REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ SAAD DAHLAB BLIDA

Faculté des Sciences
Département : Informatique

Mémoire de fin d'étude pour l'obtention du diplôme de Master en Informatique
OPTION : Sécurité des Systèmes d'Information

Présenté par :
HELAL Sonya

Thème

Authentification Anonyme et Contrôle d'Accès dans un Environnement Cloud : Application au Domaine e-santé

Organisme d'accueil: Centre de Recherche sur l'Information Scientifique et Technique (CERIST)

Soutenu le 29/09/2019 devant le jury composé de :

Mme Toubaline	**Présidente**
Mme Ghebghoub	**Examinatrice**
Mme AROUSSI Sana	**Promotrice**
Mrs SAIDI Ahmed	**Encadreur**

Promotion : 2018-2019

Remerciement

*Mes remerciements, avant tout, à **DIEU** tout puissant pour la volonté, la santé et la patience qu'il m'a données durant toutes ces longues années d'études afin que je puisse arriver à ce stade.*

Je remercie mes très chers parents qui ont toujours été là pour moi, « Vous avez tout sacrifié pour vos enfants n'épargnant ni santé ni efforts. Vous m'avez donné un magnifique modèle de labeur et de persévérance. Je suis redevable d'une éducation dont je suis fier ». Je remercie également mon frère Salim, et mes sœurs Sara et Soraya pour leur encouragement.

Je tiens à exprimer toute ma reconnaissance à mon encadreur Madame AROUSSI Sana. Je la remercie pour sa disponibilité et la confiance qu'elle m'a accordée. J'aimerais aussi la remercier pour les soutiens et ses précieux conseils qui m'ont permis de mener à bien ce travail. Je ne la remercierai jamais assez, qu'elle trouve en ce mémoire l'expression de ma profonde gratitude et mon respect infini.

Je remercie tout particulièrement Monsieur SAIDI Ahmed, attaché de recherche au CERIST, promoteur de thèse, de m'avoir orienté, corrigé mon travail et encouragé. Merci pour sa disponibilité et sa gentillesse sans égale. J'ai profité pendant longtemps du savoir et du savoir-faire dont j'ai pu bénéficier au cours de nombreuses discussions.

Je remercie également le service Formation du CERIST de nous avoir prodigué un excellent environnement de travail.

J'adresse mes sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé mes réflexions et ont accepté à me rencontrer et répondre à mes questions durant mes recherches.

Je remercie le membre de jury pour m'avoir fait l'honneur de juger mon travail.

Afin de n'oublier personne, mes vifs remerciements s'adressent à tous ceux qui m'ont aidée à la réalisation de ce modeste mémoire

Tout simplement Merci !

Dédicace

Je dédie ce modeste travail en signe de respect, reconnaissance et de remerciement :

A mes chers parents, qui ne cessent de me donner avec amour le nécessaire pour que je puisse arriver à ce que je suis aujourd'hui. Que dieux vous protège et que la réussite soit toujours à ma portée pour que je puisse vous combler de bonheur.

A mes chers sœurs Sara et Soraya ainsi qu'à mon frère Salim, en reconnaissance de leur affection toujours constante

Tous mes proches

Tous mes amis, mes collègues et tous ceux qui m'estiment.

Résumé

Le cloud computing, ou informatique en nuage est une technologie qui a connu un développement rapide aux cours de ces dernières années. D'un point de vue sécurité de l'information, un certain nombre de questions se posent sur les différentes menaces que font face le Cloud, notamment la confidentialité des utilisateurs et l'intégrité des données stockées. L'objectif de ce mémoire est de mettre en place des solutions à ces deux problématiques dans le cadre de domaine de e-santé.

Pour ce faire, nous avons développé deux méthodes dont chacune fait l'objet d'une contribution. La première contribution fournit un modèle de contrôle d'accès basé sur la méthode de chiffrement à base d'attribut CP ABE, qui permet aux propriétaires de données de garantir la sécurité des données et de fournir aux utilisateurs un accès fin aux données en utilisant des politiques et des contraintes définies. La seconde contribution fournit une authentification anonyme qui est l'une des solutions qui peuvent être utilisées pour préserver la confidentialité des données personnelles c'est-à-dire qu'elle rend les identités des utilisateurs anonyme une fois qu'ils s'authentifient au sein du cloud. Les résultats expérimentaux de notre solution montrent que ces deux méthodes sont efficaces pour gérer en toute sécurité les données stockées dans le cloud et assure la confidentialité des utilisateurs.

Mots – clés : Sécurité cloud computing, confidentialité, intégrité de données, Chiffrement CP-ABE, e-santé, Contrôle d'accès, Authentification anonyme.

Abstract

Cloud computing is a technology that has grown rapidly in recent years. From an information security perspective, a number of questions arise about the various threats facing the cloud, including user privacy and the integrity of stored data. The objective of this thesis is to implement solutions to these two problems within the framework of the e-health domain.

To do this, we have developed two methods, each of which is the subject of a contribution. The first contribution provides an access control model based on the CP ABE attribute-based encryption method, which allows data owners to ensure data security and provide users with fine access to data using defined policies and constraints. The second contribution provides anonymous authentication, which is one of the solutions that can be used to preserve the confidentiality of personal data, i.e. it makes users' identities anonymous once they authenticate in the cloud. The experimental results of our solution show that both methods are effective in safely managing data stored in the cloud and ensuring user confidentiality.

Key – words: Cloud computing security, confidentiality, data integrity, CP-ABE encryption, e-health, access control, anonymous authentication.

ملخص

الحوسبة السحابية هي تقنية شهدت تطوراً سريعاً في السنوات الأخيرة. من منظور أمن المعلومات ، يطرح عدد من الأسئلة حول التهديدات المختلفة التي تواجه السحابة ، بما في ذلك خصوصية المستخدم وسلامة البيانات المخزنة. الهدف من هذه الرسالة هو تنفيذ حلول لهاتين المشكلتين في إطار مجال الصحة الإلكترونية.

للقيام بذلك ، قمنا بتطوير طريقتين ، كل منها موضوع مساهمة. توفر المساهمة الأولى نموذج التحكم في الوصول استنادًا إلى طريقة التشفير المعتمدة على سمة CP ABE ، والتي تتيح لأصحاب البيانات ضمان أمان البيانات وتزويد المستخدمين بوصول جيد إلى البيانات باستخدام سياسات وقيود محددة. توفر المساهمة الثانية مصادقة مجهولة المصدر ، وهي أحد الحلول التي يمكن استخدامها للحفاظ على سرية البيانات الشخصية ، أي أنها تجعل هويات المستخدمين مجهولة بمجرد المصادقة في السحابة. توضح النتائج التجريبية لحلنا أن كلتا الطريقتين فعالتين في إدارة البيانات المخزنة في السحابة بأمان وضمان سرية المستخدم.

الكلمات – المفتاحية: أمان الحوسبة السحابية ، السرية ، تكامل البيانات ، تشفير CP-ABE ، الصحة الإلكترونية ، التحكم في الدخول ، المصادقة المجهولة.

Table des matières

Liste des figures

Liste des tableaux

Liste des acronymes

Acronymes	Signification
ABAC	Attribut Based Access Control
ABE	Attribute Based-Encryption
AT	Access Ticket
ACL	Access Control List
AES	Advanced Encryption Standard
API	Application Programming Interface
BDD	Base De Données
CDPH	Computational Diffie-Hellman Problem
CID	Confidentialité Intégrité Disponibilité
CP-ABE	Ciphertext-Policy ABE
CSP	Cloud Service Provider
DAC	Discretionary Access Control
DACC	Distributed Access Control in Cloud
DES	Data Encryption Standard
DoS	Denial of Service
DDoS	Distributed Denial of Service
ECC	Elliptic Curve Cryptography
EMR	Electronic Medical Records
EHR	Electronic Health Records
IBE	Identity Based-Encryption
IaaS	Infrastructure-as-a-Service
KGC	Key Generation Center

KDC	Key Distribution Centers
KP-ABE	Key Policy ABE
MAC	Mandatory Access Control
MITC	Man In The Cloud
MK	Master Key
NIST	Institute National of Standards and Technology
OS	Operating System
PaaS	Plateforme-as-a-Service
PBAC	Policy Based Access Control
PCA	Plan de Continuité d'Activité
PDP	Policy Decision Point
PEP	Policy rEnforcement Point
PHR	Personal Health Records
PIPE	Pseudonymization of Information for Privacy in Ehealth
PK	Public Key
RBAC	Role Based Access Control
RC4	Rivest Cipher 4
RPO	Recovery Point Objective
RTM	Registration and Ticket Manager
RTO	Recovery Time Objective
PKI	Public Key Infrastructure
SaaS	Software-as-a-Service
SET	Secure Electronic Transaction
SK	Secrete Key

SLA	Service Level Agreement
SM	Service Manager
SSH	Secure SheL Protocol
SSL	Secure Socket Layer
SSO	Single Singe on
SQL	Structured Query Language
TAAC	Temporal Attribute-Based Access Control
TOT	Ticket Of Ticket
TTP	Trusted Third Party
TLS	Transport Layer Security
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
IT	Information technology
VM	Virtual Machine
XOR	eXclusive OR

Introduction Générale

Les industries des technologies de l'information (Information Technology, IT) font évoluer la technologie de temps à autre vers une nouvelle arène. L'Internet est l'une des technologies les plus populaires de nos jours par l'élégance de l'IT. Aujourd'hui, c'est au bord de la révolution, où les ressources sont interconnectées à l'échelle mondiale. Ainsi, les ressources peuvent être facilement partagées et gérées de n'importe où et n'importe quand. Le cloud est l'élément principal de cette norme, qui fournit un vaste espace de stockage où les ressources sont disponibles de n'importe où et pour tous, sous forme de service plutôt que de produit. Aujourd'hui, plusieurs entreprise considèrent le cloud computing comme une force majeure à modifier de façon significative l'ensemble de la technologie d'information, la façon dont les centres de données sont construits, la façon dont les logiciels sont déployées, le traitement des mises à jour, etc..

Cependant, d'un point de vue sécurité de l'information, un certain nombre de questions se posent sur les différentes menaces que font face le Cloud, notamment la confidentialité des utilisateurs et l'intégrité des données stockées. Des questions logiques se posent : Suis-je connu au sein du cloud une fois que j'y accède ? Comment puis-je m'assurer que mes données sont bien sécurisées ? Que deviennent mes données si une personne malséante s'est introduite au cloud ? Comment puis-je être sûr que d'autres utilisateurs ne puissent pas accéder à mes données? Comment puis-je partager certaines de mes données avec d'autres personnes de confiances ?..... C'est dans ce contexte que s'inscrit notre problématique.

L'objectif de ce travail est de mettre en place des solutions aux problématiques de la confidentialité et l'intégrité des données stocké dans le cloud et les partager avec les utilisateurs autorisées sans dévoilé leur identité, et ce dans le cadre de la e-santé comme domaine d'application du cloud. Au fait, le domaine de la e-santé est utilisé dans le but d'améliorer les services médicaux, de rendre l'accès facile aux utilisateurs et de faciliter la communication et le partage des données médicales entres les utilisateurs.

D'une part, la confidentialité permet d'assurer la protection et la sécurité des informations personnelles des utilisateurs (patients et médecins). Pour la garantir, de nombreuses techniques ont été proposées dans la littérature. Certaines sont basées sur des méthodes et des algorithmes conventionnels, d'autres sont modernes et utilisées en combinaison pour atteindre

une sécurité à toute épreuve. L'authentification des utilisateurs est l'une des approches pour y parvenir à assurer la confidentialité. Toutefois, les utilisateurs authentifiés restent visible au sein du cloud, donc il est nécessaire de combiner ce mécanisme de sécurité avec l'anonymat afin de garantir non seulement la confidentialité mais aussi la confiance des utilisateurs.

D'autre part, une mauvaise utilisation des données par le cloud ou un accès non autorisé par des utilisateurs externes pourrait constituer une menace potentielle pour les données stockées (e.g. dossiers médicaux). Les personnes (patients) souhaitent que leurs données sensibles ou privées ne soient accessibles qu'aux personnes autorisées avec les identifiants qu'elles ont spécifiés (e.g certains médecins). Une solution consiste à employer une approche de contrôle d'accès basé sur une technique de chiffrement des données avant de stocker les données dans le cloud. Le chiffrement CP-ABE (Ciphertext-Policy Attribute-Based Encryption) offre une solution efficace pour garantir l'intégrité des données et fournir un contrôle d'accès des utilisateurs.

Ce présent mémoire est organisé comme ceci :

- Dans le chapitre 1, nous donnons un aperçu sur le cloud et la sécurité dans cloud
- Dans le chapitre 2, nous présentons les méthodes de chiffrement et les modèles de contrôle d'accès dans le cloud.
- Dans le chapitre 3, nous étudions les modèles d'anonymat et les méthodes d'authentification au sein du cloud.
- Dans le chapitre 4, nous expliquons la conception de notre solution.
- Dans le chapitre 5, nous décrivons l'implémentation de notre solution

En conclusion générale, nous synthétisons les principales parties du mémoire, en faisant ressortir les apports de notre travail ainsi que les perspectives prévues pour l'améliorer.

Chapitre I : Introduction au Cloud Computing

1. Introduction

L'Internet se développe d'une manière exponentielle depuis sa création. Actuellement, une nouvelle tendance a fait son apparition dans le monde de l'IT, il s'agit du Cloud Computing (Informatique en nuage). De nombreuses organisations transfèrent leurs services informatiques vers le cloud, ce qui rend leur traitement informatique disponible beaucoup plus pour les utilisateurs. En outre, le Cloud est une technologie attrayante et économique car il offre des options d'accessibilité et de fiabilité pour les utilisateurs. Cependant, il apporte aussi des menaces et des défis concernant la sécurité. Cette dernière est devenue une préoccupation majeure pour le cloud.

Nous allons présenter dans ce chapitre les concepts généraux sur le cloud afin de mieux le comprendre puis nous entamerons la sécurité du cloud où nous détaillerons les attaques et les mécanismes de sécurités liés au cloud.

2. Informatique en nuage (Cloud Computing)

Nous allons apercevoir dans cette partie les généralités du cloud proposées par le NIST (National Institute of Standards and Technology) du gouvernement américain ; il est important de les connaitre avant d'entamer la sécurité du cloud.

2.1 Définition

Le Cloud Computing est un modèle permettant d'établir un accès à la demande en réseau vers un bassin partagé de ressources informatiques configurables. Ces ressources sont par exemple des réseaux, des serveurs, de l'espace de stockage, des applications et des services. Elles peuvent être approvisionnées rapidement avec un effort de gestion et une interaction avec le fournisseur de services minimes**.** Le modèle Cloud met en avant la disponibilité, et se compose de cinq caractéristiques essentielles, trois modèles de livraisons et quatre modèles de déploiement [1]. Toutes ces notions sont développées dans les sections suivantes.

2.2 Caractéristiques du cloud

Les cinq caractéristiques suivantes, telles que définies par le NIST, sont considérées comme inhérentes à des services de cloud [1] :

a. Libre-service à la demande

Le libre-service à la demande permet à l'utilisateur d'être en mesure de provisionner, mais également de libérer des ressources distantes en temps réel en fonction des besoins, et sans nécessiter d'intervention humaine.

b. Accès réseau large bande

Les fonctionnalités sont disponibles sur le réseau et accessible via des mécanismes standards qui favorisent l'utilisation de plates-formes client hétérogènes (par exemple, téléphones mobiles, tablettes, ordinateurs portables et stations de travail).

c. Réservoir de ressources (non localisées)

Des ressources telles que la bande passante réseau, machines virtuelles, mémoire, puissance de traitement, capacité de stockage, etc... Sont mises en commun pour desservir plusieurs clients à l'aide d'un modèle multi-locataire. Autrement dit, les ressources virtuelles et physiques sont affectées dynamiquement et réaffectés en fonction des besoins et des exigences clients.

d. Redimensionnement rapide (élasticité)

En fonction de la demande, les ressources et les capacités peuvent être rapidement et automatiquement déployées et mises à l'échelle à n'importe quelle quantité et à tout moment.

e. Service mesurée

L'utilisation des ressources est automatiquement surveillée, contrôlée et rapportée, offrant une transparence à la fois au fournisseur et au consommateur du service utilisé.

2.3 Modèles de livraisons

Il existe trois modèles de livraison du Cloud Computing (figure 1) :

- **Logiciel en tant que service** (SaaS, Software as a Service)
- **Plate-forme en tant que service** (PaaS, platform as a Service)
- **Infrastructure en tant que service** (IaaS, infrastructure as a Service)

Ces trois modèles de service doivent être déployés sur des infrastructures qui possèdent les cinq caractéristiques essentielles citées dans la section précédente pour être considérées comme du Cloud Computing.

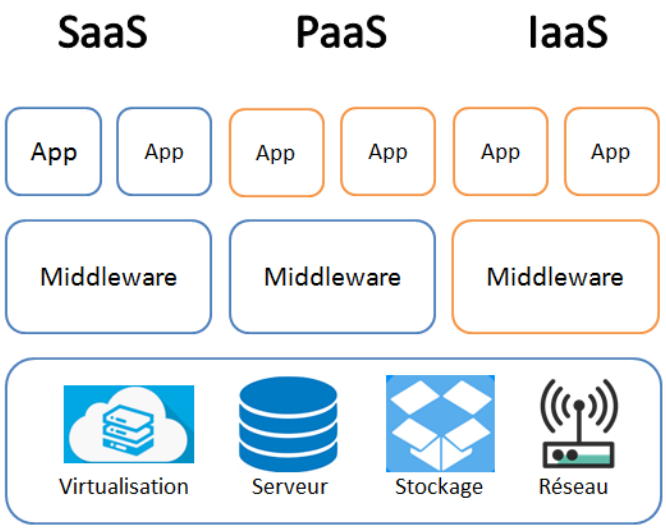

Figure 1 : Modèles de livraisons [1]

a. Logiciel en tant que service (SaaS)

Ce modèle de service est caractérisé par l'utilisation d'une application partagée qui fonctionne sur une infrastructure Cloud. L'utilisateur accède à l'application par le réseau au travers de divers types de terminaux (souvent via un navigateur web). L'administrateur de l'application ne gère pas et ne contrôle pas l'infrastructure sous-jacente (réseaux, serveurs, applications, stockage). Il ne contrôle pas les fonctions de l'application à l'exception d'un paramétrage de quelques fonctions utilisateurs limitées.

b. Plate-forme en tant que service (PaaS)

L'utilisateur a la possibilité de créer et de déployer sur une infrastructure Cloud ses propres applications en utilisant les langages et les outils du fournisseur. .L'utilisateur ne gère pas ou ne contrôle pas l'infrastructure Cloud sous-jacente (réseaux, serveurs, stockage) mais l'utilisateur contrôle l'application déployée et sa configuration.

c. Infrastructure en tant que service (IaaS)

L'utilisateur loue des moyens de calcul et de stockage, des capacités réseau et d'autres ressources indispensables (partage de charge, pare-feu, cache). L'utilisateur a la possibilité de déployer n'importe quel type de logiciel incluant les systèmes d'exploitation. L'utilisateur ne gère pas ou ne contrôle pas l'infrastructure Cloud sous-jacente mais il a le contrôle sur les systèmes d'exploitation, le stockage et les applications. Il peut aussi choisir

les caractéristiques principales des équipements réseau comme le partage de charge, les pare-feu, etc.

2.4Modèles de déploiement

Nous distinguons quatre formes de Cloud [1] :

a. Cloud privé

L'infrastructure Cloud est utilisée par une seule organisation. Elle peut être gérée par l'organisation ou par une tierce partie. L'infrastructure peut être placée dans les locaux de l'organisation ou à l'extérieur.

b. Cloud communautaire

L'infrastructure Cloud est partagée par plusieurs organisations pour les besoins d'une communauté qui souhaite mettre en commun des moyens (sécurité, conformité, etc..). Elle peut être gérée par les organisations ou par une tierce partie et peut être placée dans les locaux ou à l'extérieur.

c. Cloud publique

L'infrastructure Cloud est ouverte au public ou à de grands groupes industriels. Cette infrastructure est possédée par une organisation qui vend des services Cloud. C'est le cas le plus courant.

d. Cloud hybride

L'infrastructure Cloud est composée d'un ou plusieurs modèles (privée, communautaire ou publique) qui restent des entités séparées. Ces infrastructures sont liées entre elles par la même technologie qui autorise la portabilité des applications et des données.

Le tableau suivant montre les avantages et inconvénients de chaque modèle :

Modèle déploiement	Avantages	Inconvénients
Cloud Privé [2]	1. Contrôle complet sur les données et l'infrastructure 2. Performance améliorée 3. Sécurité et confidentialité améliorées	1. Coût d'installation élevé 2. Mise à l'échelle de la plate-forme
Cloud communautaire [3]	1. Multi-locataire sécurisé, privé et réponds aux exigences des organisations. 2. Solutions flexibles aux différents besoins du marché	1. Problèmes de confiance 2. Problèmes de sécurité entre les locataires et les intrus potentiels susceptibles d'endommager le système.
Cloud public [2]	1. Rapide et peu coûteux à déployer 2. S'adapte immédiatement à l'augmentation des besoins	1. Problème de sécurité : utilisation des services 2. Le coût est proportionnel à l'utilisation : problème de la surconsommation
Cloud hybride	On combine le meilleur des deux modèles (public et privé) prenant en considération les besoins de l'entreprise	

Tableau 1: Avantages et inconvénients des types du cloud

Il n'y a pas de bonne ou de mauvaise solution : choisir entre les différents types de cloud revient à s'interroger sur les besoins de l'entreprise. Il est essentiel de comprendre que les modèles de service, les modèles de déploiement et les cinq caractéristiques du cloud computing tels que décrits par le NIST ne fonctionnent pas de manière indépendante mais ne sont pas nécessairement liés entre eux et reliés les uns aux autres. La figure ci-dessous affiche ces interrelations et les connexions nécessaires des caractéristiques du cloud computing et les modèles. Elle démontre qu'une stratégie basée sur le cloud peut prendre des configurations différentes en fonction des besoins des institutions.

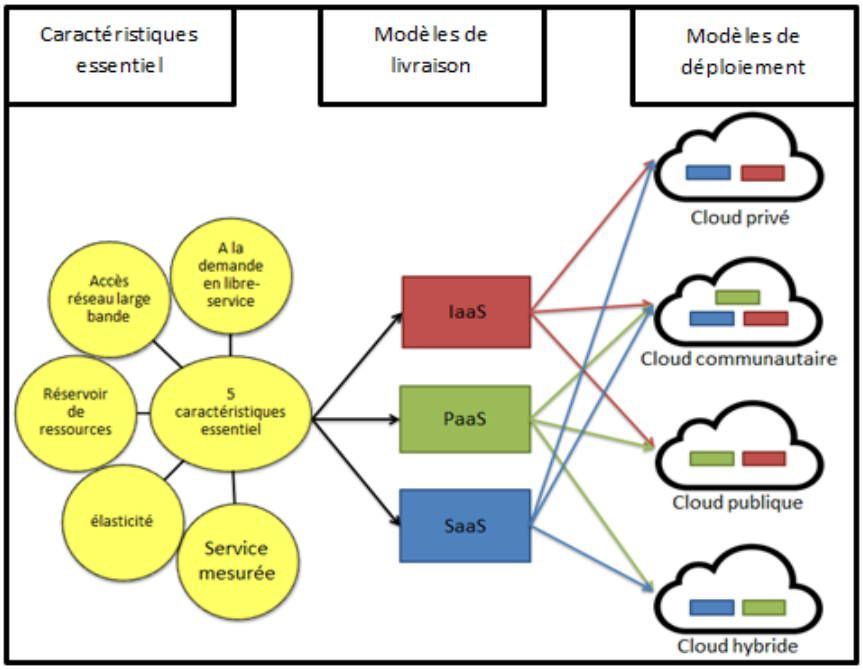

Figure 2 : Interrelation du cloud computing [4]

2.5Composantes du cloud computing

Le Cloud computing ne fait pas référence à une technologie spécifique, mais a une combinaison de technologies préexistantes et des protocoles qui ont rendu ce paradigme possible. Dans cette section, nous allons décrire les composantes technologiques et non technologiques les plus pertinentes.

2.5.1 Composantes technologiques

On distingue quatre composantes technologiques du cloud :

a. Les centres de données et fermes de serveurs

Un centre de données (data center) est un site physique sur lequel se regroupe des équipements réseaux constituants un système d'information de l'entreprise (mainframes, serveurs, baies de stockage, équipements réseaux, etc.). Il peut être interne ou externe a l'entreprise [5].

Nous considérons trois types de serveurs dans l'architecture du cloud [6] :

i. **Cloud honnête** (Trusted Cloud) **:** fait référence à un serveur entièrement sécurisé. Tout en appliquant le protocole spécifié, le serveur est supposé ne jamais obtenir aucune information sur les données stockées.

ii. **Cloud honnête mais curieux** (Semi-trusted Cloud) : signifie que le serveur fait exactement ce que dit le protocole, mais il peut également vouloir apprendre des informations supplémentaires à partir du protocole.

iii. **Cloud malhonnête** (Untrusted Cloud) : signifie que le serveur n'est pas approuvé et qu'il ne dispose pas de techniques de contrôle d'accès ni de protection de la confidentialité ; il ne conviendrait pas de stocker les données en texte brut, par conséquent des méthodes de cryptage/chiffrement appropriées doivent être utilisées.

b. La virtualisation

La virtualisation est la principale technologie dans le cloud, c'est une manière pour partitionner une ressource physique en plusieurs ressources virtuelles, par exemple : un serveur, un espace de stockage ou un réseau lors de la création des machines virtuelles. Elle permet d'intégrer différents serveurs de façons plus flexibles pour faciliter l'utilisation [5].

c. Interfaces de programmation d'applications

Les APIs (Application Programmin Interface) offrent des fonctionnalités aux clients, comme l'auto-approvisionnement et le contrôle des services et des ressources. Elles permettent également la communication entre les applications étrangères et le service sur le cloud. Le type d'API dépend du modèle de déploiement [5].

d. Le chiffrement

Le chiffrement (ou cryptage) est le processus de codage des messages ou des informations d'une manière que seules les parties autorisées puissent lire cette information, empêchant ainsi des parties indésirables à les intercepter. Cela se fait habituellement à l'aide d'une clé de chiffrement qui spécifie la façon dont le message est codé. Ensuite, la partie autorisée utilise une clé de déchiffrement secrète pour déchiffrer le message et le lire. Le chiffrement est la technique couramment utilisée pour protéger la confidentialité des messages [5].

2.5.2 Composantes non technologiques

Le cloud comporte d'autres composantes qui ne sont pas technologiques :

a. Accords de niveau de service

Le contrat mutuel entre les fournisseurs et les utilisateurs, appelé généralement SLA (Service Level Agreement), offre des garanties sur la qualité du service en définissant ce qui est à attendre du service offert, et définit les schémas de compensation dans le cas où le fournisseur ne respecte pas ce contrat. Quelques exemples de ces accords pourraient être de

99% de la disponibilité, la résolution des incidents dans une période de temps spécifiée, la sécurité des données, etc... [5].

b. Les politiques de confidentialité

Une politique est une déclaration d'intention, mise en œuvre grâce à des protocoles ou des procédures, afin de guider les décisions et d'atteindre un but désiré.

Lorsqu'elles sont appliquées sous le thème de la vie privée, ces politiques représentent des documents juridiques qui expliquent et décrivent la façon dont une partie recueille, utilise, divulgue et gère les données du client (par exemple le nom, l'adresse, les antécédents médicaux, etc..).

Elles informent également le client à propos des informations recueillies et si elles sont gardées confidentielles, partagées avec des partenaires ou vendues à d'autres entreprises. Ces aspects sont généralement recueillis dans un document écrit qui énonce les règles, fournit les principes qui guident les actions, définit les rôles et les responsabilités, reflète les valeurs et les croyances et indique un protocole d'actions. Dans ce sens, il existe des différences dans la façon dont ces lois sur la protection sont mises en œuvre et appliquées dans différents pays [5].

2.6 Utilisation du cloud dans le domaine de la santé

Le Cloud est un terme qui est utilisé presque dans tous les champs liés aux tendances informatiques [7]. Il est utilisé dans les domaines du divertissement, de la santé, des opérations militaires, des affaires et de la finance, etc… Dans notre travail, nous nous intéressons au domaine de la santé.

Dans un hôpital ou dans un cabinet médical, les dossiers des patients et leurs antécédents médicaux sont souvent conservés dans un système d'information. Cela est pratique avec l'augmentation du nombre de patients.

Le cloud peut aider à faciliter l'accès et la distribution d'informations entre les divers professionnels de la santé pouvant entrer en contact avec chaque patient. Un système basé sur le cloud améliorera le partage d'informations en permettant à tous d'être hébergé au même endroit, permettant ainsi à un médecin de saisir les résultats des tests dans le laboratoire, mettant à jour instantanément le dossier d'un patient dans une aile complètement séparée. De même, les bâtiments et les installations de traitement tels que les laboratoires, les médecins

effectuant des visites d'urgence à domicile et les ambulances peuvent avoir et mettre à jour des informations à distance, au lieu d'attendre d'avoir accès à un ordinateur de l'hôpital.

Dans le secteur de la santé, on peut trouver plusieurs acronymes, ce qui rend parfois difficile de garder les termes simples, en particulier ceux liés au dossier de santé électronique [8] :

- EMR (electronic medical records)
- EHR (electronic health records)
- PHR (personal health records)

La transition vers les dossiers médicaux électroniques (EHR) a permis aux praticiens et à leurs clients de connaître plusieurs acronymes qui sont désormais inscrits de manière permanente dans le lexique des soins de santé. Bien que «EMR», «EHR» et «PHR» soient souvent utilisés de manière interchangeable, ils ont des significations très différentes :

a. EMR (Electronic Medical Records)

Un dossier médical électronique (EMR) est la forme numérique des cartes papiers que les établissements de santé utilisaient auparavant pour suivre les traitements, les médicaments, les modifications de l'état, etc. Un dossier médical électronique représente une mise à niveau par rapport à la carte papier traditionnelle, car il est plus facile pour les praticiens de suivre les données au fil du temps et de surveiller la santé du client de manière plus fiable, ce qui conduit à de meilleurs soins de longue durée.

b. EHR (Electronic Health Records)

Le « EHR » fournit un enregistrement numérique des informations sur la santé. Cependant, le « EHR » comprend plus de données que le « EMR ». Il contient les commentaires de tous les praticiens impliqués dans les soins du client (patient). Ainsi, le « EHR » offre une vision plus complète de la santé du client et de son historique de traitement. Ces informations sont entrées dans une base de données commune qui peut être partagée entre des utilisateurs autorisés dans plusieurs organisations de soins de santé.

c. PHR (Personal Health Records)

« PHR » fournit un enregistrement électronique des informations relatives à la santé du client, mais la différence est que, contrairement aux « EMR » et « EHR » sont gérés par les praticiens, le « PHR » est géré par le client. Le « PHR » permet à chaque client de visualiser et de contrôler les données dans un environnement sécurisé et de les partager avec d'autres

parties si nécessaire. Un « PHR » peut contenir des informations provenant de sources multiples telles que des médecins, des dispositifs de surveillance à domicile et d'autres données fournies par le client.

Dans notre travail, nous allons utiliser le système PHR. Dans un tel système, il existe plusieurs menaces qui touchent la confidentialité et la préservation de la vie privé des patients : En effet, les dossiers médicaux des patients contiennent des informations sensibles telles que des informations détaillées sur la maladie d'un patient, l'usage de drogues etc… La divulgation inappropriée d'un enregistrement ou l'ajout/modification/suppression d'un enregistrement peut changer la vie du patient et il n'existe aucun moyen de réparer ces dommages financièrement ou techniquement. Par conséquent, il faut sécuriser le système PHR dans le cloud. Cela fait l'objet de ce mémoire.

3. Sécurité dans le cloud

La sécurité est l'un des principaux problèmes qui freinent la croissance du cloud. De plus des risques et des menaces inhérents aux technologies de l'information traditionnelles, le Cloud présente une organisation avec son propre ensemble de problèmes de sécurité.

3.1 Problèmes généraux

Les trois principes fondamentaux de la sécurité des informations - Confidentialité, Intégrité et Disponibilité (CID) - définissent la posture de sécurité d'une organisation. Tous les contrôles et les sauvegardes de la sécurité des informations, ainsi que toutes les menaces, les vulnérabilités et les processus de sécurité sont soumis à la norme CID [9]:

a. Confidentialité

Est la prévention de la divulgation non autorisée, intentionnelle ou non intentionnelle, de contenus.

b. Intégrité

Est la garantie que le message envoyé est le message reçu et que ce dernier n'est pas modifié intentionnellement ou non.

c. Disponibilité

Ce concept fait référence aux éléments qui créent la fiabilité et la stabilité dans les réseaux et les systèmes. Il garantit que la connectivité est accessible en cas de besoin, permettant aux utilisateurs autorisés d'accéder au réseau ou aux systèmes.

3.2 Menaces du cloud

Les menaces contre le cloud ont pour principaux objectifs d'obtenir un accès aux données des utilisateurs et d'empêcher l'accès aux services de cloud. Ces deux menaces peuvent causer de graves dommages aux utilisateurs en brisant la confiance en la sécurité du cloud.

Dans le cloud, les pirates informatiques s'immiscent généralement dans les communications entre les utilisateurs du cloud et les services ou applications en :

a. exploitant les vulnérabilités dans le cloud.
b. volant les identifiants des utilisateurs quelque part en dehors du cloud.
c. utilisant un accès antérieur légitime au cloud après avoir déchiffré les mots de passe d'un utilisateur.
d. agissant en tant qu'initié malveillant.

