations dont le recoupement permet d'identifier une personne précise. (ex : une empreinte digitale, l'ADN, une date de naissance associée à une commune de résidence …).

Les technologies de l'information et de la communication génèrent de nombreuses données personnelles (un appel passé par un téléphone portable, une connexion à Internet) et aussi des « traces informatiques » facilement exploitables grâce aux progrès des logiciels, notamment les moteurs de recherche.

3.1.2.1. Traitement de données à caractère personnel

Nous donnons également la définition du traitement de données à caractère personnel selon l'article 3 de la loi informatique et liberté :

Constitue un traitement de données à caractère personnel toute opération ou tout ensemble d'opérations portant sur de telles données, quel que soit le procédé utilisé, et notamment la collecte, l'enregistrement, l'organisation, la conservation, l'adaptation ou la modification, l'extraction, la consultation, l'utilisation, la communication par transmission, diffusion ou

toute autre forme de mise à disposition, le rapprochement ou l'interconnexion, ainsi que le verrouillage, l'effacement ou la destruction.

3.1.2.2. Les règles relatives au traitement automatisé des données à caractère personnel

Secret professionnel et données personnelles sont intrinsèquement liés : Comme exposé précédemment, le premier se définit comme l'obligation de ne pas divulguer les secondes. Si ce secret est aisément imposable à une personne physique : le « professionnel » manipulant les données – il en va différemment lorsque ce traitement est automatisé, c'est-à-dire effectué par une machine et/ ou dans le but d'insérer les données à un fichier structuré. En effet, le propre de l'outil informatique est de stocker l'information puis de la restituer (directement ou après opérations) sur demande. Il apparait donc crucial, dans le cadre d'un système automatisé et eu égard à la protection de l'individu et de la vie privée, de bien encadrer le « traitement » des données personnelles, terme sous lequel on regroupe ensemble des opérations appliquées à ces données, depuis leur collecte et enregistrement jusqu'à leur effacement ou destruction, en passant par leur organisation, leur conservation, leur consultation, leur modification, leur diffusion ou leur rapprochement. [4]

3.2. Le code Algérien de déontologie médicale

Intéressons nous à présent au code de déontologie médical Algérien et voyons comment il défini le secret professionnel et comment il responsabilise également les médecins en ce qui concerne la protection des données médicales de leurs patients.

3.2.1. Le secret professionnel

Voici la définition que nous trouverons dans [6].

« Le secret professionnel, institué dans l'intérêt du malade et de la collectivité, s'impose à tout médecin et chirurgien dentiste sauf lorsque la loi en dispose autrement. » . [6]

« Le médecin, le chirurgien dentiste doit veiller à la protection contre toute indiscrétion des fiches cliniques et documents qu'il détient concernant ses malades ». [6]

Nous pouvons remarquer que le code de déontologie médicale Algérien responsabilise le médecin quand à la protection des données de ces patient, ce qui ne pourrait être le cas dans

un système informatisé où le médecin n'est qu'un utilisateur et qu'il ne peut nullement être tenu responsable de cela.

Maintenant que nous avons défini ce que sont les informations à caractère et personnel et parlé de leur sensibilité, nous allons nous intéresser au problème de contrôle d'accès à ces informations et voir quelques techniques utilisées afin de ne donner accès à ses données qu'aux personnes adéquates. De nombreux modèles de contrôle d'accès ont été proposés dans la littérature, nous allons en citer 3 :

3.2.2. Les modèles de contrôle d'accès

Modèles de contrôle d'accès discrétionnaires (DAC)

Le contrôle d'accès discrétionnaire (DAC) est un moyen pour contrôler l'accès d'un sujet à un objet. Dans ce modèle les décisions d'accès sont généralement basées sur l'identité du sujet (nom d'utilisateur, mot de passe, jeton, etc.).

Les contrôles sont discrétionnaires dans le sens où le sujet est capable de transférer les permissions d'accès à d'autres sujets.

Le modèle de contrôle d'accès discrétionnaire représente les politiques de contrôle d'accès sous forme d'un triplet (S,O, Mso) où S désigne les sujets, O les objets, et Mso la matrice de contrôle d'accès.

Pour utiliser le modèle discrétionnaire dans le domaine de la santé, il est indispensable de déterminer le propriétaire du DMP, car dans ce type de modèle c'est le propriétaire des données qui contrôle l'accès. Cependant, l'information médicale est partiellement détenue par le patient, le médecin et l'autorité médicale, ce qui pose un réel problème de propriété.

Modèles de contrôle d'accès obligatoires(MAC)

Le modèle MAC se base sur la classification des sujets et des objets. A chaque sujet et objet on attribue une classe dans l'ensemble (Confidentiel, Secret, Top Secret, etc). La classe de l'objet indique la sensibilité de l'information, alors que la classe du sujet indique le degré de confiance accordé au sujet en termes de divulgation d'information. Ainsi, les droits d'accès dépondront des classifications attribuées aux objets et des habilitations attribuées aux sujets.

L'utilisation du modèle MAC dans le domaine de la santé implique que l'autorité de santé est la seule instance à pouvoir définir la politique de sécurité. Ceci implique que ni le personnel de santé, ni le patient n'a le moyen d'intervenir et donc d'exprimer les besoins en matière de protection de données personnelles. Cependant, dans le modèle MAC, une donnée est associée à seulement un niveau de sécurité.

Modèle de contrôle d'accès à base d'organisation (OR-BAC)

Le concept central d'Or-BAC est l'organisation. Une organisation peut être vue comme un groupe structuré de sujets qui jouent certains rôles. Ça peut être un hôpital, une clinique, un service d'urgence, etc. Le but principal du modèle est de permettre de définir une politique de sécurité indépendamment de son implémentation. Pour cela, un niveau d'abstraction a été introduit. Les sujets sont abstraits en rôle, les actions en activité et les objets en vue. Pour avoir plus d'information sur le sujet, consulter [5].

Nous avons donné quelques modèles de contrôle de droits d'accès aux données, nous comptons par la suite (après ce mini projet) proposer un autre modèle basé sur une autre technologie.

3.3. Le dossier médical personnel (DMP) et système d'information hospitalier

Nous allons à présent parler plus en détail du dossier médical personnel (nous en avions déjà parlé plus haut dans ce chapitre), ainsi que sur les systèmes d'informations hospitaliers étant donné que nous allons par la suite nous concentrer sur les données personnelles qu'il contient et sur comment garantir leur sécurité, (en implémentant un deuxième facteur d'authentification afin de garantir le contrôle d'accès aux dossiers des patients).

3.3.1. Définition

Le dossier médical personnel est une base de données contenant des informations concernant le patient, notamment l'historique médical de celui-ci et les données essentielles a sa prise en charge

Le DMP est un outil médical et paramédical, utilisé à des fins préventive, diagnostique et thérapeutique. Il est partagé sur un réseau informatique (internet) et diffusé largement afin de permettre à tous les professionnels de santé d'y accéder en cas de besoin, qu'ils soient privés ou exerçant dans le secteur publique.

3.3.2. Les objectifs du DMP

De nombreux pays ont tenté de mettre en place un DMP dans leurs systèmes de santé, avec plus ou moins de succès. Cette démarche avait pour objectifs de :

- Améliorer la qualité des soins.
- D'avoir un rôle juridique important dans le cas d'une recherche de responsabilité pour une erreur médicale.
- Faciliter le partage des informations entre le patient et les professionnels de santé (dans le respect du secret médical et de la vie privée du patient).
- L'optimisation de la durée de la visite.
- Éviter la répétition d'informations.
- Eliminer les déplacements pour compléter le dossier.
- Accès simultané à un même dossier par plusieurs médecins ou intervenants.
- Lisibilité des informations consultées.
- Réduire la perte de données.
- Répondre aux besoins sécurité et de traçabilité.

3.3.3. Inconvénients du DMP

- Ouverture très lourde du DMP pour le patient et le médecin.
- Le DMP reste lourd à utiliser : empilement de fichiers au format PDF.
- Comment retrouver une information ancienne ?.
- Le DMP ne permet pas d'ajouter des antécédents à une liste préalablement définie.
- Le tri des données s'avère difficile et aléatoire.
- La lisibilité du dossier est difficile.

Parallèlement au DMP, les établissements de santé peuvent également avoir leur propre SIH qui peut contenir des informations similaires la plupart du temps à celles contenues dans le DMP. Dans notre travail, étant donné que nous n'avons pas accès à une plateforme de DMP afin d'effectuer nos recherches, nous allons nous orienter vers des SIH (open source).

Le paragraphe suivant contient une brève description de ce que doit contenir le dossier patient dans un SIH standard.Ensuite dans le paragraphe qui le suivra, nous présenteront les Système d'information hospitaliers plus en détails.

3.3.4. Le contenu du dossier patient

3.3.4.1. Le dossier médical

- Les informations formalisées recueillies lors des consultations externes dispensées dans l'établissement, lors de l'accueil au service des urgences ou au moment de l'admission et au cours du séjour en établissement de santé, et notamment :

1. Le rapport du médecin qui est à l'origine de la consultation ou de l'admission.
2. Les motifs d'hospitalisation.
3. La recherche d'antécédents et de facteurs de risques.
4. Les conclusions de l'évaluation clinique initiale.
5. Le type de prise en charge prévu et les prescriptions effectuées à l'entrée.
6. La nature des soins dispensés et les prescriptions établies lors de la consultation externe ou du passage aux urgences.
7. Les informations relatives à la prise en charge en cours d'hospitalisation : état clinique, soins reçus, examens para cliniques, notamment d'imagerie.
8. Les informations sur la démarche médicale.
9. Le dossier d'anesthésie.
10. Le compte rendu opératoire ou d'accouchement.
11. Le consentement écrit du patient pour les situations où ce consentement est requis sous cette forme par voie légale ou réglementaire.
12. Les éléments relatifs à la prescription médicale, à son exécution et aux examens complémentaires.