Ainsi, il existe de nombreuses menaces qui attaquent les services d'informatique du cloud [10] :

a. Attaques par déni de service

Les attaques par déni de service sont conçues pour surcharger un système et rendre les services inaccessibles à ses utilisateurs. Ces attaques sont particulièrement dangereuses pour les systèmes du cloud, car de nombreux utilisateurs peuvent en pâtir, même après l'inondation d'un seul serveur en nuage. En cas de charge de travail élevée, les systèmes en nuage commencent à fournir plus de puissance de calcul en impliquant davantage de machines virtuelles et d'instances de service. En essayant d'empêcher une cyberattaque, le système en nuage le rend encore plus dévastateur. Enfin, le système en nuage ralentit et les utilisateurs légitimes perdent toute disponibilité pour accéder à leurs services en nuage. Dans l'environnement en nuage, les attaques DDoS[1] peuvent être encore plus dangereuses si les pirates informatiques utilisent davantage de machines zombies pour attaquer un grand nombre de systèmes.

b. Attaques par canal auxiliaire

Une attaque par canal auxiliaire est organisée par les pirates informatiques lorsqu'ils placent une machine virtuelle illicite sur le même hôte que la machine virtuelle cible. Lors

[1] Déni de service distribué est un type d'attaque très évolué visant à faire planter ou à rendre muette une machine en la submergeant de trafic inutile

d'une attaque par canal auxiliaire, les pirates informatiques ciblent les implémentations des algorithmes cryptographiques. Cependant, ce type de menace peut être évité avec une conception de système sécurisée.

c. Attaque l'homme dans le nuage

Au cours de ce type d'attaque, les pirates informatiques interceptent et reconfigurent les services de cloud en exploitant les vulnérabilités du système de jeton de synchronisation afin que lors de la prochaine synchronisation avec le cloud, le jeton de synchronisation soit remplacé par un nouveau qui permette l'accès aux attaquants. Les utilisateurs peuvent ne jamais savoir que leurs comptes ont été piratés, car un attaquant peut remettre les jetons de synchronisation d'origine à tout moment. De plus, il existe un risque que les comptes compromis ne soient jamais récupérés.

d. Attaques par injection de logiciels malveillants

Les attaques par injection de logiciels malveillants servent à prendre le contrôle des informations d'un utilisateur dans le cloud. À cette fin, les pirates informatiques ajoutent un module d'implémentation de service infecté à une solution SaaS ou PaaS ou une instance de machine virtuelle à une solution IaaS. Si le système cloud est correctement trompé, il redirigera les requêtes de l'utilisateur du cloud vers le module ou l'instance du pirate informatique, en initiant l'exécution de code malveillant. L'attaquant peut alors commencer son activité malveillante, telle que la manipulation ou le vol de données ou l'espionnage.

Les formes les plus courantes d'attaques par injection de logiciels malveillants sont les attaques de script intersite (entre sites) et les attaques d'injection SQL. Lors d'une attaque de script intersite, les pirates informatiques ajoutent des scripts malveillants (JavaScript, flash, etc.) à une page Web vulnérable. Dans le cas d'une injection SQL, les attaquants ciblent des serveurs SQL avec des applications de base de données vulnérables.

e. Attaques d'initiés (insider attack)

Cette attaque est initiée par un utilisateur légitime qui enfreint délibérément la politique de sécurité. Dans un environnement cloud, un attaquant peut être un administrateur de fournisseur ou un employé d'une société cliente disposant de privilèges étendus. Pour

empêcher toute activité malveillante de ce type, les développeurs de cloud devraient concevoir des architectures sécurisées avec différents niveaux d'accès aux services de cloud.

3.3. Mécanismes de sécurité

La sécurité dans le Cloud est classifiée selon trois mécanismes de sécurité [11] : la sécurité physique, la sécurité logique et la sécurité des données.

3.3.1 Sécurité physique

Le Cloud Computing, par nature, est associé à une sorte de « dématérialisation » de l'hébergement (le nuage). En effet, le lieu d'hébergement du Cloud est généralement multiple, et réparti sur plusieurs centres de données (data centers).

Dans le cas du Cloud public, le client ne connaît donc pas avec précision le ou les lieux d'hébergement du Cloud. Cette caractéristique, gage de disponibilité du Cloud, entraîne un changement important pour le client dans le mode de sélection de l'hébergeur.

Une visite du centre de données ne suffit plus pour évaluer le niveau d'hébergement garanti par un fournisseur de Cloud. Ce dernier doit être en mesure d'apporter des garanties sur les conditions d'hébergement associées à son offre. Un certain nombre de certifications et/ou de classifications existent à ce sujet, sont reconnues et adoptées par l'ensemble des hébergeurs.

a. Contrôle et traçabilité des accès

L'accès physique d'une seule personne mal intentionnée qui possède une excellente connaissance de l'implémentation physique du Cloud Computing peut suffire à mettre hors service le Cloud, provoquant une rupture dans la continuité du service et empêchant tout accès externe au Cloud. Les conséquences d'une telle intrusion peuvent être désastreuses :

- Isolement complet ou partiel du service dans le cas de coupure des liaisons d'accès.
- Perte des données en production et des données sauvegardées, sans aucune possibilité de récupération si celles-ci sont détruites ou détériorées.
- Risque d'incendie élevé ou d'inondation, etc.

C'est pourquoi, il est nécessaire de contrôler l'accès aux centres de données du cloud et garder trace aux accès permis.

En effet, le contrôle des accès doit être maitrisé, que l'on soit dans un contexte de Cloud privé, public ou privé externalisé (hybride). Dans ces deux derniers cas, c'est au client final de s'assurer que les bonnes pratiques sont mises en œuvre chez son prestataire de

service/opérateur de Cloud. Concrètement, les va-et-vient du personnel interne et des prestataires externes (informatique et télécoms, société de maintenance, de nettoyage, etc.) sont nombreux dans une salle informatique ou dans les locaux techniques. Ce flux représente une source potentielle de dysfonctionnements, volontaires ou non. Il convient de bien délimiter les zones les plus sensibles et de mettre en place des garde-fous suffisamment efficaces pour retrouver, le cas échéant, l'origine d'un incident (figure 3). L'accès aux zones sensibles (serveurs, réseau, etc.) sera interdit et le passage dans les zones intermédiaires sera limité. Le personnel autorisé devra être informé du caractère sensible des zones dans lesquelles il est amené à intervenir.

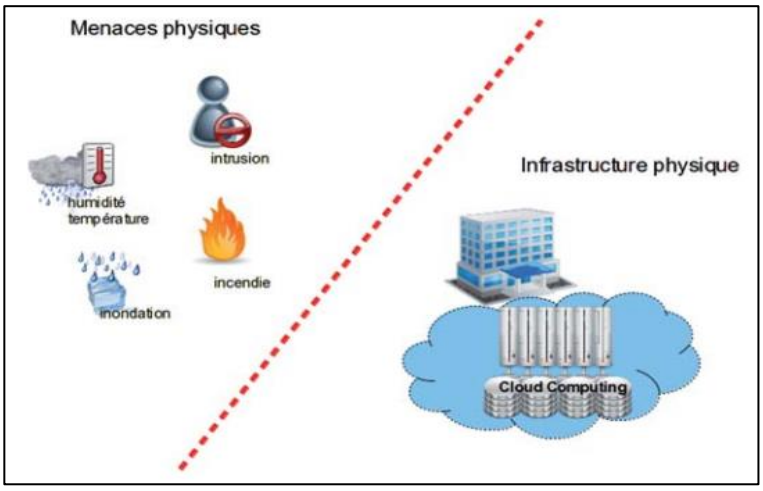

Figure 3 : sécurisation de l'environnement [11]

b. Redondance matérielle

L'architecture du Cloud Computing doit garantir un accès au service en très haute disponibilité avec des performances optimales. La seule défaillance d'un équipement matériel peut engendrer une dégradation ou une coupure du service voire une perte de données. Pour limiter les risques d'arrêt de service liés à la défaillance d'un équipement, il est nécessaire de le redonder. Une réplication des configurations entre les équipements peut faciliter la bonne prise en charge de la redondance et ainsi augmenter la haute disponibilité du service. La mise en œuvre d'une redondance différentielle avec une sélection d'équipements de natures différentes (ex : différents constructeurs, composants d'origines différentes, etc.) permet de se protéger d'un problème survenu à un équipement donné.

De plus, une redondance des moyens de connexion, par la multiplication des liaisons, des opérateurs, et des chemins d'accès permet une accessibilité accrue au service en augmentant la tolérance aux pannes.

c. Résilience

Une catastrophe d'origine humaine ou naturelle peut avoir des impacts radicaux sur le fonctionnement du Cloud Computing et amplifier une panne totale ou partielle du service. La perte totale de l'infrastructure du Cloud Computing pourrait entrainer une interruption de service d'une durée indéterminée et une perte de données irrémédiable sans possibilité de remise en service de l'infrastructure. Une architecture de secours doit exister, sur un site géographiquement éloigné, avec des équipements redondants et permettant de réaliser un PCA (Plan de Continuité d'Activité) sans interruption de service comme montré dans les figures suivants :

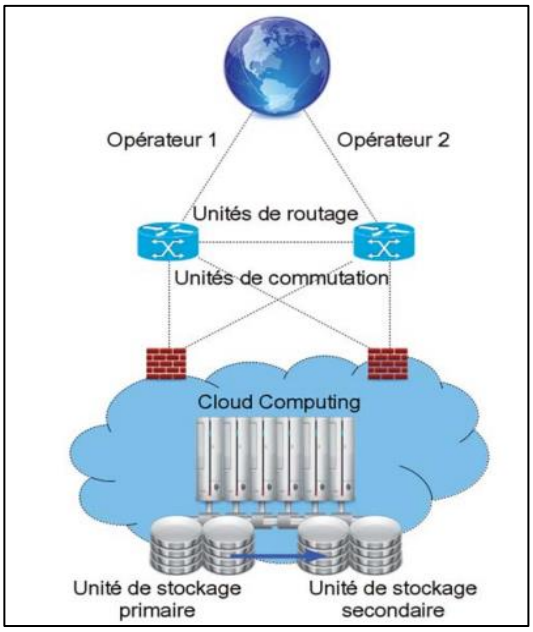

Figure 4 : architecture mono-data center [11]

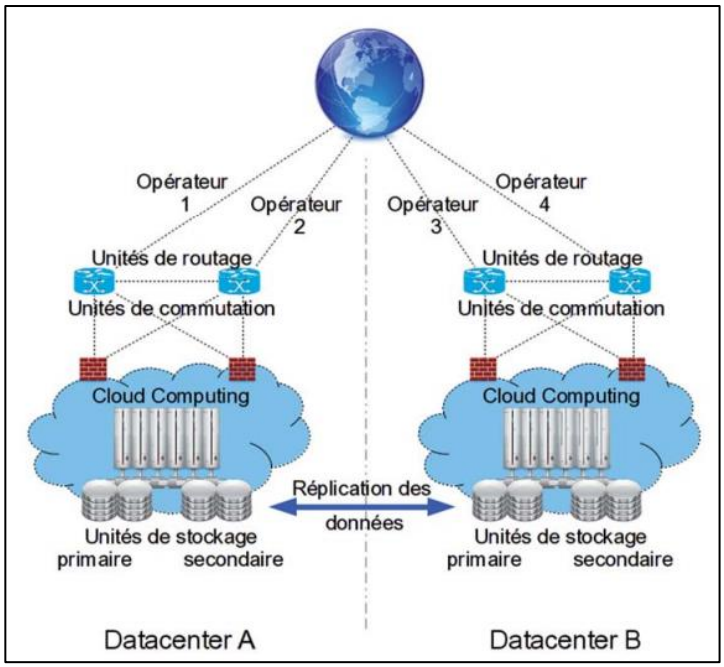

Figure 5 : architecture multi-data center [11]

3.3.2 Sécurité logique

Ce mécanisme et ses éléments permettent la protection de l'accès aux systèmes informatiques.

a. Sécurité des serveurs virtuels

Le Cloud Computing s'appuie fortement sur les technologies d'abstraction de services. Dans le cadre d'un modèle IaaS, c'est la virtualisation de serveurs qui fournit cette abstraction ; L'élément de base, visible ou non, étant une machine virtuelle (Virtual Machine, VM) sur un hyperviseur. L'hyperviseur héberge également une VM particulière appelé de manière général « partition de gestion ». Elle permet d'administrer l'hyperviseur, de gérer le matériel et les ressources virtualisées. Généralement, on distingue les bonnes pratiques de sécurité liées à la virtualisation en deux familles.

En premier lieu, il s'agit de sécuriser les systèmes en assurant une gestion des mises à jour de sécurité. La mise à jour de l'hyperviseur et de la partition de gestion, a priori à la charge de l'hébergeur, conduit dans la plupart des cas à un redémarrage du serveur. Pour éviter que les VM soient indisponibles durant l'opération, un mécanisme de déplacement automatique des VM vers un autre serveur est possible. La sécurisation des systèmes suppose également la réduction des surfaces d'attaque, en fixant au strict minimum les services de la « partition de gestion ». On protège également les fichiers des disques virtuels par du contrôle d'accès, de

l'audit, voire du chiffrement. Idéalement, on agit conformément aux recommandations des fournisseurs de l'hyperviseur (configuration des disques virtuels, installation de composants d'intégration dans les VM, etc.) et des OS, en mettant en place un contrôle de conformité automatisé.

La seconde famille de bonnes pratiques concerne la notion d'isolation : isolation des flux réseaux, isolation des VM par niveau de sécurité, délégation de l'administration. L'infrastructure de type cloud doit automatiser la plupart des contraintes évoquées précédemment, en plus des processus liées à l'administration, la supervision et l'allocation automatique de ressources.

b. Colocation sécurisée

La colocation sécurisée consiste en l'hébergement sur le Cloud des applications et données de multiples clients (sociétés, organisations, entités métier…) au sein d'une seule et unique infrastructure physique, mutualisée, tout en respectant la sécurité, notamment au sens de la confidentialité. Les données ainsi que leur traitement fait par des société-clientes du Cloud doivent être isolés et protégés des autres environnements hébergés sur l'infrastructure partagée. C'est souvent une obligation légale, exemple : stockage des données personnelles, médicales…De plus, appliquer rigoureusement les bases de la sécurité d'un système d'information mutualisé tel que la planification rigoureuse des droits d'accès, des privilèges administrateurs, sécurisation des mots de passe, etc.… Certaines techniques ou architectures permettent de satisfaire confidentialité des données comme le chiffrement.

c. Sécurité de l'interface de l'administration

L'accessibilité aux interfaces d'administration via une vulnérabilité applicative expose aux risques d'une coupure partielle ou totale du service ou à une perte irrémédiable des données. Par cet accès, l'introduction de virus ou de vers peut également détériorer les applications, faciliter la corruption des données ou nuire à l'image de marque du service. Les solutions préventives consistent en la mise en place d'équipements de filtrage (pare-feu, proxy, sondes IPS/IDS...) et de solutions antivirus afin de contrôler la légitimité des requêtes entrantes et ainsi garantir l'intégrité des données hébergées. La planification de tests de vulnérabilités et d'intrusions doit être régulière et fréquente. Le développement des applications doit être soumis à des audits de codes réguliers, et à l'implémentation de règles et contrôles des données.

Par ailleurs, l'accès aux interfaces d'administration du Cloud Computing pourrait permettre à une personne mal intentionnée de provoquer une coupure de service ou de corrompre les données hébergées. Si l'accès authentifié n'est pas clairement identifié, alors il sera impossible de pouvoir tracer la connexion et la modification des données ou du service qui en résulte. L'authentification doit apporter une preuve de l'identité si on veut pouvoir enquêter sur d'éventuels accès suspects. Une vulnérabilité provenant d'erreur, de faille « humaine » ou «d'hameçonnage» dans le processus d'authentification peut aussi être exploitée. Celle-ci pourrait donner des accès de type «administrateur » à l'architecture Cloud Computing et entraîner une corruption de la plate-forme. Les bonnes pratiques en la matière sont :

• La mise en place de mécanismes d'authentification forte (reposant sur deux facteurs ou plus) : identifiant, mot de passe, accès par jeton, certificat électronique, contrôle biométrique ...Elle est utilisé pour renforces le processus de contrôle d'accès.

• L'identification de l'authentification afin de disposer d'une traçabilité des accès

• La journalisation des authentifications réussies ou échouées

• La stricte application d'une politique de sécurité : changement des mots de passe tous les mois, politique de mots de passe complexes, formation du personnel...

3.3.3 Sécurité des données

Plus l'infrastructure est confiée au fournisseur Cloud, plus sa responsabilité vers les la sécurité des données est importante. Le fournisseur est en charge de la sauvegarde des données, d'un niveau bien défini de la disponibilité du service et de la confidentialité des données.

a. Responsabilité juridique de la sécurité et de la confidentialité des données dans le cloud

Le Client est juridiquement responsable de ses données et de leur utilisation, notamment de tout ce qui concerne leur conformité aux obligations juridiques. Le prestataire est soumis à des obligations techniques et organisationnelles. Il s'engage à préserver l'intégrité et la confidentialité des données, notamment en empêchant tout accès ou utilisation fraudeuse et en prévenant toutes pertes, altérations et destructions. Sa responsabilité juridique peut être engagée dans le cas où il aurait transféré les données de son client en dehors de sa zone géographique sans l'en prévenir et sans s'assurer que les déclarations nécessaires ont été faites.

b. Protection et récupération des données

Il existe deux métriques qui permettent de mesurer l'efficacité d'un processus de protection des données. Le premier est le « Recovery Time Objective » ou RTO qui mesure le temps acceptable ou toléré de rétablissement du service lors d'une panne. Le second étant le « Recovery Point Objective » ou RPO qui mesure la quantité de données que l'on s'accorde ou tolère à perdre due à une panne ou au processus de restauration.

c. Chiffrement lié à la donnée

Les défis de la cryptographie dans le Cloud sont complexes, notamment lorsqu'il s'agit de protéger les données hébergées contre des accès non-autorisés de la part de l'hébergeur. Nous allons étudier ce point (cryptographie dans le cloud) dans le chapitre 2.

d. Intégrité des données

Pour un Cloud plus sûr il est important aussi de réfléchir au contrôle d'accès. C'est une politique qui permet de protéger les accès aux ressources du système, elle peut soit les autoriser ou les interdire. Le contrôle d'accès renforce particulièrement l'intégrité des données.

Dans ce mémoire, nous allons nous concentrer sur le chiffrement, l'authentification et le contrôle d'accès pour sécuriser le Cloud. Ces mécanismes seront détaillés dans les prochains chapitres.

4. Conclusion

Le Cloud étant une technologie à croissance rapide qui offre une vaste gamme d'avantages aux utilisateurs surtout dans les systèmes de santé. Cependant, la sécurité, la confidentialité des données personnelles et la confiance restent les principales préoccupations qui empêchent l'adoption massive du Cloud.

C'est pour cela, nous avons besoin de plusieurs mécanismes de sécurités à différents niveaux (physique, logique et données). Dans ce présent mémoire, nous nous concentrons sur le chiffrement (la sécurité des données), sur l'authentification et le contrôle d'accès (la sécurité logique) dans le cloud, appliqué dans le système de santé PHR. Ces mécanismes de sécurité seront détaillés dans les prochains chapitres.

Chapitre II : Chiffrement et Contrôle d'Accès dans le Cloud

1. Introduction

Dans un système Cloud, le stockage et le traitement des données sont effectués par des organisations ou avec l'aide de fournisseurs tiers. Le fournisseur de services doit s'assurer que les données et les applications stockées dans le cloud sont protégées, ainsi que l'infrastructure dans un environnement sécurisé. De nombreux problèmes compromettent la sécurité des données dans le processus d'accès et de stockage des données, en particulier dans le cas du stockage de données avec l'aide de fournisseurs tiers qui peuvent être eux-mêmes un attaquant malveillant.

Pour la sécurité dans le cloud, un mécanisme d'autorisation est nécessaire pour sécuriser les données et les ressources en raison de son utilisation. L'absence de ce dernier, crée de nombreux problèmes dans l'environnement Cloud, notamment la gestion de la confiance, la sécurité des données, la confidentialité, la transparence et les fuites de données

Dans ce chapitre, nous allons commencer d'abord par présenter des notions sur le chiffrement, puis une étude sur le contrôle d'accès ainsi que ses modèles existants dans le Cloud. Par la suite, nous verrons une combinaison de ces deux aspects de sécurité, il s'agit du chiffrement lié aux modèles de contrôle d'accès.

2. Chiffrement

Dans cette section, nous allons d'abord décrire les notions de base du chiffrement, puis les algorithmes de chiffrement que nous utiliserons dans notre mémoire.

2.1 Concepts généraux

Le chiffrement est un procédé de cryptographie[2] grâce auquel on souhaite rendre la compréhension d'un document impossible à toute personne qui n'a pas la clé de déchiffrement. On distingue deux types de cryptographie [12] :

- Cryptographie symétrique
- Cryptographie asymétrique

2.1.1 Cryptographie symétrique

Dans un système de chiffrement symétrique ou chiffrement à clé secrète, un expéditeur et un destinataire partagent une même clé secrète. Cette clé est utilisée à la fois pour le

[2] La cryptographie est une science s'attachant à protéger des messages (assurant confidentialité, authenticité et intégrité) en s'aidant souvent de secrets ou clés.

chiffrement et pour le déchiffrement et doit rester secrète de toute autres personnes. Ce fonctionnement est présenté dans la figure suivante :

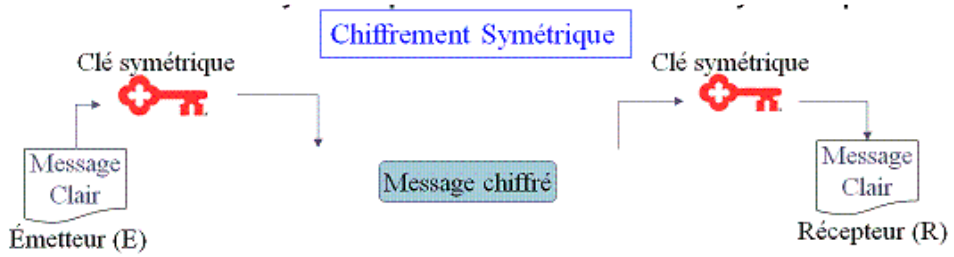

Figure 6 : principe du chiffrement symétrique [13]

Les algorithmes symétriques sont de deux types [14] :

1- Les algorithmes de chiffrement de flux (ou chiffrement par flot), qui agissent sur le message en clair un bit à la fois.

2- Les algorithmes de chiffrement par bloc, qui opèrent sur le message en clair par groupes de bits appelés bloc.

i. Algorithme de chiffrement de flux

Le principe consiste à générer un flux pseudo aléatoire et de le combiner avec l'information bit à bit par l'opération XOR[3]. A la réception, on applique le même mécanisme, et on restitue l'information. [14] Quelques exemples sur les algorithmes de chiffrement : RC4, E0, …

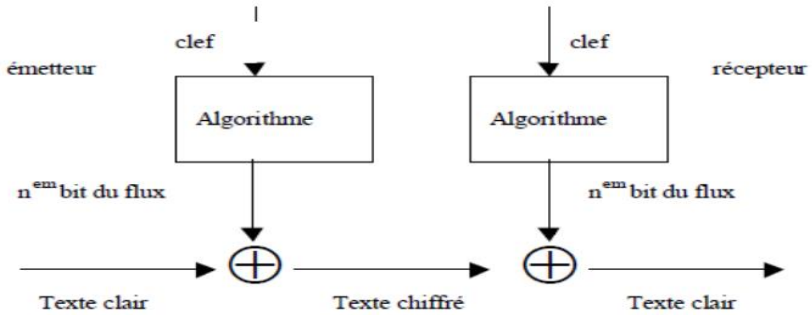

Figure 7 : Chiffrement par flux[14]

[3] La méthode XOR, appelée plus généralement fonction OU Exclusif est un opérateur logique.
Le principe repose sur 2 opérandes qui peuvent avoir comme valeur VRAI (1) ou FAUX (0), le résultat prendra lui aussi comme valeur VRAI ou FAUX ; VRAI dans le cas ou seulement l'un des deux est VRAI.

ii. Algorithme de chiffrement par bloc

Un algorithme de chiffrement par bloc (Block Cipher) transforme des blocs de données de taille fixe en bloc de données chiffrées de la même taille. Les blocs font généralement 128 bits, mais ils peuvent aller de 32 à 256 bits selon l'algorithme. La transformation reste la même pour chaque bloc. Quelques exemples sur les algorithmes de chiffrement : DES, AES, …

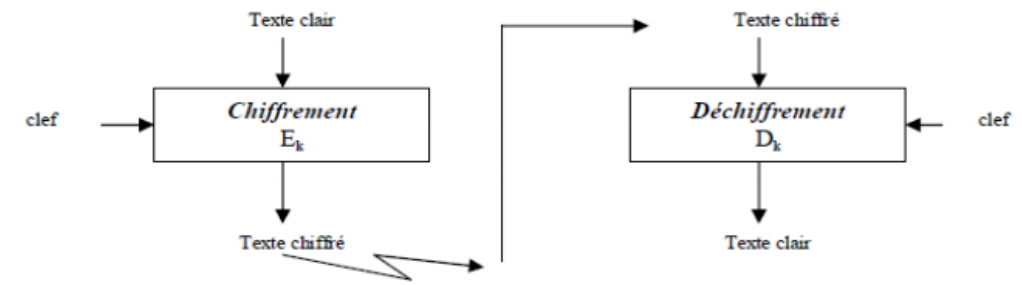

Figure 8 : Chiffrement par bloc[14]

2.1.2 Cryptographie asymétrique

Le principe de la cryptographie asymétrique (appelé aussi chiffrement à clé publique) est une méthode de chiffrement qui s'oppose à la cryptographie symétrique. Elle repose sur l'utilisation de clés, une clé publique (qui est diffusée) et une clé privée (gardée secrète), la première permettant de coder le message et la deuxième de le décoder, comme le montre la figure ci-dessous. Exemples sur le chiffrement asymétrique : RSA, ElGamel, ECC

Figure 9 : principe du chiffrement asymétrique [13]

Le tableau suivant représente les avantages et les inconvénients des deux types de la cryptographie [15] :

Cryptographie	Avantage	Inconvénient
Symétrique	Le chiffrement/déchiffrement est très rapide.	Sécurité faible ; l'utilisation d'une clé unique présente un problème lors de l'échange de clé.
Asymétrique	Renforce la sécurité, Même en interceptant le message, impossible de le décrypter sans la clé privée	Le chiffrement/déchiffrement est lent en raison de ces algorithmes complexes

Tableau 2 : Avantages et incovénients de la cryptographie symétrique/asymétrique

2.2 Algorithmes de chiffrement

Dans cette section, nous allons énumérer les différents algorithmes de chiffrements cités tout au long de mémoire comme présenté dans la figure suivante :

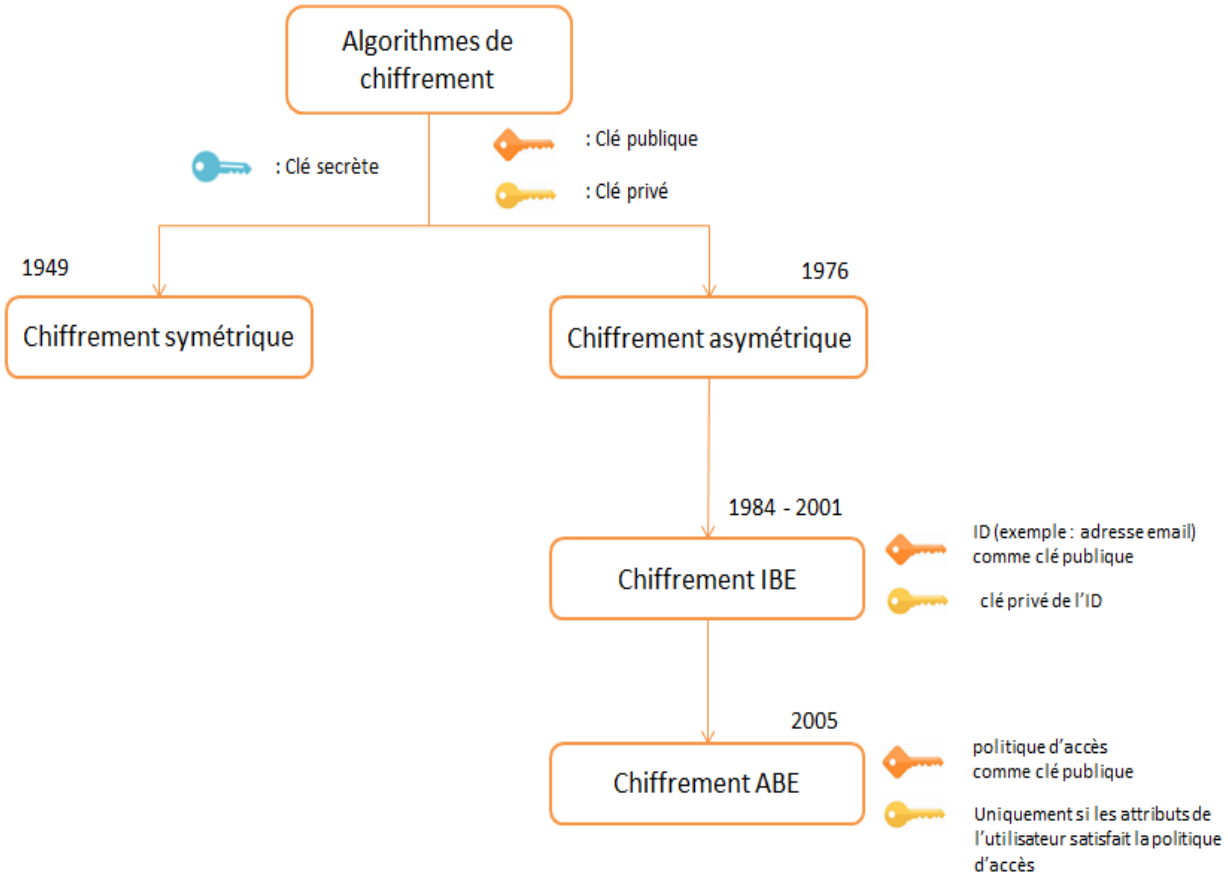

Figure 10 : Algorithmes de chiffrement

2.2.1 Chiffrement IBE (Identity Based-Encryption)

Le chiffrement asymétrique a résolu le problème d'échange de clefs, cependant le chiffrement par identité ou chiffrement IBE (Identity Based-Encryption) [16] permet de résoudre le problème de certification des clefs publiques. Dans le cadre de ce chiffrement, un utilisateur peut publiquement dériver une clef de chiffrement pour une identité donnée par exemple par une adresse de courrier électronique. L'utilisateur possédant l'adresse e-mail obtient de la part d'une autorité la clef secrète permettant de déchiffrer pour son identité.

Son principe est de prendre comme clef publique l'identité de l'utilisateur, par exemple son nom, prénom, date de naissance, ou son numéro social. Puis créer des clefs de chiffrement secrètes relatives à ces identités de telle sorte que deux individus différents ne puissent avoir la même clef secrète, alors il n'est plus utile de certifier les clefs publiques.

La contrainte de cette approche est que le niveau de confiance qui est accordé au générateur de clefs privées doit être très élevé, car il est intrinsèquement capable de régénérer la clef privée de tout utilisateur, et donc de pouvoir réaliser sans autorisation des signatures ou des déchiffrements.

Un certain nombre de variantes ont été proposées afin d'éviter ce problème tel que le chiffrement ABE

2.2.2 Chiffrement ABE (Attribut Based-Encryption)

Un schéma de chiffrement comme RSA et AES fournit une transmission et un stockage de données sécurisées dans un environnement Cloud mais l'inconvénient de ces schémas est la difficulté de mettre en place un contrôle d'accès avec une granularité fine pour le partage des données, en particulier dans le cas où nous ne connaissons pas l'identité des utilisateurs au préalable. Il se pose également le problème des mécanismes de révocation[4]. Une solution à ces problèmes est donnée par le chiffrement par attributs ou chiffrement ABE (Attribute Based Encryption) [17]

Le chiffrement ABE est un concept du chiffrement IBE, il incorpore un processus de génération des clés de chiffrement et de déchiffrement et la notion de politique d'accès basée sur des attributs, offrant des fonctionnalités de chiffrement et de contrôle d'accès. Le chiffrement ABE est un schéma de chiffrement à clé publique du type un-à-plusieurs, c'est-à-dire qu'on chiffre avec une seule clé et on a la possibilité de générer plusieurs clés pour déchiffrer. Un avantage évident de cette technique est que chaque utilisateur à une clé dédiée,

[4] Révocation : révoquer un utilisateur c'est à dire annuler ces droits pour accéder aux données d'où l'annulation de sa clé de déchiffrement

en cas de révocation d'une clé, il n'est pas nécessaire de refaire le chiffrement des données. Les données peuvent être chiffrées à la source et entreposées telles quelles, et à aucun moment le fournisseur de service n'accède au clair même que ce soit pour le processus de partage ou de révocation de droit. En plus de sécuriser la transmission et le stockage des données, ABE fournit un contrôle d'accès à forte granularité, une gestion des clés évolutives et une distribution de données flexible, Il permet de chiffrer les données et d'assurer le partage sur la base d'attributs descriptifs, sans aucune connaissance préalable de l'identité des destinataires.

Dans ABE, les données sont chiffrées et déchiffrées en fonction d'attributs et de politique d'accès. Seules les entités avec des attributs qui satisfont une politique d'accès aux données peuvent déchiffrer un texte.

La politique d'accès est représentée sous forme d'un arbre. Chaque nœud non foliaire de l'arbre représente une porte de seuil, décrite par ses enfants et une valeur de seuil. Chaque porte de seuil peut prendre soit « OR » qui prend la valeur 1of2 ou « AND » de la valeur 2of2. [18]

Afin de mieux comprendre, nous allons la donner un exemple ; supposons que l'univers d'attributs soit défini comme étant {Scanner, Grippe, Cancer, Infermière} et que le patient 1 reçoit une clé pour les attributs {Scanner, Cancer} et que le patient 2 reçoit une clé pour l'attribue {Grippe}.

Si le médecin chiffre un fichier et défini la politique étant : (Scanner ∧ Infermière) ∨ Grippe. Le patient 2 pourra alors déchiffrer, tandis que le patient 1 ne pourra pas déchiffrer. Comme présenter dans la figure 11 :

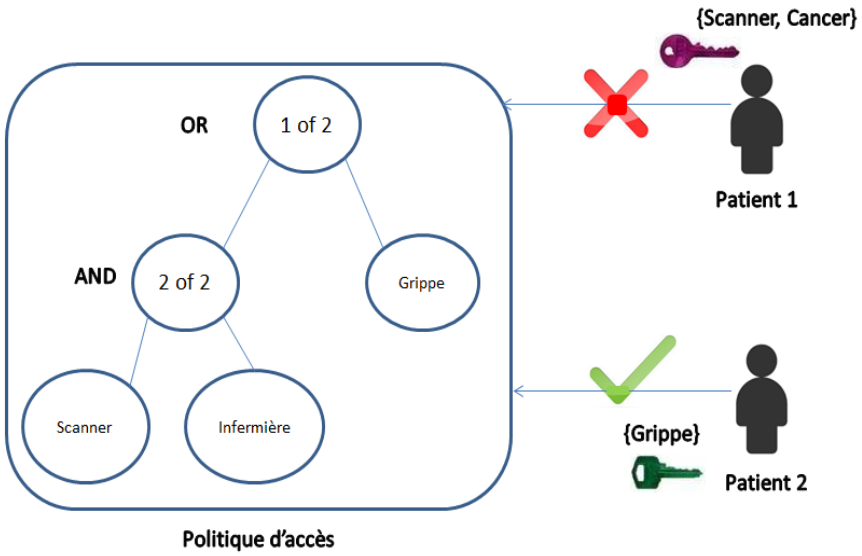

Figure 11 :Exemple structure d'accès ABE

I. Analyse de sécurité ABE

Un système basé sur ABE permet en théorie de fournir globalement les fonctionnalités suivantes [19] :

a. la confidentialité des données

Un utilisateur non autorisé ne peut pas connaître les informations sur les données chiffrées. Dans ce cas, les utilisateurs non autorisés n'ont pas assez attributs satisfaisant la politique d'accès. Ainsi, les accès non autorisés à partir du centre de génération des clés (KGC, Key Generation Center) doit être évité.

b. Résistance à la collusion

Les utilisateurs malhonnêtes ne peuvent pas combiner leurs attributs pour déchiffrer les données chiffrées. La résistance à la collusion est une des propriétés de sécurité importantes requises dans les systèmes ABE.

c. Révocation d'utilisateur / attribut

Les systèmes peuvent révoquer le droit d'accès de la personne qui quitte le système.

d. Evolutivité

Le nombre d'utilisateurs autorisés ne peut pas affecter les performances du système, c'est-à-dire que le système peut traiter avec le cas où le nombre d'utilisateurs autorisés augmente de manière dynamique.

II. Algorithme ABE

Le système ABE est constitué de quatre algorithmes [17] comme présenté dans la figure 12 :

a. Configuration (Setup)
b. Chiffrement (Encryption)
c. Génération des clés (KeyGen)
d. Déchiffrement (Decryption).

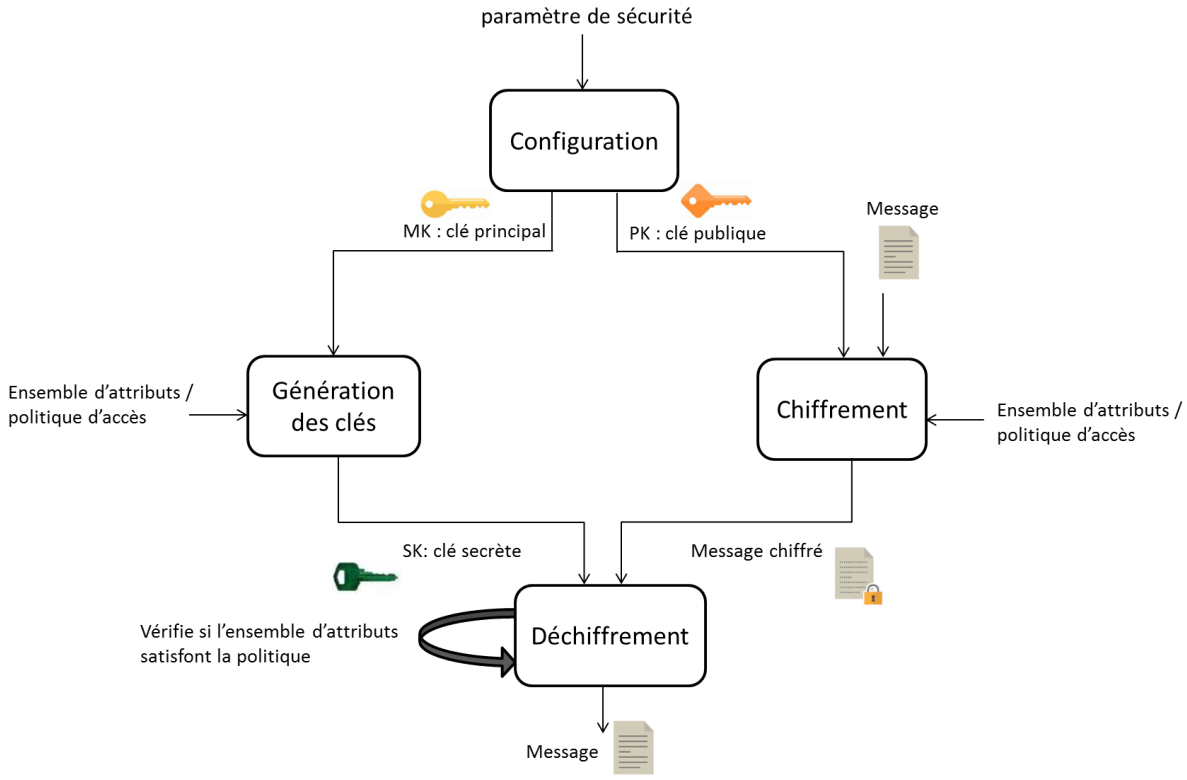

Figure 12 : Algorithme ABE

III. Approches ABE

Les entrées des algorithmes Génération des clés, Chiffrement et Déchiffrement prennent soit un ensemble d'attributs ou une politique d'accès selon l'approche utilisé :

a. Key Policy Attribute Based Encryption (KP-ABE), introduite par Goyal et al. [20].
b. Ciphertext-Policy Attribute Based Encryption (CP-ABE) introduite par Bethencourt et al. [21]

Dans le chiffrement KP-ABE (figure 13), l'utilisateur "A" chiffre un message à l'aide d'un ensemble d'attributs « I ». Il définit une structure d'accès, qui est un arbre de seuil de la politique que l'utilisateur "A" veut appliquer. L'utilisateur "B" et l'utilisateur "C" essaient de déchiffrer le message. Les attributs que "B" possède, satisfont la structure d'accès et lui permettent donc d'obtenir la clé et de déchiffrer le document. Les attributs que "C" possède, ne satisfont pas la structure d'accès et ne peuvent donc pas dériver la clé pour déchiffrer le message. L'idée de cette approche est que la clé est associée à la politique en utilisant une structure d'accès.

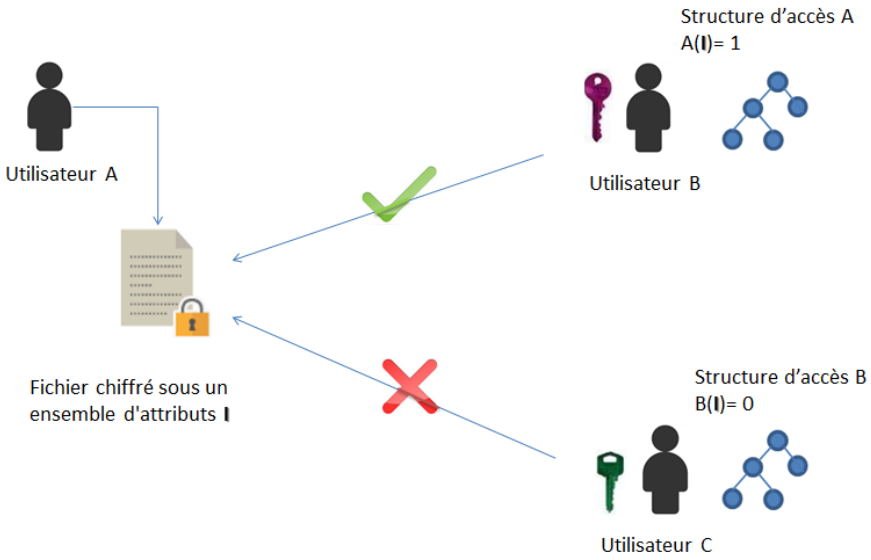

Figure 13 : Chiffrement KP-ABE [22]

Par ailleurs, le chiffrement CP-ABE (figure 14) inverse le rôle du chiffrement et de la génération de la clé. Le chiffrement est associé à une structure d'accès qui est construite à l'aide de la politique. Le serveur qui permet la génération des clés KGC émet simplement des clés privées pour les attributs dont disposent les utilisateurs. Si les attributs des utilisateurs satisfont la structure d'accès définie par le propriétaire, ils peuvent déchiffrer le message. Cette deuxième approche est plus proche du chiffrement que l'on trouve dans les systèmes ouverts car le texte chiffré est associé à la politique.

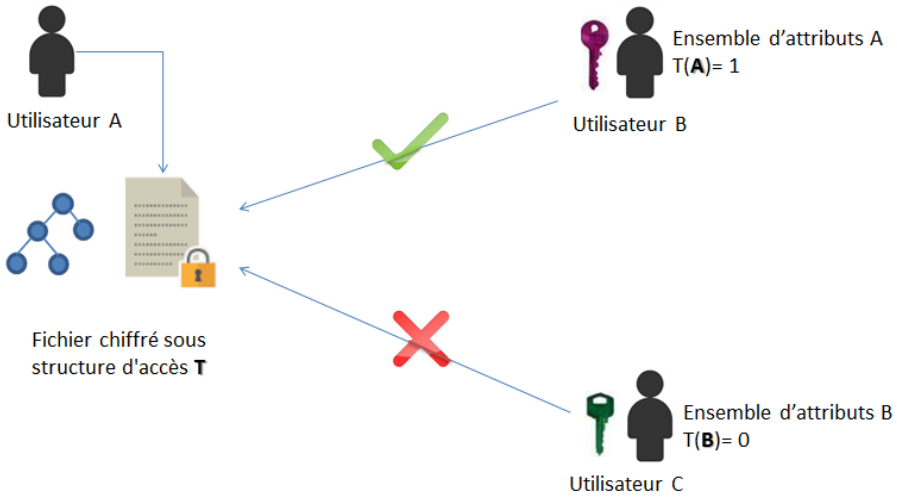

Figure 14 : Chiffrement CP-ABE [22]

Le chiffrement CP-ABE est plus approprié pour le système de partage de données et dans l'utilisation dans le domaine tel que la santé car il garde les décisions de politique d'accès sous la main des propriétaires de données. Cela améliore l'inconvénient de KP-ABE, à savoir que les données chiffrées ne peuvent pas choisir qui peut déchiffrer. Il peut prendre en charge le contrôle d'accès dans l'environnement réel. En outre, la clé privée de l'utilisation est dans le schéma, une combinaison d'un ensemble d'attributs, de sorte qu'un utilisateur utilise cet ensemble d'attributs uniquement pour satisfaire la structure d'accès dans les données chiffrées. Cependant, le régime CP-ABE présente encore quelques inconvénients. Les inconvénients des systèmes CP-ABE les plus existants ne répondent toujours pas aux exigences de contrôle d'accès de l'entreprise, qui exigent une flexibilité et une efficacité considérables. CP-ABE a des limites en termes de spécification de stratégies et de gestion des attributs d'utilisateur. [22]

Dans un schéma CP-ABE, les clés de déchiffrement ne prennent en charge que les attributs utilisateur organisés de manière logique en un seul jeu. Les utilisateurs ne peuvent donc utiliser que toutes les combinaisons possibles d'attributs d'un seul jeu émis dans leurs clés pour satisfaire les règles.

Au cours de notre mémoire, nous nous intéressons à l'approche CP-ABE

3. Contrôle d'accès

Le contrôle d'accès est une technique de sécurité qui contrôle qui ou quoi peut visualiser ou utiliser les ressources dans un environnement informatique. Il s'agit d'un concept fondamental en matière de sécurité qui minimise les risques pour l'entreprise. [23]

Il existe deux types de contrôle d'accès :

a. Le contrôle d'accès **physique** qui limite l'accès aux campus, aux bâtiments, aux salles et aux biens matériels informatiques.
b. Le contrôle d'accès **logique** qui limite les connexions aux réseaux informatiques, aux fichiers système et aux données.

Le contrôle d'accès est nécessaire au sein du cloud car il augmente l'efficacité et la sécurité d'une entreprise pour les raisons suivante [24] :

a. Flexibilité

La gestion et la surveillance centralisées des droits d'accès sur tous les sites de bureaux offrent l'agilité pour une entreprise en croissance et un avantage réel par rapport à ses concurrents.

b. Conformité

Grâce au contrôle d'accès basé sur le cloud, il permet à l'utilisateur de se connecter à n'importe quel service et de télécharger les données. Cela réduit le temps de transfert des données à partir des systèmes sur sites.

c. Mises à jour automatiques du logiciel

Les systèmes de contrôle d'accès Cloud résident dans le fait que les serveurs sont hors site et à l'abri des regards. Les fournisseurs gèrent les serveurs et publient des mises à jour logicielles régulières, y compris de nouvelles fonctionnalités et des mises à jour de sécurité, dans le but de ne pas perdre du temps l'utilisateur afin de gérer les différents systèmes des sites.

d. Collaboration accrue

L'utilisateur peut offrir des privilèges à un nombre de personne afin qu'il puisse avoir accès à son même service Cloud. Cela permet des flux de travail plus efficaces sans perdre le contrôle. Les différents rôles d'administrateur et la visibilité permettent de responsabiliser les personnes sans perdre aucun contrôle.

e. Zone géographie ouverte

Avec le contrôle d'accès en nuage, il est possible de gérer et surveiller l'accès aux services depuis n'importe quel endroit situé.

f. Sécurité

La perte de données sensibles met en danger la confidentialité, l'intégrité et la disponibilité du Cloud. La mise à jour du système de contrôle d'accès de n'importe où donne la possibilité de mettre à jour ou de supprimer les droits d'accès immédiatement. De plus, l'utilisateur peut synchroniser les droits d'accès d'étage avec une base de données.

3.1 Modèles de contrôle d'accès

Pour maitriser la sécurité de bout en bout, il faut sécuriser les éléments constituant la plate-forme Cloud, mais également l'accès à cette plate-forme, Il existe plusieurs modèles de contrôle d'accès [25] :

3.1.1 Contrôle d’accès obligatoire (MAC)

Le mécanisme de contrôle d'accès obligatoire (MAC : Mandatory Access Control) est le mécanisme traditionnel permettant de définir les droits d'accès des utilisateurs. MAC donne l’autorisation d’accès via le système d’exploitation. Il contrôle la capacité des propriétaires de données à accorder ou à refuser des droits d'accès aux clients du système de fichiers. Tous les droits de contrôle d'accès sont définis par le gestionnaire du système et imposés par le système d'exploitation. Les clients n'ont aucun droit de modifier ces droits d'accès. Dans ce modèle, chaque objet du système de fichiers possède une étiquette de classification telle que niveau secret, niveau top secret ou confidentiel. Chaque appareil et chaque client se voient attribuer un niveau de classification et de dégagement similaire. Le système d'exploitation vérifie les informations d'identification de chaque personne ou système lors de l'accès à une ressource particulière pour déterminer les droits d'accès de cette personne ou de ce périphérique spécifique. Même si MAC offre plus de sécurité pour accéder aux ressources, il dispose d'un environnement moins flexible pour traiter les droits d'accès.

3.1.2 Contrôle d'accès discrétionnaire (DAC)

Le contrôle d’accès matriciel ou bien le contrôle d’accès discrétionnaire (DAC : Discretionary Access Control) est un mécanisme de contrôle d’accès de sécurité qui contrôle les autorisations d’accès via le propriétaire des données. Les modèles DAC sont discrétionnaires car le propriétaire détermine les privilèges d'accès. Dans ce modèle, les droits d'accès de chaque utilisateur sont définis lors de l'authentification en validant le nom d'utilisateur et le mot de passe. De plus, les fichiers ou les données se trouvent chez le propriétaire et les politiques d'accès aux données sont contrôlées par le propriétaire des données [26]. Le DAC offre plus de flexibilité que le MAC, cependant, il fournit moins de sécurité.

3.1.3 Contrôle d'accès basé sur les rôles (RBAC)

Le contrôle d'accès basé sur les rôles (RBAC : *Role-Based Access Control*) fournit des droits d'accès basés sur les rôles et privilèges des utilisateurs. Les autorisations utilisateur sont définies par différents paramètres de RBAC, tels que les rôles d'utilisateur, les permissions de rôle et les relations de rôle. Les rôles sont classés en deux catégories [25]:

- Rôle d’application / technique qui contient la combinaison de différents droits spécifiques à une application ou autorisations basées sur des tâches et son étendue est limitée à l'application spécifique.
- Rôle d’organisationnel/professionnel qui est généré en fonction de différentes fonctions et droits d'accès attribués à un employé.

Un rôle organisationnel/professionnel est une combinaison de différents rôles applicatifs/techniques. RBAC fournit un environnement hautement sécurisé pour l'attribution d'autorisations d'accès et assure la sécurité de l'administration dans les organisations avec un grand nombre d'utilisateurs. Les autorisations d'accès aux données sont fournies aux utilisateurs en fonction de ces règles.

La principale limitation de RBAC est que les rôles attribués peuvent changer de temps en temps, ce qui nécessite un environnement en temps réel pour vérifier et valider les modifications.

3.1.4 Contrôle d'accès basé sur les attributs (ABAC)

Le contrôle d’accès basé sur les attributs (ABAC : Attribute Based Access Control) est un mécanisme permettant de contrôler les autorisations d’accès. ABAC définit le mécanisme de contrôle d'accès par l'utilisation de politiques qui déterminent différents ensembles d'attributs pour vérifier les droits d'accès de chaque utilisateur. Les politiques sont générées à l'aide de différents types d'attributs en fonction de la politique, le système détermine les autorisations d'accès. Les attributs considérés sont les attributs de sujet, les attributs d'objet, les attributs de ressource et les attributs environnementaux. Dans le modèle ABAC, les rôles et les privilèges de chaque utilisateur sont prédéfinis. Il résout de nombreux problèmes d'autorisation, et il permet d'obtenir un enregistrement efficace.

3.1.5 Comparaison entre les modèles

Dans la figure 15, nous illustrons les différentes architectures des modèles de contrôles d’accès.

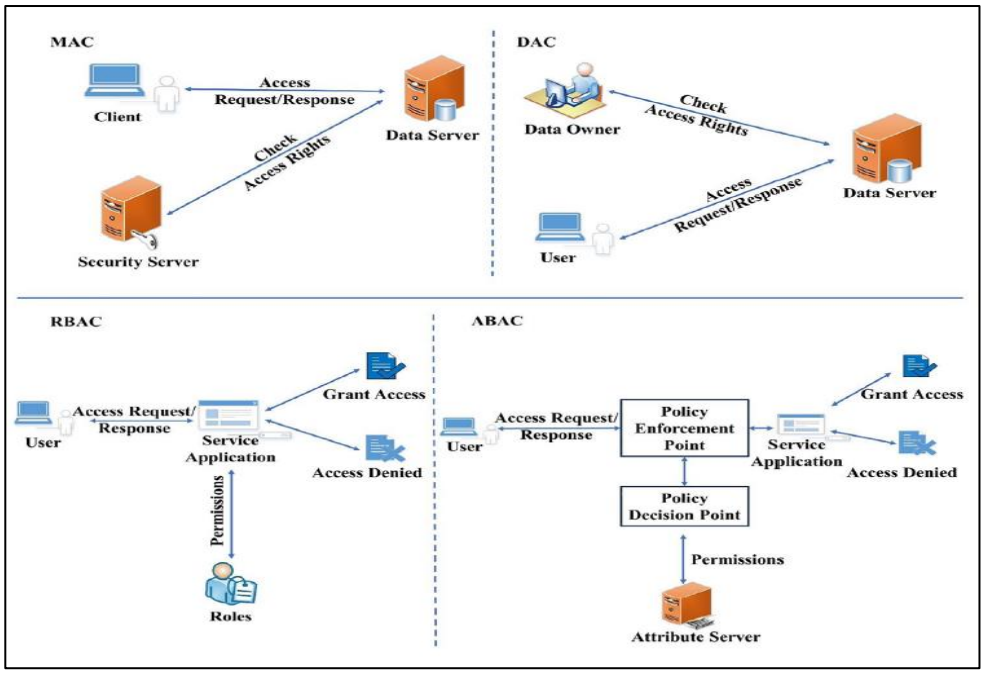

Figure 15 : les architectures des modèles contrôle d'accès [25]

Dans le tableau suivant, nous comparons les quatre modèles de contrôle d'accès en termes de sécurité (points forts et menaces/attaques possibles dans chaque modèle).

Mécanismes	Points forts (sécurité)	Points faibles (menaces)
DAC	L'application possède les autorisations d'accès individuelles	Accès non autorisé, sur classification des données, difficile à mettre en œuvre,
MAC	Les droits d'accès à une application sont détenus et contrôlés par une autre application.	Sensibilité aux chevaux de Troie, Logiciels défectueux, Failles d'information, Attaques malveillantes
RBAC	Les droits et les privilèges d'accès à plusieurs applications sont regroupés en tant que rôle organisationnel	Problèmes administratifs, problèmes d'abstraction des données, problèmes en temps réel
ABAC	Les Droits d'accès et les privilèges sur une application sont déterminés en fonction des attributs de sujet, d'objet, de stratégie et d'environnement	Problèmes de délégation, problèmes d'administration, problèmes d'audit et d'évolutivité

Tableau 3 : sécurité et menaces des modèles de contrôle d'accès [25]

Ces modèles de contrôle d'accès dans le cloud posent certain problèmes au niveau de la sécurité comme résumé dans le tableau 3. Il existe plusieurs approches, modèles et architectures qui ont été proposés pour le contrôle d'accès aux données dans le Cloud. La majorité des approches proposées reposent sur des techniques cryptographiques pour renforcer le contrôle d'accès, assurer la confidentialité des données et des privilèges des utilisateurs et sécuriser la distribution de ces derniers. En effet, les politiques d'accès sont spécifiés et renforcer d'une façon cryptographique par l'utilisation des algorithmes de chiffrement. Dans la section 4, nous allons présenter le contrôle d'accès par le chiffrement.

4. Contrôle d'accès par chiffrement

Il existe de nombreuse approches proposées qui se reposent sur le chiffrement afin de renforcer le contrôle d'accès et d'assurer la confidentialité des données et des privilèges des utilisateurs et pour sécuriser la distribution de ces derniers [27] :

4.1 DACC : Contrôle d'accès distribué dans le Cloud

Ruj, Nayak et Stojmenovic [28] ont proposé une approche distribuée pour le contrôle d'accès appelé Distributed Access Control in Cloud (DACC).

DACC utilise l'approche cryptographique ABE (Attribute-Based Encryption) et se base sur l'utilisation d'une architecture décentralisée où plusieurs centres de distribution des clés (Key Distribution Centers, KDC) sont utilisés. Cette approche permet le stockage et le partage des données qui résident dans les serveurs Cloud en une seule copie où le prioritaire des données chiffre leur données en utilisant ABE avant de les stocker dans le Cloud. Les utilisateurs récupèrent les données dans leurs formats chiffrés et seuls les utilisateurs autorisés par l'accès à ces données sont capables d'avoir les données en claire après le déchiffrement par les clés secrètes.

Dans l'approche DACC (figure 16), le système est composé de quatre éléments : le propriétaire des données, les utilisateurs, les centres de distribution des clés (KDC) et les serveurs Cloud où les données sont stockées.

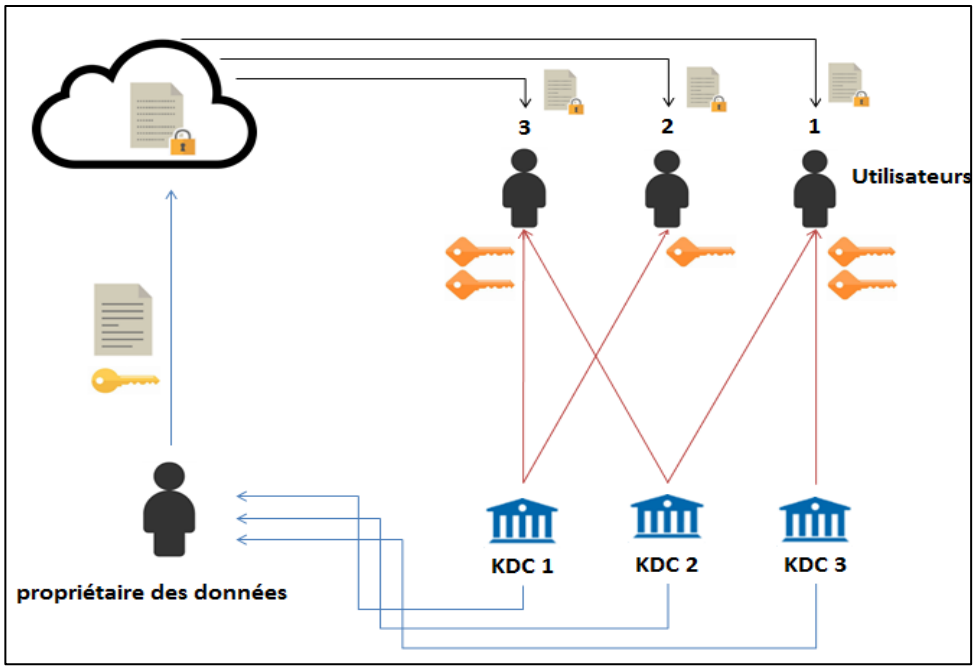

Figure 16: Approche DACC

Le propriétaire des données définit un ensemble des attributs et les associe à ses données. Par la suite, ces données sont stockées dans les serveurs Cloud dans un format chiffré. Les KDC distribuent les clés aux utilisateurs en se basant sur leurs attributs individuels. Les utilisateurs sont capables de récupérer les données chiffrées des serveurs Cloud et les attributs associés qui leur permettent l'accès aux données si ces données sont signées par les même attributs possédés par ces utilisateurs.

DACC utilise le protocole sécurisé Secure SHell (SSH) pour transférer les clés privées à partir des centres de distribution des clés vers les utilisateurs et pour transférer les données chiffrées du Cloud aux utilisateurs, ceci augmente la confidentialité dans DACC.

L'approche DACC assure que les données ne sont pas accessibles ni par le Cloud, ni par les utilisateurs non autorisés, et elle assure que si un ensemble d'utilisateurs non autorisés tente d'accéder individuellement à une donnée, ces utilisateurs ne seront pas capables de combiner leurs clés de décryptage pour accéder à cette donnée.

DACC permet aussi une révocation des utilisateurs et les utilisateurs révoqués perdent ces droits d'accès aux données.

La révocation des utilisateurs se base sur le recalcule et la redistribution d'une partie du texte chiffré et n'exige aucun besoin pour redistribuer les clés à nouveau pour les utilisateurs non révoqués.

4.2 TAAC : Le contrôle d'accès basé sur les attributs temporels pour les systèmes de stockage multi-autorités dans le Cloud

Le contrôle d'accès basé sur les attributs temporels pour les systèmes de stockage multi-autorités dans le Cloud (Temporal Attribute-Based Access Control for Multi-Authority Cloud Storage Systems, TAAC) proposée dans [29] est une approche efficace pour le contrôle d'accès aux données pour les systèmes de stockage des données multi-autorités dans le Cloud où les attributs des utilisateurs appartiennent à des domaines différents. Les attributs dans TAAC sont gérés par des autorités indépendantes les unes aux autres et aucune autorité centrale n'est utilisée pour les faire communiquer. L'utilisateur dans un tel système peut avoir un ou plusieurs attributs qui appartiennent à un ou plusieurs autorités où chaque attribut appartient à une et une seule autorité.

TAAC se repose sur l'utilisation de l'approche CP-ABE, mais à l'inverse des approches CP-ABE de base, il résout le problème de révocation des attributs pour les utilisateurs.

TAAC permet un contrôle d'accès temporel où le temps dans le système est composé à des intervalles. Au début de chaque intervalle, chaque autorité et sans avoir besoin d'interagir avec les autres peut révoquer ou bien associer des attributs appartenant à son domaine de n'importe quels utilisateurs et à n'importe quels utilisateurs. Lors l'association d'un attribut à un utilisateur, un nœud feuille de la structure de l'arbre définie pour cet attribut est associé à l'identité globale de cet utilisateur. La structure de l'arbre est utilisée pour gérer les utilisateurs assignés avec cet attribut où chaque autorité définit une telle structure pour chaque attribut appartient à son domaine.

Pour la révocation des attributs, au début de chaque intervalle toutes les autorités maintiennent une liste des attributs révoqués qui convient à certains utilisateurs. TAAC utilise un algorithme efficace pour calculer l'ensemble minimal des utilisateurs non révoqués et mettre à jours leurs clés secrètes. En plus, aucune opération de re-chiffrement pour le texte chiffré n'est exigée lors la révocation. La révocation des attributs n'affecte pas les privilèges de déchiffrement pour les autres attributs possédés par cet utilisateur et n'affecte pas les autres utilisateurs qui possèdent cet attribut révoqué, ceci qui rend TAAC efficace et flexible pour les applications.

Le système dans l'approche TAAC est composé de quatre entités (figure 17) : le prioritaire des données, l'utilisateur, le serveur Cloud et les autorités des attributs.

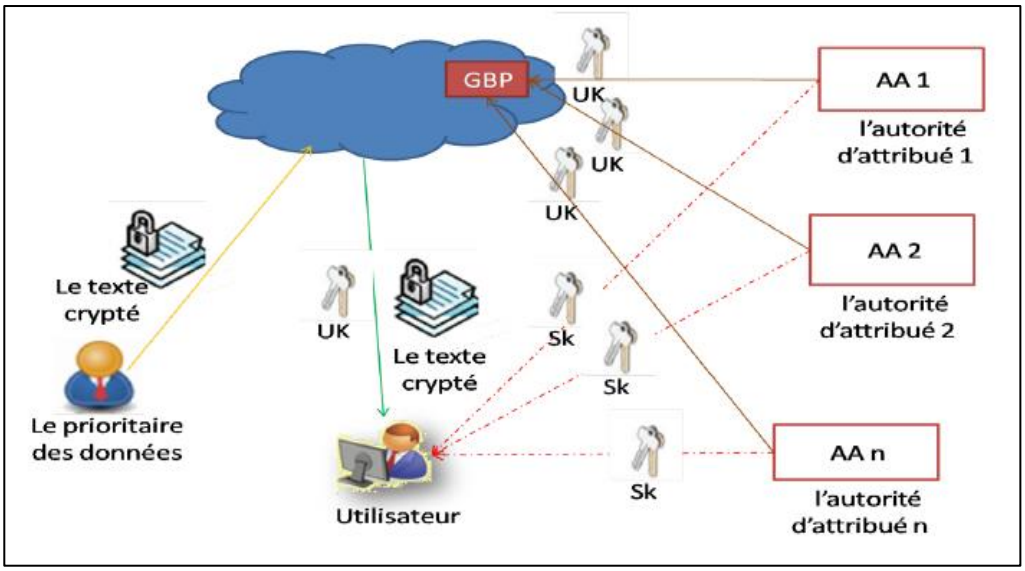

Figure 17 : Approche TAAC [27]

Le prioritaire des données définit la politique d'accès aux données à un intervalle de temps précis, la structure d'accès et l'intervalle de temps sont associés avec le texte chiffré sauvegardé dans le Cloud en une seul copie. L'utilisateur peut récupérer le texte chiffré du Cloud et seulement les utilisateurs qui possèdent les attributs et satisfont la politique d'accès à l'intervalle associé avec le texte chiffré peuvent générer la clé de décryptage et par la suite de décrypter le message. Cette génération de la clé de décryptage se fait par l'utilisation des clés secrètes ou bien des clés secrètes modifiées pour ces attributs, les utilisateurs reçoivent leurs clés secrètes des autorités des attributs. En cas de la révocation de certains de ces attributs, les utilisateurs non révoqués sont capables de récupérer les clés modifiés du Grand Bulletin publique des autorités de ces attributs révoqués qui résident dans le Cloud.

4.3 Un mécanisme basé sur les provenances pour le contrôle d'accès aux données

Les provenances des données sont les historiques des métadonnées qui détaillent l'origine d'un objet. Ces provenances sont riches en informations contextuelles, ceci permet l'utilisation de ces derniers dans les applications critiques de la sécurité. L'utilisation des provenances dans les applications de contrôle d'accès permet une décision expressive indépendamment des politiques.

Bates et ses collègues ont proposé dans [30] une architecture et des protocoles associés pour un management sécurisé et distribué des provenances dans l'environnement Cloud. Par la suite, ces protocoles sont utilisés pour développer un mécanisme de contrôle d'accès qui se base sur les provenances des données. Dans ce mécanisme, la décision d'accès se base sur les

attributs provenances. Le contrôle d'accès basé sur les attributs (ABAC) offre plus de la scalabilité, la flexibilité et la capacité pour la modularité des politiques. Lorsqu'un changement est produit à une politique, un seul attribut doit être vérifié. Les attributs des données peuvent être extraits et utilisés pour créer des labels qui sont utilisés dans la décision de contrôle d'accès. Les labels qui se basent sur les provenances possèdent la capacité de s'adapter en cas de changement dans les politiques et d'exprimer la décision d'accès en se basant sur la variété des attributs et des niveaux de granularité. Les provenances sont détachées des données et ils sont gérés par les différentes autorités des provenances Cloud où ces dernières peuvent coopérer entre eux pour avoir les provenances des données.

L'autorité des provenances Cloud proposée dans [30] collecte et manage les métadonnées des provenances Cloud. Cette autorité est le responsable sur l'arbitrage du processus de contrôle d'accès aux données. Les composants principaux de l'autorité des provenances Cloud sont (figure 18):

- PEP (Policy rEnforcement Point) : qui est le point de renforcement de la politique. Ce module contrôle l'intégrité des requêtes des utilisateurs et renforce le contrôle d'accès aux données selon les politiques d'organisation.
- PDP (Policy Decision Point) : qui est le point de la décision de la politique. Son rôle est d'avoir les politiques d'une organisation et ensuite d'évaluer les requêtes utilisateur selon ces politiques.
- La base de données des politiques : qui contient les politiques de sécurité.
- La base de données des provenances Cloud : qui contient les informations des provenances Cloud.

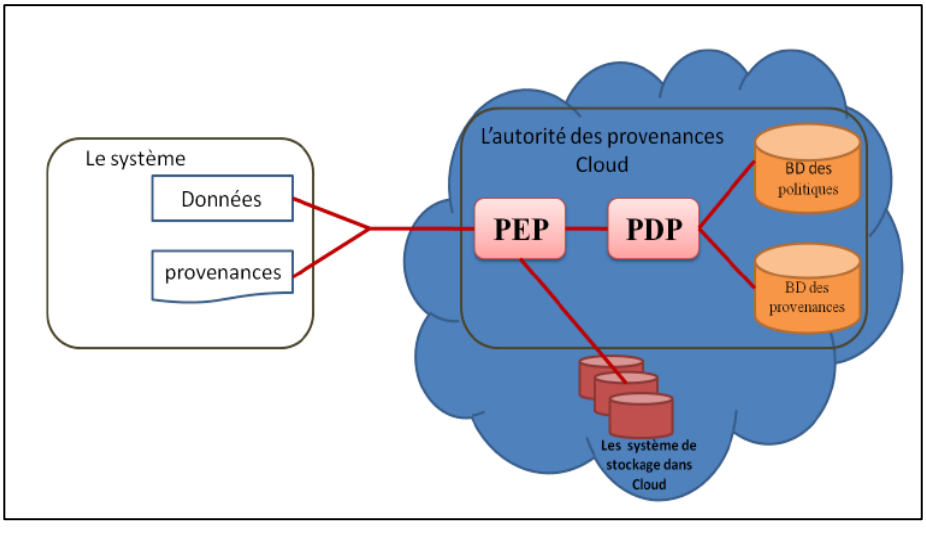

Figure 18 : L'architecture de l'autorité des provenances Cloud [27]

Ce système supporte trois types des opérations : l'écriture des données vers le Cloud, la lecture des données stockées dans le Cloud et la lecture des provenances stockées dans le Cloud. Les opérations de lecture et d'écriture d'après ou vers les serveurs Cloud sont similaires. Lors la réception d'une requête d'écriture qui contient les données et les provenances de ces données, le PEP reçoit la requête, vérifie l'intégrité de cette requête et par la suite, il envoi cette dernière au PDP. Le PDP évalue la requête selon l'ensemble de ses politiques et il envoi ensuite la réponse au PEP. Si la requête est acceptée, le PEP stockera les données dans les serveurs de stockage et les provenances dans la base des provenances du PDP, après il informe l'utilisateur.