13. Le dossier de soins infirmiers ou, à défaut, les informations relatives aux soins infirmiers.
14. Les informations relatives aux soins dispensés par les autres professionnels de santé.
15. Les correspondances échangées entre professionnels de santé.

- Les informations formalisées établies à la fin du séjour. Elles comportent notamment :

1. Le compte rendu d'hospitalisation et la lettre rédigée à l'occasion de la sortie.
2. La prescription de sortie et les doubles d'ordonnance de sortie.
3. Les modalités de sortie (domicile, autres structures).
4. La fiche de liaison infirmière.

- La partie administrative du dossier du patient fournit les données d'identification du patient et les données sociodémographiques régulièrement tenues à jour (suivi de l'identité de l'état civil, de la couverture sociale, du statut matrimonial, des employeurs, etc.).

Le dossier patient comporte les données de santé dont le volume et la complexité croit sous l'influence du développement des spécialités médicales et de leur technicité.

Le dossier patient comporte également les modalités selon lesquelles le patient a été informé de la démarche diagnostique et thérapeutique mise en œuvre et le cas échéant les traces de son consentement aux soins ou à l'utilisation de ses données cliniques pour la recherche ou la sante publique.

En effet, si le Dossier patient (DP) est destiné à la prise en charge globale d'un individu unique, son contenu doit répondre aux besoins spécifiques des différents professionnels de santé impliqués dans le processus de soin mais également dans des activités de recherche ou de santé publique.

Au-delà d'une utilisation individuelle, le DP est utilisé de façon collective pour caractériser une population a des fins de santé publique ou de recherche (épidémiologique, recherche clinique) ou dans le cadre de l'évaluation des pratiques professionnelles.

Bien sûr, ces informations ne seront pas toutes présentent à chaque fois, et le contenu du dossier peut sensiblement varier d'un système à un autre.

3.4. Les systèmes d'informations hospitaliers

3.4.1. Définition

Un système d'information hospitalier est un système informatique facilitant la gestion des données médicales et administratives dans un établissement hospitalier

3.4.2. Objectifs des SIH

Le tableau suivant précise les objectifs de la constitution de SIH

Principaux	**Contributifs**
Amélioration de la qualité et de la continuité des soins	• Uniformisation des pratiques • Aide à la prise de décisions • Amélioration des résultats • Amélioration des communications • Réduction des délais d'attente • Dossier patient intégré • Aide à la prise de décisions
Maîtrise des coûts	• Optimisation des processus médicaux • Réduction des tâches administratives • Réduction de la durée des séjours • Mise à disposition d'outils de pilotage médico-économique • Diminution des frais de personnel • Optimisation des ressources

Tableau 1 – Objectifs des systèmes d'information hospitaliers. [7]

3.5. Système d'information hospitalier open source

Nous allons à présent donner quelques systèmes d'informations hospitaliers open source.

1. **Mediboard** : Mediboard est un système d'information hospitalier libre, basé sur les technologies web. Il s'agit du premier produit Open Source en langue française dans le domaine des systèmes d'information hospitaliers. Les fonctionnalités de ce système, la possibilité d'une implémentation directement en français, et sa facilité d'adaptation font qu'il serait un modèle intéressant pour les pays en développement notamment pour les pays francophones. Il possède également de nombreux autres avantages notamment son système modulaire ainsi que son ouverture sur les standards de l'smart-health comme le HL7. Il est en outre utilisable sur un simple navigateur web, ce qui le rend compatible avec n'importe quel dispositif qui en dispose. Malheureusement sa communauté n'est plus aussi active qu'elle ne l'était par le passé et le fait qu'il soit directement implémenté en français rend le nombre de ses contributeurs moins nombreux que pour les logiciels anglophones.

2. **OpenEMR :** OpenEMR est un logiciel de gestion d'établissements de santé incluant la gestion de dossier patient. Il est gratuit et open source, et utilisé dans de nombreux établissements de santé à travers le monde. Il bénéficie de la certification ONC (Office of the National Coordinator for Health Information Technology certification) qui est une certification accordée par le département d'état américain de la santé aux logiciels de santé et qui garantit leur respect d'un certain nombre de standards (d'interopérabilité notamment).OpenEMR bénéficie d'une large communauté de développeurs. Il utilise des technologies web (PHP, html, MySQL) ce qui le rend utilisable à partir d'un navigateur web et par conséquent sur tous types de supports qui en disposent.

3. **OpenMRS (Medical Record System) :** c'est un système de dossiers médicaux électroniques, conçu pour être utilisé dans les pays en voie de développement. Sa première implémentation date de 2004. Il permet la conception d'un système personnalisé de rapports médicaux sans connaissances en programmation. Aujourd'hui, le système s'est transformé en une plate-forme d'informatique médicale utilisée sur les cinq continents. OpenMRS a était développé en Java et il utilise une base de données MySQL ainsi qu'un serveur web apache. Il fonctionne à partir d'un simple navigateur web et respecte le standards HL7 pour garantir l'inter-opérabilité avec les autres systèmes d'information hospitaliers. Il est divisé en modules que l'utilisateur pourra ajouter ou supprimer selon les besoins du site où il est installé.

OpenMRS est très utilisé en Afrique subsaharienne ainsi qu'on Amérique du sud et en Asie (Argentine, Botswana, Cambodge, Congo, Ethiopie, Gabon, Ghana, Haïti, Honduras, Inde, Indonésie, Kenya, Lesotho, Malawi, Malaysia, Mali, Mozambique, Népal, Nicaragua, Nigeria, Pakistan, Pérou, Philippines, Rwanda, Sénégal, Afrique du sud, Sri Lanka, Tanzanie, Gambie, Uganda, USA, Zanzibar, Zimbabwe).OpenMRS n'est pas seulement utilisé dans de nombreux pays, il est également utilisé pour répondre à de nombreux besoins. Au Kenya par exemple, il est utilisé pour appuyer les prestations de soins délivrées à des centaines de milliers de patients, à l'aide d'un réseau de près de 50 cliniques. Quelques-unes sont connectées entre elles par un réseau classique, mais la plupart se synchronisent hors ligne en utilisant des supports de stockage externes qui sont transportés physiquement entre les sites. Une autre organisation non gouvernementale à relié un serveur central d'OpenMRS à des cliniques situées sur des pays différents en utilisant une connexion internet via satellite.Sa communauté de développeurs est parmi les plus importantes dans le monde des SIH open source, il est également en train d'être traduit dans de nombreuses langues (dont le Français) et bénéficie du soutien de nombreuses organisations internationales. [4].

4. **Open Hospital :** « Open hospital » est un SIH Open Source écrit en java et développé initialement afin de répondreaux besoin de l'hôpital St. Luke en Uganda, cependant, de nombreux hôpitaux l'ont adopté par la suite.

5. Hospital Os : C'est un SIH open source développé pour les hôpitaux thaïlandais de moins de 100 Lits. Il est également écrit en java et permet de gérer les dossiers médicaux des patients, les services d'imagerie médicale, les pharmacies et de nombreux autres services hospitaliers.

Conclusion

Les systèmes de smart health comportent énormément de données sensibles. Un des aspects les plus importants de leur sécurisation est celui du contrôle d'accès aux dossiers des patients.

Au cours de ce chapitre nous avons cité quelques exemples de techniques de contrôle d'accès dans ce genre de système. Nous avons également cité plusieurs techniques d'authentification et défini également ce qu'est une authentification forte, nous allons au cours du chapitre suivant détailler une technique d'authentification forte que nous avons implémenté afin de garantir un meilleur control d'accès aux données contenues dans le SIH.

Nous avons également constaté qu'il existe un choix assez large en ce qui concerne les SIH open source, ceux-ci peuvent constituer des plateforme intéressantes afin d'y implémenter nos travaux, cependant nous avons choisit une autre alternative, qui consiste en l'utilisation d'un système d'information développé au sein d'un projet de PFE de licence, notre choix s'est orienté vers cette plateforme à la place des SIH open source précédemment cités en raison la simplicité d'intégration de notre solution au sein de cette plateforme en comparaison avec les autres.

CHAPITRE III :

Conception et mise en œuvre

3.1 Introduction

Nous avons au cours du chapitre précédent défini les notions principales de sécurité dans les systèmes d'information hospitaliers, nous avons notamment parlé de la sensibilité des données contenues dans ce genre de système et de la nécessité d'en garantir l'accès, seulement au personnel médical adéquat.

Pour garantir la sécurité ainsi que le caractère privé des DMP, la plupart des approches emploient des techniques de cryptographie et de contrôle d'accès basées sur des cartes à puce électroniques. Les cartes à puce électroniques sont généralement utilisées pour :

- Permettre aux patients (détenteurs du DMP) ainsi qu'aux professionnels de santé de s'authentifier.
- Signer électroniquement des documents afin de garantir leur authenticité.
- Chiffrer les DMP avant de les stocker sur les serveurs
- Autoriser l'accès aux données médicales. Un exemple de cette approche est la carte électronique de santé Allemande [11] ou bien le « Taiwan electronicmedical Record Template (TMT) » [12].

Le chiffrement point à point entre les professionnels de santé est requis. Les diagnostics sont souvent entrés dans le système bien après que le patient ait quitté le cabinet médical. Les données ne pourront pas être saisies sur le dossier du patient, vu que celui-ci n'est également pas physiquement présent.

Si un patient est trop souffrant (dans les cas d'urgences par exemple) il peut être admis inconscient, et ne plus être en mesure de donner l'autorisation d'accès à ses données médicales. (Il ne sera pas en mesure de fournir son code PIN, même si les urgentistes sont en possession de sa carte électronique).

Tous ces problèmes montrent qu'il existe un conflit entre sécuriser l'accès aux données des patients et garder le système exploitable en garantissant la fourniture de soins optimaux aux patients. Un autre problème est que les cartes électroniques doivent être connectées en local à un dispositif relié au poste du professionnel de santé, ce qui implique qu'une autorisation d'accès ou de modification envoyée à distance depuis internet par exemple n'est pas possible.

Par exemple dans le système Allemand [11], la carte doit être connectée à un dispositif de chiffrement et de déchiffrement dans le cabinet du médecin, le patient devra alors entrer un code PIN afin que le médecin puisse accéder à son dossier. Ceci veut dire que le patient doit être physiquement présent dans le cabinet du médecin pour l'autoriser à accéder à son dossier, ceci implique également que le médecin ne pourra pas, par exemple entrer des résultats d'analyse, ou demander l'expertise d'un confrère si le patient n'est pas physiquement présent.

Nous allons au cours de ce chapitre, détailler la solution de control d'accès que nous avons implémenté dans notre travail.

3.2 La solution de contrôle d'accès que nous proposons

Etant donné les inconvénients que nous avons cité pour les cartes à puce électroniques, et étant donné également leur indisponibilité actuelle en Algérie, nous avons choisit de ne pas les utiliser dans la solution de contrôle d'accès que nous proposons, nous les avons remplacé par les Smartphones et l'application « Google Authenticator ». Nous nous servirons de cette application afin d'ajouter un autre facteur d'authentification des utilisateurs, en plus des mots de passe et des noms d'utilisateurs traditionnels et de garantir un meilleur contrôle d'accès aux dossiers des patients, ce deuxième facteur d'authentification sera un HMAC time based one time passeword (TOTP).

3.3 Présentation du HMAC time based one time password (TOTP)

Apporter une méthode afin de vérifier l'intégrité d'une information transmise au travers d'un réseau ou stockée sur un périphérique à la sécurité peu fiable est une nécessité dans un monde de communication ouverte. Les mécanismes qui fournissent ce genre d'intégrité en se basant sur une phrase secrète sont généralement appelés « Message Authentication Code » (MAC).

En général, les MAC sont utilisés entre deux parties se partageant un secret dans le but de valider une information transmise entre eux. Nous allons présenter un mécanisme MAC basé sur une fonction de hachage cryptographique. Ce mécanisme appelé HMAC est basé sur le travail de [16]. Nous nous référerons à ce travail pour les questions de sécurité.

HMAC peut être utilisé en combinaison avec n'importe quelle fonction itérative cryptographique de hachage. MD5 et SHA-1 sont des exemples de ce genre de fonction. Il utilise également un secret partagé pour le calcul et la vérification de l'intégrité des valeurs d'authentification du message. Les objectifs principaux derrière cette construction sont :

- D'utiliser sans modification les fonctions de hachage disponibles. En particulier les fonctions qui sont performantes et dont le code source est disponible gratuitement et librement transmis.
- D'utiliser et manipuler les clés de façon simple
- D'avoir une analyse correcte de la sécurité cryptographique du mécanisme d'authentification basée sur une hypothèse raisonnable sur la fonction de hachage.
- De garantir la facilité de remplacement de la fonction de hachage si le besoin s'en fait sentir dans le cas par exemple de la découverte d'une fonction plus rapide ou plus sécurisée. Ou bien dans l'éventualité d'une avancée majeure dans les mathématiques ou l'informatique qui rendrait la fonction que nous utilisons vulnérable.

Nous allons spécifier HMAC en utilisant une fonction de hachage générique, (notée H). Pour une implémentation spécifique, il sera nécessaire de choisir une fonction du genre MD5, SHA-1, Ces différentes implémentations seront notée HMAC-SHA-1, HMACMD5.

Néanmoins SHA-1 est la fonction la plus utilisée dans le HMAC au jour où nous écrivons ces lignes.

Note : au jour où nous écrivons ce document, SHA-1 semble être cryptographiquement sûre, tout du moins pour son utilisation pour le HMAC, en effet SHA-1 n'est plus considérée comme une fonction de hachage sûre pour les autres applications.

3.3.1 Définition du HMAC

La définition du HMAC exige une fonction de hachage cryptographique, que nous notons H, et une clé secrète K. On suppose que H est une fonction de hachage cryptographique où les données sont hachées en itérant une fonction de compression de base sur les blocs de données. On note B la longueur en octets de ces blocs (B = 64 pour tous les exemples de fonctions de hachage susmentionnés) et L la longueur en octets des résultats du

hachage (L = 16 pour MD5, L = 20 pour SHA-1). La clé d'authentification K peut être de n'importe quelle longueur jusqu'à B, la longueur de bloc de la fonction de hachage. Les applications qui utilisent des clés plus longues que B octets vont d'abord hacher la clé en utilisant H et ensuite utiliser la chaîne d'octets résultante L comme clé réelle pour HMAC. Dans tous les cas la longueur minimale recommandée pour K est L octets (comme la longueur du résultat du hachage).

On définit deux chaînes ipad et opad fixes et différentes comme suit (le 'i' et le 'o' sont des mnémoniques pour interne et externe) :

Ipad = l'octet 0x36 répété B fois.

Opad = l'octet 0x5C répété B fois.

Pour calculer HMAC sur les données "text" on effectue : H (K XOR opad, H(K XOR ipad, text)

1. À savoir : Ajouter des zéros à la fin de K pour créer une chaîne d'octets B (par exemple, si K est de 20 octets et B = 64, alors 44 octets zéro 0x00 seront ajoutés à K).
2. Combiner avec l'opérateur OUX (OU exclusif au bit près) la chaîne d'octets B calculée à l'étape (1) avec ipad
3. Ajouter le flux de données 'text' à la chaîne d'octets B résultant de l'étape (2)
4. Appliquer H au flux généré à l'étape (3)
5. Combiner avec l'opérateur OUX (OU exclusif au bit près) à la chaîne d'octets B calculée à l'étape (1) avec opad
6. Ajouter le résultat H de l'étape (4) à la chaîne d'octets B résultant de l'étape (5)
7. Appliquer H au flux généré à l'étape (6) et sortir le résultat.

3.3.2 Les clés

Les clés dans HMAC peuvent être de n'importe quelle longueur (les clés plus longues que B octets sont d'abord hachées en utilisant H). Cependant une clé moins longue que L octets est moins recommandée, parce qu'elle va diminuer la sécurité de la fonction. Des clés plus longues que L sont acceptables mais ne vont pas significativement augmenter la sécurité de l'algorithme (une clé plus longue pourrait être recommandée si le caractère aléatoire de la clé est considéré comme étant faible).

Les clés doivent être choisies de façon aléatoire (ou en utilisant un générateur pseudo aléatoire cryptographiquement sûr et alimenté par un germe aléatoire). Ces clés devront aussi être périodiquement changées. (Les attaques connues actuellement ne nous font pas préconiser une fréquence spécifique pour le changement de clés, parce que ces attaques sont infaisables (irréalisables) en pratique. Cependant le changement périodique des clés secrètes est une pratique de sécurité fondamentale qui va aider à se protéger contre les faiblesses potentielles de la fonction et des clés utilisées et limiter les dommages causés par une compromission de clé.

3.3.3 Résultat tronqué

Une pratique très répondue avec les codes d'authentification par message est de tronquer la sortie du MAC et ne donner en sortie qu'une partie des bits, La troncation a des avantages (moins d'informations dans le résultat du hachage pour un attaquant) et des désavantages (moins de bits à prédire pour un attaquant). Les applications pour la HMAC peuvent choisir de tronquer la sortie de l'algorithme en ne donnant que les T bits les plus à gauche. Il est recommandé dans [14], que T ne soit pas inférieur à la moitié de la longueur du message en sortie de la fonction de hachage, afin de correspondre à la limite de l'attaque par anniversaires et qu'il ne soit pas inférieur à 80 bits (une limite acceptable au nombre de bits pour qu'un attaquant ne puisse pas prédire la sortie facilement). [9] propose de désigner une réalisation de HMAC qui utilises une fonction de hachage H avec T bits en sorties par HMAC-H-T par exemple HMAC-SHA1-80 désigne un HMAC utilisant SHA-1 avec une sortie tronquée à 80 bits. Si le paramètre t n'est pas défini, nous voudrons dire que la sortie n'est pas tronqu.

Pour une implémentation correcte des mécanismes du HMAC le choix de secrets aléatoires ou pseudo aléatoires à l'aide de générateurs cryptographiques, un mécanisme d'échange de clés sécurisé, un changement de clés périodique ainsi qu'une bonne protection des clés sont les ingrédients essentiels pour la sécurité des mécanismes d'authentification fournis par la HMAC.

A présent que nous avons défini le HMAC, nous allons donner un exemple de calcul d'un HMAC TOTP en utilisant la fonction de hachage SHA-1

3.4 Exemple de calcul d'un HMAC TOTP

Algorithme TOTP

K est la clef secrète (clef partagée).

T : est le compteur de temps.

Etape 1 :

Génération d'un HMAC-SHA-1

HMAC (K, TC) = SHA1 ((K xor 0x5c5c…) || SHA1 ((K xor 0x3636… || TC)).

Retourne 20 octets (longueur du HMACSHA1).