Dans le cas d'une requête de lecture, le PDP reçoit la requête, il retrouve les provenances dans la base des provenances et il évalue la requête selon ses politiques. Si l'accès est accepté, le PEP envoi le fichier de données à l'utilisateur. Si un utilisateur demande la lecture des provenances de données le PEP reçoit la requête et il approuve cette dernière à travers le PDP qui partage les informations des provenances au PEP pour qu'il les retourne à l'utilisateur.

Dans l'autorité des provenances Cloud, le PEP est semblable à un proxy transparent pour toutes les opérations de stockage dans Cloud.

4.4 Discussion des travaux

Comme résumé dans le tableau 4, Les approches proposées DACC, TAAC et le mécanisme basé sur les provenances sont des approches de contrôle d'accès aux données. Elles permettent le stockage des données dans les serveurs Cloud et le partage de ces dernières entre plusieurs utilisateurs.

Le contrôle d'accès DACC et TAAC utilisent la technique ABE. Afin de prévenir les violations causées par les vulnérabilités de sécurité et assurer ainsi la confidentialité des données et les privilèges des utilisateurs. Par ailleurs, le mécanisme basé sur les provenances utilise une implémentation PEP/PDP, les attributs provenances sont utilisés dans la décision d'accès. L'utilisation de ces attributs permet une décision expressive de contrôle d'accès.

Contrairement à l'approche [30], les approches DACC et TAAC permettent la révocation des utilisateurs. Cependant, DACC exige le recalcule d'une partie du texte chiffré et la redistribution de cette partie à nouveau aux utilisateurs non révoqués. Dans cette approche, la révocation d'un attribut d'un utilisateur exige la révocation de tous les privilèges de déchiffrement pour les autres attributs de cet utilisateur. Dans l'approche TAAC, la révocation n'exige aucun recalcule et redistribution du texte chiffré et la révocation d'un attribut n'affecte pas les autres attributs possédés par cet utilisateur.

	DACC : mécanisme distribué [28]	TAAC : mécanisme basé sur les attributs temporels [29]	Mécanisme basé sur les provenances [30]
Modèle de contrôle d'accès	ABAC	ABAC	ABAC + implémentation PEP/PDP
Gérant des attributs	Propriétaire des données	Autorité des attributs	Autorités des provenances
Technique de chiffrement	ABE	CP ABE	Aucune
Composants du système	1. le propriétaire des données 2. les utilisateurs 3. les centres de distribution des clés (KDC) 4. les serveurs Cloud.	1. le prioritaire des données 2. l'utilisateur 3. le serveur Cloud 4. les autorités des attributs.	1. l'utilisateur 2. le serveur Cloud 3. les autorités des provenances
Architecture distribuée	Multi KDCs	Multi Autorité des attributs	Multi Autorités des provenances
Révocation des utilisateurs	Oui avec l'opération de calcule	Oui sans opération de calcule	non
Autres mécanismes de sécurité	Protocole SSH pour distribuer les clés privées et les données chiffrées.	Contrôle d'accès temporel : la structure d'accès et l'intervalle de temps sont associés avec le texte chiffré.	Non spécifié
Domaine d'application	e-santé (EHR)	Non spécifié	Non spécifié

Tableau 4 : résumé de travaux des modèles de contrôle d'accès

Après cette synthèse, nous optons pour un modèle de contrôle d'accès basé sur les attributs (ABAC) et sur le chiffrement CP-ABE pour sécuriser notre système de e-santé dans le cloud. La conception de notre approche sera détaillée dans le chapitre 4

5. Conclusion

Au niveau de ce deuxième chapitre, nous avons étudié certain nombre des travaux du contrôle d'accès dans le Cloud par le chiffrement ABE. En effet, le contrôle d'accès est le mécanisme fondamental de sécurité dans l'environnement Cloud mais il est insuffisant pour la sécurisation de cet environnement où les accès par le réseau rendent les attaques et les intrusions de plus en plus dommageables. C'est pour cela que les contrôles d'accès par chiffrement sont apparus afin de renforcer la sécurité. Les fonctionnalités du chiffrement ABE sont intéressantes pour une solution assurant la protection de la vie privée. Aucune autre personne ne pourra pas donc avoir accès aux données en clair car les données sont d'abord chiffrées ensuite stockées dans le cloud.

Cependant, autres informations (comme les informations d'inscription ou de connexion) sont envoyées en clair au cloud, ce qui risque de dévoiler certaines informations sensibles de l'utilisateur tel que son nom, numéro social etc… Le mécanisme de contrôle d'accès n'est pas capable de contrôler les activités des clients sur ces objets après leur allocation. Le chapitre suivant est consacré à l'étude des approches d'anonymat et d'authentification qui permettent de rendre le Cloud plus sécurisé.

Chapitre III : Anonymat et Authentification au sein du Cloud

1. Introduction

Les données stockées dans le cloud peuvent être très sensibles d'où la nécessité de garantir la confidentialité et la sécurité de ces derniers en utilisant des techniques et des méthodes adaptées, tel que le contrôle d'accès et le chiffrement. D'autre part, dans certaines circonstances, les utilisateurs souhaitent rester anonymes (i.e. garder leur identité inconnues) dans le cloud. Par exemple, dans le domaine de la santé, l'anonymisation est nécessaire pour plusieurs raisons. Il peut être utilisé pour fournir des statistiques sur les données médicales sans révéler l'identité des patients. Il permet également de masquer l'identité des patients de certaines institutions (par exemple, des compagnies d'assurance). En outre, les tiers non autorisés ne doivent pas apprendre la communication entre le patient et son médecin.

Ce chapitre est divisé en trois parties. Nous allons commencer par présenter les notions de base l'anonymat ainsi que ses approches et ses modèles existants. Nous étudions ensuite l'authentification dans le cloud afin d'achever avec l'authentification anonyme.

2. Anonymat

L'anonymat est l'un des éléments pouvant masquer les informations d'identité de l'utilisateur afin de préserver la confidentialité de l'utilisateur ou de l'organisation. Il peut protéger les informations de l'utilisateur contre les abus de la part d'un attaquant. Il peut également protéger contre les attaques malveillantes. Dans cette partie, nous allons décrire les propriétés, les méthodes et les approches de l'anonymat.

2.1 Propriétés d'anonymat

L'anonymat signifie que l'identité d'un utilisateur est inconnue dans le système. Il fournit les propriétés suivantes [31] :

a. **L'identifiabilité :** est la possibilité de connaître la véritable identité d'une partie au système par le biais de données réelles échangées dans le système.
b. **La traçabilité :** est la capacité d'obtenir des informations sur les parties en communication en observant le contexte de communication (par exemple via l'adresse IP).

c. **La non-liaison (Unlinkability) :** signifie qu'un adversaire qui suit la transaction de données entre certains expéditeurs et un destinataire ne peut pas établir relation entre les données et l'expéditeur.

2.2 Niveau de sécurité de l'anonymat

La sécurité de l'anonymat peut être classifiée en plusieurs niveaux selon la portée de la confiance, comme présenté dans la figure suivante [35] :

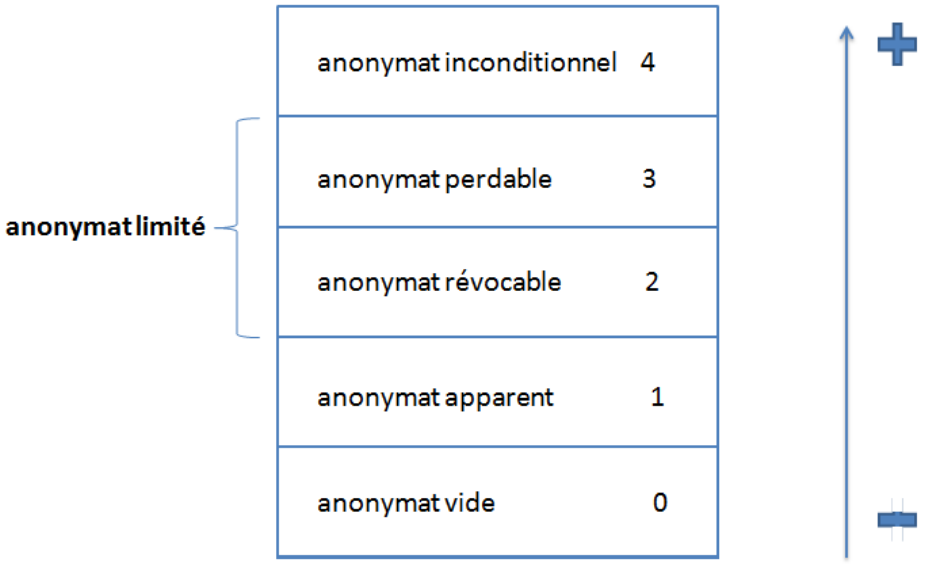

Figure 19 : Niveaux de sécurité de l'anonymat

a. Niveau 0 (anonymat vide)

À un extrême de l'échelle, les personnes sont complètement privées de la possibilité de dissimuler leur identité et peuvent devenir directement responsables de leurs actes. Certaines données IPI (Informations Personnellement Identifiables)[5] directe sont publiquement observables. Par conséquent, si un service expose des pseudonymes publics (par exemple, des clés publiques non chiffrées dans une PKI), il ne peut pas être reconnu comme anonyme. L'anonymat de ce niveau assure la liaison (inconditionnelle/linkability).

[5] Informations Personnellement Identifiables (IPI) : des informations suffisantes pour identifier de manière unique, ou pour retrouver une personne spécifique ou, éventuellement, une partie de ces informations. En conséquence, les informations personnelles sont simplement destinées à assurer l'identification des individus

b. Niveau 1 (anonymat apparent)

Un moyen légèrement plus avancé de fournir l'anonymat peut être l'application des IPI indirectes, ce qui complique quelque peu l'appel des personnes. Supposons que ces données indirectes relatives aux données personnelles sont observables par des entités méfiantes (par exemple, par le public). L'identification peut encore être effectuée, même si c'est un peu plus difficile, avec l'aide d'une tierce partie prenante, l'anonymat apparent présume qu'au moins une de ces entités appartient à l'ensemble des personnes méfiantes - sinon, il faudrait passer à l'anonymat révocable (niveau 2). L'utilisation de pseudonymes initialement non publics[6] (par exemple, des numéros de carte de crédit ordinaires) fournit un exemple évident de ce scénario, pour autant que les conditions de visibilité ci-dessus soient remplies. Comme scénario alternatif pour atteindre l'anonymat apparent. L'introduction des données IPI directes, à condition qu'elles soient publiquement non observables, tout en étant observable pour toutes les entités de confiance et certains participants méfiants. Lorsqu'il n'y a pas d'entité méfiante, il faut procéder à l'anonymat limité (niveau 2-3).

c. Niveau 2 (anonymat révocable)

Comme mentionné précédemment à propos de l'anonymat apparent, les IPI indirects non cachés peuvent promouvoir un anonymat révocable, à condition qu'il soit observable par des entités méfiantes, mais aucune identification ne peut être effectuée sans impliquer un responsable de l'identité. Par conséquent, si ces conditions sont remplies, les pseudonymes initialement non publics peuvent à nouveau illustrer le niveau. Un autre scénario, qui peut être affecté au même périmètre de confiance, consiste à exposer les informations personnelles confidentielles directes à des participants de confiance de manière à ce qu'elles ne soient pas observables par des entités méfiantes.

Le terme "révocable" découle de la considération suivante. Il n'est pas difficile de concevoir que, pour des raisons valables, un responsable de l'identité devienne enclin à révéler son identité à une tierce personne et révoque ainsi l'anonymat de l'utilisateur. Un tel processus de révocation peut être déclenché, par exemple,

[6] Le lien entre un pseudonyme initialement non public et son titulaire peut être connu de certaines parties mais n'est pas public au moins au début. Par exemple, un compte bancaire sur lequel la banque peut rechercher la liaison peut servir de pseudonyme non public. Pour certains pseudonymes non publics spécifiques, les autorités de certification pourraient révéler l'identité du titulaire en cas d'abus.

- dans les cas de fraude lorsque les autorités réclament l'identité d'un adversaire
- par désobéissance, dès que certaines règles et / ou politiques prédéfinies ont été violées
- par expiration du droit à l'anonymat.
- si une personne a elle-même intérêt à révoquer son identité.

Assurer la révocabilité est l'impossibilité de reconnaître ou traçabilité des informations des personnes. À ce niveau, les identités sont entièrement soumises aux gestionnaires d'identité, étant ainsi exposé à la révocation.

d. Niveau 3 (anonymat perdable)

À ce niveau, les individus restent anonymes aussi longtemps que les politiques, les règles et, surtout, la loi sont respectées. En cas de violation de l'un d'entre eux, la personne incriminée perd son anonymat. Le raisonnement derrière l'idée de confiscation peut être étayé par diverses situations réalistes

La possibilité de confiscation peut être avantageuse, par exemple
- dans les protocoles de paiement électronique pour éviter les doubles dépenses ou
- dans les systèmes de tourniquets pour limiter les déplacements d'une personne à une zone définie (par exemple, sur le lieu de travail).

Le fait de permettre la confiscation de l'anonymat peut permettre d'éviter un comportement inapproprié des utilisateurs en dissuadant les individus de commettre une fraude ou d'être désobéissants.

e. Niveau 2-3 (anonymat limité)

La distinction établie entre les anonymats niveau 2 et 3 est que la révocabilité nécessite un participant (responsable de l'identité) qui est responsable de la révocation; alors que la confiscation exiger un mécanisme de confiscation induit par des événements qui permet aux IPI de l'utilisateur (direct ou indirect) d'être observables et aux adversaires de devenir connus.

Un anonymat limité peut non seulement empêcher les abus, mais aussi assurer aux personnes bienfaisantes (et même aux autorités) que les adversaires peuvent potentiellement être appelés à rendre des comptes, dans le respect des droits de la personne. Sous tous les aspects, l'anonymat limité semble être le meilleur compromis entre les préoccupations de confidentialité et la responsabilité.

Les deux niveaux doivent être examinés conjointement. Leur analogie découle directement de la nature commune et fondée sur la confiance des concepts, à savoir. Dans le

premier cas, ce sont les utilisateurs qui doivent faire confiance aux TTP (Trusted Third Parties), tandis que le second repose sur l'hypothèse que chaque individu est enclin à obéir non seulement à la loi, mais également aux règles et / ou politiques correspondantes. En tout état de cause, les deux niveaux garantissent l'anonymat grâce à un cadre de confiance relativement étroit et limité.

f. Niveau 4 (anonymat inconditionnel)

À l'autre extrémité du spectre de l'anonymat, après s'être assuré qu'aucune des définitions de classe précédentes n'est satisfaite, on peut reconnaître un service comme étant inconditionnellement anonyme. L'anonymat inconditionnel, s'agit d'un moyen plus avancé de protection de la vie privée, minimise les fuites d'informations, mais n'exclut pas complètement la possibilité d'utiliser des informations personnelles. À savoir, les IPI peuvent être observables sans permettre la reconnaissance personnelle, par exemple afin que les interactions des utilisateurs soient personnalisées. En conséquence, nous pouvons appliquer des pseudonymes initialement non liés (par exemple, des adresses IP hachées) pour maintenir des relations dans des systèmes adaptables à l'utilisateur. L'anonymat inconditionnel implique également une traçabilité totale.

2.3 Approches d'anonymats

Il existe quatre approches de l'anonymat [6] représenté sur cette figure :

a. Anonymat des données (data anonymity)
b. Anonymat de la communication (communication anonymity)
c. Non-liaison (Unlinkability)
d. Anonymat des utilisateurs (user anonymity)

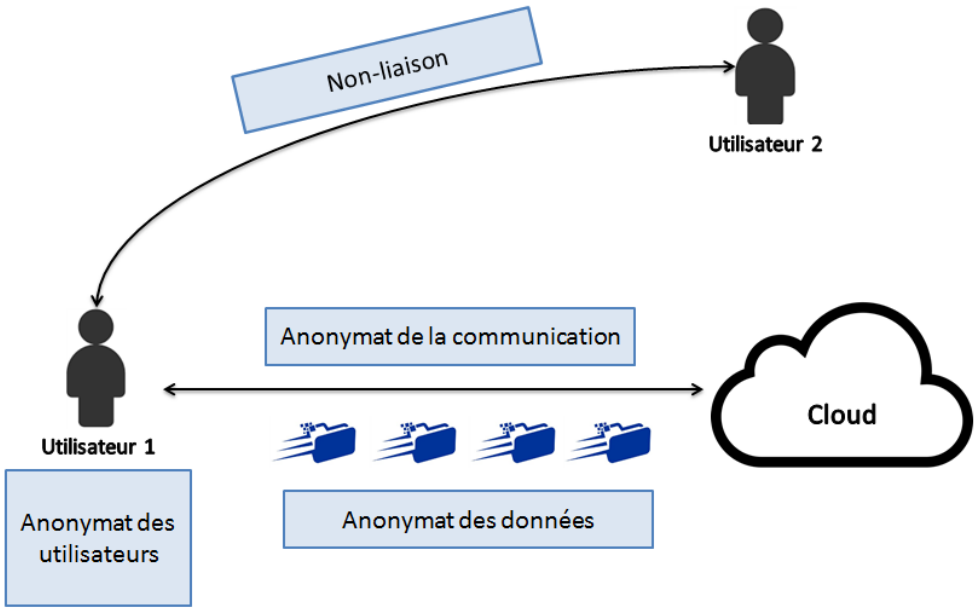

Figure 20 : Approches d'anonymat

2.3.1 Anonymat des données

Les données anonymes se rapportent pour ne pas être identifiée, que ce soit par la société qui les traitent ou par une autre personne. Les données ne peuvent être considérées comme anonymes que si la ré-identification est impossible, ce qui signifie que la ré-identification d'un individu doit être impossible par toute partie et par tous les moyens susceptibles d'être raisonnablement utilisés pour cette tentative.

Par exemple, dans un système de santé, au lieu d'utiliser la propre identité d'une personne, les patients ou les médecins utilisent des pseudonymes pour interagir avec l'un l'autre. Dans ces systèmes, personne ne devrait être capable d'établir une relation entre les attributs et l'identité des patients ou des médecins.

Les méthodes Preuves à zéro connaissance (Zero-knowledge proofs) [32] sont utilisées pour fournir l'anonymat des données. Avec ces méthodes, il est possible de prouver qu'une affirmation est vraie sans donner aucune information supplémentaire.

2.3.2 Anonymat de la communication

L'anonymat de la communication, assure la confidentialité en masquant le lien entre l'utilisateur et le système. Dans certaines solutions, des pseudonymes résistants aux collisions sont utilisés pour fournir communication anonyme semblable à l'anonymat de l'utilisateur. Le routage d'oignon (Onion routing) [33] est l'une des techniques populaires

qui permet une communication anonyme. Dans ce routage, un message est chiffré à plusieurs reprises et envoyé sur le réseau.

Les données passent par plusieurs routeurs d'oignon jusqu'à ce qu'elles atteignent la destination. Chaque routeur oignons décrypte le message, détermine où envoyer ensuite et envoie les données au prochain routeur d'oignons jusqu'à ce qu'elles atteignent la destination.

2.3.3 Non liaison

La non liaison ou le déliement des liaisons (Unlinkability) est l'une des propriétés les plus importantes dans le système d'anonymat.

Il Il existe des différentes méthodes pour fournir le déliement des liaisons dans les systèmes. Dans [34], les auteurs présentent dans un système dans lequel les professionnels de la santé utilisent différents pseudonymes et différents engagements pour les attributs à chaque communication. Par conséquent, un attaquant qui observe la communication entre les professionnels de santé ne peut pas déterminer quel attribut appartient à quelle identité. Dans certaines études, la fonction de hachage[7] est calculée sur la concaténation d'identificateurs internes tels que identifiant du patient, identifiant du centre de santé et identifiant du partage. Depuis que les fonctions de hachage sont difficiles à inverti, il est difficile de trouver l'identité du patient auprès d'un attaquant ou d'un fournisseur de cloud. Cependant, ces systèmes peuvent être vulnérables aux attaques du dictionnaire contre les identificateurs du patient et du prestataire de soins de santé.

2.3.4 Anonymat des utilisateurs

L'anonymat des utilisateurs fournit si le message de l'utilisateur ne révèle aucune information sur son identité. Dans un système de santé, l'identité des patients doit être protégée. Par exemple, il peut y avoir des recherches sur les statistiques des patients atteints de cancer. En rassemblant ces statistiques, l'identité des patients doit être anonyme. Certaines solutions étaient proposées telles que :

[7] C'est une fonction mathématique à sens unique qui permet de chiffrer un message dont son déchiffrement est impossible.

i. Pseudo-anonymat

Dans cette technique, il y a un tiers de confiance TTP (Trusted Third Party) qui accède aux données du patient et remplace son identifiant par une valeur qu'on ne peut tracer afin de trouver l'identité du patient. Par exemple, la valeur donnée au patient peut être le chiffrement symétrique de l'identifiant du patient. Si la clé est gardée secret, il est impossible d'inverser le pseudonyme en identifiant du patient. Dans certaines solutions, le hachage est utilisé sur l'identifiant pour créer des pseudo-identifiants des patients.

ii. **PIPE** (Pseudonymization of Information for Privacy in Ehealth)

Cette méthode masque le lien entre les données de l'hôpital et l'identité du patient. Quand les données sont stockées, elles sont divisées en deux parties, qui sont les données personnelles du patient et les données de l'hôpital pseudonymisées. Le patient contrôle et a l'accès complet sur ses données. PIPE comprend trois couches. Aux niveaux externe et intermédiaire, il existe des paires de clés asymétriques utilisées pour les utilisateurs authentifiés pour déchiffrer les données. Le troisième niveau (niveau caché) comprend les pseudonymes qui sont attribués aux données du patient pour l'utilisation secondaire afin de masquer le lien entre les informations personnelles et les données de l'hôpital.

2.4 Modèles d'anonymisation

Il existe cinq types de modèles d'anonymisation [35] : la pseudonymisation, le k-anonymat, la l-diversité, la t-proximité et la confidentialité différentielle (differential privacy).

2.4.1 La pseudonymisation

Elle consiste à supprimer les champs directement identifiants des enregistrements, et à rajouter à chaque enregistrement un nouveau champ, appelé pseudonyme, dont la caractéristique est qu'il doit rendre impossible tout lien entre cette nouvelle valeur et la personne. Une fonction de hachage est utilisée pour créer les pseudonymes. Elle est appliquée à l'un des champs identifiants (par exemple le numéro de sécurité sociale), qui est un type de fonction particulier qui rend impossible le fait de déduire la valeur initiale.

L'avantage de la pseudonymisation est qu'il n'y a aucune limite sur le traitement subséquent des données.

Un exemple de calcul de la moyenne d'âge pour une pathologie donnée montré dans la figure 21

Pseudonyme

Donnée sensible

ID	Age	CP	Sexe	Pathologie
1	75	75005	F	Cancer
2	40	75012	F	Grippe
3	12	78000	M	Grippe

Pathologie	MOY(age)
Cancer	75
Grippe	26

Figure 21 : Pseudonymisation et exemple de calcul [35]

Toutefois, la pseudonymisation n'est pas reconnue comme un moyen d'anonymisation, car elle ne donne pas un niveau de protection suffisamment élevé : la combinaison d'autres champs peut permettre de retrouver l'individu concerné. Sweeney l'a mis en évidence aux Etats-Unis en 2001 en croisant deux bases de données : une base de données médicale pseudonymisée et une liste électorale avec des données nominatives. Le croisement a été effectué sur un triplé de valeurs : code postal, date de naissance et sexe, qui est unique pour environ 80% de la population des Etats-Unis. Elle a ainsi pu relier des données médicales à des individus. Ce type d'attaque est appelé record linkage [36].

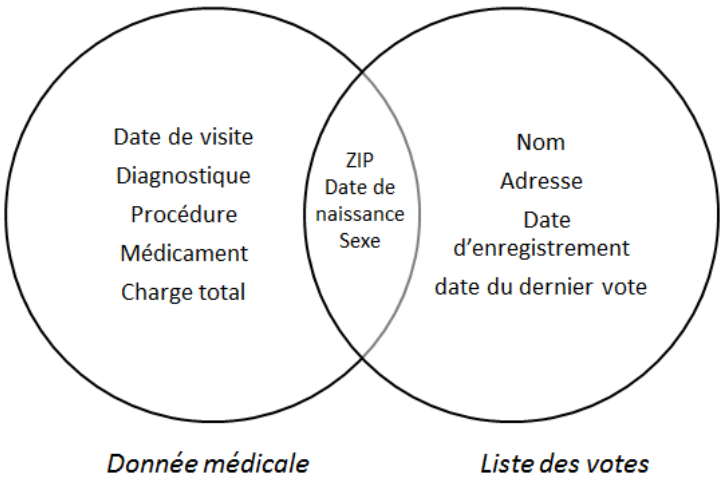

Figure 22 : exemple de recoupement d'une base anonyme [35]

2.4.2 k-anonymat

Afin de se protéger contre l'attaque couplage d'enregistrement[8] (record linkage), Sweeney [37] a proposé la technique de *k*-anonymat. Elle empêche de lier un *n*-uplet anonyme à un *n*-uplet non anonyme de la manière suivante :

1) détermine les ensembles d'attributs (appelés *quasi*identifiants) qui peuvent être utilisés pour croiser les données anonymes avec des données identifiants.

2) réduit le niveau de détail des données de telle sorte qu'il y a au moins « *k* » *n*-uplets différents qui ont la même valeur de *quasi*-identifiant, une fois celui-ci généralisé (ce terme signifie : enlever un degré de précision à certains champs).

Ainsi, il est impossible d'être sûr à plus d'une chance sur *k* qu'on a bien lié un individu donné avec son *n*-uplet anonyme.

L'avantage du *k*-anonymat est que l'analyse des données continue de fournir des résultats exacts, sans dissocier les individus d'un groupe.

La figure 23 représente un exemple de généralisation des champs activité et âge d'une base de données médicale sur des étudiants et enseignants d'une université. Les étudiants sont identifiés par leur niveau d'étude (L3, M1, etc.), qui se généralise en « étudiant », et les enseignants par leur position académique (doctorant, maître de conférences, etc.), qui se généralise en « enseignant ».

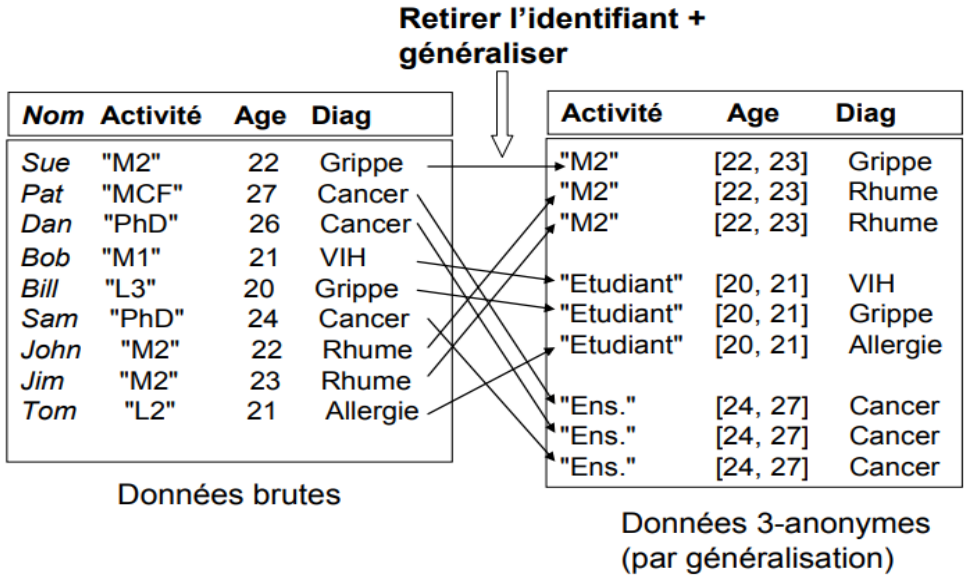

Figure 23 :anonymisation d'une table sur des données universiaitres [35]

[8] Une attaque par couplage d'enregistrement a pour but de tente de ré-identifier des individus dans un ensemble de données anonymisées en combinant ces données avec un autre ensemble de données.

Toutefois, une certaine quantité d'information sera déjà dévoilée, en particulier de l'information négative : si le quasi-identifiant d'une personne est connus, tout l'ensemble de ces valeurs sera dévoilé, sinon avoir plus de grandes chances afin de savoir une certaine valeur sensible.

Certains cas peuvent aussi apparaître : si tous les individus d'une classe d'équivalence possèdent les mêmes valeurs sur un champ intéressant l'attaquant, alors ce dernier sera capable d'identifier cette valeur.

Par exemple, un attaquant peut déduire les informations d'un enseignant sachant qu'il connait que celui-ci ayant un âge entre 24 et 27 ans a forcément le cancer.

Pour la réalisation du k-anonymat il faut être capable de déterminer les généralisations à effectuer pour produire les quasi-identifiants, ce qui fait soit par un expert qui connaît le domaine, ou bien par un calcul informatique.

2.4.3 La l-diversité

Le modèle de la l-diversité répond au problème du k-anonymat, la solution est en rajoutant une contrainte supplémentaire sur les classes d'équivalence : non seulement au moins

« *k* » *n*-uplets doivent apparaître dans une classe d'équivalence, mais en plus le champ sensible associé à la classe d'équivalence doit prendre au moins « l » valeurs distinctes. Dans l'exemple suivant, pour la constitution de telles classes il faut regrouper l'ensemble des étudiants et des enseignants. Leur activité est alors désignée de façon encore plus générale (« université ») ou d'autres valeurs possibles, par exemple avoir une modalité « Étudiant ou Doctorant » (Etu/PhD).

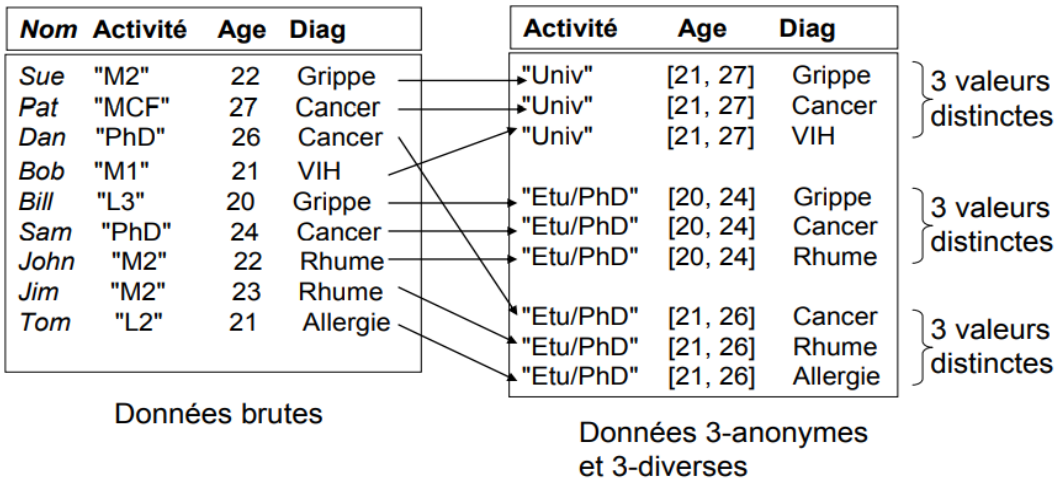

Figure 24 :exemple la I-diversité [35]

Cependant, en menant une attaque par croisement du même type que celle de Sweeney, il reste possible de déduire des informations. Exemple dans la Figure 24 un attaquant peut déduire qu'un étudiant de 20 ans aura une probabilité 0.33 (soit 1/*k*) d'avoir la grippe, 0.33 d'avoir le cancer et 0.33 d'avoir un rhume etc… S'il existe seulement de première personne dans la base de donnée de cas de figure, ceci sera facile pour l'attaquant de déduire des informations sensibles à son sujet.

2.4.4 La t-proximité

Le modèle de la *t*-proximité permet de réduire encore l'information qui peut être observée directement. Toujours à partir d'un regroupement de données en classes d'équivalences selon le processus du *k*-anonymat. Ce nouveau modèle est basé sur une connaissance globale de la distribution des données sensibles, c'est-à-dire en ce cas les pathologies, pour essayer de faire coller au mieux les valeurs sensibles d'une classe d'équivalence à cette distribution, et ainsi éviter le problème de déduction d'informations soulevé par la *l*-diversité.

Age	Sexe	Département	Pathologie	Nombre d'individus
<45	M	75	Grippe	400
<45	M	75	Rhume	800
>45	M	75	Grippe	500
>45	M	75	Rhume	1000
<35	F	75	Grippe	300
<35	F	75	Rhume	600
>35	F	75	Grippe	600
>35	F	75	Rhume	1200
…				

Figure 25 : t-proximité [35]

Cependant la t-proximité a des inconvénients : l'attaquant peut exploiter des données *k*-anonymes ou même *l*-diverses pour découvrir des corrélations entre des données appartenant au quasi-identifiant et des données sensibles. Toutefois, le but de la t-proximité est de réduire au maximum ces corrélations, puisque toutes les données sensibles de chaque classe d'équivalence vont se ressembler.

2.4.5 La confidentialité différentielle (Différential Privacy)

C'est une méthode très utile dans les milieux de la recherche en informatique car contrairement aux méthodes précédentes, elle donne des garanties formelles, c'est-à-dire des preuves mathématiques, sur la possibilité de borner les informations des individus.

Cette méthode introduit un échantillonnage des vraies données (avec une probabilité a), et une génération de données fictives avec une probabilité b >> a.

Les garanties formelles permettent de quantifier le risque de ré-identification des n-uplets. En effet, lors de l'observation des données anonymes, l'information obtenu sur le fait qu'un n-uplet soit vrai ou faux est doublement bornée : l'attaquant ne pourra jamais être sûr qu'un n-uplet soit vrai avec une probabilité supérieure à a, ni qu'il soit faux avec une probabilité inférieure à b.

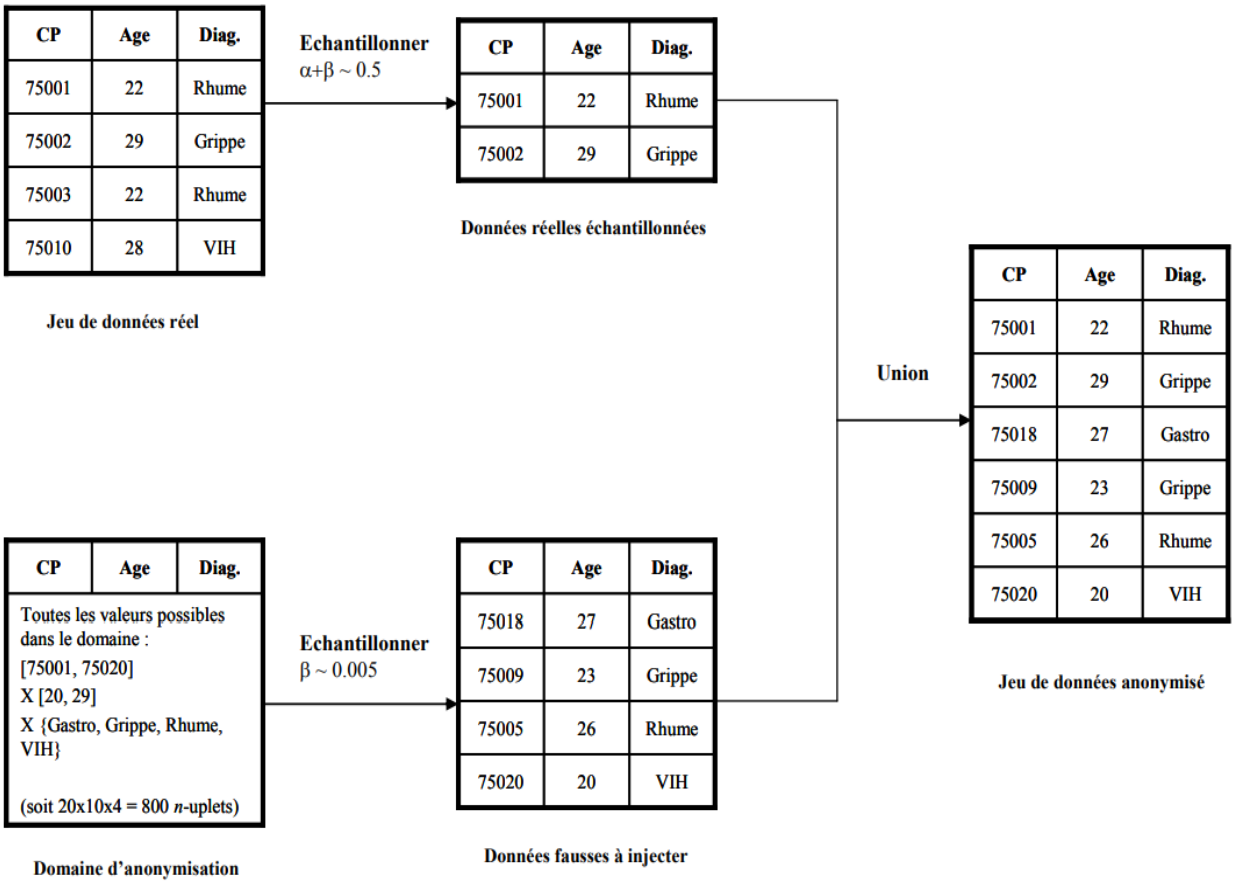

Figure 26 : confidentialité différentielle [35]

La confidentialité différentielle oblige à calculer un estimateur d'un agrégat du domaine souhaité.

L'exemple du calcul du nombre moyen de malades de la grippe par département, l'estimation du nombre total de malades de la grippe par la fonction suivante, dont l'objectif est de soustraire le bruit (connu) introduit :

$$Nb_{Rhume_{estimé}} = \frac{(Nb_{Rhume_{anonyme}} - \beta \times Nb_{Rhume_{domaine}})}{\alpha} = (2 - 200 \times 0.005)/0.5 = 2$$

Le problème principal de la mise en œuvre de la confidentialité différentielle réside dans la vraisemblance des données fictives. Ainsi, cette technique s'applique surtout lorsqu'on cherche à protéger des données de géolocalisation, où il est facile de générer des données fausses « plausibles » En revanche, d'après l'exemple cité, il parait plus difficile d'exploiter cette méthode d'anonymat sur des données médicales.

3. Authentification

L'authentification est le processus de donner l'accès aux objets du système individuellement. Elle peut être réalisée de différentes manières en se basant sur un ou plusieurs de ces facteurs :

1) les facteurs de connaissance : Quelque chose que l'utilisateur connait (c'est spécifique et secret) comme un mot de passe, une réponse à une demande d'information qu une question secrète qui peut-être d'autres ne connaissent pas.

2) les facteurs de propriété : Quelque chose que l'utilisateur possède. C'est un objet qui appartient à l'utilisateur, comme une carte à puce ou un appareil similaire. .

3) les facteurs d'inhérence : Quelque chose est un utilisateur. C'est un attribut physique comme les empreintes digitales ou la voix, qui peuvent être identifiées. .

La décision la plus cruciale dans la conception de systèmes sécurisés est l'importance de choisir une méthode d'authentification adaptée à l'environnement. Les différentes méthodes d'authentification font l'objet de la section suivante.

3.1 Méthodes d'authentification

Les méthodes d'authentification assurent que la communication de l'entité est celle revendiquée. Il existe cinq méthodes [38] comme illustrées dans la figure suivante :

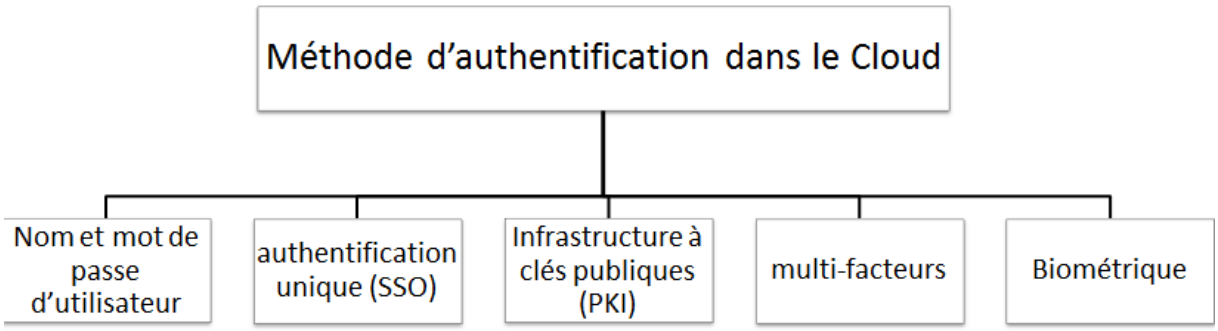

Figure 27 : méthodes d'authentification dans le cloud [39]

3.1.1 Authentification par nom d'utilisateur et mot de passe

Dans cette technique d'authentification, la confidentialité et la vie privée peuvent être maintenues jusqu'à un certain niveau. Pour accéder aux informations du CSP (cloud service provider), l'utilisateur doit entrer le nom d'utilisateur et le mot de passe dans le système. Cette technique d'authentification ne permet pas d'obtenir une sécurité plus élevée et fiable, car la plupart des utilisateurs choisissent des mots de passe très faciles à deviner. Il est possible que le meilleur mot de passe soit volé par la force brute et par le dictionnaire. Dans un environnement informatique en nuage, les contraintes de saisie empêchent les utilisateurs d'utiliser des mots de passe complexes, ce qui entraîne l'utilisation de mots de passe simples et courts. De plus, les utilisateurs réutilisent leurs mots de passe pour se reconnaître sur de nombreux serveurs, ce qui accroît les risques pour la sécurité des informations mises en commun des utilisateurs. Les mots de passe forts aident à rendre impraticables les attaques par force brute et à éviter les attaques par dictionnaire.

De plus, une fois l'utilisateur est authentifié, ces informations sont visibles dans le cloud, et cela peut représenter un risque si l'attaquant peut avoir l'accès à ce dernier et connaitre les informations de l'utilisateur. Cependant, il existe une solution afin de renforcer la sécurité, c'est en combinant cette méthode avec une l'approche anonymat.

3.1.2 Authentification unique (SSO)

L'authentification unique ou SSO (Single Sing On) est une méthode permettant d'accéder à plusieurs systèmes logiciels indépendants de manière à ce que lorsqu'un utilisateur se connecte à un système, sans être amené à se reconnecter dans chaque application, obtient l'accès à tout le système. Ce processus aide les utilisateurs à accéder à de nombreux services

et réduit la menace pour les administrateurs de diriger les utilisateurs de manière pratique. En empêchant l'utilisateur de se souvenir de nombreux mots de passe, il contribue à améliorer l'efficacité de l'utilisateur et à réduire le temps nécessaire à l'utilisateur pour saisir plusieurs mots de passe.

3.1.3 Infrastructure à clé publique (PKI)

Le schéma d'authentification traditionnel est basé sur la clé secrète et prend principalement en charge le placement d'algorithmes chiffrement asymétriques traditionnels, tels que RSA. Pour prouver l'identité de l'utilisateur, une clé privée est utilisée. Dans la conception de protocoles de sécurité tels que Secure Electronic Transaction (SET) et Secure Socket Layer (SSL / TLS), une infrastructure à clé publique (Public Key Infrastructure, PKI) a été utilisée afin d'être responsable de l'authentification. Le mécanisme PKI doit garantir l'intégrité, la confidentialité, la non-répudiation, l'authentification forte et l'autorisation des données. Les caractéristiques de sécurité de l'environnement en nuage ont été proposées et associent SSO, infrastructure à clé publique, techniques de chiffrement pour garantir l'intégrité, l'authentification et la confidentialité des données et des communications. Ainsi, ce modèle a présenté les avantages des technologies uniques et de leur combinaison. L'infrastructure à clé publique joue un rôle clé dans la sécurité et l'authentification des utilisateurs dans un environnement distribué comme celui du cloud et du réseau de capteurs sans fil.

3.1.4 Authentification biométrique

La biométrie est une approche courante pour l'authentification. Beaucoup les industries utilisent la biométrie comme mécanisme d'authentification pour l'accès aux guichets automatiques bancaires, au contrôle d'accès aux portes et à l'information générale. L'accès à un ordinateur de bureau ainsi que l'enregistrement des présences dans diverses organisations. Ces systèmes reconnaissent en fonction de leurs attributs physiques (empreintes digitales, visage, iris, voix) ou des attributs comportementaux tels que la signature.

Comme ces caractéristiques sont physiquement associées à un utilisateur particulier, la reconnaissance biométrique est un mécanisme naturel et plus efficace pour s'assurer que seuls les utilisateurs autorisés peuvent accéder à un système [2]. L'authentification biométrique s'est également avérée utile en cas de cartes d'identité multiples (passeport, carte d'électeur, etc.) pour la même personne. Ceci conduit à une plus grande sécurité dans le système. Les pièces biologiques utilisées dans ce processus donnent des résultats d'authentification différents.

La biométrie présente les avantages suivants :

1) Les mesures biométriques ne contiennent pas de renseignements personnels et sont plus difficiles à voler.

2) Des mesures biométriques peuvent être utilisées à la place d'un nom ou d'un numéro (tel que le numéro de carte bancaire) pour sécuriser les différentes transactions.

3.1.5 Authentification multifactorielle

Pour renforcer la sécurité des informations dans un environnement informatique en nuage, il est nécessaire d'utiliser une combinaison de techniques d'authentification. Ce schéma est plus sécurisé car il ne fait pas que valider la paire de nom d'utilisateur / mot de passe, il nécessite également un autre facteur, par exemple. Authentification biométrique. C'est l'une des techniques d'authentification les plus puissantes. En réalité, l'attente d'authenticité augmente de façon exponentielle lorsque des facteurs supplémentaires interviennent dans le processus de vérification.

Pour l'environnement informatique en nuage, un système d'authentification biométrique multifactorielle incluant empreintes digitales et veines palmaires a été proposé dans [6]. L'objectif est de traiter les données biométriques de manière protégée en conservant les données d'empreintes digitales dans la base de données centrale du serveur de sécurité du cloud et les données biométriques de la veine palmaire dans des cartes à puce multi-composants.

3.2 Attaques d'authentification

Le processus d'authentification est un élément crucial dans la sécurisation d'une application logicielle ou d'un système. Par conséquent, la sécurisation du processus d'authentification devient une tâche même importante et impérative. En effet, les conséquences d'une pénétration réussie dans un système d'authentification peuvent être gravement préjudiciables. Un pirate informatique peut accéder à des informations sensibles et peut supprimer, altérer ou corrompre des données importantes. En outre, le pirate informatique peut assumer l'identité d'une personne, entraînant des dommages personnels tels que le vol d'identité ou monétaire. Si l'identité piratée est celle d'un administrateur de réseau ou de serveur, les dommages sont inimaginables.

Nous allons présenter quelques types d'attaques d'authentification [40] :

a. Force brute pour les mots de passe

Dans un système logiciel client-serveur, les mots de passe constituent un point de défaillance en matière de sécurité. Les attaquants utilisent des scripts et des programmes logiciels personnalisés qui sont nourris avec des tonnes de combinaisons d'identifiants utilisateur et de mots de passe. Ces scripts sont exécutés sur le système d'authentification pour être piratés. Au début, un ensemble de combinaisons de tous les mots anglais est introduit dans le programme. Cela s'appelle une attaque par dictionnaire. Après cela, une pile d'identifiants d'utilisateur et de mot de passe très utilisés est essayée, ainsi que des valeurs numériques et des caractères spéciaux. Enfin, de grandes bases de données de références déjà utilisées par différentes personnes sont tentées de pénétrer dans le système. L'idée est d'essayer toutes les combinaisons possibles d'informations d'identification, jusqu'à ce qu'une combinaison fonctionne, car elle est acceptée par le système d'authentification. Bien que cela semble être et qu'il s'agisse d'un travail très exhaustif, avec l'amélioration de la puissance de calcul et de la bande passante du réseau, il est devenu plus facile de pénétrer les mécanismes d'authentification. Les attaquants modernes utilisent également des tables de hachage d'identification, appelées tables arc-en-ciel, pour réduire davantage le temps d'attente.

b. Écoute de session

Lorsque le processus d'authentification implique une interpellation et une réponse entre deux systèmes, les informations d'identité sont transmises par fil. Un pirate informatique peut déployer des outils de capture de paquets et attendre que la victime utilise ses informations d'identification. Pendant que le processus d'authentification est en cours, un pirate informatique intercepte les sessions et déchiffre les informations d'identification.

c. Attaque par rejeu

Dans ce type, peu différent de la prise de session, le pirate enregistre simplement les données d'une authentification réussie et lance une nouvelle demande au serveur ou au vérificateur. Parallèlement à cette nouvelle demande, le pirate informatique relit les informations enregistrées pour authentifier faussement et imiter la victime et obtenir ainsi le même ensemble de droits et d'accès que la victime.

Dans le cloud, plusieurs parties, telles que l'utilisateur, le propriétaire des données, le fournisseur de services, l'auditeur et autres, sont impliquées directement ou indirectement dans le système. Chacune de ces parties a son propre rôle dans le système. La communication

dans un environnement dynamique, de partage et multipartite exposera les informations d'identité de l'utilisateur, en particulier lorsqu'il communiquera sur un canal non sécurisé. C'est pour cette raison, les méthodes d'authentification sont combinées avec des fonctions d'anonymats en offrant une authentification anonyme comme nous allons présenter dans la section suivante.

4. Authentification anonyme

Le schéma d'authentification basé sur un mot de passe a été largement utilisé au fil des années. C'est devenu une méthode couramment utilisée pour l'authentification en raison de sa simplicité et de sa facilité d'implémentation. L'anonymat est l'une des meilleures solutions pouvant être appliquées dans le processus d'authentification pour sécuriser et préserver la confidentialité des utilisateurs. Dans cette section, nous allons présenter quelques travaux sur l'authentification anonyme.

4.1 Ticket Anonyme

Cette approche [41] permet d'assurer la confidentialité de l'identité des patients grâces à une authentification anonyme fournissant un accès anonyme dans un cloud e-santé. Elle est utilisée dans le domaine médical où la révélation d'identité d'un patient est une violation de sa vie privée, même si ses données médicales sont confidentielles (souvent chiffrées). Le but de l'approche est d'obtenir l'autorisation d'accès en fournissant des tickets anonymes pour permettre aux patients de faire des demandes anonymes sur la consommation.

Cette approche comporte trois composants principaux :

a. L'utilisateur est un patient qui utilise les applications e-santé et les services de stockage.

b. Le gestionnaire d'enregistrement (RTM) qui est responsable de l'enregistrement d'un nouvel utilisateur, il doit être en mesure de traiter avec succès l'enregistrement des différents patients. Aussi, il attribue des tickets d'accès anonyme via la signature aveugle[9] basée sur le système RSA.

[9] Est une forme de signature numérique dans laquelle le contenu d'un message est déguisé (en aveugle) avant sa signature. La signature aveugle qui en résulte peut être vérifiée publiquement par rapport au message original non aveuglé de la même manière qu'une signature numérique ordinaire. Les signatures aveugles sont généralement utilisées dans les protocoles relatifs à la protection de la vie privée où le signataire et l'auteur du message sont des parties différentes. Par exemple, les systèmes d'élection cryptographiques et les systèmes de paiement en espèces numériques.

c. Le gestionnaire de services (SM) fournit la consommation de services après avoir reçu un ticket d'accès (Access Ticket, AT) du patient, l'AT est vérifiée avec RTM avant de permettre au patient d'accéder et de démarrer la procédure de consommation de services.

Comme indique dans la figure suivante, les différentes étapes que les utilisateurs de cloud e-santé (souvent les patients) suivront pour être authentifiés dans ce système sont :

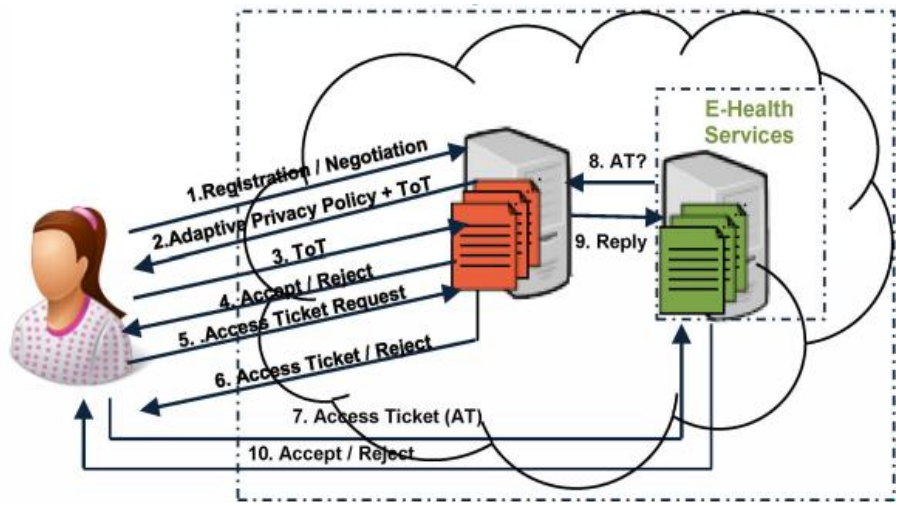

Figure 28 : aperçu du schéma d'authentification [41]

- Etape 1 Enregistrement et génération de la politique de confidentialité adaptative du patient : l'objectif de cette étape est de présenter le cloud e-santé aux utilisateurs potentiels (souvent les patients), le SLA et la politique de confidentialité à leur disposition. Ceci est fait pour les familiariser et les motiver à rejoindre ce cloud e-santé pour bénéficier de ses services, y compris être anonyme. Le résultat sera d'identifier et d'enregistrer les patients concernés et de générer la Politique de confidentialité adaptée aux exigences de chacun. À la fin de cette étape, RTM remet un ticket initial nommé ToT "Ticket of Ticket", qui représente la preuve d'inscription.
- Etape 2 Obtention d'un AT (Access Ticket) : Après l'achèvement de l'inscription, le patient obtiendra un ticket anonyme (nommé Anonymous Access Ticket) via la technique de la signature aveugle lui permettant de consommer des services de cloud ultérieurement. Ainsi chaque utilisateur consomme le service anonymement sans révéler son identité auprès du fournisseur de services cloud(CSP). Ce ticket anonyme est considéré comme l'information d'identification dans ce système d'authentification.
- Etape 3 Consommation du service e-santé souhaité : dans cette étape, le patient devra seulement présenter son ticket d'accès obtenu à l'étape 2 pour demander un service donné (stockage pour accéder à ses données, applications de santé en ligne, etc.). CSP ne sait pas

quel patient demande l'accès aux services, mais seulement qu'un patient légitime veut accéder.

L'avantage de cette approche est que même si les deux gestionnaires (RTM et SM) communiquent, la divulgation de l'identité du patient reste difficile à réaliser et cela est atteint grâce à la consommation anonyme de services via des tickets anonymes, ce qui a permis à cette approche d'être indépendant de la notion de confiance qui exige un certain niveau de confiance pour l'assurance de l'anonymat de l'utilisateur. Cependant, l'inconvénient de cette approche est que l'adresse IP de l'utilisateur est visible pour le fournisseur de services.

4.2 Authentification anonyme sans certificat

Le schéma proposé par [42], est basé sur computational Diffie-Hellman problem (CDPH)[10]. Vu qu'il n'a pas de centre de certification, ce schéma se compose de deux entités: les utilisateurs et le serveur cloud et se déroule comme suit:

a. l'utilisateur envoie une demande d'anonymat au serveur cloud; le message contient les informations d'authentification de l'utilisateur.

b. A la réception d'une demande d'anonymat, le serveur génère un nombre aléatoire (x), et l'envoie à l'utilisateur.

c. Lorsque l'utilisateur reçoit le nombre aléatoire, il démarre le "processus de résolution CDHP" en envoyant d'abord la clé publique (Public Key, PK) $PK = g^{1/x}$ au serveur cloud. Ce dernier sélectionne un nombre aléatoire (y) parmi un ensemble donné d'entier et l'envoie à l'utilisateur qui calcule ensuite la fonction $A = g^{x/y}$ et envoie le résultat "A" au serveur cloud. Le serveur vérifie si la fonction $e(A,PK^y)$ [11] retourne la valeur "I"[12] pour autoriser le l'utilisateur à continuer le processus de communication, sinon (i.e l'utilisateur est une tierce personne qui a volé les informations de l'utilisateur réel) le serveur cloud arrête le processus de communication.

d. Ensuite, le serveur cloud et l'utilisateur négocient sur la clé de session K (qui est envoyée de manière sécurisée) pour pouvoir chiffrer les informations d'authentification de l'utilisateur pendant tout le long de la session.

e. Lorsque l'utilisateur quitte le cloud, il envoie un message de fermeture au serveur cloud afin clôturer la connexion.

[10]Computational Diffie-Hellman est une hypothèse de dureté de calcul concernant le problème de Diffie-Hellman, il se base sur un problème mathématique proposé dans le contexte de la cryptographie.

[11] $E(A, PK^y)$=fonction exponentielle calculé avec les valeurs (A et PK) envoyé par l'utilisateur

[12] I = valeur d'identité de l'utilisateur que le proxy calcule

L'avantage de cette approche, elle résout la question de la préservation de la vie privée par l'authentification tout en cachant l'identifiant de l'utilisateur. Elle n'a pas de centre de certification et évite les problèmes de révocation de clé dans les schémas d'authentification basés sur un certificat à clé publique [43]. Toutefois, l'approche proposée a des limites, La plus grande faiblesse est que l'accord de la clé est partagé entre les deux parties, ce qui la rend vulnérable à une attaque de l'homme du milieu.

4.3 Discussion

Dans le tableau suivant, nous comparons entre les deux approches d'authentification anonyme présentées ci-dessus :

	Ticket Anonyme	Authentification Anonyme sans Certificat
Méthode d'authentification	Non spécifié	Non spécifié
Niveaux d'anonymat	2	2-3
Approche d'anonymat	Utilisateur Anonyme	Utilisateur Anonyme
Modèle d'anonymat	K-anonymat	K-anonymat
Moyens utilisé	-La signature aveugle pour consommer anonymement un service. -Les tickets anonymes	-Calcul mathématique DHP
Avantages	-la divulgation de l'identité du patient reste difficile à réaliser grâce à la consommation anonyme de services via des tickets anonymes.	- préserve de la vie privée par l'authentification -ne possède pas de centre de certification -évite les problèmes de révocation de clé
Inconvénients	-L'adresse IP de l'utilisateur est visible pour le fournisseur de services.	-La clé (K) qui est partagée entre l'utilisateur et le serveur est vulnérable à une attaque de l'homme du milieu.

Tableau 5 : résumé des approches d'authentification anonyme

Dans notre travail, nous optons pour une authentification anonyme sans certificat car elle est simple à mettre en œuvre; au fait, elle est définie par des formules mathématiques simples à programmer et difficiles à trouver leur inverse (revenir aux valeurs initiales), par contre, dans la méthode Ticket anonyme, cette dernière se base sur le principe du protocole des services ticket et de la signature aveugle. **De plus, nous allons utiliser une méthode d'authentification à base du nom d'utilisateur et mot de passe. La conception de notre approche sera détaillée dans le prochain chapitre.**

5. Conclusion

Dans ce chapitre, nous avons vu les différents modèles et approches de l'anonymat. Cette dernière représente un des éléments pouvant masquer les informations d'identité de l'utilisateur afin de préserver la confidentialité de l'utilisateur ou de l'organisation. Il peut protéger les informations de l'utilisateur contre les abus de la part d'un attaquant. Il peut également protéger contre les attaques malveillantes. Par la suite, nous avons vu les différentes méthodes d'authentification et quelques travaux qui utilisent une des méthodes d'authentification avec l'anonymat afin de renforcer sécurité.

Le chapitre suivant est consacré à la conception de notre solution qui consiste à utiliser une approche anonymat basé sur une méthode d'authentification et un modèle de contrôle d'accès par chiffrement comme déjà vu dans le chapitre précédent.

Chapitre IV : Conception d'un Cloud E-santé Sécurisé

1. Introduction

Dans le domaine de la santé, le cloud peut être considéré comme une plate-forme qui permet de stocker d'énormes volumes de données sur la santé (dossiers médicaux). Il sert également pour une gestion structurée des données entre les médecins et les patients. Une grande partie de données stockées dans le cloud e-santé sont très sensibles et doit être sécurisée afin de protéger la vie privée des patients et parvenir à assurer la confidentialité au sein du cloud. Dans ce chapitre, nous proposons une solution pour un système de cloud e-santé sécurisé. Nous commençons par décrire l'architecture générale de notre système. Ensuite, nous détaillons la conception de notre application.

2. Description de la Solution

Dans le système PHR (Personal Health Records), l'accès aux données stockées doit être contrôlé pour garantir la confidentialité et l'intégrité des données. Le modèle de contrôle d'accès permet de contrôler les autorisations d'accès, et les techniques de chiffrement/déchiffrement peuvent aussi renforcer le contrôle d'accès, assurer la confidentialité des données et des privilèges des utilisateurs, et sécuriser la distribution de ces derniers. De plus, les utilisateurs, notamment les patients, doivent garder leur l'identité anonyme dans le cloud pour se protéger des autres utilisateurs. La meilleure façon de protéger l'identité du patient dans le serveur cloud est l'authentification anonyme. Notre solution combine tous ces mécanismes (contrôle d'accès + chiffrement/déchiffrement + authentification anonyme) pour sécuriser un système de cloud e-santé.

A partir de notre étude de travaux existants (chapitre 2 sections 4 et chapitre 3 sections 3), notre solution englobe deux approches : le contrôle d'accès basé sur les attributs (ABAC) et sur le chiffrement CP-ABE et l'authentification anonyme sans certificat [42] qui est basé sur le problème de Diffie-Hellman (CDPH). L'application de ces deux approches dans notre système est décrite dans les sections suivantes.

2.1 Contrôle d'Accès basé sur les Attributs et sur le Chiffrement CP ABE

Dans notre système, l'utilisateur (patient ou médecin) possède un ou plusieurs attributs qui sont gérés par une autorité de confiance. Notre univers d'attributs est défini comme étant

{Allergies, Diabète, Cancer, hypertension artérielle, Médecin, professeur, Ingénieur, Retraité, Etudiant, Homme, Femme, Dénutrition, Maigreur, Corpulence normale, Surpoids, Obésité[13] }

Notre modèle se repose aussi sur l'utilisation de l'approche de chiffrement par attributs CP-ABE qui incorpore un processus de génération des clés de chiffrement et de déchiffrement et la notion de politique d'accès basée sur des attributs, offrant ainsi des fonctionnalités de chiffrement et de contrôle d'accès. En effet, l'autorité de confiance définit les attributs pour chaque utilisateur et génère les clés secrètes qui sont une combinaison d'un ensemble d'attributs. Les propriétaires de données (médecins) définissent des politiques d'accès pour le chiffrement. Seuls les utilisateurs avec des attributs qui satisfont une politique d'accès aux données chiffrées peuvent déchiffrer ces données. Notre version de l'algorithme de CP-ABE est illustrée dans la figure suivante :

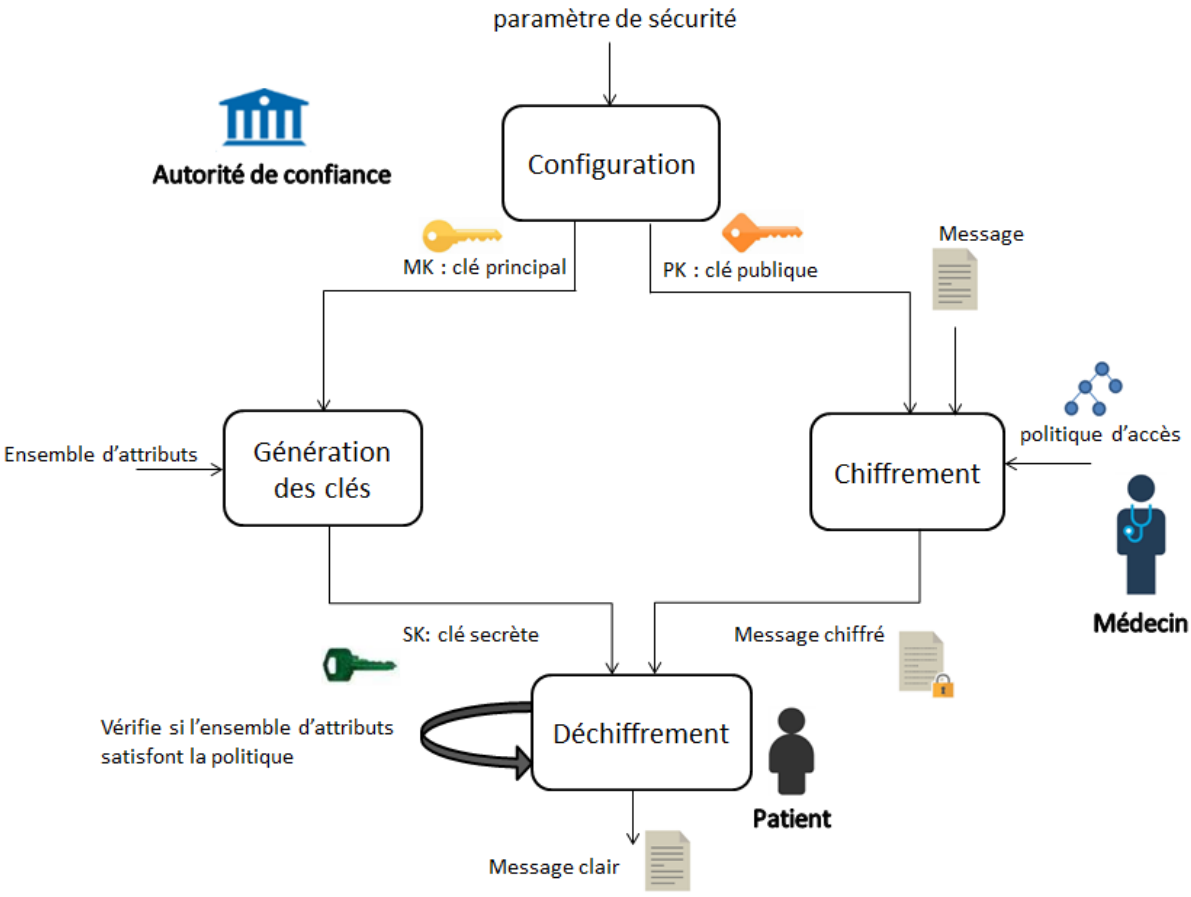

Figure 29 : algorithme CP ABE

Afin de mieux comprendre, nous allons la donner un exemple ; supposons que le patient 1 reçoit une clé pour les attributs {Retraité, Diabète} et que le patient 2 reçoit une clé pour les

[13] Les derniers attributs (Dénutrition jusqu'à Obésité) sont déterminés à partir de l'Indice de Masse Corporelle (IMC) qui est un indice pondéral calculé en divisant le poids d'une personne par le carré de sa taille.

attributs {Retraité, Cancer}. Si le médecin chiffre un fichier et définit la politique étant {Retraité ∧ (Diabète v hypertensions artérielle)} alors le patient 1 pourra alors déchiffrer, tandis que le patient 2 ne pourra pas déchiffrer.