Etape 2 :

Extraction d'une chaine de 31 bits à partir du HMAC calculé à l'étape 1

Sbits = DT (HMAC (K, TC))

La fonction DT (DynamicTruncation) est détaillée plus bas elle retourne une chaine de 31 bits

Etape 3 :

Génère un token à partir de la chaine Sbits calculée à l'étape 2

Snum = StToNum(Sbits)

StToNum convertit Sbits en décimal

T = Snummod 10^Digit

Digit : longueur du token : 6 dans notre exemple (ainsi que dans la majorité des cas)

mod : modulo

T : Retourne une valeur comprise entre 0 et 999999 qui correspond au token

Description de la fonction DT (DynamicTruncation) de l'étape 2

DT(String) String, qui est le HMAC calculé à l'étape 1, est une chaine composée de 20 octets (String[0]…String[19]).

Nous allons extraire les 4 bits de poids faible de l'octet 19. La valeur est contenue dans OffsetBits

OffsetBits est donc compris entre 0 et 15

OffsetBits donne la position dans String pour extraire les 4 octets

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9

xx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Les 4 derniers bits (x) donnent la position où sont extraits les 4 octets.

Offset = StToNum(OffsetBits)

Extraction de 4 octets à partir de OffsetBits

Retourne 31 bits de Offset (le bit de signe est positionné à 0)

Exemple :

Si OffsetBits est égal à 5 (0101 en binaire),

On extrait aabb cc dd à partir de l'octet 5

Offset 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9

xxxxxxxxxxaabb cc dd xx xxxxxxxxxxxxxxxxxxxx

Puis on positionne à 0 le bit le plus significatif

aa b b c c d d

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

=> 0xxx xxxxxxxxxxxxxxxxxxxxxxxxxxxx

La raison de masquer le bit le plus significatif (bit de signe) est d'éviter des erreurs de calcul (bit de signe) en effet selon le type de processeur qui effectue ce calcul, il peut en résulter des erreurs si nous ne l'enlèvons pas, donc pour lever toute ambiguïté l'algorithme supprime le bit de signe.

3.5 Préparation des données

K le secret partagé est saisie en base 32 afin d'éviter les erreurs de saisie :

K = LVACVMXXD4AIPVLFZ6QL337GS2MM7X5Y

Conversion en hex : k = 5d402ab2f71f0087d565cfa0bdefe69698cfdfb8.

C (compteur) est le compteur de temps

1400320015 = Samedi 17 mai 2014 / 09h 46m 55s

TC = (unixtime(now) – unixtime(T0)) / TS (1400320015 /30) = 46677333

TS est généralement égale à 30, cela permet de découper le temps en tranche de 30 secondes

Conversion en hexa :

46677333d = 0x2C83D55

Conversion en 8 octets :

0000000002C83D55

Les données sont préparées il faut maintenant appliquer les 3 étapes de l'algorithme :

Etape 1 : HMACSHA1

HMAC (K,TC) = SHA1((K xor 0x5c5c…) || SHA1((K xor 0x3636… || TC))

HMAC (5d402ab2f71f0087d565cfa0bdefe69698cfdfb8, 0000000002C83D55)

On obtient: e562f2dadefc81a384551c3278d5ec42417ab4da

Etape 2 : extraction des 4 octets et positionnement à 0 du bit de signe

Extraction des 4 octets:

offset

Numéro de l'octet

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

Valeur de l'octet

e5 62 f2 da de Fc 81 a3 84 55 1c 32 78 d5 ec 42 41 7a b4 da

Le dernier Octet (le 19ème octet) possède la valeur 0xda.

Les 4 derniers bits du 19ème octet donnent l'offset : 0xa (10d, (d comme décimal)).

Nous allons donc extraire les 4 octets à partir de la 10ème colonne du tableau.

Les 4 octets à extraire sont donc : 1c 32 78 d5

Positionnement à 0 du 31ème bit.

1c 32 78 d5 on ne conserve que les 31 bits.

(Le dernier bit à gauche est positionné à 0)

1c 32 78 d5 = 00111000011001001111000011010101

Dans cet exemple, le bit de signe étant déjà égal à 0, cela ne change rien

Etape 3 : Génération du token

Conversion en décimal :

1c3278d5 = 473069781

Modulo 10^6

473069781 modulo 10^6 = 069781

Nous obtenons le token : 069781

3.6 Description de notre API

L'algorithme TOTP est utilisé pour déterminer si un mot de passe fournit par un utilisateur est valide. Cette vérification se fait par le serveur de validation, en sachant la phrase secrète partagée entre les deux et l'heure posix actuelle.

Les deux opérations de base que cette API effectue sont :

- La création d'identifiants
- L'authentification d'identifiants.

La création d'identifiant est le processus au travers duquel le serveur va générer la phrase secrète et la partager avec le client. La phrase secrète est générée en utilisant un algorithme cryptographique fort de génération pseudo aléatoire et doit être communiquée au client pour qu'il puisse la stocker et configurer son token.L'utilisateur n'a besoin que de scanner le QR code avec l'application Google Authenticator et s'en débarrasser tout de suite après. Le QR Code est un code barre à 2 dimensions qui permet de stocker des informations numériques.

Peut-être déchiffré à partir d'un téléphone mobile équipé d'un appareil photo et du lecteur approprié. De cette façon seul les individus en possession physique du générateur peuvent utiliser cette phrase pour générer un mot de passe TOTP. De plus ceci permet également de prévenir qu'un éventuel pirate qui entre temporairement en possession du téléphone d'un patient ne puisse voler le secret de celui-ci pour usurper son identité ultérieurement.

3.6.1 Authentification des identifiants

Authentification des identifiants :

C'est le processus durant lequel le serveur d'authentification applique l'algorithme TOTP sur la phrase secrète et vérifie que le TOTP fournit par le client est valide.

La création d'un secret est d'une importance capitale, parce que la capacité de deviner la phrase secrète d'un autre peut aboutir à la prise de contrôle d'un compte utilisateur.

En plus du mot de passe et du nom d'utilisateur, le professionnel de santé doit fournir un TOTP avant de pouvoir se connecter à son compte utilisateur.

3.6.2 Création d'un compte utilisateur ou d'un compte de patient

L'administrateur du système devra remplir les informations relatives au patient ou au professionnel de santé pour créer un compte pour cette personne. Le générateur pseudo aléatoire va générer par la suite automatiquement un QR code représentant le secret, qui sera scanné par l'utilisateur à l'aide de Google Authenticator. Pour la génération de nos secrets nous nous sommes basés sur le générateur pseudo aléatoire fourni par Oracle dans java. Cependant, il existe plusieurs algorithmes de génération de nombres pseudo aléatoires. Il existe également plusieurs tests afin de déterminer le caractère aléatoire d'un générateur. Nous recommandons cependant de suivre les recommandations de [15] pour toute génération de nombres pseudo aléatoires.

3.7 Discussion de notre proposition

Etant donné que nous nous basons sur un générateur HMAC SHA-1 TOTP pour la génération de nos tokens, la sécurité de notre architecture dépend principalement de la sécurité de la fonction de hachage SHA-1 appliquée avec le HAMC- TOTP.

3.7.1 Discussion de la sécurité du HAMC SHA-1 TOTP

Le schéma d'authentification à double facteur basé sur un TOTP requiert en général que l'utilisateur envoie son TOTP en même temps que ses identifiants. Le TOTP est basé sur le temps ou l'heure et est calculé à l'aide d'un algorithme cryptographique et sa période de validité est très courte. Il est recommandé une taille de fenêtre de 30 secondes. Cependant dans notre approche, nous avons choisi une taille de fenêtre de 60 secondes. Pour éviter les problèmes d'authentification à cause des latences dans les transmissions réseau, le serveur d'authentification vérifie en général les TOTP dans la fenêtre dans laquelle le message arrive, et les deux fenêtres en arrière L'analyse de sécurité [13] de l'algorithme conclu que la meilleure attaque possible contre une fonction HOTP est une attaque par force bute. La sécurité de l'algorithme peut être évaluée par :

$$\text{Sécurité} = \frac{S.v}{10^d}$$

Où s est la taille de la fenêtre d'anticipation pour la resynchronisation, v est le nombre de tentatives de vérification infructueuse autorisée avant le blocage du compte et d est le nombre de chiffres composant un mot de passe TOTP.

Si une attaque par force brute survient avec v comme tentative de vérification infructueuse autorisée, alors sa probabilité de succès sera de p = $\frac{S.v}{10^d}$[13].

[13] recommande également que d soit de 6 chiffres, et c'est également la longueur du mot de passe que nous avons utilisé. Nous pouvons cependant aller jusqu'à 9 chiffres, qui est le maximum à pouvoir être représenté par le String de 31 bits qui sera retourné par la fonction de truncation.

Pour plus de détails sur la sécurité du TOTP consulter [13].

La sécurité du HMAC dépend également de la sécurité de la fonction de hachage utilisée, la résistance aux collisions en supposant que la valeur initiale est secrète et aléatoire et où les sorties de la fonction ne sont pas explicitement disponibles pour un attaquant.

Ses propriétés sont généralement présumées pour une fonction de hachage avant qu'elle ne soit utilisée avec le HMAC. Une fonction de hachage n'assumant pas ces propriétés est de toute façon inadaptée pour la plupart des applications cryptographiques. Pour une analyse complète des propriétés de sécurité de HMAC voir [16].

Etant donnée la confiance limitée dans la sécurité des fonctions de hachage observée jusqu'à présent, il est important d'observer ces deux propriétés lors de l'implémentation du HMAC :

1. L'implémentation doit être indépendante de la fonction de hachage H et donc cette fonction doit pouvoir être remplaçable à n'importe quel moment.
2. Par opposition au chiffrement, le message d'authentification à un effet éphémère. La publication d'un exploit sur un schéma d'authentification aura pour effet l'abandon de celui-ci mais par opposition au chiffrement il n'aura aucun impact sur les sessions d'authentification précédentes, tandis que le chiffrement sera complètement compromis au cas où un algorithme de chiffrement est cassé.