Enfin, une comparaison de notre modèle avec les modèles existants est donnée dans le tableau suivant :

	DACC [28]	TAAC [29]	Mécanisme basé sur les provenances [30]	**Notre Modèle**
Modèle de contrôle d'accès	ABAC	ABAC	ABAC + implémentation PEP/PDP	**ABAC**
Gérant des attributs	Propriétaire des données	Autorité des attributs	Autorités des provenances	**Propriétaire des données**
Technique de chiffrement	ABE	CP-ABE	Aucune	**CP-ABE**
Composants du système	(1) le propriétaire des données (2) les utilisateurs (3) les centres de distribution des clés (KDC) (4) les serveurs Cloud.	(1) le prioritaire des données (2) l'utilisateur (3) le serveur Cloud (4) les autorités des attributs.	(1) l'utilisateur (2) le serveur Cloud (3) les autorités des provenances	**(1) le propriétaire des données (médecin), (2) les utilisateurs (patients), (3) le serveur Cloud, (4) l'autorité de confiance**
Architecture distribuée	Multi KDCs	Multi Autorité des attributs	Multi Autorités des provenances	**Non**
Révocation des utilisateurs	Oui avec l'opération de calcule	Oui sans opération de calcule	non	**Non**
Autres mécanismes de sécurité	Protocole SSH pour distribuer les clés privées et les données chiffrées.	Contrôle d'accès **temporel** : la structure d'accès et l'intervalle de temps sont associés avec le texte chiffré.	Non spécifié	**Protocole SSH pour distribution des clés (publiques et secrètes)**
Domaine d'application	e-santé	Non spécifié	Non spécifié	**e-santé (PHR)**

Tableau 6 : Comparaison de notre modèle avec les modèles existants

2.2 Authentification anonyme sans certificat des utilisateurs

Dans notre système, l'utilisateur (médecin ou patient) se connecte par la méthode d'authentification à base de nom d'utilisateur (pseudo) et de mot de passe. L'anonymat se produit lors de l'enregistrement des données chiffrés dans le cloud par le médecin (i.e: les autres utilisateurs ne peuvent pas connaître l'identité du médecin qui a stocké ces données) et lors de téléchargement des données chiffrés par le patient (i.e: les autres utilisateurs ne peuvent pas connaître l'identité du patient qui a téléchargé ces données). Le processus d'anonymat se déroule entre l'utilisateur et un proxy[14] dédié (que nous appelle proxy d'anonymat) en passant par les phases suivantes (figure 30):

1. Phase d'Initialisation: l'utilisateur (patient ou médecin) envoie une demande d'anonymat au proxy d'anonymat contenant son nom d'utilisateur. A la réception d'une demande d'anonymat, le serveur génère un nombre aléatoire (x), et l'envoie à l'utilisateur.

2. Phase de résolution de CDHP: Lorsque l'utilisateur reçoit le nombre aléatoire, le processus de résolution CDHP s'exécute (voir la section 3.2 dans le chapitre 3) dans le but de vérifier l'authenticité de l'utilisateur et continuer ainsi le processus de communication.
3. Phase d'anonymisation de pseudo: le serveur proxy et l'utilisateur se met en accord sur une clé K comme suit :
   a. Le proxy choisit un nombre premier "p" et une base "g" et l'envois a l'utilisateur.
   b. Le proxy choisit un entier secret "a" puis calcule C = $g^{\mathbf{a}}$ mod p.
   c. L'utilisateur choisit un entier secret "b" puis calcule B = $g^{\mathbf{b}}$ mod p.
   d. L'utilisateur et le proxy s'envoient les valeurs calculées C et B
   e. Le proxy calcule la clé K = $B^{\mathbf{a}}$ mod p.
   f. L'utilisateur calcule la même clé K = $C^{\mathbf{b}}$ mod p.
4. Phase de transfert de données : L'utilisateur accède au cloud avec son nouveau nom d'utilisateur (chiffré) afin de stocker ou télécharger les données dans le cloud
5. Phase de terminaison: L'utilisateur envoie un message pour quitter la session.

[14]Un serveur proxy vérifie et transmet les demandes des clients entrants à d'autres serveurs pour une communication supplémentaire. Un serveur proxy est situé entre un client et un serveur où il agit comme intermédiaire entre les deux, comme un navigateur Web et un serveur Web.

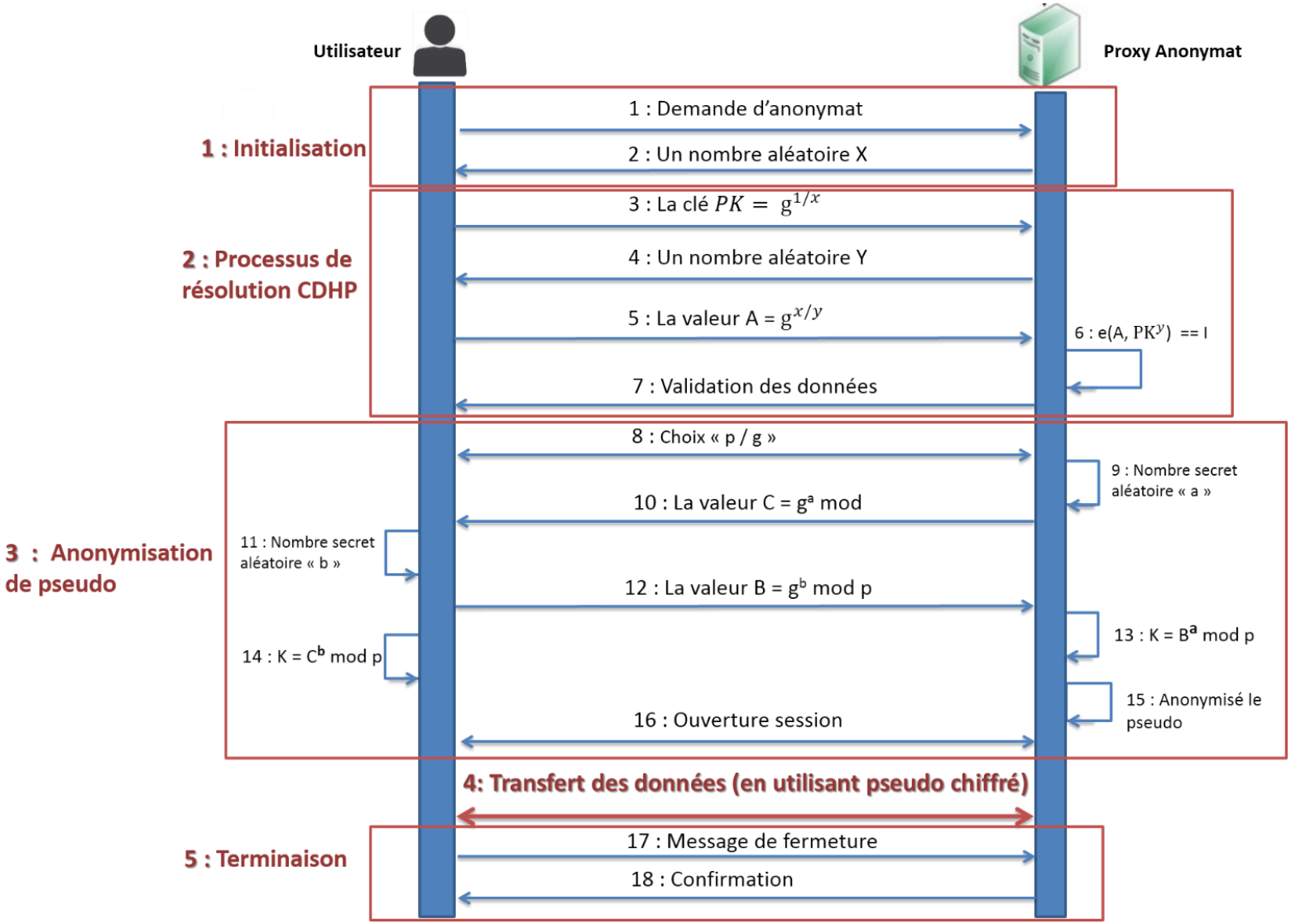

Figure 30 : Schéma d'authentification anonyme sans certificat

2.3 Architecture Générale

En résumé, notre système se compose de cinq éléments (Figure 31) : Cloud, Médecin (propriétaire de données), Patient (utilisateur), Autorité de confiance et proxy anonymat.

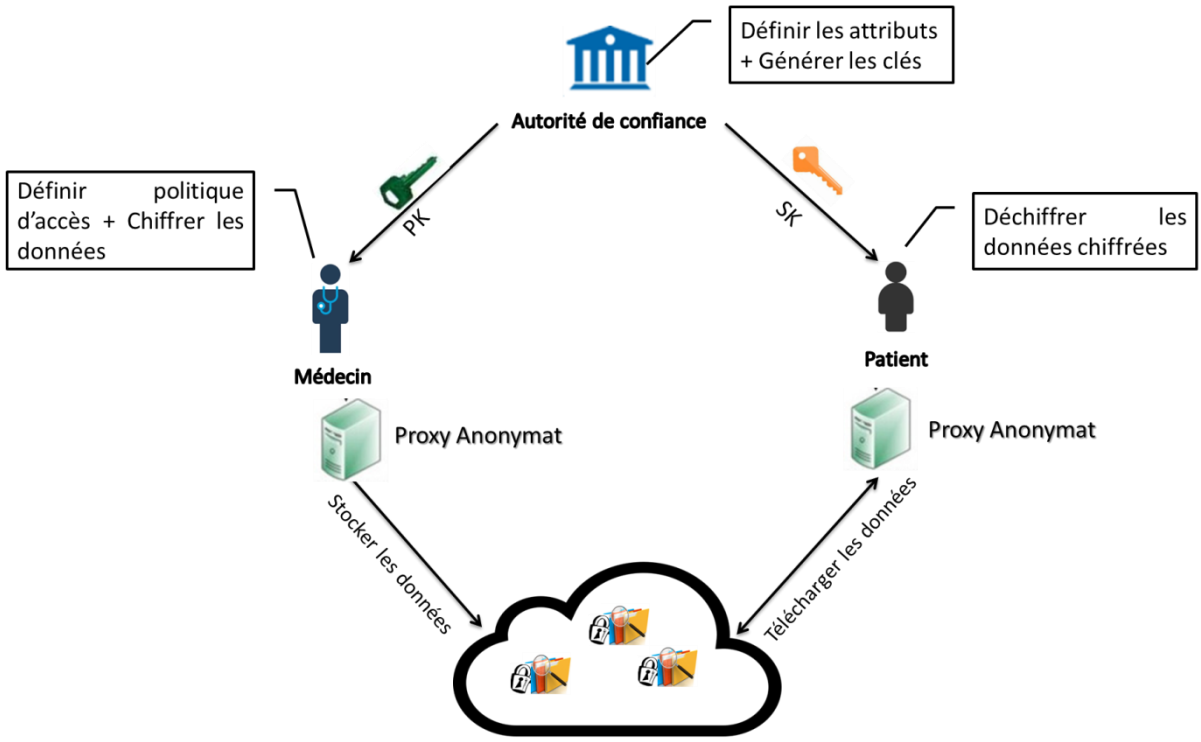

Figure 31 :architecture du système

Chaque composant de notre système possède un rôle précis dans le système :

- Cloud : C'est une entité qui fournit un service de stockage de données. Il est chargé de contrôler les accès des utilisateurs extérieurs aux données stockées. Nous supposons que le Cloud est honnête mais curieux. C'est-à-dire qu'il exécutera honnêtement les tâches assignées dans le système ; cependant, il aimerait apprendre le plus d'informations possible sur les contenus chiffrés.

- Médecin (propriétaire des données) : Il s'agit d'un individu qui possède des données et qui souhaite les stocker dans le cloud externe pour faciliter le partage ou pour réduire les coûts. Le médecin est responsable de définir une politique d'accès (basée sur les attributs) et de l'appliquer à ses propres données qui souhaitent les chiffrer.

- Patient: C'est une entité qui veut accéder aux données. Elle peut seulement consulter les données du cloud. Si un patient possède un ensemble d'attributs satisfaisant à la politique d'accès des données chiffrées, il pourra déchiffrer les données chiffrées par son médecin et obtenir les informations.
- Autorité de confiance : C'est une entité qui génère les clés publiques, principales et secrètes pour l'algorithme de chiffrement/déchiffrement CP-ABE. Il est chargé de définir et d'émettre les principaux d'attributs des utilisateurs. En fonction des attributs, il génère les clés secrètes (SK) de déchiffrement.
- Proxy d'anonymat: c'est une entité qui permet de rendre l'utilisateur anonyme dans le cloud lors de stockage ou téléchargement des données chiffrées.

3. Etude conceptuelle de notre application

Pour mettre en place notre solution, nous allons développer une application qui doit satisfaire les besoins fonctionnels suivants :

- Chiffrement/Déchiffrement à base d'attribut (CP-ABE) des données.
- Stockage des données dans le cloud.
- Téléchargement des données du cloud.
- Partage des données entre les utilisateurs
- Authentification anonyme sans certificat des utilisateurs.

3.1 Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation est destiné à représenter les besoins des utilisateurs par rapport au système. Dans notre solution, nous distinguons trois acteurs :

- médecin : un individu jouant un rôle au sein de l'hôpital et accède aux dossiers médicaux de ses patients
- patient : un individu qui accède à son dossier médical.
- autorité de Confiance (AC) : est responsable de la gestion des attributs des utilisateurs et des clés

Ces acteurs peuvent établir plusieurs fonctionnalités comme illustré par les diagrammes de cas d'utilisations suivants :

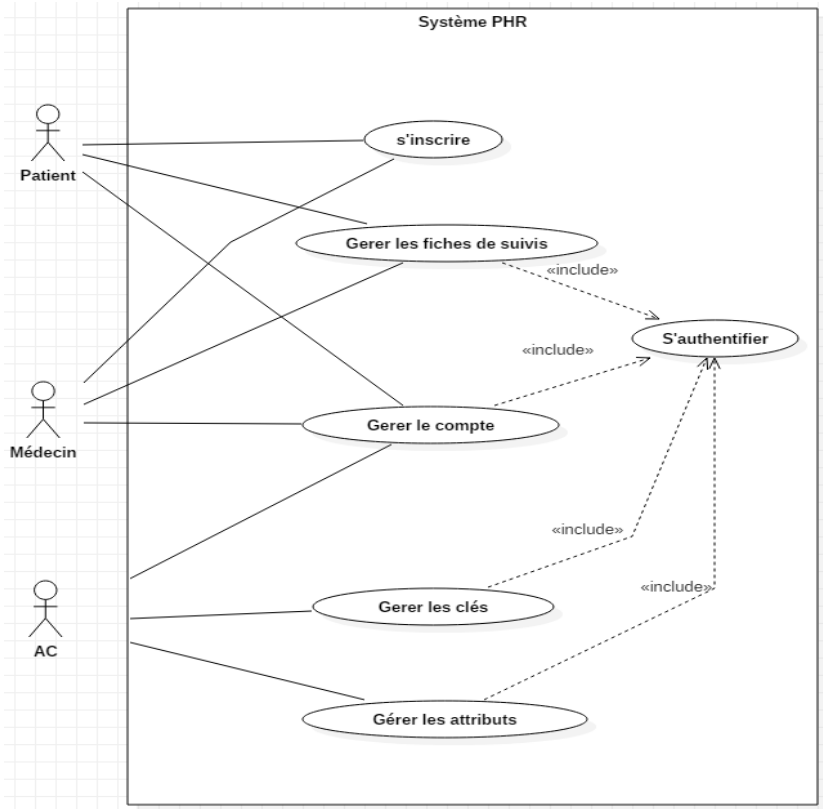

Figure 32 : diagramme de cas d'utilisation globale

Cas d'utilisation	Acteurs	Description
S'inscrire	Patient, Médecin	Le patient et le médecin s'inscrivent au système à partir d'un formulaire d'inscription.
S'authentifier	Patient, Médecin, AC	L'authentification est obligatoire pour pouvoir accéder à l'espace de l'application
Gérer le compte	Patient, Médecin, AC	Les utilisateurs peuvent consulter et modifier les informations de leur compte et se déconnecter.
Gérer les fiches de suivis	Patient, Médecin	Le médecin peut rechercher, ajouter, modifier, supprimer une fiche de suivis. Le patient peut seulement consulter les fiche de suivis
Gérer les clés	AC	générer les clés de chiffrement et déchiffrement puis les envoyer aux utilisateurs.
Gérer les attributs	AC	L'administrateur peut consulter le profil des utilisateurs et définir leurs attributs à partir de l'univers des attributs

Tableau 7 : Descriptions des cas d'utilisation du diagramme globale

Dans ce qui suit, nous allons détailler les fonctionnalités suivantes : gérer les fiche de suivis (figure 33 avec tableau 8) et gérer les clés (figure 34 avec tableau 9)

3.1.1 Gérer les fiches de suivis

Elle permet au patient et au médecin de gérer une fiche de suivis, sachant que cette dernière contient les informations personnelles du patient ainsi que les fiches historiques de ses maladies

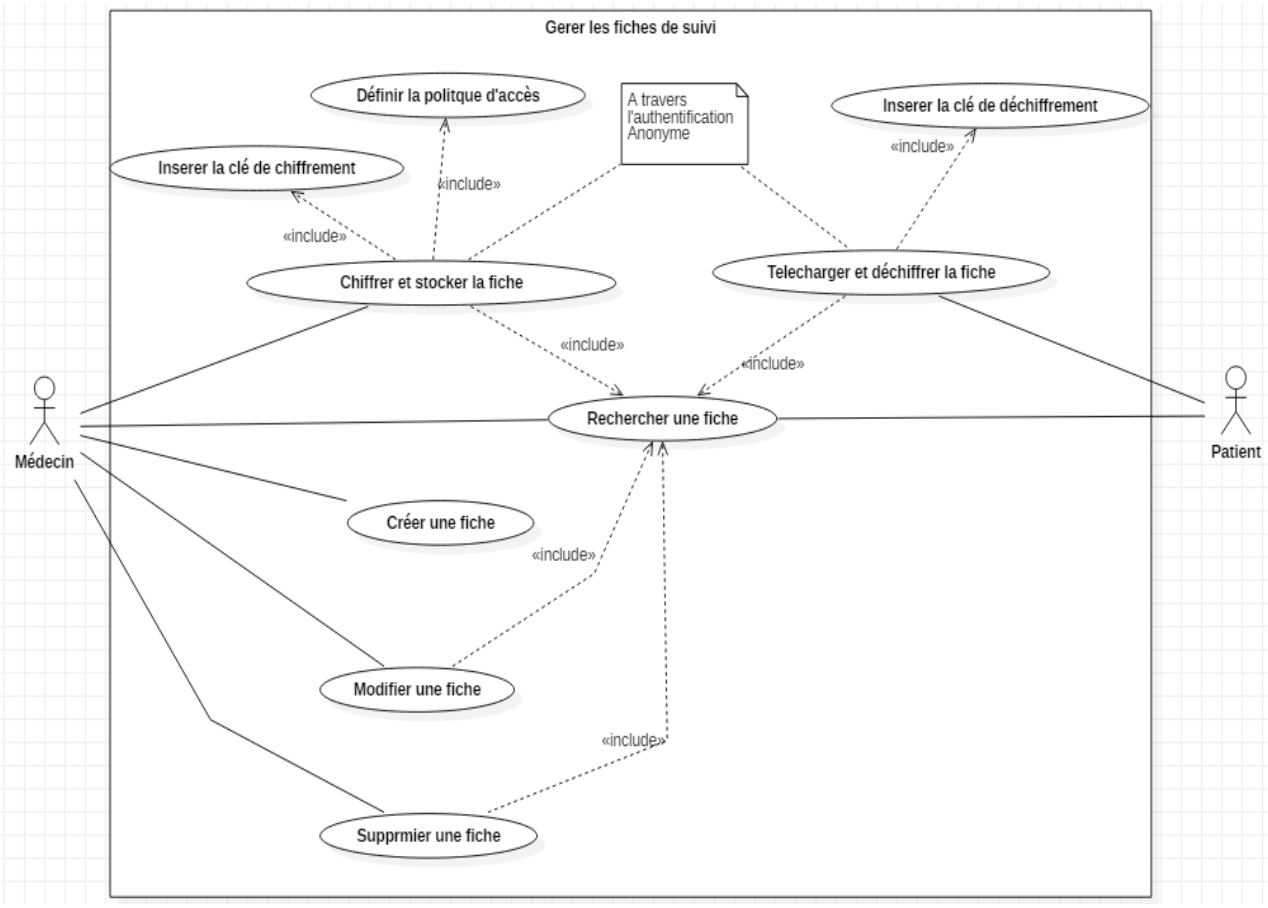

Figure 33 : Diagramme de cas d'utilisation Gerer une fiche de suivis

Cas d'utilisation	Acteurs	Description
Créer une fiche	Médecin	Créer une fiche de suivi pour un patient
Chiffrer et stocker la fiche	Médecin	Définir d'abord la politique d'accès en fonction des attributs utilisateurs, ensuite insérer la clé publique pour chiffrer la fiche du suivi et enfin la stocker dans le cloud de façon anonyme
Rechercher la fiche	Médecin, Patient	Rechercher pour consulter la fiche de suivi
Modifier une fiche	Médecin	Modifier les informations contenues dans une fiche
Supprimer une fiche	Médecin	Supprimer une fiche
Télécharger et déchiffrer la fiche	Patient	Insérer la clé secrète, ensuite il déchiffrer la fiche après l'avoir téléchargé du cloud de manière anonyme

Tableau 8 : Descriptions des cas d'utilisation du diagramme Gerer de fiche de suivis

3.1.2 Gérer les clés

Elle permet à l'administrateur de confiance de définir les clés de chiffrement (PK) et déchiffrement (SK)

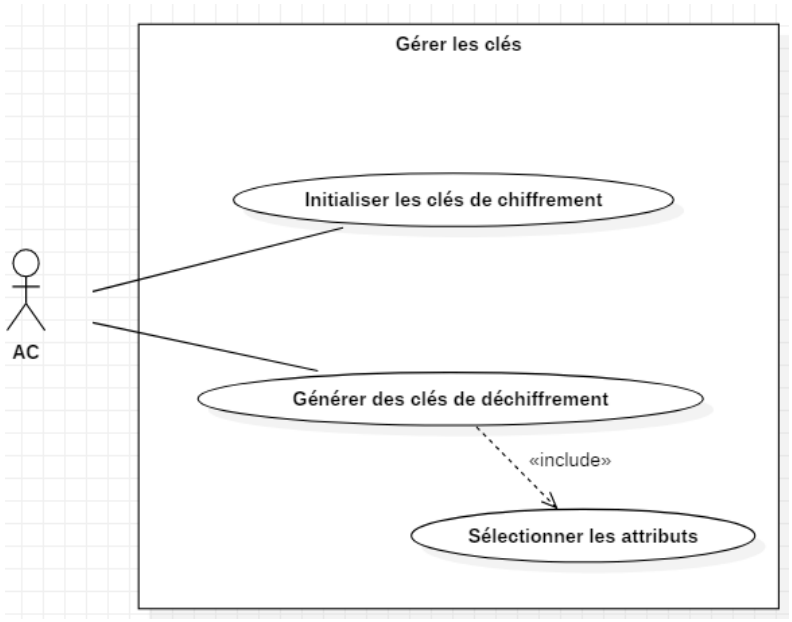

Figure 34: Diagramme cas d'utilisation Gerer des clés

Cas d'utilisation	Acteurs	Description
Initialiser les clés de chiffrement	Autorité confiance	Définir les deux clés: PK (clé publique) pour le chiffrement et MK (clé principale) pour la génération de clé de déchiffrement
Générer les clés de déchiffrement	Autorité confiance	Génération de la clé SK (clé secrète) pour le déchiffrement après avoir sélectionné un ensemble d'attributs de l'univers qui correspondent à l'utilisateur choisi

Tableau 9: Descriptions des cas d'utilisation du diagramme Gerer les clés

3.2 Diagrammes de séquence

Nous allons décrire le fonctionnement de notre système à l'aide du diagramme de séquence qui expose en détail la façon dont les opérations sont effectuées. Nous illustrons les diagrammes de séquence suivants : Inscription (figure 35), Authentification (figure 36), Génération des clés (figure 37) et Chiffrement et Stockage (figure 38) et Téléchargement et Déchiffrement (figure 39)

.

3.2.1 Inscription

Les utilisateurs (médecin, patients) doivent s'inscrire pour pouvoir accéder à notre application. Pour cela, ils doivent remplir un formulaire qui contient leurs informations du compte (pseudo, Mot de passe, Email), leurs informations d'identité (Nom de l'utilisateur, Prénom de l'utilisateur, Date de naissance, Sexe, Profession, Adresse, Numéro de téléphone, région, poids, taille) et leurs antécédents médicaux (Maladie chronique).

Le diagramme de séquence suivant illustre les interactions décrites pour l'inscription d'un utilisateur :

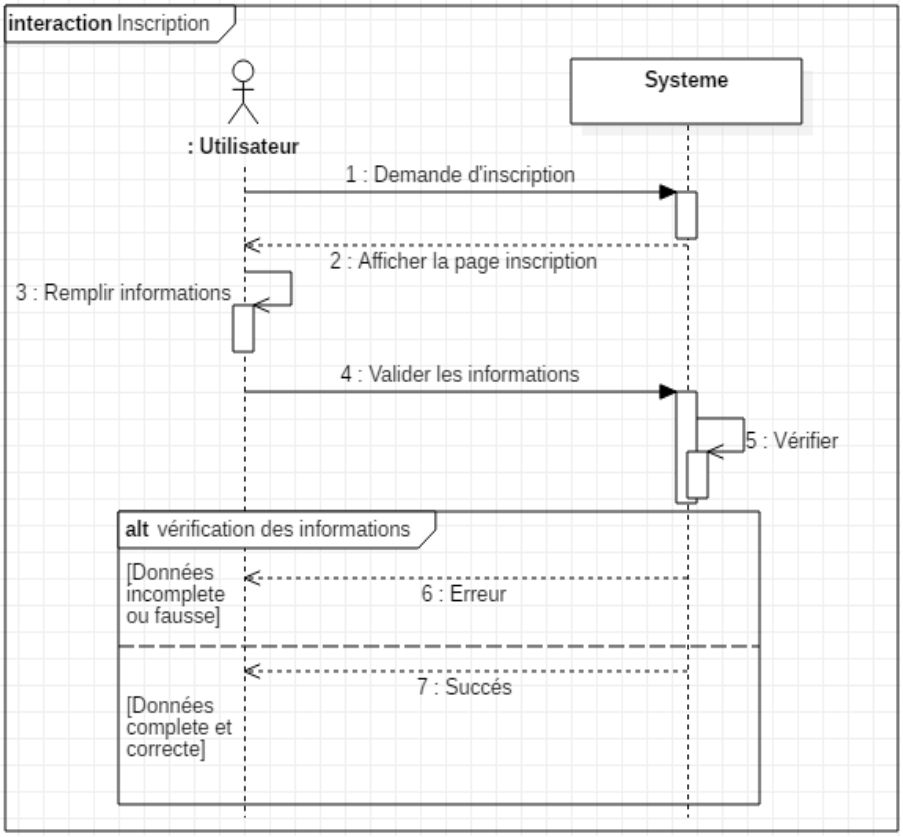

Figure 35 : diagramme de séquence ‘’Inscription’’

3.2.2 Authentification

Les utilisateurs (médecin, patients et administrateur) doivent s'authentifier avant d'accéder à l'application en entrant leur pseudo et mot de passe.

Le diagramme de séquence suivant illustre ces étapes :

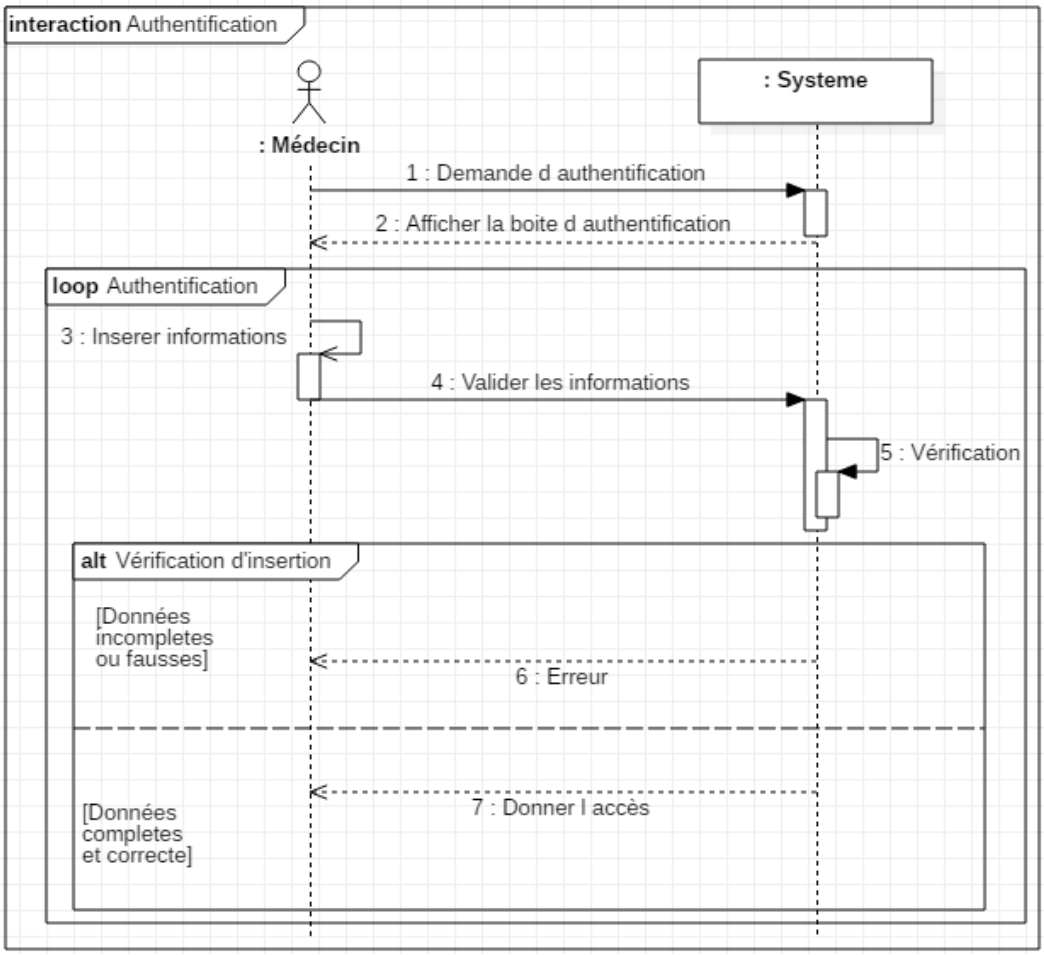

Figure 36 : diagramme de séquence ''Authentification''

3.2.3 Gestion des clés

L'administrateur établit deux étapes (figure 37):

a. Initialisation des deux clé (publique PK, principale MK), La clé publique sera envoyée à l'utilisateur (médecin) pour que ce dernier puissent chiffrer.

b. Génération des clés secrètes SK, pour cela l'administrateur doit avant tout accéder aux profiles des utilisateurs afin de déduire leur attributs. Ensuite il génère la clé SK pour chaque utilisateur (médecin, patient) à l'aide de la clé principal MK et l'ensemble d'attributs de l'utilisateur. Enfin, il envoie cette dernière à l'utilisateur pour qu'il puisse déchiffrer.

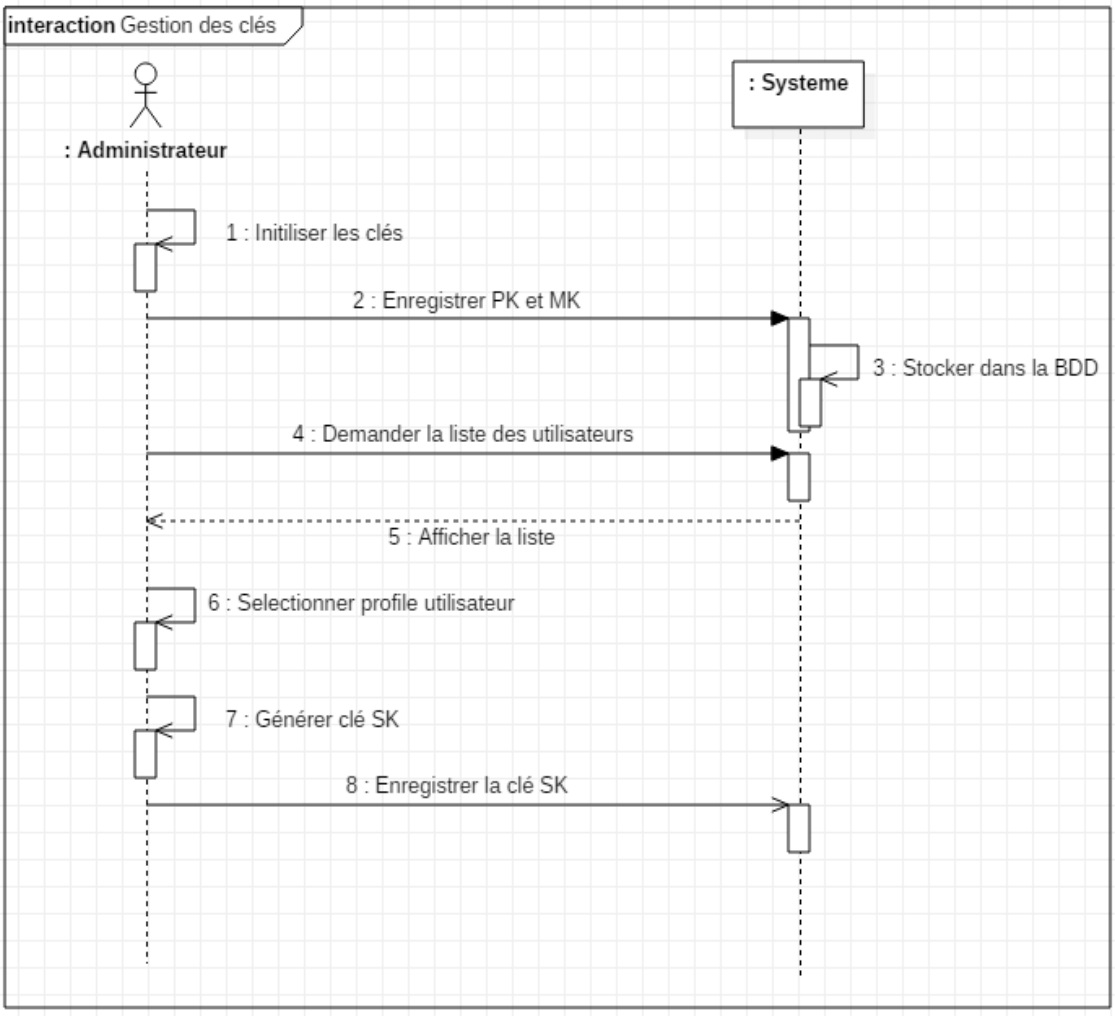

Figure 37 : diagramme de séquence ''Gestion des clés''

3.2.4 Chiffrement et Stockage

Comme déjà décrit précédemment, le médecin définit la politique d'accès. Ensuite, le système exécute l'algorithme de chiffrement en utilisant la fiche, la politique et la clé PK. Enfin, Il accède au cloud de manière anonyme pour stocker la fiche chiffrée. Le diagramme de séquence suivant illustre ces étapes :

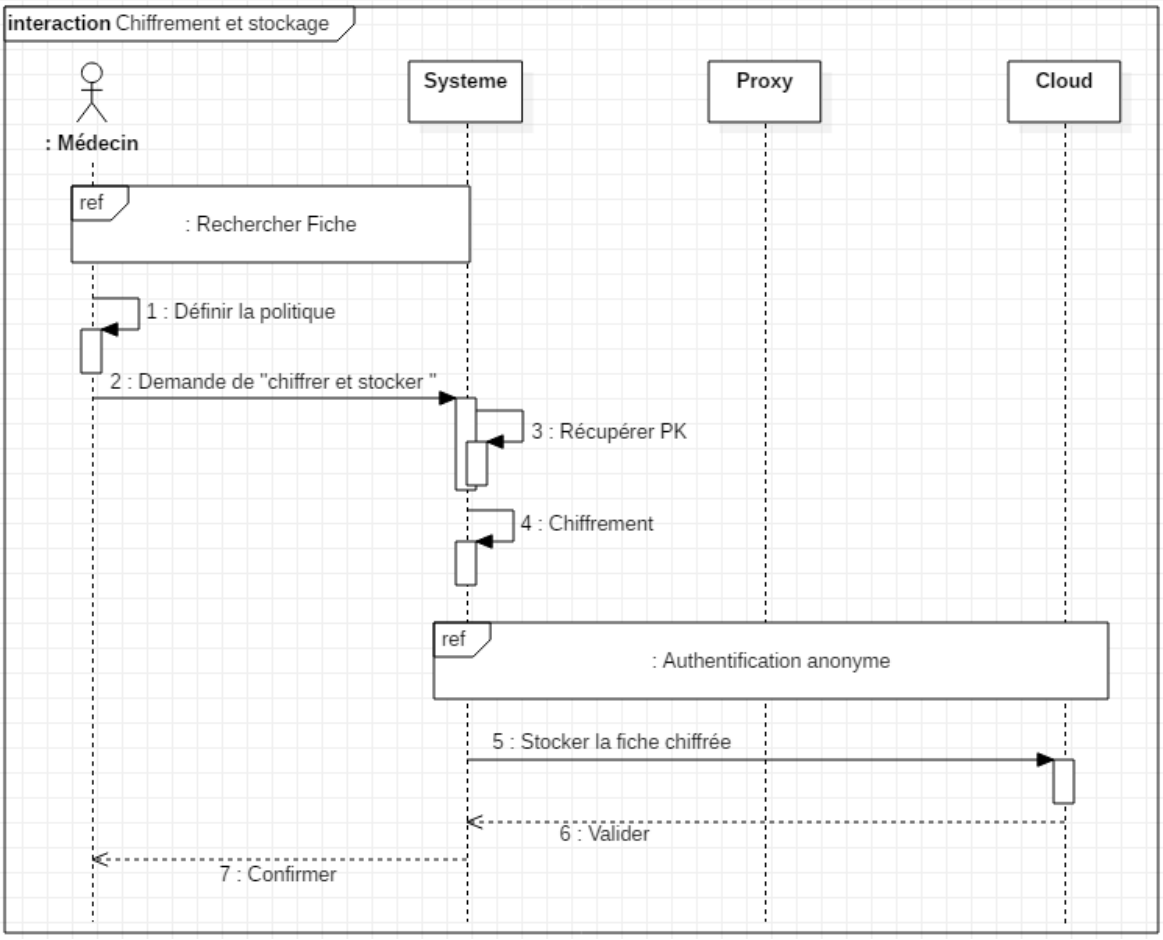

Figure 38 : diagramme de séquence ''Chiffrerement et stockage''

3.2.5 Téléchargement et Déchiffrement

L'utilisateur (médecin, patient) recherche une fiche dans le but de la télécharger et déchiffrer. Le système récupère la fiche de suivi désirée en établi une connexion anonyme avec le cloud. Ensuite, il récupère la clé SK pour déchiffrer la fiche en vérifiant si les attributs de l'utilisateur correspondent à la politique d'accès. Si oui alors le système pourra déchiffrer la fiche sinon un message d'échec sera affiché à l'utilisateur. Le diagramme de séquence suivant illustre ces étapes :

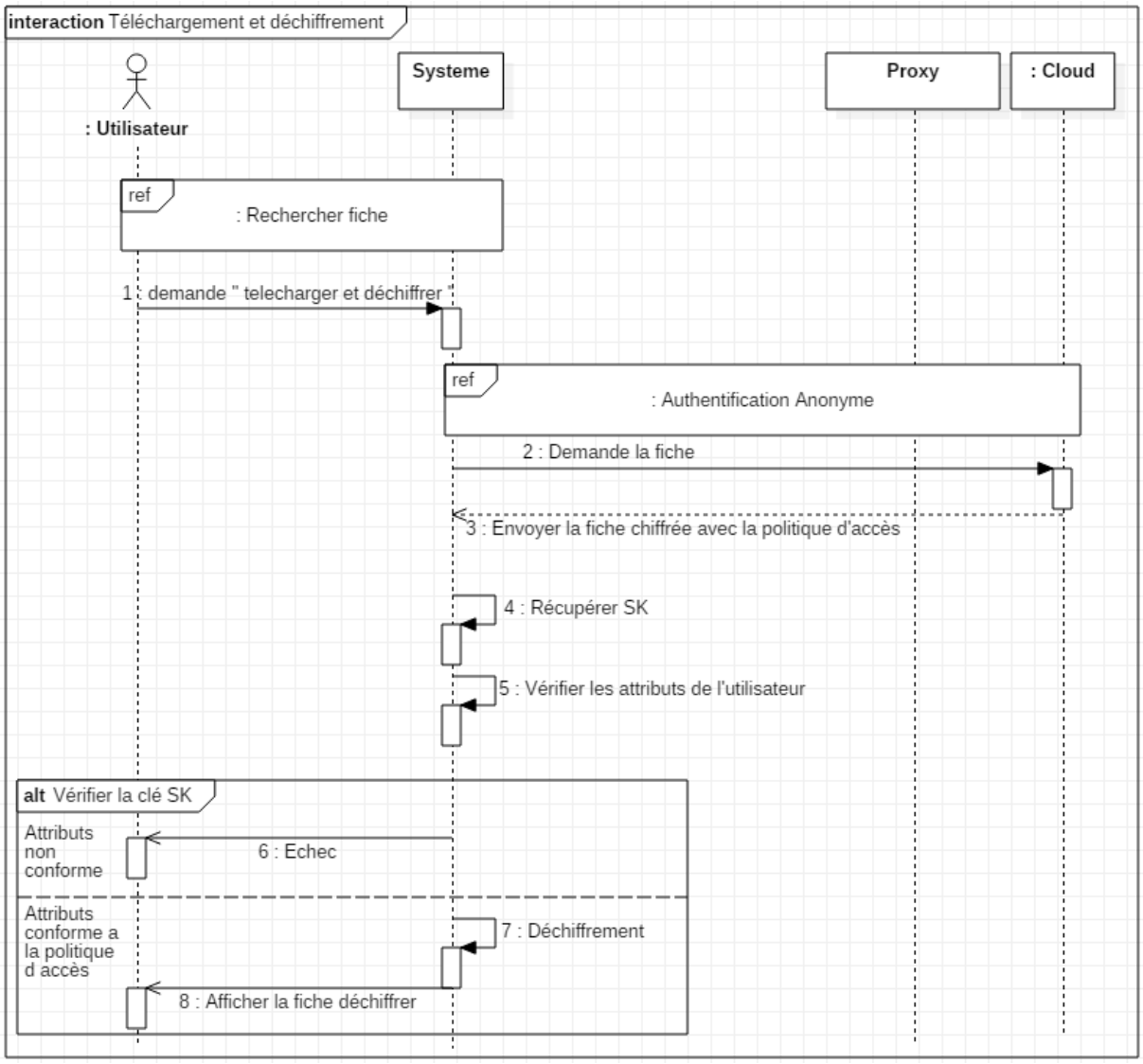

Figure 39 : diagramme de séquence ‘’Télechargement et Déchiffrement’’

3.3 Schéma relationnelle de la base de données

Les données de notre application sont stockées dans une base de données que nous allons concevoir son schéma relationnel à partir du diagramme de classe en appliquant des règles de transformation. Cela fait l’objet de cette section

3.3.1 Diagramme de classe

Notre diagramme de classe présenté dans la figure 40 décrit la structure du système en montrant les classes intervenantes et les relations entre elles : [44]

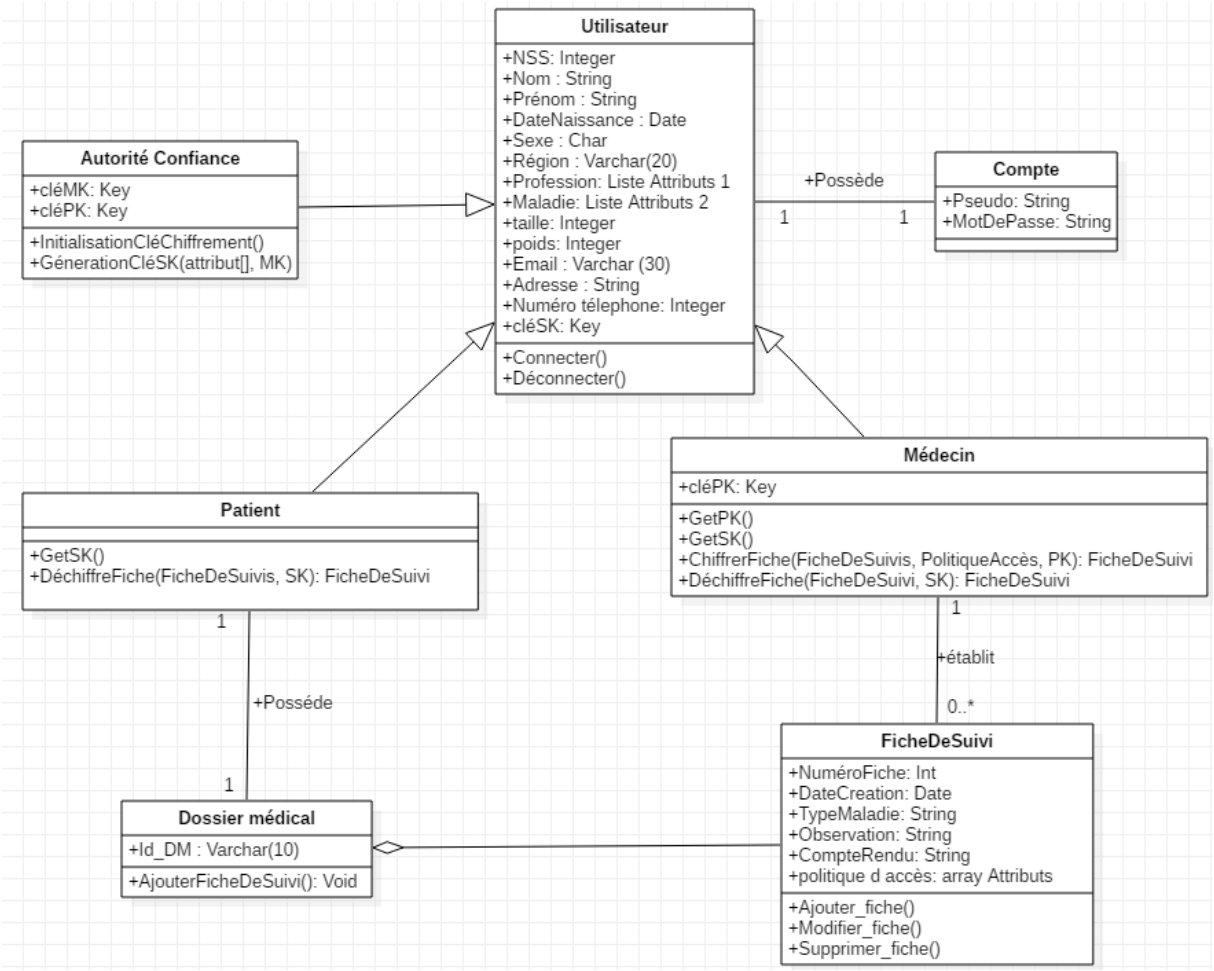

Figure 40 : diagramme de classe

3.3.2 Passage au modèle relationnel

Les règles de passage au modèle relationnel sont :

- **Transformation des classes :** chaque classe du diagramme UML devient une relation, il faut choisir un attribut de la classe pouvant jouer le rôle de clé.
- **Transformation des associations :** Nous distinguons trois familles d'associations

  - **Association 1..*:** il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association.
  - **Association *..* et n-aire et classes-association** : la classe-association devient une relation. La clé primaire de cette relation est la concaténation des identifiants des classes connectées à l'association.

- **Association 1..1 :** il faut ajouter un attribut de type clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de la classe connectée à l'association. Si les deux multiplicités minimales sont à un, il est préférable de fusionner les deux classes en une seule.

3.3.3 Schéma relationnel

En appliquant ces règles de transformation d'un diagramme de classe vers un modèle relationnel, nous avons abouti au schéma relationnel suivant :

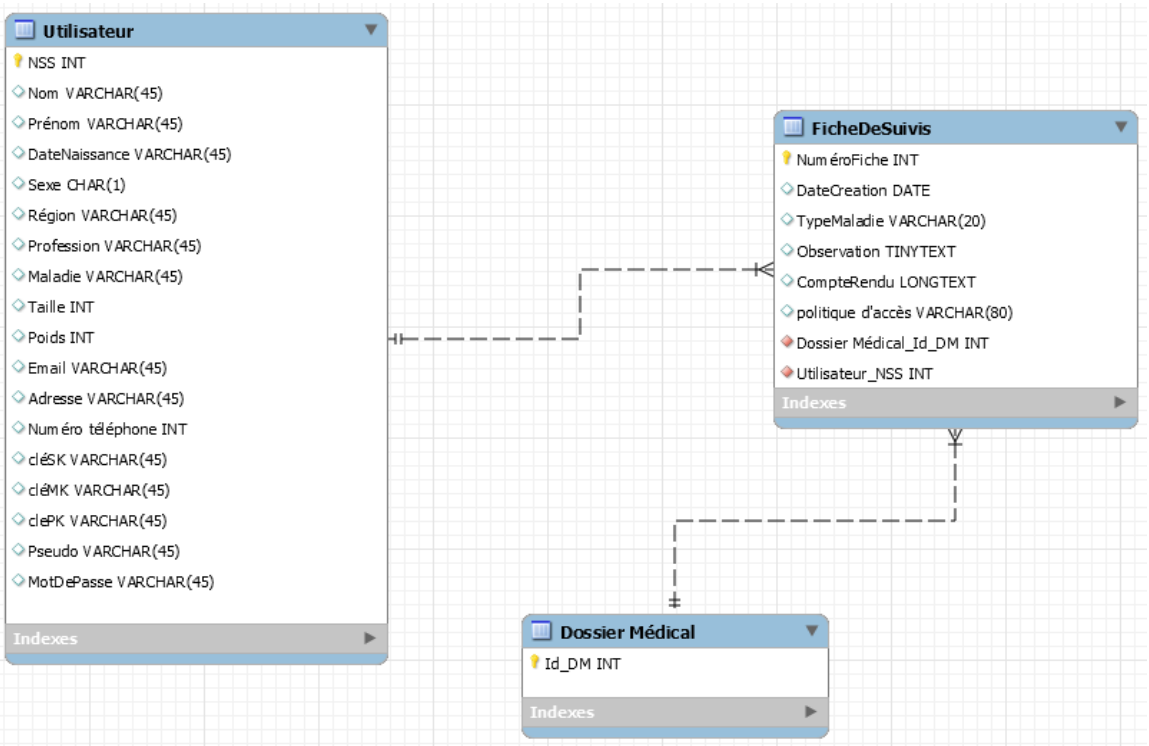

Figure 41 : schéma relationnelle

4. Conclusion

Dans ce chapitre, nous avons décrit notre approche pour un système du cloud e-santé sécurisé. L'approche se repose sur le contrôle d'accès ABAC avec chiffrement CPABE qui offre une solution efficace pour garantir le contrôle des données et sur l'authentification anonyme qui permet de garantir non seulement la confidentialité mais aussi la confiance des utilisateurs et. Pour mettre en place notre solution, nous avons conçu une application incluant toutes les fonctionnalités nécessaires. Dans le chapitre suivant, nous présenterons les étapes suivies dans l'implémentation et la réalisation de cette application.

Chapitre V : Réalisation

1. Introduction

Nous allons présenter dans ce chapitre la partie réalisation de notre application qui a pour objectif de mettre en œuvre la solution décrite dans le chapitre précèdent. Pour ce faire, nous allons commencer tout d'abord par préciser l'environnement matériel et logiciel de travail. Ensuite, nous décrivons l'implémentation des approches proposées (contrôle d'accès basé sur le chiffrement CP-ABE et authentification anonyme sans certificat). Enfin, nous présenterons les principales interfaces graphiques de notre application.

2. Environnement de développement

Un environnement de développement se réfère à une suite d'applications et d'outils que nous avons installés sur nos machines pour nous aider à développer notre application. Comme montré dans la figure 42, nous avons utilisé deux machines virtuelles pour mettre en œuvre notre système. Les machines virtuelles sont créées en utilisant Oracle VM Virtuel Box[15] avec les caractéristiques décrites dans le tableau 10. Notre système est implémenté comme une application client-serveur en utilisant le langage JAVA[16] et l'environnement de développement Netbeans[17] ainsi que ses interfaces graphiques JAVASwing[18]. Le serveur cloud est un serveur uwamp[19] qui dispose d'un SGBD (système de gestion de base de données) appelé phpMyAdmin[20] pour administrer notre base de données MySQL.

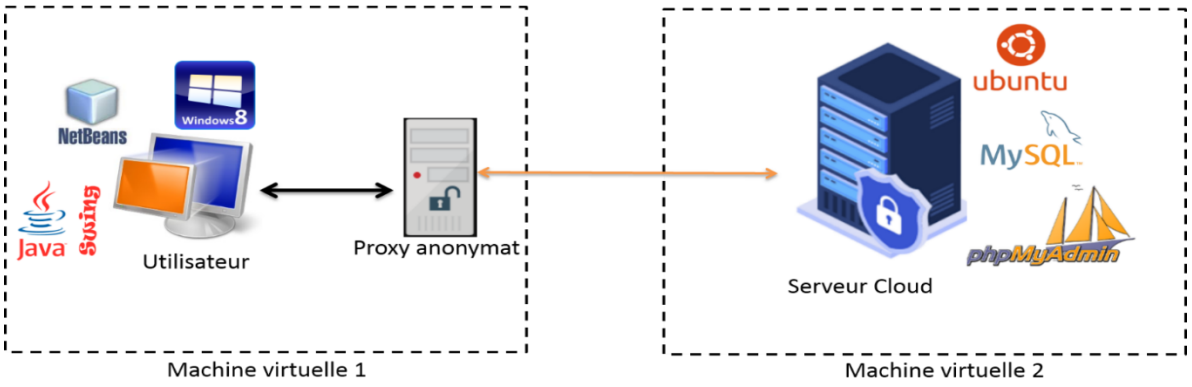

Figure 42 : Environnement de développement

[15] https://www.virtualbox.org/
[16] https://www.java.com/fr/
[17] https://netbeans.org/
[18] https://netbeans.org/features/java-on-client/swing.html
[19] https://www.uwamp.com/fr/
[20] https://www.phpmyadmin.net/

Machine virtuelle	1	2
Système d'exploitation	Windows 8	Ubuntu 18.04
RAM	8GO	1GO
Processeur	Intel® Pentium® CPU N3710 @1.60 GHz	Intel® Pentium® CPU N3710 @1.60 GHz
Acteurs présentés	Autorité de Confiance, Médecin, Patient et Proxy anonymat	Serveur cloud
Logiciels installées	MVJ, Netbeans 8.1 (JAVA, JAVASwing, DET ABE),	uWamp (MySQL, PHPMyAdmin)

Tableau 10 : Caractéristiques des machines virtuelles.

3. Implémentation

Notre application est un ensemble de package (API) JAVA comme illustré dans la figure suivante :

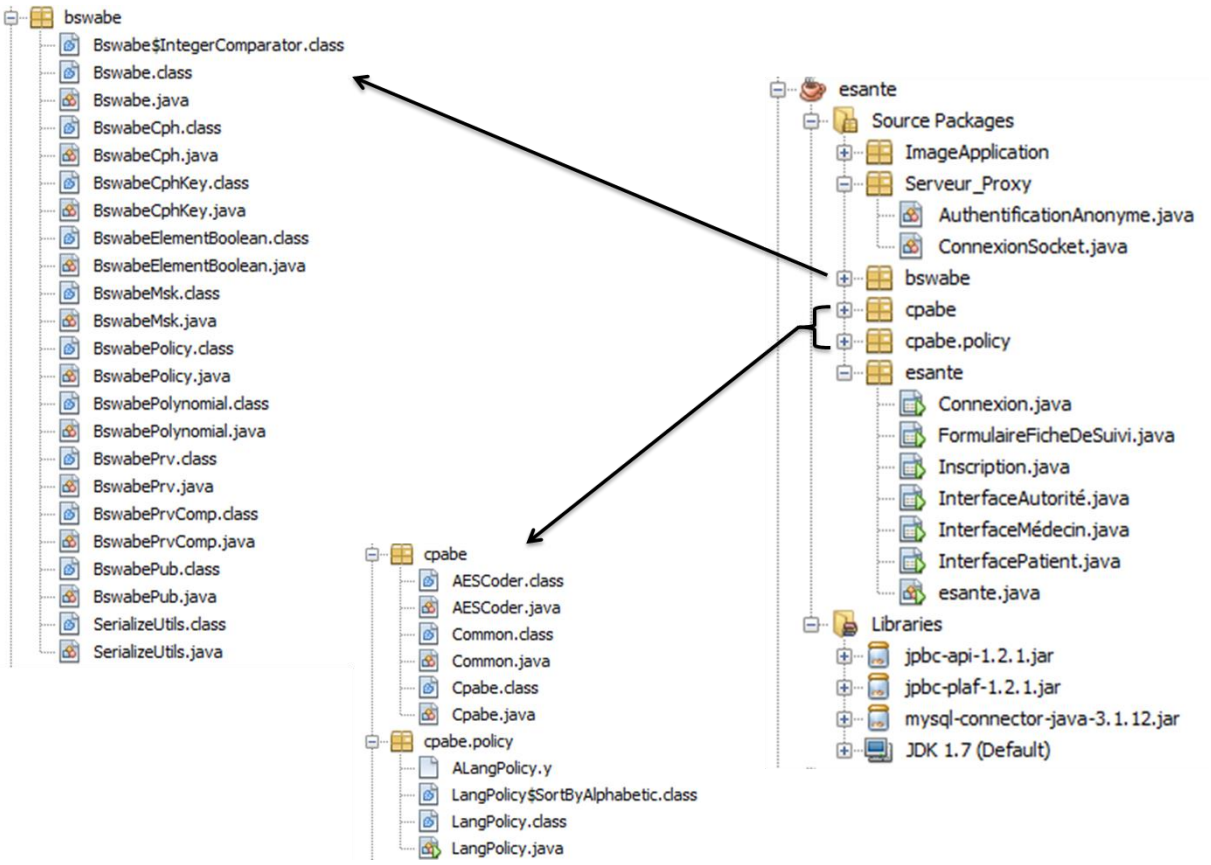

Figure 43: API de notre application

Dans ce qui suit, nous décrivons comment nous avons mis en place le chiffrement CP ABE en utilisant l'API DET ABE et comment nous avons implémenté le proxy anonymat.

3.1 Implémentation du chiffrement CP ABE

Pour la construction du chiffrement CP ABE, nous avons déployé un ensemble pratique d'outils, appelé l'API DET ABE [45] qui utilise l'API jPBC (java Pairing Based Cryptography)[46], fondée sur la bibliothèque de cryptographie PBC. Cette dernière est livrée avec un support pour les deux chiffrements symétrique et asymétrique et dispose de solides primitives cryptographique, C'est une API qui permet de cacher les opérations cryptographiques de base en offrants aux utilisateurs une souplesse de programmation.

L'API DET-ABE contient plusieurs packages, mais pour notre application, nous avons utilisé que les suivants :

- **Bswabe** qui fournit les classes concernant l'implémentation des clés MK, PK, et SK.

- **Cpabe** qui fournit les quatre fonctions de commande :
  - **cpabe-setup** pour générer une clé publique et une clé master.
  - **cpabe-keygen** pour générer une clé secrète en utilisant une clé master.
  - **cpabe-enc** pour chiffrer avec une clé publique, un fichier sous une arborescence d'une politique d'accès spécifiée dans une langue de stratégie.
  - **cpabe-dec** pour déchiffrer un fichier avec une clé privée.
- **Cpabe.policy** qui permet de définir l'arborescence de la politique en utilisant des règles de seuil qui se décrit comme « au moins attributs parmi », par exemple : au lieu de représenter un arbre avec des portes "AND" et "OR" en utilise respectivement 2 of 2 et 1 of 2 portes de seuil.

3.2. Proxy Anonymat

Nous avons créé serveur proxy qui exécute le processus d'authentification anonymat sans certificat (voir la section 2.2. du chapitre IV). La figure 44 présente les principales fonctions de la classe «AuthentificationAnonyme». Ce serveur communique avec l'utilisateur à l'aide des sockets Java, les fonctions de communications (envoi et réception) sont illustrées dans la figure 45.

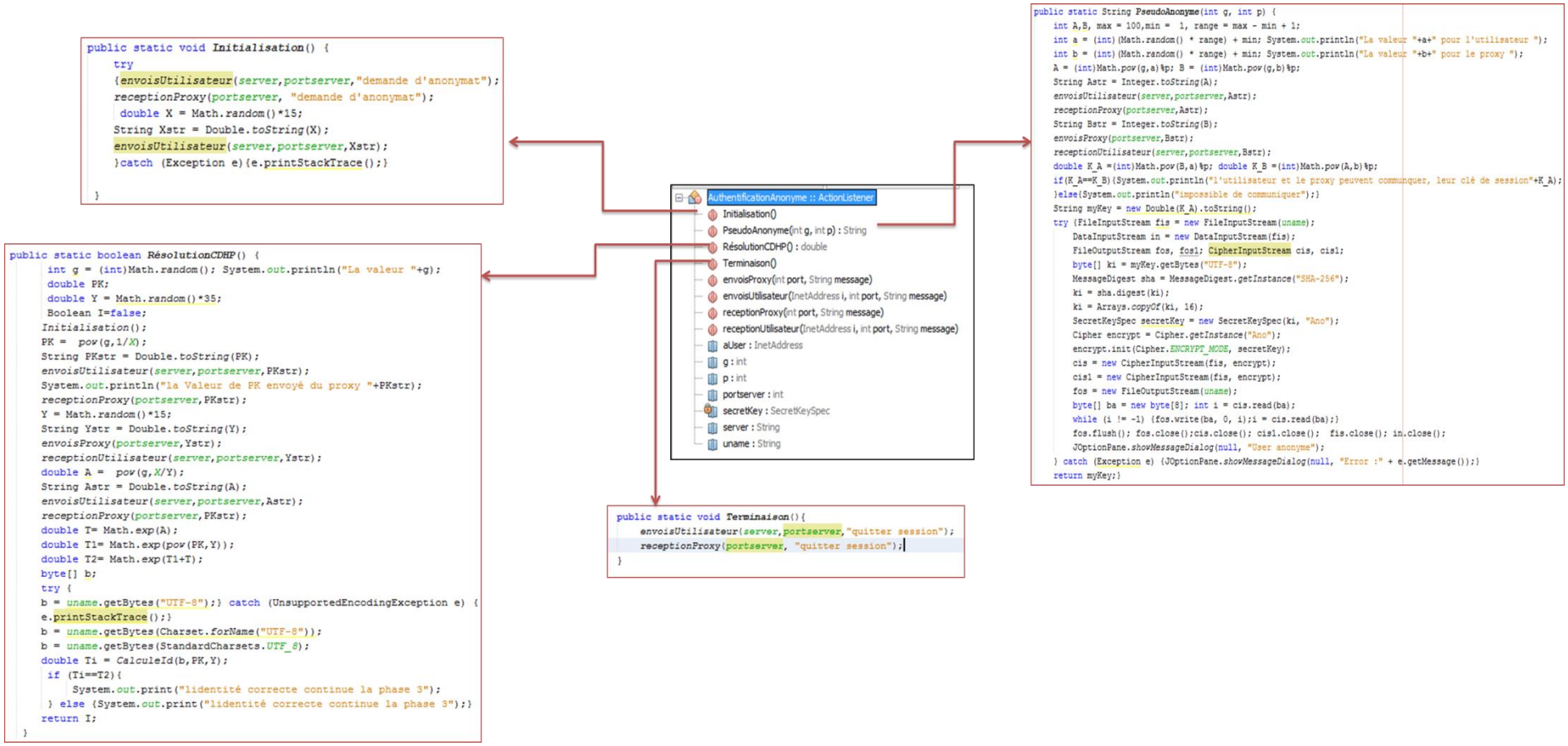

Figure 44 : Les fonctions d'Authentification Anonyme

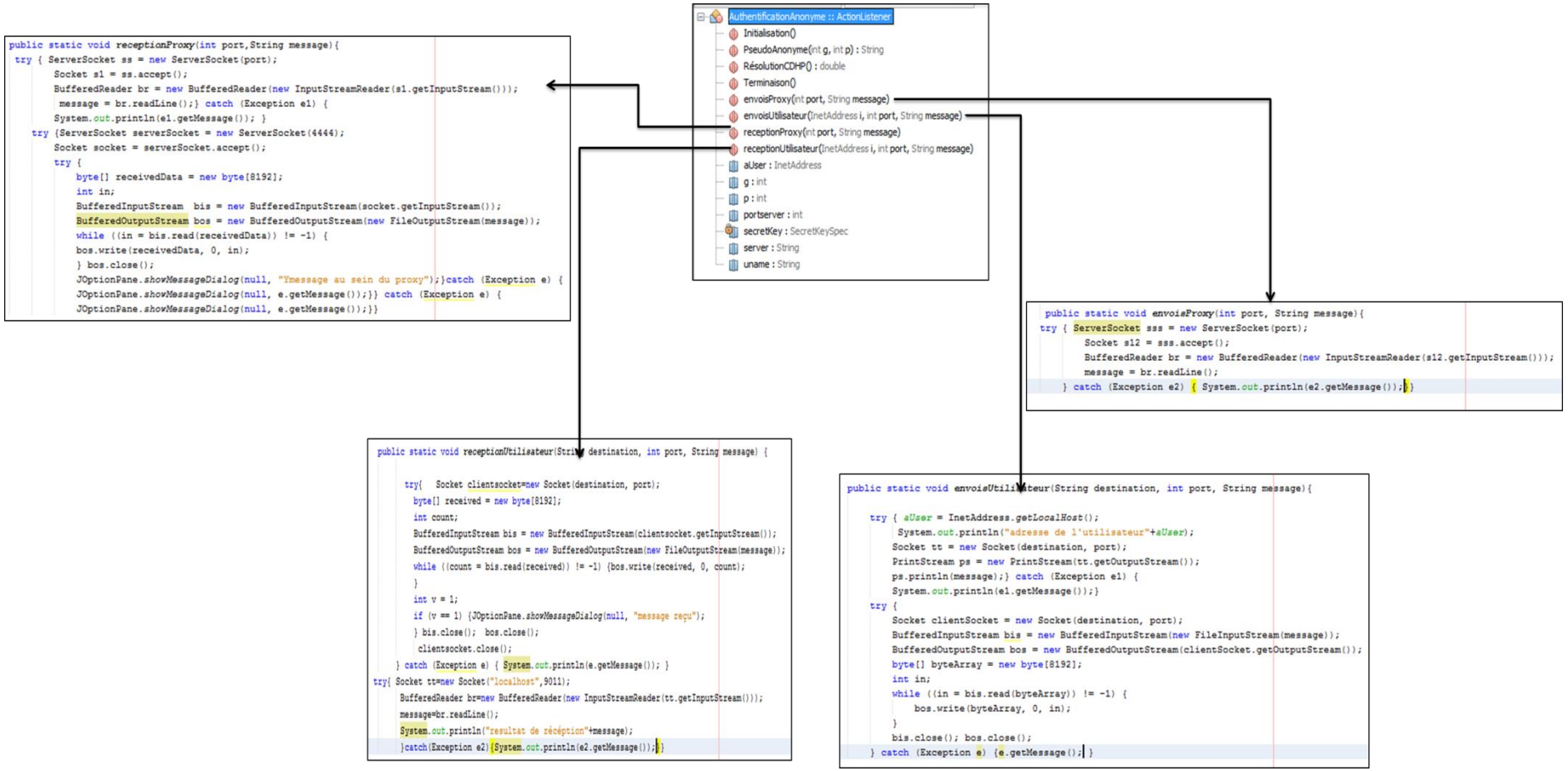

Figure 45 : Les fonctions de communication entre le proxy et l'utilisateur

4. Présentation de l'application

Au lancement de l'application, la fenêtre de connexion s'affiche (Figure 46) qui permet à un nouvel utilisateur de s'inscrire s'il ne possède pas déjà un compte, sinon de se connecter en introduisant son pseudo et mot de passe.