L'attaque la plus dangereuse connue contre le HMAC est basée sur la fréquence de collisions de la fonction de hachage H.

Note : ce genre d'attaque est très différent des attaques habituelles sur les fonctions de hachages où il n'y a pas de clé secrète et où 2^{64} opérations suffisent à trouver des collisions. Cette dernière est jugée faisable tandis qu'une « birthdayattack » sur le HAMC est jugée

infaisable. Nous pouvons en effet limiter le nombre de tentatives de connexion infructueuses, de plus la validité du TOTP est limitée dans le temps, ce genre d'attaque à une très faible probabilité de réussite.

Pour une implémentation correcte des mécanismes du HMAC, le choix de secrets aléatoire ou pseudo aléatoires à l'aide de générateurs cryptographiques, un mécanisme d'échange de clés sécurisé, un changement de clés périodique ainsi qu'une bonne protection des clés sont les éléments essentiels pour la sécurité des mécanismes d'authentification fournis par la HMAC.

Conclusion

Nous avons au cours de ce chapitre, détaillé l'algorithme HMAC TOTP et discuté de sa sécurité, nous allons au cours du suivant, détailler notre implémentation.

CHAPITRE IV :

Implémentation

4.1 Introduction

Après notre étude conceptuelle, nous allons passer à la partie implémentation de notre application où nous expliquerons son mode de fonctionnement et où nous allons présenter et les différents outils logiciels utilisés pour sa réalisation.

4.2 Choix du langage de programmation

Java : Java est un langage de programmation et une plate-forme informatique qui ont été créés par Sun Microsystems en 1995, rachetée plus tard par Oracle Corporation. Beaucoup d'applications et de sites Web ne fonctionnent pas si Java n'est pas installé et leur nombre ne cesse de croître chaque jour. Java est rapide, sécurisé et fiable. Il est utilisé partout, des ordinateurs portables aux centres de données, des consoles de jeux aux superordinateurs scientifiques, des téléphones portables à Internet, la technologie java est présente sur tous les fronts. Nous avons opté pour ce langage pour la réalisation de notre application logicielle.

4.3 NetBeans IDE 8.0

L'IDE NetBeans : C'est un environnement de développement intégré basé sur Java (IDE) qui est également désigné comme une plate-forme de composants utilisés pour développer des applications Java.

L'IDE NetBeans : Est un environnement de développement intégré basé sur Java (IDE) qui est également désigné comme une plate-forme de composants utilisés pour développer des applications Java.

L'IDE est conçu pour limiter les erreurs de codage et faciliter la correction d'erreur avec des outils tels que les NetBeans FindBugs ou le Debugger pour gérer du code complexe, les points d'arrêt et le suivi de l'exécution. Bien que NetBeans est conçu spécifiquement pour les développeurs Java, il peut supporter également d'autres langages comme C, C++, PHP ,XML, HTML en plus de JavaScript ... Cet outil inclu un éditeur de texte riche en fonctionnalités avec des outils de Refactoring (ré-usinage de code) et des modèles de code, un glisser-déposer pour permettre la conception de l'interface graphique... L'IDE NetBeans peut fonctionner sur tout système d'exploitation qui prend en charge une JVM compatible, y compris Linux et Windows.

La plate-forme prend en charge la création de nouvelles applications et le développement des applications existantes en utilisant des composants logiciels

modulaires. Cette plateforme est en elle-même extensible et peut être étendue pour supporter de nouveaux langages. Il a été convertit à l'open source par Sun Microsystems en juin 2000.

Cet IDE est donc le choix idéal pour notre application. L'installation de l'IDE comprend l'installation du logiciel NetBeans lui-même et l'installation du JDK - ou Java Development Toolkit.

4.4 Google Authenticator

L'application Google Authenticator est certainement le générateur de token one time password la plus largement utilisée à travers le monde. Elle est également disponible gratuitement sur tous les principaux OS mobiles. Elle est de ce fait probablement le générateur TOTP standard à travers le monde et n'est pas seulement utilisée pour la génération d'OTP pour les comptes Google mais aussi pour d'autres comptes appartenant à d'autres compagnies.

Google Authenticator est une application gratuite, flexible, facile à utiliser et à configurer. Elle peut être exploitée par n'importe quel système à condition qu'il dispose des mécanismes qui lui permettent de valider un TOTP. De plus elle a été open source jusqu'à sa version 5.0. Ceci nous a poussé à l'utiliser lors de notre mise en œuvre. Cependant vu que le TOTP que nous avons utilisé est conforme aux standards internationaux en termes de fonction de hachage notamment, l'utilisateur final pourra choisir d'utiliser n'importe quelle autre application de Smartphone qui génère des tokens TOTP compatibles. Nous pouvons citer l'application pour Android freeOTP disponible gratuitement sur le Play store (boutiques d'applications Android en ligne) de Google comme une autre alternative à Google Authenticator. [4]

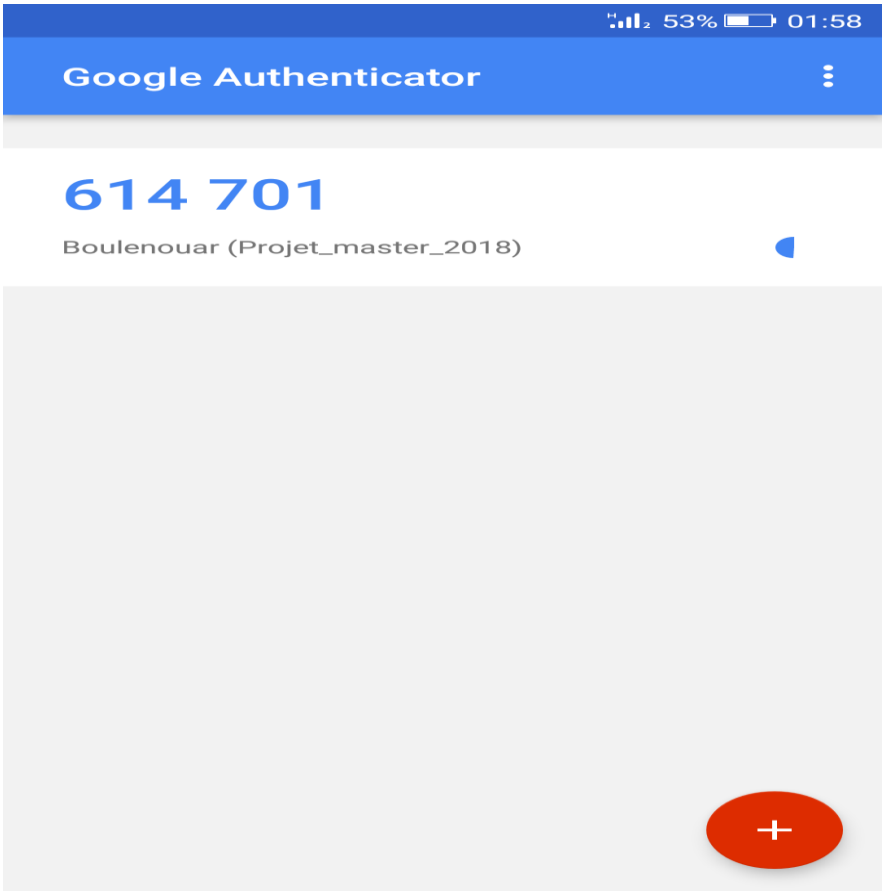

Figure 3:Interface Google Authenticator

4.4.1 Interfaces d'accueil de l'application

Cette Figure ci-dessous montre la première interface de l'application.

En plus du mot de passe et du nom d'utilisateur, le professionnel de santé doit fournir un TOTP avant de pouvoir se connecter à son compte utilisateur.

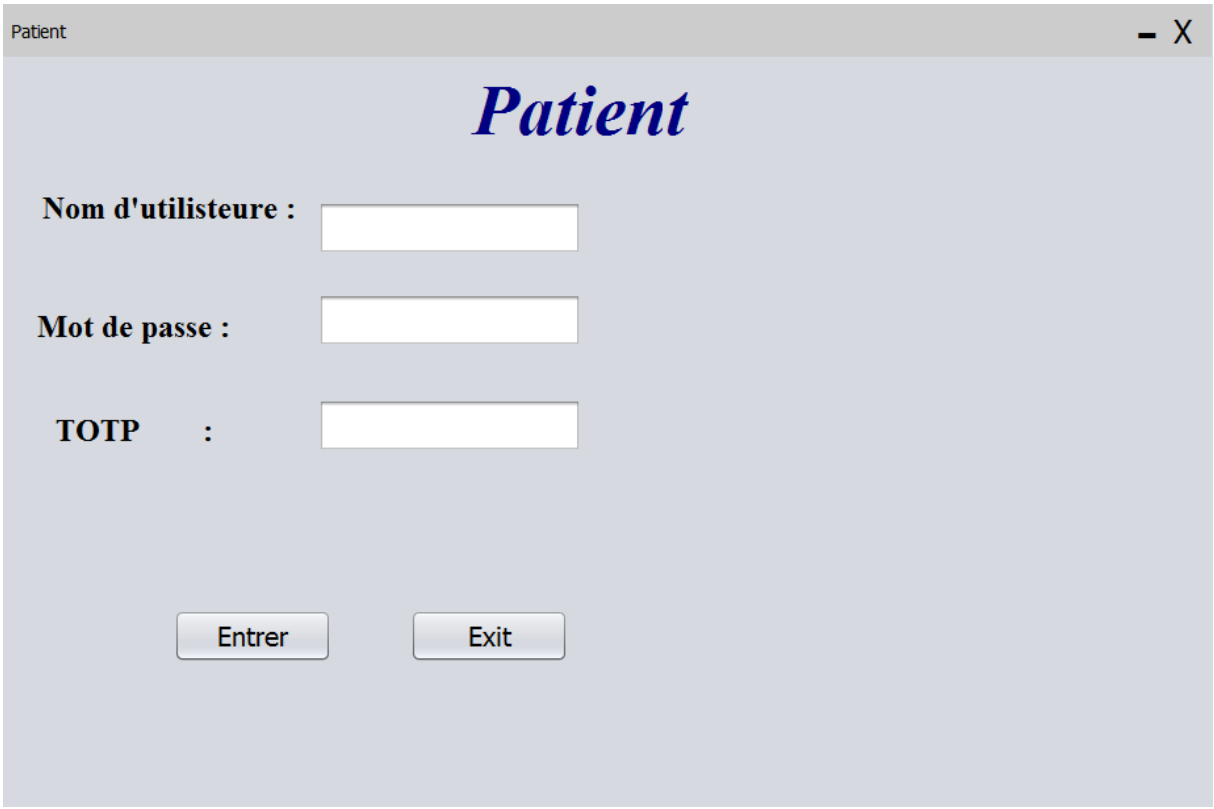

Figure 4:Interface d'accueil de l'application

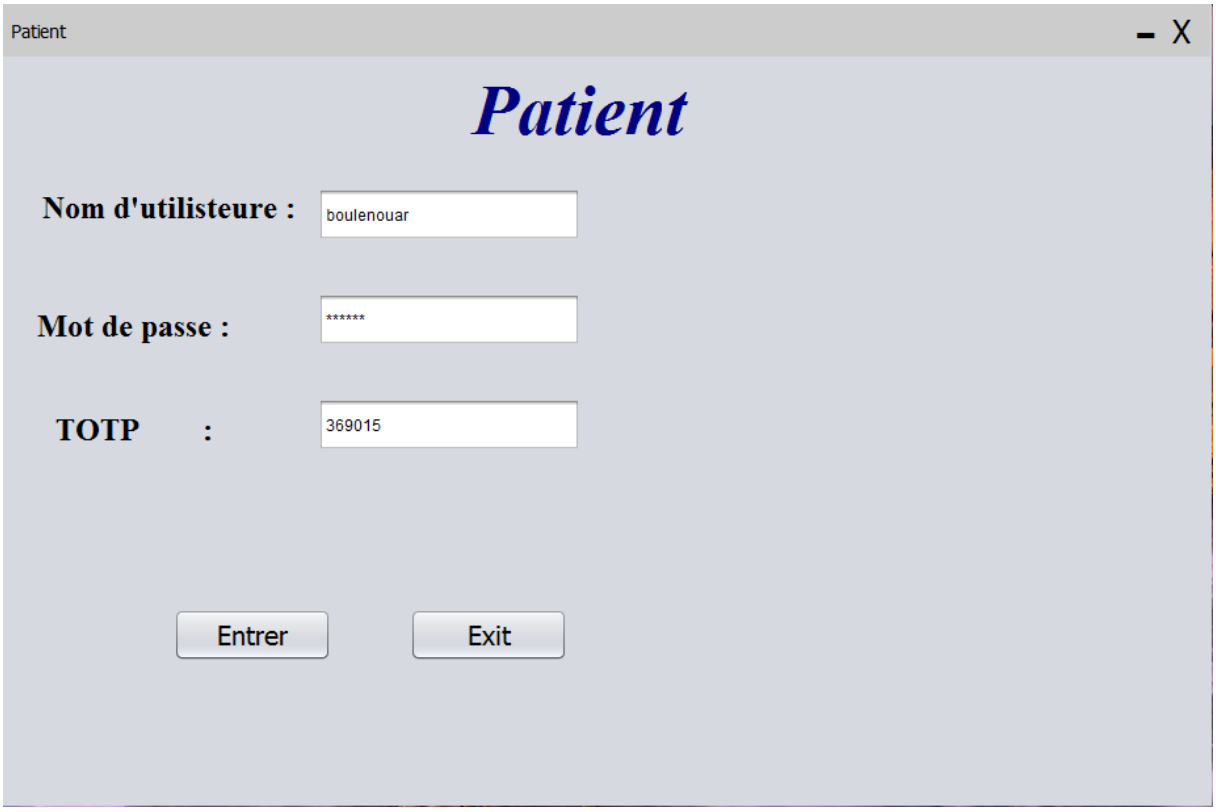

Figure 5:Interface génération du code TOTP

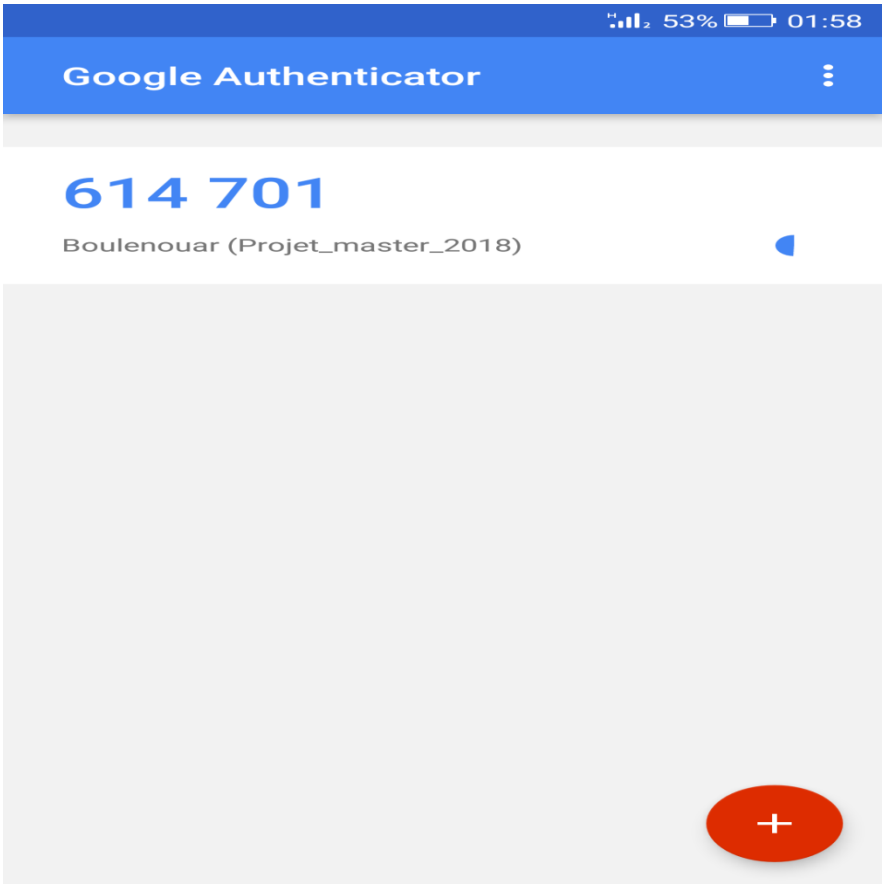

Figure 6:Interface Google Authenticator

Après avoir saisi le code secret **TOTP** et cliqué sur le bouton **Entrer,** si les informations saisies sont correctes, à savoir le TOTP (qui est correcte et encore valide) et le mot de passe ainsi que le nom d'utilisateur, l'accès l'application est autorisé. Est la fenêtre représentée dans la figure 7 est affichée.

Figure 7:Interface Accueil

La figure 8 montre un message d'erreur l'ors de la saisie d'un code secret **TOTP** erroné ou périmé. Elle peut également survenir si le mot de passe ou le nom d'utilisateur ne sont pas corrects.

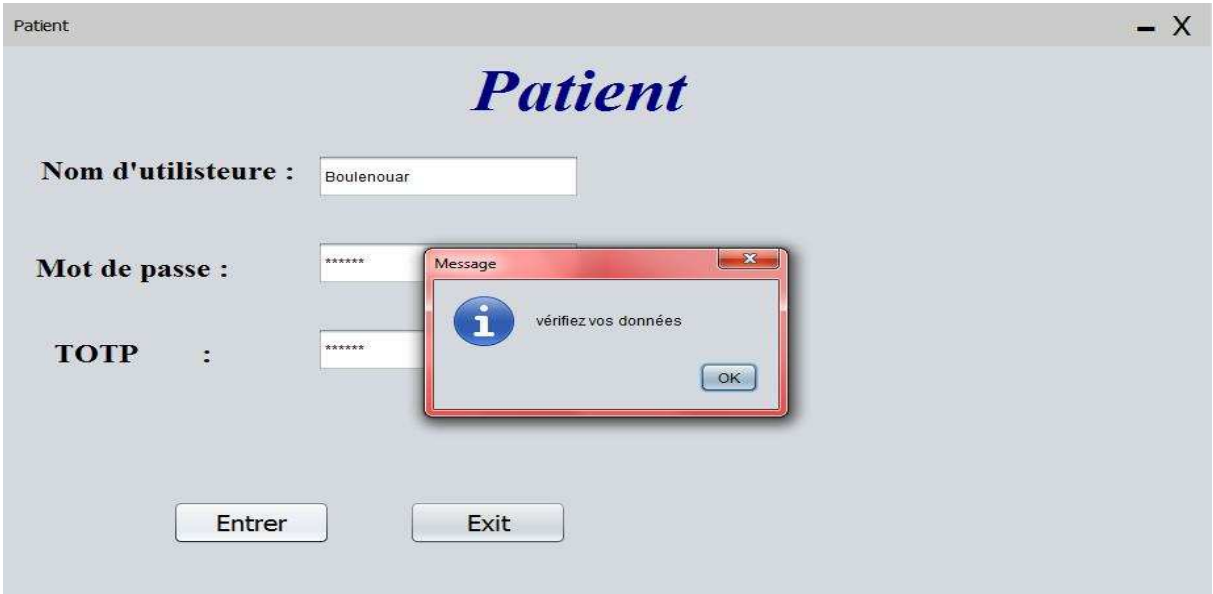

Figure 8:Interface génération du code TOTP erroné

Cette interface vous permet d'ajouter de nouveaux utilisateurs. Grâce à cette interface, vous pouvez saisir toutes les informations, personnelles ou autres, pour chaque utilisateur. Ces informations peuvent être modifiées plus tard à partir de cette interface. Il est également possible de supprimer un utilisateur et de le rechercher.