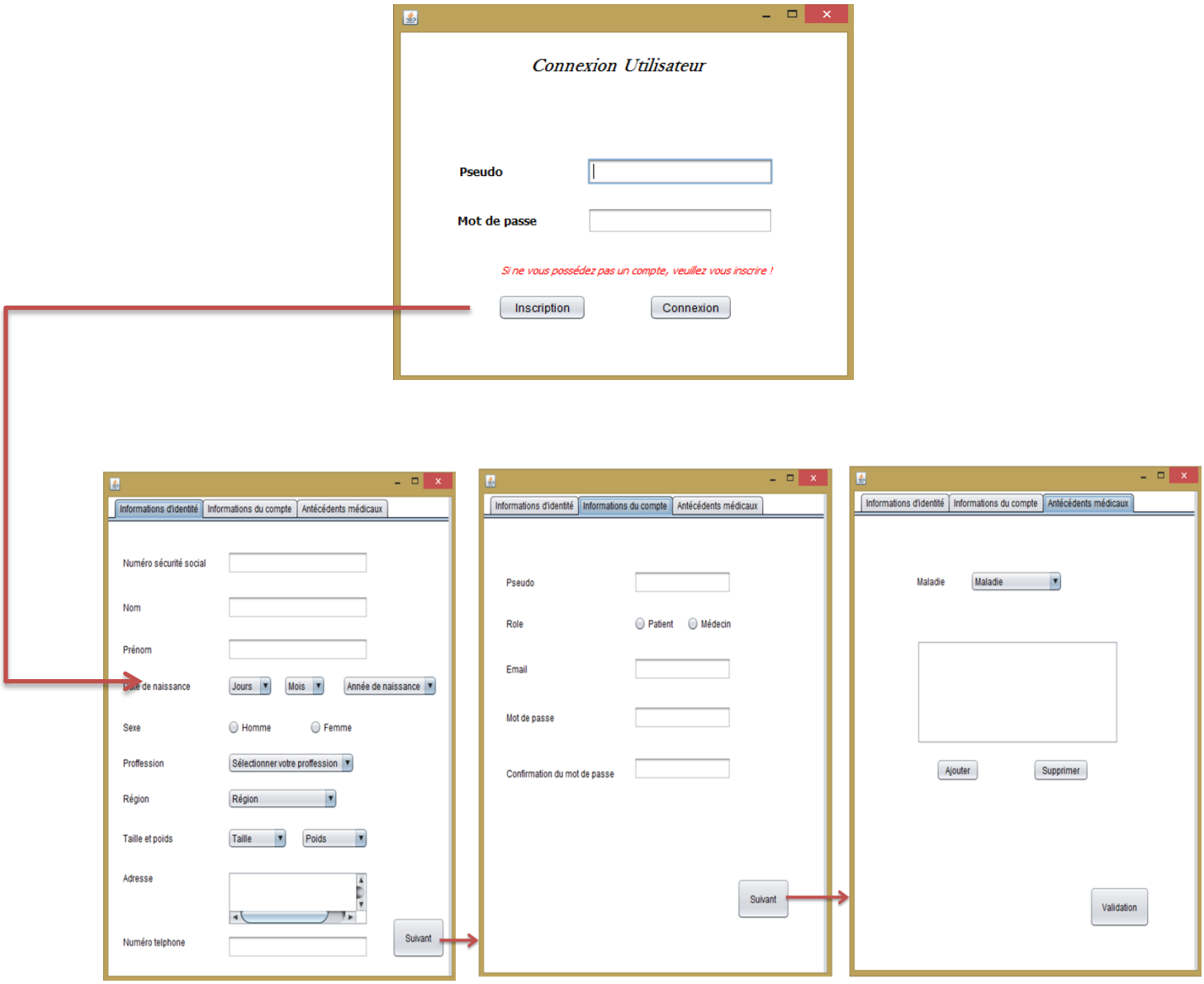

Figure 46 : Connexion/inscription d'un utilisateur

4.1 Espace administrateur

Chaque utilisateur possède son propre espace. L'espace de l'administrateur (figure 48) permet de gérer les clés (figure 47 et figure 48) et les attributs des utilisateurs.

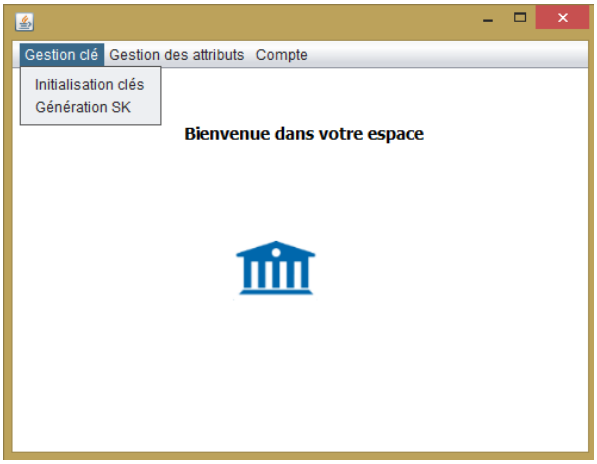

Figure 48 : Espace de l'administrateur

Figure 47 : Initialisation des clés MK et PK

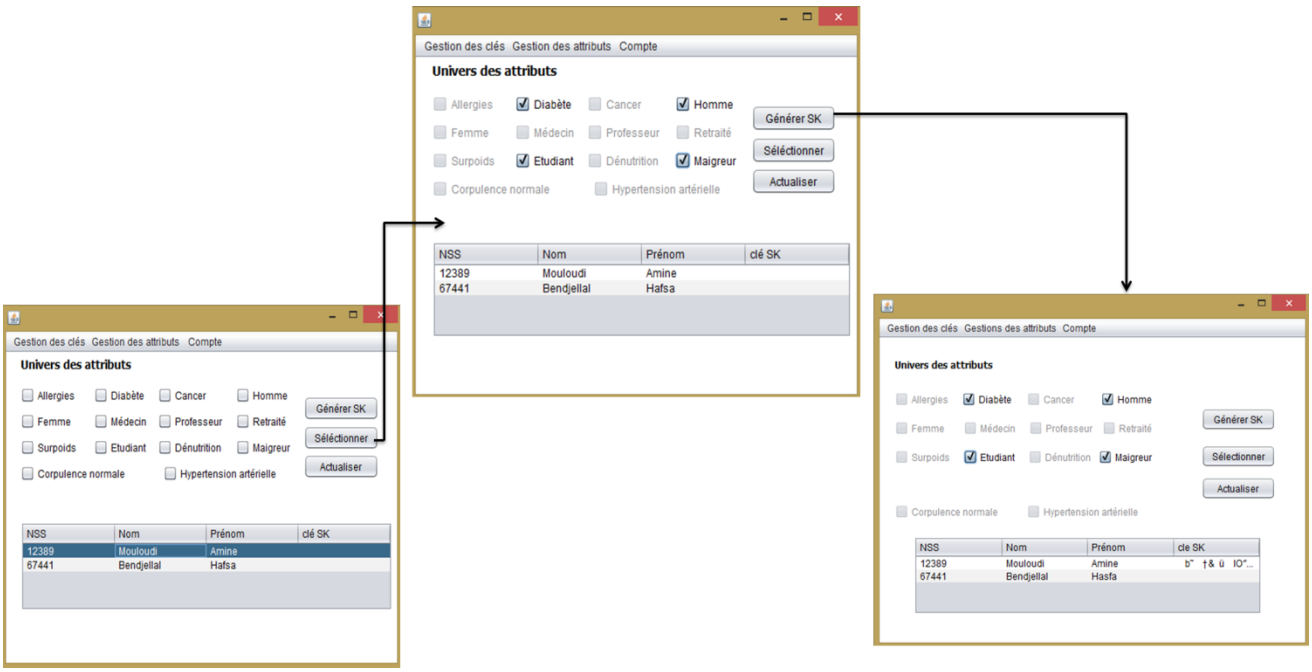

Figure 49 : Génération des clés SK

4.2. Espace Médecin

Cet espace (figure 50) permet au médecin de créer une fiche de suivi (figure 51) ensuite de la chiffrer et la stocker dans le cloud de manière anonyme (figure 52). Notons ici que dans la figure 52, l'arborescence de la politique d'accès est créée de feuilles à la racine (bas en haut) comme suit :

a. Sélectionner les attributs « Diabète » et « Etudiant » et l'opérateur AND « 2 of 2 »
b. Cliquer sur le bouton insérer
c. Sélections l'attribut « Médecin » et l'opérateur OR « 1 of 2 »
d. Cliquer sur le bouton insérer

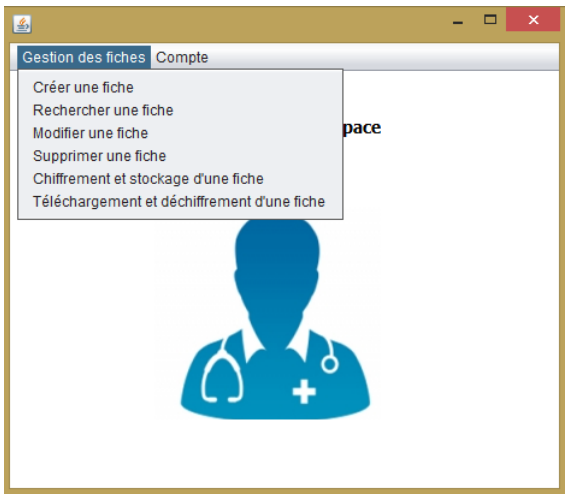

Figure 50 : Espace Médecin

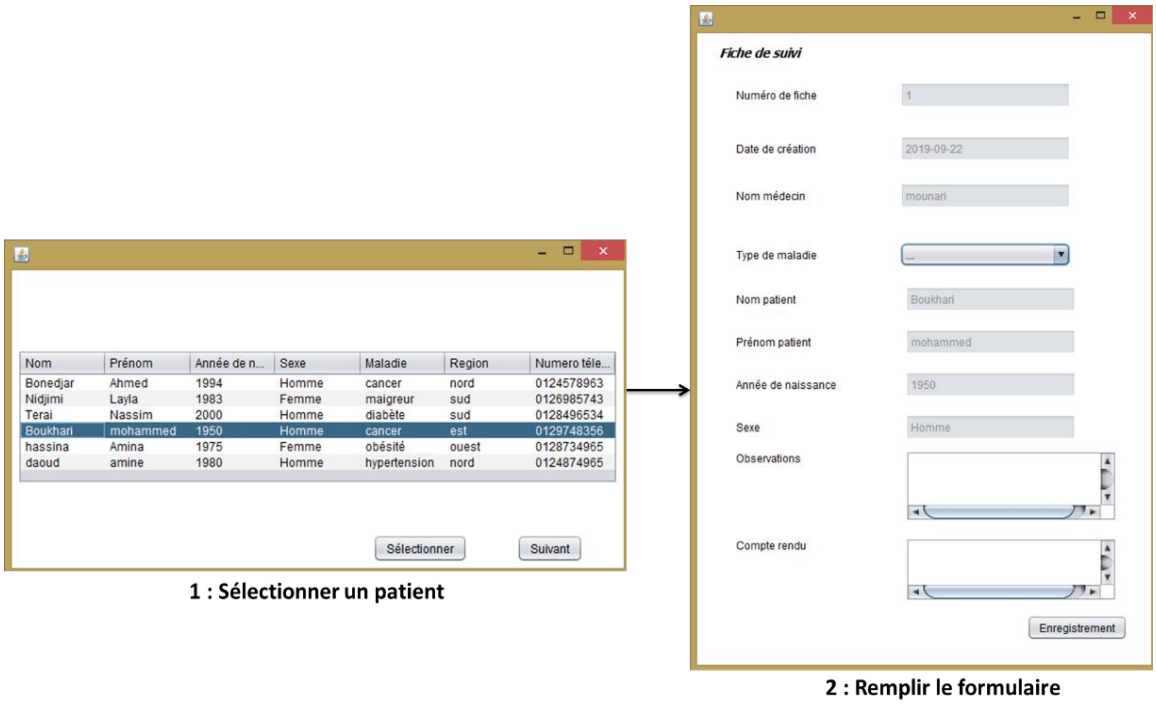

Figure 51 : Créer une fiche de suivi

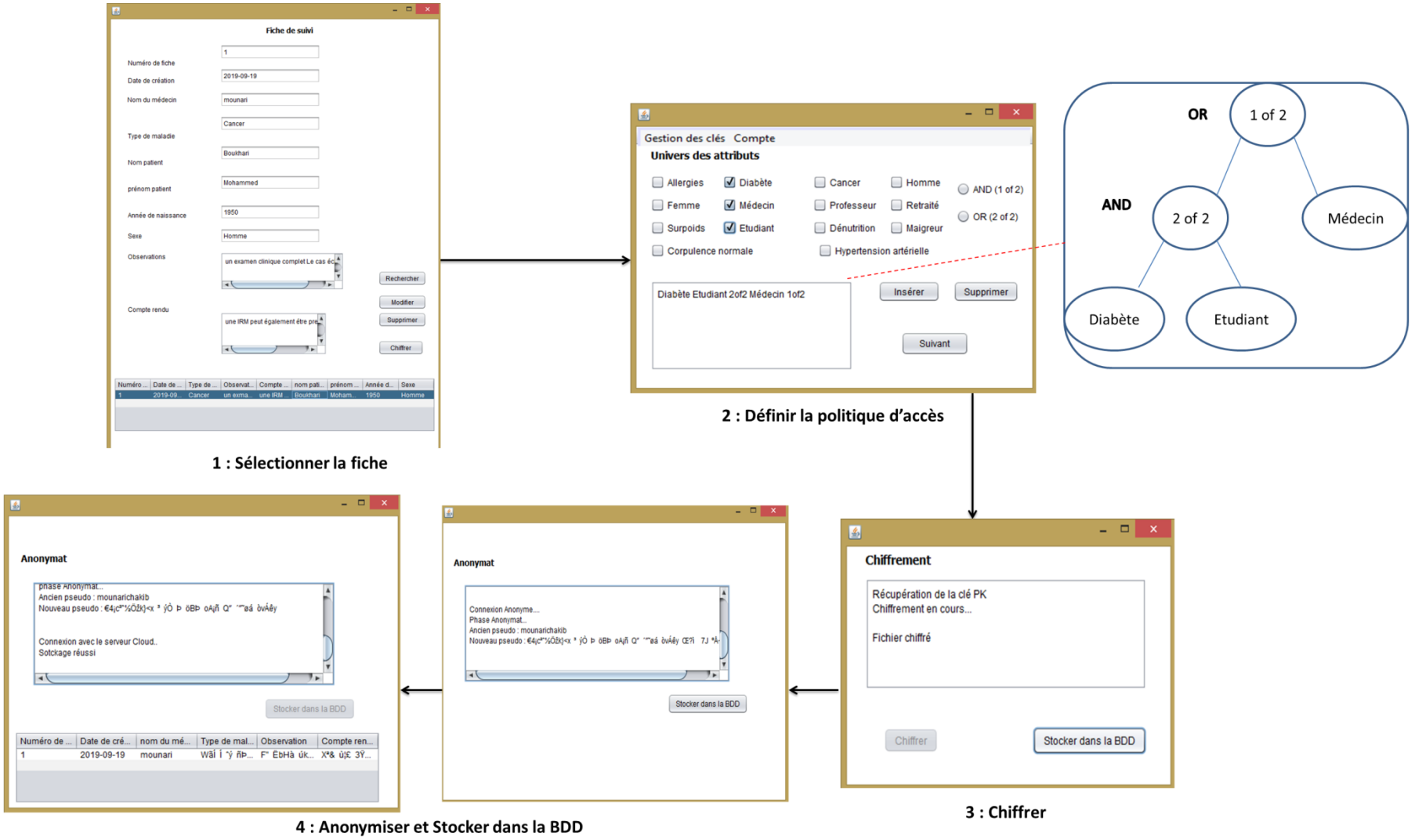

Figure 52 : Chiffrement et Stockage d'une fiche.

4.3 Espace Patient

Comme l'espace médecin, cette espace (figure 53) permet au patient de télécharger et déchiffrer une fiche illustré dans la figure 53.

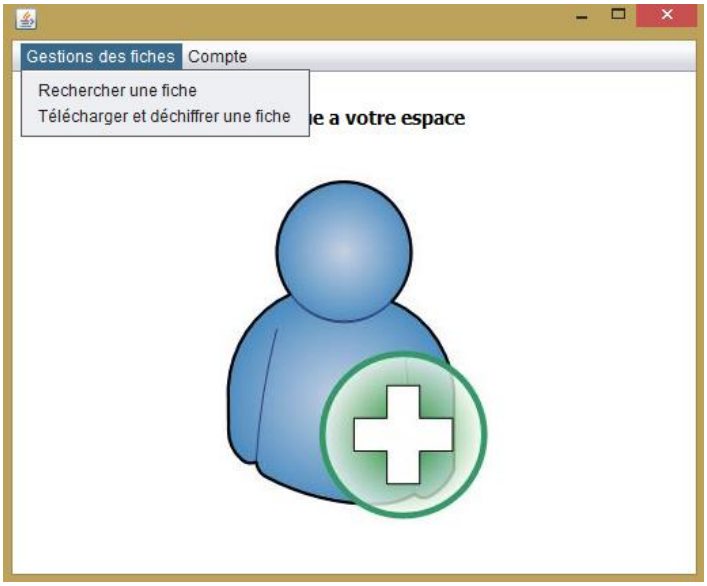

Figure 53 : Espace patient

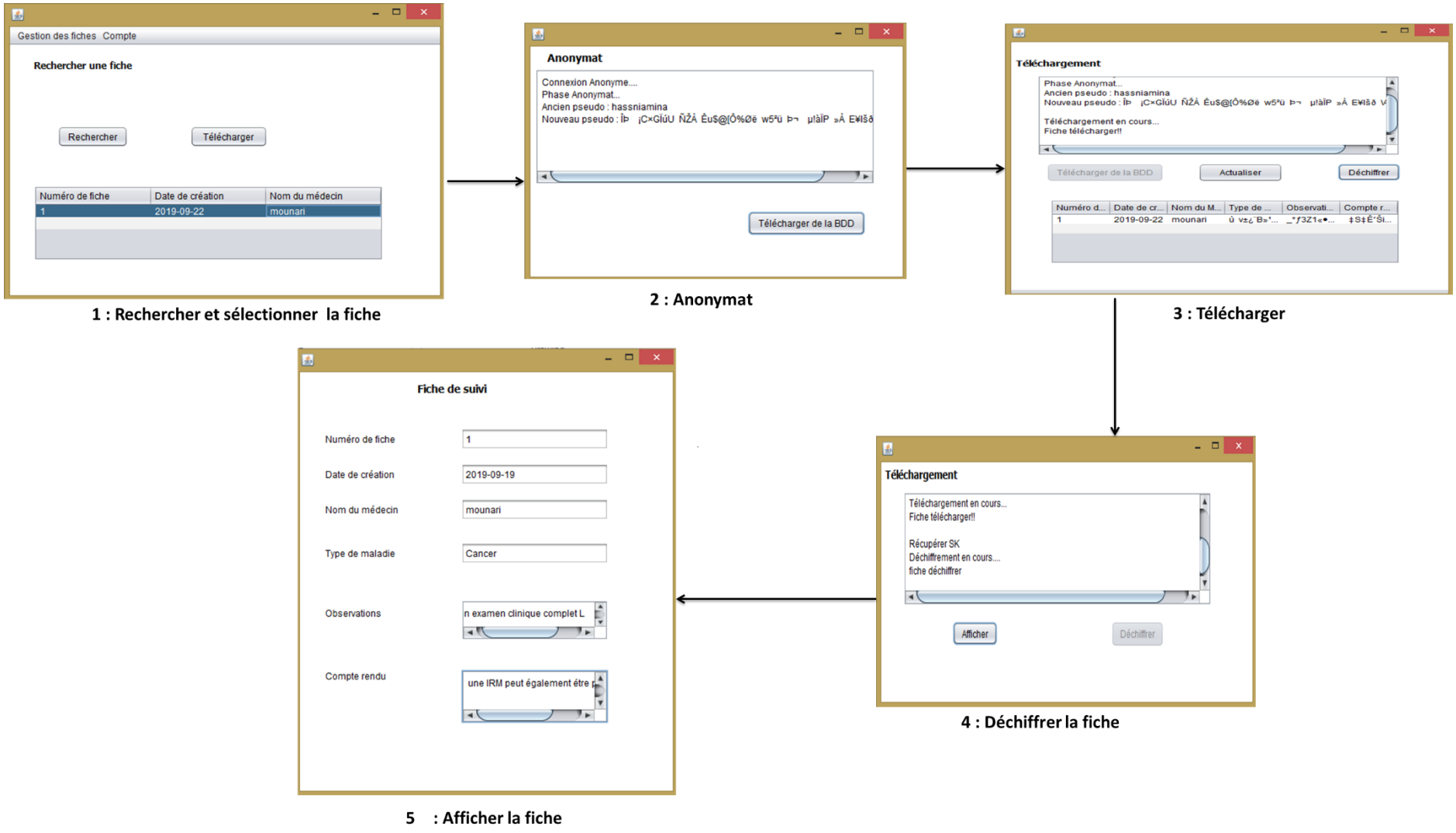

Figure 54 : Téléchargement de déchiffrement d'une fiche

5. Conclusion

Dans ce chapitre, nous avons décrit le processus de réalisation de notre application, en spécifiant les outils de développement et les librairies utilisés. Nous avons présenté les différentes interfaces graphiques qui composent les espaces des utilisateurs de notre système tout en exposant les fonctions qui constituent chaque espace. Nous avons achevé l'implémentation de notre application tout en respectant la conception élaborée.

Conclusion Générale

L'objectif visé à travers ce travail est de s'intéresser à la problématique de la sécurité des données stockées dans le cloud e-santé afin de pouvoir les partager avec les utilisateurs autorisés tout en préservant la confidentialité et leur vie privée.

Pour atteindre notre objectif, nous avons fait une étude bibliographique que nous avons répartie sur trois chapitres. Dans le chapitre 1, nous avons présenté de manière générale le cloud computing, notre domaine d'application e-santé, les menaces et les mécanismes de sécurité. Ensuite, nous avons détaillé les principales approches de sécurité qui nous intéressent : Chiffrement et contrôle d'accès dans le chapitre 2 et Anonymat et authentification dans le chapitre 3.

Après, nous avons proposé un système du cloud e-santé sécurisé qui combine deux approches de sécurité. La première approche concerne le contrôle d'accès basé sur les attributs (ABAC) et sur le chiffrement CP ABE où les clés privées de l'utilisateur sont spécifiées par un ensemble d'attributs et les données chiffrées ne peuvent être déchiffrés que par les utilisateurs autorisés par une politique d'accès à base de ces attributs. D'autre part, la deuxième approche consiste à utiliser une méthode d'authentification à nom d'utilisateur et mot de passe combiné à l'approche d'anonyme sans certificat. Cette approche permet de cacher l'identité de l'utilisateur pour préserver leurs vies privées lorsqu'il accède au cloud pour stocker ou télécharger les données chiffrées. Elle protège les informations de l'utilisateur d'être abusé par un agresseur et d'être vulnérables aux attaques malveillantes. Cette solution permet de fournir un niveau de sécurité élevé et de préserver la vie privée de l'utilisateur afin d'atteindre un niveau élevé de protection de la vie privée.

Pour mettre en place notre solution, nous avons implémenté nos deux approches en une application java en utilisant une librairie pour le chiffrement CP ABE (API DET ABE) et en écrivant en code java l'algorithme de l'authentification anonyme sans certificat. Notre application permet :

- à l'utilisateur de s'authentifier par nom d'utilisateur et mot de passe,
- à l'autorité de confiance de gérer les attributs des patients et les clés de chiffrement et de déchiffrement,
- au médecin de créer, rechercher, modifier et supprimer des fiches de suivis. Ces dernières sont chiffrées avant d'être stockées dans le cloud de manière anonyme

- au patient de rechercher et télécharger ses fiches de suivis de manière anonyme avant de les déchiffrer.

Faute du temps, nous avons résumé l'environnement cloud en un serveur stockage de BDD. Il aurait judicieux de créer un serveur cloud en utilisant OuwCloud [47] par exemple, et de faire le stockage de manière transparente en établissement une connexion par socket entre notre application java et le serveur OwnCloud.

Par ailleurs, notre solution de sécurité soulève un certain nombre de questions ouvertes intéressantes telles que :

- Concernant le modèle de contrôle d'accès par chiffrement CP-ABE :

- **Définition des attributs pertinents** : qui doit être faite avec l'aide des professionnels du domaine.
- **Sécurité de la politique d'accès** : la politique d'accès est sauvegardé en clair, elle doit être chiffrée elle aussi pour empêcher un utilisateur malveillant de l'utiliser et déchiffrer les fiches de suivis des autres patients.
- **Evolution de la gestion des clés** : Utilisation des plusieurs autorités de domaine qui seront gérés par une autorité de confiance ; chaque autorité de domaine gère les clés des utilisateurs qui appartiennent à son domaine. Elle doit coordonner avec l'autorité de confiance.
- **Intégration d'autres algorithmes de chiffrement** : comme le chiffrement homomorphe qui permet d'effectuer des types spécifiques de calculs sur du texte crypté et de générer un résultat crypté qui, une fois décrypté, correspond au résultat des opérations effectuées sur le texte en clair.
- **Révocation (Annulation) des droits d'accès de l'utilisateur** : pour accéder aux données d'où l'annulation de la clé de déchiffrement. La révocation des attributs utilisées dans le modèle TAAC [29], semble être efficace et flexible car elle n'affecte pas les privilèges de déchiffrement pour les autres attributs possédés par cet utilisateur et n'affecte pas les autres utilisateurs qui possèdent cet attribut révoqué.

- Concernant l'authentification anonyme :

- **Utilisation d'autres méthodes d'authentification :** Pour renforcer la sécurité de la méthode d'authentification par nom utilisateur et mot de passe, il est nécessaire d'utiliser une combinaison de techniques d'authentification et ainsi un autre facteur comme la carte à puce comme la carte Chiffa ou empreinte digitale. Il serait également intéressant d'utiliser le

protocole SRP (Secure Remote Password) pour réaliser une authentification asymétrique à l'aide de mots de passe.

- **Amélioration du niveau de sécurité d'anonymat** : en utilisant d'autres pseudonymes (comme adresse IP) combinés à des mécanismes de sécurité.
- **Intégration d'autres approches d'anonymat** : comme l'approche communication anonyme avec le protocole SSH afin d'assurer le transfert sécurisé des données chiffrées d'un serveur à un autre. De plus, le routage d'oignon permet de chiffrer un message à plusieurs reprises au cours de son envoi sur le réseau.

Enfin, les problèmes de sécurité dans un environnement cloud demeurent toujours des problèmes ouverts en conséquence beaucoup de pistes restent à explorer.

Bibliographie

[1] P. Mell *et al.*, “Google Application Engine Introduction,” *Futur. Gener. Comput. Syst.*, vol. 25, no. 6, p. 17, 2011.

[2] S. P. Ec, “Le cloud privé et ses avantages métiers : des coûts réduits et une réactivité accrue,” *Emc*, 2010.

[3] P. Farhen, “Community Cloud Computing with association of cloud computing paradigm : A study,” International Journal of Advanced Research in Computer Science and Software Engineering, volume 4, no. 5, pp. 1484–1485, 2014.

[4] Desai, J, (2013, 21 Janvier). “Comprendre le Cloud Computing 4 Modèles de déploiement.” Consulté le 03/06/2019 sur http://cloudcomputinginfrench.blogspot.com/2013/01/comprendre-le-cloud-computing-4-modeles.html.

[5] D. Jiménez Martínez, “Privacy and confidentiality issues in cloud computing architectures,” mémoire de master, unisersitat politecnica de catalunya, Espagne, pp. 142, 2014.

[6] B. Yüksel, A. Küpçü, and Ö. Özkasap, “Research issues for privacy and security of electronic health services,” *Futur. Gener. Comput. Syst.*, vol. 68, pp. 1–13, 2017.

[7] M. Chopra, J. Mungi, and K. Chopra, “Cloud computing is the delivery of computing services over the Internet.,” International Journal of Science, Engineering and Technology Research (IJSETR), volume 2, no. 2, pp. 480–488, 2013.

[8] Kreamer, M, “Psytechsolutions Advancing with you”. *EMR vs EHR vs PHR*. Consulté le 27/06/2019 sur https://www.psytechsolutions.net/emr-vs-ehr-vs-phr.” .

[9] T. TagElsir Ahmed Osman, A. A. babiker, and N. Mustafa, “Internal & External Attacks in cloud computing Environment from confidentiality, integrity and availability points of view,” *IOSR J. Comput. Eng. Ver. V*, vol. 17, no. 2, pp. 2278–661, 2015.

[10] M. S. Mahmoud, Y. Xia, “Cloud Computing,” *Networked Control Systems*. pp. 91–125, 2019.

[11] P. Grange, *Syntec numérique* “Livre Blanc - SÉCURITÉ DU CLOUD COMPUTING,”, p. 24, 2010.

[12] J. Blanc, A. De Georges, “Technique de cryptographie”, mémoire licence informatique, France, pp. 1–30, 2004.

[13] Y. Challal, H. Bettahar, “Introduction à la sécurité informatique.”, pp. 1–52, 2008 disponible sur : https://moodle.utc.fr/pluginfile.php/16778/mod_resource/content/0/Intro-securite.pdf

[14] H. Benzenine “ Principe de base de la cryptographie. Introdction,” mémoire de master pp. 24–36, 2014.

[15] B. Debbagh, N. Bounegeb, “Etude et comparaison de principaux système crypto Fournis par le package de Bouncy Castle Plate forme Java SDK ” 2016.

[16] N. Döttling and S. Garg, “Identity-based encryption from the diffie-hellman assumption,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10401 LNCS, pp. 537–569, 2017.

[17] Y. O. Yahia and P. Paradinas, “Applications e-santé , le contrôle des données personnelles un enjeu majeur pour la protection de la vie privée,” École Polytechnique de Montréal, Canada, 2017.

[18] M. Ambrosin *et al.*, “On the Feasibility of Attribute-Based Encryption on Internet of Things Devices,” *IEEE Micro*, vol. 36, no. 6, pp. 25–35, 2016.

[19] A. Rajeev, T. S. Kenu, R. Arun, and S. S. Babu, “A Survey on Attribute Based Encryption Schemes for Data Sharing,” *Journal of Computer Engineering*, pp. 19–23, 2016.

[20] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” *Proc. ACM Conf. Comput. Commun. Secur.*, pp. 89–98, 2006.

[21] J. Bethencourt, *A. Sahai, B. Waters*, “Ciphertext-Policy Attribute-Based Encryption,” IEEE Symposium on Security and Privacy (SP ’07), May 2007, Berkeley, France. 10.1109/SP.2007.11 . hal-01788815

[22] P. V. Kumar and J. A. R. Aluvalu, “International Journal of Innovative and Emerging Research in Engineering Key Policy Attribute Based Encryption (KP-ABE): A Review,” *Int. J. Innov. Emerg. Res. Eng.*, vol. 2, no. 2, pp. 49–52, 2015.

[23] Scheafer, C, (2017, 5 Mai) “Understanding the difference between Physical Access Control and Logical Access Control controle d acces physique et logique.” Consulté le 12/06/2019 sur http://www.mintcontrols.com/understanding-the-difference-between-physical-access-control-and-logical-access-control/ .

[24] M. Schuetz, (2018, 18 Avril) “Why Move Your Access Control To The Cloud.” Consulté le 12/03/2019 sur https://www.getkisi.com/blog/move-access-control-cloud-

9-benefits-cloud-access-control-systems.

[25] I. Indu, P. M. R. Anand, and V. Bhaskar, “Identity and access management in cloud environment: Mechanisms and challenges,” *Eng. Sci. Technol. an Int. J.*, vol. 21, no. 4, pp. 574–588, 2018.

[26] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, “An Efficient File Hierarchy Attribute-Based Encryption Scheme in Cloud Computing,” *IEEE Trans. Inf. Forensics Secur.*, vol. 11, no. 6, pp. 1265–1277, 2016.

[27] L. Nawal, “Conception d'une Architecture distribuée de controle d'accès avec la détection et la prévention de clients intrus dans le cloud,” mémoire de magistére, université de bejaia, Algérie, 2014.

[28] S. Ruj, A. Nayak, and I. Stojmenovic, “DACC: Distributed access control in clouds,” *Proc. 10th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. Trust. 2011, 8th IEEE Int. Conf. Embed. Softw. Syst. ICESS 2011, 6th Int. Conf. FCST 2011*, pp. 91–98, 2011.

[29] K. Yang, Z. Liu, Z. Cao, X. Jia, D. S. Wong, and K. Ren, “TAAC: Temporal Attribute-based Access Control for Multi-Authority Cloud Storage Systems.,” *IACR Cryptol. ePrint Arch.*, vol. 2012, p. 651, 2012.

[30] A. Bates, B. Mood, M. Valafar, and K. Butler, “Towards secure provenance-based access control in cloud environments,” *CODASPY 2013 - Proc. 3rd ACM Conf. Data Appl. Secur. Priv.*, pp. 277–284, 2013.

[31] P. Pleva, “A Revised Classification of Anonymity,” 2012.

[32] D. Cerezo Sánchez, “Zero-Knowledge Proof-of-Identity: Sybil-Resistant, Anonymous Authentication on Permissionless Blockchains and Incentive Compatible, Strictly Dominant Cryptocurrencies,” *SSRN Electron. J.*,pp. 1-54, 2019.

[33] P. F. Syverson, D. M. Goldschlag, and M. G. Reed, “Anonymous connections and onion routing,” *IEEE J. Sel. Areas Commun.*, vol. 16, no. 4, pp. 482–494, 1997.

[34] L. Guo, C. Zhang, J. Sun, and Y. Fang, “PAAS: A privacy-preserving attribute-based authentication system for eHealth networks,” *Proc. - Int. Conf. Distrib. Comput. Syst.*, pp. 224–233, 2012.

[35] B. Nguyen, “La pseudonymisation,” *Stat. société*, vol. 2, no. 4, pp. 43–50, 2014.

[36] J. Miracle and M. Cheatham, “Semantic web enabled record linkage attacks on anonymized data,” *CEUR Workshop Proc.*, vol. 1750, 2016.

[37] L. Sweeney, “A model for protecting privacy 1,” *Ieee Secur. Priv.*, vol. 10, no. 5, pp. 1–14, 2002.

[38] D. Pandya, K. Ram, S. Thakkar, T. Madhekar, and B. S. Thakare, “An Overview of Various Authentication Methods and Protocols,” *Int. J. Comput. Appl.*, vol. 131, no. 9, pp. 25–27, 2015.

[39] S. Milad Dejamfar and S. Najafzadeh, “Authentication Techniques in Cloud Computing: A Review,” *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 7, no. 1, pp. 95–99, 2017.

[40] “auth attack.” .

[41] A. Djellalbia, N. Badache, S. Benmeziane, and S. Bensimessaoud, “Anonymous authentication scheme in e-Health Cloud environment,” *2016 11th Int. Conf. Internet Technol. Secur. Trans. ICITST 2016*, pp. 47–52, 2017.

[42] Z. H. Zhang, J. J. Li, W. Jiang, Y. Zhao, and B. Gong, “An new anonymous authentication scheme for cloud computing,” *ICCSE 2012 - Proc. 2012 7th Int. Conf. Comput. Sci. Educ.*, no. Iccse, pp. 896–898, 2012.

[43] J. Lopez, R. Oppliger, and G. Pernul, “Classifying public key certificates,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3545 LNCS, pp. 135–143, 2005.

[44] A. Anwar, S. Ebersold, B. Coulette, M. Nassar, and A. Kriouile, “Vers une approche à base de règles pour la composition de modèles. Application au profil VUML,” *International journal of Science and research.* volume. 13, no. 4. pp. 73–103, 2007.

[45] Miguel Morales-Sandoval and Arturo Diaz-Perez, “DET-ABE: A Java API for Data Confidentiality and Fine-Grained Access Control from Attribute Based Encryption,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9311, pp. 20–35, 2015.

[46] A. De Caro and V. Iovino, “jPBC: Java Pairing Based Cryptography,” *Proc. - IEEE Symp. Comput. Commun.*, pp. 850–855, 2011.

[47] The ownCloud developers, “ownCloud User Manual,” p. 68, 2015 disponible sur https://doc.owncloud.org/server/9.1/ownCloud_User_Manual.pdf.