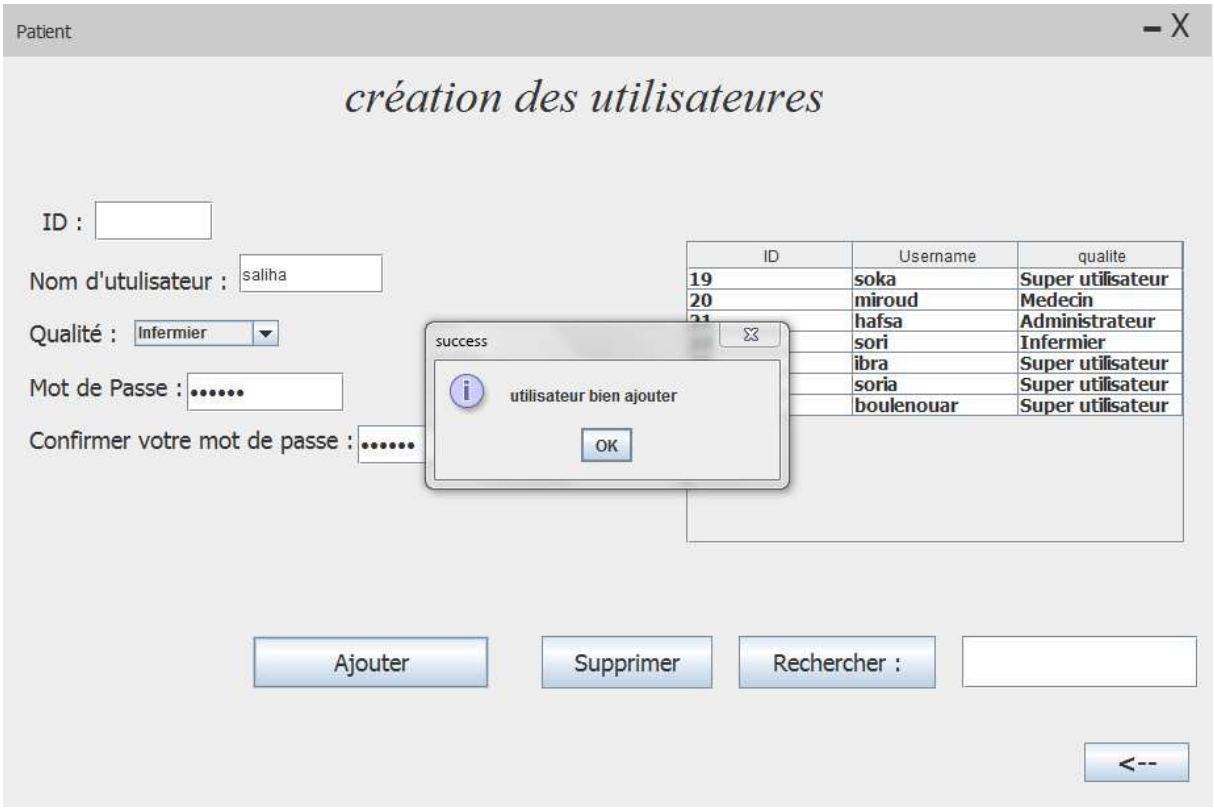

Figure 9:Interface création utilisateures

La figure précédent montre que l'utilisateur saisie les informations exigés par cet interface ensuite il clique sur le bouton ajouter si les informations sont saisies correctement une boite de dialogues s'afficher pour confirmer que l'utilisateur à bien été ajouté avec succès.

En cliquant sur le bouton ajouter, une phrase secrète est automatiquement générée par l'application et elle est affichée à l'utilisateur grâce à un QR code. L'utilisateur devra scanner ensuite le QR code grâce à l'application google authenticator.

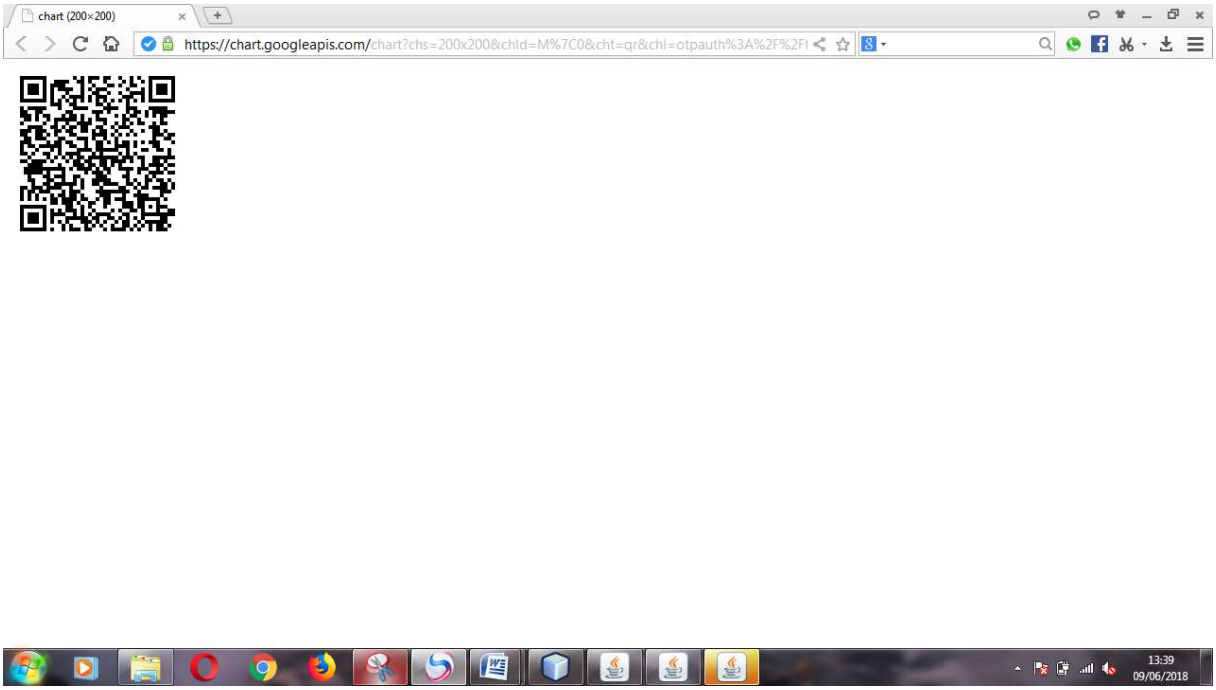

Figure 10:Exemple de QRCode généré

La figure 10 montre un exemple de QR code généré par l'application.

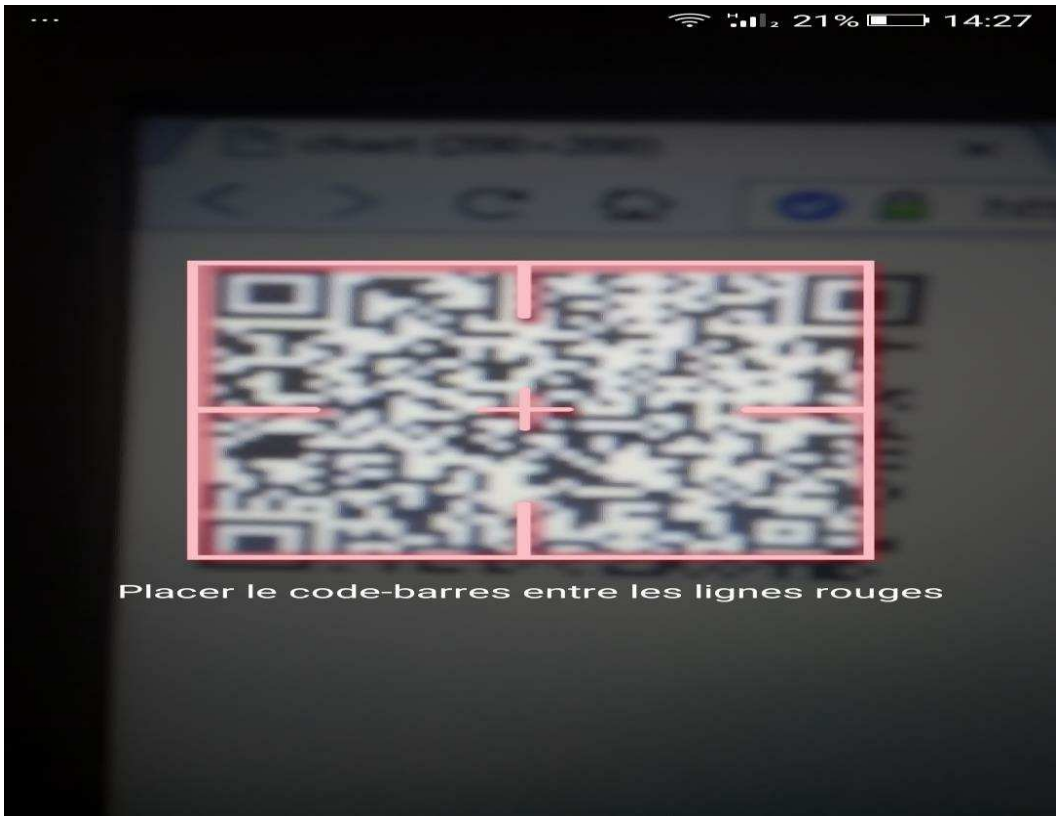

Figure 11:Interface google authenticator scanné QRCode

La figure 11 montre la phase de scanne de la phrase secrète grâce à l'application Google authenticator.

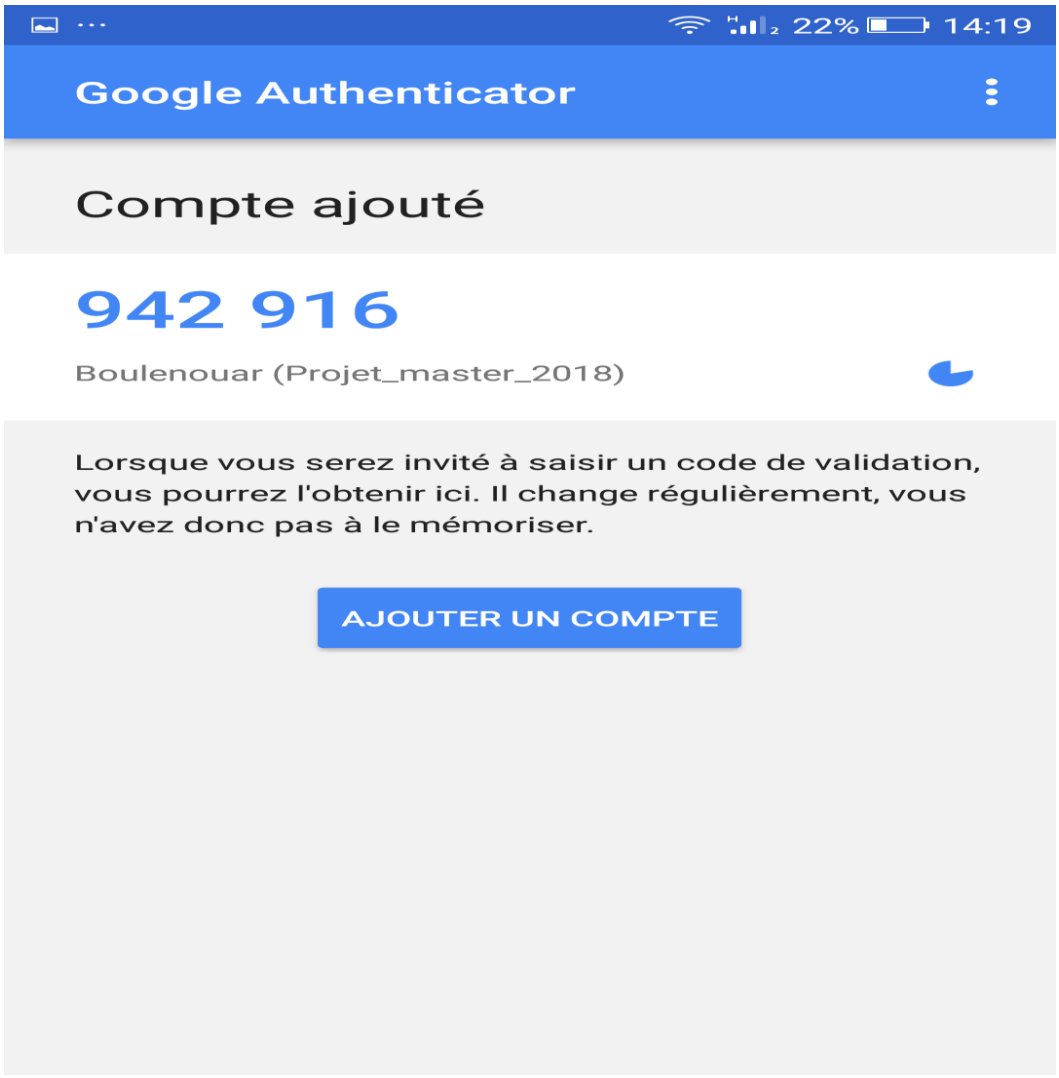

Figure 12:Interface google authenticator ajouter un compte

Après avoir scanné le QR code, le TOTP est affiché et il changera automatiquement après chaque fenêtre. (figure 12)

Conclusion

Dans cette dernière partie, nous avons, en premier lieu, présenté les différents outils et langages de programmation utilisés pour implémenter notre application.par la suite, nous sommes passés à l'explication des interfaces de l'application à travers la présentation des différentes interfaces permettant d'accéder à ses différentes fonctionnalités.

Conclusion Générale

La smart health est en train de se développer rapidement, les menaces qui pèsent sur ce genre de systèmes se développent elles aussi à mesure que leur utilisation croit. Une des principales choses dont il faut tenir compte en sécurisant ce genre de système est le fait que les données qui y sont contenues sont toutes et exclusivement la propriété du patient. Donc lui seul peut en disposer et décider qui peut y accéder et quand.

Aucour de ce mémoire nous avons dans le premier chapitre défini les concepts fondamentaux de la smart health. Dans le deuxième chapitre, nous avons parlé de la sécurité des systèmes d'informations, de la sécurité des système de smart health et cité quelque techniques de contrôle d'accès qui garantisse la confidentialité des données sensible qu'ils contiennent.

Dans le troisième chapitre, nous avons détaillé l'algorithme HMAC TOTP, et discuté de sa sécurité, dans le quatrième et dernier chapitre, nous avons détaillé notre implémentation.

Comme perspectives futures à nos travaux, nous pourrions envisager d'utiliser la cryptographie afin de chiffrer les données contenues dans les SIH. Cependant avec l'indisponibilité des cartes à puce, il faudra trouver une alternative à celle-ci afin de garantir la sécurité des clés de chiffrement.

Bibliographie

[1] Joëlle Simard, “La ville intelligente comme vecteur pour le développement durable : le cas de la ville de Montréal“, Centre universitaire de formation en environnement et en développement durable de l’université de Sherbrooke, Sherbrooke, 2015.

[2] Nathalie Crutzen, “Les villes durables et intelligentes – contexte, définition et présentation du smart city institute“, HEC ULg, Liège, 2015.

[3] Olivier Perroud, “Mobile e-health services“, Université de Fribourg, Fribourg, 2008.

[4] Mohammed El Mustapha MROUD, “La sécurité dans les systèmes de e-santé“, USTOMB, Oran, 2016.

[5] Mekanne salem, Meziane Abdelkrim, “Un modèle de contrôle d’accès pour la protection des données personnelles dans le dossier médical partageable“, Biomedical Engineering International Conference Tlemcen-Algeria, October 15-16, 2014.

[6] Code Algérien, “Le code Algérien de la déontologie médicale“.

[7] Patrice Degoulet, “Systèmes d’informations hospitaliers", Faculté de Médecine Broussais-Hôtel-Dieu, Paris, 2001.

[8] Ahmed Ghali, "Amélioration de la reconnaissance par le visage", USTOMB , Oran, 2015.

[9] National Institute of Standards and Technology, “Secure Hash Standard (SHS) FIPS PUB 180-4, mars 2012 Gaithersburg.

[10] Ron Rivest, "The MD4 Message Digest Algorithm", Advances in Cryptology - CRYPTO '90 Proceedings, Springer-Verlag, 1991, pp. 303-311.

[11] Gematik - "Gesellschaft fur Telematikanwendungen der Gesundheitskarte", Die elektronische Gesundheitskarte, http://www.gematik.de

[12] H.-H. Rau, C.-Y. Hsu, Y.-L. Lee, W. Chen, et W.-S. Jian. "Developing electronic health records in Taiwan", IT Professional, 12:17–25, 2010.

[13] D. M'Raihi, M. Bellare, F. Hoornaert, D. Naccache, O. Ranen,"HOTP: An HMAC-Based One-Time Password Algorithm", Network Working Group, December 2005.

[14] H. Krawczyk, M. Bellare, R. Canetti, "RFC 2104: HMAC: Keyed-Hashing for Message Authentication", Network Working Group, Février 1997.

[15] D. Eastlake, J. Schiller, S. Crocker, "RFC 4086: Randomness Requirements for Security", Network Working Group, June 2005.

[16] Mihir Bellare, Ran Canetti, Hugo Krawczyk, "Pseudorandom Functions Revisited: The Cascade Construction and its Concrete Security", Proceedings of the 37th Symposium on Foundations of Computer Science, IEEE, 1996.

Résumé

Ce travail est une recherche bibliographique qui porte sur les systèmes de smart health ainsi que sur leur aspect sécurité. Nous avons étudié ces systèmes, et le caractère particuliers des données qu'ils contiennent, nous avons par la suite étudié différents systèmes d'informations hospitaliers open source afin de choisir lequel nous allons utiliser afin de développé une nouvelle solution de contrôle d'accès. Enfin, nous en avons choisit un et y avons implémenté une solution de mot de passe à usage unique afin d'optimiser la sécurité au sein de ce système.

Mots clés : smart health, sécurité des systèmes d'information, contrôle d'accès, e-santé, SIH open source. HMAC one time passe word